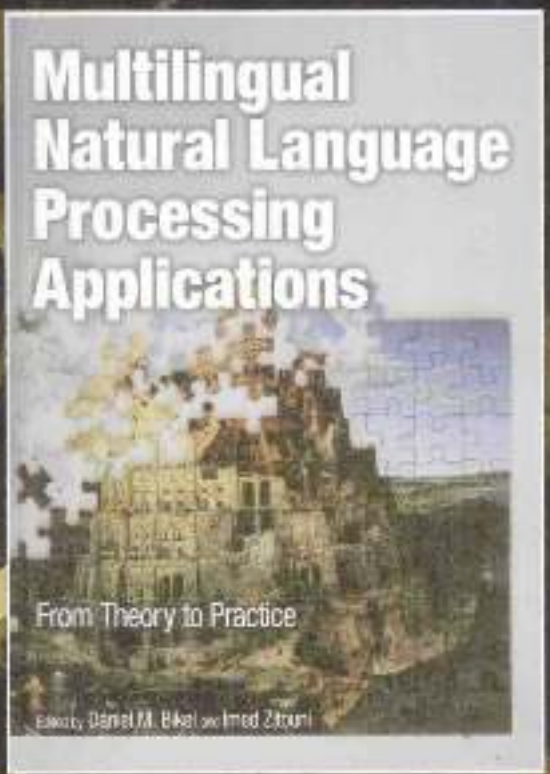
计 算 机 科 学 丛 书

PEARSON

多语自然语言处理
从原理到实践

（美）Daniel M. Bikel Imed Zitouni 编
史晓东 陈毅东 等译

Multilingual Natural Language Processing Applications
From Theory to Practice



机械工业出版社
China Machine Press



多语自然语言处理 从原理到实践

Multilingual Natural Language Processing Applications From Theory to Practice

本书是第一本全面阐述如何构建健壮和准确的多语自然语言处理系统的图书，由两位资深专家编辑，集合了该领域众多尖端进展以及从广泛的研究和产业实践中总结出的实用解决方案。

第一部分介绍现代自然语言处理的核心概念和理论基础，展示了如何理解单词和文档结构、分析语法、建模语言、识别蕴涵和检测冗余。第二部分彻底阐述与构建真实应用有关的实际考量，包括信息抽取、机器翻译、信息检索、文摘、问答、提炼、处理流水线等。

作者简介

Daniel M. Bikel 现为Google公司高级研究科学家，正在开发用于自然语言处理和语音识别的新方法。在IBM工作期间，他为IBM的GALE多语种信息抽取和自动应答系统构架了拦截系统。在宾夕法尼亚大学攻读博士后期间，他建造了第一个可扩展的多语种语法分析引擎。

Imed Zitouni 现为微软公司高级研究员。2004~2012年，他是IBM公司高级研究科学家，领导IBM公司的阿拉伯语信息抽取和数据资源工作组。在此之前，他还曾领导DIALOCA的语音/自然语言处理组和Bell实验室/阿尔卡特朗讯的语言建模和呼叫路由工作。他的研究涉及机器翻译、自然语言处理和口语对话系统。





PEARSON

ISBN 978-7-111-48491-2

定价：99.00元

PEARSON
www.pearson.com

投稿热线：(010) 88379604
客服热线：(010) 88378991 88361066
购书热线：(010) 68326294 88379649 68995259
华章网站：www.hzbook.com
网上书店：www.china-pub.com
数字阅读：www.hzmedia.com.cn

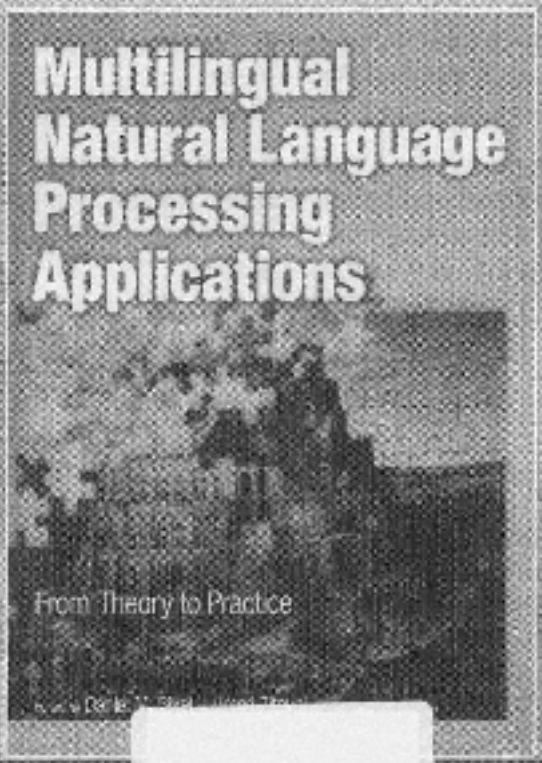
计 算 机 科 学 丛

多语自然语言处理

从原理到实践

(美) Daniel M. Bikel Imed Zitouni 编
史晓东 陈毅东 等译

Multilingual Natural Language Processing Applications
From Theory to Practice



TP391
D94

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

多语自然语言处理：从原理到实践 /（美）比凯尔（Bikel, D. M.），（美）兹图尼（Zitouni, I.）编；史晓东等译．—北京：机械工业出版社，2015.1
（计算机科学丛书）
书名原文：Multilingual Natural Language Processing Applications: From Theory to Practice

ISBN 978-7-111-48491-2

I. 多… II. ① 比… ② 兹… ③ 史… III. 自然语言处理：研究 IV. TP391

中国版本图书馆 CIP 数据核字（2014）第 262220 号

本书版权登记号：图字：01-2013-0217

Authorized translation from the English language edition, entitled *Multilingual Natural Language Processing Applications: From Theory to Practice*, 9780137151448 by Daniel Bikel, Imed Zitouni, published by Pearson Education, Inc. publishing as IBM Press, Copyright © 2012 International Business Machines Corporation.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Chinese simplified language edition published by Pearson Education Asia Ltd., and China Machine Press Copyright © 2015.

本书中文简体字版由 Pearson Education（培生教育出版集团）授权机械工业出版社在中华人民共和国境内（不包括中国台湾地区和中国香港、澳门特别行政区）独家出版发行。未经出版者书面许可，不得以任何方式抄袭、复制或节录本书中的任何部分。

本书封面贴有 Pearson Education（培生教育出版集团）激光防伪标签，无标签者不得销售。

本书全面阐述了自然语言处理的多个方面，涵包括形态学、文档分割、句法、语义分析、语言模型、监督推理、情感分析等理论部分，也包括文体检测、关系识别、机器翻译、信息检索、自动文摘、问答系统、对话系统、多引擎处理等实践部分。本书内容丰富，不仅引用了很多最新的文献，而且还展示了从广泛的研究和产业实践中总结出来的实用解决方案。

本书可供广大的自然语言处理研究者和开发者参考。

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码：100037）
责任编辑：姚 蕾 刘立卿　　责任校对：殷 虹
印　　刷：北京市汉德彩色印刷有限公司　　版　　次：2015 年 2 月第 1 版第 1 次印刷
开　　本：185mm × 250mm 1/16　　印　　张：29.5
书　　号：ISBN 978-7-111-48491-2　　定　　价：99.00 元

凡购本书，如有缺页、倒页、脱页，本社发行部调换
客服热线：(010) 88378991 88361066　　投稿热线：(010) 88379604
购书热线：(010) 68326294 88379649 68995259　　读者信箱：hzjsj@hzbook.com

版权所有 · 侵权必究
封面无防伪标均为盗版
本书法律顾问：北京大成律师事务所 韩光 / 邹晓东

文艺复兴以来，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了突飞猛进的发展；也正是这样的优势，使美国在信息技术发展的六十多年间名家辈出、领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅奠定了研究的范畴，还揭示了学术的源流，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切，这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短的现状下，美国等发达国家在其计算机科学发展的几十年间积淀和发展的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起到积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章公司较早意识到“出版要为教育服务”。自 1998 年开始，我们就将工作重点放在了遴选、移译国外优秀教材上。经过多年的不懈努力，我们与 Pearson，McGraw-Hill，Elsevier，MIT，John Wiley & Sons，Cengage 等世界著名出版公司建立了良好的合作关系，从他们现有的数百种教材中甄选出 Andrew S. Tanenbaum，Bjarne Stroustrup，Brain W. Kernighan，Dennis Ritchie，Jim Gray，Alfred V. Aho，John E. Hopcroft，Jeffrey D. Ullman，Abraham Silberschatz，William Stallings，Donald E. Knuth，John L. Hennessy，Larry L. Peterson 等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及珍藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力相助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专门为本书的中译本作序。迄今，“计算机科学丛书”已经出版了近两百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍。其影印版“经典原版书库”作为姊妹篇也被越来越多实施双语教学的学校所采用。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证。随着计算机科学与技术专业学科建设的不断完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都将步入一个新的阶段。我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。华章公司欢迎老师和读者对我们的工作提出建议或给予指正。我们的联系方法如下：

华章网站：www.hzbook.com
电子邮件：hzjsj@hzbook.com
联系电话：(010) 88379604
联系地址：北京市西城区百万庄南街 1 号
邮政编码：100037

华章教育
华章科技图书出版中心

译者序

Multilingual Natural Language Processing Applications: From Theory to Practice

本书对自然语言处理的多语言相关课题做了深入的研究，内容丰富，引用了很多最新的文献，对广大的自然语言处理研究者和开发者来说，这是一本非常好的参考书。

全书分为理论和实践两部分，理论部分有7章，实践部分有9章，各章可单独阅读。下面对各章内容进行简要评述，以供读者参考。

第1章主要讨论形态学，重点关注阿拉伯语等屈折语的形态处理。该章提到了汉语的分词问题，但是没有任何描述，另外还讨论了很有意思的形态归纳问题。

第2章主要讨论文档结构，包括句子边界检测、话题边界检测，主要讨论了基于特征的机器学习方法，对语音的分割也进行了讨论。

第3章讨论了句法分析，涉及到内容丰富而具体。

第4章讨论了语义分析，是本书篇幅最大的一章，内容非常详尽，从各类语义问题描述、资源、方法到具体系统，应有尽有。

第5章讨论了语言模型，介绍各种先进的语言模型，有很多最新的内容和文献可供读者参考，阅读该章需在理解了n元模型的基础上进行。

第6章讨论了文本蕴涵识别，提出了一个文本蕴涵框架，介绍了各类文本蕴涵算法和系统及其性能评测，提供了很多相关资源。

第7章讨论了情感和主观性分析，强调了举荐（bootstrapping）方法的使用（特别是跨语言举荐）。

第8章讨论了提及检测和共指消解，这是两个信息抽取中的基本问题。该章写得非常简明扼要，而且提供了一种实现。

第9章讨论了关系抽取和事件抽取，也属于信息抽取的范畴。该章探讨了机器学习的方法，并提倡将实体检测和关系抽取结合在一个模型里。

第10章讨论了机器翻译及其现状、评测与各种模型。

第11章讨论了信息检索，内容翔实，特别区分了跨语言信息检索和多语言信息检索。

第12章讨论了自动文摘，对其历史、方法、评测、系统构造、工具、多语问题都有细致的描述。自动文摘也可以看作是信息抽取问题。

第13章讨论了问答系统，对涉及的实现技术和相关算法都进行了详细的描述。问答系统也可以看作是高级的信息抽取。

第14章讨论了提炼，这是介于信息检索和问答系统的一类新兴问题，需要融合多个信息源的知识。

第15章讨论了口语对话系统，包括其体系结构、技术和方法，以及实现中的一些问题。

第16章讨论了自然语言处理的多引擎融合，包括其常见的体系结构，并在GALE项目背景下讨论了一个详细案例。

虽然本书的目的之一是在基础知识方面尽量完整，读者不需要为了自然语言处理基本任务去看很多书，但是，对于已经有自然语言处理基础的读者而言，本书提供了很多最新的研究内容，其参考文献和提供的大量可下载资源的链接非常有价值，省去了读者很多宝贵的时间。对自然语言处理系统的研发者，特别是信息抽取和信息检索相关的开发者，本书是非常好的参考。

译者在翻译时全书尽量采用统一的术语，并且采用浅显的译法来帮助读者理解，然而语言学和

自然语言处理方面的术语迄今还有很多不如意之处，因此可能仍然不能使读者看其言而知其义。

本书由多人翻译，基本上每人一章，翻译人员按照章节顺序分别为史晓东、谭波、徐伟、陈毅东（其中黄哲煌翻译了4.5.2节）、黄萌刚、林达真、苏劲松、胡金铭、何中豪、邹昌兴、方瑞玉、罗凌、张志健和方瑞玉、甘星超、王晓苏、曹茂元。校对工作由史晓东、陈毅东、谭波等多人参加。全书由史晓东统校，对存在的翻译错误负主要责任。出版社的王春华老师对本书的翻译给出了很多指导性意见，特此感谢。

由于译者水平有限，翻译时间也很仓促，译文中肯定还存在不少错误，欢迎读者批评指正，以便将来修订。译者的联系地址为 mandel@xmu.edu.cn。

前 言

Multilingual Natural Language Processing Applications: From Theory to Practice

最近几乎每个人都在一定程度上受到了信息技术的发展和互联网繁荣的影响。近来，多媒体信息逐变得日益普及，不过，未加工的自然语言文本的总量在不断增长，并且地球上各种主要语言都在不断产生大量未处理文本。例如，英语维基百科报导已有 101 种语言的维基百科，而每种语言至少有 10 000 篇文章。因此，不管是国家、公司，还是个人，都迫切需要来分析、翻译、综合或者提炼这些海量文本。

以前，要开发鲁棒、精确的多语自然语言处理（Natural Language Processing，NLP）应用，研究者或者开发人员需要查阅若干本参考书、几十个期刊或者会议论文。本书旨在为开发此类应用提供所需的所有背景知识和实际建议，虽然这个要求很高，但我们希望本书至少是本有用的参考书。

过去 20 年来，自然语言研究者开发了可处理多种语言的大量文本的若干优秀算法。迄今为止，主流的方法是建立可从实例中学习的统计模型，这样的模型能鲁棒地应对其处理文本的类型甚至语言的变化。如果设计适当，同样的模型可用于新的领域或新的语言，只需要提供相应领域或语言的新的训练实例。这种方法也使得研究者没有必要辛苦地写出处理问题的所有规则以及这些规则联合使用的方式。统计系统一般只要研究者提供可能的输入特征的抽象表示，其相对重要性可在训练（training）阶段学习而得，并在解码（decoding）或者推理（inference）阶段应用于新的文本。

统计自然语言处理领域在快速变化，部分变化源于其快速发展。例如，该领域的主要会议之一是计算语言学年会，其参会人数在过去五年已经翻番。另外，IEEE 语音和语言处理会议和期刊上自然语言处理的文章数目也在过去十年中翻了一番以上。IEEE 是世界上推进技术发展的最大的专业学会之一，自然语言处理研究者不但在解决本领域的问题上取得了内在的进步，也从机器学习和语言学领域的进展中借鉴良多。本书虽注意先进的算法和技术，但主要目的是对该领域的最佳实践进行详尽的阐明。另外，每章会描述所述方法在多语（multilingual）环境下的适用性。

本书分成两部分。第一部分是理论，包括前七章，展示了自然语言处理的各种基础问题以及解决这些问题的算法。头二章关注的是找出各种不同粒度层次的语言结构。第 1 章引入了一个重要概念——形态学（morphology），研究词的结构，以及世界上各种语言的不同形态现象的处理方法。第 2 章讨论了多种方法，文档可由此分解为更易处理的部分，如句子，以及通过主题联系的更大的单位。第 3 章研究了发现句子内部结构的方法，也即句法（syntax）。句法一直都是语言学最重要的研究领域，这种重要性也反映在自然语言处理领域。说其重要，部分原因是句子的结构和句子的意义相关，所以找出句法结构是理解句子的第一步。

找出句子或者其他文本单位的结构化的意义表示，经常称作语义分析（semantic parsing）。这是第 4 章的内容。第 4 章还特别讨论了近年来引起诸多关注的语义角色标注（semantic role labeling）问题，其目的是找出可作为动词或谓词的论元的句法短语，对动词的论元进行了识别和分类。我们离生成句子的逻辑形式（logical form）又靠近了一步，而逻辑形式是句子意义的一种表示，这种表示方式容易被机器处理，而用于处理基础的多种工具人类自古代就开始研究了。

然而，如果我们不需要语义分析生成的深层句法语义结构呢？如果我们的问题只是确定多个句子中哪个句子是人最可能写或者说的呢？解决此问题的一种方法是开发一个可根据语法合法性而为句子打分的模型并以此选取分值最高的句子。给出一个词串的分值或概率估计的问题称为语言模型（language modeling），这是第 5 章的主题。

表示意义和判断句子的语法合法性只是处理语言前期步骤中的两种。为了进一步理解意义，我

们需要一个算法，该算法可对一段文本中表示的事实进行推理。例如，我们想要知道一个句子中提到的事实是否被文档中前面的某个句子所蕴涵。这种推理被称为识别文本蕴涵（recognizing textual entailment），这是第 8 章的主题。

找出陈述或事实的相互蕴涵显然对文本自动理解很重要，但是这些陈述的来源也有待考究。理解一个陈述是否是主观的，并找出其表述的意见的倾向性是第 7 章的主题。由于人们经常表达意见，这显然是一个重要的问题，尤其在社交网络已经成为互联网上人际交流的最重要形式的时代，这一点更显重要。本书第一部分以本章作结。

本书第二部分是实践，讲述如何将第一部分描述的自然语言处理基础技术应用于现实世界中的问题。应用开发经常要做权衡，如时间和空间的权衡，因此本书应用部分的章节探讨了在构建一个鲁棒的多语自然语言处理应用时，如何进行各种算法和设计决策的权衡。

第 6 章描述识别和区分命名实体（named entity）以及这些实体在文本中提及的办法，也描述了识别同个以上的实体提及共指（corefer）的方法。这两个问题一般称为提及检测（mention detection）和共指消解（coreference resolution），它们是一个重大的应用领域——信息抽取（information extraction）的两个核心部分。

第 9 章继续信息抽取的讨论，探索找出两个实体如何发生关系的技术，也称为关系抽取（relation extraction）。要识别事件，并对此进行分类，称为事件抽取（event extraction）。此外，事件涉及多个实体，我们希望机器能找出事件的参与者及其所起的作用。因此，事件抽取与自然语言处理中的一个关键问题“语义角色标注”紧密相关。

第 10 章描述自然语言处理领域中最古老的问题之一，这本质上也是一个多语自然语言处理问题：机器翻译（Machine Translation，MT）。从一种语言翻译为另外一种语言，一直是 NLP 研究追求的目标。在学术界几十年的努力之后，近年来已经研究出多种方法，在现有的硬件条件下可以进行实用的机器翻译了。

翻译文本是一回事，但是我们如何理解现存的海量文本呢？第 8、9 章对帮助我们自动产生文本中信息的结构化记录进行了一些探索。解决海量问题的另一个办法是通过查找与某个搜索查询相关的少量文档或者文档的一部分来缩小范围。该问题称为信息检索（information retrieval），这是第 11 章的主题。像 Google 一样的商用搜索引擎在很多方面可看作大规模的信息检索系统。由于搜索引擎非常流行，因此这是个很重要的 NLP 问题——考虑到有大量语料是非公开的，从而不能被商业引擎搜索到，所以信息检索越发重要。

处理大量文本的另一个办法是自动文摘，这是第 12 章的主题。摘要很困难。一般有两种做法：找到若干个句子或句子片段来表示文本的大意；理解文本，将其意义进行某种内部表示，然后生成摘要，与人为的操作一样。

人们经常倾向于使用机器自动处理文本，因为他们有很多问题要找到答案。这些问题可以是简单的事实性问题，如“约瑟·肯尼迪何时出生”，也可以是复杂的问题，如“德国巴伐利亚的最大城市是哪个”。第 13 章讨论如何建造自动回答这类问题的系统。

如我们想回答的问题还更复杂那该怎么办？我们的查询可能有多个答案，如“找出奥巴马总统在 2013 年会见的外国政府首脑”。这类查询可由在 NLP 中被称为提炼（distillation）的一门较新的子学科处理。提炼需要真正地把信息检索和信息抽取技术结合起来，同时还要增加自己的技术。

在许多情形下，我们希望机器能利用语音识别和合成技术交互式地处理语言。这样的系统称为对话系统（dialog system）。这在第 15 章讨论。由于在语音识别、对话管理和语音合成方面的技术进展，对话系统越来越实用，并且已经在实际场合中广泛安装使用。

最后，我们作为 NLP 研究者和工程师，希望用世界上开发的大量不同的部件来构造系统。这种

处理引擎的聚合在第 16 章介绍。虽然这是本书的最后一章，但从某种意义上来讲这代表处理文本的开始而非结尾，因为该章描述了一个通用的架构，可用来生成不同组合的一系列处理流水单元。

我们希望本书是有益的，同样希望读者将其作为学习的开始而不是结束。每章都有大量参考文献，读者可以用来继续深入研究任何话题。NLP 的研究队伍在全世界越来越壮大，我们希望你加入我们的行列，一起进行自动文本处理的激动人心的探索。你可以在大学、研究所、会议、博客甚至社交网络上和我们一起交流。多语自然语言处理系统的未来是十分光明的，我们期待你的贡献！

致谢

写作本书伊始，我们就将它定位为多个作者通力合作的成果。我们对 IBM 出版社/Prentice Hall 在起步阶段给予的鼓励和支持怀有无限的感激，特别要感谢 Bernard Goodwin 和所有其他在 IBM 出版社工作的员工，他们在项目的开展和结束过程中给予了帮助。这样一本书当然也离不开我们各章节作者大量的时间，努力和技术才能的投入，所以我们非常感谢 Otakar Smrž、Hyun-Jo You、Dilek Hakkani-Tür、Gokhan Tur、Benoit Favre、Elizbeth Shriberg、Anoop Sarkar、Sameer Pradhan、Katrin Kirchhoff、Mark Sammons、V. G. Vinod Vydiswaran、Dan Roth、Carmen Banea、Rada Mihalcea、Janyce Wiebe、Xiaqiang Luo、Philipp Koehn、Philipp Sorg、Philipp Cimiano、Frank Schilder、Liang Zhou、Nico Schlaefer、Jennifer Chu-Carroll、Vittorio Castelli、Radu Florian、Roberto Pieraccini、David Suendermann、John F. Pitrelli 以及 Bon Lewis、Daniel M. Bikel 还有 Google Research 表示感谢，特别对 Corinna Cortes 在本项目最后阶段给予的支持表示感谢。最后我们（Daniel M. Bikel 和 Imed Zitouni）要对 IBM Research 给支持表示由衷的感谢，特别要感谢 Ellen Yoffa，没有她，本项目就不可能完成。



Daniel M Bikel（dbikel@google.com）是 Google 的高级研究科学家。他于 1993 年荣誉毕业于哈佛大学，获得古希腊语和拉丁语古典学学位。1994～1997 年，他在 BBN 工作，参加多项自然语言处理研究，包括开发首个高精度随机名字发现程序，并拥有专利。他分别在 2000 年和 2004 年获得宾夕法尼亚大学计算机科学硕士和博士学位，发现了统计句法分析算法的新特性。2004～2010 年，他是 IBM 研究院的研究人员，参与多项自然语言处理研究，包括句法分析、语义角色标注、信息抽取、机器翻译、问答等。Bikel 博士是《计算语言学》杂志的审稿人，ACL、NAACL、EACL 和 EMNLP 会议程序委员。他还在一流的会议和杂志上发表了大量同行评审的论文，并开发了在自然语言处理界广泛使用的软件工具。在 2008 年的“ACL-08：HLT”会议上获得了最佳论文奖（出色短文）。2010 年以来，Bikel 博士一直在 Google 从事自然语言处理和语音处理研究。



Imed Zitouni（izitouni@us.ibm.com）2004 年迄今是 IBM 的高级研究员。他分别于 1995 年和 2000 年从法国南锡大学荣誉毕业并且获得计算机科学硕士和博士学位，他于 1995 年获得突尼斯一家著名的国家计算机学院（Ecole Nationale des Sciences de l'Informatique）的工程硕士学位。

在加入 IBM 前，他在 1999 年和 2000 年是一家初创公司 DIALOCA 的首席科学家。2000～2004 年，他作为研究人员加入了 Lucent-Alcatel 贝尔实验室。他的研究兴趣包括自然语言处理、语言模型、口语对话系统、语音识别和机器学习。Zitouni 博士是 2009～2011 年 IEEE 语音和语言技术委员会委员，他是《ACM Transactions on Asian Language Information Processing》的副主编，计算语言协会（Association for Computational Linguistics，ACL）阿米特语计算方法特别兴趣组的信息官。他是 IEEE 高级会员、ISCA 和 ACL 会员，在多个同行评审会议和杂志担任程序委员和主席。他在自己的研究领域内拥有数个专利，在同行评审的会议和杂志上发表了 70 多篇论文。



Carmen Banea（carmen.banea@gmail.com）是北得克萨斯大学计算机科学和工程系的博士生。她的研究领域是自然语言处理。她的研究工作集中于多语主观性和情感分析，她开发了基于词典和基于语料库的方法，利用资源丰富的语言来建立其他语言的工具和数据。Carmen 在主流的自然语言处理会议上发表了多篇论文，会议包括 ACL、EMNLP（Empirical Methods in Natural Language Processing）、ICCL（International Conference on Computational Linguistics）等。她在多个大型会议上担任程序委员，也是《计算语言学》杂志和《自然语言工程》杂志的审稿人。她在与 ACL 2010 共同召开的 TextGraphs-2010 Workshop 上担任共同主席，也是 2008～2011 年北美计算语言学奥林匹克赛的北得克萨斯大学站的组织者之一。

Vittorio Castelli（vittorio@us.ibm.com）1988 年毕业于米兰理工大学，获得电子工程学士学位，并于 1990 年、1994 年和 1995 年分别获得电子工程硕士学位、统计学硕士学位和电子工程博



士学位。其中博士学位的论文是关于信息论和统计分类的研究。1995 年他加盟 IBM T. J. Watson Research Center，最近他的研究方向是自然语言处理，特别是信息抽取领域。他致力于研究 DARPA GALE 和机器阅读项目。Vittorio 在此之前启动了 Personal Wizards 项目，该项目的目标是通过观察专家执行任务的过程来捕捉执行流程知识。他已经完成的工作涉及信息论、内存压缩、时间序列预测和索引、性能分析，提出了对计算机系统的可靠性和服务性能与科学图形数字库的改进方法。1996～1998 年，他是编号为 NCC5-101 的 NASA/CAN 项目的共同研究人员。他主要的研究兴趣包含信息论、数学论、统计和统计模式识别。1998～2003 年，他是哥伦比亚大学的助理教授，讲授信息论和统计模式识别。他是 IEEE IT Society 的 Sigma Xi 成员，也是美国统计协会的成员。Vittorio 发表的论文涉及自然语言处理、计算机辅助教学、统计分类、数据压缩、图像处理、多媒体数据库、数据库挖掘、多维度索引结构、智能用户接口以及信息论的基本问题，并共同编辑了《Image Databases: Search and Retrieval of Digital Imagery》(Wiley，2002)。



Jenifer Chu-Carroll (jencc@us.ibm.com) 是 IBM T. J. Watson Research Center 语义分析与集成部门的研究人员。她于 2001 年加盟 IBM，在此之前，她以技术人员的身份在 Lucent Technologies 贝尔实验室工作了五年。她的研究兴趣包含问答、语义搜索、会话处理和口语对话管理。



Philipp Cimiano (cimiano@cit-ec.uni-bielefeld.de) 是德国比勒费尔德大学的计算机科学教授。他领导的 Semantic Computing Group 隶属于 Cognitive Interaction Technology Excellence Center，该中心在卓越创新体系下由德国研究基金会 (Deutsche Forschungsgemeinschaf) 资助。Philipp Cimiano 在斯图加特大学的主攻专业是计算机科学，辅修专业是计算语言学。他在卡尔斯鲁厄大学获得了博士学位（最高荣誉）。他主要的研究兴趣在于如何将语义技术与自然语言相结合。在过去的几年里，他致力于多语言信息的访问的研究。他作为主要研究人员参加了许多欧洲研究项目（Dot.Kom，X-Media，Monnet）和国际研究项目，例如 SmartWeb（BMBF）和 Multipla（DFG）。



Benoit Favre (benoit.favre@lif.univ-mrs.fr) 是位于法国马赛的艾克斯-马赛大学的副教授。他的研究领域是自然语言理解。他的研究兴趣在于利用机器学习方法来解决语音和文本理解问题。他于 2007 年在法国阿维尼翁大学获得博士学位，其中论文的主题是语音自动摘要。2008～2007 年，Benoit 在阿维尼翁大学担任教学助理，并在同一时期作为巴黎 Thales Land & Joint Systems 的研究工程师。2007～2009 年，Benoit 在国际计算机研究所（Berkeley，CA）语音组做博士后研究。2009～2010 年，他在法国勒芒大学做博士后研究。从 2010 年开始，他成为艾克斯－马赛大学的终身副教授和 Laboratoire d'Informatique Fondamentale 的会员。Benoit 在国际会议和期刊上合著的审阅论文超过 35 篇。他是该领域主要会议（ICASSP、Interspeech、ACL、EMNLP、Coling、NAACL）和期刊《IEEE Transactions on Speech and Language Processing》的审稿人。他是 International Speech Communication Association 和 IEEE 的会员。



Radu Florian（raduf@us.ibm.com）是 IBM 统计内容分析（信息抽取）组的经理。生于 2002 年在约翰斯·霍普金斯大学获得博士学位，同年加入 IBM 多语自然语言处理组。在 IBM，他参与了信息抽取领域很多不同的研究项目：提及检测、共指消解、关系抽取、跨文本共指和目标信息检索。Radu 领导研究组参加了几个 DARPA 项目（GALE Distillation，MRP）和 NIST 组织的评测（ACE，TAC-KBP），并且和 IBM 合作伙伴（Nuance）共同开发了用于医疗领域的文本挖掘项目，并为 Watson Jeopardy! 项目做出了贡献。



Dilek Hakkani-Tür（Dilek.Hakkani-Tur@microsoft.com）是微软首席科学家。在加入微软之前，她在国际计算机科学研究所（International Computer Science Institute，ICSI）语言组和 AT&T Labs-Research（2001～2005 年）从事研究工作。她于 1994 年在中东技术大学获得学士学位，并分别于 1996 年和 2000 年在宾夕法尼亚大学计算机工程系获得硕士和博士学位。她的博士论文是关于韵律语法的统计语言建模。她于 1997 年和 1998 年分别在卡耐基梅隆大学语言技术研究所和约翰斯·霍普金斯大学从事机器翻译研究。1998～1999 年，Dilek 在 SRI International 利用词汇和韵律信息来完成语音的信息抽取。她的研究兴趣包含自然语言和语音处理、口语对话系统以及针对语言处理的主动和无监督学习。她拥有 13 个专利，参与撰写的关于自然语言和语音处理的论文数量超过 100 篇。她在 2005～2008 年是《IEEE Transactions on Audio，Speech and Language Processing》的副主编。她现在是 IEEE Speech 和 Language Technical Committee 的当选委员（2009～2012 年）。



Katrin Kirchhoff（kk2@u.washington.edu）是华盛顿大学电子工程专业的研究副教授。她主要的研究兴趣是自动语音识别、自然语言处理和人机交互，特别是针对多语言的应用。她写作的同行审阅的出版物数量超过 70 篇，并且是《Multilingual Speech Processing》的共同编辑。Katrin 现在是 IEEE Speech Technical Committee 的会员，也是《Computer，Speech and Language》和《Speech Communication》的编委。



Philipp Koehn（pkoehn@inf.ed.ac.uk）是爱丁堡大学的教授。他在南加州大学获得博士学位，并于 1997～2003 年在该大学的信息科学研究所担任研究助理。他于 2004 年在麻省理工学院担任博士后研究助理，并于 2005 年加盟爱丁堡大学成为讲师。他主要研究统计机器翻译，但也涉及语音、文本分类和信息抽取。他对机器翻译领域的主要贡献是 Europarl 语料的预备与发布、Pharaoh 和 Moses 解码器的开源。他是 ACL 机器翻译特殊兴趣组的组长，也是专著《Statistical Machine Translation》的作者（剑桥大学出版社，2010）。



Burn L. Lewis（burnl@us.ibm.com）是 IBM T. J. Watson Research Center 计算机科学部门的成员。他分别于 1967 年和 1968 年在奥克兰大学的电子工程专业获得学士和硕士学位，并于 1974 年在加州伯克利大学的电子工程和计算机科学专业获得博士学位。他随后加盟 IBM 的 T. J. Watson Research Center，其主要研究方向是语音识别和非结构化的信息管理。



Xiaoqiang Luo（xiaoluo@us.ibm.com）是 IBM T. J. Watson Research Center 的研究人员。他对人类语言技术有广泛的研究经历，包含语音识别、口语对话系统和自然语言处理。在 IBM 语音和语言技术领域的很多由政府资助的成功项目中，他是主要的贡献者。他在 2007 年获得 IBM 杰出技术成就奖，在 2006 年获得 IBM ThinkPlace Brave 奖和许多发明成就奖。Luo 博士分别于 1999 年和 1995 年在约翰斯·霍普金斯大学获得博士和硕士学位，于 1990 年在中国科学技术大学电子工程专业获得学士学位。Luo 博士是计算语言学协会成员，并且作为多个人类语言和人工智能主要技术会议的程序委员。他是中国科学与技术协会大纽约分会（Greater New York Chapter）委员会的成员。他于 2007～2010 年担任《ACM Transactions on Asian Language Information Processing（TALIP）》的副主编。



Rada Mihalcea（rada@cs.unt.edu）是北得克萨斯大学计算机科学与工程系副教授。她的研究兴趣是计算语言学，特别是词汇语义学、自然语言处理中基于图的算法以及多语自然语言处理。她目前参与了多项研究项目，其中包含词义消歧、单语言和交叉语言的语义相似度、关键词自动抽取、文本摘要、情感分析和计算机幽默。Rada 现担任或曾经担任《Journals of Computational Linguistics》、《Language Resources and Evaluations》、《Natural Language Engineering》和《Research in Language in Computation》等杂志的编委。她的研究获得了 National Science Foundation、Google、National Endowment for the Humanities、State of Texas 的资助。她获得了国家科学基金会 CAREER 奖（2008 年）和美国总统青年科技奖（PECASE，2009 年）。



Roberto Pieraccini（www.robertopieraccini.com）是 SpeechCycle 公司首席技术官。Roberto 在 1980 年毕业于意大利的比萨大学电子工程专业。1981 年，他是 CSELT 的语音识别研究人员，CSELT 是意大利电话运营公司的研究机构。他于 1990 年加入贝尔实验室（美利山，新泽西州），成为一名从事语音识别和口语理解研究的技术人员。随后他于 1996 年加入 AT&T 实验室，在这里他开始了口语对话的研究。1999 年他担任 SpeechWorks International 的研发主管。2003 年，他加盟 IBM T. J. Watson Research Center，管理高级会话互动技术部，在 2005 年加盟 SpeechCycle，成为首席技术官。Roberto Pieraccini 在语音识别、语言建模、字符识别、语言理解和自动口语对话管理等领域所著的论文和文章超过 120 篇。他是 ISCA 和 IEEE 会员，是《IEEE Signal Processing Magazine》和《International Journal of Speech Technology》的编委。他也是 Applied Voice Input Output Society and Speech Technology Consortium 委员会成员。



John F. Pitrelli (pitrelli@us.ibm.com) 是 IBM T. J. Watson Research Center 多语自然语言处理部门的成员。他分别于 1983 年、1985 年、1990 年在麻省理工学院电子工程与计算机科学专业获得学士、硕士和博士学位。研究生时的工作是关于语音识别与合成的。在担任当前的职务之前，他在纽约怀特普莱恩斯的 NYNEX Science & Technology 公司的 Speech Technology Group 工作，是 IBM Pen Technologies 组的成员。他也在 Watson 的 Human Language Technologies 组从事语音合成和韵律学研究。John 的研究兴趣包含自然语言处理、语音合成、语音识别、手写体识别、统计语言建模、韵律学、非结构化的信息管理和用于识别的信心建模。他已经发表论文 40 篇，并拥有 4 个专利。



Sameer Pradhan（sameer.pradhan@Colorado.edu）是剑桥大学 BBN Technologies 和麻省理工学院的科学家。他在计算语义领域发表的文章和书籍中的章节得到了大量的引用。他目前正在开创下一代语义分析引擎及其应用。实现这个目标可以通过算法创新；通过研究工具的广泛分布，例如 Automatic Statistical SEmantic Role Tagger（ASSERT）；伸或是通过生成一个丰富、多层、多语言和资源集成的平台，比如 OntoNotes。最后这些语义模型应该替代当前在大多数应用领域普遍使用的简便的基于词的模型，并帮助丰富语言理解领域达到一个新的水平。Sameer 于 2005 年在科罗拉多大学获得博士学位，随后他在 BBN Technologies 致力于开发 OntoNotes 语料。其中 OntoNotes 是 DARPA Global Autonomous Language Exploitation 项目的一部分。他是 ACL 成员，是针对注解、促进注解领域创新的 ACL 特殊兴趣组的创始成员。他经常担任不同自然语言处理会议和研讨会的程序委员，比如 ACL、HLT、EMNLP、CoNLL、COLING、LREC 和 LAW。他也是一位很有成就的厨师。



Dan Roth（danr@illinois.edu）是伊利诺伊大学厄巴纳—香槟分校计算机科学系和贝克曼研究所的教授。他是 AAAI 的会员，伊利诺伊大学学者，在图书馆与信息科学研究生院和统计语言系担任教师职务。Roth 教授的研究横跨机器学习和知识推理的理论研究，特别是自然语言处理的学习和推导，以及文本信息的智能访问等领域。他在该领域已经发表论文超过 200 篇，并且他的论文获得了多个奖项。他在自然语言应用方面已经开发出了不同的基于高级机器学习的工具，这些工具已经广泛应用在研究界。其中包含一个屡获殊荣的语义分析器。他是 AAAI'11、CoNLL'02 和 ACL'03 的程序委员会主席，并且现在是几个他所在领域的期刊的编委。他现在是《Journal of Artificial Intelligence Research》和《Machine Learning Journal》的副主编。Roth 教授以优异的成绩获得以色列理工学院数学专业的学士学位，并在哈佛大学计算机科学专业获得博士学位。



Mark Sammons（mssammon@illinois.edu）是伊利诺伊大学厄巴纳—香槟分校认知计算组的首席研究科学家。他主要的研究兴趣是自然语言处理和机器学习，特别专注于将不同的信息源集成到文本蕴涵的上下文中。他的工作已专注于开发一个文本蕴涵框架，使得新的资源可以容易地融入进来，设计出一个合理的推导程序来识别蕴涵，鉴别和开发自动的方法来识别和表达自然语言文本的隐含的内容。Mark 于 2004 年在伊利诺伊大学计算机科学专业获得硕士学位，于 2008 年在英格兰的利兹大学机械工程专业获得博士学位。



Anoop Sarkar（www.cs.sfu.ca/~anoop）是位于加拿大不列颠哥伦比亚省的西蒙·弗雷泽大学的计算科学副教授，他是自然语言处理实验室（http://natlang.cs.sfu.ca）的主要负责人之一。他在宾夕法尼亚大学计算机与信息科学系获得博士学位。在 Aravind Joshi 教授的指导下完成了半监督的统计句法分析和树邻接文法的句法分析。Anoop 当前专注于研究统计句法分析和机器翻译（利用句法或形态学，或者两者结合）。他的兴趣还包含正规语言理论和随机文法，特别是树自动机和树邻接文法。



Frank Schilder（frank.schilder@thomsonreuters.com）是 Thomson Reuters 研发部的首席研究科学家。他于 2004 年加盟 Thomson Reuters，致力于研究摘要技术和信息抽取系统。他关于摘要的工作已经实现为摘要生成器，用于 WestLawNext 的搜索结果（WestLawNext 是 Thomson Reuters 新开发的法律研究系统）。他当前的研究涉及参加不同的研究比赛，比如由美国国家标准与技术研究所举办的文本分析会议。他于 1997 年在苏格兰的爱丁堡大学认知科学专业获得博士学位。1997～2003 年，他受聘于德国汉堡大学信息系，开始作为博士后研究人员，后来成为助理教授。Frank 已经在几个期刊上发表了多篇论文，并编写了一些书的章节，其中包括《Encyclopedia of Language and Linguistics》（Elsevier，2006）书的“Natural Language Processing: Overview”，内容由他与 Thomson Reuters 的首席科学家 Peter Jackson 合著。2011 年，他获得了 Thomson Reuters Innovation 挑战。他在计算语言学期刊担任审稿人，并多次成为由 Association of Computational Linguistics 组织的会议的议程委员会成员。



Nico Schlaefer（nico@cs.cmu.edu）是卡耐基梅隆大学计算机科学学院的博士研究生，也是 IBM 博士 Fellow。他的研究主要是将机器学习技术应用在自然语言处理任务中。Schlaefer 开发的算法能够让问答系统找到正确的答案——尽管原始的信息源几乎没有包含相关的内容，并开发了一个灵活的框架来支持集成这样的算法。Schlaefer 是 OpenEphyra 的主要作者（OpenEphyra 是最广泛使用的开源问答系统之一）。Nico 对 Watson 贡献了一个统计的源扩展方法（Watson 是一台在 Jeopardy! 智力竞赛表演中战胜人类的计算机）。他利用网络和其他大型文本语料库来自动扩展知识源，使得 Watson 能够更加容易地找到答案和支持的证据。



Elizabeth Shriberg（elshribe@microsoft.com）当前是微软首席科学家。之前她在 SRI International（加利福尼亚州门洛帕克）工作，她也隶属于国际计算机科学研究所（加州大学伯克利分校）和 CASL（马里兰大学）。她在哈佛（1987 年）获得学士学位，在加州大学伯克利分校（1994 年）获得博士学位。Elizabeth 主要的兴趣是使用词汇和韵律信息来完成自发语言建模。她的工作旨在将语言学知识与语料、自动语音、说话者辨别技术结合，进而提高科学理解和技术。她在语音科学和技术领域已经发表了大约 200 篇论文，并担任《语言和语音》的副主编，是 Speech Communication and Computational Linguistics 委员会委员，是许多会议和研讨会的委员会委员，是 ISCA Advisory Council 和 ICSLP Permanent Council 的委员会委员。她已经组织了多个研讨会，并担任 National Science Foundation、European Commission、NOW（荷兰）的委员会委员。她已经审阅过许多跨学科的会议、研讨会和期刊（例如《IEEE Transaction on Speech and Audio Process-

ing》、《Journal of the Acoustical Society of America，Nature》、《Journal of Phonetics，Computer Speech and Language》、《Journal of Memory and Language，Memory and Cognition，Discourse Processes》），2009年，他获得了ISCA Fellow奖。2010年他成为了SRI的会员。



Otakar Smrž（otakar.smrz@cmu.edu）是位于卡塔尔的卡耐基梅隆大学博士后研究人员，他致力于通过学习可比语料的方法来改进以阿拉伯语作为源语言和目标语言的机器翻译。Otakar在位于布拉格的查尔斯大学完成他的数学语言学的博士研究。他使用函数式编程来设计和实施阿拉伯形态学的ElixirFm计算模型，并开发了其他自然语言处理的开源软件。他曾经是Prague Arabic Dependency Treebank的主要研究人员。Otakar过去是IBM Czech Republic的研究科学家，致力于开发无监督的语义分析和对多语言的声音建模。Otakar是位于卡塔尔的Dōm-e Dānn语言学院的联合创办者。



Philipp Sorg（philipp.sorg@kit.edu）是德国卡尔斯鲁厄技术研究所的博士研究生。他是应用信息与形式化描述方法学院的研究人员。Philipp毕业于卡尔斯鲁厄大学计算机科学专业，他主要的研究兴趣是多语言信息获取。他特别关注利用社会语义应用到Web 2.0的上下文中。他已经参与了欧洲研究项目Active，还参加了国际研究项目Multipla（DFG）。



David Suendermann（david@speechcycle.com）是SpeechCycle Labs（纽约）的首席语音科学家。Suendermann博士在过去的十年里探索了语音技术研究的很多不同领域，他在多个企业和学术机构从事研究，其中包括西门子（慕尼黑）、哥伦比亚大学（纽约）、南加州大学（洛杉矶）、加泰罗尼亚理工大学（巴塞罗那）和亚琛工业大学（亚琛，德国）。他参与出版的书籍和专利数目超过了60，其中包括一本书和5本书的部分章节，他在慕尼黑的德国联邦国防军大学获得博士学位。



Gokhan Tur（gokhan.tur@ieee.org）目前是微软的首席科学家。他分别于1994年、1996年和2000年在土耳其的毕尔肯大学获得学士、硕士和博士学位。1997～1999年，Tur访问卡耐基梅隆大学的机器翻译中心，然后访问了约翰斯·霍普金斯大学的计算机科学系，最后访问了SRI International的语音技术和研究实验室。他于2001～2006年在AT&T Labs-Research工作，2006～2010年在SRI International的语音技术和研究实验室工作。他的研究兴趣包含口语理解、语音和语言处理、机器学习及信息获取和抽取。他所著或与他人合著的论文在双盲期刊或书籍上发表的数量已经超过100篇，并出席了一些国际会议。他是《Spoken Language Understanding：Systems for Extracting Semantic Information from Speech》（Wiley，2011）的编审。Tur博士是IEEE、ACL和ISCA的高级会员，也是IEEE Signal Processing Society（SPS）、2006年～2008年的Speech and Language Technical Committee（SLTC）的会员。目前他是《IEEE Transactions on Audio，Speech，and Language Processing》的副主编。



V. G. Vinod Vydiswaran（vgvinodv@illinois.edu）目前是伊利诺伊大学厄巴纳—香槟分校计算机科学系的博士研究生。他的论文是关于网络的信息可信度检测，他的导师是 ChengXiang Zhai 教授和 Dan Roth 教授。他的研究兴趣包含文本信息、自然语言处理、机器学习和信息抽取。V. G. Vinod 的工作包含开发文本蕴涵系统并将文本蕴涵应用在关系抽取和信息获取中。他于 2004 年在印度理工学院孟买分校获得硕士学位，他在导师 Sunita Sarawagi 教授的指导下研究信息抽取的条件模型。毕后他在印度的班加罗尔 Yahoo 研发中心工作，研究网络规模信息抽取技术。



Janyce Wiebe（wiebe@cs.pitt.edu）是匹兹堡大学计算机科学专业教授和智能系统计划的联合主任。她与学生和同事的研究方向是自然语言处理的话语处理、语用学、词义消歧和概率分类。她的研究主要关注主观性分析，对文本的情感和意见表达的识别和解释，用于支持自然语言处理的应用，例如问答、信息抽取、文本分类和概要。Janyce 在专业领域曾担任的角色包括 ACL 议程联合主席、NAACL 程序主席、NAACL 执行委员会委员、计算语言学家、语言资源和评估专家、编辑委员会委员、AAAI 研讨会联合主席、ACM 人工智能（SIGART）特殊兴趣组副主席和 ACM-SIGART/AAAI 博士论坛主席。



Hyun-Jo You（youhyunjo@gmail.com）目前是首尔国立大学语言系讲师。他在首尔国立大学获得博士学位。他的研究兴趣包含定量语言学、统计语言建模和计算语料分析。他对研究形态变化多样、无词序语言的形态句法和话语结构特别感兴趣，例如汉语、韩点语和俄罗斯语。



Liang Zhou（liangz@isi.edu）是 Thomson Reuters 公司的研究科学家。她在自然语言处理方面有广博的知识，包括情感分析、自动文本摘要、文本理解、信息抽取、问答和信息提炼。她在信息科学研究所做研究生时，积极参与了由政府资助的多个项目，比如 NIST Document Understanding 会议和 DARPA Global Autonomous Language Exploitation。Zhou 博士于 2006 年在南加州大学获得博士学位，于 2001 年在斯坦福大学获得硕士学位，于 1999 年在内纳西州大学获得学士学位，专业都是计算机科学。

出版者的话
译者序
前言
关于作者

第一部分 理论

第1章 找出词的结构 …… 2
1.1 词及其部件 …… 2
1.1.1 词元 …… 2
1.1.2 词形 …… 3
1.1.3 词素 …… 4
1.1.4 类型学 …… 5
1.2 问题和挑战 …… 6
1.2.1 不规则性 …… 6
1.2.2 歧义性 …… 7
1.2.3 能产性 …… 9
1.3 形态模型 …… 10
1.3.1 查词典 …… 11
1.3.2 有限状态形态 …… 11
1.3.3 基于合一的形态 …… 13
1.3.4 函数式形态 …… 13
1.3.5 形态归纳 …… 14
1.4 总结 …… 15
第2章 找出文档的结构 …… 21
2.1 概述 …… 21
2.1.1 句子边界检测 …… 22
2.1.2 主题边界检测 …… 23
2.2 方法 …… 24
2.2.1 生成序列分类方法 …… 25
2.2.2 判别性局部分类方法 …… 26
2.2.3 判别性序列分类方法 …… 26
2.2.4 混合方法 …… 28
2.2.5 句子分割的全局建模扩展 …… 29
2.3 方法的复杂度 …… 29
2.4 方法的性能 …… 30
2.5 特征 …… 30
2.5.1 同时用于文本与语音的特征 …… 30
2.5.2 只用于文本的特征 …… 32
2.5.3 语音特征 …… 33
2.6 处理阶段 …… 35
2.7 讨论 …… 35
2.8 总结 …… 38
第3章 句法 …… 42
3.1 自然语言分析 …… 42
3.2 例证：句法分析的数据驱动方法 …… 45
3.3 句法结构的表示 …… 46
3.3.1 使用依存图的句法分析 …… 46
3.3.2 使用短语结构树的句法分析 …… 49
3.4 分析算法 …… 52
3.4.1 移进和的分析 …… 53
3.4.2 超图和线图分析 …… 53
3.4.3 最小生成树和依存分析 …… 58
3.5 分析中的歧义消解模型 …… 59
3.5.1 概率上下文无关文法 …… 59
3.5.2 句法分析的生成模型 …… 61
3.5.3 句法分析的判别模型 …… 62
3.6 多语言问题：什么是词元 …… 65
3.6.1 词元切分、实例和编码 …… 65
3.6.2 分词 …… 66
3.6.3 形态学 …… 67
3.7 总结 …… 68
第4章 语义分析 …… 71
4.1 概述 …… 71
4.2 语义解释 …… 72
4.2.1 结构歧义 …… 72
4.2.2 词义 …… 72
4.2.3 实体与事件消解 …… 73
4.2.4 谓词－论元结构 …… 73
4.2.5 意义表示 …… 73
4.3 系统范式 …… 74
4.4 词义 …… 74
4.4.1 资源 …… 76
4.4.2 系统 …… 77
4.4.3 软件 …… 85
4.5 谓词－论元结构 …… 85
4.5.1 资源 …… 86
4.5.2 系统 …… 89
4.5.3 软件 …… 106

4.5 意义表示 …… 106
4.5.1 要素 …… 107
4.5.2 系统 …… 108
4.5.3 软件 …… 109
4.7 总结 …… 109
4.7.1 词义消歧 …… 115
4.7.2 谓词 论元结构 …… 120
4.7.3 意义表示 …… 121
第5章 语言模型 …… 122
5.1 概述 …… 122
5.2 n元模型 …… 122
5.3 语言模型评价 …… 123
5.4 参数估计 …… 125
5.4.1 最大似然估计和平滑 …… 125
5.4.2 贝叶斯参数估计 …… 125
5.4.3 大规模语言模型 …… 126
5.5 语言模型适应 …… 127
5.6 语言模型的类型 …… 128
5.6.1 基于类的语言模型 …… 128
5.6.2 变长语言模型 …… 129
5.6.3 结构化语言模型 …… 129
5.6.4 基于句法的语言模型 …… 130
5.6.5 最大熵语言模型 …… 131
5.6.6 因子化语言模型 …… 132
5.6.7 其他基于树的语言模型 …… 133
5.6.8 基于主题的贝叶斯语言模型 …… 134
5.6.9 神经网络语言模型 …… 135
5.7 特定语言建模问题 …… 136
5.7.1 形态丰富语言的建模 …… 136
5.7.2 建词单元的选择 …… 138
5.7.3 形态类别建模 …… 139
5.7.4 无分词语言 …… 140
5.7.5 口语与书面语言 …… 140
5.8 多语言和跨语言建模 …… 141
5.8.1 多语言建模 …… 141
5.8.2 跨语言建模 …… 141
5.9 总结 …… 143
第6章 文本蕴涵识别 …… 151
6.1 概述 …… 151
6.2 文本识别蕴涵任务 …… 151
6.2.1 问题定义 …… 152
6.2.2 RTE的挑战 …… 153
6.2.3 评估文本蕴涵系统性能 …… 154
6.2.4 文本蕴涵解决方案的应用 …… 155
6.2.5 其他语言中的RTE研究 …… 157
6.3 文本蕴涵识别的框架 …… 158
6.3.1 要求 …… 158
6.3.2 分析 …… 159
6.3.3 有用的软件 …… 159
6.3.4 组成模型 …… 162
6.3.5 实现 …… 164
6.3.6 对齐 …… 166
6.3.7 推理 …… 171
6.3.8 训练 …… 172
6.4 案例分析 …… 172
6.4.1 抽取语篇约束 …… 172
6.4.2 基于编辑距离的RTE …… 172
6.4.3 基于特征的方法 …… 174
6.4.4 逻辑表示及推理 …… 176
6.4.5 独立于蕴涵学习对齐 …… 176
6.4.6 在RTE中利用多对齐 …… 177
6.4.7 自然逻辑 …… 177
6.4.8 句法树核 …… 178
6.4.9 使用有限标签上下文的全局相似度 …… 178
6.4.10 RTE的潜在对齐推理 …… 179
6.5 RTE的进一步研究 …… 179
6.5.1 改进分析器 …… 179
6.5.2 发明新解决方案 …… 180
6.5.3 开发知识库 …… 180
6.5.4 更好的RTE评价 …… 181
6.6 有用资源 …… 182
6.6.1 文献 …… 182
6.6.2 知识库 …… 182
6.6.3 自然语言处理包 …… 182
6.7 总结 …… 183
第7章 多语情感与主观性分析 …… 188
7.1 概述 …… 188
7.2 定义 …… 188
7.3 英语中的情感及主观性分析 …… 190
7.3.1 词典 …… 190
7.3.2 语料库 …… 191
7.3.3 工具 …… 191
7.4 词级和短语级标注 …… 192
7.4.1 基于字典的方法 …… 192
7.4.2 基于语料库的方法 …… 194
7.5 句子级标注 …… 195
7.5.1 基于字典 …… 195

7.5.2 基于语料库 …… 197
7.6 文档级标注 …… 198
7.6.1 基于字典 …… 198
7.6.2 基于语料库 …… 199
7.7 什么有效，什么无效 …… 200
7.7.1 最佳情况：已有人工标注的语料库 …… 200
7.7.2 次优情形：基于语料库的跨语言映射 …… 200
7.7.3 第三优情形：单份词典 …… 201
7.7.4 第四优情形：翻译词典 …… 201
7.7.5 各种可行方法的比较 …… 201
7.8 总结 …… 202

第二部分 实践

第8章 实体检测和追踪 …… 205
8.1 概述 …… 208
8.2 提及检测 …… 209
8.2.1 数据驱动的分类 …… 210
8.2.2 搜索提及 …… 211
8.2.3 提及检测特征 …… 213
8.2.4 提及检测实验 …… 215
8.3 共指消解 …… 216
8.3.1 Bell树的构建 …… 217
8.3.2 共指模型：链接和引入模型 …… 218
8.3.3 最大熵链接模型 …… 219
8.3.4 共指消解实验 …… 220
8.4 总结 …… 221
第9章 关系和事件 …… 225
9.1 概述 …… 225
9.2 关系与事件 …… 225
9.3 关系类别 …… 226
9.4 将关系抽取视为分类 …… 227
9.4.1 算法 …… 227
9.4.2 特征 …… 228
9.4.3 分类器 …… 230
9.5 关系抽取的其他方法 …… 231
9.5.1 无监督和半监督方法 …… 231
9.5.2 核方法 …… 232
9.5.3 实体和关系检测的联合方法 …… 233
9.6 事件 …… 233
9.7 事件抽取方法 …… 234
9.8 细句 …… 235
9.9 事件匹配 …… 235
9.10 事件抽取的未来方向 …… 237
9.11 总结 …… 237
第10章 机器翻译 …… 241
10.1 机器翻译现状 …… 241
10.2 机器翻译评测 …… 241
10.2.1 人工评测 …… 242
10.2.2 自动评测 …… 243
10.2.3 WER、BLEU、METEOR等 …… 244
10.3 词对齐 …… 246
10.3.1 表现 …… 246
10.3.2 IBM模型1 …… 247
10.3.3 期望最大化 …… 247
10.3.4 对齐模型 …… 248
10.3.5 对称化 …… 248
10.3.6 作为机器学习问题的词对齐 …… 250
10.4 基于短语的翻译模型 …… 250
10.4.1 模型 …… 251
10.4.2 划线 …… 251
10.4.3 解码 …… 252
10.4.4 立方剪枝 …… 254
10.4.5 对数线性模型和参数调节 …… 254
10.4.6 控制模型的大小 …… 255
10.5 基于树的翻译模型 …… 256
10.5.1 基次短语翻译模型 …… 256
10.5.2 线图解码 …… 257
10.5.3 基于句法的模型 …… 258
10.6 语言学挑战 …… 259
10.6.1 译词选择 …… 259
10.6.2 形态学 …… 260
10.6.3 词序 …… 260
10.7 工具和数据资源 …… 261
10.7.1 基本工具 …… 261
10.7.2 机器翻译系统 …… 262
10.7.3 平行语料 …… 262
10.8 未来的方向 …… 262
10.9 总结 …… 263
第11章 考语言信息检索 …… 267
11.1 概述 …… 267
11.2 文档预处理 …… 268
11.2.1 文档句法和编码 …… 268
11.2.2 词元化 …… 270
11.2.3 规范化 …… 271
11.2.4 预处理最佳实践 …… 272
11.3 单语信息检索 …… 272

11.3.1 文档表示 …… 272
11.3.2 索引结构 …… 273
11.3.3 检索模型 …… 274
11.3.4 查询扩展 …… 275
11.3.5 文档先验模型 …… 276
11.3.6 模型选择的最佳实践 …… 276
11.4 CLIR …… 277
11.4.1 基于翻译的方法 …… 277
11.4.2 机器翻译 …… 278
11.4.3 中间语言主题表示 …… 279
11.4.4 最佳实践 …… 280
11.5 多语言信息检索 …… 280
11.5.1 语言识别 …… 280
11.5.2 MLIR 的索引建立 …… 281
11.5.3 翻译查询串 …… 281
11.5.4 聚合模型 …… 282
11.5.5 最佳实践 …… 282
11.6 信息检索的评价 …… 283
11.6.1 建立实验环境 …… 283
11.6.2 相关性评估 …… 284
11.6.3 评价指标 …… 284
11.6.4 已有测试集 …… 285
11.6.5 最佳实践 …… 286
11.7 工具、软件和资源 …… 287
11.8 总结 …… 288
第 12 章 多语自动文摘 …… 291
12.1 概述 …… 291
12.2 自动文摘方法 …… 293
12.2.1 传统方法 …… 293
12.2.2 基于图的方法 …… 294
12.2.3 学习如何做摘要 …… 297
12.2.4 多语自动摘要 …… 300
12.3 评测 …… 302
12.3.1 人工评价 …… 302
12.3.2 自动评价 …… 304
12.3.3 自动文摘评测系统的近期发展 …… 306
12.3.4 多语自动文摘的自动评测方法 …… 307
12.4 如何搭建自动文摘系统 …… 307
12.4.1 材料 …… 309
12.4.2 二步 …… 309
12.4.3 说明 …… 310
12.5 评测竞赛和数据集 …… 311
12.5.1 评测竞赛 …… 311
12.5.2 数据集 …… 311
12.6 总结 …… 312
第 13 章 问答系统 …… 317
13.1 概述和历史 …… 317
13.2 架构 …… 318
13.3 问题获取和预处理 …… 320
13.4 问题分析 …… 322
13.5 搜索及候选抽取 …… 324
13.5.1 非结构化资源搜索 …… 324
13.5.2 半结构化源文本的候选抽取 …… 326
13.5.3 结构化源文本的候选抽取 …… 329
13.6 回答评分 …… 329
13.6.1 方法概述 …… 330
13.6.2 证据结合 …… 331
13.6.3 扩展到列表型问题 …… 332
13.7 跨语言问答 …… 333
13.8 案例研究 …… 334
13.9 评测 …… 337
13.9.1 评测任务 …… 337
13.9.2 判断答案正确性 …… 338
13.9.3 评价度量 …… 339
13.10 当前和未来的挑战 …… 340
13.11 总结和进一步阅读 …… 340
第 14 章 提炼 …… 348
14.1 概述 …… 348
14.2 示例 …… 349
14.3 相关性和冗余性 …… 349
14.4 Rosetta Consortium 提炼系统 …… 351
14.4.1 文档和语料库准备 …… 352
14.4.2 索引 …… 353
14.4.3 查询回答 …… 354
14.5 其他提炼方法 …… 357
14.5.1 系统架构 …… 357
14.5.2 相关度 …… 357
14.5.3 冗余 …… 358
14.5.4 多模态提炼 …… 358
14.5.5 跨语言提炼 …… 359
14.6 评测和指标 …… 360
14.7 总结 …… 362
第 15 章 口语对话系统 …… 364
15.1 概述 …… 364
15.2 口语对话系统 …… 364
15.2.1 语音识别和理解 …… 365

15.2.2 语音生成 …… 367
15.2.3 对话管理器 …… 367
15.2.4 语音用户接口 …… 369
15.3 对话形式 …… 371
15.4 自然语言呼叫路由选择 …… 372
15.5 三代对话应用 …… 372
15.6 持续的改进循环 …… 373
15.7 口语句子的转录和标注 …… 374
15.8 口语对话系统的本地化 …… 374
15.8.1 呼叫流程本地化 …… 375
15.8.2 提示本地化 …… 375
15.8.3 文法的本地化 …… 376
15.8.4 源编数据 …… 376
15.8.5 训练 …… 377
15.8.6 测试 …… 377
15.9 总结 …… 379
第16章 聚合自然语言处理引擎 …… 381
16.1 概述 …… 381
16.2 聚合语音和NLP引擎架构的期望属性 …… 382
16.2.1 灵活的分布式组件化 …… 382
16.2.2 计算效率 …… 382
16.2.3 数据操作功能 …… 383
16.2.4 普遍性处理 …… 383
16.3 聚合的架构 …… 383
16.3.1 UIMA …… 384
16.3.2 GATE …… 385
16.3.3 InfoSphere Streams …… 386
16.4 案例研究 …… 386
16.4.1 GALE互操作性演示系统 …… 387
16.4.2 跨语言自动语言开发系统 …… 391
16.4.3 实时翻译服务 …… 393
16.5 经验教训 …… 393
16.5.1 分割涉及延迟和精度之间的权衡 …… 393
16.5.2 聚合优化与互操作性 …… 393
16.5.3 数据模型需要使用约定 …… 394
16.5.4 性能评估的挑战 …… 394
16.5.5 引擎的前向波训练 …… 394
16.6 总结 …… 394
16.7 UIMA样本代码 …… 395
索引 …… 401

理　论

第1章“找出词的结构”，描述如何识别人类语言中不同类型的词，如何建立词的内部结构、语法性质、词法概念的模型。

第2章“找出文档的结构”，讨论如何找出文档结构，并将其分解为更容易处理的单位，例如句子或表示同一话题的文本段。

第3章“句法”，描述如何找出句子的结构。

第4章“语义分析”，探索找出句子意义表示的自动方法。

第5章“语言模型”，讨论如何建立一个模型，该模型可对每个可能的有限长度的词串赋以一个概率估算或分数。

第6章“文本蕴涵识别”，讨论确定一段文本中的指定事实是否为另一段文本中的事实所蕴涵的方法。

第7章“多语情感与主观性分析”，探索确定句子是否是主观的并确定所表达的意见的倾向性和其他性质的方法。

第1章

Multilingual Natural Language Processing Applications: From Theory to Practice

找出词的结构

Otakar Smrž, Hyun Jo You

人类语言很复杂。我们用语言来表示思想，获取信息，推断出意义。语言表达并非没有组织。其结构多样，复杂程度千差万别，复杂结构由基本部件组成，在一定的上下文中通过共现来表示比其孤立使用时更精细的意义及其意义间的关系。

整体上理解语言不可行。语言学家从不同的角度、不同的细节层次来考察语言，比如形态学研究词的可变形式和功能，而句法则研究词如何排列构成短语、子句和句子。由于发音而导致的词结构限制由语音学描述，而书写的规则则构成了语言的正字法。语言表达式的意义属于语义学的内容。词源学和词汇学则研究词的演变并解释词之间的语义、形态和其他联系。

词可能是语言最直观的单位，但实际上定义什么是词颇为棘手。词的研究是句法、语义抽象及其他与语言相关的高级话题的前提。形态学是语言处理的必要部分，尤其在多语的环境下变得越来越重要。

本章将探索如何识别人类语言中不同类型的词，如何建立词的内部结构、语法性质、词法概念的模型，词结构的发现称为**形态分析**（morphological parsing）。

这个任务有多困难？决定因素有很多。在某些语言中，词由空格或标点分割；但是在另一些语言中，书写系统使读者区分词或者确定其精确的语音形式。有些语言的词不随上下文变化，而另一些语言的词会根据句法和语义有不同的词形变化。

1.1 词及其部件

在大多数语言中，词被定义为能形成完整言语的最小语言单位。词的最小语义部分称为词素（morpheme）。根据交流方式的不同，词素可用形素（grapheme）（比如字母和字符等书写符号）拼写出或用音素（phoneme）（口语中可区分的语音单位）说出[1]。确定词、词素和短语之间精确的分界并不总是很容易[1, 2]。

1.1.1 词元

假设英语中的词只由空格和标点隔开[3]，考虑例1-1：

例1-1 Will you read the newspaper? Will you read it? I won't read it.

如果我们懂词源和句法知识，那么我们注意到这里有两个词可能和假设有些冲突：*newspaper* 和 *won't*。前者是一个复合词，有明显的派生结构。如果有词典或其他语言证据可佐证该词的来源的假设，我们可能会更详细地描述它。书面上，*newspaper* 及其相关概念和单独的 *news* 与 *paper* 是不同的。然而，在口语中其区别却不甚明显，词的识别成了一个问题。

---

[1] 在手语中用的符号也称为音素的元素构成。

为了一般性，语言学家喜欢把 *won't* 分解为两个语法词，或称词元，其中每个词元有其独立的作用并有规范形式。从结构上说，*won't* 可被分析为 *will* 后面跟随 *not*。在英语中，这种词的切分（tokenization）和规范化（normalization）也许很少，而在其他语言中，这种现象可能很多。

在阿拉伯语或希伯来语中［4］，某些词元在书写时需要与前后的词元连写，也可改变其形式，其内在的词法或句法单位可能体现在紧缩的一串字母中，并非能明晰地分解为词。很多语言中的词元有这种行为，这种词元经常被称为附着词。

在汉语、日语［5］、泰语的书写系统里，不采用空格来隔开词。在某种程度上形式地可区分的单位是句子或子句。在韩语中，字符串称为 *eojeol*（词节），粗略地对应于语音或认知单位，比词大，比子句小［6］，如例 1-2 所示：

例 1-2 학생들에게만 주셨는데

hak. sayng. tul. ey. key. man cwu. syess. nun. te[⊖]

haksayng tul eykey man cwu si ess nunte

student − plural − dative − only give + honorific + past + while

while(he/she)gave(it)only to the students

尽管如此，基本的形态单位被视为有其句法地位［7］。在这些语言中，词的切分，或称分词（word segmentation），是形态分析的基础性步骤，也是大多数语言处理应用的前提。

1.1.2 词形[⊖]

词这个术语，通常我们不但指其在给定上下文中的语言形式，而且表示其形式背后的概念，以及可表示该概念的其他形式的集合。该集合被称为词形，或词项。它们构成了一个语言的词典。词可根据其行为分为动词、名词、形容词、连词、小品词等词类（词性）。词形的引用形式也称为原形（lemma）。

当我们把词转化为其他形式时，比如把单数的 *mouse* 转为复数 *mice* 或 *mouses*，我们说对该词形进行了屈折变化。当把一个词形变化为形态上相关的另一个词，而不管其词类是否相同时，我们称对该词形进行了派生。例如，名词 *receiver* 和 *reception* 是由动词 *to receive* 派生而来。

例 1-3 Did you see him? I didn't see him. I didn't see anyone.

例 1-3 提出了 *didn't* 的切分和 *anyone* 的内部结构问题。在释义 *I saw no one* 中，词 *to see* 被屈折变化成 *saw* 以表示其过去时态的语法功能。同样，*him* 是 *he* 或甚至表示所有人称代词的更抽象的语素的从格形式。在上述释义中，*no one* 可以被认为是和词 *nobody* 同义的最小词。如果我们把两个紧密相关的词元 *no one* 当作一个固定的词项解，那么，对于用语法描述什么是一个词的困难就不复存在了。

在例子 1-3 的捷克语翻译中，词 *vidět* "to see" 屈折变化为过去时，而形式是由第一人称和第二人称的两个词元组成（即 *viděla jsi* 'you-FEM-SG saw' and *neviděla jsem* 'I-FEM-SG did not see'）。捷克语的否定是一个屈折变化参数，而不仅是句法的，而同时在动词及其相关代词中标记，正如例 1-4 所示：

例 1-4 Viděla ho? Neviděla jsem ho. Neviděla jsem nikoho.

⊖ 使用耶鲁拼音表示韩文，通过点号标出原始的字符，使用连字号标记形态学边界，加号分开词元。

⊖ 原文 lexeme 按照字面意义是指词典的基本单位，实际就是"词"。当强调其基本意义时，也翻译为"语素"。这里为了和"word"相区分，译为"词形"，不采用目前的流行翻译"词位"。——译者注

saw+you-are him? not-saw I-am him. not-saw I-am no-one.

这里，*vidēlas* 是 *vidēla jsi* "you-FEM-SG saw" 的紧缩形式。*jsi* "you are" 中的 *s* 是附着词，由于捷克语的自由语序，可以附着在几乎任何词的后面。因此我们可提问：*Nikphos nevidēla?* "Did you see no one?"，此处代词 *nikoho* "no one" 后面跟了这个附着词。

1.1.3 词素

形态理论的主要差别在于是否并且如何将词形的性质与其结构部件联系起来 [8，9，10，11]。这些部件通常称为"节"（segment）或"形元"（morph）。词的表意形元称为某种功能的词素（morpheme）。

5

人类语言采用很多手段，可将形元或词素合并成词形。最简单的形态过程将形元一个接一个连接起来，如 *dis-agree-ment-s*，其中 *agree* 是一个自由词素，其他三个是表达语法意义的黏着词素，合起来表示词的整体意义。

在更复杂的情形中，形元间可互相作用，其形式可有语音或书写的变化，称为"形音"（morpho-phonemic）变化。词素的其他形式称为**变体词素**（allomorph）。

在韩语中，形态变化和词素的形式依赖于语音的例子比比皆是。很多词素随着其语音上下文不同而系统地改变其形式。下面的例 1-5 列出了表示过去时态的时态标记的变体词素 *ess*、*ass*、*yess*。前两个根据其前面动词词干的语音而变化，最后 1 个经常和动词 *ha-* "do" 一起使用。适当的变体可直接跟在词干后面，也可以进一步紧缩，如例 1-2 中 *-si-ess-* 紧缩为 *-syess-*。在形态分析中，变体词素规范化为词素的正规形式是有益的，尤其是当形元的紧缩与简单的切分相干扰的时候。

例 1-5

	连接		紧缩		
(a)	보았-	*po-ass-*	봤-	*pwass-*	'have seen'
(b)	가지었-	*ka.ci-ess-*	가졌-	*ka.cyess-*	'have taken'
(c)	하었-	*ha-yess-*	했-	*hayss-*	'have done'
(d)	되었-	*toy-ess-*	됐-	*twayss-*	'have become'
(e)	놓았-	*noh-ass-*	놨-	*nwass-*	'have put'

紧缩形式 (a)，(b) 是普通的，但是需要引起注意，因为两个字符缩成了一个。其他类型 (c)，(d)，(e) 语音上不可预测，或与具体词相关。例如，*coh-ass-* "have been good" 永远不能紧缩，而 *noh-* 和 *ass-* 被合并成了 *nwass-*，如例 1-5(e) 所示。

还有形成词的其他语言手段需要加以解释，因形态分析过程本身并不是小事。连接操作可能伴有形元的嵌入或交缠，这在阿拉伯语中很普遍。即使在英语中，也存在将词内部的元音进行改变的非连接的屈折变化：请比较 *mouse* 和 *mice*、*see* 和 *saw*、*read* 和 *read* 的音变。

在阿拉伯语中，内部的屈折变化经常发生，并且具有不同的性质。词内部的一部分，称为词干，可由词根和词素模式来描述。词的结构因此可由抽象了词根的、只显示模式和附着在其左右的其他形元来描述。

例 1-6 hl stqrO h*h AljrA}d?[⊖] هل ستقرأ هذه الجرائد؟

hal sa-taqra'u hāḏihi 'l-ǧarā'ida?

whether will+you-read this the-newspapers?

hl stqrWhA? ln OqrOhA. هل ستقرؤها؟ لن أقرأها.

hal sa-taqra'uhā? lan 'aqra'ahā.

6 whether will+you-read+it? not-will I-read+it.

⊖ 使用 Hackwalter 标记直译后来的阿拉伯文字。为了方便阅读，也给出了标准的语音转写，以减少歧义。

例1-6的意义和例1-1类似，只是短语 *hādihi 'l-garā'ida* 指“these newspapers”。*sa taqraʾu* “you will read”在陈述语气和主动语态中合并了将来态标记 *sa* 和未完成第二人称阳性单数动词 *taqraʾu*，而 *sa-taqraʾuha* “you will read it”也增加了在宾格附着的阴性单数人称代词[8]。

taqraʾu “you-MASC-SG read”所属的词形的引用形式是 *qaraʾa*，大意是“to read”。语言学家把这种形式分类为由模板 *faʿal* 与辅音词根 *q r ʾ* 合并的基本动词形式。其中模板的 *f ʿ l* 符号被相应的词根辅音所代替。这个词形的屈折变化可把词目的词干的模式 *faʿal* 修改为 *faʿal*，并且根据形音变化规则和更多的前缀和后缀进行连接。*taqraʾu* 的结构因此可分析为模板 *ta-faʿal u* 和不变词根。

在宾格和确定态的词 *al-garā'ida* “the newspaper”是另一个内部屈折变化的例子。其结构来自于模板 *al-faʿāʾid-a* 和词根 *g r d*。这个词是有模板 *faʿīl-ah* 的 *garīdah* “newspaper”的复数。单、复数模板的联系有一定的规律，应该在词典中声明。

不考虑内含的形态过程，词的特性不一定能从其形态结构中明显看出。其现有的结构部件可能同时配合或依赖于几个功能，但不一定有特别的词法意义或讲法解释。

garīdah “newspaper”的后缀 *-ah* 与该词的内在的阴性相应。事实上，词素 *-ah* 通常（虽然不是在所有情况下）用来标记形容词的阴性单数形式。例如，*gadīd* 变成了 *gadīdah* “new”。然而，后缀 *-ah* 也可以是非阴性的词的一部分，在这种情况下其功能可看作是被虚化或代替[12]。一般情况下，语言形式应该与其功能相区分，也不是每一个形元都能被假设为一个词素。

1.1.4 类型学

形态类型学根据语言的主要的形态现象把语言划分成若干组。可以考虑多种标准，在语言学的历史上，提出了多种分类法[13, 14]。我们简单地刻画一下基于词、词素及其特征的数量关系的类型学。

孤立型（isolating）或**分析型**（analytic）语言不包含或仅少量包含可被划分为多个词素的词（典型成员包括汉语、越南语、泰语；分析型趋势也可以在英语中找到）。

综合型（synthetic）语言可在一个单词中合并多个词素，可进一步被区分为**黏着语**和**屈折语**。

黏着语（agglutinative）的词素一次只能有一个功能（如韩语、日语、芬兰语、泰米尔语等）。

屈折语（fusional）定义为其词素特征比大于1的语言（如阿拉伯语、捷克语、拉丁语、梵语、德语等）。

相对上面提及的词构成过程的概念，我们也区分：

连接型（concatenative）语言可把形元和词素一个接一个连起来。

非线性（nonlinear）语言允许把结构部件进行非顺序的合并，并且可应用声调词素或改变词的元音或辅音模板。

尽管有些形态现象，如字母接合、语音紧缩、复杂的曲折或派生变化，在有些语言中比另一些语言中更明显，但理论上我们可以在不同语言家族或类型中找到这些现象，并且也应该能处理这些现象。

---

8 在阿拉伯语中，动词都编成复数形式上要表示单数。

1.2 问题和挑战

形态分析试图消除或减少词形的可变性，以提供更高级的、其词法和形态性质被明确表示或定义的语言单位。它试图去除不必要的不规则性、限制歧义，而这两者在人类语言中是内在的。

不规则性意味着存在不能用一个典型的语言学模型来描述的形态和结构。有些不规则性可以通过重新设计模型或改进规则来解决，但是其他依赖于词的不规则性经常不能被一般化。

歧义是语言表达解释的不确定性，除了偶然的歧义、由于多义词而导致的歧义，还有一种叫同态（syncretism），即系统性歧义。

形态建模也面临着语言能产性和创造性问题，因为新词或旧词新义不断产生。不过通常而言，没列在形态分析词典中的词一般无法分析，这个问题称为未登录词（unknown word）问题，不管是在口语或书面语中，只要和语言学模型期望的领域不一致时这种情况就会很严重，例如当语篇中存在专门术语或外来词的时候，或者当多种语言或方言混在一起的时候。

1.2.1 不规则性

形态分析追求词的世界的一般性和抽象性。对给定的语言数据的快速描述不一定是终极描述，因为数据可能是不精确的或其复杂性是不适当的，可能需要更好的表述。因此，形态模型的设计原则非常重要。

8

在阿拉伯语中，深入研究在屈折变化和派生中起作用的形态过程，甚至所谓的不规则词，对精通整个形态和语音系统也是必要的。采用适当的抽象，不规则的形态可被看作只是在内在的或典型的规则词形上强制服从某些语音的扩充规则［15，16］。

例 1-7 hl rDyth? lm Orh. lm Or OHdA. هل رأيته؟ لم أره. لم أر أحدا.

hal ra'aytihi? lam 'arahu. lam 'ara 'aḥadan.

whether you-saw+him? not-did I-see+him. not-did I-see anyone.

在例 1-7 里，*ra'ayti* 是主动语态的第二人称阴性单数完成态动词，是有词根 *r'y* 的词 *ra'a* "to see" 的一个变化形式。其引用形式的典型规范模式是 *fa'al*，正如例 1-6 的词 *qara'*，或者，我们也可假设 *ra'ā* 的模式是 *fa'ā*，因此可简洁地断定最后的根辅音和其元音上下文应进行特定的语音变化，导致 *ra'a*（类似 *fa'a*）而不是 *ra'ay*（类似 *fa'al*）。引用形式的这种变化可能对整个词的形态行为产生影响。

表 1-1 显示了朴素的阿拉伯语的词结构模型和 Smrž［12］、Smrž、Bielický［17］提出

表 1-1 利用形音模板发现阿拉伯语形态的规则性。统一的结构操作适用于多种词干。表的行中，*qara'* "to read" 和 *ra'a* "to see" 及其屈折变化的表层形式 S 被分析为直接的 I 和形音的 M 模板。其中连字符标记了结构边界，此处要用合并规则。表外面的列对应于词典中声明的完成态（P）和未完成态（I）的词干，内列处理具有下列形态句法性质的主动态动词：I 陈述式、S 虚拟式、J 祈使式；1 第一人称、2 第二人称、3 第三人称；M 阳性、F 阴性；S 单数、P 复数

P-STEM	P-3MS	P-2FS	P-3MP	II2MS	IS1-S	II1-S	I-STEM
qara'	*qara'a*	*qara'ti*	*qara'ā*	*taqra'a*	*'aqra'a*	*'aqra'*	*qra'* S
fa'al	*fa'al-a*	*fa'al-ti*	*fa'al-ū*	*ta-f'al-a*	*'a-f'al-a*	*'a-f'al*	*f'al* I
fa'al	*fa'al-a*	*fa'al-ti*	*fa'al-ū*	*ta-f'al-a*	*'a-f'al-a*	*'a-f'al-*	*f'al* M
…	…-a	…-ti	…-ū	ta-…-a	a-…-a	a-…-	…
fa'a	*fa'ā-a*	*fa'ā-ti*	*fa'ā-ū*	*ta-fā-a*	*'a-fā-a*	*'a-fā-*	*fā* M
fa'ā	*fa'ā*	*fa'ā-ti*	*fa'aw*	*ta-fā*	*'a-fā*	*'a-fā*	*fā* I
ra'a	*ra'a*	*ra'ayti*	*ra'aw*	*tarā*	*'arā*	*'arā*	*rā* S

9

的包含形音合并规则和模板的模型之间的差别。形音模板通过组织词干模式和一般词缀来刻画形态过程，不需要词缀任何上下文相关的变化或词干的随意修改。合并规则非常简洁，确保这样的结构化表示可精确地转化为语言的表层形式。不管是书面形式还是语音形式。应用这些合并规则与除了包含在模板内的任何语法参数或信息是独立并且无关的。因此，大多数形态的不规则性被成功去除了。

与此相反，有些不规则性依附于具体的词和上下文，无法用通用的规则来说明。韩语的不规则动词有不少这样的例子。

韩语对语法词素的选择有很多例外。在其他的黏着语中很难找到不规则屈折变化的例子：日语中只有两个不规则动词[18]，芬兰语中只有1个[19]。这些语言中有大量的形态变化，可用精确的语音规则加以形式化。韩语还有和具体词相关的词干变化。和很多其他语言一样，i "be"和ha "do"有独特的不规则词尾。其他的不规则动词可由处于词干尾部的音加以分类。表1-2比较了在同样的语音条件下主要的不规则动词类和规则动词。

表1-2 韩语主要的不规则动词类和规则动词对比的实例

基本形式		(-e)		含义	注释
집-	cip-	집어	cip.e	'pick'	规则
깁-	kip-	기워	ki.we	'sew'	p-不规则
믿-	mit-	믿어	mit.e	'believe'	规则
싣-	sit-	실어	sil.e	'load'	t-不规则
씻-	ssis-	씻어	ssis.e	'wash'	规则
잇-	is-	이어	i.e	'link'	s-不规则
낳-	nah-	낳아	nah.a	'bear'	规则
까맣-	kka.mah-	까매	kka.may	'be black'	h-不规则
치르-	chi.lu-	치러	chi.le	'pay'	规则 u-ellipsis
이르-	i.lu	이르러	i.lu.le	'reach'	le-规则
흐르-	hu.lu-	흘러	hul.le	'flow'	lu-不规则

1.2.2 歧义性

形态歧义是指词形在其语篇上下文外可以以多种方式理解的可能性。同形看起来一样，但是具有不同的功能或意义的词称为同形词（homonyms）。

歧义存在于整个形态处理的各个方面，也存在于整个语言处理中。但是，形态分析并不需要对上下文中的词进行完全消歧，只是有效地限制一个给定词形的可能解释[20，21]。

在韩语中，同形词是形态分析中问题很多的地方之一，因为很多同形词是常用词。表1-3基于不同词尾的行为来排列同形词。例1-8是名词、动词同形词的例子。 [10]

表1-3 韩语中当动词与不同词尾结合时产生系统性的同形词

(-ko)		(-e)		(-un)		含义
묻고	mwut.ko	묻어	mwut.e	묻은	mwut.un	'bury'
묻고	mwut.ko	물어	mwul.e	물은	mwul.un	'ask'
물고	mwul.ko	물어	mwul.e	문	mwun	'bite'
걷고	ket.ko	걷어	ket.e	걷은	ket.un	'roll up'
걷고	ket.ko	걸어	kel.e	걸은	kel.un	'walk'
걸고	kel.ko	걸어	kel.e	건	ken	'hang'
굽고	kwup.ko	굽어	kwup.e	굽은	kwup.un	'be bent'
굽고	kwup.ko	구워	kwu.we	구운	kwu.un	'bake'
이르고	i.lu.ko	이르러	i.lu.le	이른	i.lun	'reach'
이르고	i.lu.ko	일러	il.le	이른	i.lun	'say'

例 1-8　난 'orchid'　←　난 *nan* 'orchid'
　　난 'I'　←　나 *na* 'I' + *-n* (topic)
　　난 'which flew'　←　날- *nal-* 'fly' + *-n* (relative, past)
　　난 'which got out'　←　나- *na-* 'get out' + *-n* (relative, past)

我们根据标准韩语词典考察名词 *nan* 的歧义：*nan*[1] "egg"，*nan*[2] "revvolt"，*nan*[3] "section (in newspaper)"，*nan*[6] "orchid"，还有其他不常用的意义。

阿拉伯语是在形态的派生和屈折变化方面都很丰富的语言。由于阿拉伯语字体通常不编码某些短元音，还省略可精确记录语音形式的某些变音符号，其形态歧义增加了不少。阿拉伯语的正字法把一些词形缩写在一起。阿拉伯语的形态消歧问题不但包括词的结构部件和形态句法性质的确定（即形态标注［22，23，24］），也包括切分、规范化、词形还原、词干化、变音符号还原［26，27，28］。

正如图 1-1 所示，在言语中当屈折变化的句法词合并在一起的时候，可能产生另外的语音和书写的变化。在梵语中，这样的一条语音规则称为外连音变读（sandhi）［29，30］，在切分阶段逆转连音变读通常是不确定的，因为有多个解决方案。在任何语言里，切分决策可能对重建的词元的形态句法性质加上限制，这些必须在进一步的处理中保持。形态和句法间的紧密结合启发人们提出了同时进行消歧而不是顺序地做［4］。

dirāsati	دراستي	drAsty	→	*dirāsatu ī*	دراسة ي	drAsp y
			→	*dirāsati ī*	دراسة ي	drAsp y
			→	*dirāsata ī*	دراسة ي	drAsp y
mwallimīya	معلمي	mElmy	→	*mwallimū ī*	معلمو ي	mElmw y
			→	*mwallimī ī*	معلمي ي	mElmy y
katabtumuhā	كتبتموها	ktbtmwhA	→	*katabtum hā*	كتبتم ها	ktbtm hA
ʾiǧrāʾuhu	إجراؤه	IjrAWh	→	*ʾiǧrāʾu hu*	إجراء ه	IjrA' h
ʾiǧrāʾihi	إجرائه	IjrAJh	→	*ʾiǧrāʾi hu*	إجراء ه	IjrA' h
ʾiǧrāʾahu	إجراءه	IjrA'h	→	*ʾiǧrāʾa hu*	إجراء ه	IjrA' h
li-ʾl-ʾasafi	للأسف	llOsf	→	*li ʾl-ʾasafi li*	ل الأسف	l AlOsf

图 1-1　阿拉伯语中谐音的复杂词元化和规范化。三种不同的名词格由同一个词形（*dirāsati* "my study" 与 *mwallimīya* "my teacher"）表示，但是原来的格结尾是不同的。在 *katabtumuhā* "you-MASC-PL wrote them" 里，当切分时，连读元音 *u* 被丢弃。在规范化有些书写约定时，例如 *ʾiǧrāʾ* "carrying out" 和附着的 *hu* "his" 保持格结尾之间的互相作用，或 *ʾasaf* "regret" 的定冠词和介词 *li* "for" 的合并，必须加以特别注意

捷克语是具有高度屈折变化的屈折语。与黏着语不同，屈折词素经常同时表示若干功能，在形式和功能之间没有特别的一一对应关系。捷克语中的屈折范式（paradigm）（即找出与要求的性质相联系的词形的方案）有很多种，但是几乎都包含非唯一的形式。

表 1-4 列出了几个常用的捷克语词的范式。名词的屈折变化范式依赖于词的语法上的性和语言结构。一个范式中的个别形式随着语法的数和格而变化，这些是只能在词使用的上下文中才能决定的自由参数。

看一下词 *stavení* "building" 的形态变化，我们也许会疑惑，既然这个词只能呈现 4 种不同的词形，为什么还要区别它所有的格呢？格系统的细节是否适当？回答是肯定的，因为我们能找到导致这种格范畴抽象的语言学证据。仅考虑在各种上下文中代替 *stavení* 的同义词，我们断定内在的系统的确作了一个格的区分，但是不一定以词的形式清晰且唯一地表达。

表 1-4 捷克语词 dům "house"、budova "building"、stavaba "building"、staveni "building" 的形态范式。尽管它们有系统性歧义，如不丢失包含其他地方的所有不同形式的能力，屈折变化参数的空间无法约简：S 单数、P 复数；1 主格、2 所有格、3 与格、4 宾格、5 呼格、6 位置格、7 工具格

	阳性非人称	阴 性	阳 性	中 性
S1	dům	budova	stavba	stavení
S2	domu	budovy	stavby	stavení
S3	domu	budově	stavbě	stavení
S4	dům	budovu	stavbu	stavení
S5	dome	budovo	stavbo	stavení
S6	domu/domě	budově	stavbě	stavení
S7	domem	budovou	stavbou	stavením
P1	domy	budovy	stavby	stavení
P2	domů	budov	staveb	stavení
P3	domům	budovám	stavbám	stavením
P4	domy	budovy	stavby	stavení
P5	domy	budovy	stavby	stavení
P6	domech	budovách	stavbách	staveních
P7	domy	budovami	stavbami	staveními

有些词或词类呈现出系统性的同形词的形态现象称为同态。与某些形态句法参数相关的中性化（neutralization）和零屈折变化（uninflectedness）可导致同形词。这些形态同态可由上下文要求讨论的形态句法性质的能力加以区分。正如 Baerman，Brown 和 Corbett 所说[10，32 页]：

中性化是形态中表现的句法无关性，而零屈折变化是形态对句法上相关的特征的不反应。

例如，在捷克语或阿拉伯语中，句法上要求第一人称阴性单数人称代词（等价于"I"）是合法的，尽管它与第一人称阳性单数同形。原因是，对人称范畴的其他值，阳性和阴性的形式是不同的，并且存在与性有关的句法依赖关系。并不是第一人称单数代词没有性，也不是既有阳性又有阴性。我们只是在这里看到了零屈折变化。另一方面，我们也许可以声称在英语或翻译里，如果性范畴存在，那么语法上是中性化的，*he* 和 *she*、*him* 和 *her*、*his* 和 *hers* 的细微差别是纯语义的。

我们已经知道了总聘和同态的概念，那么什么是覆盖一种语言中屈折变化多样性的形态句法屈折变化参数组合的最小集合呢？为多种语言定义一个内在的形态句法性质的联合系统的形态模型必须相应地一般化参数空间，并中性化任何系统的无效结构。

1.2.3 能产性

语言中词的总数是有限的还是无限的？这个问题直接导致了两种处理语言的基本方法，正如索绪尔对语言（langue）和言语（parole）的区分或乔姆斯基对语言能力和运用的二分所概括的。

一种观点：语言可被视为说出的或写出（运用）的所有言语的集合。这个理想的数据集在实践中可由语料库来近似。语料库是语言数据的有限集合，通常以经验方法来研究，开发语言模型的时候可进行比较。

但是，如果我们把语言考虑为一个系统，我们就会在其中发现一些结构手段，例如递归、重复、复合，可产生（能力）实在言语的无限集合。这种一般的能力对形态过程也成立，称为形态能产性[31，32]。

我们把语料库中发现的词形的集合称为词汇。这个集合的成员称为词型，而一个词形的每次原始实例称为词例。

词的分布[33]，或语言的其他元素遵从“80/20原则”，也称为“能者多劳”定律。就是说一个给定的语料库中最常用的词只占词汇表中很少的词型，词汇表中的其他词在语料中出现的次数很少。而且，当语料增大时，新词或没预料到的词总会出现。

在捷克语中，否定是一个能产的形态操作。动词、名词、形容词和副词可加上前缀 *ne* 以定义其词法概念的否定。在例1-9中，*budeš* “you will be” 是 *být* “to be” 的第二人称单数，*nebudu* “I will not be” 是 *nebýt*（*být* 的否定）的第一人称单数。我们有 *číst* “to read”，*nečíst* “not to read”，也可以创造像 *noviny nenoviny* 的副词短语，一般表达“对报纸漠不关心”：

例1-9　Budeš číst ty noviny? Budeš je číst? Nebudu je číst.
you-will read the newspaper? you-will it read? not-I-will it read.

例1-9和例1-1，例1-6的意思一样。词 *noviny* “newspaper” 只有复数形式，表示一张或很多报纸。我们可字面上把 *noviny* 翻译为 *novina* “news” 的复数从而看到该词的来源，恰好英语中也类似。

可以在词典中包含所有的否定词形，而且词汇的总数仍然是有限的。不过，通常语言形态系统的丰富性使得这种策略一点也不实用。

大多数语言包含允许其结构部件自由重复的词。考虑捷克语中与 “generation” 相关的前缀 *pra-*，在例1-10中可重复或不可重复的情况：

例1-10

vnuk ‘grandson’	*pravnuk* ‘great-grandson’
	prapra…vnuk ‘great-great-…grandson’
les ‘forest’	*prales* ‘jungle’, ‘virgin forest’
zdroj ‘source’	*pramen* ‘urquell’, ‘original source’
starý ‘old’	*prastarý* ‘time-honored’, ‘dateless’

在创造性的语言中，如博客、聊天、富有情绪的非正式交流，重复经常被用来加强表达的强度。创造性当然比能产性走得更远[32]。

我们给出一个例子，其中创造性、能产性和未登录词都很好地融合在一起。根据维基百科，词 *googol* 是造出来的词，表示数“1后面跟了100个0”，而Google公司的名字Google是其无意中的拼写错误。尽管如此，这两个词都成功地进入了英语词典，而在此形态的能产性开始起作用，因此我们现在有动词 *to google*，名词 *googling*，甚至 *googlish* 或 *googleology*[34]。

Google这个词也被其他语言采纳，每种语言自己的形态过程也被激发。在捷克语中，人们说 *googlovat*，*googlit* “to google” 或 *vygooglovat*，*vygooglit* “*to google out*”，*googlování* “googling” 等。在阿拉伯语中，上面两个词被写作 *ǧūǧul* “googol” 和 *ǧugil* “Google”，后者通过内部屈折变化又变为动词 *gawgal* “to google”，好像有一个真的词根 *gwgl*，并且相应的名词 *gawgalah* “googling” 也存在。

1.3　形态模型

有很多方法设计并实现形态模型。多年来，计算语言学家已经看到了若干形式体系和框架的发展，尤其是多种不同表达能力的文法，用于处理自然语言或人工语言的系列问题。

多种领域相关的程序语言也被发明出来，允许我们直观地用最小的编程量来实现理论

问题。这些专用语言通常引入特殊的程序记号，用某些受限的计算模型加以解释。这样做的动机是因为当初的计算资源很有限，而要解决的任务发要求很高，复杂度很大。当然也有理论的动机，如找到一个简单、精确且一般的模型是科学抽象的追求。

也有很多方法不用领域相关的编程，当然也要考虑运行时性能和计算模型本身的效率。编程方法和设计风格的选择决定这些模型最终是否是纯粹、直观、充分、完备、可重用、优美等。

现在让我们看看处理形态最著名的几个计算方法。毫无疑问，这种分类肯定不是排他的，因为综合的形态模型及其应用能合并多个不同的实现方面，见下面的讨论。

1.3.1 查词典

形态分析是语言的词形和相应的语言学描述相联系的过程。一个一个地枚举这些联系的形态系统没有任何的一般化手段。词形的分析被简化为在词表、词典或数据库中进行字面查找的系统也同样如此，除非系统是根据更先进的语言学模型建造和同步发展的。

在本节，将词典理解为一种数据结构，可以直接得到一些预先计算的结果，也即词的分析。为了高效的查找，数据结构可以被优化，结果也可以共享。查找操作相对简单，通常也很快。词典可以被实现为表、二叉搜索树、trie树、哈希表等。 15

因为词形和其期望的描述的联系集合是由简单的枚举声明的，模型的范围是有限的，语言的生成潜力没有被利用。开发并且验证联系表是枯燥的、易错的，也可能是低效的或不精确的，除非数据是从大而可靠的语言资源中自动检索而来。

尽管如此，对于一个给定的目的，枚举模型经常是足够的，可容易地处理例外，也能实现复杂的形态分析。例如，韩语基于词典的方法 [35] 依赖于含有所有可能的变体词素和形态变化的一个大词典。不过，这些方法不允许开发可重用的形态规则 [36]。

对于很多语言，词表或基于词典的方法在许多特定的实现中经常使用。我们也可以假设，随着大规模在线数据的易获得性，目前抽取词形的高覆盖率的词汇表是可行的 [37]。联系的标注如何构建、有多精确的问题仍然存在。关于无监督学习和形态归纳（导致结构化、非枚举模型的方法）的参考文献在本章后面描述。

1.3.2 有限状态形态

通过有限状态形态模型，程序员可以将写的描述直接编译为有限状态转录机。两个最流行的工具：XFST（施乐有限状态工具）[9] 和 LexTools [11]㊀ 都是这类，文献中引用不少，多种语言的样本实现也可网上获得。

有限状态转录机是扩充有限状态自动机能力的计算手段，由有限个节点构成，节点之间有有向边，边上标了一对输入、输出符号。在这个网络或图里，节点也称作状态，边也称作弧。沿着弧从初态集走到终态集等价于读入遇到的输入符号序列并写出相应的输出符号序列。

转录机接受的所有可能的序列集合称作输入语言，输出的所有可能的序列集合称作输出语言。例如，一个有限状态转录机可把无限的正规语言 *vnuk*，*pravnuk*，*prapravnuk*，… 翻译为无限的正规语言 *grandson*，*great grandson*，*great great grandson*，… 16

有限状态转录机的作用是描述并且计算集合之间的正规关系（regular relation）[38，

㊀ 分别参见 http://www.fsmbook.com/ 和 http://compling.ui.uiuc.edu/catms/。

9，11]⊖，也即转录机说明输入、输出语言间的关系。事实上，可以逆转关系的定义域和值域，也即输入、输出。在有限状态计算形态学的术语中，通常把输入词形称为表层串（surface string），输出描述称为词法串（lexical string）。如果转录机是用来做形态分析，或者反过来，用来做形态生成。

我们为词形及其部件给出的语言学描述可以是任意的，并且显然依赖于处理的语言和采用的形态理论。例如，在英语中，一个有限状态转录机可将表层串 children 分析为词法串 child [+plural]，或者，从 woman [+plural] 生成 women。参见例 1.8 或图 1-1 的其他输入、输出串的例子。

语言上的关系也可以被视为函数。假设我们有关系 $\mathcal{R}$，用 $[\Sigma]$ 表示有限符号集 Σ 上的所有序列的集合，那么 $\mathcal{R}$ 的定义域和值域都是 $[\Sigma]$ 的子集。我们可把 $\mathcal{R}$ 看作是从输入串到输出串集的映射。用如下的公式表示，其中 $[\Sigma]$ 等于 *String*：

$$\mathcal{R}::[\Sigma] \rightarrow 2^{[\Sigma]} \quad \mathcal{R}::String_1 \rightarrow 2^{String_2} \tag{1.1}$$

有限状态转录机的代数性质已经被深入研究，其模型已被证明对其他问题也很适用[9]。用来把表层串（而非词法串）的联系编码为语言和形态的重写规则（rewrite rules）早就见于两层形态模型[39]，后来还在形态和句法的计算方法[11]和形态和计算[40]中有进一步的研究。

人类语言中的形态操作和过程在大多数情况下能用有限状态的形式进行充分的描述。Beesley 和 Karttunen [9] 强调把转录机串联起来作为把表层和词法语言分解为更简单的模型的方法，并且提出了一个不太系统的**编译-替换**（compile-replace）转录操作以处理形态中的非拼接现象。但是，Roark 和 Sproat [11] 论证了用转录机的复合来构造一般的形态模型是更加纯粹通用的做法。

形态的有限状态模型的一个理论限制是描述若干人类语言中的词及其要素的**重复**问题（例如表示复数）。只包含具有形如 λ^{1+k} 的词的语言，其中 λ 是字母表的任意符号序列，$k\in\{1,2,\cdots\}$ 是任意自然数，表示 λ 在本身后的重复次数，不是一个正规语言，甚至不是一个上下文无关语言。不限定长度的串的重复，因此不是一个正规语言操作。Roark 和 [17] Sproat 讨论了在有限状态转录机的框架下如何处理这个问题[11]。

有限状态技术能直接适用于处理孤立语和黏着语的形态建模。韩语的有限状态模型由 Kim 等 [41]、Lee 和 Rim [42]、Han [43] 等讨论，仅举几例。在有限状态框架下处理非拼接的形态，特别参见 Kay [44]、Beesley [45]、Kiraz [46]，以及 Habash、Rambow 和 Kiraz [47]。如要比较捷克语的丰富形态的有限状态模型，可参考 Skoumalová [48] 或 Sedláček 和 Smrž [49]。

实现一个精细的有限状态形态模型要求仔细调试词典、重写规则和其他部件，而扩充代码可能带来不可预料的交叉影响，正如 Oazer 指出的 [50]。上面所述的方便的描述语言是必要的，因为直接对有限状态转录机进行编码是极其繁重、易错、难懂的。

大多数程序语言都以支持正则表达式匹配或替换的方式提供了有限状态工具。这些不一定是开发完备的形态分析器或自然语言产生器的最终选择，然而的确很适宜开发词切分程序，形态猜测程序，可对完整的形态分析中遇到的正确形成的、却不能找到其对应词条的词的结构提出建议[9]。

⊖ 正规关系和正规语言在结构上被有限内存所限（例如，可能出现的有界的配整集）。与正则语言不一样的是，通常情况下正规关系的交集可能产生非正规的结果[38]。

1.3.3 基于合一的形态

基于合一的形态分析是受旨在提供人类语言的完整语言描述的各种形式的语言学框架（尤其是中心词驱动的短语结构文法（Head driven Phrase Structure Grammar，HPSG）[51]）以及词法知识表示语言的开发（尤其是 DATR [52]）的启发而提出，这些形式体系的概念和方法经常和逻辑程序设计紧密联系在一起。在 Erjavec [53] 的优秀论文中，科学背景得到了深入而广泛的讨论，参见 Carpenter [54] 和 Shieber [55] 的专著。

在有限状态形态模型中，表层和词形都是原子符号的非结构串。在高级体系中，语言信息由更适当的数据结构所表达，这种结构可以包含更复杂的值，必要时可以嵌套。因此形态分析 $\mathcal{P}$ 把线性形式 ϕ 和结构化内容 ψ 联系起来，比较（1.1）：

$$\mathcal{P}::\phi \rightarrow \langle \psi \rangle \qquad \mathcal{P}::form \rightarrow \langle content \rangle \tag{1.2}$$

Erjavec [53] 论证，在形态建模中，词形最适宜用正则表达式描述，而语言内容最适宜用**类型特征结构**（typed feature structure）来描述。特征结构可被视为有向无环图。特征结构中的节点由多个属性组成，而属性值又可以是特征结构。节点具有类型，原子值是由类型区分的无属性节点。为了避免每个地方都用唯一值的实例，可引入指针。特征结构通常可显示为“属性-值”矩阵或嵌套的符号表达式。

特征结构可通过合一操作合并成更详细的特征结构。合一操作也可能失败，这通常意味着其信息不兼容。依赖于处理逻辑，合一可以是单调的（信息保持），也可允许缺省值的继承或重写，不管是哪一种情况，模型中的信息可通过定义在特征结构类型上的继承体系高效地共享和重用。 18

这种形态模型典型地被形式化为逻辑程序，用合一来解决模型引入的限制。这种方法的优点是包含更好的抽象能力，以开发形态文法，同时消除冗余信息。

然而，用 DATR 实现的形态模型在某些假设下，可被转化为有限状态机，因此在形式上它们在描述形态现象上是等价的 [11]。有趣的是，一层语音模型 [56] 把语音限制描述为逻辑表达式，而逻辑表达式能被编译为有限状态自动机，该自动机可和形态转录机求交集，排除造成干扰的语音上无效的表层串 [参见 57，53]。

基于合一的模型已经在很多语言实现，如俄语 [58]、捷克语 [59]、斯洛文尼亚语 [53]、波斯语 [60]、希伯来语 [61]、阿拉伯语 [62，63] 等。有些依赖于 DATR，有些采用、改写或开发其他合一引擎。

1.3.4 函数式形态

这类形态模型不但包括那些遵循函数式形态 [64] 的方法学形态模型，也包括相关的如文法框架的形态资源文法 [65]。函数式形态用函数式编程和类型论的原理来定义模型，把形态操作和过程看作纯数学函数，并把模型的语言学和抽象元素组织为值的不同类型和类型的类。

虽然函数式形态不限于为人类语言的特别形态类型建模，但它尤其适用于屈折语。语言学概念，如范式、规则和例外、文法范畴和参数、词、语素、形元等可用这种方法直观且简洁地描述。通过计算设定可以精确而优美地设计一个形态系统，支持子问题的逻辑分解，通过强类型检查保证一个程序的语义结构。

函数式形态的实现一般被作为程序库重用，可处理语言的完整形态，也可集成到各类应用。形态分析只是系统的一种用法，其他如形态生成、词典浏览等。紧接着分析式（1.2），

我们可将屈折变化 $\mathcal{I}$、派生 $\mathcal{D}$、查找 $\mathcal{L}$ 描述为这些通用类型：

$$\mathcal{I}::lexeme \rightarrow \{parameter\} \rightarrow \{form\} \tag{1.3}$$

$$\mathcal{D}::lexeme \rightarrow \{parameter\} \rightarrow \{lexeme\} \tag{1.4}$$

19

$$\mathcal{L}::content \rightarrow \{lexeme\} \tag{1.5}$$

函数式形态模型可以在必要时编译成有限状态转录机，也可以交互地解释执行。模型的计算可利用惰性计算，并且采用其他高效分析、查找等方法［参见 66，12］。

许多函数式形态实现嵌入在通用的程序语言中，采用高级编程技术，从而可给程序员更多自由，并且允许他们开发真实世界的完整应用。梵语形态的禅工具包［67，68］使用 OCaml 书写。它影响了用 Haskell 的函数式形态框架［64］，该框架已经实现了拉丁语、瑞典语、西班牙语、乌尔都语［69］和其他语言的形态分析。

在 Haskell 中，开发者可利用其语法灵活性，设计用来处理给定问题的函数结构的记号体系。这个体系构成了所谓的领域相关的嵌入式语言，使得编程更有趣。图 1-2 解释了阿拉伯语形态的 ElixirFM 实现［12，17］，定义了词的结构和词典。尽管词条显得信息量多，但其格式非常类似于纸质词典。>|、|<、<<之类的运算符和 verb 之类的标签就是中缀函数，FaCY、FCI、At 之类的模式和词缀是数据构造子。

```
|> "d r y" <| [                                                  d r y دري
FaCY                          'verb'   [ "know", "notice" ]        fa-a
    'imperf'  FCI,                                                  fi
FACY                          'verb'   [ "flatter", "deceive" ],    fa-a
HaFCY                         'verb'   [ "inform", "let know" ],    ufa
IA >| "'a" >>| FCI |<< "ly"            'adj'    [ "agnostic" ],    lā-a fā-ly
FiCAL |< aT                   'noun'   [ "knowledge", "knowing" ],  fi-al-ah
MuFACY |< aT                  'noun'   [ "flattery" ]               mufā-ah
    'plural'  MuFACY |< At,                                         mufā-at
FACI                          'adj'    [ "aware", "knowing" ] ]      fā'i

know, notice          I (i)   dirā دري   knowledge, knowing   dirāyah دراية
flatter, deceive      III     dārā داري   flattery            mudārāh مداراة
inform, let know      IV      adrā أدري                       (mudārayāt) (مداريات)
agnostic              lā-adrīy لاأدري   aware, knowing        dārin دار
```

图 1-2　ElixirFM 的一个词典片段和从其产生的一个版式。嵌套在词根 d r y 下面的词条源代码以等宽字体显示。注意定制记号、声明的精简性和大信息量

23

即使没有通用程序语言提供的多选择，函数式形态模型也能获得高度抽象。文法框架的形态文法［65］可用语言的句法和语义结构来扩充。文法框架本身支持多语言，十几种语言的模型已可作为开源软件获得［70，71］。

OpenCCG 项目的文法［72］也可被视为函数式模型。该形式体系区分特征、范畴、世系的声明，提供类似类型系统的手段以表示结构值和继承体系。文法非常倚重这个功能定义以参数化宏来最小化模型的冗余性，并做出要求的一般化。源代码的宏扩展和内联函数类似。文法的原始文本被简化为描述词形及其形态句法和词法性质的联系。

1.3.5　形态归纳

迄今为止，我们一直在假设我们知道在找什么的基础上找出多种语言中词的结构。我

们还没有考虑在没有人干预的情形下（即以无监督或半监督的方式）发现并归纳词的结构。这种方法的动机是，对于很多语言，我们可能无法得到足够的语言学专业知识，满足某种目的的实现可能根本不存在。形态和词法信息的自动获取，即使不很完美，也可以用来初始化或改进经典的形态模型。

让我们简略地看一下该领域的研究方向。在 Hammarström [73] 和 Goldsmith [74] 的研究中，详细地综述了形态的无监督学习的文献。Hammarström 把多种方法划分为三组。有些工作比较词，并根据由各种各样的统计学而获得的相似性进行聚类 [75, 76, 77, 78]；有些试图识别词的显著特征，使之和不相关的词区分开来。大部分发表的方法把形态归纳的问题视为词边界和词素边界检查，有时也自动获取词典及其范式 [79, 80, 81, 82, 83][⊖]。

从词形及其上下文中推断出词结构有多个挑战性的难点，如形态的歧义 [76] 和不规则性 [75]、书写和语音的变化 [85]，以及非线性形态过程 [86, 87]。

为了改进统计推理的性能，Snyder 和 Barzilay [88] 提出了多种语言的并行形态学习，从而导致了抽象词素的发现。Poon, Cherry 和 Toutanova [89] 的判别性对数线性模型在进行切分决策时，通过采用交叉的上下文特征提高了其一般化能力 [参见 90]。 21

1.4 总结

本章中，我们认识到形态可从相对的观点进行研究：一种观点是通过词的构成找出词的结构化部件，另一种是句法驱动的观点，其中词的功能才是关注的焦点。一种观点重视形态分析，另一种关注形态生成。一种强调人工的形态框架，另一种注重无监督形态归纳系统。另外，其他问题包括形态模型的实现的优劣和难易。

我们把形态分析描述为从符号的线性序列中得出结构化信息的形式过程，其中存在歧义，而且可能有多种解释。

我们探索了不同类型语言的有趣的形态现象，也对多语言处理和模型开发给出若干提示。

我们看到在韩语中，语音规则调节的黏着过程是一种主导的形态过程，一种有效的词分解模型可工作于词素层次，而不管词素是词法的还是语法的。

在捷克语和阿拉伯语等屈折语中，有复杂的屈折变化和派生变化参数，以及词法相关的词干变形。分解的方法不太有用。形态最好通过范式来描述，把词可能的形式和其相应的性质联系起来。

我们讨论了用现代编程技术实现以下这些模型的多种方法。

致谢

感谢 Petr Novák 对本章初稿给出的有价值的意见。

参考文献

[1] M. Liberman, "Morphology." Linguistics 001, Lecture 7, University of Pennsylvania, 2009. http://www.ling.upenn.edu/courses/Fall_2009/ling001/morphology.html.

[2] M. Haspelmath, "The indeterminacy of word segmentation and the nature of morphology and syntax." *Folia Linguistica*, vol. 45, 2011.

⊖ 可将这些和把词用空字符断开的平凡方法进行比较 [84]。

[3] H. Kučera and W. N. Francis, *Computational Analysis of Present-Day American English*. Providence, RI: Brown University Press, 1967.

[4] S. B. Cohen and N. A. Smith, "Joint morphological and syntactic disambiguation," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 208–217, 2007.

[5] T. Nakagawa, "Chinese and Japanese word segmentation using word-level and character-level information," in *Proceedings of 20th International Conference on Computational Linguistics*, pp. 466–472, 2004.

[6] H. Shin and H. You, "Hybrid n-gram probability estimation in morphologically rich languages," in *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation*, 2009.

[7] D. Z. Hakkani-Tür, K. Oflazer, and G. Tür, "Statistical morphological disambiguation for agglutinative languages," in *Proceedings of the 18th Conference on Computational Linguistics*, pp. 285–291, 2000.

[8] G. T. Stump, *Inflectional Morphology: A Theory of Paradigm Structure*. Cambridge Studies in Linguistics, New York: Cambridge University Press, 2001.

[9] K. R. Beesley and L. Karttunen, *Finite State Morphology*. CSLI Studies in Computational Linguistics, Stanford, CA: CSLI Publications, 2003.

[10] M. Baerman, D. Brown, and G. G. Corbett. *The Syntax–Morphology Interface. A Study of Syncretism*. Cambridge Studies in Linguistics, New York: Cambridge University Press, 2006.

[11] B. Roark and R. Sproat, *Computational Approaches to Morphology and Syntax*. Oxford Surveys in Syntax and Morphology, New York: Oxford University Press, 2007.

[12] O. Smrž, "Functional Arabic morphology. Formal system and implementation," PhD thesis, Charles University in Prague, 2007.

[13] H. Eifring and R. Theil, *Linguistics for Students of Asian and African Languages*. Universitetet i Oslo, 2005.

[14] B. Bickel and J. Nichols, "Fusion of selected inflectional formatives & exponence of selected inflectional formatives," in *The World Atlas of Language Structures Online* (M. Haspelmath, M. S. Dryer, D. Gil, and B. Comrie, eds.), ch. 20 & 21, Munich: Max Planck Digital Library, 2008.

[15] W. Fischer, *A Grammar of Classical Arabic*. Trans. Jonathan Rodgers. Yale Language Series, New Haven, CT: Yale University Press, 2002.

[16] K. C. Ryding, *A Reference Grammar of Modern Standard Arabic*. New York: Cambridge University Press, 2005.

[17] O. Smrž and V. Bielický, "ElixirFM." Functional Arabic Morphology, SourceForge.net, 2010. http://sourceforge.net/projects/elixir-fm/.

[18] T. Kamei, R. Kōno, and E. Chino, eds., *The Sanseido Encyclopedia of Linguistics, Volume 6 Terms* (in Japanese). Sanseido, 1996.

[19] F. Karlsson, *Finnish Grammar*. Helsinki: Werner Söderström Osakeyhtiö, 1987.

[20] J. Hajič and B. Hladká, "Tagging inflective languages: Prediction of morphological categories for a rich, structured tagset," in *Proceedings of COLING-ACL 1998*, pp. 483–490, 1998.

[21] J. Hajič, "Morphological tagging: Data vs. dictionaries," in *Proceedings of NAACL-ANLP 2000*, pp. 94–101, 2000.

[22] N. Habash and O. Rambow, "Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pp. 573–580, 2005.

[23] N. A. Smith, D. A. Smith, and R. W. Tromble, "Context-based morphological disambiguation with random fields," in *Proceedings of HLT/EMNLP 2005*, pp. 475–482, 2005.

[24] J. Hajič, O. Smrž, T. Buckwalter, and H. Jin, "Feature-based tagger of approximations of functional Arabic morphology," in *Proceedings of the 4th Workshop on Treebanks and Linguistic Theories (TLT 2005)*, pp. 53–64, 2005.

[25] T. Buckwalter, "Issues in Arabic orthography and morphology analysis," in *COLING 2004 Computational Approaches to Arabic Script-based Languages*, pp. 31–34, 2004.

[26] R. Nelken and S. M. Shieber, "Arabic diacritization using finite-state transducers," in *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, pp. 79–86, 2005.

[27] I. Zitouni, J. S. Sorensen, and R. Sarikaya, "Maximum entropy based restoration of Arabic diacritics," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 577–584, 2006.

[28] N. Habash and O. Rambow, "Arabic diacritization through full morphological tagging," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pp. 53–56, 2007.

[29] G. Huet, "Lexicon-directed segmentation and tagging of Sanskrit," in *Proceedings of the XIIth World Sanskrit Conference*, pp. 307–325, 2003.

[30] G. Huet, "Formal structure of Sanskrit text: Requirements analysis for a mechanical Sanskrit processor," in *Sanskrit Computational Linguistics: First and Second International Symposia* (G. Huet, A. Kulkarni, and P. Scharf, eds.), vol. 5402 of *LNAI*, pp. 162–199, Berlin: Springer, 2009.

[31] F. Katamba and J. Stonham, *Morphology*. Basingstoke: Palgrave Macmillan, 2006.

[32] L. Bauer, *Morphological Productivity*, Cambridge Studies in Linguistics. New York: Cambridge University Press, 2001.

[33] R. H. Baayen, *Word Frequency Distributions*, Text, Speech and Language Technology. Boston: Kluwer Academic Publishers, 2001.

[34] A. Kilgarriff, "Googleology is bad science," *Computational Linguistics*, vol. 33, no. 1, pp. 147–151, 2007.

[35] H.-C. Kwon and Y. S. Chae, "A dictionary-based morphological analysis," in *Proceedings of Natural Language Processing Pacific Rim Symposium*, pp. 178–185, 1991.

[36] D.-B. Kim, K.-S. Choi, and K. H. Lee, "A computational model of Korean morphological analysis: A prediction-based approach," *Journal of East Asian Linguistics*, vol. 5, no. 2, pp. 183–215, 1996.

[37] A. Halevy, P. Norvig, and F. Pereira, "The unreasonable effectiveness of data," *IEEE Intelligent Systems*, vol. 24, no. 2, pp. 8–12, 2009.

[38] R. M. Kaplan and M. Kay, "Regular models of phonological rule systems," *Computational Linguistics*, vol. 20, no. 3, pp. 331–378, 1994.

[39] K. Koskenniemi, "Two-level morphology: A general computational model for word form recognition and production," PhD thesis, University of Helsinki, 1983.

[40] R. Sproat, *Morphology and Computation*. ACL–MIT Press Series in Natural Language Processing. Cambridge, MA: MIT Press, 1992.

[41] D.-B. Kim, S.-J. Lee, K.-S. Choi, and G.-C. Kim, "A two-level morphological analysis of Korean," in *Proceedings of the 15th International Conference on Computational Linguistics*, pp. 535–539, 1994.

[42] S.-Z. Lee and H.-C. Rim, "Korean morphology with elementary two-level rules and rule features," in *Proceedings of the Pacific Association for Computational Linguistics*, pp. 182–187, 1997.

[24]

[43] N. R. Han, "Klex: A finite-state transducer lexicon of Korean," in *Finite-state Methods and Natural Language Processing: 5th International Workshop, FSMNLP 2005*, pp. 67–77, Springer, 2006.

[44] M. Kay, "Nonconcatenative finite-state morphology," in *Proceedings of the Third Conference of the European Chapter of the ACL (EACL-87)*, pp. 2–10, ACL, 1987.

[45] K. R. Beesley, "Arabic morphology using only finite-state operations," in *COLING-ACL'98 Proceedings of the Workshop on Computational Approaches to Semitic languages*, pp. 50–57, 1998.

[46] G. A. Kiraz, *Computational Nonlinear Morphology with Emphasis on Semitic Languages*. Studies in Natural Language Processing, Cambridge: Cambridge University Press, 2001.

[47] N. Habash, O. Rambow, and G. Kiraz, "Morphological analysis and generation for Arabic dialects," in *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, pp. 17–24, 2005.

[48] H. Skoumalová, "A Czech morphological lexicon," in *Proceedings of the Third Meeting of the ACL Special Interest Group in Computational Phonology*, pp. 41–47, 1997.

[49] R. Sedláček and P. Smrž, "A new Czech morphological analyser ajka," in *Text, Speech and Dialogue*, vol. 2166, pp. 100–107, Berlin: Springer, 2001.

[50] K. Oflazer, "Computational morphology." ESSLLI 2006 European Summer School in Logic, Language, and Information, 2006.

[51] C. Pollard and I. A. Sag, *Head-Driven Phrase Structure Grammar*. Chicago: University of Chicago Press, 1994.

[52] R. Evans and G. Gazdar, "DATR: A language for lexical knowledge representation," *Computational Linguistics*, vol. 22, no. 2, pp. 167–216, 1996.

[53] T. Erjavec, "Unification, inheritance, and paradigms in the morphology of natural languages." PhD thesis, University of Ljubljana, 1996.

[54] B. Carpenter, *The Logic of Typed Feature Structures*. Cambridge Tracts in Theoretical Computer Science 32, New York: Cambridge University Press, 1992.

[55] S. M. Shieber, *Constraint-Based Grammar Formalisms: Parsing and Type Inference for Natural and Computer Languages*. Cambridge, MA: MIT Press, 1992.

[56] S. Bird and T. M. Ellison, "One-level phonology: Autosegmental representations and rules as finite automata," *Computational Linguistics*, vol. 20, no. 1, pp. 55–90, 1994.

[57] S. Bird and P. Blackburn, "A logical approach to Arabic phonology," in *Proceedings of the 5th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 89–94, 1991.

[58] G. G. Corbett and N. M. Fraser, "Network morphology: A DATR account of Russian nominal inflection," *Journal of Linguistics*, vol. 29, pp. 113–142, 1993.

[59] J. Hajič, "Unification morphology grammar. Software system for multilanguage morphological analysis," PhD thesis, Charles University in Prague, 1994.

[60] K. Megerdoomian, "Unification-based Persian morphology," in *Proceedings of CICLing 2000*, 2000.

[61] R. Finkel and G. Stump, "Generating Hebrew verb morphology by default inheritance hierarchies," in *Proceedings of the Workshop on Computational Approaches to Semitic Languages*, pp. 9–16, 2002.

[62] S. R. Al-Najem, "Inheritance-based approach to Arabic verbal root-and-pattern morphology," in *Arabic Computational Morphology. Knowledge-based and Empirical Methods* (A. Soudi, A. van den Bosch, and G. Neumann, eds.), vol. 38, pp. 67–88, Berlin: Springer, 2007.

[63] S. Köprü and J. Miller, "A unification based approach to the morphological analysis and generation of Arabic," in *CAASL-3: Third Workshop on Computational Approaches to Arabic Script-based Languages*, 2009.

[64] M. Forsberg and A. Ranta, "Functional morphology," in *Proceedings of the 9th ACM SIGPLAN International Conference on Functional Programming, ICFP 2004*, pp. 213–223, 2004.

[65] A. Ranta, "Grammatical Framework: A type-theoretical grammar formalism," *Journal of Functional Programming*, vol. 14, no. 2, pp. 145–189, 2004.

[66] P. Ljunglöf, "Pure functional parsing. An advanced tutorial," Licenciate thesis, Göteborg University & Chalmers University of Technology, 2002.

[67] G. Huet, "The Zen computational linguistics toolkit," ESSLLI 2002 European Summer School in Logic, Language, and Information, 2002.

[68] G. Huet, "A functional toolkit for morphological and phonological processing, application to a Sanskrit tagger," *Journal of Functional Programming*, vol. 15, no. 4, pp. 573–614, 2005.

[69] M. Humayoun, H. Hammarström, and A. Ranta, "Urdu morphology, orthography and lexicon extraction," in *CAASL-2: Second Workshop on Computational Approaches to Arabic Script-based Languages*, pp. 59–66, 2007.

[70] A. Dada and A. Ranta, "Implementing an open source Arabic resource grammar in GF," in *Perspectives on Arabic Linguistics* (M. A. Mughazy, ed.), vol. XX, pp. 209–231, John Benjamins, 2007.

[71] A. Ranta, "Grammatical Framework." Programming Language for Multilingual Grammar Applications, http://www.grammaticalframework.org/, 2010.

[72] J. Baldridge, S. Chatterjee, A. Palmer, and B. Wing, "DotCCG and VisCCG: Wiki and programming paradigms for improved grammar engineering with OpenCCG," in *Proceedings of the Workshop on Grammar Engineering Across Frameworks*, 2007.

[73] H. Hammarström, "Unsupervised learning of morphology and the languages of the world," PhD thesis, Chalmers University of Technology and University of Gothenburg, 2009.

[74] J. A. Goldsmith, "Segmentation and morphology," in *Computational Linguistics and Natural Language Processing Handbook* (A. Clark, C. Fox, and S. Lappin, eds.), pp. 364–393, Chichester: Wiley-Blackwell, 2010.

[75] D. Yarowsky and R. Wicentowski, "Minimally supervised morphological analysis by multimodal alignment," in *Proceedings of the 38th Meeting of the Association for Computational Linguistics*, pp. 207–216, 2000.

[76] P. Schone and D. Jurafsky, "Knowledge-free induction of inflectional morphologies," in *Proceedings of the North American Chapter of the Association for Computational Linguistics*, pp. 183–191, 2001.

[77] S. Neuvel and S. A. Fulop, "Unsupervised learning of morphology without morphemes," in *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*, pp. 31–40, 2002.

[78] N. Hathout, "Acquistion of the morphological structure of the lexicon based on lexical similarity and formal analogy," in *Coling 2008: Proceedings of the 3rd Textgraphs Workshop on Graph-based Algorithms for Natural Language Processing*, pp. 1–8, 2008.

[79] J. Goldsmith, "Unsupervised learning of the morphology of a natural language," *Computational Linguistics*, vol. 27, no. 2, pp. 153–198, 2001.

[80] H. Johnson and J. Martin, "Unsupervised learning of morphology for English and Inuktitut," in *Companion Volume of the Proceedings of the Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics 2003: Short Papers*, pp. 43–45, 2003.

[81] M. Creutz and K. Lagus, "Induction of a simple morphology for highly-inflecting languages," in *Proceedings of the 7th Meeting of the ACL Special Interest Group in Computational Phonology*, pp. 43–51, 2004.

[82] M. Creutz and K. Lagus, "Unsupervised models for morpheme segmentation and morphology learning," *ACM Transactions on Speech and Language Processing*, vol. 4, no. 1, pp. 1–34, 2007.

[83] C. Monson, J. Carbonell, A. Lavie, and L. Levin, "ParaMor: Minimally supervised induction of paradigm structure and morphological analysis," in *Proceedings of Ninth Meeting of the ACL Special Interest Group in Computational Morphology and Phonology*, pp. 117–125, 2007.

[84] F. M. Liang, "Word Hy-phen-a-tion by Com-put-er," PhD thesis, Stanford University, 1983.

[85] V. Demberg, "A language-independent unsupervised model for morphological segmentation," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 920–927, 2007.

[86] A. Clark, "Supervised and unsupervised learning of Arabic morphology," in *Arabic Computational Morphology: Knowledge-based and Empirical Methods* (A. Soudi, A. van den Bosch, and G. Neumann, eds.), vol. 38, pp. 181–200, Berlin: Springer, 2007.

[87] A. Xanthos, *Apprentissage automatique de la morphologie: le cas des structures racine-schème*. Sciences pour la communication, Bern: Peter Lang, 2008.

[88] B. Snyder and R. Barzilay, "Unsupervised multilingual learning for morphological segmentation," in *Proceedings of ACL 08: HLT*, pp. 737–745, 2008.

[89] H. Poon, C. Cherry, and K. Toutanova, "Unsupervised morphological segmentation with log-linear models," in *Proceedings of Human Language Technologies: Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 209–217, 2009.

[90] S. Della Pietra, V. Della Pietra, and J. Lafferty, "Inducing features of random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 380–393, 1997.

第2章

Multilingual Natural Language Processing Applications: From Theory to Practice

找出文档的结构

Dilek Hakkani-Tür, GokhanTur, Benoit Favre, Elizabeth Shriberg

2.1 概述

在人类语言中，词和句子一般会具有结构，它们并不随机出现。例如，词可以组成句子——一个具有完整意义的语法单元，如陈述、请求、命令等。同样，在书写文本中，句子可以组成段落——一个关于某个观点或想法的自我包含的语篇单元。通过显式地使用"因此"这样的连词，句子之间可以互相关联。

自动提取文档结构对随后的自然语言处理（Natural Language Processing，NLP）任务很有帮助。例如，句法分析、机器翻译和语义角色标注均使用句子作为基本处理单元[1，2]。句子边界标注对提高人类理解自动语音识别（Automatic Speech Recognition，ASR）系统的输出有很大的帮助。另外，将输入文本或对话按照主题分割成块也可以使数据的组织与索引变得更好。例如，与特定主题相关的片段可以从长对话中提取。同样，属于相同主题的文章可以归类并做进一步的处理。由于书写与口头信息的负载问题日益增加，在大多数音频与语言处理应用中，提取文本以及音频文档的结构是极其有意义甚至有时是必不可少的一步。

在此，我们讨论找出文本结构的方法。为简单起见，只有与主题相关的句子和句子组被认为是结构部件。

在本章中，我们把判断一个给定字符序列中句子开始与结束的任务称为**句子边界检测**（sentence boundary detection）。类似地，我们把判断一个给定句子序列中主题开始与结束的任务称为**主题分割**（topic segmentation）。我们用统计分类方法进行分割[⊖]，该方法在给定训练集后推断出句子与话题边界的存在与否。这些方法使用输入数据的特征来进行预测。特征包含句子或主题边界存在与否的证据，如标点符号、对话中的停顿，以及文章中的新词。特征是分类方法的核心，只有通过特征的精心设计与选择，才能防止过拟合与噪声问题，进而取得成功。

应该注意到，尽管本章描述的统计方法与语言无关，但每一种语言都具有挑战性。例如，在处理中文文档时，因为中文词一般不用空格分开，处理器先需要将字序列分割成词。同样地，对于形态丰富的语言，需要分析词的结构来获得额外的特征。这些一般都是在预处理阶段完成的，预处理阶段会确定词元序列。词元可以是词或者比词小的单元，由具体的任务和语言决定。预处理完成以后，统计算法将应用到词元序列上。分割问题的目标是确定两个词元间的边界是否应该标记为句子（或主题）的边界。

我们先用一个统一的框架来定义句子和主题分割任务，并描述处理它们的方法，而不是单独地研究句子与主题分割的方法。然后，我们描述用于分割文本和语音的特征。

⊖ 分割法用于两类任务。

2.1.1　句子边界检测

句子边界检测（也称为句子分割）的任务是自动地将词元序列分割成句子单元。在英语等语言的书写文本中，句首会用大写字母进行标记，句末会有一个句点（.），问号（?），叹号（!）或者其他类型的标点符号。但是，除了用于做句子边界标记外，大写字母也用于区分专有名词，句点也用于缩写之中，数字以及其他标点符号也在专有名词中使用。例如，Brown 语料库中 10%的句点用于诸如“Dr.”这样的缩写中，“Dr.”一般是 doctor 或 driver 的缩写，而且句末的缩写中的句点同时也是句子的结束标记。例如，考虑如下句子：“I spoke with Dr. Smith.”与“My house is on Mountain Dr.”在第一个句子中，缩写“Dr.”并不结束一个句子，但在第二个句子中却用于结束句子。在华尔街日报（Wall Street Journal）语料库中，高达 47%的句点用于缩写词中。例如下面一句从 OntoNotes 语料库 [6] 的华尔街日报部分找到的句子中，只有最后一个句点用于结束句子：

“This year has been difficult for both Hertz and Avis,” said Charles Finnie, car-rental industry analyst—yes, there is such a profession—at Alex. Brown & Sons.

这种包含其他句子的句子并不少见。而用引号括起来的句子总是大问题，因为说话者说出了多个句子，而且引号内的句子边界也用同样的标点符号。根据标点符号分割句子结尾的自动方法可能会导致错误的分割。更严重的是，如果前面的句子是说出来的而非书写文本，韵律提示经常用来标记结构。

具有歧义的缩写和大写不是书写文本中句子分割的唯一困难之处。“自发”的书写文本，如短信（Short Message Service，SMS）或即时消息（Instant Messaging，IM），一般没有完整的语法结构而且会错误地使用标点符号，甚至不使用标点符号，这使得句子分割
[30] 更具有挑战性 [7，8]。

同样，如果用来分割成句子的文本从自动系统得来，如光学字符识别（Optical Character Recognition，OCR）或 ASR 这类试图将手写、打印或口语翻译成机器可编辑文本的系统，寻找句子边界的任务还需要处理这些系统中产生的错误。例如，Taghva 等人 [9] 发现 OCR 系统经常将逗号和句号混用，这会产生毫无意义的句子。ASR 转写一般缺少标点符号而且通常是单字符，因此所有 ASR 输出的单词边界都可能是句子的开始或结束。Stevenson 与 Gaizauskas [10] 请人手工分割无标点的文本，他们一般能达到 80%左右的 F1 值，这表示该任务很有难度。在这种输入下，句子分割方法一般假设每两个词元间都有一个句子边界。

另一方面，对于会话，文本或多方会议包含的不合语法或不流畅的句子，大多数情况下很难判断句子边界在哪里。由语言数据联盟（Linguistic Data Consortium，LDC）发行的 ICSI Meeting 语料库 [12] 中，标注者在分割时的一致性非常低。以“okay no problem”作为例子，很难判断这应该看作一个句子还是两个句子。这个问题可以重新定义为会话领域内的对话行为分割（dialog act segmentation）任务，因为有许多标注标准如 Dialog Act Markup in Serveral Layers（DAMSL）[13] 或 Metting Recorder Dialog Act（MRDA）[14] 使得会话中的对话行为有精确的定义。根据这些标准，例句“okay no problem”有两个句子单元（或对话行为单元）：“okay”和“no problem”。

在大多数依赖于自动句子分割的实际应用中，自动分割任务可以根据随后任务的需要进行重新定义。例如，句子“I think so but you should also ask him”是一个合乎语法的

完整句子，但是根据 DAMSL 与 MRDA 标准，它们是两个对话行为标记，一个是肯定，一个是建议。诸如说话者角色检测或情感分析等对话分析中需要这种改变。自动分割任务应被视为语义边界检测任务而非语法边界检测任务。

编码切换，即使用多语言说话人所说的多种语言中的词、短语及句子，是另一个影响句子特性的问题。例如，当切换到另一个语言时，作者可以保留第一种语言的标点符号使用规则，或者遵循第二种语言的格式（例如西班牙语在问句前面需要加上倒问号）。编码转换同样影响技术文本，在技术文本中标点符号的意义可以被重新定义，如统一资源定位符（Uniform Resource Locator，URL）、编程语言和数学。我们必须通过检测和分析这些特殊的构造才能充分地处理技术文本。

传统基于规则的句子分割系统分割结构完好的文本，依赖于模式来识别句子可能的结尾，以及缩略词表来进行消歧[5, 15, 16, 17]。例如，如果边界前的词是一个已知的缩略词，比如“Mr.”或“Gov.”，尽管有些句点有例外情况，但在该位置并不分割文本。虽然规则包含了绝大多数的情况，但它们不能处理未知的缩写词，句子末端的缩写词以及输入文本中的错误。而且，当文本结构并不完好，例如论坛、聊天与博客的文本，或完全没有文字信息的口语输入时，规则不够鲁棒。最后，每一种语言都需要一套特殊的规则。

为了能够获得比基于规则方法更好的结果，句子分割被看作分类问题。给定一个句子边界标记好的训练数据，我们可以训练一个能够识别它们的分类器，将在 2.2 节描述。文本中的句子分割通常使用标点作为分割符，并且试图判断它们是否为句子开始或结束。另一方面，对于语音输入，所有词边界都应考虑为可能的句子边界。

2.1.2 主题边界检测

主题分割（有时称为篇章或文本分割），是一个自动将文本或语言流分割成主题一致的块的任务。即给定词（书写的或语音的）序列，主题分割的目标是寻找主题变化的边界。图 2-1 给了一个广播新闻节目中的主题变化边界的例子。

> Tens of thousands of people are homeless in northern China tonight after a powerful earthquake hit an earthquake registering 6.2 on the Richter scale at least 47 people are dead. Few pictures are available from the region but we do know temperatures there will be very cold tonight –7 degrees. <TOPIC CHANGE>Peace talks expected to resume on Monday in Belfast, Northern Ireland—

图 2-1 新闻文章中的主题边界示例

主题分割对很多语言理解应用而言是很重要的任务，如信息抽取、检索以及文本摘要。例如，在信息抽取中，如果长文档可以被分隔成比较短的、主题一致的片段，那么接下来只需要抽取与用户查询有关的片段。

在 20 世纪 90 年代，美国国防部先进研究项目局（Defense Advanced Research Project Agency，DARPA）发起主题检测与跟踪（Topic Detection and Tracking，TDT）计划，以促进查找与跟踪广播新闻报道流中新主题问题的研究进展[18]。TDT 的任务之一便是将新闻流分割为单个报道。TDT 建设了一个通用的测试平台，不过大多数研究人员也使用模拟环境，比如从路透社拼接新闻报道。

对于多方会议，主题分割任务从篇章分析中获得灵感。对于官方的以及具有良好结构的会议，主题根据议项进行分割，然而对比较随意的对话会议，主题边界并不明显。

主题分割并不是一个简单的问题，因为许多自然语言相关的问题需要一个好的主题类

别及粒度的定义，在该问题上人类的一致性并不是很高。例如，主题一般根据语义呈现为层次结构而非扁平的结构。当一个关于足球的句子紧跟着一个关于棒球的句子，有的标注者会认为是主题改变，而有的标注者不那么认为，他们认为足球和棒球都属于体育
[32] 主题。这也是一个粒度区分的例子。即使告诉标注者要将文本分割成预定义数目的主题，定义什么是主题也是一个很难的问题，因为它随着语义内容而变化。尽管在 TDT 语料库中取得了标注者间的高度一致性（Cohen 的 kappa 值为 0.7～0.9），该语料库包含了广播新闻，文档和故事，但新闻和主题一般具有相同的边界。对于多方会议的主题分割，其一致性更低［20］（kappa 值一般为 0.6～0.7）。注意在会话语音中，主题边界并不绝对，例如在一个多方会议中，在转换到另一个主题以后，一个参与者会说一个关于先前主题的句子。

在文本中，主题边界通常使用特殊的分割提示，如标题和段落分隔符。这些提示在语音中并不存在。但是，语音提供了其他的提示，比如等顿间隔和说话人切换。这类似于文本和语音的句子分割的差异，2.5 节将会对特征类型进行更细致的分析。

2.2　方法

句子和主题分割一般被考虑为边界分类问题。给定一个边界候选（对句子分割而言在两个词元之间，对主题分隔而言在两个句子之间），我们的目标是判断候选是不是一个真正的边界（句子或主题边界），形式地讲，令 $x\in X$ 为候选对应的特征向量，$y\in Y$ 为候选预测的标记，标记 y 可以取 b 和 $\overline{b}$，分别表示边界和非边界。这样便导致一个分类问题：给定一个训练例子的集合 $(x_i, y_i)_{i=1,N}$，寻找一个函数，能够对未见例子 x_{unseen} 赋值一个最精确的标签 y。除了视为二元分类问题，也可以用更细的粒度来建模边界类型。例如，Gillick[21] 建议文本的句子分割应该是一个 3 类问题：句子边界伴随缩写词记为 b^a，不伴随记为 b^n，缩写词并非句子边界记为 b^a。类似地，对于口语，也可以视为 3 类问题：非边界 $\overline{b}$，陈述边界 b^s 以及疑问边界 b^q。

特征可以是：候选边界周围存在特定的词 n 元组，在文本引号中的指示，先前词元在缩写词表中的指示等；持续问题，语调，能量或者其他语音中持续时间相关的特征。2.5 节中有更多对特征的讨论。

对于句子或主题的分割，问题定义为寻找最可能的句子或主题边界。句子分割的自然单位是词，而对主题分割而言是句子，因为我们通常假设一个句子中的主题不变[⊖]。然后
[33] 词或句子可组织成属于句子或主题的一个连续块，即将词或句子的边界分类为句子或主题的边界与非边界。每个潜在的边界 i 都可以进行分类（局部模型），目标是对每个例子 x_i 估计最可能的边界类型 $\hat{y}_i$：

$$\hat{y}_i = \underset{y_i\in Y}{\operatorname{argmax}} P(y_i \mid x_i) \tag{2.1}$$

这里，^ 表示估计的类别，没有 ^ 的变量表示可能的类别。在这个公式中，对每个例子单独指定类别，因此决策是局部的。然而，连续的边界类型存在相互关联。例如，在新闻广播语言中，一个词形成句子，进而产生两个连续的边界，这种情况是很少见的。在局部模型中，可以从候选边界附近的实例上下文中抽取特征来建模这种依赖性。也可以将候选边界看作一个序列，给定候选例子 $X=x_1, \cdots, x_n$，搜索可以最大化概率的边界类型序列

⊖ 同样，对于主题分割，有时假设主题仅在改变边界上改变［22］。

$\hat{Y}=\hat{y}_1,\cdots,\hat{y}_n$

$$\hat{Y}=\underset{Y}{\operatorname{argmax}}P(Y\mid X) \tag{2.2}$$

在下面的讨论中，我们将方法归类到局部以及序列分类这两类。另一种分类方法是按照机器学习算法进行分类：生成性和判别性。生成序列模型估计观察值 $P(X, Y)$（如词、标点）和标记（句子边界、主题边界）的联合概率，这一般需要特定的假设（如，使用回退来考虑未知事件）以及良好的泛化能力。另一方面，判别性序列模型主要关注能区分实例标记的特征。

这些方法（下一节中描述）可以同样地用于文本或者与口语的句子和主题分割中，但是有一个区别：在文本中，所有不包含潜在结束句子标记符（句点、问号、叹号）的边界类别被预先设定为非句子或非主题边界类型，但是在语音中，一般要考虑所有相邻词元间的边界。

2.2.1 生成序列分类方法

主题与句子分割中最常使用的生成序列分类方法是隐马尔可夫模型（Hidden Markov Model，HMM）。式（2.2）的概率可以通过贝叶斯公式重写如下：

$$\hat{Y}=\underset{Y}{\operatorname{argmax}}P(Y\mid X)=\underset{Y}{\operatorname{argmax}}\frac{P(X\mid Y)P(Y)}{P(X)}=\underset{Y}{\operatorname{argmax}}P(X\mid Y)P(Y) \tag{2.3}$$

分母中的 $P(X)$ 可以去掉，因为对于不同的 Y，它的值不变，因此不会改变结果。$P(X\mid Y)$ 和 $P(Y)$ 可以估计为：

$$P(X\mid Y)=\prod_{i=1}^{n}P(x_i\mid y_1,\cdots,y_i) \tag{2.4}$$

24

和

$$P(Y)=\prod_{i=1}^{n}P(y_i\mid y_1,\cdots,y_{i-1}) \tag{2.5}$$

为使计算可解，需要简化假设：

$$P(x_i\mid y_1,\cdots,y_i)\approx P(x_i\mid y_i) \tag{2.6}$$

可以假设二元模型来建模输出类别：

$$P(y_i\mid y_1,\cdots,y_{i-1})\approx P(y_i\mid y_{i-1}) \tag{2.7}$$

二元情形使用一个完全连接的 m 状态马尔可夫模型，在这里 m 是边界类别的个数。对于句子（主题）分割，这些状态生成词（句子或段落），并且估计能够最可能生成词（句子）序列的状态序列。状态转移概率 $P(y_i\mid y_{i-1})$ 与状态观察值似然 $P(x_i\mid y_i)$ 使用训练数据进行估计。使用动态规划来计算最可能的边界序列。解码马尔可夫模型使用 Viterbi 算法 [23]。以增加复杂度为代价，二元模型可以推广为更高阶的 n 元模型。

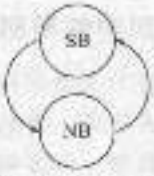


图 2-2 一个假想的具有两个状态的分割问题的隐马尔可夫模型：一个具有段边界，一个为其他类型

图 2-2 是一个两类问题模型的例子，非边界（NB）和边界（SB）作为句子分割的标记。表 2-1 显示了一个生成词组成的序列例子。

表 2-1 使用简单的两个状态的马尔可夫模型得到的句子分割

生成词	…	people	are	dead	low	pictures	…
状态序列	…	NB	NB	SB	NB	NB	…

对于主题分割，一般使用 n 个状态而不是两个状态，这里 n 是主题的数目。但是，在不知道主题类别的情况下获得状态观察值似然是一个很大的挑战。Yamron 等人 [24] 使用一元语言模型来建模主题，而状态观察值似然使用 k-means 聚类算法来训练。

注意，句子或主题分割中使用 HMM 与使用 HMM 的其他任务，如词性（Part-Of-Speech，POS）标注 [25]] 或命名实体抽取 [26]，并没有太大的区别。但是传统 HMM
35 方法被证实具有一定缺陷。例如，该模型不能使用比词更多的信息，如词的 POS 标记或者语音分割中的韵律提示等。

为达到这个目的，有两种简单的扩展。Shriberg 等人 [27] 建议使用显式状态来生成边界词元，因此可以通过结合其他模型的方法来融入非词法信息。这种方法用于句子分割之中，它受到隐事件语言模型（Hidden Event Language Model，HELM）的启发。HELM 是 Stolcke 与 Shriberg [28] 提出的，原意是针对语音不流畅问题设计，该方法将这种事件作为额外的元词元。在 Shriberg 等人设计的模型中，对于每个边界词元 SB 和 NB，需要保留一个状态，其他的状态用于生成词。为了简化计算，如果前面的词不是不流畅单元的一部分，那么所有连续词间都要插入一个虚拟的词元。例 2-1 是一个具有边界词元的序列设想中的概念性表示：

例 2-1　…*people NB are NB dead YB few NB pictures*…

最可能的边界词元序列也是通过 Viterbi 解码得到的，图 2-3 描述了设想的用于分割的理论隐事件语言模型。

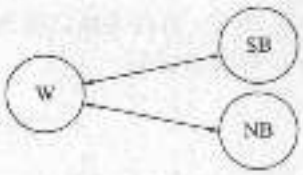


图 2-3　分割问题的理论隐事件语言模型

这些额外的边界记号用来获取其他元信息。最常用的元信息是其他分类器的反馈。一般地，在除以先验概率以后 [27]，在边界状态中的后验概率用作状态观察值似然。这些额外的分类器也可以使用其他特征集来训练，比如韵律或句法。这种混合方法在 2.2.4 节描述。

对于主题分割，Tur 等人 [29] 采用了同样的想法，显式地模型主题开始及主题结束节，对广播新闻主题分割有极大的帮助。

第二个扩展是受到分解式语言模型 [30] 的启发。分解式语言模型不仅包含了词的信息，也包含了形态、句法以及其他的信息。Gus 等人 [31] 提出对句子分割使用分解式 HELM（fHELM），除了词以外还使用 POS 标记信息。

2.2.2　判别性局部分类方法

判别性分类器的目标是直接对式 (2.1) 中的 $P(y_i|x_i)$ 进行建模。在如朴素贝叶斯这种生成模型方法中，类别密度 $P(x|y)$ 是模型的假设，但在判别方法中，用特征空间的判别函
36 数来定义模型。许多判别性分类方法，如支持向量机、boosting、最大熵与回归等，均是基于不同的机器学习算法。尽管判别性方法在许多对话及语言处理任务中被证明可以超过生成方法，但它的训练一般需要进行迭代优化。

在判别性局部分类方法中，每个边界通过使用局部特征与上下文特征进行单独处理。与序列分类模型不同，判别性局部分类方法没有进行全局（即句子或文档级）优化，但一些与更大的上下文有关的特性可以纳入特征集合中。例如，可以通过迭代的方式来使用前一个或后一个边界的预测类别。

对于句子分割，应用于报纸文章的主要是有监督学习方法。Stamatatos、Fakotakis 与

Kokkinakis [32] 使用基于转换的学习（Transformation-Based Learning，TBL）方法来得到寻找句子边界的规则。许多分类器都尝试过处理这个问题，比如，回归树 [33]、神经网络 [34，35]、C4.5 分类树 [36]、最大熵分类器 [37，38]、支持向量机（Support Vector Machine，SVM）还有朴素贝叶斯分类器 [21]。通过给标点符号赋于一个标记 [39]，Mikheev 将句子分割问题看作词性标注的子问题。他使用了 HMM 与最大熵相结合的方法来处理标注问题。

主题分割 [40，22] 中常用的 TextTiling 方法使用了词向量空间的词法连贯性度量作为主题相似的指示。TextTiling 也可以看做使用单个相似性特征的局部分类方法。图 2-4 是一个典型的相邻分割单元相似度的图。当相似度低于某个阈值时，文档会被切分。

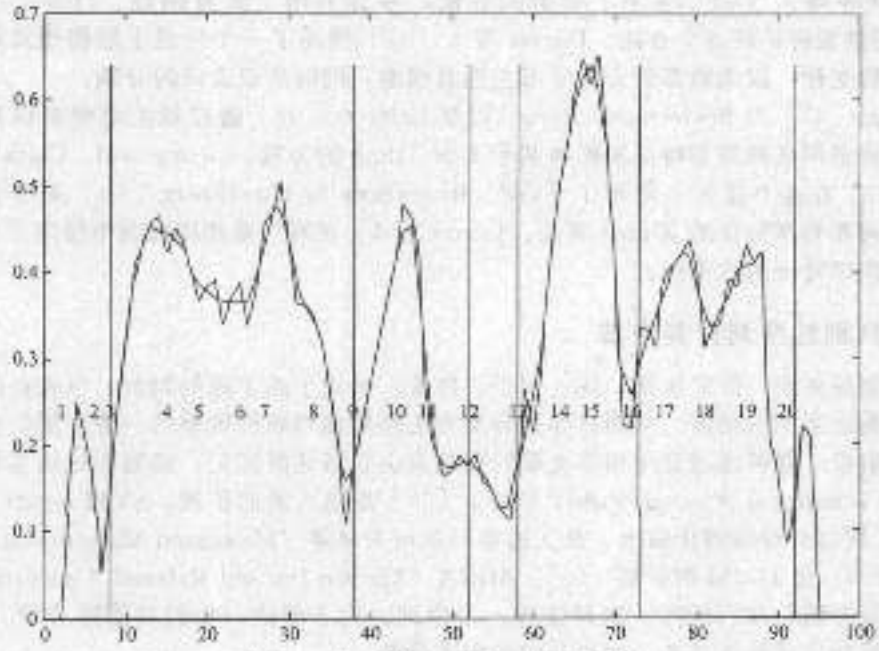


图 2-4 TextTiling 例子（源于 [22]）

最初提出了两种计算相似度的方法：块比较以及词汇引入。第一种方法：块比较方法，根据相邻块中相同词的个数来计算相似度。块的大小可以是变动的，可以使用一个窗口而不是只看相邻的块。给定两个块 b_1 和 b_2，每块有 k 个词元（句子或段落），相似度（或主题连贯性）分值由以下公式计算：

$$\frac{\sum_t w_{t,b_1} * w_{t,b_2}}{\sqrt{\sum_t w_{t,b_1}^2 \sum_t w_{t,b_2}^2}}$$

其中 $w_{t,b}$ 是赋给块 b 内项 t 的权重。权重可以是二值的，也可以使用如词语频率这种基于信息检索的信息度量。

第二种方法：词汇引入方法，根据以当前词为中点的区间中有多少新词，来计算词元序列的分值。与块比较公式类似，给定两个具有同样词数 w 的块 b_1、b_2，主题连贯性分值按如下公式计算：

$$\frac{NumNewTerms(b_1) - NumNewTerms(b_2)}{2 \times w}$$

其中 $NumNewTerms(b)$ 返回文本的块 b 中第一次见到的词语的个数。

Brants、Chen 以及 Tsochantaridis [41] 将该方法扩展以利用潜在语义分析。与仅看所有词不同，这种方法处理转换过的词法空间。因为这种方法能够隐式地捕捉到语义相似度，所以它能取得更好的效果。

Morris 与 Hirst [42] 提出用词汇链而非词法相似度来计算连贯性。稍后，Kan、Klavans 以及 McKeown [43] 提出了使用更简单的词汇链解释方法。只当非功能词和句法短语出现在 n 个句子中时，才把它们连接到一起。在这里，n 和连接的权重根据句法类别进行调整。

Banerjee 和 Rudnicky [44] 将最原始的 TextTiling 方法应用到会议领域。对于会议分割，Galley 等人 [45] 使用了相似的方法，并且采用了重复词链。Hsueh 与 Moore [20] 使用决策树扩展这个方法。Purver 等人 [46] 使用了一个生成主题模型及潜在狄利克雷分配的变种，以无监督的方式学习主题的模型，同时生成会议的分割。

Reynar [47] 与 Beeferman、Berger 以及 Lafferty [48] 通过最大熵模型以及跟踪词汇转移的许多词法和篇章特征来扩展基于 TextTiling 的方法。Georgescul、Clark 和 Armstrong [49] 在这个任务上使用了 SVM。Rosenberg 与 Hirschberg [50] 采用了用词汇链、提示词和韵律特征的 Ripper 算法。Levow [51] 在对广播新闻分割中使用了基于余弦相似度和韵律特征的决策树。

2.2.3　判别性序列分类方法

在分割任务中，给定实例（词、句子、段落）的句子或主题的判断，很大程度上依赖于该实例附近实例的判断。判别性序列分类方法是局部判别性模型的一般扩展，它拥有额外的解码阶段，能够通过使用相邻决策的信息来决定最佳的标记，进而标记该实例。条件随机场（Conditional Random Field，CRF）[52] 是最大熵的扩展，SVM struct [53] 是 SVM 的扩展以处理结构化输出。最大边界马尔可夫网络（Maximum Margin Markov Network，M^3N）是 HMM 的扩展 [54]。MIRA（Margin Infused Relaxed Algorithm）是一个在线学习方法，在训练时一次只读取一个序列。为了简洁，我们只描述 CRF。CRF 为许多序列成功完成标注任务，如语音中的句子分割。

[38]

CRF 是用于标注结构的一类对数线性模型 [52]。与独立预测句子或主题边界的局部分类器不同，CRF 可以使用整个序列的边界假设来做出判断。形式上讲，在给定从输入 $(X=x_1, \cdots, x_n)$ 上下文抽取的特征集合以后，CRF 赋予边界标注序列 $(Y=y_1, \cdots, y_n)$ 的条件概率：

$$P(Y \mid X) = \frac{1}{Z(X)} \exp\left(\sum_{i=1}^{n} \sum_{j=1}^{m} \lambda_j f_j(y_{i-1}, y_i, x_i) \right) \tag{2.8}$$

$$Z(X) = \sum_{Y} \exp\left(\sum_{i=1}^{n} \sum_{j=1}^{m} \lambda_j f_j(y_{i-1}, y_i, x_i) \right)$$

其中 $f_j(\cdot)$ 是观察值以及标记团的特征函数，λ_j 是相应的权重。$Z(\cdot)$ 是归一化函数，只与观察值有关。CRF 在训练时，寻找能够最大化训练数据似然的 λ 参数，同时经常添加调节项来避免过拟合。常使用的训练算法有梯度、共轭梯度以及在线方法 [56, 57, 58]，使用动态规划（Viterbi 解码）来计算 $Z(\cdot)$ 函数以及在测试时寻找最可能的标记分配。

2.2.4　混合方法

非序列判别分类算法一般会忽略上下文，而这对分割问题而言是很重要的。虽然我们

可以将上下文作为特征或者使用本身就考虑上下文的 CRF，但这些方法在处理如停顿时间和音高区间这样的实数值特征时，得到的结果是次优的。之前的研究在处理这个问题时，都是简单地用手工或自动方法将特征空间离散化[59]。

另一种方法是使用混合分类器方法，正如 Shriberg 等人[27]所建议的那样。方法的主要思想是对于每个候选边界，使用从诸如 boosting 或 CRF 这种分类器得到后验概率 $P_c(y_i|x_i)$，按照贝叶斯公式除以先验得到状态观察值似然：

$$\underset{y_i}{\operatorname{argmax}} \frac{P_c(y_i \mid x_i)}{P(y_i)} = \underset{y_i}{\operatorname{argmax}} P(x_i \mid y_i) \tag{2.9}$$

将 Viterbi 算法应用到 HMM 即可得到最可能的分割。为了处理状态转移概率和观察值似然的动态区间，可以应用文献中经常提到的加权方法：

$$\underset{y_i}{\operatorname{argmax}} P_c(x_i \mid y_i)^{\alpha} \times P(y_i)^{\beta} \tag{2.10}$$

其中 $P(y_i)$ 使用 HELM 估计，α 和 β 使用开发集优化。

Zimmerman 等人在多语话句分割实验中对比了多种局部判别分类方法，即 boosting、最大熵、决策树以及它们的混合版本，结论是混合方法总是要好。Guz 等人[31]对 CRF 得到了同样的结论，尽管 CRF 与混合方法的差距较小。

2.2.5 句子分割的全局建模扩展

到目前为止，大多数句子分割方法主要关注识别边界，对句子本身并不关心。这是因为如果关心句子，需要评估比目前多二次方数量的句子假设，这比边界的数目要多。为了解决这个问题，Roark 等人[61]使用局部模型判断的最可能的句子边界来分割输入，然后用分割的 n-best 列表来训练一个重排器。这种方法能够利用一些句子级特征，如句法分析器输出分值或全局韵律特征。Favre 等人[62]使用剪枝的句子格来扩展该方法，使得能够更有效地融合局部分值与句子级分值。

2.3 方法的复杂度

我们描述的方法有不同的优缺点。在给定的上下文和特征集合中，有的方法可能比另外的要好。这些方法可以根据训练和预测算法的复杂度（时间和空间），以及在真实数据集上的表现进行评价。有些方法需要特殊的预处理，如将连续特征转换或者标准化为离散特征。

就复杂度而言，判别性方法的训练比生成方法的训练要复杂，因为它们一般需要通过处理多遍训练数据来调整它们的特征权重。然而，诸如 HELM 这种生成模型，可以通过使用大规模的训练数据来获得提升，例如使用数十年的新闻文稿。另一方面，这些模型只能使用比较少的特征（对 HELM 而言只有词）并且不能有效地处理未知事件。判别性分类器允许使用更多的特征，在训练数据较小的情况下有更好的结果。即使使用相对简单的模型（线性或对数线性），判别性分类器的预测仍然比较慢，因为提取特征占据了大量的时间。

与局部方法相比，序列方法使解码更加复杂：寻找最优序列的决策需要评价所有可能的序列决策。幸运的是，条件独立假设使得动态规划可行，进而平衡空间与时间，使得解码能在多项式时间内完成。复杂度一般随着模型的阶（同时处理的候选边界的个数）以及类别的数目（边界状态的数目）成指数级增长。判别性序列分类器，如 CRF，还需要在训练数据上重复进行推理，这使得它们的代价更高。

2.4 方法的性能

对于语音中的句子分割，性能通常使用错误率（错误的数量与总数量之比）评估，如F1值（有回率与精确率的调和平均数，召回率是正确返回的句子边界数目与参考标注中句子边界数目之比，精确率是正确返回的句子边界数与所有自动预测的句子边界数目之比）以及NIST（National Institute of Standards and Technology）错误率（错误标记的候选数目与实际边界数目之比）。

对于文本中的句子分割，研究人员汇报了华尔街时报语料库中约27 000个句子的错误率，例如，Mikheev [39] 报告称他的基于规则的系统能达到1.41%的错误率。使用额外的缩略词表能使该系统的错误率降到0.45%，再结合使用词性标记特征的有监督分类器能得到错误率为0.31%的结果。不使用手写规则以及缩略词表，Gillick [21] 的基于SVM的系统得到更低的错误率，为0.25%。尽管这些错误率看起来很低，但句子分割是任何NLP任务的第一步，并且每一步的错误都会影响到随后的步骤，尤其是最终句子呈现给用户时尤为严重，如自动摘要任务。

对于语音中的句子分割，Doss等人 [63] 使用MaxEnt分类器在TDT4 Multilingual Broadcast News Speech的普通话语料库中取得了69.1%的F1值，使用同样的特征，Adaboost可以达到72.6%，SVM能达到72.7%。他们还提出了使用逻辑回归来融合三种分类器的方法，在Turkish broadcast news语料库上，Guz等人 [31] 使用HELM得到了78.3%的F1值，利用形态学特征的fHELM得到了86.2%的F1值，Adaboost得到了86.9%的F1值，CRF得到89.1%的F1值。这些结果中，HELM（还有fHELM）以及其他分类器都由同样的语料库训练。但是它们还可以用更大的语料库训练，并通过结合判别性分类器来提升性能。例如，Zimmerman等人 [64] 报告TDT4 broadcast news的英文语料库中，用Adaboost结合HELM可以得到67.3%的F1值，而单独用Adaboost只能得到65.5%。

2.5 特征

尽管许多方法与它们所使用的特征紧密相连，但是出于演示的目的，将它们分离是很有利的。同样地，尽管大多数的特征类别，如词法或韵律特征，在句子和主题分割中很常见，但它们的使用却差别很大。当特征可以同时用于句子和主题分割时，我们用分割来统称，其他情况会显式说明。

41

本节我们将落在边界观察值的特征用向量x来表示。特征可以是二元的（存在触发词用$x_f=1$表示，不存在用$x_f=0$表示），也可以取实数值（如句子的长度、暂停的持续时间），即$x_f \in \mathbf{R}$。对于二元特征，我们用x_f表示$x_f=1$，忽略$x_f=0$。

有些分类器会假定输入特征的性质，要求特征必须是二元或者它们的分布最好标准化。通过量化以及投影到高维空间，实值特征可以转换为二元特征。如果一个特征的值在某个区间里，那么投影空间中对应维的值为1，其他维为0。

2.5.1 同时用于文本与语音的特征

1. 词法特征

对于文本和语音的句子及主题分割而言，词法特征都是非常重要的特征。句子和主题的首尾词元以及短语可以被先前描述的统计机器学习算法利用。一般地，对于句子（或主题）分割，会分析一个大小为n的词元（或句子）窗口。序列分类方法频繁地使用这类分

析信息，而局部分类方法需要使用对应的特征，如与上一个句子重合的实词等。

对于文本的句子分割，词法提示是文本中的词元，主要任务是对句子结尾标点符号进行消歧。对于语音的句子分割而言，词法提示是原始词元，因为语音中缺乏符号提示。

注意词法特征有两种用法。第一种用法是基于边界附近的词汇特征出现与否，如提示短语。例如，在 TDT 的广播新闻语料中，新闻单元（即主题）通常以相似的短语结束。这类用法被描述为“篇章特征”。第二种用法类似于基于 TextTiling 的方法，它一般使用计算余弦距离时的实词词干。第一种用法与类别和语言有关，而第二种用法与语言无关。这两种用法并非非此即彼，而是可以融于同一分类框架中。Reynar 的工作[47]可以看作是达成这一框架的先锋。在最大熵的框架中，Reynar 使用了边界之前与之后窗口中实词和重复名字的计数为特征。

更形式化地，令 w_1，w_2，…，w_n 为输入词元，我们从 w_i 和 w_{i+1} 间的边界候选中提取词特征。对于句子分割，最相关的特征一般是边界之前、之后以及跨边界的词元 n 元组。对于二元组，会提取出以下特征：x_{w_{i-1},w_i}，$x_{w_i,w_{i+1}}$ 和 $x_{w_{i-1},w_{i+1}}$。跨边界特征能够抓住句子边界的一些信息，如 *Geo. Smith* 后面不可能是句子边界，而 *government. The* 中间可能有句子边界。

对于主题分割，候选边界出现在句子间。如果将边界前的句子记为 s_i，将边界后的句子记为 s_{i+1}，而如果这些句子中存在提示短语 c 则记为 $x_{c\in s_i}$ 和 $x_{c\in s_{i+1}}$。第二类特征是边界前后内容的相似内容，一般通过前后句子的相似度表示： 43

$$x_{\text{cosine}(s_i,s_{i+1})}=\frac{\sum_w \text{tf}(w,s_i)\text{tf}(w,s_{i+1})\text{idf}(w)}{\sqrt{\sum_w(\text{tf}(w,s_i)\text{idf}(w))^2}\sqrt{\sum_w(\text{tf}(w,s_{i+1})\text{idf}(w))^2}}$$

其中 $\text{tf}(w,s)=\frac{n_{w,s}}{\sum_v n_{v,s}}$ 表示句子 s 中记号 w 的项频率，$\text{idf}(w)=\log\frac{D}{df(w)}$ 表示该记号的逆文档频率，它可以表明词元是否常见，一般在另外的语料库中计算（D 是文档的总数目，$df(w)$ 是包含 w 的文档的个数）。内容可以从多个级别上比较：比如，边界前 n 句和边界后 n 句。

词汇链是另一个与主题分割有关的特征。我们一般计算开始于和结束于候选边界的链的个数。令 $c\in C$ 为有关词汇链的词的集合（例如叶子、玫瑰、花）。基于实用性考虑，一个词汇链经常缩减为一个词元（所有出现的叶子）。然后，对于 w_i 与 w_{i+1} 之间的候选边界，broken-lexical-chain 特征可以由以下公式计算：

$$x_{\text{chain}}=\left|\left\{c\in C:\min_{w_j,w_k\in c,j\le i<k}(k-j)\ge d_{\min}\right\}\right|$$

大多数基于文本源的自动主题分割工作都以某种方法探索主题词使用提示。Kozima[65]使用文本序列中词的相互相似度作为文本结构的指示。Reynar[66]通过图模型来建模重复词的分布，可以找到主题相似的区域。Ponte 与 Croft[67]使用局部上下文分析中的信息检索技术来提取主题片段的相关词集合，然后与扩展词集合进行对比。

Beeferman 等人[48]使用了最大熵模型，融合了一个由自动选择的词汇篇章提示组成的大特征集合。他们通过建立两个统计语言模型，也将主题词使用融合进模型中。这两个语言模型为：一个静态的（主题无关的）语言模型和根据过去词更改词预测的语言模型。它们表明了两个预测器的对数似然率之比可以作为主题边界的指示，因此可以用于指数模型分类器的一个额外特征。

2. 句法特征

一系列的研究成功地应用了句法信息。Mikheev[39] 在句子分割任务中隐式地使用了 POS 标记。同样地，对于 2.2.5 节介绍的全局重排序方法，使用了以成分树或依存树形式出现的句法特征。

对于形态学丰富的语言，如捷克语和土耳其语，用词的形态分析作为附加提示[31, 68]。

形式上说，令 t_1，…，t_n 为 POS 序列或者从词 w_1，…，w_n 中抽取的形态标记。可
13 以提取和词类似的特征（边界之前、之后以及跨边界的 n 元组）。例如，$x_{t_{i-1}t_i}$，$x_{t_it_{i+1}}$ 和 $x_{t_{i-1}t_{i+1}}$。对于主题分割而言，句法特征不那么有用，因为主题的变化一般是因为内容的转换。

在概率上下文无关文法（Probabilistic Context-Tree Grammar，PCFG）的全局模型下评价一个候选句子合乎语法的程度，我们可以计算该句子所有可能分析树的概率的和：

$$x_{\text{pcfg}} = \sum_{t \in T_s} P(t) = \sum_{t \in T_s} \prod_{r \in t} P(r)$$

其中 t 是分析树，r 是树中使用的产生式规则[69]。

3. 篇章特征

无论语音还是文本，篇章特征对于分割而言都是非常重要的。例如，在广播新闻中，播音员首先说出标题，然后是赞助广告，最后报道才一个个呈现，中间可能会有播音员或记者的交互或者主题开始、结束短语。

之前文本和语音分割的工作已经显示出提示短语或篇章助词（比如 now 或 by the way）以及其他词法提示，是篇章结构单元非常有价值的指示[如 70，71]。类似地对于语音来说，说话人的改变可以作为句子边界的指示，广告可以是广播新闻或会话主题边界的指示。形式上讲，对于所有出现在边界附近的事件 $e \in E$，特征 x_e 可以表示该事件出现，$x_{\bar{e}}$ 表示该事件没有出现。事件需要由本书没有详细描述的系统（如一个商业检测器）进行检测，同时可能会输出置信分值。在这种情况下，特征为 $x_e = cs$，其中 cs 是被识别事件的置信分值。

尽管之前的方法试图使用预先定义的篇章提示，更多基于语料库的方法，使用具有有效特征集的机器学习方法来自动学习这种模式。例如，Tur 等人[29] 对主题的开始与结束句子使用显式的 HMM 状态，取得了很大的提高，Rosenberg 与 Hirschberg[50] 使用统计假设检验来确定这样的短语。

对于会议或会话分割，篇章特征更加复杂而且依赖于辩论结构。大多数工作简单地使用前一个或后一个的说话人切换来作为篇章特征。更高级的语义信息，如对话行为标记或会议议程项目，也是可以利用的篇章信息[72]。

2.5.2　只用于文本的特征

排版与结构特征

对于句子和主题分割而言，如标点符号和标题这样的排版与结构提示，是十分有用的。句子分割系统使用边界之前与之后的词，还有词的大小写及 POS 标记、长度以及它们在非句子边界上下文（如在小写词前）中出现的频率。同样地，包含缩写词的地名词典、预处理及后处理模式也用来处理文本。

形式上讲，令 g 为地名词典中出现的词的集合。如果 $x \in g$，那么生成一个特征 $x_{g(x)} =$

1，类似地，记录小写格式词的频率的特征可以用 $x_{lc(w)}=\frac{|lc(w)|}{|w|}$ 计算，其中 $lc(w)$ 表示 w 的小写格式。

在 Gillick [21] 的句子分割工作中，他发现给定一个特征集合，相对于训练与测试数据的不一致以及输入词切分的不一致而言，分类器的选择影响不大。Kiss 与 Strunk [73] 提出了一种寻找句子边界的无监督方法，它能够用全局统计量在未标记的语料中学习缩写词。尽管这个方法与语言无关，但是如果缩写词在测试语料中没有多次出现，那么它也是不能被识别的。

其他的结构提示包括段落边界、标题、节号等。这种提示只存在于结构文本资源中，在博客、聊天室这种文本中并不存在。

2.5.3 语音特征

当使用语音识别的输出时，因为识别错误，有些词可能不正确，这使得词法特征的质量下降。同样地，词元开始与结束时间有可能估计得不对，使得在计算韵律特征时产生错误。典型地，为了鲁棒地应对这些错误，会抽取出大量的韵律特征。

韵律特征

当分割语音而非文本时，可以使用同样的方法，但是要三思。首先，对于自动语音处理，词法信息是从语音识别的输出而来，这其中一般会包含错误。其次，口语一般缺乏显式的标点符号、大写以及格式信息。而这些信息通过语音以及韵律传达。再次，尽管口语如广播新闻是从文本中读出的，但大多数自然语音是会话形式的。在自然、自发的语音中，句子可以是“不合语法”（从形式句法的角度）的，并且一般包含相当数目的语音间断，如填充的停顿、重复和纠正。

另一方面，口语语音输入提供额外“超过词汇”的信息，这些信息可以从声调和节奏，即韵律中得到。韵律是音高（基频）、音量（能量）以及时间（由发音时长和停顿传达）的模式。韵律提示在自然口语中与篇章结构相关，因此可以在指示句子边界及主题转换中起作用。而且韵律提示天生与词是独立的，因此它们的错误比自动语音识别中的词法特征要少。

图 2-5 描述了句子语音分割中的韵律特征和词汇特征。广义地说，句子边界的韵律特征与主题边界的相似，因为它们都有用于表示分块信息的停顿。在大的（即主题）间断中，停顿长度、音高以及能量重置通常要大一些，但是相似类型的韵律特征可以用于这两个任务中，特征是从当前任务中训练而来。

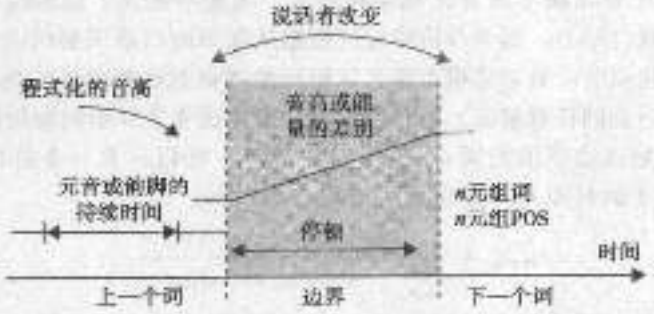


图 2-5 语音分割中的基本韵律与词法特征

大量的句子分割研究 [74, 75, 27, 76, 77, 78, 51, 11, 79, 69, 80] 使用了韵律

特征。最简单且最常用的特征是当前边界的停顿。对于自动处理，停顿比其他韵律特征更容易获得，这是因为与音高和能量特征不同，停顿信息可以从自动语音识别输出中提取。当然，不是所有的句子边界都包含停顿，尤其是在自然语音中。反之，不是所有的停顿都对应句子边界。例如，有些句子内部的不连贯也包含停顿。一些方法只是简单地查看是否存在停顿，而另外一些方法建模停顿的时间。停顿持续时间在会话的说话人切换边界时非常长，这是因为这段时间另一个人要讲话。某些对话行为，如一些反向信道（如 "uh huh"）倾向于单独出现，因此句子分割只用停顿信息也会取得相当成功的结果。

停顿特征计算为 $x_{pause}=start(w_{i+1})-end(w_i)$，其中 $start()$ 和 $end()$ 表示语音识别结果中开始与结束词的时间，以秒为单位。相关的边特征是词前的停顿（以了解它是否是单独的）以及量化停顿，$x_{qpause}(w_i)=1$ 当且仅当 $x_{pause}>thr_{pause}$，其中 thr_{pause} 可以设为 0.2 秒。停顿持续并不服从正态分布，这会使一些假设正态分布的分类器遇到问题。然而，这个特征往往是语音分割中最相关的特征。

更细致的韵律建模包括音高、音长、能量信息。音高使用语音中有声部分的基频来建模。音高传达了一系列的信息，如突出的音节。不过在句子分割中，使用音高一般是为了找到音高的重置。因此，一般方法是看词边界音高的变化，越大的负值越表示可能是句子边界。除了建模词边界的音高中断外，有的方法 [37] 也建模了说话人相关的言语结尾的音高值，这不仅提高了性能，也使因果建模成为了可能。这是因为它不依赖于停顿之后的语音 [81]。

音高不是一个连续函数，也不能在声音范围外进行计算。因此，给定一个候选边界，音高特征可能没有定义，这对一些分类器而言是很大的问题。计算音高、加以正确平滑和插值不是本书的内容，它们应该由合适的软件进行处理，如广泛使用的 Praat 工具 [82]。

[46]

一般地，特征是从候选边界前一词之前的窗口以及边界后一词之后的窗口中统计的音高值来计算。例如，前段描述的音高差别特征计算为：

$$x_{pitch}=\left(\max_{t\in W_1(w_i)}\text{pitch}(t)\right)-\left(\min_{t\in W_2(w_{i+1})}\text{pitch}(t)\right)$$

其中 $pitch(t)$ 是时间 t 的音高值，$W_1(w_i)$ 是词 w_i 之后的临时窗口，$W_2(w_{i+1})$ 是词 w_{i+1} 之前的相似窗口。该特征的变化可以通过改变窗口大小（如 200ms、500ms），改变边界两边的统计量（如 min、max、mean），按照不同的因素（即 log 空间投影，当前说话人音高值分布标准化）规范化音高值这几种方法来创建。

句子分割的持续时间特征旨在捕捉一种称为边界前延长（preboundary lengthening）的现象，即最后一个单元前的语音区域会被拉长（有意思的是，这种现象也在音乐中，甚至鸟的歌声中被发现 [83]）。当音节持续时间根据从相似的口语风格的语料库中该音节持续时间的平均值标准化以后，自动建模方法可以很好地能捕捉到边界前延长现象。倒数第二个音节的韵（元音和它后面的任意辅音）的持续时间一般比该音节开始的韵的持续时间要长。

例如，令 v 为候选边界前的词 w_i 中最后的元音。可以计算一个特征作为该元音的相对持续时间，相对于语料库 C 中该元音的平均持续时间：

$$x_{vowel}=\frac{\text{end}(v_{w_i})-\text{start}(v_{w_i})}{\sum_{w\in C}\text{end}(v_w)-\text{start}(v_w)}$$

能量特征在句子边界建模中也被使用，不过没有那么成功。从描述的角度来看，能量与音高类似，在句子结尾时减弱，下一个句子时又会重置。但是能量受很多因素影响，包括录制方法，而且很难标准化，无论是对同一个人而言还是对许多人而言。因此该特征没

有自动分割中的停顿、音高以及持续时间这几个特征有效。

韵律建模中最后一个特征是音质。一些描述性工作显示音质的改变与句子边界有一定关联，但是这种现象与说话人高度相关，而且很难用自动方法捕捉到。所以大多数自动分割方法还是依赖于先前提到的那些韵律特征。

主题边界的一些描述性工作发现主题的明显转变往往会伴随长的停顿、额外高的 F0 开始与重置、更高的最大音峰值、说话速率的转变以及更广的 F0 和强度（例如，84，85，86，87，22）。这些提示对人类听众而言很明显。事实上，即便通过谱过滤使得语音本身变得难以捉摸 [88]，测试者仍然可以感知到大的篇章边界。在自动主题转换的研究中，Galley 等人 [45] 发现说话人行为的改变、沉默的程度、重叠的语音以及特定提示短语的存在与否都能指示主题的变化。将这些特征加入到他们的方法中极大地提高了分割的准确率。Georgescul、Clark 和 Armstrong [89] 发现他们的方法加上类似的特征也得到了一些提升。但是，Hsueh、Moore 以及 Renals [90] 发现这只对粗粒度的主题转换（对应活动的变化或者会议陈述的变化，如简介、闭幕或回顾）有效，在细粒度的主题转换中没有发现效果。

2.6 处理阶段

一般地，分割任务的第一步是预处理，以用来确定词元和候选边界。在诸如英语这样的语言中，词就是候选词元，不过也存在缩写和首字母缩略词等特殊情况。诸如汉语，如有文本源，可以先进行分词处理。

接下来，如上一节所述，对于每个候选边界需要提取特征集合。对于语音数据，参考口语发音中通常没有词元开始时间及持续时间，但是计算韵律特征需要这些数据。一般地，解码过程中会强制进行对齐来获得这种特征。

一旦特征提取出来，每个候选边界可以用前几节描述的方法进行分类。

对于测试，自动估算的词元边界会与参考数据中的边界进行对比。当语音识别的输出用于训练或测试时，参考词元会与语音识别输出的词对齐，使用动态规划来最小化对齐错误（比如使用 NIST sclite 对齐工具），边界标注会转移到语音识别的输出中。不幸的是，有时完美的对齐并不存在。例如，参考标注中的两个词元间有句子边界，但这两个词元可能会被语音识别器识别为一个词元。在这种情况下，语音识别标注中是应该省略句子边界还是包含它，并不清楚，因此可使用启发式规则。

2.7 讨论

尽管句子分割是许多语言处理中非常有用的一个步骤，但实践发现，针对随后步骤细致优化分割参数比独立地优化预测句子边界的分割质量得到的效果要好。例如，Walker 等人 [91] 发现，与机器学习方法相比，使用手写规则进行句子分割的机器翻译系统的表现要差得多。Matusov 等人 [92] 发现优化源语言端句子分割参数对口语文档的机器翻译很有效。同样地，与单独优化句子分割任务不同，Favre 等人 [93] 以及 Liu 和 Xie [94] 分别研究了面向信息抽取和语音摘要的参数优化的效果。

对于主题分割，自动语音转写使用语言模型来预测语言模型的主题信息。不管是使用相同主题训练的语言模型，还是使用一个通用的语言模型，其中的主题作为一个隐变量在解码中进行估计，实验显示 ASR 的性能均有提高。更一般地，主题驱动的领域自适应在自然语言处理任务中被大量使用。在信息检索中，通过允许词在包含它的主题函数中有不同的贡献，可以显式建模主题模型；使用同现空间降维技术可以隐式地建模主题模型。在

自动摘要中，Tang、Yao 与 Chen [97] 提出需要重新审视“文档是由单一主题组成”这一假设，并且在他们的模型中加入了主题相关的信息。主题信息对词义消歧也有帮助，因为给定一个主题后，许多词一般会有一个主要的含义 [98]。

2.8 总结

本章描述了用文本和语音作为输入的句子和主题分割任务，还描述了许多处理这些任务的不同类别的算法。根据输入类别（如文本与语言）的不同，可以把许多不同种类的特征应用到这些任务中。例如，在文本中，排版的提示（大小写和标点符号）很有用，而对于语音，韵律特征非常实用。

与近期在语音处理和判别性机器学习的进展相同步，通过使用高维特征集，句子与主题分割系统的性能也得到提升。但是，这些系统仍然存在错误。因此随后的处理阶段，如机器翻译，必须足够鲁棒来应对噪声。未来需要对分割及随后的处理系统的联合优化做更多的研究。

参考文献

[1] J. Mrozinski, E. W. D. Whittaker, P. Chatain, and S. Furui, "Automatic sentence segmentation of speech for automatic summarization," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2005.

[2] J. Makhoul, A. Baron, I. Bulyko, L. Nguyen, L. Ramshaw, D. Stallard, R. Schwartz, and B. Xiang, "The effects of speech recognition and punctuation on information extraction performance," in *Proceedings of International Conference on Spoken Language Processing (Interspeech)*, 2005.

[3] D. Jones, W. Shen, E. Shriberg, A. Stolcke, T. Kamm, and D. Reynolds, "Two experiments comparing reading with listening for human processing of conversational telephone speech," in *Proceedings of EUROSPEECH*, pp. 1145–1148, 2005.

[4] W. Francis, H. Kučera, and A. Mackie, *Frequency Analysis of English Usage: Lexicon and Grammar*. Boston: Houghton Mifflin, 1982.

[5] M. Liberman and K. Church, "Text analysis and word pronunciation in text-to-speech synthesis," in *Advances in Speech Signal Processing* (S. Furui and M. M. Sondi, eds.), pp. 791–831, New York: Marcel Dekker, 1992.

[6] E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel, "OntoNotes: The 90% Solution," in *Proceedings of the Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, p. 57, 2007.

[7] L. Zhou and D. Zhang, "A heuristic approach to establishing punctuation convention in instant messaging," *IEEE Transactions on Professional Communication*, vol. 48, no. 4, pp. 391–400, 2005.

[8] A. Aw, M. Zhang, J. Xiao, and J. Su, "A phrase-based statistical model for SMS text normalization," in *Proceedings of the COLING/ACL*, 2006.

[9] K. Taghva, A. Condit, J. Borsack, and S. Erva, "Structural markup of OCR generated text," *Information Science Research Institute 1994 Annual Research Report*, p. 61, 1994.

[10] M. Stevenson and R. Gaizauskas, "Experiments on sentence boundary detection," in *Proceedings of the Conference on Applied Natural Language Processing (ANLP)*, 2000.

[11] J. Kolar, E. Shriberg, and Y. Liu, "Using prosody for automatic sentence segmentation of multi-party meetings," in *Proceedings of the International Conference on Text, Speech, and Dialogue (TSD)*, 2006.

[12] A. Janin, J. Ang, S. Bhagat, R. Dhillon, J. Edwards, J. Macias-Guarasa, N. Morgan, B. Peskin, E. Shriberg, A. Stolcke, C. Wooters, and B. Wrede, "The ICSI meeting project: Resources and research," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2004.

[13] M. Core and J. Allen, "Coding dialogs with the DAMSL annotation scheme," in *Proceedings of the Working Notes of the Conference of the American Association for Artificial Intelligence (AAAI) Fall Symposium on Communicative Action in Humans and Machines*, 1997.

[14] E. Shriberg, R. Dhillon, S. Bhagat, J. Ang, and H. Carvey, "The ICSI Meeting Recorder Dialog Act (MRDA) Corpus," in *Proceedings of the SigDial Workshop*, 2004.

[15] C. Hoffmann, "Automatische Disambiguierung von Satzgrenzen in einem maschinenlesbaren deutschen Korpus," Manuscript, University of Trier, Germany, 1994.

[16] G. Grefenstette and P. Tapanainen, "What is a word, what is a sentence? Problems of tokenization," Rank Xerox Research Centre, 1994.

[17] T. Briscoe, J. Carroll, and R. Watson, "The second release of the RASP system," in *Proceedings of the Interactive Demo Session of COLING/ACL*, vol. 6, 2006.

[18] C. L. Wayne, "Topic Detection and Tracking (TDT) overview and perspective," in *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[19] G. Doddington, "The Topic Detection and Tracking Phase 2 (TDT2) evaluation plan," in *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[20] P.-Y. Hsueh and J. Moore, "Automatic topic segmentation and labeling in multiparty dialogue," in *Proceedings of the 1st IEEE/ACM Workshop on Spoken Language Technology (SLT)*, 2006.

[21] D. Gillick, "Sentence boundary detection and the problem with the U.S.," in *Proceedings of NAACL: Short Papers*, 2009.

[22] M. A. Hearst, "TextTiling: Segmenting text into multi-paragraph subtopic passages," *Computational Linguistics*, vol. 23, no. 1, pp. 33–64, 1997.

[23] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, pp. 1260–1269, 1967.

[24] J. Yamron, I. Carp, L. Gillick, S. Lowe, and P. van Mulbregt, "A hidden Markov model approach to text segmentation and event tracking," in *Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 333–336, May 1998.

[25] K. W. Church, "A stochastic parts program and noun phrase parser for unrestricted text," in *Proceedings of the Conference on Applied Natural Language Processing (ANLP)*, pp. 136–143, 1988.

[26] D. M. Bikel, R. Schwartz, and R. M. Weischedel, "An algorithm that learns what's in a name," *Machine Learning Journal Special Issue on Natural Language Learning*, vol. 34, no. 1-3, pp. 211–231, 1999.

[27] E. Shriberg, A. Stolcke, D. Hakkani-Tür, and G. Tur, "Prosody-based automatic segmentation of speech into sentences and topics," *Speech Communication*, vol. 32, no. 1-2, pp. 127–154, 2000.

[28] A. Stolcke and E. Shriberg, "Statistical language modeling for speech disfluencies," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1996.

[29] G. Tur, D. Hakkani-Tür, A. Stolcke, and E. Shriberg, "Integrating prosodic and lexical cues for automatic topic segmentation," *Computational Linguistics*, vol. 27, no. 1, pp. 31–57, 2001.

[30] J. A. Bilmes and K. Kirchhoff, "Factored language models and generalized parallel backoff," in *Proceedings of the Human Language Technology Conference (HLT)-Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2003.

[31] U. Guz, B. Favre, G. Tur, and D. Hakkani-Tür, "Generative and discriminative methods using morphological information for sentence segmentation of Turkish," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 5, pp. 895–903, 2009.

[32] E. Stamatatos, N. Fakotakis, and G. Kokkinakis, "Automatic extraction of rules for sentence boundary disambiguation," in *Proceedings of the Workshop on Machine Learning in Human Language Technology*, pp. 88–92, 1999.

[33] M. D. Riley, "Some applications of tree-based modelling to speech and language indexing," in *Proceedings of the DARPA Speech and Natural Language Workshop*, pp. 339–352, 1989.

[34] D. Palmer and M. Hearst, "Adaptive sentence boundary disambiguation," in *Proceedings of the Fourth ACL Conference on Applied Natural Language Processing*, 1994.

[35] T. Humphrey and F. Zhou, "Period disambiguation using a neural network," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, p. 606, 1989.

[36] J. Shim, D. Kim, J. Cha, G. Lee, and J. Seo, "Multistrategic integrated web document pre-processing for sentence and word boundary detection," *Information Processing and Management*, vol. 38, no. 4, pp. 509–527, 2002.

[37] J. Reynar and A. Ratnaparkhi, "A maximum entropy approach to identifying sentence boundaries," in *Proceedings of the Conference on Applied Natural Language Processing (ANLP)*, 1997.

[38] H. Le and T. Ho, "A maximum entropy approach to sentence boundary detection of Vietnamesetexts," in *IEEE International Conference on Research, Innovation and Vision for the Future*, 2008.

[39] A. Mikheev, "Tagging sentence boundaries," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2000.

[40] M. A. Hearst, "Multi-paragraph segmentation of expository text," in ACL [99], pp. 9–16.

[41] T. Brants, F. Chen, and I. Tsochantaridis, "Topic-based document segmentation with probabilistic latent semantic analysis," in *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*, 2002.

[42] J. Morris and G. Hirst, "Lexical cohesion computed by thesaural relations as an indicator of the structure of text," *Computational Linguistics*, vol. 17, no. 1, pp. 21–48, 1991.

[43] M.-Y. Kan, J. L. Klavans, and K. R. McKeown, "Linear segmentation and segment significance," in *Proceedings ACL/COLING Workshop on Very Large Corpora*, Canada 1998.

[44] S. Banerjee and A. Rudnicky, "A TextTiling based approach to topic boundary detection in meetings," in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, 2006.

[45] M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing, "Discourse segmentation of multi-party conversation," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2003.

[46] M. Purver, K. Körding, T. Griffiths, and J. Tenenbaum, "Unsupervised topic modelling for multi-party spoken discourse," in *Proceedings of the International Conference on Computational Linguistics (COLING)—Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 17–24, 2006.

[47] J. Reynar, "Statistical models for topic segmentation," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 357–364, 1999.

[48] D. Beeferman, A. Berger, and J. Lafferty, "Statistical models for text segmentation," *Machine Learning*, vol. 34, no. 1-3, pp. 177–210, 1999.

[49] M. Georgescul, A. Clark, and S. Armstrong, "Word distributions for thematic segmentation in a support vector machine approach," in *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)*, pp. 101–108, 2006.

[50] A. Rosenberg and J. Hirschberg, "Story segmentation of broadcast news in English, Mandarin, and Arabic," in *Proceedings of the Human Language Technology Conference (HLT) and Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2006.

[51] G. A. Levow, "Assessing prosodic and text features for segmentation of Mandarin broadcast news," in *Proceedings of the Human Language Technology Conference (HLT)-Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) 2004*, 2004.

[52] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the 18th International Conference on Machine Learning (ICML)*, pp. 282–289, 2001.

[53] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support vector machine learning for interdependent and structured output spaces," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2004.

[54] B. Taskar, "Learning structured prediction models: A large margin approach," PhD thesis, Stanford University, 2004.

[55] K. Crammer, R. Mcdonald, and F. Pereira, "Scalable large-margin online learning for structured classification," in *Annual Conference on Neural Information Processing Systems (NIPS)*, 2005.

[56] H. Wallach, "Efficient training of conditional random fields," in *Proceedings of the Annual CLUK Research Colloquium*, vol. 112, 2002.

[57] S. Vishwanathan, N. Schraudolph, M. Schmidt, and K. Murphy, "Accelerated training of conditional random fields with stochastic gradient methods," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2006.

[58] S. Sarawagi and W. Cohen, "Semi-Markov conditional random fields for information extraction," *Advances in Neural Information Processing Systems*, vol. 17, pp. 1185–1192, 2005.

[59] H.-K. J. Kuo and Y. Gao, "Maximum entropy direct models for speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 14, no. 3, pp. 873–881, 2006.

[60] M. Zimmerman, D. Hakkani-Tür, J. Fung, N. Mirghafori, L. Gottlieb, E. Shriberg, and Y. Liu, "The ICSI+ multilingual sentence segmentation system," in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, 2006.

[61] D. Roark, Y. Liu, M. Harper, R. Stewart, M. Lease, M. Snover, I. Shafran, B. Dorr, J. Hale, A. Krasnyanskaya, and L. Yung, "Reranking for sentence boundary detection in conversational speech," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006.

[62] B. Favre, D. Hakkani-Tür, S. Petrov, and D. Klein, "Efficient sentence segmentation using syntactic features," in *Proceedings of the IEEE/ACL Spoken Language Technologies (SLT) Workshop*, 2008.

[63] M. Doss, D. Hakkani-Tür, O. Cetin, E. Shriberg, J. Fung, and N. Mirghafori, "Entropy based classifier combination for sentence segmentation," in *Proceedings of the IEEE ICASSP Conference*, pp. 189–192, 2007.

[64] M. Zimmerman, D. Hakkani-Tür, J. Fung, N. Mirghafori, L. Gottlieb, E. Shriberg, and Y. Liu, "The ICSI+ multilingual sentence segmentation system," in *Proceedings of the 9th International Conference on Spoken Language Processing*, ISCA, 2006.

53

[65] H. Kozima, "Text segmentation based on similarity between words," in *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pp. 286–288, 1993.

[66] J. C. Reynar, "An automatic method of finding topic boundaries," in ACL '99, pp. 331–333.

[67] J. M. Ponte and W. B. Croft, "Text segmentation by topic," in *Proceedings of the First European Conference on Research and Advanced Technology for Digital Libraries*, pp. 120–129, 1997.

[68] J. Kolar, Y. Liu, and E. Shriberg, "Genre effects on automatic sentence segmentation of speech: A comparison of broadcast news and broadcast conversations," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009.

[69] M. Johnson, "PCFG models of linguistic tree representations," *Computational Linguistics*, vol. 24, no. 4, pp. 613–632, 1998.

[70] B. Grosz and C. Sidner, "Attention, intention, and the structure of discourse," *Computational Linguistics*, vol. 12, no. 3, pp. 175–204, 1986.

[71] R. J. Passonneau and D. J. Litman, "Discourse segmentation by human and automated means," *Computational Linguistics*, vol. 23, no. 1, pp. 103–139, 1997. 54

[72] S. Banerjee and A. Rudnicky, "Segmenting meetings into agenda items by extracting implicit supervision from human note-taking," in *Proceedings of the International Conference on Intelligent User Interfaces (IUI'07)*, 2007.

[73] T. Kiss and J. Strunk, "Unsupervised multilingual sentence boundary detection," *Computational Linguistics*, vol. 32, no. 4, pp. 485–525, 2006.

[74] V. Warnke, R. Kompe, H. Niemann, and E. Nöth, "Integrated dialog act segmentation and classification using prosodic features and language models," in *Proceedings of the 5th European Conference on Speech Communication and Technology*, pp. 207–210, 1997.

[75] C. Chen, "Speech recognition with automatic punctuation," in *Proceedings of EUROSPEECH*, pp. 447–450, 1999.

[76] H. Christensen, Y. Gotoh, and S. Renals, "Punctuation annotation using statistical prosody models," in *Proceedings of the ISCA Workshop on Prosody in Speech Recognition and Understanding*, 2001.

[77] A. Srivastava and F. Kubala, "Sentence boundary detection in Arabicspeech," in *Proceedings of EUROSPEECH*, 2003.

[78] J.-H. Kim and P. C. Woodland, "A combined punctuation generation and speech recognition system and its performance enhancement using prosody," *Computer Speech and Language*, vol. 41, no. 4, pp. 563–577, Nov. 2003.

[79] M. Tomalin and P. C.Woodland, "Discriminatively trained Gaussianmixture models for sentence boundary detection," in *Proceedings of ICASSP*, pp. 549–552, 2006.

[80] Y. Liu, E. Shriberg, A. Stolcke, D. Hillard, M. Ostendorf, and M. Harper, "Enriching speech recognition with automatic detection of sentence boundaries and disfluencies," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1526–1540, 2006.

[81] L. Ferrer, E. Shriberg, and A. Stolcke, "Is the speaker done yet? Faster and more accurate end-of-utterance detection using prosody in human-computer dialog," in *Proceedings of the International Conference on Spoken Language Processing*, pp. 2061–2064, 2002.

[82] P. Boersma and D. Weenink, "Praat, a system for doing phonetics by computer, version 3.4," Tech. Rep. 132, Institute of Phonetic Sciences of the University of Amsterdam, 1996.

[83] J. Vaissière, "Language-independent prosodic features," in *Prosody: Models and Measurements* (A. Cutler and D. R. Ladd, eds.), ch. 5, pp. 53–66, Berlin: Springer, 1983.

[84] B. Grosz and J. Hirschberg, "Some intonational characteristics of discourse structure," in Ohala et al. [100], pp. 429–432.

[85] S. Nakajima and J. F. Allen, "A study on prosody and discourse structure in cooperative dialogues," *Phonetica*, vol. 50, pp. 197–210, 1993.

[86] J. Hirschberg and C. Nakatani, "A prosodic analysis of discourse segments in direction-giving monologues," in *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pp. 286–293, 1996.

[87] M. Swerts, "Prosodic features at discourse boundaries of different strength," *Journal of the Acoustical Society of America*, vol. 101, pp. 514–521, 1997.

[88] M. Swerts, R. Geluykens, and J. Terken, "Prosodic correlates of discourse units in spontaneous speech," in Ohala et al. [100] pp. 421–424.

[89] M. Georgescul, A. Clark, and S. Armstrong, "Exploiting structural meeting-specific features for topic segmentation," in *Actes de la 14è me Conférence sur le Traitement Automatique des Langues Naturelles*, Association pour le Traitement Automatique des Langues, June 2007.

[90] P.-Y. Hsueh, J. Moore, and S. Renals, "Automatic segmentation of multiparty dialogue," in *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2006.

[91] D. Wallace, D. Clements, M. Darwin, and J. Amtrup, "Sentence boundary detection: A comparison of paradigms for improving MT quality," in *Proceedings of the MT Summit VIII*, 2001.

[92] E. Matusov, D. Hillard, M. Magimai-Doss, D. Hakkani-Tür, M. Ostendorf, and H. Ney, "Improving speech translation with automatic boundary prediction," in *Proceedings of International Conference on Spoken Language Processing (Interspeech)*, 2007.

[93] B. Favre, R. Grishman, D. Hillard, H. Ji, D. Hakkani-Tür, and M. Ostendorf, "Punctuating speech for information extraction," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008.

[94] Y. Liu and S. Xie, "Impact of automatic sentence segmentation on meeting summarization," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008.

[95] J. Becker and D. Kuropka, "Topic-based vector space model," in *Proceedings of the 6th International Conference on Business Information Systems*, pp. 7–12, 2003.

[96] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[97] J. Tang, L. Yao, and D. Chen, "Multi-topic based query-oriented summarization," in *Proceedings of SDM*, 2009.

[98] J. Boyd-Graber, D. Blei, and X. Zhu, "A topic model for word sense disambiguation," in *Proceedings of the the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 1024–1033, 2007.

[99] *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, New Mexico State University, Las Cruces, New Mexico*. Morriston, NJ: ACL, 1994.

[100] J. J. Ohala, T. M. Nearey, B. L. Derwing, M. M. Hodge, and G. E. Wiebe, eds., *Proceedings of the International Conference on Spoken Language Processing*, Edmonton: University of Alberta, 1992.

第3章

Multilingual Natural Language Processing Applications: From Theory to Practice

句　法

Anoop Sarkar

句法分析揭示语言的内部结构，在自然语言处理等应用领域，句子的谓词-论元结构非常有用，语言的句法分析提供了一种手段，显式地发现句子中可能存在的各种谓词-论元的依存关系。在自然语言处理中，自然语言输入的句法分析可以是低层次的，如进行简单的词性标注；也可以是高层次的，比如，结构分析，识别句子论元间及其显式论元和隐式论元的依存关系。自然语言分析的主要瓶颈是普遍存在的歧义性。在句法分析中，歧义消解是特别困难的，因为句法分析树数目随着句子长度呈指数增长。从标注到句法分析，为了进行歧义消解，算法的选取显得特别重要。本章主要研究句法分析方法，从标注到全分析，以及应用有监督的机器学习方法进行歧义消解。

3.1　自然语言分析

在语音合成应用中，输入句子转换成语音输出，听起来像说母语的人说出的一样。考虑下面两个例子（想象它们读出来而不是写出来）[1]：

①He wanted to go for a drive in movie.

②He wanted to go for a drive in the country.

在第二个句子中，单词 *drive* 和 *in* 之间会有个自然的停顿，这表明句子有基本的内部结构。句法分析提供的结构性的描述可以识别发音的停顿。如下面这个简单的例子：

③The cat lives dangerously had nine lives.

在这个例子中，语音合成系统需要知道第一个单词 *lives* 是动词，而第二个单词 *lives* 是名词，才能做出正确的语调发音。这是词性标注的一个实例，句中的每个单词都赋予最可能的词性。上述这些例子来源于开源的 Festival 语音合成系统（www.festvox.org），该系统利用分析技术进行歧义消解。

句法分析的另一个动机是自然语言的自动文摘任务，把相同主题的若干文档浓缩成100--250词的文摘。文摘可以用来（可能以多种方式）回答文档集合的问题。在这种情况下，一个有用的子任务就是压缩单个句子，仅在文摘中保留相关的部分[1]。这使得文摘精确、信息量大、流利。下例中，句子④可以压缩成句子⑤：

④Beyond the basic level, the operations of the three products vary widely.

⑤The operations of the products vary.

完成上述任务的一个优秀方法是对句子进行句法分析，找出句子的不同成分，把句子划分成单独的短语，比如动词短语、名词短语。对第①个句子的句法分析输出结果如图3-1所示。句法分析的分析树经过压缩模型编辑，删除可选择成分，最后生成原始句子流利的压缩句。

---

[1] 在书面语表述中，第一个句子的 *drive in* 之间会有逗字符。

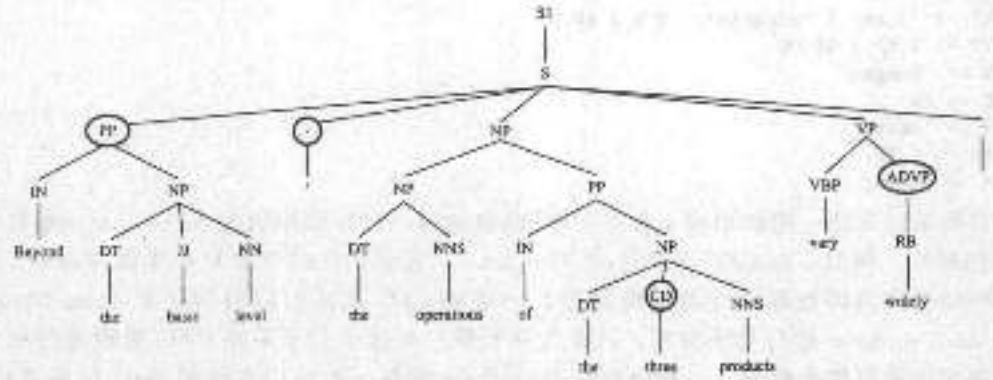


图 3-1 分析器对句子④的输出结果。删除画圈的非成成分 PP、CD 和 ADVP，产生一个较为流利的句子：The operations of the products vary（选自 Knight 和 Marcu[1] 的例子）

另一个例子是文本复述 [2]。在句子⑥中，大写的短语 EUROPEAN COUNTRIES 可以在不改变句子基本意思的情况下由其他短语复述，句子⑦～⑪中的斜体就是该短语的复述实例。这种短语替换不是简单的任意词语替换，因为容易导致不连贯和不流畅的复述。复述模型是在句法分析的基础上进行，识别目标短语成分，找到合适的替换短语，最终对原始短语进行替代。复述在统计机器翻译等领域有着广泛的用途。 58

⑥open borders imply increasing racial fragmentation in EUROPEAN COUNTRIES.

⑦open borders imply increasing racial fragmentation in *the countries of europe*.

⑧open borders imply increasing racial fragmentation in *european countries*.

⑨open borders imply increasing racial fragmentation in *europe*.

⑩open borders imply increasing racial fragmentation in *european countries*.

⑪open borders imply increasing racial fragmentation in *the european countries*.

在现阶段的自然语言处理中，句法分析可用于很多领域而不仅是统计机器翻译 [3]、文本信息抽取 [4]、语言摘要 [5]、在语言生成中产生实体网格 [6]、文本语法校正 [7]、从语言中获取知识（例如发现语义类或 x IS A y 关系）[8]、在语音识别系统中作为语言模型（语言模型为候选输出句子赋予一个概率—句法对不流利的或有错的语音输入尤为有用）[7]、对话系统 [9]、语言合成系统（www.festvox.org）。句法分析是多语言处理任务的必要组成部分，现已有多种自然语言的句法分析器。

3.2 树库：句法分析的数据驱动方法

句法分析可以揭示输入句子的不明确信息，这意味着，分析器需要除了输入句子之外的其他信息；句法分析结果的输出形式。提供这种信息的一种方法就是写出该语言的文法规则集合。例如，可以写出上下文无关文法（Context-Free Grammer，CFG）的句法规则。本章的其余部分都假设读者熟悉 CFG（参考 Sipser [10]，对形式文法及其产生的形式语言，尤其是 CFG，有很好的介绍）。

下面是一个 CFG（以简单的 Backus-Naur 格式书写），表示英语中及物动词的一个简单文法，及物是可以带主语或宾语名词短语（Noun Phrase，NP）的动词（V），加上动词短语（Verb Phrase，VP）的修饰语，如介词短语（Prepositional Phrase，PP），

```
S -> NP VP
NP -> 'John' | 'pockets' | D N | NP PP
VP -> V NP | VP PP
V -> 'bought'
D -> 'a'
N -> 'shirt'
PP -> P NP
P -> 'with'
```

自然语言文法一般以单词 ω 作为 CFG 的终结符，产生式规则为 $X \to \omega$，X 一般代表单词 ω 的词性。例如，在 CFG 的产生式 $V \to$ 'saw'，表示词性的符号 V 产生动词 saw。这样的非终结符称为词性标记或者前终结符。上述的 CFG 文法可以对句子如 John bought a skirt with pockets 进行句法分析，S 作为开始符。对这个句子采用 CFG 规则进行句法分析，得到两种可能的推导。一种分析是把 pockets 看做一种可用来购买 skirt 的流通货币；另一种分析理解更为普通，即 John 购买了一个有口袋的 skirt。

```
(S (NP John)
   (VP (VP (V bought)
           (NP (D a)
               (N shirt)))
       (PP (P with)
           (NP pockets))))
```

```
(S (NP John)
   (VP (V bought)
       (NP (NP (D a)
               (N shirt))
           (PP (P with)
               (NP pockets)))))
```

然而为自然语言写出 CFG 文法进行句法分析是有问题的。不像程序设计语言，自然语言太过复杂以至于不能够列出所有的 CFG 规则。一组简单的文法规则不考虑文法不同组成部分的交互影响，可以考虑拓展文法，包括更多的结构和词汇类型，但是对于语言而言列出所有的句法结构是项很困难的工作。另外，穷尽列出单词的所有性质也是困难的，例如，列出与单词有关的所有语法规则，这是一个典型的知识获取问题。

除了知识获取，仍存在一个明显问题：规则之间可互相作用产生组合爆炸。考虑一个简单的 CFG，对下面的名词短语进行句法分析：

```
N -> N N
N -> 'natural' | 'language' | 'processing' | 'book'
```

递归规则产生歧义：N 作为开始符，输入第一个单词 natural 产生一棵分析树（N natural），继续输入单词 natural language，利用递归规则产生一棵分析树（N (N natural) (N language)），继续输入 natural language processing，应用递归规则两次，产生两棵分析树：

```
(N (N (N natural)
      (N language))
   (N processing))
```

```
(N (N natural)
   (N (N language)
      (N processing)))
```

注意到这个句子的句法分析歧义反映了真歧义：这是一种自然语言处理吗？还是语言处理的一种自然方式？这个问题不能仅通过改变书写规则的形式体系得以解决（例如，应用有限状态自动机，它是确定性的，但不能根据单一语法同时给出一个句子的两个意思）。任何句法规则系统都应当表示出这种歧义。然而，应用递归规则 3 次，可以得到句子 *natural language processing book* 的 5 种分析树；再长一些的名词短语，运用递归 4 次，可以得到 14 种分析树；递归 5 次，42 种分析树；递归 6 次，可以得到 132 种分析树。事实上，对于 CFG，运用递归 n 次可以分析得到的分析树的数目为 Catalan 数：

$$\mathrm{Cat}(n) = \frac{1}{n+1}\binom{2n}{n}$$

不仅是并列结构，如名词短语，递归规则也存在于修饰语，如本节开始部分提到的CFG产生式 VP→VP PP 中的介词短语。事实上，介词短语修饰语的歧义并非独立于并列结构歧义：在两类歧义都存在的句子中，输入句子的句法分析树数目等于子文法的分析树数目的叉积。这使得句法分析的时间复杂度很高。对 n 个单词的输入句子，其所有可能的分析树是 n 的幂次。

对大多数自然语言处理任务，不需要搜索整个歧义空间，即使（本节后面会提到）我们可在多项式时间内（对 CFG，时间复杂度为 $O(n^3)$）把指数数量级的分析树数目进行压缩，产生一个紧致的表示，并且存储在多项式空间内（对于 CFG，所需空间在 n^2 的数量级）。

例如，对输入句子 *natural language processing book*，用CFG分析得到的5种分析树中，仅有一种是正确的（理解为 a book about the processing of natural language）：

```
(N (N (N (K natural)
         (K language))
      (N processing))
   (N book))
```

这是第二种知识获取问题。不仅需要知道一种语言的句法规则，还需要知道输入句子的各种分析结果中，哪种分析最合理。树库的构建采用句法分析数据驱动的方法，可以一次性解决两种知识获取的瓶颈问题。

树库简单来说是句子的集合（也称文本语料库），其中每个句子都有完整的句法分析结果。每个句子的句法分析结果都由人类专家判定以作为该句最合理的分析。在人工标注阶段，需要重点关注，以保证对相关的语法现象进行了一致的处理。典型地，在人工标注开始前，先制定一个标注指南，以保证树库标注的一致性。 51

树库没有提供句法规则或语言文法，也没有明确地列出句法结构。事实上，即使树库中隐含着一个句法假设，也不可能存在穷尽的规则集。关于句法更细粒度的假设经常用作为标注指南，以帮助人类专家标注语料时，产生语料库中句子的单个最合理的句法分析。树库中句法分析的一致性可以通过标注者间的一致性来衡量，即不止一个标注者标注大约10%的重叠语料。

树库解决了我们前面讨论的知识获取的两个瓶颈问题。树库提供了大量句子示例的句法结构的标注，可以运用有监督的机器学习方法，通过适当泛化树库的训练语料，训练一个句法分析器，对输入句子进行句法分析。

树库通过找出隐含在句法分析树中的文法解决了第一个知识获取问题，因为句法分析树而不是文法已经直接给出了。事实上，句法分析器不一定需要显式的文法规则集，只要它可以忠实地对输入句子产生一个句法分析树，尽管训练的句法分析器使用的信息也可以被认为代表了一些隐式的文法规则集。Nivre[11] 进一步讨论了应用文法进行分析和应用数据驱动的方法进行分析的微妙差别（数据驱动的方法不一定是基于文法的）。

树库同样也解决了第二个知识获取问题。因为树库里的每个句子已给出最合理的句法分析，有监督的机器学习方法可以用来学习一个评分函数，对所有可能的句法分析结果打分。用树库训练的统计句法分析器试图模仿人类的标注决策，应用输入的某些指示以及分析器先前的决策结果，来学习评分函数。对在训练数据中未出现的句子，统计句法分析器应用评分函数返回得分最高的一个句法分析结果，认被当作该句子最合理的分析。评分函数也可以用来对句子产生 k-best 句法分析。

两种主要的句法分析方法用来构建树库：依存图和短语结构树。这两种表示相互之间

很接近，在一定假设条件下，一种表示也可以转换成另一种。依存分析一般用于词序较自由的语言，比如捷克语、土耳其语，其谓词-论元在句中的顺序可变；短语结构树分析一般用于词序较固定的语言，如英语、法语，可提供长距离的依存信息。

本章的其余部分介绍了构建句法分析器的3个主要部分：3.3节涉及利用不同语言知识来构建树库；3.4节处理指数级的搜索空间；3.5节提供评价分析树的方法，找出最可
62 能的分析结果。

3.3　句法结构的表示

3.3.1　使用依存图的句法分析

依存图的主要思想是连接短语的中心词与其依存词，用有向边（因此不对称）把中心词与依存词连接起来[12]。依存图与短语结构树一样，是和很多不同的语言学框架一致的一种表示方法。中心词与依存词的依存关系可以是语义上的（中心词-修饰语，head-modifier），也可以是句法上的（中心词-限定语，head-specifier）。依存图与短语结构树的主要不同是，依存分析一般对句法结构做最小的假设，并且避免隐藏结构的任何标注，例如，用空元素作为占位符以表示缺失、取代谓词-论元或任何不必要的层次结构。输入句子的单词被视为图中的节点，节点之间用有向弧连接起来表示句法的依存性。CoNLL2007依存分析共享任务[13]上分享了任务，提供了下述依存图的定义：

基于依存的句法分析，其任务是通过识别句子每个单词的句法中心，推导出输入句子的句法结构。定义依存图为：其节点是输入句子的单词，弧是二值关系，从中心词指向依存词。经常（但不总是）假定所有单词除了一个之外都具有句法中心词，这意味着图是一棵有一个独立节点作为根的树。在有标签的依存分析中，我们同时需要分析器为每个中心词和依存词之间的依存关系指定一个特定的类型（或者标签）。

据此定义，我们只考虑依存树分析，其中每个单词都准确地依存于一个父节点，或者其他单词或虚拟的根符号。按规定，在依存树中索引0被用来表示根符号，有向弧从中心词指向依存词。例如，图3-2展示了一个捷克句子的依存树，句子来源于布拉格依存树库，这是一个标记了依存树的捷克语文本的大语料库。每个树库都有自己的标注风格，布拉格树库也标注了其他信息，比如主题和句子焦点结构，但是我们这里只展示依存树信息。

有很多不同的依存句法分析，但是依存树的基本文本结构可以按下列形式写出，在句子中每个依存词明确指定一个中心词，并且仅有一个单词依存于句子的根节点。下面展示
63 了一个典型的有标签的依存树的原文表示：

索　引	单　词	词　性	中　心	标　签
1	They	PRP	2	SBJ
2	personded	VBD	0	ROOT
3	Mr.	NNP	4	NMOD
4	Trotter	NNP	2	IOBJ
5	to	TO	6	VMOD
6	take	VB	2	OBJ
7	it	PRP	6	OBJ
8	back	RB	6	PRT
9	.	.	2	P

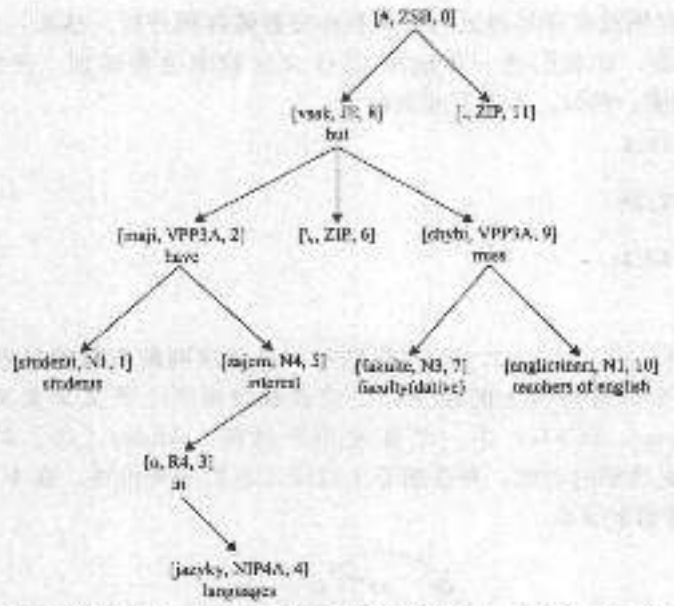


图 3.2 布拉格依存树库中的一个捷克语句子的依存图句法分析实例。图中每个节点是一个词，词性，在句子中的位置的三元组，例如 [fakulte, N3, 7] 是句子的第 7 个词，词性为 N3，受格，节点 "*"，ZSB，0" 是依存树的根节点。每个节点都添加了英语翻译

依存分析中一个重要的概念是**投射性**（projectivity），是由单词之间依存的线性词序决定的一种约束 [14]。如果我们把根符号在第一位置的句子的单词按照线性顺序排列，那么单词之间的依存弧画出来没有任何的交叉，就是**投射性依存树**（projective dependency tree）。投射性的另一种表述是，对句子的每个单词，其后代形成一个句子的连续子串。例如，图 3-3 展示了一个英语句子的依存分析，该句子右端有个后置的名词短语作为修饰语，结果是需要交叉依存。然而，英语在树库中有很少的例子需要这样的非投射分析，在其他语言中，比如捷克语、土耳其语，非投射分析的数量就比较高。交叉依存即使是在这些语言中，在整个依存数量中所占的比例也很小。然而，在一定比例的句子中包括至少一个交叉依存，这在一些语言中也是一个重要的问题。表 3-1 包含多种语言交叉依存的对比，这是 CoNLL2007 依存分析共享任务的一部分。 84

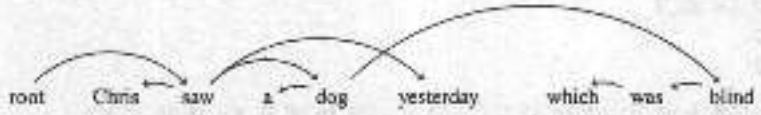


图 3.3 具有交叉依存的无标签非投射依存树

表 3-1 交叉依存比例的多种语言对比和非投射句子的比例，来自 CoNLL2007 共享任务数据集。Ar = Arabic、Ba = Basque、Ca = Catalan、Ch = Chinese、Cz = Czech、En = English、Gr = Greek、Hu = Hungarian、It = Italian、Tu = Turkish。注意一些依存树的例子是由原始的短语结构树通过一些启发式规则转换来的。来自 Nivre 等 [13]

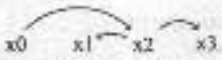
	Ar	Ba	Ca	Ch	Cz	En	Gr	Hu	It	Tu
%deps	0.4	2.9	0.1	0.0	1.9	0.3	1.1	2.9	0.5	1.5
%sents	10.1	25.8	2.9	0.0	23.2	7.7	20.3	22.4	7.4	11.3

在树库中，依存图没有详尽地区别投射和非投射依存树分析。然而，分析算法经常区别投射和非投射依存，让我们进一步应用 CFG 文法检测这种区别。注意，我们可以在 CFG 中设置依存连接。例如，考虑下述文法：

```
X0_2 -> X0_1* X2_1
X0_1 -> x0*
X2_1 -> X1_1 X2_2*
X1_1 -> x1*
X2_2 -> X2_3* X3_1
X2_3 -> x2*
X3_1 -> x3*
```

55

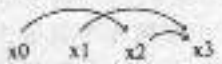
在这个 CFG 中，x0、x1、x2、x3 是终结符，每条规则最右部的星号表示依存连接。可以把星号看做一个非终结符上的独立标记或者看做概率上下文无关文法（Probalistic Contex Free Grammar，PCFG）中一个新的非终结符。Abney [15] 对投射依存图的 PCFG 形式提供了更详细的对比，并在细节上讨论了它们的等价性。在本例中，等价于前面的 CFG 的依存树如下所示：



我们可以证明，如果把依存树转换成等价的 CFG（应用上面的记号），那么依存树是投射性的。在由依存树转换来的 CFG 中，我们得到仅有的下面 3 条规则，其中一条规则是引入终结符，其余两条规则是 Y 依赖于 X，反之亦然。中心词 X 或者 Y 可以由下面的星号跟踪。

```
Z -> X* Y
Z -> X Y*
A -> a*
```

假定我们有一棵非射影依存树，例如：



用星号标记将这棵依存树转换为 CFG，给予我们两种选择。一种可以描述为 X3 依赖于 X2，但是不能描述为 X1 依赖于 X3：

```
X2_3 -> X1_1 X2_2*
X1_1 -> x1
X2_2 -> X2_1* X3_1
X2_1 -> x2
X3_1 -> x3
```

另一种可以描述为 X1 依赖于 X3，但是不能描述为 X3 依赖于 X2：

```
X2_3 -> X1_1 X3_2*
X1_1 -> x1
X3_2 -> X2_1 X3_1*
X2_1 -> x2
X3_1 -> x3
```

事实上，CFG 不能描述非投射性依存。投射性可以定义如下：对句子的每个单词，其子节点形成句子的连续子串。因此，非投射性可以定义如下：非投射依存意味着句子中有一个词（或等价地，由依存树创建的 CFG 的一个非终结符），其子节点不能形成句子的连续子串。换句话说，对 $p>0$ 存在一个 Z 可以推导出跨度 (x_1, x_k) 以及 (x_{k+p}, x_l)。这意味着

66

一定有规则 $Z \rightarrow PQ$，其中 P 推导出 (x_i, x_k) 并且 Q 推导出 (x_{k+p}, x_j)。然而，由定义，仅当 $k=0$ 时，这种推导在 CFG 文法下才是有效的，因为 P 和 Q 一定是连续子串。所以，非投射性依存的依存树不能转换成等价的（星号标记的）CFG。

这就用 CFG 给了投射依存一个有用的鉴证。如果我们想要一个仅得到投射依存的依存分析器，则可以隐含地创建一个等价的 CFG，这样就会忽略所有非投射依存。当我们讨论分析算法时会进一步研究这个话题。

3.3.2 使用短语结构树的句法分析

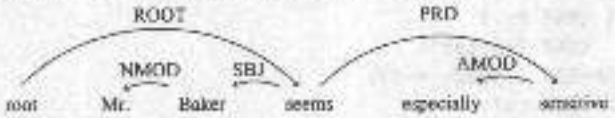
句子的短语结构句法分析源于传统的句子图解法，即把句子分割为组成成分，较大的组成成分由较小的组成成分合并得到。短语结构分析通常吸收生成文法（来自语言学）的观点处理组成成分词序或者明显的中心词与依存词之间的长距离关系。短语结构树可以被视为隐含地与谓词-论元结构联系在一起。例如，下列句子 *Mr. Baker seems especially sensitive* 的短语结构分析（来自英语宾州树库），句子的主语用 -SBJ 标记，句子的谓词用 -PRT标记。基本的谓词-论元结构在树下面展示，使用由短语结构树描述的信息的非正式标注。

```
(S (NP-SBJ (NNP Mr.)
           (NNP Baker))
   (VP (VBZ seems)
       (ADJP-PRD (RB especially)
                 (JJ sensitive))))
```

谓词-论元结构：

```
seems((especially(sensitive))(Mr. Baker))
```

对上面同一个句子可得到下面的依存树分析。应当注意一些短语结构树中括号标签内的信息是如何映射到依存分析的有标记的弧上。通常，依存分析不会直接连接主语与谓语，因为这会在 *seems* 和根符号之间带来不便利的交叉依存。



为了解释树库中短语结构分析的一些细节，我们使用句法分析的一些例子来说明空元素（没有输出的组成成分）在树结构中如何被用来确定谓词-论元间的依存关系。这些例子选自文献 [16]，描述了英语宾州树库的标注标准。英语宾州树库是用短语结构树标注了摘自《华尔街日报》的 40 000 个句子的工程。为了简化短语结构树，单词的词性标签被省略。

在第一个例子，我们看到 NP 支配一个“迹”标记 *T*，表示空元素。与形式语言理论中的符号 ε 一样，表示空输入。这个空标记有索引值（这里是 1，但是实际值是不重要的），并且与句子组成成分 WHNP 有相同的索引值。这个共同的索引值使得我们可以推断句子的谓词-论元结构（显示在短语结构树下面）。

```
(SBARQ (WHNP-1 What)
       (SQ is (NP-SBJ Tim)
              (VP eating (NP *T*-1)))
       ?)
```

谓词-论元结构：

```
eat(Tim, what)
```

在第二个例子中，由于在被动语态中句子的主语被置换，句子的主语 *The ball* 实际上不是句子谓词的逻辑主语。句子的逻辑主语 *Chris*，被标记为-LGS，从而保证了这个句子的谓词-论元结构的恢复。

```
(S (NP-SBJ-1 The ball)
   (VP was (VP thrown)
           (NP *-1)
           (PP by (NP-LGS Chris))))
```

谓词-论元结构：

```
throw(Chris, the ball)
```

第三个例子展示了不同的句法现象在语料库中经常被结合在一起，并且两种分析被结合起来用以提供这些情况下谓词-论元的结构。

```
(SBARQ (WHNP-1 Who)
       (SQ was (NP-SBJ-2 *T*-1)
               (VP believed (S (NP-SBJ-3 *-2)
                               (VP to (VP have
                                          (VP been
                                              (VP shot
                                                  (NP *-3))))))))
       ?)
```

谓词-论元结构：

```
believe(*someone*, shoot(*someone*, who))
```

58 第四个例子展示了空元素如何被用来标记缺失的谓词主语，即使谓词主语没有直接出现在句中。在第一种情况下，树库中短语结构标记出缺失的 *take back* 的主语，即动词 *persuaded* 的宾语。

```
(S (NP-SBJ (PRP They))
   (VP (VP (VBD persuaded)
           (NP-1 (NNP Mr.)
                 (NNP Trotter))
           (S (NP-SBJ (-NONE- *-1))
              (VP (TO to)
                  (VP (VB take)
                      (NP (PRP it))
                      (PRT (RB back))))))))
```

谓词-论元结构：

```
persuade(they, Mr. Trotter, take_back(Mr. Trotter, it))
```

在第二种情况下，树库中短语结构标记出缺失的 *take back* 的主语，即动词 *promised* 的主语。

```
(S (NP-SBJ-1 (PRP They))
   (VP (VP (VBD promised)
           (NP (NNP Mr.)
               (NNP Trotter))
           (S (NP-SBJ (-NONE- *-1))
              (VP (TO to)
                  (VP (VB take)
                      (NP (PRP it))
                      (PRT (RB back))))))))
```

谓词-论元结构：

```
promise(they, Mr. Trotter, take_back(they, it))
```

对 *persuaded* 和 *promised* 的依存分析不会做这种区分。对上例中两个句子的依存分析将会是一样的，如下所示：

```
1 They      PRP 2 SBJ
2 persuaded VBD 0 ROOT
3 Mr.       NNP 4 NMOD
4 Trotter   NNP 2 IOBJ
5 to        TO  6 VMOD
6 take      VB  2 OBJ
7 it        PRP 6 OBJ
8 back      RB  6 PRT
9 .         .   2 P
```

```
1 They      PRP 2 SBJ
2 promised  VBD 0 ROOT
3 Mr.       NNP 4 NMOD
4 Trotter   NNP 2 IOBJ
5 to        TO  6 VMOD
6 take      VB  2 OBJ
7 it        PRP 6 OBJ
8 back      RB  6 PRT
9           .   2 P
```

然而，当指出依存树库和短语结构树库中标记思想的差异时，注意用短语结构树库训练的大多数的统计分析器通常忽略这些差异。逻辑主语、空元素等大量的标记在现代分析器中几乎都被忽略。已经有一些工作在试图恢复空元素，其最初被用在英语宾州树库中，而在训练统计分析器时被丢弃。例如，Johnson [17] 在后处理阶段恢复了空元素并且识别它们的先行词。由 Rimell、Clark、Steedman [18] 提出的评估模板，就先前在几个例子中展示的每个句子谓词-论元结构的恢复而言，展示了如何比较不同的分析器。

同种语言的不同树库，或者不同种语言的同种树库，短语结构标记可能会有很大的不同。符号的选择以及符号的意义会有不同。下面的例子来自中文树库，符号 *IP* 用来代替 S，这反映了从基于短语结构的美国宾州树库转换文法到基于支配约束（Goverment Binding，GB）的短语结构的转变。不同之处还会与特殊的句法结构有关。在下例中，对所有格“的”做了特别分析，导致对“新的”的包含几个空元素的、相当复杂的结构分析，其中一个空元素是 WHNP，即使中文并没有关系代词。为了理解整个树库中从句和类从句成分的短语结构一致性，需要这种结构。这些不同意味着，最初在英语分析上开发并在英语树库上训练的短语结构分析器不容易适用于另一种语言，即使这种语言有短语结构树库。Levy 和 Manning [19] 讨论了把基于 CFG 分析器（最初为英语分析开发）用于中文短语结构树库训练的中文句法分析过程中的很多挑战。

```
(IP (NP-SBJ (NP (NN 结算/settlement and sale)
                (NN 制度/system))
            (CC 和/and)
            (NP (CP (WHNP-2 (-NONE- *OP*))
                    (CP (IP (NP-SBJ (-NONE- *T*-2))
                            (VP (VA 新/new)))
                        (DEC 的)))
                (NP (NN 核销/verification and cancellation)
                    (NN 制度/system))))
    (VP (PP-LOC (P 在/in)
                (NP-PN (NR 西藏/Tibet)))
        (ADVP (AD 全面/fully))
        (VP (VV 实施/operating))))
```

英语翻译为：

```
A (foreign exchange) settlement and sale system and a verification and
cancellation system that is newly created is fully operational in Tibet.
```

3.4　分析算法

70

给定输入句子，分析器给出句子的输出分析，我们假定这种分析与用于训练分析器的树库保持一致。树库分析器不需要详细的文法，但是为了使得分析算法的解释更简单，我们首先考虑分析算法假定存在一个 CFG 文法。

考虑下面简单的 CFG 文法 G，可以用来推导字符串，例如开始符为 N 的字符串 *a and b or c*：

```
N -> N 'and' N
N -> N 'or' N
N -> 'a' | 'b' | 'c'
```

分析的一个很重要的概念是**推导**（derivation）。对于输入字符串 "*a and b or c*"，下面的动作序列由一>符号分开，表示一系列步骤，称为推导：

```
N
=> N 'or' N
=> N 'or c'
=> N 'and' N 'or c'
=> N 'and b or c'
=> 'a and b or c'
```

在这个推导中的每一行称为句型（sentential form）。此外，推导的每一行都应用 CFG 规则。最为了说明输入可以由开始符 N 推导出。在上述推导中，我们限制每个句型仅从最右非终结符开始扩展。这种方法称为使用 CFG 输入的**最右推导**（rightmost derivation）。如果我们把推导按相反的次序显示，则最右推导的一个有意思的性质就显示了出来：

```
'a and b or c'
=> N 'and b or c'          # 使用规则 N -> a
=> N 'and' N 'or c'        # 使用规则 N -> b
=> N 'or c'                # 使用规则 N -> N and N
=> N 'or' N                # 使用规则 N -> c
=> N                       # 使用规则 N -> N or N
```

这种推导序列与下面的从左到右的句法树构建完全相同，每次一个符号。

```
(N (N (N a)
      and
      (N b))
   or
   (N c))
```

然而，不能保证得到一个唯一的推导序列。可能会有很多不同的推导，正如我们前面看到的，推导的数目随输入长度的增加而呈指数增长。例如，存在另一个最右推导产生下面的分析树：

71

```
(N (N a)
   and
   (N (N b)
      or
     (N c)))
'a and b or c'
=> N 'and b or c'          # 使用规则 N -> a
=> N 'and' N 'or c'        # 使用规则 N -> b
=> N 'and' N 'or' N        # 使用规则 N -> c
=> N 'and' N               # 使用规则 N -> N or N
=> N                       # 使用规则 N -> N and N
```

3.4.1 移进归约分析

为了构建一个分析器，我们需要设计一个算法，该算法对任何文法以及任何输入句子都能够执行之前的最右推导。每一个 CFG 文法都有一个自动机与之等价，称为下推自动机（正如正则表达式可以转换为有限状态自动机）。下推自动机是一种简单的有限状态自动机，具有栈形式的额外的内存。这是一个限量的内存，因为只有栈顶元素被机器使用。这提供了一种分析算法，适用于任何给定的 CFG 文法和输入字符串。这种算法称为**移进归约**（shift-reduce）分析，使用两种数据结构：输入字符的缓冲区和存储 CFG 符号的栈。该算法定义如下：

1）以空栈和包含输入字符的缓冲区开始；

2）如果栈顶元素包含文法的开始符并且缓冲区为空，则返回成功；

3）选择下面的两个步骤之一（如果选择有歧义，则按照预定义的策略）：

- 把符号从缓冲区移入栈；
- 如果栈顶的 k 个符号是 $\alpha_1\alpha_2\cdots\alpha_k$，符合 CFG 规则 $A\rightarrow\alpha_1\alpha_2\cdots\alpha_k$ 的右边部分，则用非终结符 A（规则左边部分）取代栈顶的 k 个符号 $\alpha_1\alpha_2\cdots\alpha_k$；

4）如果上一步没有相应动作，则返回失败；

5）否则，回到步骤 2。

对本节前部分出现过的 CFG 文法 G 以及输入 "*a and b or c*"，我们在图 3-4 展示了移进归约分析算法的每个步骤。

分 析 树	栈	输 入	动 作
		a and b or c	初始化
a	a	and b or c	移进 a
(N a)	N	and b or c	归约 N->a
(N a) and	N and	b or c	移进 and
(N a) and b	N and b	or c	移进 b
(N a) and (N b)	N and N	or c	归约 N->b
(N (N a) and (N b))	N	or c	归约 N->a
(N (N a) and (N b)) or	N or	c	移进 or
(N (N a) and (N b)) or c	N or c		移进 c
(N (N a) and (N b)) or (N c)	N or N		归约 N->c
(N (N (N a) and (N b)) or (N c))	N		归约 N->a or N
(N (N (N a) and (N b)) or (N c))	N		接受

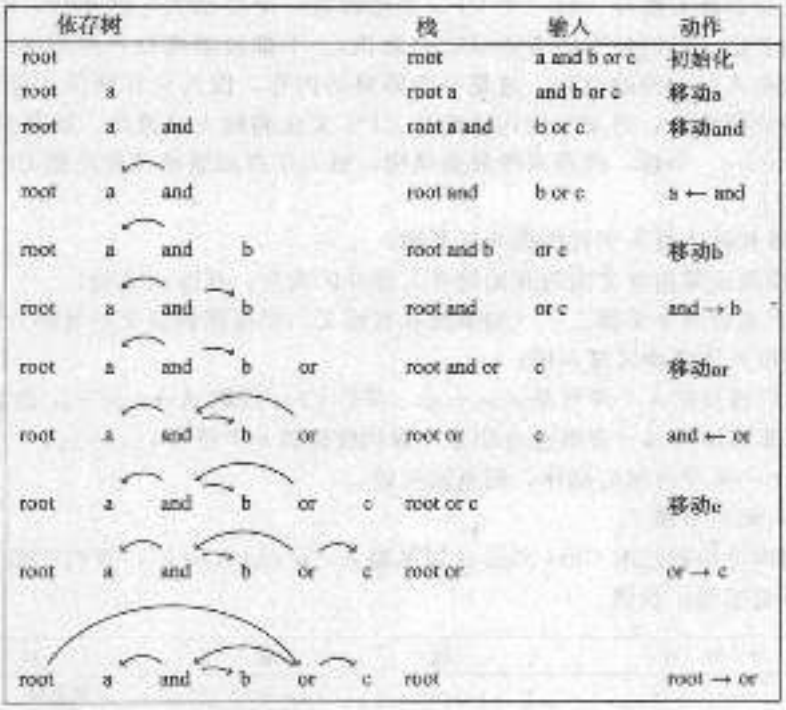
图 3-4 对本节开始定义的 CFG 文法 G 以及输入 "*a and b or c*"，移进归约分析算法的每个步骤

该算法也适用于依存分析，应用移进归约分析器对上个例子进行依存分析如图 3-5 所示。在每一步，分析器会选择：或者移新词入栈或者用中心词→依存词连接或依存词→中心词连接结合栈顶两个元素。当在统计依存分析器中使用移进归约算法时，应尽可能把移进和归约步骤合并。Nivre [20] 讨论了其他分析方法，对分析器行为和统计决策之间的关系有不同处理。

3.4.2 超图和线图分析

移进归约分析可以在线性时间内分析，但是要在不犯错误的情况下。对于一般的 CFG 文法，在最坏的情形下，这样的分析器可能要借助于回溯，这意味着要重新分析输入。这

样就导致了在最坏的情形下时间随着文法大小呈指数增长。另一方面，CFG 文法的最坏情况分析算法的复杂度为$O(n^3)$，其中 n 为输入句子的长度。这种算法的多种变化形式常用于统计分析器，试图搜索所有可能的分析树空间，而不受限于纯自左至右的分析。



72 / 73

图 3.5　对于依存分析，移进归约分析算法的步骤

我们的 CFG 文法 G 的实例如下所示：

```
N -> N 'and' N
N -> N 'or' N
N -> 'a' | 'b' | 'c'
```

可以重写为新的 CFG 文法 G_r，其右边至多包含两个非终结符。这可以通过引入两个新的非终结符 N^和 Nv 做到：

```
N -> N N^
N^ -> 'and' N
N -> N Nv
Nv -> 'or' N
N -> 'a' | 'b' | 'c'
```

对这种分析算法的一个关键性认知是，我们可以通过创建一个新的 CFG，把上面的 CFG 文法 G_r 针对一个特定的输入字符串进行专门化，这个新的 CFG 表示对这个特定的输入句子有效的文法 G_r 下所有可能的分析树的一个紧致编码。例如，对于输入字符串 "*a and b or c*"，这个新的 CFG 文法 G_f 表示下面显示的分析树的**森林**（forest）。想象输入字符串被分解为跨度：0 *a* 1 *and* 2 *b* 3 *or* 4 *c* 5，那么 a 即是跨度 0，1，字符串 "*b or c*" 在这个字符串中是跨度 2，5。在这个森林文法 G_f[6] 中的非终结符包括跨度信息。应用这个文

---

⑥ 原文为 G_r，疑误。　译者注

法产生的不同的分析树对于输入句子都是有效分析树。

```
N[0,5] -> N[0,1] N^[1,5]
N[0,3] -> N[0,1] N^[1,3]
N^[1,3] -> 'and'[1,2] N[2,3]
N^[1,5] -> 'and'[1,2] N[2,5]
N[0,5] -> N[0,3] Nv[3,5]
N[2,5] -> N[2,3] Nv[3,5]
Nv[3,5] -> 'or'[3,4] N[4,5]
N[0,1] -> 'a'[0,1]
N[2,3] -> 'b'[2,3]
N[4,5] -> 'c'[4,5]
```

以这种观点，分析算法可以被定义为：对一个 CFG 和输入字符串，产生专门对输入的所有合理分析紧致表示的 CFG，如图 3-5 所示。分析器需要创建所有有效的专门规则集 74
或者创建一条从跨整个字符串的开始符号到由单词组成的叶子节点的路径。

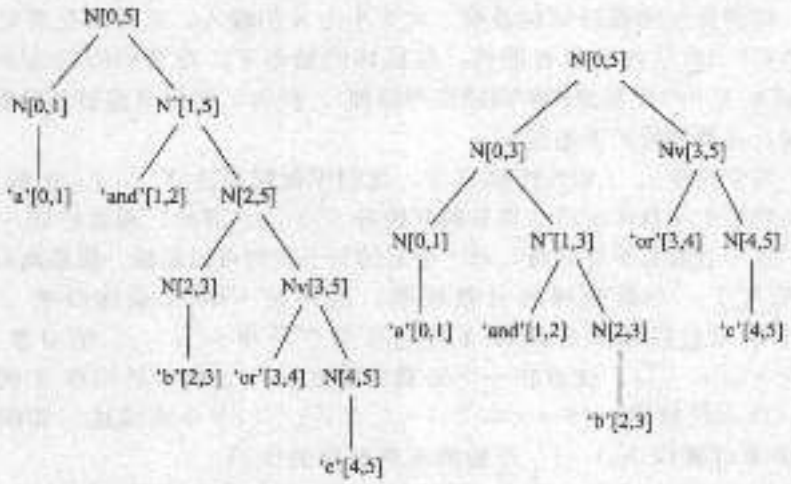


图 3-5 用于给定输入字符串的、嵌入专门 CFG 的分析树。在相同标签的节点，比如N[0，5]、N[0，1]、'and'[1，2]、N[2，3] 和 Nv[3，5]，可以合并形成一个对输入进行的所有分析的紧图表示

让我们检查一下构建专门的 CFG 分析器需要采取的步骤。首先我们仅生成词的规则：

```
N[0,1] -> 'a'[0,1]
N[2,3] -> 'b'[2,3]
N[4,5] -> 'c'[4,5]
```

这些规则可以简单地通过检查对任意输入单词 x 的类型 $N \to x$ 规则的存在性和创建对单词 x 的专门规则来构建。这一步骤的伪代码如下所示：

```
for i = 0...n do
  if 对任意 x 跨度 i+1，存在具有分数 s 的规则 N → x，then
    增加具有分数 s 的专门规则 N[i,i+1] → x[i,i+1]
    记为：N[i,i+1]：s
  end if
end for
```

下一步是基于对先前产生的专门规则递归地产生新的专门规则。如果先前创建的规则左边存在Y[i,k]和 Z[k,j]，并且如果在 CFG 内有规则 $X \to YZ$，则我们可以推断应该存在新的专门规则 X[i，j]→Y[i,k]Z[k,j]。每个非终结符跨度被赋于一个分数 s，X[i,j]:s。每个

75 非终结符仅保留最高得分的跨度。因此 $X[i,j]=\max_s X[i,j]_s$。

```
for j = 2 ... n do
  for i = j - 1 ... 0 do
    for k = i + 1 ... j do
      if Y[i,k]:s1 和 Z[k,j]:s2 在专门文法内 then
        if X → YZ 分数 s，在原始文法中存在 then
          增加专门规则 X[i,j] → Y[i,k]Z[k,j]，分数为 s = s1 + s2
          保留最高得分的规则：X[i,j] → α
        end if
      end if
    end for
  end for
end for
```

这称为 CKY 算法（以 Cocke、Kasami、Younger 命名，他们各自独立发现该算法）。该算法考虑每个长度的跨度，以每种可能的方式分割跨度，并检测跨度是否可由 CFG 规则推导出，最终我们要保证能寻找到一条规则（如果这条规则存在的话），其跨度是整个输入字符串。检测算法的循环结构表明，对大小为 n 的输入，该算法花费时间为 n^3。然而，从专门的 CFG 穷尽列出所有的树，在最坏的情形下，花费的时间为指数幂（基于 CFG 最坏的情形下可产生指数棵树的同样的推理）。然而，使用有监督的机器学习挑选出最有可能的树，花费的时间不超过 n^3。

注意对于每个跨度 i、j 和非终结符 X，我们仅保留到达 $X[i,j]$ 的最高得分路径。因此，可以从跨整个字符串的最高得分的开始符 $S[0][n]$ 开始，通过扩展 $S[0][n]$ 的右部分并且递归这一过程直到终结符，对于给定的句子我们可以创建一棵最高分数分析树。

在概率框架下，分数被视为对数概率，这即是 Viterbi 最优分析（Viterbi-best parse）。每个单元包括由非终结符 X 经过推导字符串 $w[i,j]$ 的对数概率，可以写为 $\Pr(X \Rightarrow^* w[i,j])$。注意在一个特别的跨度 i、j 上的非终结符 X 的作用依赖于开始符 S，可以由外概率 $\Pr(S \Rightarrow^* w[0,i-1]Xw[j-1,N])$ 来描述。实际上，可以使用内、外概率来计算以 $X[i,j]$ 开始的每条规则的作用。

有很多方式可以通过去除一些不太可能的搜索空间加快分析器。例如，我们可以对比 $X[i,j]$ 的分数与 $Y[i,j]$ 当前的最高分数，如果 $X[i,j]$ 与 $Y[i,j]$ 比可能性太小，则舍弃任何以 $X[i,j]$ 开始的规则。这可能导致搜索错误（失去了分数最高的分析树），但是一般情况下这种情况不会发生，因而我们在更快的分析时间和精确度间做出权衡。这种分析技术称为柱阈值（beam thresholding）。我们可以通过增加全局非负阈值限制来扩充它。例如，如果以 $X[i,j]$ 开始的规则加没有相邻规则与之合并，是不行的。这种技术称为全局阈值（global thresholding）。如果我们有一个非常复杂的非终结符集合（如
76 图 3-7 所示），则可以先用粗粒度的非终结符代替更细粒度的非终结符（例如，用简单的非终结符 VP 代替非终结符 VP-S）进行分析，然后使用粗糙的非终结符 $VP[i,j]$ 的分数修剪同一跨度上更细粒度的非终结符。这种方法称为粗到细的分析（coarse to fine parsing），这种分析很有用，因为除了内概率，在粗分析步骤中的外概率可以用来做更有效的剪枝。柱阈值、全局阈值、粗到细的分析，这三种技术在 Joshua Goodman [21] 分析 PCFG 时都被讨论过。

分析器可以进一步利用 A^* 搜索加快，而不使用先前提到的算法 [22] 穷尽搜索整个分析空间，大量启发式的选择使得 A^* 搜索可以提供更快的分析速度，而在最坏情形下其复杂度与 CKY 算法一样。

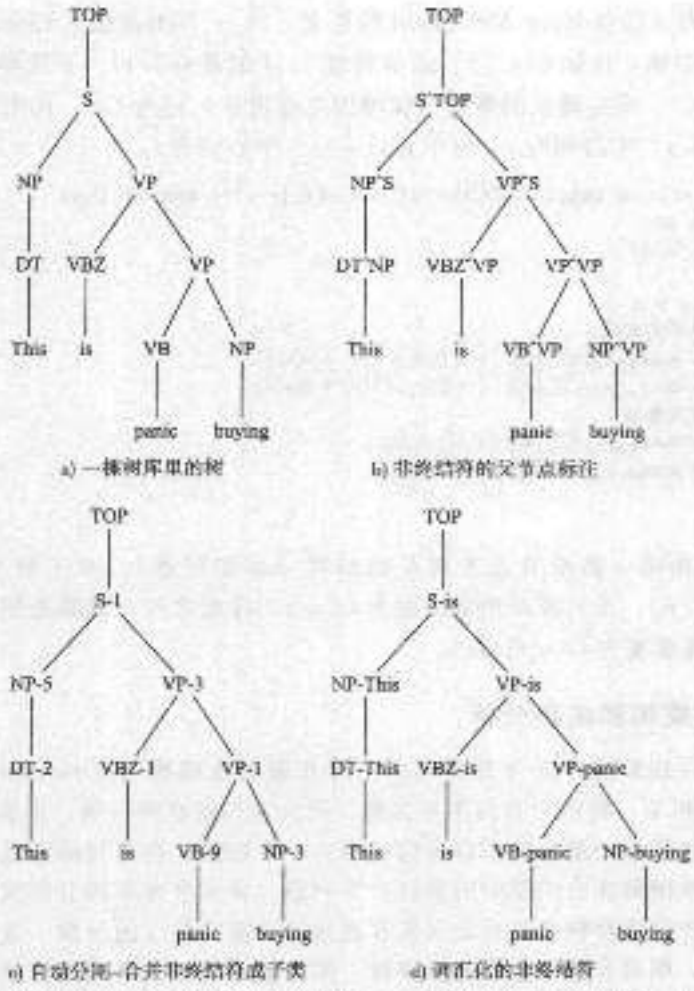


图 3.7 为上除独立性假设（不基于分析执行），一棵树库的树被转换成：a）原始树库里的树，从数据中很容易提取出 PCFG；b）把父节点标签转接到每个节点标签上；c）对每个非终结符使用未监督学习方式创建子类；d）通过词汇化非终结符使用词汇项过独热模型

对于投射性依存分析，同样的算法可以通过创建一个产生依存分析的 CFG 使用（前面章节已提到）。然而，对于依存分析，上面的循环在最坏的情形下有 n^5 数量级，因为每个 Y 和 Z 都是词汇化的，在最坏的情形下存在 n 个不同的非终结符 Y 和 n 个不同的非终结符 Z，这样在 CKY 算法的内循环中就有 n^5 种不同的组合。

然而，对于依存分析，Eisner [23] 观察到不使用因单词增加的非终结符，面对输入字符串的每个跨度的不同依存树集合进行紧致表示更有优势。其思想就是独立地收集中心词的左右依存词，然后在下一步合并它们。这样带来了**分割中心词**（split-head）的概念，中心词被分割成两部分：一个对左依存词，一个对右依存词。除了中心词，存储跨度的每一项，我们都存储一个标记指明中心词是在收集左依存词还是右依存词，并且该存储项是否完整（完整的项不能被扩展为更多的依存部分）。这样在最坏情形下依存分析算法是 n^3 数量级。这样也减少了中间状态的数目，不允许左依存和右依存的任何交叉操作，不同于用于依存分析的 CKY 算法。

下面的伪代码（源自 Ryan McDonald 的论文 [24]）详细描述了 Eisner 算法。跨度以线图数据结构 C 存储，比如 $C[r][t]$ 表示跨度 r、t 的依存分析。不完整的跨度记做 C^{i}，完整的跨度记做 C^{c}，向左增长的跨度（仅增加左边依存）记为 $C_{\leftarrow}$，向右增长的跨度记为 $C_{\rightarrow}$。对于 $C_{\leftarrow}[r][t]$ 中心词为 t，对于 $C_{\rightarrow}[r][t]$ 中心词为 r。

```
初始化: for s=1...n chart C^c[s][s] = 0.0 for d ∈ {←, →} and c ∈ {i, c}
for k = 1...n do
  for s = 1...n do
    t = s + k
    break if t > n
    首先: 创建不完整项
    C^i_←[s][t] = max_{s≤r<t} C^c_←[s][r] + C^c_→[r+1][t] − s(t, s)
    C^i_→[s][t] = max_{s≤r<t} C^c_→[s][r] + C^c_←[r+1][t] − s(s, t)
    其次, 创建完整项
    C^c_←[s][t] = max_{s≤r<t} C^c_←[s][r] + C^i_←[r][t]
    C^c_→[s][t] = max_{s<r≤t} C^c_→[s][r] + C^i_→[r][t]
  end for
end for
```

77 / 78

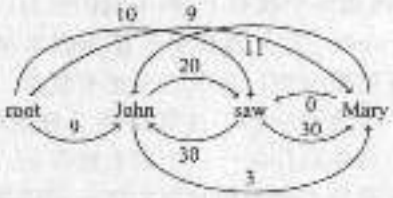
我们假定存在唯一的根节点为最左边的词（如前所述）。对于整个句子，最优树的分数为 $C_{\rightarrow}[1][n]$。运行算法的复杂度为 $O(n^3)$。除此之外，该算法还可以扩展为提供 k-best 分析，其复杂度为 $O(n^3 k \log k)$。

3.4.3　最小生成树和依存分析

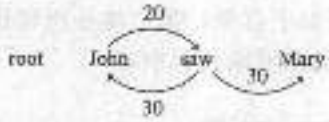
在有向图中寻找最优分支与在无向图中寻找**最小生成树**（Minimum Spanning Tree，MST）问题紧密相关。我们对有向图有兴趣，因为它与依存树一致，总是有根节点并且无环。前提是单词之间每个潜在的依存连接应当有一个权值。在自然语言处理中，传统方法是借助于最小生成树解决有向图中的最优分支问题。在依存树库的分析实例中，我们假定有一些模型可以提供依存树中单词之间依存连接的可能性估计的分数。这些分数可以用来找出最小生成树，即具有最高分数的依存树。因为输入句子的单词线性顺序并没有考虑在最小生成树的框架内，这样交叉或非投射依存就会被这种分析器找到。这对于英语这样投射性的语言可能会存在问题，但是对于像捷克语这样的语言却提供了一种找出交叉依存的很自然的方式。

下面并没有给出最小生成树算法的伪代码（McDonald [24] 有提供），我们只是展示了 MST 算法是如何使用一个依存分析的例子工作的。

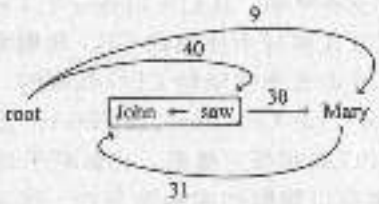
对输入句子 *John saw Mary* 考虑下面的全连接图。每条边都具有权重，值的计算基于边上的一些求分函数（计算这些分数源于边上不同的特征，这些特征下节讨论）。



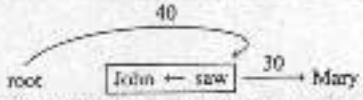
第一步是找出分值最高的入（incoming）边。如果这一步最终得到是一棵树，那么我们把这棵分析树作为分析结果返回，因为这棵树就是最小生成树。在本例中，经过在图中挑选最高分值的入边，得到一个回路。



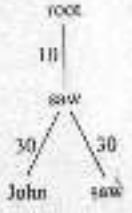
把回路收缩为单个节点，并且计算这些边的权重。当计算每个节点到收缩节点的边权重时，我们要记住合并节点的哪个组成部分有最大的权值。比如，对于上面的图，计算入边：$root \rightarrow |saw \rightarrow John|$：wt=40 与 $root \rightarrow |John \rightarrow saw|$：wt=29；计算入边：$Mary \rightarrow |saw \rightarrow John|$：wt=30 与 $Mary \rightarrow |John \rightarrow saw|$：wt=31。 79



我们在这个图中递归运行最小生成树算法，意味着找出图中每个词的最佳入边。在这个例子中，对比：$root \rightarrow Mary \rightarrow |John \rightarrow saw|$：wt=9+31 与 $root \rightarrow |John \rightarrow saw| \rightarrow Mary$：wt=40+30，这样形成了下面的图：



展开递归步骤即得到最高得分的输入依存分析的最小生成树：



3.5 分析中的歧义消解模型

本节主要集中讨论分析中的难题：如何设计特征并消解分析中的歧义。在3.4节讨论分析算法时包含运用模型进行高效分析的内容。3.4节的算法为本节描述的模型所用，用来寻找最高分数分析树或依存分析，有时也可用来训练模型。

3.5.1 概率上下文无关文法

考虑先前讨论过的歧义问题，下面我们将会对句子 *John bought a shirt with pockets* 中有歧义的分析做出一个选择。 80

```
(S (NP John)
   (VP (VP (V bought)
           (NP (D a)
               (N shirt)))
       (PP (P with)
           (NP pockets))))
```

```
(S (NP John)
   (VP (V bought)
       (NP (NP (D a)
               (N shirt))
           (PP (P with)
               (NP pockets)))))
```

我们想提出一个模型，基于如下直觉：第二棵分析树优于第一棵分析树。这两棵分析树可看作是下述 CFG 歧义的（最左或最右）推导：

```
S -> NP VP
NP -> 'John' | 'pockets' | D N | NP PP
VP -> V NP | VP PP
V -> 'bought'
D -> 'a'
N -> 'shirt'
PP -> P NP
P -> 'with'
```

为了对每个推导提供分数或者概率，我们可以在这个 CFG 规则上附加分数或者概率。推导的概率是分数之和或者所有在推导中使用的 CFG 规则概率的乘积。因为分数可以简单地被视为对数概率，当分数或者概率被赋给 CFG 规则时，我们即是在使用概率上下文无关文法（Probabilitic Context-Free Grammar，PCFG）。为保证由 PCFG 生成的树集合可以被很好地定义，我们为 CFG 规则指定概率，比如对于规则 $N \to \alpha$，概率即是 $P(N \to \alpha \mid N)$，那就是说，每条规则概率以规则的左边为条件。这意味着，在上下文无关的非终结符扩展中，概率要分布在该非终结符所有的扩展规则上。换句话说：

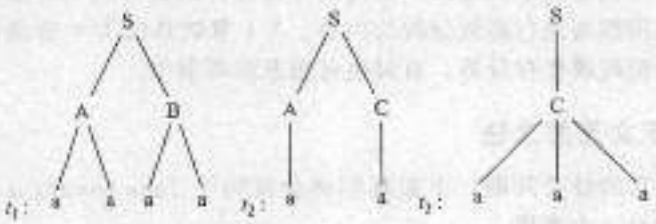
$$1 = \sum_{\alpha} P(N \to \alpha)$$

因此，在我们的例子中，为 CFG 中的规则指定一个概率，是为了使得更合理的分析具有更高的概率。

```
S -> NP VP (1.0)
NP -> 'John' (0.1) | 'pockets' (0.1) | D N (0.3) | NP PP (0.5)
VP -> V NP (0.9) | VP PP (0.1)
V -> 'bought' (1.0)
D -> 'a' (1.0)
N -> 'shirt' (1.0)
PP -> P NP (1.0)
P -> 'with' (1.0)
```

从上述规则概率可以看出，决定输入句子 *"John bought a shirt with pockets"* 分析结
81 果的规则仅是 NP→NP PP 和 VP→VP PP，因为其他规则在两种分析中都有出现。由于 NP→NP PP 在先前 PCFG 中设置的概率值较高，因此合理的句法分析结果具有更高的概率。

规则概率可以从树库中计算出，如下例所示。考虑具有三棵树 t_1、t_2、t_3 的树库：



假定树 t_1 在树库中出现 10 次，树 t_2 在树库中出现 20 次，树 t_3 在树库中出现 50 次，则该树库的 PCFG 规则概率计算为：

$$\frac{10}{10+20+50} = 0.125 \quad S \to A\ B$$

$$\frac{20}{10+20+50} = 0.25 \quad S \to A\ C$$

$$\frac{50}{10+20+50} = 0.625 \quad S \rightarrow C$$

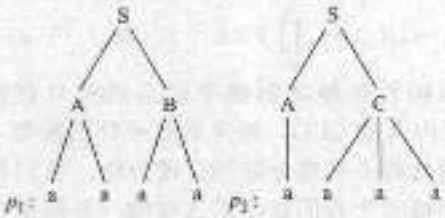
$$\frac{10}{10+20} = 0.334 \quad A \rightarrow a\ a$$

$$\frac{20}{10+20} = 0.667 \quad A \rightarrow a$$

$$\frac{20}{20+50} = 0.285 \quad B \rightarrow a\ a$$

$$\frac{50}{20+50} = 0.714 \quad C \rightarrow a\ a\ a$$

对于输入句子“$a\ a\ a\ a$”，应用上面的 PCFG 分析为两棵分析树：



其概率计算分别为 $p_1 = 0.125 \times 0.334 \times 0.285 = 0.01189$，$p_2 = 0.25 \times 0.667 \times 0.714 = 0.119$。第二棵分析树 p_2 即为输入句子最可能的分析树。最可能的分析树在树库中甚至是不存在的！这也是 PCFG 上下文无关的一个重要性质，即被非终结符可被左部使用的任何其他规则扩展。为了采取更合适的独立性假设，统计分析器采取的一般方法是扩展节点标签以避免精炼的独立性假设。

宾州树库包含的树如图 3-7a 所示。第一种方法是通过标记非终结符的父节点移除一些独立性假设 [25]。第二种方法是通过使用未监督的学习算法自动地学习这些非终结符 [32]
的分割 [26]（树中的分割—合并使用期望最大化（Expectation Maximization，EM）算法）。第三种方法 [27] 是词汇化非终结符，这样可以创建更好的模型，因为在考虑连接附属节点时把具体的单词也考虑在内。

当每个非终结符词汇化之后，标准的分析算法要做出适当的调整以处理众多的词汇化规则。因为稀疏问题，在模型中具体使用词汇化的非终结符时也要特别谨慎。经过非终结符词汇化之后的 PCFG，非终结符的展开从中心词开始：先预测中心词，产生左子树，然后产生右子树。

对于给定的输入找出最可能的分析树的另一种方式是找出最可能的组成成分集合。基本思想是找出一棵有最多正确组成成分的树，而不是得分最高的树。Goodman [28] 提到 CKY（我们在 3.4.2 节中给出了定义）算法可用内、外概率的乘积而不仅内概率取代对每个 $X[i, j]$ 的评分函数，最终找出具有最多正确组成成分集合的分析树，这种技术经常被称为最大规则分析（max-rule parsing），并且可以生成 PCFG 无法生成的分析树，与我们之前讨论的例子相似。最大规则分析明确地在每个成分级别上最大化召回率，因此在分析器评估中经常给出较高的召回率。

3.5.2 句法分析的生成模型

为了找出最合理的分析树，分析器要从可以表示为决策序列的可能推导中做出选择。假设用以构建分析树的决策序列，即推导为 $D = d_1, \cdots, d_n$。对输入句子 x，输出分析树

y 可由推导步骤序列定义。我们引入每个推导的概率：

$$P(x,y) = P(d_1,\cdots,d_n) = \prod_{i=1}^{n} P(d_i \mid d_1,\cdots,d_{i-1})$$

概率 $P(d_i \mid d_1, \cdots, d_{i-1})$ 中的条件部分称为**历史**（history），相当于一棵部分建成的分析树（由推导序列定义）。我们做一个简单的假设，用函数 Φ 把历史分组为等价类，使条件部分成为一个有限集合。

$$P(d_1,\cdots,d_n) = \prod_{i=1}^{n} P(d_i \mid \Phi(d_1,\cdots,d_{i-1}))$$

对所有的 x、y 应用函数 Φ，将每个历史 $H_i = d_1, \cdots, d_{i-1}$ 映射到函数 $\phi_1(H_i),\cdots,\phi_k(H_i)$ 特征函数的一些固定的有限集合。由这 k 个特征函数：

$$P(d_1,\cdots,d_n) = \prod_{i=1}^{n} P(d_i \mid \phi_1(H_i),\cdots,\phi_k(H_i))$$

83 然而，PCFG 的定义意味着各种规则概率应当调整以获得正确的分析分数。并且，PCFG 的独立性假设受制于内含的 CFG，经常导致不好的模型，这种模型不能使用有效信息对规则分数进行选择，因此得不到高分数的合理分析。我们希望可以使用分析树的任意特征对这些歧义建模。判别模型为我们提供了这样的一类模型。

3.5.3　句法分析的判别模型

Collins[29] 拓展了 Freund 和 Schapire[30] 的思想，创建了一种简单的记号和框架，可描述不同的判别方法以学习分析（分块或标注）。这种框架称为**全局线性模型**（global linear model）[29]。假设 x 为输入集合，y 为可能的输出集合，可以是词性序列、分析树或依存分析树。

- 每一个 $x \in x$，$y \in y$ 映射到一个 d 维的特征向量 $\Phi(x, y)$，其中的每一维都是实数，概括了包含在 (x, y) 内的部分信息；
- 表示特征重要性的权重参数向量 $w \in R^d$ 对应于 $\Phi(x,y)$ 每个特征的权重。$\Phi(x,y) \times w$ 的值表示 (x, y) 的分数，分数越高 y 作为 x 的输出结果的可能性越大；
- 函数 $GEN(x)$ 表示输入 x 对应的所有输出 y 的集合。

有了 $\Phi(x, y)$、w 以及 $GEN(x)$ 的详细说明，我们可以选择属于集合 $GEN(x)$ 并且分数最高的 y^* 作为最合理的输出结果，即

$$F(x) = \underset{y \in GEN(x)}{\operatorname{argmax}} p(y \mid x, w)$$

其中，$F(x)$ 返回属于集合 $GEN(x)$ 的分数最高的 y^*。**条件随机场**（Conditional Random Field，CRF）[31] 把条件概率定义为每个候选 y 和全局归一化项的线性分数：

$$\log p(y \mid x, w) = \Phi(x,y) \cdot w - \log \sum_{y' \in GEN(x)} \exp(\Phi(x,y') \cdot w)$$

忽略归一化项的一个简单的全局线性模型为：

$$F(x) = \underset{y \in GEN(x)}{\operatorname{argmax}} \Phi(x,y) \cdot w$$

很多分析的实验结果表明，全局线性模型忽略归一化项之后可以更快地训练模型，在精确度上与更高代价的归一化模型的训练结果一样。

感知机（perceptron）[32] 最初是作为单层的神经网络被引入的。运用在线学习的训练方式（即一次处理一个实例），调整权重参数向量，该向量以后可用于对输入数据进行分析以产生相应的输出结果。权重调整过程中，对出现在事实中的特征进行奖励，若没有

出现则进行相应的惩罚。经过更新之后，感知机模型能确保当前的权重参数向量可以正确识别当前的训练实例。

假定训练集上有 m 个实例。原始的感知机学习算法 [32] 如算法 3-1 所示：

算法 3-1 原始感知机学习算法

输入：训练数据，$\{(x_1,y_1),\cdots,(x_m,y_m)\}$；迭代次数 T
初始化：设置 $w=0$
算法：

```
for t = 1, ..., T do
  for i = 1, ..., m do
    计算 y_i', 其中 y_i' = argmax_{y∈GEN(x_i)} Φ(x_i, y)·w
    if y_i' ≠ y_i then
      w = w + Φ(x_i, y_i) − Φ(x_i, y_i')
    end if
  end for
end for
```

输出：更新的权重参数向量 w

权重参数向量 w 初始设置为 0。然后通过这 m 个训练实例算法进行迭代。对每个实例 x，产生候选集合 $GEN(x)$，根据当前的权重参数向量 w 选出具有最高分数的最合理候选。之后，算法把选择的候选与事实比较，如果不相同则更新权重 w：特征出现在事实中则权重值相应增加；若特征出现在这个最高的候选内则减少权重值。如果训练数据是线性可分离的，那么这意味着训练数据可以利用一个特征的线性组合函数进行区分，学习过程可证明在有限次数的迭代后收敛 [30]。

原始感知机学习算法易于理解和分析。然而，增量式权重更新具有过拟合的问题，从而导致可以很好地分类训练数据但是以不可见数据的结果更差为代价。并且，感知机算法不能处理线性不可分离的训练数据。

Freund 和 Schapire [30] 提出了一种变型的感知机学习方法，即投票的感知机算法（voted perceptron algorithm）。算法的学习过程并没有用单个权重参数向量存储和更新参数值，而是跟踪所有的中间权重向量，这些中间权重向量在分类阶段用来对答案投票。算法的初衷是较好地预测向量有较长的生存周期，因此在投票时有较高的权重。算法 3-2 源自 Freund 和 Schapire（有微小的改动），展示了投票感知机训练和预测阶段。

算法 3-2 投票感知机算法

训练阶段：
输入：训练数据 $\{(x_1,y_1),\cdots,(x_m,y_m)\}$；迭代次数 T
初始化：$k=0, w_0=0, c_1=0$
算法：

```
for t = 1, ..., T do
  for i = 1, ..., m do
    计算 y_i', 其中 y_i' = argmax_{y∈GEN(x_i)} Φ(x_i, y)·w_k
    if y_i' = y_i then
      c_k = c_k + 1
    else
      w_{k+1} = w_k + Φ(x_i, y_i) − Φ(x_i, y_i')
      c_{k+1} = 1
      k = k + 1
    end if
  end for
end for
```

输出：权重向量列表 $\{(w_1,c_1),\cdots,(w_k,c_k)\}$

预测阶段：

输入：权重向量列表 $\{(w_1, c_1), \ldots, (w_k, c_k)\}$，句子 x

计算：

$$y^* = \underset{y \in GEN(x)}{\operatorname{argmax}} \left(\sum_{i=1}^{k} c_i \Phi(x, y) \cdot w_i \right)$$

输出：投票最高的候选 y^*

投票感知机中参数 c_i 用以记录特定的权重参数向量 (w_i, c_i) 在训练过程中的生存次数。对于一个训练实例，如果其选择的最高候选词与事实不同，则新的记录次数变量 c_{i+1} 被赋值为 1，然后更新权重向量 (w_{i+1}, c_{i+1})，同时将原有的 c_i 和权重向量 (w_i, c_i) 存储起来。

与原始感知机比，由于为投票而维持中间权重向量的列表，投票感知机更为稳定。然而，存储这些中间权重向量是低效率的。并且，在预测阶段使用所有的中间权重向量进行权重计算，也是耗时的。

平均感知机算法 [30]（averaged perceptron algorithm）是投票感知机的一种近似，换句话说，维持了投票感知机算法算法的稳定性，但有效地减少了空间和时间复杂度。在平均感知机算法中，没有用权重参数 w，而是使用 m 个训练实例的平均权重参数向量 γ 对未知数据进行预测：

$$\gamma = \frac{1}{mT} \sum_{i=1 \cdots m, t=1 \cdots T} w^{i,t}$$

85
|
86

为了计算 γ，维护一个累积参数向量 σ，并且用每个训练实例的 w 更新。在最后一次迭代后，$\sigma/(mT)$ 产生出最终的参数向量 γ。算法 3.3 展示了整个算法。

算法 3.3　平均感知机学习算法

输入：采样数据 $\{(x_1, y_1), \ldots, (x_m, y_m)\}$；迭代次数 T

初始化：设置 $w = 0, \gamma = 0, \sigma = 0$

算法：

```
for t = 1, ..., T do
  for i = 1, ..., m do
    计算 y_i' 其中 y_i' = argmax_{y∈GEN(x_i)} Φ(x_i, y) · w
    if y_i' ≠ y_i then
      w = w + Φ(x_i, y_i) − Φ(x_i, y_i')
    end if
    σ = σ + w
  end for
end for
```

输出：平均权重参数向量 $\gamma = \sigma/(mT)$

当特征数量很多时，计算每个训练实例的参数 σ 的代价很高。为了进一步减少时间复杂度，Collins [33] 提出了一种懒惰更新程序，避免了在每次迭代过程中更新整个权重向量。处理每个训练句子后，并不是 σ 所有的维都被更新。相反，更新向量 τ 被用来存储准确位置 (p, t)，即平均参数向量的每一维最后被更新的位置，当然只有出现在当前句子中的特征相应的维才会被更新，这里 p 代表最后被更新的特征的训练实例索引，t 表示其相应的迭代次数。在最后一个实例的最后一次迭代中，无论其候选输出结果正确与否，τ 的每一维都得到更新。算法 3.4 展示了基于懒惰更新程序的平均感知机算法。

算法 3-4 基于贪婪更新程序的平均感知机学习算法

```
输入：训练数据 {(x_1, y_1), ..., (x_m, y_m)}；迭代次数 T
初始化：设置 w = 0, γ = 0, σ = 0, τ = 0
算法：
  for i = 1, ..., T do
    for i = 1, ..., m do
      计算 y_i', 其中 y_i' = argmax_{y∈GEN(x_i)} Φ(x_i, y) · w
      if i ≠ T or i ≠ m then
        if y_i' ≠ y_i then
          // 更新当前句子的活跃特征
          for {Φ(x_i, y_i) − Φ(x_i, y_i')} 每一维 s do
            if 这时量 s 的维度 then
              // 包括自从上次更新以来本特征一直不活跃的时段的所有权重
              σ_s = σ_s + w_s · (i · m + i − τ_s · m − i_{τ_s})
            end if
            // 根据 y_i' 与 y_i 对比时计算的权重
            w_s = w_s + Φ(x_i, y_i) − Φ(x_i, y_i')
            σ_s = σ_s + Φ(x_i, y_i) − Φ(x_i, y_i')
            // 记录维度 s 更新后位置
            τ_s = (i, i)
          end for
        end if
      else
        // 在最后一次循环中处理最后一个句子
        for
          // 包括自从上次更新以来 s 的每个特征一直不活跃的时段的所有权重
          σ_s = σ_s + w_s · (T · m + m − τ_s · m − i_{τ_s})
        end for
        // 更新在这个最后句子出现的特征的权重
        if y_i' ≠ y_i then
          w = w + Φ(x_i, y_i) − Φ(x_i, y_i')
          σ = σ + Φ(x_i, y_i) − Φ(x_i, y_i')
        end if
      end if
    end for
  end for
输出：平均权重参数向量 γ − σ/(mT)
```

3.6 多语言问题：什么是词元[3]

3.6.1 词元切分、实例和编码

到目前为止，我们假定在一个文法体系中或者在一个树库中，词的概念，或者更详细地说一个单词词元的概念，是良定义的。然而，这种定义一般是在给定的树库或分析器中是良定义的，但是对于不同的树库或分析器就会有很多变化。例如，英语中的所有格和系动词 's（be 的一个变体）。在英语中，词元之间一般由空格隔开。然而，在英语的分析器或树库中，诸如 *today's* 或 *There's* 被视为两个独立的词元，即 *today* 和 *'s* 或 *There* 和 *'s*。正如在宾州树库的标记集标准中指出的，所有格可以适用于一些前面的组成成分而不仅是前面的词元：

```
(NP (NP (NP First)
        (PP of
            (NP America))
        's)
    operating results)
```

87 / 88

[3] 本节讨论与句法分析相关的形态和分词问题。对形态处理的深入论述参见第 1 章。

类似地，对于系动词 's：

```
(S (NP-SBJ (EX There))
   (VP (VBZ 's)
       (NP-PRD (NP (NN nothing))
               (ADJP (RB very)
                     (JJ hot))))))
```

在一些语言中也会有大写和小写的问题，把整个树库的数据变为小写，并对分析器仅输入小写文本，这确实是吸引人的。然而，大小写（case）可以携带有用的信息，如果词 *Boeing* 在树库中未出现过，那么训练数据看起来像是个进行时动词如 *singing*；但是初始的大写字母使其更像一个合适的名词。然而，依赖于类型和可见的训练数据，一些大小写变换，比如选择性地把句子的第一个词变为小写，需要进行以便从树库中得到合理的估计。低频词元可以被保留大小写信息的模式取代，比如单词 *Patagonia* 在树库中出现了两次，那么它可以被 *Xxx* 替代以表示匹配同样模式的。新的未登录词可以被当作这种模式下的已知词。这种技巧同样适用于日期、时间、IP 地址、URL 等。

有些语言文本并没有用 ASCII 编码，因此不同的编码也要考虑，特别地，分析器使用的数据要被编码为与树库一致的编码格式，反之亦然。对于句子标点如（.,），在有些语言中编码为 ASCII，而在某些语言中编码为 UTF-8 格式。有些语言，比如汉语，在不同的地区可能会有不同的编码格式，如 GB、BIG5 和 UTF-8 格式等都可以在中文文本中见到。

从算法角度讲，与编写分析器相比，这些是琐碎的问题，但是实际上这些问题是具有挑战性和耗时费力的。虽然更具体地讨论这些问题不在本章的范围之内，然而应该指出，在具体编写一门新语言的句法分析器或考虑编写句法分析器时诸如分词、大小写、编码等问题都需要考虑。

3.6.2　分词

在很多语言的书写格式中，包括中文，缺少识别词的标记。给定中文文本：北京大学生比赛，一个合理的分词应当是“北京（Beijing）/大学生（university students）/比赛（competition）‘competition among university students in Beijing’”。然而，如果把北京大学（Beijing University）看成一个词，分成“北京大学（Beijing University）/生（give birth to）/比赛（competition），‘Bejing University give birth to competition’”，这种分词是不合理的。

89

分词是对字符序列进行分块处理的过程，其输出结果由分开的有意义的词元组成。仅当我们识别并赋予句子中每个单词的词性（例如，NNP 或者 DT）时，整个句子的句法树才可以构建。在处理英语或法语的系统中，词元是可以直接利用的，因为在这些语言中单词之间有空格隔开，而对于中文，字符紧挨着书写，对于词的识别没有标记。

对于中文的分词已有众多的研究者并组成了一个大的社区，而且举办了三次 SIGHAN 评测［34，35，36］。本书第 1 章讨论了这些问题，本节仅关注对于句法分析的影响。

对于中文分析［37］一个有意思的方法就是直接分析汉字序列本身。分析器指定词边界（作为分析过程的一部分），树中包含一组汉字跨度的非终结符也可以认为是明确了词边界。然而，研究发现，最近的上下文在预测词边界上非常有用。全局句子上下文在词边界的发现问题上的作用不大，尽管在某些情形下分词的歧义消解过程需要获得分析树的长距离依赖关系。

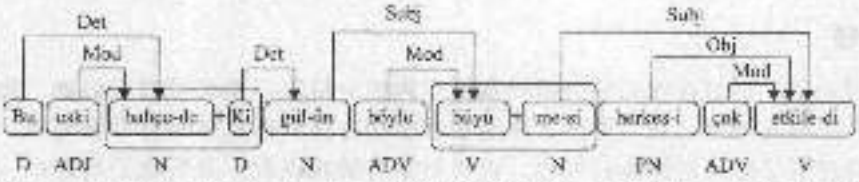
分词模型如只输出一个最优分词结果，则句法分析器不能对多种合理的分词结果进行选择。应用 Bar-Hillel、Perles 和 Shamir［38］的结果，我们知道基于 CFG 的分析器可以

分析输入网络（表示为有限自动机的有限语言）。分析器把自动机的状态作为索引，可看作是输入字符串索引的一种泛化。输入的词格可以用来表示中文输入分词的多个结果，然后分析器选择其中那一种结果可导致最准确的分析树。

3.6.3 形态学

在很多语言中，用空格分割词元是有问题的，因为每个单词包含一些称为词素的成分，以至于单词的意思可以看做是词素意思的组合。此时，单词被分解成一个词干和若干词素。

例如，下面土耳其树库中的一个依存分析例子显示出句法依存应该了解单词内的词素。在本例中，单词内的词素边界用"+"号表示。词素，不是单词，被用作为中心部分以及依存部分。



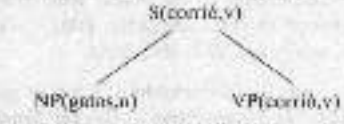
[93]

土耳其语、芬兰语以及其他的一些黏着语都有这样共同的性质：被个子句和词素结合形成非常复杂的单词。

诸如捷克语、俄语等屈折语言虽然没有那么极端，但是很多不同的词素也被用来标记文法格、性等，并且每一种词素都和其他词素是正交的（因此它们可独立地同现）。例如，Hajic 和 Hladka [38] 提到捷克语的大多数形容词可以潜在地形成共 4 种性、共 7 种格、共 3 种比较程度，以及附加两种极性。仅对于形容词，这样就导致了 168[2] 种不同的屈折语词。除了具有大量的词形，对每个屈折变化的单词的词素切分也有歧义。除了句法歧义外，分析器还要处理形态歧义问题。

为了处理形态歧义问题，把单词切分成最可能的词素序列可被简化成词性标注任务（非常复杂）。每个单词都被标注一个编码了不同词素的复杂标签。例如，词性标签 V M 3 表示每个单词都由可以在 10 个不同维度进行屈折变化的词素组成，其中词干是 V（动词），M 表示阳性，3 表示第三人称，其他类型的词素需要赋值，本例表明它们没有在这个分析中出现。词性标注器需要产生这个复杂的标签，典型做法是对词性的每个部分训练不同的子分类器，然后合并各子分类器的输出而得到整体的词性标签 [39]。单词本身并没有被切分成词素，但是每个词用一个编码了很多关于词素的信息的标签标注。这种增强的词性标签集合可以作为统计分析器丰富的特征来源用于屈折语。

在统计分析器里面增加词素建模，可以提高分析精度。例如，在西班牙语 [40] 中，如果我们想创建下面的分析树片段，分析器具有词素分析的能力，那么复数名词 *gatos*(cats) 就不可能修饰单数动词 *corrió*(ran)，即使在训练数据中没有这个特别的双词依存关系。



在统计分析器（特别是依存分析）的判别模型中，加入词素分析信息是非常直接的，因为判别模型允许加入大量的重叠特征，单词的词素信息可以融入进这些混合特征以构建

[2] 原文为 555，疑误。——译者注

更好的句法分析器。正如 CoNLL2007 共享任务 [13] 以及有些详细描述了判别式依存分析（例如 [41，42]）中每种语言有用的特征的论文所表明的，词素信息可以帮助统计句法分析器提高精度，尤其是对于形态复杂的语言。

在短语结构分析中，Cowan 和 Collins [40] 提出了一种判别式模型用以分析西班牙语，采用生成式模型的 k-best 输出结果，并且采用形态特征对输出结果进行重排序。不同的形态信息被用在词性标注和重排序模型中，并且已证明只要标签集合不是过于庞大，形态信息的增加就可以提高句法分析的精度。Sarkar 和 Han [43] 在分析朝鲜语的生成模型中加入了词素信息。依存概率由完整词形和词的各种形态分解形式插值而得。研究表明，在这个特别的模型中，使用词干信息而不是后缀可以帮助分析器泛化形态复杂的词形，并帮助提高句法分析的精度。

3.7　总结

本章讨论了自然语言的句法分析以及如何构建分析器，使其可以有效地、精确地分析自然语言，生成句法树。我们论述了使用数据驱动方法分析自然语言的必要性，介绍了为分析语言提供训练数据的树库的概念。以机器学习的观点看，分析也是有趣的，因为它是复杂的、结构化预测的任务，分析的输出标签不是简单的分类标签而可分解为更小的单元，结构化输出标签的数目随着输入的大小呈指数幂增长。我们讨论了短语结构分析和依存分析的使用，这是两种以不同的方式表示自然语言的句法分析的方法。本章包括可以有效分析输入句子的分析算法，以及在分析方面进行歧义消解的机器学习模型。针对不同语言编写分析器存在很多问题，我们讨论了一些在分析与英语有很大不同的语言时出现的问题：比如词元切分、大小写、编码问题、分词以及形态学。对于每种情况，我们探讨了这些问题的解决方案如何融入这些语言的统计分析器。

致谢

感谢爱丁堡大学信息学院在我离开西蒙·弗雷泽大学休假的一年给予的热情招待。本章的大多数章节都是在爱丁堡完成。

参考文献

[1] K. Knight and D. Marcu, "Summarization beyond sentence extraction: A probabilistic approach to sentence compression," *Artificial Intelligence*, vol. 139, no. 1, pp. 91–107, 2002.

[2] C. Callison-Burch, "Syntactic constraints on paraphrases extracted from parallel corpora," in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pp. 196–205, 2008.

[3] M. Galley, M. Hopkins, K. Knight, and D. Marcu, "What's in a translation rule?," in *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2004): Main Proceedings* (D. M. Susan Dumais and S. Roukos, eds.), pp. 273–280, 2004.

[4] S. Miller, H. Fox, L. Ramshaw, and R. Weischedel, "A novel use of statistical parsing to extract information from text," in *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference (NAACL 2000)*, pp. 226–233, 2000.

[5] R. Barzilay and K. R. McKeown, "Sentence fusion for multidocument news summarization," *Computational Linguistics*, vol. 31, no. 3, 2005.

[6] R. Barzilay, "Probabilistic approaches for modeling text structure and their application to text-to-text generation," in *Empirical Methods in Natural Language Generation* (E. Krahmer and M. Theune, eds.), Lecture Notes in Computer Science (LNAI 5790), Berlin: Springer, 2010.

[7] M. Lease, E. Charniak, and M. Johnson, "SS-11.4: Parsing and its applications for conversational speech," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05) 5: V-961–V964*, 2005.

[8] P. Pantel and D. Lin, "Concept discovery from text," in *Proceedings of Conference on Computational Linguistics (COLING-02)*, pp. 577–583, 2002.

[9] A. Rudnicky, C. Bennett, A. Black, A. Chotimongkol, K. Lenzo, A. Oh, and R. Singh, "Task and domain specific modeling in the Carnegie Mellon communicator system," in *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP)*, 2000. Paper G4-01.

[10] M. Sipser, *Introduction to the Theory of Computation*, 2nd ed., Boston: PWS Publishing Co., 2005.

[11] J. Nivre, "Two notions of parsing," in *A Finnish Computer Linguist: Kimmo Koskenniemi. Festschrift on the 60th Birthday* (A. Arppe, L. Carlson, O. Heinämäki, K. Lindén, M. Miestamo, J. Piitulainen, J. Tupakka, H. Westerlund, and A. Yli-Jyrä, eds.), pp. 111–120, Stanford, CA: CSLI Publications, 2005.

[12] L. Tesnière, *Éléments de syntaxe structurale*, Paris: C. Klincksieck, 1959.

[13] J. Nivre, J. Hall, S. Kübler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret, eds., *Proceedings of the CoNLL Shared Task Session of Empirical Methods on Natural Language Processing-Conference on Natural Language Learning 2007*, 2007.

[14] H. Gaifman, "Dependency systems and phrase structure systems," Tech. Rep. P-2315, The RAND Corporation, Santa Monica, CA, May 1961.

[15] S. Abney, "Dependency grammars and context-free grammars," manuscript presented at meeting of Linguistic Society of America, Jan. 1995.

[16] M. Marcus, G. Kim, M. A. Marcinkiewicz, R. MacIntyre, A. Bies, M. Ferguson, K. Katz, and B. Schasberger, "The Penn Treebank: Annotating predicate argument structure," in *Proceedings of the ARPA Human Language Technology Workshop*, pp. 114–119, 1994.

[17] M. Johnson, "A simple pattern-matching algorithm for recovering empty nodes and their antecedents," in *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pp. 136–143, 2002.

[18] L. Rimell, S. Clark, and M. Steedman, "Unbounded dependency recovery for parser evaluation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'09)*, pp. 813–821, 2009.

[19] R. Levy and C. D. Manning, "Is it harder to parse Chinese, or the Chinese treebank?," in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pp. 439–446, 2003.

[20] J. Nivre, "Algorithms for deterministic incremental dependency parsing," *Computational Linguistics*, vol. 34, no. 4, pp. 513–553, 2008.

[21] J. Goodman, "Global thresholding and multiple-pass parsing," in *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing*, 1997.

[22] P. F. Felzenszwalb and D. McAllester, "The generalized A* architecture," *Journal of Artificial Intelligence Research*, vol. 29, pp. 153–190, 2007.

[23] J. Eisner, "Three new probabilistic models for dependency parsing: An exploration," in *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*, pp. 340–345, August 1996.

[24] R. McDonald, "Discriminative training and spanning tree algorithms for dependency parsing," PhD thesis, University of Pennsylvania, July 2006.

[25] M. Johnson, "PCFG models of linguistic tree representations," *Computational Linguistics*, vol. 24, no. 4, 1998.

93

[26] S. Petrov, L. Barrett, R. Thibaux, and D. Klein, "Learning accurate, compact, and interpretable tree annotation," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 433–440, 2006.

[27] M. Collins, "Three generative, lexicalised models for statistical parsing," in *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pp. 16–23, 1997.

[28] J. Goodman, "Parsing algorithms and metrics," in *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pp. 177–183, 1996.

[29] M. Collins, "Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms," in *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1–8, 2002.

[30] Y. Freund and R. E. Schapire, "Large margin classification using the perceptron algorithm," *Machine Learning*, vol. 37, no. 3, pp. 277–296, 1999.

[31] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the 18th International Conference on Machine Learning (ICML)*, pp. 282–289, 2001.

94

[32] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958.

[33] M. Collins, "Ranking algorithms for named entity extraction: Boosting and the voted perceptron," in *Proceedings of Association for Computational Linguistics 2002*, pp. 489–496, 2002.

[34] R. Sproat and T. Emerson, "The 1st international Chinese word segmentation bakeoff," in *Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing*, pp. 123–133, 2003.

[35] T. Emerson, "The 2nd international Chinese word segmentation bakeoff," in *Proceedings of the 4th SIGHAN Workshop on Chinese Language Processing*, pp. 123–133, 2005.

[36] G.-A. Levow, "The 3rd international Chinese language processing bakeoff," in *Proceedings of the 5th SIGHAN Workshop on Chinese Language Processing*, pp. 108–117, 2006.

[37] X. Luo, "A maximum entropy chinese character-based parser," in *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, Volume 10*, pp. 192–199, 2003.

[38] Y. Bar-Hillel, M. Perles, and E. Shamir, "On formal properties of simple phrase structure grammars," in *Language and Information: Selected Essays on Their Theory and Application* (Y. Bar-Hillel, ed.), ch. 9, pp. 116–150, Reading, MA: Addison-Wesley, 1964.

[39] J. Hajic and B. Hladka, "Tagging inflective languages: Prediction of morphological categories for a rich structured tagset," in *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pp. 483–490, August 1998.

[40] B. Cowan and M. Collins, "Morphology and reranking for the statistical parsing of spanish," in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 795–802, 2005.

[41] J. Nivre, J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov, and E. Marsi, "Maltparser: A language-independent system for data-driven dependency parsing," *Natural Language Engineering*, vol. 13, no. 2, pp. 95–135, 2007.

[42] G. Eryigit, J. Nivre, and K. Oflazer, "The incremental use of morphological information and lexicalization in data-driven dependency parsing," in *Proceedings of the 21st International Conference on the Computer Processing of Oriental Languages*, pp. 498–507, 2006.

95
|
96

[43] A. Sarkar and C. hye Han, "Statistical morphological tagging and parsing of Korean with an LTAG grammar," in *Proceedings of the Sixth Workshop on Tree Adjoining Grammars and Related Formalisms: TAG+6*, 2002.

第4章

Multilingual Natural Language Processing Applications: From Theory to Practice

语义分析

Sameer Pradhan

按字典定义，语义学（semantics）指关于意义的研究，而分析则指对某事物的细致检查，即识别出待分析的信息片断并将它们联系起来。将这两个概念放在一起，就是语义分析（semantic parsing）。按最宽泛的方式来解释，所谓语义分析就是指在信息信号中识别出意义块（meaning chunk）并尝试将其转换为某种数据结构的过程。利用该数据结构，计算机将可以执行更高层的任务。本书所考虑的信息信号是人类语言文本。不幸的是，在自然语言处理学界，语义分析这一术语有时是有歧义的。多年来，研究者们曾将此术语用于表示各种不同层次和粒度的意义表达方式。由于语义学是这样一个模糊的术语，它曾被用于代表各种不同深度的意义表示，从实体间的领域相关关系识别这种基础问题到事件中各实体、角色识别这种中层任务，甚至到将文本转换为一系列特殊的逻辑表达式等。本章我们将该术语的意思限定为研究如何把自然文本映射成某种计算机可处理的意义表示，利用这些表示，计算机将可进一步达成某些目标，例如，信息检索、回答问题、填充数据库或执行操作等。

4.1 概述

语言理解研究的最高境界是尽可能详细地识别出意义的表示，以便使推理系统据此能完成推演；同时该表示又要足够通用以便能在无须（或仅用少量）自适应的情况下用于跨多领域的应用。是否能为各种以某种方法使用语言接口的应用建立一种最终的、低层的、细致的语义表示现在尚不明了，或者说，目前还不清楚是否能创建出一种能包含上述应用所使用的意义的各种粒度与侧面的本体（ontology）——迄今一个也没有被创建出来。因此，在自然语言处理学界的语言理解社区中就出现了两条折衷的途径。

第一条途径是，针对诸如航空订票、足球游戏仿真、地理数据库查询等的受限领域应用创建专门但丰富的语义表达，然后构造系统以使其将文本转换为这种丰富但受限领域的意义表达。第二条途径是，建立一套中间意义表达方式（从低层到中层分析），然后把理解任务分解成多个小的、更可控的子任务，比如，先做词义消歧，再做谓词-论元结构识别等。一旦将问题按这种方式分解，每个中间表示将只负责获取整体语义中相对较小的部分，因而对它们进行定义和建模都会变得更容易些。和第一条途径不同的是，第二种方法中的每个语义表达不会与特殊领域绑定（尽管只覆盖整体语义的某个小部分）。因此，依此所创建的数据和方法可适用于通用目的。

不幸的是，我们还未得到详细且全局的语义表示形式的圣杯（holy grail），即该表示既能很容易学习，也具有很高的跨领域覆盖率。所以，本章接受两类意义表达的并存：一个是领域相关的、深层表达；另一个是一组相对浅层的但通用、低中层的表达。能产生出第一类输出的任务通常称为深层语义分析（deep semantic parsing），而生成第二类输出的任务则通常称为浅层语义分析（shallow semantic parsing）。生成这两类输出的算法本章都会进行讨论。

上述两条途径各有很多问题，前者由于面向专用领域，每移植到新领域就需要对原有

表示进行修改甚至从头开始。换句话说，该表示方法在跨领域时的重用性是非常有限的。后者的问题是，很难构建出一种通用目的的本体，并创建一种浅显而容易学习但又详细到适用于所有可能应用的符号系统。因此，我们必须构造一个在通用表示和专用表示之间的特定应用翻译层。当然，和将专用表示迁移到新领域所做的自适应相比，这种翻译组件相对还是小的。这些工作也都没开始考虑跨语言使用这类系统的问题、不同语言结构对这些意义表达所起的作用以及它们的可学习性等。基于这些原因，在语言处理历史上，相关研究社区已总体上从更细节、深度、领域相关的表示转移到更浅层的表示了。

4.2　语义解释

语义分析可以看成是一个更大的过程——**语义解释**（semantic interpretation）中的一部分。该过程整体而言是让我们可以定义出文本的意义表示，该表示将进一步提供给计算机以便使它执行语言理解系统或应用所需的进一步的计算处理与搜索。语义理解过程涉及几个不同的部分。下面几个小节将讨论其中的主要部分。

我们的讨论将从 Chomsky 影响深远的著作《Syntactic Structures》[1] 开始。该著作
| 98 | 引入了转换短语结构文法（transformational phrase structure grammar）的概念，该概念给出了一种人类的自然语言组合形式的可操作定义。1957 年 Chomsky 的书出版后不久，Katz 和 Fodor [2] 就发表了在生成文法范型内的首个语义相关工作。他们发现，Chomsky 的转换文法并不是语言的完整描述，因为它并没有考虑语义问题。在他们 1963 年发表的文章 "The Structure of a Semantic Theory" 中，Katz 和 Fodor 提出他们所构想的一个语义理论应拥有的属性。一个语义理论应该能：

1）解释带有歧义的句子。例如，该理论应能识别出句子 "The bill is large" 中 bill 单词的歧义（可能代表钱或鸟嘴）。

2）在上下文中消解词语歧义。例如，如果同一个句子扩展到 "The bill is large but need not be paid" 形式，则该理论应该能消解出 bill 单词与金融有关的词义。

3）识别出符合语法但无意义的句子，比如 Chomsky 给出的著名例子：Colorless green ideas sleep furiously。

4）识别出与语法或转换无关的释义复述（它们具有相同的语义内容）。

下面几个小节将探讨为获取语义表示需处理的问题。

4.2.1　结构歧义

当我们讨论结构时，通常指的是句子的语法结构。这是一个句子级的现象，本质含义是将句子转换为其内含的句法表示。由于句法和语义有太多的强互动，因此多数语义解释的理论都使用深层的句法表示。通常而言，句法已成为语义解释的第一步，后面还有不少阶段（关于句法处理的相关信息请参看第 3 章）。

4.2.2　词义

任意给定一种语言，几乎可以肯定存在同一词形（可能具有不同的形态变换）在不同上下文中用于表示不同实体或概念的情形。例如，单词 nail 既可以表示人体解剖学的某部分（即指甲），也可以表示固定其他物体的金属物（即钉子）。人们擅长于通过上下文识别出作者或说话者在使用该词时的实际意图。请看如下 4 个例子，诸如句子①和③中 hammer、hardware store 等单词以及句子②和④中的 clipped 和 manicure 等单词都让人们

可以很容易消解出 nail 的实际意义。

①He *nailed* the loose arm of the chair with a hammer.

②He bought a box of *nails* from the hardware store.

③He went to the beauty salon to get his *nails* clipped.

④He went to get a manicure. His *nails* had grown very long.

因此，在语篇中消解词义已构成了语义解释过程中的一个步骤，我们将在 4.4 节中更深入地对该问题进行讨论。

4.2.3 实体与事件消解

任何语篇都包含了在一段时间内发生的一系列显式或隐式的事件以及参与其中的一组实体。语义解释的下一个重要组件是识别出散布在语篇中的各种实体，这些实体可能使用相同或不同的短语来表示。消解出语篇中所涉及的实体或事件类型，同时对语篇中同一实体的不同的表达方式进行消歧，对于创建语义表示十分关键。多年来，两个主流任务变得越来越流行，即命名实体识别（named entity recognition）和共指消解（coreference resolution），这两个任务一般归入信息抽取（information extraction）问题，将在第 8 章中详细讨论。

4.2.4 谓词-论元结构

一旦完成了词义消歧、实体和事件识别，就轮到其他层的语义结构处理登场了，即如何识别出事件与实体间的关联。确定句子中各谓词的论元结构相当于识别出哪个实体扮演哪个事件的什么部分。一般而言，这一过程可定义为识别出谁（Who）在什么时间（When）、什么地点（Where）对谁（Whom）做了什么事（What），以及如何做的（How）。

图 4-1 显示了 say 和 acquire 两个事件的各参与者。

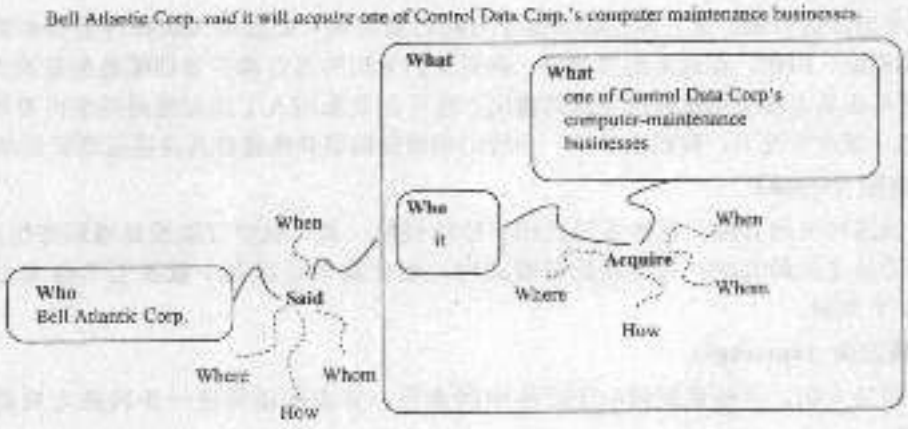


图 4-1 关于“谁（Who）在什么时间（When）、什么地点（Where）对谁（Whom）做了什么事（What），以及如何做的（How）”的一个语义表示

4.2.5 意义表示

语义解释的最后一个步骤是建立可供各种应用算法进一步处理的语义表示或意义表示。这一过程有时也称为深度表示（deep representation）。很不幸，正如我们早前提到过

的，由于目前还没有适用于任意应用的通用且深度足够的表示，本领域中的多数研究都是与应用相关或与领域相关的特定应用。下面两个例子各给出了一个例句及其意义表示，两个表示分别适用于机器人世界杯（RoboCup）和GeoQuery两个领域（具体描述见4.6.1节）：

1) If our player 2 has the ball, then position our player 5 in the midfield.

```
((bowner (player our 2)) (do (player our 5) (pos (midfield))))
```

2) Which river is the longest?

```
answer(x1, longest(x1 river(x1)))
```

这是领域相关的方法，本章的余下部分将更多关注领域无关的方法。

4.3　系统范式

本章所讨论的是计算语言学和语言学界所熟知的问题。这些研究者已经在不同粒度和通用性层次上考察了意义表示以及其分析方法，也涉足了大量语言。很多时候并没有可用的手工标注数据。因此，对处理语义解释问题中涉及的各种主要维度进行观察是很重要的。本章不可能覆盖所有这些维度，尽管我们会提到许多历史方法，但我们还是尽量关注在实际应用中流行并成功的方法。这些方法总体上可以按如下3种方式进行分类。

1. 系统架构（system architecture）

1）基于知识库的方法：正如其名所暗示的，这类系统使用预先定义好的规则集或知识库来解决新问题。

2）无监督的方法：这类系统倾向于通过利用现存的资源来减少人工干预。这些资源为特殊应用或问题领域而举箱（bootstrapping）出来。

3）有监督的方法：这类系统涉及由人工针对规模足够的数据中所出现的现象进行标注，以便可应用机器学习算法。采用这种方法，研究者通常会创建特征函数并将每个问题实例都投射到特征空间中。这类方法所训练出的模型将进一步利用这些特征来预测标注，
101 并应用于未见数据。

4）半监督的方法：手工标注通常是十分耗时费力的，而且也无法获得足够的数据以囊括所有的现象。因而，在这类事实例中，研究者们采用的是自动扩展训练数据集的方式。扩展的途径可能是直接应用机器所生成的输出，也可能是采用人工校对数据的输出对现有模型进行拓展。很多情况下，我们会采用一个特定领域的模型并快速将其自适应到新领域。

2. 范围（scope）

1）领域相关的方法：这些系统适用于特定领域，如，航空订票或足球训练仿真等。

2）领域无关的方法：这类系统足够通用，相关技术可以在少量甚至不修改的前提下适用于多个领域。

3. 覆盖面（coverage）

1）浅层方法：这些系统倾向于产生中间表示，该表示还需进一步转换为机器操作所需的结构。

2）深层方法：这些系统通常创建机器或应用可直接使用的最终表示。

4.4　词义

在组合语义学框架中，整体意义可由各部分的意义组合而成。在文本语篇中，人们所考虑的最小部分就是词本身了——可能是文本中直接出现的词元或其还原后的原形。词义的研究已有很长的时间[3, 4, 5]，但研究者们仍然尚未达其本质。语言中每个词在不同

上下文里的不同出现是否能确定出有限多个不同的含义集？这一问题目前尚不明了，即使这确实是能做到的，但上下文中的给定词究竟归属于单个含义或是对某几个含义（依据不同的分布，为所有含义的子集）都有关联也还是不清楚。

有许多尝试解决此问题的方案，包括：基于规则和基于知识库的方法、完全无监督的学习方法、有监督的学习方法以及半监督的学习方法等。早期的主流系统是基于规则和基于知识库的方法，使用的是依据词典定义的词义。无监督的词义推导或消歧技术则尝试依据词语在各种语料库中的出现推导出其词义。这些系统执行聚类时可能使用软聚类也可能使用硬聚类，并倾向于根据特定应用的需要调节这些聚类。目前多数的有监督词义消歧方法则不同，其主要假设是，在预先定义好的粒度层次（通常是与应用程序相关的）中，特定上下文里的一个词只能表达一个特殊的词义（尽管有监督方法的输出结果也还可以进一步处理以生成多个候选词义的排序或分布）。有监督的词义消歧方法需要人工标注的数据，此时，对词义的精细化区分和让多个标注者在给定词义集时能有较高的标注一致性，这两者往往有一种微妙的平衡。词义粒度越粗，标注者之间就越容易通过学习而达成一致。然而，对于应用程序而言，这种低粒度的词义很可能不能很好地标识出词的微妙差别。标注－学习循环中观察到成功并不能直接说明该意义表示的深度已符合应用程序的需要。对这一问题 Palmer、Dang 和 Fellbaum 等人[5]曾进行了详尽的讨论。 102

尽管词义消歧在理论上被假想为语言理解的重要方面，但其适用性似乎还是一个备受争议的问题。构造大规模手工标注词义的语料库本身十分困难，词义消歧系统在各种应用中适用性的复杂现状及模棱两可的状态部分地导致了只有很少的计算资源被产生，以支持创建更好的自动系统。而且，标准的缺乏也使得包含词义信息的各种资源的合并无法运行。实际上，有些尝试正是希望在这些资源间建立映射。

根据 Resnik 和 Yarowsky 的观察[7]，存在这种矛盾心理的主要原因之一是，在诸如信息检索和语音识别等在内的许多更为成熟的语言处理应用中，词义消歧技术要么显得多余要么有更廉价且更好的替代品。在信息检索领域里，广泛接受的事实是，查询中的多个词匹配文档上下文中的多个词，以提供包含相当好的词义信息的隐消歧过程，常规的消歧技术往往很难超越其效果[8]。在语音识别领域中，上下文类[9,10]常常被证明比词语类[11]更为适用。特定领域或文本体裁往往倾向于映射给定词词义的较小子集，甚至只映射给定词的其中一个词义。因此，鉴于有些语义分析系统是特定领域的，而有些则是领域无关的，后者较前者而言更需要词义消歧。此外，特定领域应用中一个词通常会映射到单一概念，而找到这种映射会相对简单的问题，这也进一步削弱了词义消歧的必要性。Resnik 和 Yarowsky[7]指出词义消歧缺乏进展的几个原因：缺乏标准的评测；相比其他任务，本任务需要更大数量的资源以提供所需的知识；以及很难获得足够大的词义标注数据集。受该研究驱动，SIGLEX（Special Interest Group on LEXicon）举行了多次评测，SENSEVAL 1、2 和 3 以及 SEMEVAL 1 和 2。这些比赛在生成标准数据集和评价标准方面非常成功，同时也确定了推进对词义消歧及其应用的理解的相关任务。

如何测量自动词义消歧系统的性能是一个重要问题。Gale、Church 和 Yarowsky[12]对此问题进行了详细的讨论。他们的建议是，词义消歧系统的性能下界应该是将词语的每个实例都对应为其在足够大语料库中最频繁出现的词义。该建议目前仍然被普遍遵循。这也就是通常所谓的最频繁词义（Most Frequent Sense，MFS）基线系统。词义标注的标准答案（gold-standard）语料应具备的一个好的属性是，它应在一定程度上是可复现的（replicable），换句话说，多个标注者对同一个语料的标注应有足够高的一致性。北方说， 103

该一致性是 $x\%$，则我们通常会将 $x\%$ 看作自动系统的性能上限。

词汇歧义有 3 种主要类型：1）同形异义（homonymy）；2）多义（polysemy）；3）兼类歧义（categorial ambiguity）[13]。同形异义表示一个拼写相同的词有不同的含义。同形异义的每个词义又可能包含更细的词义差别，需要依据上下文才能确定，这种现象称为多义。例如，bank 的如下两个词义是完全不相关的：financial bank（即银行）和 river bank（即岸）。进一步说，bank 的词义还有一些更细的相关词义——表明事物的集合，因而，financial bank（即银行）和 bank of clouds（云层），两个 bank 就构成了多义。我们可用一个例子来解释兼类歧义。book 可以表示书本，也可以表示立案，前者的语法范畴是名词，后者的语法范畴则是动词。区分这两类有助于上述两个词义的消解。因此，兼类歧义可仅利用句法（词性）信息消解，多义和同形异义则需要更多句法之上的信息。

按传统做法，英语词义标注是针对每个词性单独做的，而中文词义标注则针对词形，因而可能跨词类，部分原因是中文名词和动词间的区别更加隐晦。

4.4.1　资源

和任何语言理解任务一样，资源的可获得性对词义消歧也是十分关键的。不幸的是，至少直到最近，还没能看到词义消歧社区开发出大规模的手工标注词义的数据。词义消歧的早期工作用机器可读的字典或辞典作为知识源。两个主流的来源是朗文当代英语词典（Longman Dictionary of Contemporary English，LDOCE）[14] 和罗氏义类词典（Roget's Thesaurus）[15]。20 世纪 80 年代后期诞生的重要字典资源——WordNet [16] 一直也非常有影响力。WordNet 不仅是一个包含了大部分英语单词在多个词类上的词义的词汇资源，还包含了一个丰富的义类系统（taxonomy），该系统用许多不同的关系将各单词联系起来。这些关系有：上下位关系（hyperonymy）、同形异义关系、整体部分关系（meronymy）等。此外，为方便自动词义消歧研究，WordNet 还提供了一个标注了 WordNet 词义的语义索引（SEMCOR）语料 [18]，该语料文本为 Brown 语料 [17] 中的一小部分。最近，WordNet 又做了一些扩展，即注释部分添加了句法信息、手动和自动消歧方法以及生成逻辑形式，以使更好地应用在诸如问答的应用中 [19]。另一个语料是 DSO 英语词义标注语料（DSO Corpus of Sense-Tagged English），该语料是通过将 Brown 语料和华尔街日报（Wall Street Journal，WSJ）语料中最常见且有歧义的英文单词（包括 121 个名词和 70 个动词）标注为 WordNet 1.5 词义而得 [20]。另外，过去十年来举行的 SENSEVAL [21] 评测也创建了很多用于测试词义和相关问题系统的语料。到目前为止花费力气最大的是通过语言数据联盟（Linguistic Data Consortium，LDC）发布的 OntoNotes 语料 [22, 23, 24]。其中标注了大量的动词（大约 2700 个）和名词（大约 2200 个），覆盖大约 85% 的粗粒度词义。该集合是多文体的而且具有非常高的跨标注者一致性。SEMEVAL 2007 上 Pradhan 等人 [25] 组织的词汇抽样任务就使用该语料。Cyc [26] 是另一个有用资源的例子。它创建了一个与世界中对象和事件相关的常识知识的形式化表示，意图克服在词义消歧和许多其他自然语言任务中至关重要的所谓知识瓶颈。该知识库尽管经过几十年的手工构建，目前还有许多有待改进之处，这也突显了这一努力的难度。

[104]

英语似乎有最发达的字典，包含了词的各种语义特征以及由词语组合并形成的连贯语义类。目前大家也正在努力创建其他语言的资源。例如，知网 [27] 就是一种类似 WordNet 的中文词语资源。WordNet 全球协会（Global WordNet Association，http://www.globalwordnet.org）致力于跟踪 WordNet 的跨越语言开发。研究人员还用半自动的方法不断

扩大现有语言的覆盖面 [28，29，30，31] 或扩展到诸如希腊语的其他语言 [32]。除了词义标注语料库外，有许多类似 WordNet Domains（http://wndomains.fbk.eu/）等的资源。该资源提供了结构化的知识，可以帮助克服词义消歧的知识瓶颈。

4.4.2 系统

讨论完词的歧义问题和某些资源后，我们开始转向对一些词义消歧系统的讨论。正如更早提到过的，对于词义消解问题，研究者已探索了各种系统体系。我们可以将这些系统划分为4类：1）基于规则或知识的系统；2）有监督的系统；3）无监督的系统；4）半监督的系统。

在下面几个小节中，我们将依次介绍上述的每种系统。

1. 基于规则的方法

第一代的词义消歧系统主要基于字典词义定义和注释 [33，34]。这些技术大部分都是手工制作的，使用的资源如今已不一定可用。此外，可访问的确切规则和系统也非常有限，大部分信息只能从存档的出版物和讨论中获得，也只是那些实验过程中的某些具体词和词义。总之，大部分信息只具有历史意义，目前已无法轻易用于转换以及构建系统。然而，我们还是可以获得一些有价值的技术和算法（本节中将讨论它们）。也许最简单并且最古老的基于字典的词义消歧算法是由 Lesk 提出的 [35]。第一代的词义消歧算法大多是基于机器可读字典，例子请参看 Calzolari 和 Picchi 的文献 [33]。

第一届 SENSEVAL 评测 [36] 在比较词义消歧性能时使用一个简化的 Lesk 算法作为基准系统。该算法的伪码如算法 4-1 所示。算法的核心思想是：词在给定上下文中的词义最有可能是其字典解释与该上下文重叠最大的那条词典义。该算法在此后还有进一步的修改，以使其更鲁棒，能适用于各种不同的字典条目、上下文及定义。例如，Banerjee 和 Pedersen 的工作 [37] 就是对 Lesk 算法的修订。该算法对上下文和字典定义中的词语考虑了同义词、上位词、下位词以及整体等，以期获得更准确的重叠统计；匹配分值的取值为上下文和注释 [105]
的最大公共子序列[6]长度的平方；使用长度为5个词的上下文窗口（目标词本身以及其左右各两个词）。他们报告改进后的算法在 SENSEVAL-2 词汇范例数据集上的性能较普通 Lesk 算法提升了两倍（从16%提升至32%）。这种性能提升是显著的，毕竟该算法相当简单。

算法 4-1 简化的 Lesk 算法的伪代码函数：computeOverlap 返回两个集合中公共词个数

```
Procedure: SIMPLIFIED_LESK(word, sentence) returns word 的最佳词义
best-sense ← word 的最常见词义
max-overlap ← 0
context ← sentence 中的词语集合
for all sense ∈ word 的所有词义 do
    signature ← sense 的注释和范例中的词语集合
    overlap ← COMPUTEOVERLAP(signature, context)
    if overlap gt max-overlap then
        max-overlap ← overlap
        best-sense ← sense
    end if
end for
return best-sense
```

[6] 同一个注释中包含多个子序列的情况是可能出现的，正如，仅包含代词、介词、连词等非实体词的子序列都不会被考虑。例如，子序列 "of the" 在计算分值时是不会被考虑的。

另一种基于字典的算法是 Yarowsky 提出的 [38]，该研究使用了罗氏义类词典类别并将未见词语分类到这 1042 个类别之一。分类的依据是在大规模语料上对每个类别的每个成员各 100 个词索引句（word concordance）的统计分析。研究中用到的语料库是 1 千万词的 Grolier 百科全书（Grolier's Encyclopedia）。该方法在之前曾做过的一些定量研究的 12 个词集上表现得相当不错。尽管该研究中使用的实例和语料库和先前所报道的都不同，但它仍然体现了相对简单方法的成功。该方法包括三个步骤，如图 4-2 所示。第一步是收集上下文。第二步是为每个显著词计算权重，需要注意的一点是，此处的上下文窗口大小是目标词两侧各 50 个词，这远远大于 Gale 等人 [12] 所发现的对这类宽泛领域词义分类问题有用的上下文窗口大小。$P(w \mid RCat)$ 是单词 w 出现在 Roget 辞典分类 $RCat$ 上下文的概率。最后，在第三步中，测试集里的未见单词就被归入具最大权重的类别。

1. 收集 Roget 辞典每个类别的上下文。
2. 确定上下文中所有显著词的权重。

$$\frac{P(w_i \mid RCat)}{P(w_i)}$$

3. 将这些权重加于该测试语料中每个词的整体类别。

$$\underset{RCat}{\operatorname{argmax}} \sum_{w} \log \frac{P(w_i \mid RCat)P(RCat)}{P(w_i)}$$

图 4-2　将词义消解为 Roget 辞典类别的算法

[106] 最近，Navigli 和 Velardi [33, 40] 提出了一个基于知识的算法。该算法在为歧义词消歧时采用图表示法来表示其上下文中各单词的词义。这就是所谓的结构语义互连（Structural Semantic Interconnection，SSI）算法。它使用包括 WordNet、领域标签 [41] 及所有可能的标注语料在内的多种信息来重构成概念的结构描述，或称语义图。该算法包含两个步骤：初始化步骤和迭代步骤。算法通过不断迭代，试图消除上下文中所有单词的歧义，直到它无法再消歧或所有术语都已成功消歧。此算法的性能非常接近监督学习算法，虽然从技术上说它没有训练阶段，但是在 SENSEVAL-3 的 all-words 任务中它还是超越了最好的无监督算法。图 4-3 显示了术语 bus 的两个词义的语义图。第一个是交通工具（vehicle）义（即公共汽车），第二个则是连接器（connector）义（即总线）。

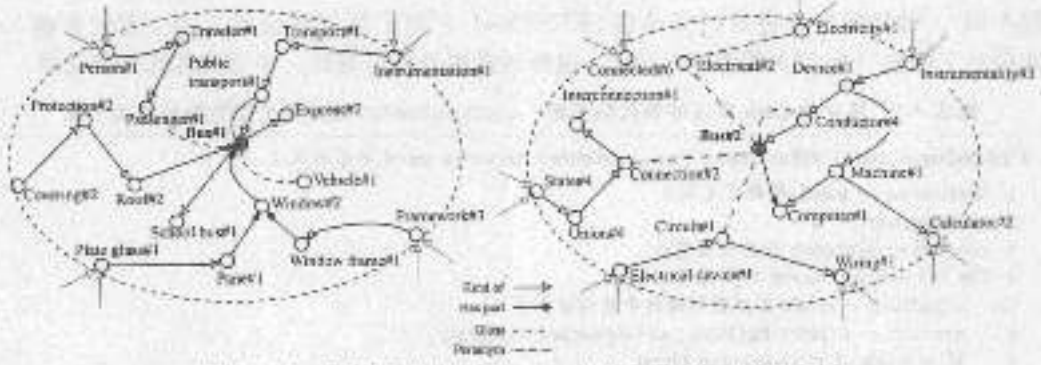


图 4-3　由 SSI 算法生成的名词 bus 的两个词义的语义图

记号：

- T，**词汇上下文**（lexical context），指待消歧术语 t 的上下文中所出现的术语列表，$T=\lceil t_1, t_2, \cdots, t_n \rceil$。
- S_1，S_2，…，S_n 是 t 所有可能概念或词义的结构描述。
- I，**语义上下文**（semantic context）也是一个列表，列表中是 $T\backslash\{t\}$ 集合（不包括 t）里每个术语的相关概念所对应的结构描述。$I=\lceil S_1', S_2', \cdots, S_n' \rceil$，也就是 T 的语

义解释（semantic interpretation）。

- G 指结构描述间各种关系的文法定义，所谓关系也就是图中的语义互连（semantic interconnection）。
- 使用 G 来确定 I 中各结构描述与 S'_1，S'_2，…，S'_n 的匹配程度。
- 选择最匹配的 S'_i。

该算法的工作原理如下。算法维护一个上下文中待消歧术语集合 $P=\{t_i \mid S_i = null\}$，在对 P 中术语进行消歧的每个循环中都使用 I。每次循环结束要么完成 P 中一个术语的消歧并将其从待消歧术语表中删除，要么结束整个算法（这时已没有可供消歧的术语了）。输出 I 将用 t 的词义进行更新。一开始 I 只包含集合 $T\backslash\{t\}$ 中的单义术语以及任何可能的已消歧的同义词集（因为我们使用了词义标注数据）[⊖]。如果 I 是一个空集，则算法的初始猜测将是上下文中具有最少歧义的词汇最有可能的词义。每次迭代，算法都会选取 P 中的一个术语 t，该术语至少有一个词义 S 与 I 中的一个或多个词义有语义互连。函数 $f_I(S,t)$ 用于衡量 S 作为 t 正确解释的可能，其定义如下： [167]

$$f_I(S,t)=\begin{cases}\rho(\{\varphi(S,S') \mid S' \in I\}), & \text{如果 } S \in Senses(t)\\ 0, & \text{否则}\end{cases} \tag{4.1}$$

其中，$Senses(t)$ 指与术语 t 相关联的词义，而

$$\varphi(S,S')=\rho'(\{w(e_1 \cdot e_2 \cdot e_n) \mid S \xrightarrow{e_1} S_1 \xrightarrow{e_2} \cdots \xrightarrow{e_{n-1}} S_{n-1} \xrightarrow{e_n} S'\}) \tag{4.2}$$

即连接每条 S 和 S' 的路径权重（w）的函数（ρ'）。其中，S 和 S' 是语义图，而 e_1 到 e_n 则为连接它们的边集。ρ 和 ρ' 的一种较好的选择是求和或平均和函数。

最后，上下文无关文法 G=(E,N,SG,PG) 编码了所有有意义的语义模式，其中

$$E=\{e_{kind-of},e_{has-kind},e_{part-of},\cdots\}$$

是边的标签集：

$$N=\{S_G,S_e,S_g,S_1,S_2,\cdots,E_1,E_2,\cdots\}$$

是用于编码词义间路径的非终结符集：

$$S_G$$ [168]

是图 G 的开始符号，而

$$P_G=\{S_G \to S_e \mid S_g,S_e \to S_1 \mid S_2 \mid S_3,S_1 \to E_1 S_1 \mid E_1,$$
$$E_1 \to e_{kind-of} \mid e_{part-of},S_g \to e_{glue}S_5 \mid S_4 \mid S_5,\cdots\}$$

则是产生式规则集（据该研究报告大约有 40 条）。

WordNet 中的层次概念信息已在许多方法中被成功应用了。关于几种基于 WordNet 的语义相似度计算的比较请参考 Patwardhan、Banerjee 和 Pedersen [42]。最近出现的诸如维基百科（Wikipedia）的无结构知识库已导致了新一代的算法，这些算法从这类知识库中提取蕴涵的知识以辅助生成覆盖范围更广且多语言的知识库（原先主要依靠类似 WordNet 的资源）并帮助许多任务（如词义消歧等）的最先进模型进一步提高。Strube 和 Ponzetto [43，44] 提供一个名为 WikiRelate! 的算法，该算法使用维基百科的分类层级估算两个概念间的距离。最近，Navigli 和 Ponzetto [45] 则介绍了一种新的自动创建多语种词汇知识库的方法，该方法实现了大规模多语资源维基百科和英语计算词典 WordNet 之间的映射。该知识库目前包括了 6 种语言（德语、西班牙语、加泰罗尼亚语、意大利语、法语和英语）。这些语言与可免费获得的 WordNet 之间的映射也可以很容易通过以英

⊖ 同义词集指具有相同词义的词构成的集合，该术语由 WordNet 的开发者创造 [16]。

语 WordNet 为中间语言的方式加以实现。随着维基百科的持续增长，许多其他语言的资源也可以使用这种方法生成。作为一个起点，Ponzetto 和 Navigli [46] 已经表明，基于 BabelNet 中英语信息所创建的词义消歧系统与以前粗粒度词义消歧任务的诸方法以及特定领域的词义消歧方法都势均力敌。

2. 有监督的方法

具有讽刺意味的是，较简单的词义消歧系统形式——有监督的方法（将复杂性推给机器学习机制并需要手工标注的数据）往往优于无监督的方法，在标注数据上的测试也能取得最好的结果 [21]。此方法的缺点是，词义库必须预先确定，词义库的任何变化都会导致一轮昂贵的重新标注。

这些系统通常包括一个机器学习分类器，该分类器会在给定的手动消歧后的语料中所抽取的词语的各种特征集上进行训练，训练后的分类器则用来对未见的测试集进行消歧。这些系统一个很好的特点是，为了达到最佳的效果（和所有三种方法对比），用户可以在特征中融合规则和知识，也可以半自动地生成训练数据以扩充手动标注的训练集。当然，前者可能有一个特定的知识源或分类器结合问题，这也导致最优特征表示很难获得；后者的问题是半自动方式所生成的词义标记数据都有不同程度的噪声。然而，最先进的系统通常结合了丰富的特征并利用了语言的冗余。

[109]

本节中我们将讨论典型的系统和特征。Brown 等人 [47] 可能是第一个在词义消歧中使用机器学习的，他们的研究使用了平行语料库中的信息。Yarowsky [48] 则是最早在机器学习框架（决策表）中使用丰富特征集解决词义消歧的。其他研究者如 Ng 和 Lee [20, 49] 等，则使用了这些特征，并在各种不同的上下文层次和粒度中完善了它们，涉及的层次和粒度包括句子、段落以及微上下文（microcontext）等。本节我们将探讨一些较流行的方法和相对较容易获得的特征。

分类器（classifier） 最常用的高性能分类器可能是支持向量机和最大熵（MaxEnt）分类器了。基于这两类分类器都有许多高质量且免费获得的系统，可以用于训练词义消歧模型。通常情况下，因为每个词语原形（lemma）有各自的词义清单，但对于每个原形和词性组合（即对于类似英语的语言，各种词性都对应各自的词义清单）都要训练一个单独的模型。

特征集（features） 我们将讨论一个常用的特征子集，其中特征都是词义消歧的有监督学习方法中较有用的。这不是一个穷尽的列表，而只是一些经过时间检验的特征。这些特征提供了一个很好的基础，可用于获得近似最优的性能。

- **词汇上下文**——此特征包含出现在整个段落或较小窗口（通常 3 个词语）内的单词或原形。
- **词性**——此特征包括待标注词义的单词周边窗口内诸词的 POS 信息。
- **上下文词袋**（bag of words context）——此特征是上下文窗口所包含词的无序集合。可以通过调整阈值将较大上下文范围内最具有信息的词语包含进来。
- **局部搭配**（local collocation） 局部搭配是目标词周边短语的有序序列，为目标词消歧提供了语义上下文。一般，目标词两侧各 3 个词左右的窗口中的二元组或三元组会被加入此特征列表。例如，如果目标词是 w，则 $C_{i,j}$ 就是该词的一个搭配，其中 i 和 j 分别指该搭配的起点以及偏移量（均为相对于词 w 的相对值），正数表示目标词右边的词，负号表示目标词左边的词。

下面这组 11 个特征是 Ng 和 Lee[20, 50] 所使用的搭配特征：$C_{-1,-1}$，$C_{1,1}$，$C_{-2,-2}$，

$C_{2,2}$、$C_{-2,-1}$、$C_{-1,1}$、$C_{1,2}$、$C_{-3,-1}$、$C_{-2,1}$、$C_{-1,2}$、$C_{1,3}$。让我们用前面关于对 *nail* 消歧的例子（*He bought a box of nails from the hardware store.*）来对其中的几个加以说明。在这个例子中，搭配 $C_{-1,1}$ 将是单词 *from*，而 $C_{1,3}$ 则是词串 *from_the_hardware* 等。通常情况下，在创建搭配之前停用词和标点是不会被删除的。边界条件则可通过在搭配中添加空词而得到处理。研究人员还可以尝试词根或其他变形，它们也可能有助于更好地泛化上下文形式。Gale 等人[12]讨论了应该以什么样的标准来选择搭配的上下文和数量。 110

- **句法关系**（syntactic relation）——如果可以获得目标词所属句子的分析结构，那么我们就可以使用句法特征。Lee 和 Ng[49] 提出的一组特征如算法 4-2 所示。
- **主题特征**（topic feature）——该词所在文章的广义话题或领域，也是该词最常见词义的一个很好的指示器。

Chen 和 Palmer[51] 最近提出了一些额外的、用于消歧的丰富特征：

- **句子的语态**（voice of the sentence）——此三值特征表明该词所在的句子是被动句、半被动句[⊖]或是主动句。
- **主语或宾语是否出现**（presence of subject/object）——此二值特征表明目标词是否有主语或宾语。给定大量训练数据，我们也可以使用实际的语素和可能的语义角色来代替句法主语或宾语。
- **句子补语**（sentential complement）——此二值特征表明单词是否有句子补语。
- **附属介词短语**（prepositional phrase adjunct）——此特征表明目标词是否有介词短语。如果是，则该介词短语里的名词短语中心词将被选为特征。
- **命名实体**（named entity）——本特征为专有名词和某类通用名词等命名实体。
- WordNet——动词和介词的名词短语论元的中心词的 WordNet 上位同义词集。

算法 4-2 将句法关系选择为特征的规则

```
1: if w 是名词 noun then
2:    选择其父中心词（parent head word），记为h
3:    选择h的词性
4:    选择h的语态（voice）
5:    选择h的位置（左或右）
6: else if w是动词 then
7:    选择w左邻居中以w为其父中心词的最近那个邻居l
8:    选择w右邻居中以w为其父中心词的最近那个邻居r
9:    选择l的词性
10:   选择r的词性
11:   选择w的词性
12:   选择w的语态
13: else if w是形容词 then
14:   选择其父中心词，记为h
15:   选择h的词性
16: end if
```

111

最近，受语义角色标注研究启发，Dligach 和 Palmer[52] 提出了如下用于动词词义消歧的特征：

- **路径**（path）——本特征是从目标动词到其论元的路径。
- **次范畴**（subcategorization）——次范畴框架本质上是由该动词短语类型与其子女

⊖ 动词的过去分词形式但又不是以 be 或 have 引导，这称为半被动。

的短语类型连接而成的字符串。

最有可能出现的情况的是，开发人员不得不为每个词都执行特征选择，以获得每个特定词的最佳特征集。

3. 无监督的方法

由于缺乏用于训练通用分类器（针对给定语言每个单词的每个词义）的标注训练数据，词义消歧的进展受到了极大的阻碍。针对这一问题，有一些解决途径：

1）设计一种方法对词的不同出现实例进行聚类，以使使每个聚出的类都有效地将该词的相关实例限定为某一特定词义。这种方式可被视为基于聚类的词义归纳（sense induction）。

2）使用某种度量标准计算给定实例与该词某些已知词义组之间的接近度并选择最接近的词义作为该实例的词义。

3）每个词义都从一个种子（seed）实例集开始，然后采用迭代的方式不断对这些集合进行扩展并最终形成聚类结果。

在这里，我们将不会对多数基于聚类的词义推导方法进行详细讨论。我们假设每词都有一个预定义的词义集，无监督的方法将使用非常少（如果有的话）的手工标注实例，并尝试把未见的测试实例分类为对应的预定义的词义类别之一。

我们先来看看使用某种形式的距离测度来标识词义的算法。Rada 等人［53］介绍了一种计算 WordNet 中一对词义间最短距离的度量。此度量假设多个同现词很可能会展现出使其在语义网络层次关系（如 WordNet 中的 IS A 关系）中距离最小化的词义。Resnik［54］提出了一种新的语义相似性测度，即 IS A 义类层级的**信息内容**（information content），并获得了比简单边计数测度好得多的结果。Agirre 和 Rigau 的工作［55］进一步完善了该测度并提出了**概念密度**（conceptual density）指标，它不仅依赖于边数，对层次结构的深度和概念的密度也很敏感，并且与被测量的概念个数无关。图 4-4 中的每个子层都定义了
[112] 概念密度。落在具有最高概念密度的子层中的词义会被选为正确的词义。

$$CD(c,m)=\frac{\sum_{i=0}^{m-1}\text{hyponyms}^{i}}{\text{descendants}_c} \tag{4.3}$$

在图 4-4 中，词义 2 具有最大的概念密度，因此将被选为目标词的词义。

Resnik［56］观察到选择性限制和词义有密切的关系，并确定了一种基于谓词-论元统计的词义计算指标。请注意，该算法主要用于对用作动词谓词-论元的名词进行消歧。

令 A_R 为与谓词 p 就论元 R 而言与概念 c 的选择性关联。A_R 按如下定义：

$$A_R(p,c)=\frac{1}{S_R(p)}P(c\mid p)\log\frac{P(c\mid p)}{P(c)}$$

如果 n 是谓词 p 论元关系 R 中的名词论元，$\{s_1, s_2, \cdots, s_k\}$ 是其可能的词义，则对 i 从 1～k，计算：

$$C_i=\{c\mid c\text{ 是 }s_i\text{ 的准先}\} \tag{4.4}$$

$$a_i=\max_{c\in C_i}A_R(p,c) \tag{4.5}$$

其中，a_i 是词义 s_i 的分值。拥有最大 a_i 值的词义 s_i 会被选为该词的词义。若有多个则随机选择一个。

Leacock、Miller 和 Chodorow 的工作［58］提供了另一种使用语料库统计信息和
[113] WordNet 关系的算法，该工作表明，单义关系可用于消除词歧义。

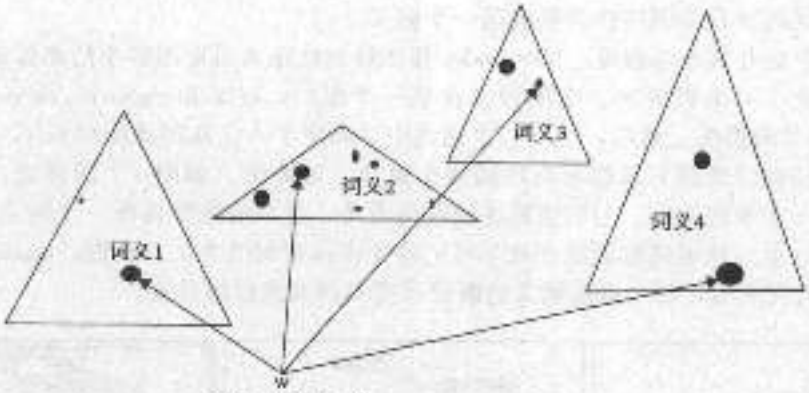


图 4-4 标志密度

4. 跨语言证据驱使的算法

有另一大类基于跨语言信息或证据的无监督算法。Brown 等人［47］可能是第一个在词义消歧中使用这些信息的工作。他们不仅对限于单语字典资源的词义区别感兴趣，也对由于跨语言翻译而带来的词义差别有特别的兴趣。他们提供了一种利用给定词的上下文信息以找出其最可能的目标语言翻译的方法。Dagan 和 Itai［59］对这一想法进行了进一步的探讨，使用双语词典配合单语语料库来自动获取词义的统计信息。他们还提出，句法关系和词同现统计信息都是词汇歧义消解很好的知识源。Diab［69］做了进一步实验，使用英语到阿拉伯语的机器翻译结果来抽取用于训练有监督分类器的词义信息，这些实验和其他纯的无监督方法相比毫不逊色。图 4-5 描述了 SALAAM 算法，该算法需要用到词对齐的平行语料。

1. L1 语言中翻译到 L2 语言后为同一个词的那些词将被划分为一个簇。
2. SALAAM（Sense Assignment Leveraging Alignment and Multilinguality，利用对齐和多语言的词义分配）依据簇中词语词义在 WordNet 中的接近度为簇中单词标识适当的词义。其中，词义接近度的计算基于 Resnik[57] 提出的信息论方法。
3. 使用一个词义选择评标来为簇中的每个词选择一个统一组合适的词义标签。
4. 簇中单词所选定的词义标签将被回传给它们各自在平行文本的上下文中。简而，SALAAM 还将回传的词义标签从 L1 语言的词选射到其 L2 语言中的对应翻译上。

图 4-5 创建使用平行的英语到阿拉伯语机器翻译进行训练的 SALAAM 算法

5. 半监督的方法

下面我们将探讨的算法是从一个小的种子实例集开始并采用迭代的方式使用分类器来识别更多的训练实例。这种额外的自动标注数据可以进一步被用于扩充分类器的训练数据，以在下一个选择周期中获得更好的预测。Yarowsky 算法［61］是这类算法的经典案例，它开创性地在词义消歧问题中引入了半监督方法。该算法所基于的假设是语料库所表现出的两个有力的特性：

1）**每个搭配一个词义**（one sense per collocation）：句法关系和给定词周边所出现词语的类型往往会对该词确定某词义提供强有力的标识。

2）**每个语篇一个词义**（one sense per discourse）：通常情况下，在一个给定的语篇

中，同一个词的所有实例往往会唤醒同一个词义。

基于存在这些属性的假设，Yarowsky 算法可迭代地为给定语篇中的多数词进行消歧。

图 4-6 显示了该算法的三个阶段。在第一个框中，*life* 和 *manufacturing* 用作识别 *plant* 两个词义的搭配。然后，在下一次迭代中，确定了一个新的搭配词 *cell*，而最后一块则显示了算法最终遗留下来的未消歧词的小集合。该算法（如图 4-7 所描述），已被证明在少量的例子中表现良好。为使该算法能获得成功，重要的是要选择一个确定种子实例的好方法并设计出一种能确定会潜在破坏标记地的错误实例的方法。最近，Galley 和 McKeown [62] 研究表明，每个语篇词义的假设可提高词义消歧的性能。

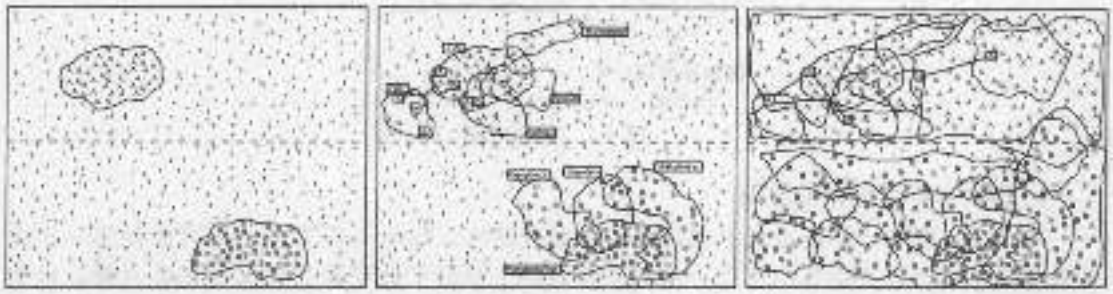


图 4-6　Yarowsky 算法的 3 个阶段

第 1 步：在一个足够大的语料库中识别出给定的待消歧多义词的所有实例，将其上下文一起存储下来。

第 2 步：确定能有力代表该词某词义的一个小实例集。这一步可以依据所考虑词词义使用的有力标识、采用完全无监督的方式进行搭配识别而完成，也可以手动标注一小部分数据。本例中，我们假设一个多义词只有两个词义，但此算法可以扩展到处理 n 个词义。

第 3 步：

第 3a 步：在此例子集上训练一个有监督的分类器。

第 3b 步：使用这些分类器为语料库中词的余下实例进行分类并选择分类置信度大于一定程度的实例。

第 3c 步：使用“每个语篇一个词义”的限制过滤掉可能错误分类的实例，并识别出新的搭配以加入种子搭配表中。

第 3d 步：不断重复第 3 步，逐步减少未分类实例。

第 4 步：停止。有时还会测量一个小的、稳定的未分类实例集。

第 5 步：训练好的分类器现在可以用于分类新数据，并且反过来也可以用于对原始语料标注词义和概率。

图 4-7　Yarowsky 算法

半监督系统的另一种变体是使用无监督方法来创建数据并结合有监督方法从数据中学习模型。其假设是该过程中从语料里所选的错误实例的潜在噪声足够低而不至影响学习的性能。另一个假设是，模型的整体甄别能力优于纯无监督方法或没有足够的手工标注数据而训练出来的纯有监督系统。Mihalcea 和 Moldovan [63] 描述了一个这样的系统，其中使用图 4-8 的算法来从大规模语料中获取特定 WordNet 词义的实例。

Mihalcea [64] 提出了如下方法，将维基百科用于自动词义消歧。

- **提取**（extract）维基百科中包含待消歧词且该词为链接的所有句子。有两种类型的链接：简单链接，如[[bar]]，或管道链接，如[[musical_notation|bar]]。
- **过滤**（filter）指向消歧页面的链接。这意味着我们需要进一步的信息对该词进行消歧。如果该词并不指向消歧页面，那么这个词本身就可以是标签。对于所有管道链接，管道前的字符串可以用作标签。

第1步 预处理

- 对于词W的每个词义，确定包含它的WordNet所有同义词集。对于每个同义词集，找出其中的单义词，对每个同义词集所附的注释定义进行分析。

第2步 搜索

- 使用如下按优先次序的过程来构造查询短语：
  1) 被取输1步中选定的同义词集中的单义同义词（如果有的话）。
  2) 将注释中无歧义的分析成分选择为搜索短语。
  3) 在分析完该注释后，将所有停用词替换为NEAR运算符并为些由同义词集中的词创建一个查询。例如，*produce*#6的同义词集为*grow*、*raise*、*farm*、*produce*，而注释是*cultivate by growing*，则所生成的查询看起来应该像：*cultivate* NEAR *growing* AND (*grow* OR *raise* OR *farm* OR *produce*)。
  4) 反使用在同一词集中采用AND运算符合并的短语的中心短语。例如，如果*company*#5的定义为*band of people*，而其同义词集是（*party*，*company*），则相应的查询将是，*band of people* AND (*party* OR *company*)。
- 使用上一步所确定的短语在网络上搜索并收集匹配的文档。
- 从这些文档中抽取包含这些单词的句子。

第3步 后处理

- 只保留那些所考虑词词作与目标词义相同的句子，而删除其他句子。

图4-6 Mihalcea和Moldovan[63]用于通过查询超大规模语料生成标记为某特定词义的词语实例的算法

- **收集**（collect）所有与该词相关的标签，然后将它们映射到可能的WordNet的词义上。它们有时可能都被映射到相同的词义上，本质上导致动词变成单义，而无用（就此目的而言）。这些类别经常可以被映射到大量的WordNet类别中，从而提供词义消歧后的数据用于训练。此手工映射是一个相对廉价的过程。 115

此算法提供了一种可抽取许多词词义信息的廉价方式，这些词显示了所需的属性。它可以减轻手动密集型的词义标注过程。如果整个维基百科中展现出该属性的单词不少，则它就是一种用于产生词义标记数据非常有用的方法。这种方法的覆盖面有多大呢？我们大致可以从如下事实中得到一些粗略的印象：SENSEVAL-2和SENSEVAL-3所用到的49个名词中大约有30个在从维基百科中抽取到的数据中能找到超过两个词义。这些词义的平均消歧准确率约为85%上下。而将这些词义映射到WordNet的多标注者一致性大约是91%。

4.4.3 软件

针对词义消歧问题研究者们开发了不少软件工具，从相似度计算模块到完整的消歧系统都有。我们不可能将所有的软件都列出来，下面仅列出了其中的一小部分。

- IMS（It Makes Sense） http://nlp.comp.nus.edu.sg/software。这是一个完整的词义消歧系统。
- WordNet-Similarity-2.05 http://search.cpan.org/dist/WordNet-Similarity。此工具包包括了一些基于WordNet的相似度计算Perl模块，可快速计算各种词语相似度。
- WikiRelate! http://www.h-its.org/english/research/nlp/download/wiki-pedia-similarity.php。此工具支持基于Wikipedia分类的词语相似度计算。 116/117

4.5 谓词-论元结构

浅层语义分析，或现在俗称的**语义角色标注**（semantic role labeling），是识别出句子中谓词的各种论元的过程。对于各种谓词的论元集究竟由什么构成以及相应论元标

签应该是什么粒度等问题在语言界已经争论了几十年，这里的谓词可以是句子中的动词、名词、形容词和介词[65，66]。

4.5.1 资源

20世纪90年代后期出现了两个重要的包含语义标注的语料库。一个是FrameNet[⊖][67，68，69，70]，另一个则是PropBank[⊜][71]。这些资源导致有悠久传统的规则（Rule-Based）方法逐渐向面向数据（Data-Oriented）的方法过渡。这些方法更侧重于将语言学知识转化为特征而不是规则，并让机器学习框架使用这些特征来学习一个模型。此模型则有助于对这些资源中编码的语义信息进行自动标记。FrameNet基于**框架语义学**（frame semantics）。其中，一个给定谓词会唤醒**语义框架**（semantic frame），进而对属于该框架的部分或所有可能的语义角色进行实例化[72]。另一方面，PropBank则基于Dowty[73]的原型理论并采用一种语言学更中性的观点。其中，每个谓词都有一组谓词相关的核心论元集，所有谓词则分享一组非核心（或附加的）论元。它建立在宾州句法树库语料基础上。下面我们将更详细地讨论这些资源。

1. FrameNet

FrameNet包含许多英语谓词的面向框架的语义标注，它也包含了标注的句子，这些句子来自英国国家语料库（British National Corpus，BNC）。FrameNet标注过程包括识别出特定语义框架并创建一组称为**框架元素**（frame element）的框架专用角色，然后，确定一组实例化该语义框架的谓词（无论其语法范畴是什么）以及标记了这些谓词的句子集。标记过程如下：首先，确定由该谓词原形实例所唤醒的框架，然后识别出该实例中的各语义论元，并将它们分别标记为该框架预定义的各框架元素之一。谓词原形与其所唤醒框架的组合称为词汇单元（Lexical Unit，LU），也就是词和其意义的组合。多义词的每个词义都倾向于与某个独特的框架相关联。例如，动词*break*可以表示“不遵守（法律、规则或协议）”的意思并与*violation*、*obey*、*flout*等词一同属于COMPLIANCE框架；也可以表示“以破坏性的方式导致突然分裂成碎片”的意思，并与*fracture*、*fragment*、*smash*等词一同属于
110 CAUSE_TO_FRAGMENT框架。

下面的例子说明了其总体思路。此处，框架AWARENESS被实例化为动词谓词*believe*和名词谓词*comprehension*。图4-9显示了AWARENESS框架和其框架元素以及可唤醒它的谓词实例集，涉及动词和名词化词。

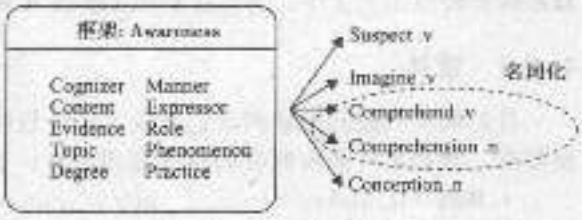


图4-9　FrameNet实例

1) [Cognizer We] [Predicate-verb believe] [Content it is a fair and generous price]

2) No doubts existed as to [Cognizer she] [Predicate-noun comprehension] [Content of it]

FrameNet囊括了多种名词性谓词，包括超名词（ultra nominals）[74，75]、名词以及名词化词（nominalization）。同时它也包含了一些形容词和介词谓词。

截至本文撰写时，最新发布的FrameNet R1.5包含大约173 000个谓词实例，覆盖BNC之上大约1000个框架的约8000个框架元素。虽然框架元素的数量看起来非常大，但

⊖ http://framenet.icsi.berkeley.edu/。

⊜ http://verbs.colorado.edu/~mpalmer/projects/ace.html。

其中很多在 11 000 个词汇单元间还是共享相同的含义。例如，框架 CURE 的框架元素 BODY_PART 和框架 GESTURE 或 WEARING 的相同元素就具有相同的含义。

2. PropBank

PropBank 只包含对动词谓语的论元的标注。宾州树库［75］华尔街日报部分中的所有非系动词都标记了其语义论元。PropBank 将论元边界限制于句法成分，句法成分由宾州树库定义。它使用了一种语言学中性的术语来标记各论元。论元可能被标记为**核心论元**（core argument），其标签类型为 ARGN（N 取值为 0～5）；也可能被标记为如表 4-2 所示的**附加论元**（adjunctive argument），其标签类型为 ARGM-X（X 可能取值为表示时间的 TMP、表示方位的 LOC 等）。附加论元对所有谓词而言意义是相同的，而在核心论元的含义则依具体谓词的不同而不同。ARG0 指 PROTO-AGENT（通常是及物动词的主语），ARG1 是 PROTO-PATIENT（通常是及物动词的直接宾语）［73］。表 4-1 显示了谓词 *operate* 和 *author* 的核心论元表。注意，author 并不包含诸如 ARG2 和 ARG3 等的核心论元。这也说明了并不是所有核心论元都能被所有谓词的所有词义实例化。谓词某词义拥有的核心论元表以及它们的实际意义被放在一个名为框架文件（frames file）的文件中。框架文件会与各自的谓词相互关联。 [119]

表 4-1 PropBank 语料中与谓词 operate.01(词义：操作) 和 author.01(词义：写作或创造) 相关的论元标签

谓词	论元	描述
operate.01	ARG0	施事，操作者
	ARG1	所操作的事物
	ARG2	显式的经历者（操作的施加对象）
	ARG3	显式的论元
	ARG4	显式的工具
author.01	ARG0	作者，施事
	ARG1	创作的文本

表 4-2 PropBank 中的附加论元列表

标签	描述	例子
ARGM-LOC	地点	*the museum, in Westborough, Mass*
ARGM-TMP	时间	*now, by next summer*
ARGM-MNR	方法	*heavily, clearly, at a rapid rate*
ARGM-DIR	方向	*to market, to Bangkok*
ARGM-CAU	原因	*In response to the ruling*
ARGM-DIS	语篇连接	*for example, in part, Similarly*
ARGM-EXT	程度	*at $38.375, 50 points*
ARGM-PRP	目标	*to pay for the plant*
ARGM-NEG	否定	*not, n't*
ARGM-MOD	情态	*can, might, should, will*
ARGM-REC	相互	*each other*
ARGM-PRD	第二谓词	*to become a teacher*
ARGM	空论元	*with a police escort*
ARGM-ADV	副词	除了上述情况以外的情况

图 4-10 显示了一个从 PropBank 语料中抽取的实例，同时也显示了其相应的句法树表示和论元标签。

大多数 Treebank 类型的树都有**迹节点**（trace node），用于指向树中的另一个节点，

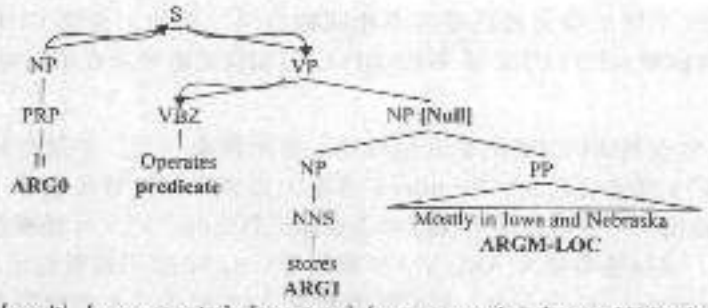


图 4-10　用于解释 PropBank 标签的一个句子的句法树

但并不与单词关联。这些节点也可以标记为论元。由于通常的语法分析器不会再生成这种迹节点，因此研究者们的大多数标准实验常常无视它们。PropBank 还包含共指论元。如同其他集成了多层标注的策略一样，Treebank 标注和 PropBank 标注之间也存在一些分歧。有时 PropBank 使用者（PropBanker）坚信在树结构中存在着错误或该树未满足论元
120 和树节点之间映射的一一对应性。这时，他们会把树中的一系列节点标注为一个论元，并把它们称为**非连续论元**（discontiguous argument）。这种情况极少（为 1%～2%）。PropBank 的 WSJ 部分大约包含 250 000 个例句，有 115 000 个对 5 000 个框架进行实例化的谓词实例，涉及大约 20 个论元类型。还有 18 000 个其他谓词用 Brown 语料的论元标注。最近的 OntoNotes 项目［22，23，24］采用 PropBank 标注指南为更大的各体裁语料库标注了谓词-论元结构，这导致了 Penn Treebank 和 PropBank 指南做出更改以便产生更好、更一致的资源［76］。在本章讨论的多数实验都使用 PropBank v1.0 的 WSJ 部分标注。

FrameNet 和 PropBank 语料之间的一个值得注意的重要区别是，FrameNet 中包含词汇单元，即表明其意义或所唤醒框架的词，而 PropBank 的每个原形词则有一个不同的**框架集**（frameset）表，代表所有具有不同论元结构的所有词义。这些与词义类似但往往是粗粒度的［77］。

3. 其他资源

为了进一步对谓词-论元识别研究，研究者们也开发了其他资源。NomBank［78］的灵感也来自于 PropBank。在识别和标注名词论元的过程中，NOMLEX（名词化词典，NOMinalization LEXicon）［79］词典被逐步扩展到覆盖了约 6000 个条目。同时，来自于 PropBank 的框架也被用于产生 NomBank 的框架文件。另一种将 PropBank 框架与更多谓词无关的专题角色联系起来、也同样提供丰富表示的资源是 VerbNet［81］，该表示将框架集与 Levin 类集［80］相关联。事实上，PropBank 框架和 Levin 动词类，特别是与交集型 Levin 类［82］，有非常强的联系。FrameNet 生成的动词类更多的是数据驱动的（而非基于理论）。从这个意义上说，FrameNet 与 Levin 类也有关系。Baker 和 Ruppenhofer［83］提出了一个关于 FrameNet 框架与 Levin 类是如何相关的有趣讨论。

虽然 FrameNet 和 PropBank 都源于对英语进行谓词-论元结构标注，但很快其理念
121 就传播到其他语言了。由于 FrameNet 基于粗粒语义框架的框架语义学，其语义本质是语言无关的，因此，这些框架很显然可以在标注其他语言数据时重用。SALSA［84，85］是第一个付诸实行的项目。FrameNet 同时标记文本的字面意义和隐喻解释，但这可能导致歧义和较低的一致性，因此 SALSA 项目更多只采用字面意义。该项目尽可能重用了现存的 FrameNet 框架，对于由于语言语义差异而导致无法一致的情况则创建了更多的新框

架。截至本文写作时已存在日语 [86, 87]、西班牙语 [88] 以及瑞典语 [89] 的 FrameNet，还有超过 10 种语言 [90] 的 FrameNet 项目正在进行。

PropBank 也促使双语 [91]、阿拉伯语 [92, 93]、韩语 [94]、西班牙语、加泰罗尼亚语 [95] 以及最近的印地语 [96] 等语言的同类资源被创建出来，许多工作涉及相同的核心研究人员。和 FrameNet 不同，每一个新的 PropBank 都要求建立一套新的框架文件。

虽然 FrameNet 和 PropBank 的设计哲学都启发了其他语言的类似项目，但这也不是实践中的唯一模式。例如，布拉格依存树库 (Prague Dependency Treebank) [97] 就采用了不同的方法——在依存结构顶层的语法构造层中标记谓词-论元结构。其核心论元称为**内部成员** (inner participant)，而附属论元则称为**自由修饰成分** (free modification)。NAIST 文本语料库 (NAIST Text Corpus) [98] 则受日本语言学传统的强烈影响。

4.5.2 系统

和词义消歧不同，从未标注的语料中学习谓词-论元结构的研究很少，这也许是因为它和实际应用更接近，并且已经或多或少地被吸收在信息抽取领域中。大多数早期的系统，如 KL-ONE [99] 和其他系统 [100, 101]，主要是基于启发式语法树，这些是基于规则的系统生成的，直到有了宾州树库可作为监督的语法分析的训练资源。这些系统大部分处理一些与谓词无关的主题角色。已经有很多涉及论元结构概念的语言学研究，但大部分并不直接适用于领域无关的理解系统。在语料库出现之前，主要的资源是基于语法分析树的规则。其中一个很有用的资源在 PropBank 早期文本中由 Levin 提出可以用于动词分类 [80] 及其转换。Absity 分析器 [102, 13] 是最早的基于规则的语义分析器，另外值得注意的是应用到 PUNDIT 理解系统的分析器 [103, 104]。后来很多工作使用混合方法把 WordNet 作为一种解释专门领域的资源 [107] 对主题角色进行标注 [105, 106]。其他值得注意的工作是 Manning、Briscoe 和 Carroll [108] 基于语料库的研究，追求从大语料库中导出次范畴的信息，Pustejovsky [110] 试图从语料库中获取词汇语义知识。

语义角色标注研究的一大飞跃发生在引入 FrameNet 和 PropBank 后，FrameNet 和 PropBank 的一大作用是，创建框架时，可在人工标注的框架集上进行动词分类。在一种或更多的语言中覆盖所有可能的动词需要很多的人力物力。Green Dorr 和 Resnik [111] 提出了自动学习框架结构的一种方法，但结果不够准确，不足以取代人工框架创建。在最新的方法中，Swier 和 Stevenson [112] 用一种无监督的方式来处理这个问题。 122

现在，让我们回顾这些语料出现以来的一些最新方法。语义角色标注的处理过程可以定义为识别出一个词序列的集合，其中每一个序列代表给定谓词的一个语义论元。例如，图 4-10 中的句子，谓词 *operates*，单词 *I* 充当角色 ARG0，单词 *stores* 充当角色 ARG1，词序列 *mostly in Iowa and Nebraska* 充当 ARCM-LOC。因为 PropBank 不承认谓词之间的共性，一个谓词的 ARGN 不必与另一个谓词的 ARGN 有类似的语义。[一]

FrameNet 是第一个手工标注谓词-论元的项目，Gildea 和 Jurafsky [113] 第一个把语义角色标注当成有监督的分类问题，谓词的论元和谓词本身可以被映射到该句的语法树中的节点。他们介绍了三个任务，可以用来对系统进行评估，并已成为标准。

论元识别——识别一个谓词的所有语义成分，这代表谓词所有有效的语义论元。

[一] PropBank 项目当然也不是任意地赋予论元角色数字。例如，实际上 ARG0 倾向于担任 Agent 角色，而 ARG1 倾向于担任 Patient 角色（指承受角色的术语）。

论元分类——对于一个给定谓词的语义成分，为它们标注适当的标签。

论元识别和分类——前两个任务的结合，确定一个谓词的所有论元，并给它们标注适当的论元标签。

一旦一句话已进行句法分析，则分析树中的每个节点可以被归类为某一个语义论元（即非空节点），或不代表任何语义论元（即空节点），非空节点可以进一步标注相应的论元标签。

例如，在图 4-10 中，名词短语 stores mostly in Iowa and Nebraska 是一个空节点，因为它不对应一个语义论元。由节点 NP 包围的 stores 是一个非空节点，因为它对应语义论元 ARG1。

一个通用的语义角色标注算法的伪代码（Semantic Role Labeling，SRL）如算法 4-3 所示。

1. 句法表示

正如我们所看到的，PropBank 是在宾州树库风格的短语结构树顶层上的一层注释。在一个早期的恢复 PropBank 注释的工作中，Gildea 和 Jurafsky [113] 把一些论元的标签添加到句法树，这些树由宾州树库训练得到的分析器产生。在随后的几年中，研究人员直
123 接使用各种其他类型的句子表示形式，或作为一种独立的数据源来解决语义角色标注问题。下面我们看一下这些句子表示形式以及用 PropBank 论元来标记文本的一些特征。

算法 4-3　语义角色标注算法

```
Procedures SRL(sentence) returns 最佳语义角色标注
输入：sentence
1: 生成 sentence 完整的句法分析树
2: 识别所有的 predicate
3: for all predicate ∈ sentence do
4:     从 predicate 于树上的每个节点抽取一组特征
5:     对每个特征向量使用训练的分类器进行分类
6:     选择具有最高得分的分类类别
7:     return 语义角色标注
8: end for
```

短语结构语法（Parse Structure Grammar，PSG）　FrameNet 标记句中的词序列表示论元，而 PropBank 将树中的节点标注为论元。因为一些高质量的统计分析器可以产生短语结构树，而且短语结构表示易于标注，所以 Gildea 和 Jurafsky [113] 使用短语结构来标注。他们介绍以下的几个特征，其中一些是从句子的分析树中抽取的。

路径——这个特征是在句法分析树中从句法成分到正在分类的谓词的语法路径。例如，在图 4-10 中，从 ARG0 到谓词的路径表示为 NP↑S↓VP↓VBZ。↑和↓分别代表在树上向上和向下移动。

谓词——谓词原形被当成一个特征。

短语类型——标注成分的句法范畴（NP、PP、S 等）。

位置——这是一个二元特征，表示短语成分在谓词前面或后面。

语态——该特征表示谓词是主动或被动形式。用一套手写的 tgrep2[⊖] 表达式来在句法树中标识被动语态的谓词。

中心词——短语树的中心词。它可以用 Magerman [114] 描述和 Collins [115] 修改
124 的中心词来计算。

⊖ 参见 http://tedlab.mit.edu/~dr/Tgrep2/。

次范畴——这个特征是短语树中谓词父节点短语结构规则的扩展。例如在图 4-10 中，谓词的次范畴信息为 VP→VBZ→NP。

动词聚类——谓词是预测论元类中最显著的特征之一。给定谓词可能出现的各种句法或语义结构，任意数量的手工标注的训练数据对于模型参数估计的作用将相对有限。任何实际应用的测试集中可能将包含一些在训练中从没见过的谓词的意思或框架。在这种情况下，研究人员发现用这样的特征创建类别可以从关于谓词的一些信息受益。Gildea 和 Jurafsky [113] 用一个距离函数作为聚类依据，从表面上看，有语义相似性的动词可能成为相同的类别。例如，动词如 eat、devour、savor 等倾向于直接描述食物。聚类算法使用 Lin [116] 提出的述宾关系数据表。这些动词可以用 Hofmann 和 Puzicha [117] 的共现概率模型被聚类成 64 类。

Surdeanu 等人 [118] 提出以下额外特征：

实词——因为一些成分的中心词特征，如 PP 和 SBAR，信息量不够，他们对一些成分类型定义了一组启发式规则，用来确定一个所谓的实词，并将其作为一个附加的特征，而不使用通常的中心词查找规则。他们使用的规则如图 4-11 所示。

中心词和实词的词性——添加一些成分的中心词和实词的词性作为特征，有助于论元识别，可显著提升基于树的系统的性能。

```
H1：if 短语类型为 PP，then 选择右孩子
    例子：短语 = "in Texas"，实词 = "Texas"
H2：if 短语类型为 SBAR，then 选择最左句子（S*）的从句
    例子：短语 = "that occurred yesterday"，实词 = "occurred"
H3：if 短语类型为 VP，then
    if 有 VP 孩子，then
      选择最左边的 VP 孩子
    else
      选择中心词
    例子：短语 = "had placed"，实词 = "placed"
H4：if 短语类型为 ADVP，then 选择最右孩子，不是 IN 或者 TO
    例子：短语 = "more than"，实词 = "more"
H5：if 短语类型为 ADJP，then 选择最右边的形容词、动词、名词或 ADJP
    例子：短语 = "51 years old"，实词 = "51"
H6：for all 其他的短语类型中，选择中心词
    例子：短语 = "red house"，实词 = "red"
```

图 4-11 实词的启发式列表

实词的命名实体——一些角色，如 ARGM-TMP 和 ARGM-LOC，倾向于包括 TIME 或者 PLACE 的命名实体。这些信息被添加到二值特征集。

布尔命名实体标志——Surdeanu 等也提出增加一些命名实体信息作为特征。他们创立了 7 个命名实体类型作为指示函数：人物、地点、时间、日期、货币、百分比、组织。

动词短语搭配——此特征包括动词及紧随其后的介词的频率统计。

Fleischman、Kwon 和 Hovy [119] 添加了以下特征到他们的系统：

逻辑函数——这是一个二值（外部论元、对象论元和其他论元）特征，需要使用一些启发式的语法树计算。

框架元素顺序——该特征指在一个句子中的框架元素相对于其他框架元素的位置。

句法模式——此特征也由使用基于短语类型和成分的逻辑函数的启发式产生。

已知角色——这是一组特征显示当前谓词由系统已观察或分配到的 N 个角色。Pradhan 等［120］使用下述的附加特征：

成分中的命名实体——Surdeanu 等［118］提出使用成分中的命名实体对成分的语义角色分类有一定的性能提升。其中一些实体，如位置和时间，对附加论元 ARGM-LOC 和 ARGM-TMP 是特别重要的。当中心词不常用，或对一些地点或时间指示词的封闭集合，如 in Mexico 或 in 2003，实体标记也是有用的。他们采用 IdentiFinder 在语料中标记 7 种命名实体［121］，并增加这 7 个二值特征。如在成分中出现某个命名实体，相应的特征为真。

动词词义信息——一个谓词携带的论元依赖于谓词的词义。在 PropBank 语料库中标注每个谓词的论元集合，依赖于它被使用的词义。这也称为框架集 ID（frameset ID）。表 4-3 说明一个词的论元集。根据谓词 talk 的词义，无论是 ARG1 或者 ARG2 都可以被识别为 hearer。没有这些信息可能对学习机制造成混乱。

从 PropBank 抽取出的动词词义信息通过把谓词的每个意思当成一个独立的谓词添加，
126 这将有助于提高性能。PropBank 框架集的消歧精确率很高［122］。

表 4-3　在 PropBank 语料库中与两个句子的谓词 talk 相关的论元标签

talk.01		talk.02	
标　签	描　述	标　签	描　述
ARG0	说话者	ARG0	谈话者
ARG1	话题	ARG1	和谁谈
ARG2	听众	ARG2	间接行动

介词短语的名词中心词——很多附加论元，如 tempuras 和 locatives，作为句子中的一个介词短语。这些短语的中心词总是介词，往往不是很有区分性。例如，in the city 和 in a few minutes 都拥有相同的中心词 in，两者都不包括命名实体，但前者是 ARGM-LOC，而后者是 ARGM-TMP。因此，Pradhan 等［120］把介词短语的中心词改为介词短语内的第一个名词短语，介词信息被附加到短语类型而得以保留。例如，for about 20 minutes 这个介词短语的中心词，原来是介词 for，变换后，中心词被改为 minutes，短语 pp 被改为 PP FOR。中心词既以其表面形式使用，也以原形使用。通过使用 XTAG 形态数据库[1]自动进行原形化［123］。

成分的第一个和最后一个单词及其词性——一些论元的第一个和最后一个单词往往具有区分性，这两个词及其词性作为 4 个新的特征。

成分位置顺序——此特征避免远离谓词的成分被不合逻辑地认定为论元。这是成分类型和谓词位置顺序的拼接。

成分树距离——这是描述已经存在的位置特征的一种更好的方式，这里谓词成分的距离指在句法树中从一个节点到另一个节点需要穿越的距离。

短语相关的特征——由 9 个特征组成，代表短语类型、父节点的中心词及其词性、短语的左右兄弟节点。增加这些信息是认为树的上下文信息可以改善系统的鲁棒性和泛化能力。

127 **时间提示词**——时间提示词没有被命名实体标注分类器识别，因此可用二值特征表示它们的存在。BOW（Bag Of Words，词袋）工具包[2]被用于识别具有 ARGM-TMP 论元类的且平均互信息最高的单词和词二元组。

[1] ftp://ftp.cis.upenn.edu/pub/xtag/morph-1.5/morph-1.5.tar.gz。
[2] http://www.cs.cmu.edu/~mccallum/bow/。

动态类的上下文——在论元分类的任务中，这些动态特征代表与待分类节点同一棵树上的最多前两个非空节点的假设。

路径泛化 我们将在4.5.2节中看到，对于论元识别任务，路径是最突出的特征之一。然而，它也是最稀疏的特征。为了克服这个问题，路径需用以下几种不同的方法泛化：

基于子句的路径变化——子句节点（S，SBAR）的位置在论元识别中是重要的特征[124]。因此，Pradhan等[120]用4个基于子句的路径特征进行实验。

- 把路径中的所有节点替换为*（不替换子句节点）。例如，路径NP↑S↑VP↑SBAR↑NP↑VP↓VBD变成NP↑S↑*↑S↑*↑*↓VBD。SBAR被替换为S。
- 只保留路径中的子句节点，这对于上面的例子产生NP↑S↑S↓VBD。
- 添加二值特征，显示成分是不是和谓词处于同一子句。
- 去掉S节点之间的节点，路径变为NP↑S↑NP↓VP↓VBD。

n元路径——这些特征分解成一系列的二项元。例如，路径NP↑S↑VP↑SBAR↑NP↑VP↓VBD成为NP↑S↑VP、S↑VP↑SBAR、VP↑SBAR↑NP、SBAR↑NP↑VP等。较短的路径变为空。

单字母短语标签——每个短语类别聚类为一个用第一个字符作为短语标签的类别。

压缩路径——压缩相同标签的序列，直觉是连续嵌套的树上相同的短语可能不会增加额外信息。

无方向路径——删除路径中的方向，从而在树中改变方向的点不那么重要。

部分路径——只使用从成分到谓词和成分的最近共同祖先的路径。例如，图4-10展示的部分路径为NP↑S。

规范化处理路径的另一个工作成果为Vickrey和Koller[125]的工作，他们执行一个基于规则的句子简化，试图自动获得路径泛化。[128]

谓词上下文——此特征能体现谓词词义变化。前两个词和后两个词被加到特征集。这两个词的词性也被加入特征集中。

标点符号 对于一些附加论元，标点符号起着重要的作用。这组特征能够体现标点符号是出现在成分之前和之后。

特征上下文——父母或兄弟姐妹的特征对组成成分的分类是有用的。传统上，每个成分被独立分类。但是，实际上一个成分可以有的论元的类别和数目存在着复杂的关系。换句话说，每个论元的分类依赖于其他节点的分类。正如我们在后面看到的，Pradhan等的方法用论元序列信息执行后处理步骤，但是这不可能覆盖所有可能的约束条件。在现有的体系结构中最好的做法之一是考虑句子的所有非空成分的特征向量组合。这就是所谓的特征上下文。它使用可能非空成分的所有其他特征向量值作为额外的上下文。

组合范畴语法（Combinatory Categorial Grammar，CCG） 我们了解到，尽管路径特征对于论元的识别非常重要，但它是最稀疏的特征之一，并可能难于训练或泛化[126，127]。一个依存语法可以产生较短的从句子中的谓词到依存词的路径。这对于从PSG解析树提取的短语结构语法路径可能是一个更强大的补充。Gildea Hockenmaier的[128]报告说明使用CCG表示提取的特征可以提高核心论元的语义角色标注的性能（ARG0～5），因为CCG树是二叉树，而且其成分很难和谓词的语义论元对齐，研究人员进行的实验中使用中心词，而不是整个跨度的短语。后来，Pradhan等（2005）[129]同这些特征增强原有的短语结构树算法，以获得更多益处。

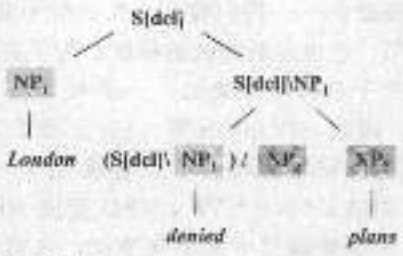
图4-12显示了CCG分析的句子：London denied plans on Monday，Gildea和Hock-

emmaier [128] 介绍了三个特征：

短语类型——这是两个词（谓词及依存词）之间最大的投影范畴。

London denied plans on Monday

w_0	w_d	c_0	l
denied	*London*	$(S[dcl]\backslash NP_1)/NP_2$	1
denied	*plans*	$(S[dcl]\backslash NP_1)/NP_2$	2
on	*denied*	$((S\backslash NP_1)\backslash(S\backslash NP_1)_2)/NP_3$	2
on	*Monday*	$((S\backslash NP_1)\backslash(S\backslash NP)_2)/NP_3$	3



129

图 4-12　范畴组合的语法分析

范畴路径——把以下三个值拼接起来形成的特征：1）依存词类别；2）依存方向；3）依存词填充的范畴槽。例如，在图 4-12 的树中，denied 和 plans 之间的路径为（S[dcl]\NP1)/NP2.←。

树的路径——这是以 Charniak 分析器为基础的系统的路径特征在 CCG 中的类似物。它追溯了在二叉 CCG 树中从依赖词到谓词的路径。

树邻接语法（Tree-Adjoining Grammar，TAG）　Chen 和 Rambow [130] 给出两种不同的特征集结果：1）Gildea 和 Palmer [131] 系统使用的表面语法特征；2）从宾州树库中的 TAG 抽取的额外特征。他们选择了 TAG，因为它有能力解决文中的长距离依赖性。他们使用的额外特征是：

超级标签路径——这些特征和前面看到的路径特征是相同的，只是它是来自于 TAG，而不是来自于 PSG。

超级标签——这个特征是树框架对应的谓词或论元。

表层句法角色——这个特征是论元的浅层语义角色。

表层次范畴——这个特征是次范畴框架。

深层语义角色——这个特征是论元的深层角色，其值包括主语和直接宾语。

深层次范畴——这是深层语法次范畴框架，比如，对于一个及物动词，它可能是 NP0_NP1。如可能，修饰 NP 的介词将被用于词汇化该特征。所以，对于谓词 load，可能的框架是 NP0_NP1_NP2(into)。

语义范畴——Gildea 和 Palmer 使用语义次范畴框架，这里，除了语义范畴，这些特征还包括语义角色信息。

虽然很多研究者手动构建各种各样的特征，Moschitti、Pighin 和 Basili [132] 尝试了不同的方法。他们使用树核从大量自动生成的模式中识别并选择子树模式，以捕获树的上下文信息。不过在这个应用中，性能比手动选择的特征稍微差一点，可能对于其他机器学习的问题，当手工选择的特征很不直观时，这个技术是有价值的。

依存树　到目前为止的一个理论问题是，系统的性能依赖于宾州树库中标注的论元精确集合。只有当它们和 PropBank 的标签是完全一样的，标注才是正确的，括号和标签都必须是匹配的。因为 PropBank 和大多数语法分析器是在宾州树库语料的基础上开发的，因此是基于相同的语法结构，可以期望上述两种形式会比其他的表示更匹配 PropBank 的标注。但是，这里的分数越高，是否意味着其输出对建立在这些角色标签之上的应用而言更好用呢？特定的括号标注经常不是那么重要，而论元中心词和谓词之间的关系才是更关键的信息。使用这些策略可以使算法的输出产生更高的性能，F 值（F-score）约为 85%（比较原来

130

的79%)。

Hacioglu [133] 给出了用 Hwa、Lopez 和 Diab [134] 的脚本程序将宾州树库的树转化为依存树的语义角色标注的问题，建立了带有 PropBank 论元的依存结构树。这个系统上的性能比其在短语树上的 F 值性能有 5 个点的提高。和其他方式比较，一个可能的缺点是，所有的分析器在相同的宾州树库上进行训练，并进行评估。如果在非 WSJ 语料上评估，则性能下降。Pradhan 等 [129] 进行实验找出基于规则的依存分析器的性能如何。Minipar [135, 136] 是一个基于规则的依存分析器。它输出中心词和其他修饰词之间的依存关系。每个词只能修饰至多一个词。依存关系形成依存树。在 Minipar 的依存树中每个节点的一组词形成一个在原句段中连续的片段，并对应于成分树中的成分。图 4-13 显示了谓词 kick 的论元如何映射到短语结构语法树和 Minipar 分析树的节点。代表成分中心词的节点是分类的目标。它们使用和 Hacioglu [133] 相同的特征（参见表 4-4）。

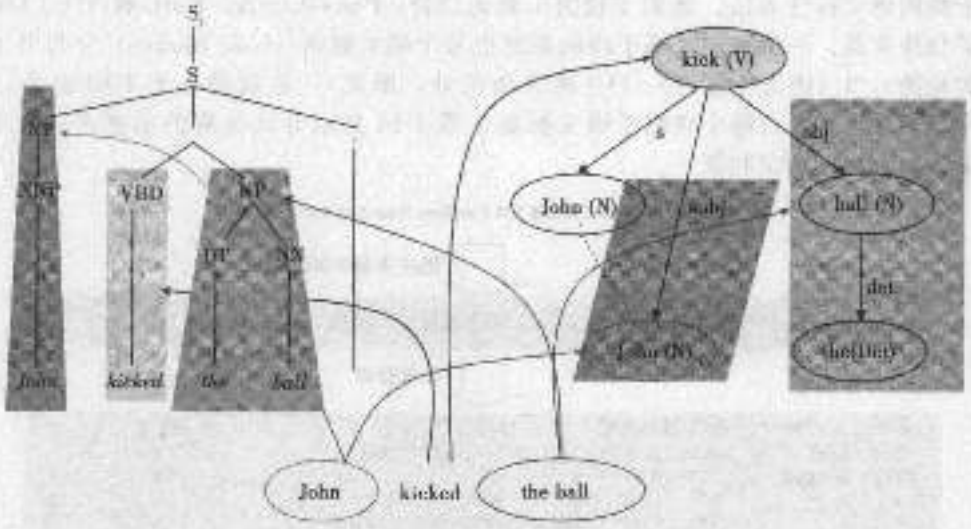


图 4-13 新的体系结构

131

在 PropBank 语料上，Minipar 性能比基于 Charniak 的系统差很多（47.2，如果使用严格的跨度标准计算）。这正如预期的，因为 Minipar 并不是设计为产生完全匹配宾州树中采用的分割成分的组成成分。

表 4-4 用 Minipar 分析器的基线系统所使用的特征

中心词	在依存树中表示节点的词
中心词词性	中心词的词性
词性路径	在树中连接各节点的词性的依存树中，指定到中心词的路径
依存路径	每个连接到中心词的词有一个依存关系，边词词组上的标签表示。本特征由连接两词的路径上的依存标签组成
语态	谓词的语态
位置	节点在谓词之前或之后

在 Hacioglu [133] 的实验中，不匹配的 8% 是从树库到依存树的转化而导致的。使用自动生成的有错的树，会产生更高的不匹配。对于 CCG 的分析树，如 Gildea 和 Hockenmaier [128] 的结论，不匹配率约为 23%。一个更可行的对性能评分的方式是对赋给成分的中心词的标注进行评分，而不考虑成分的真正边界。这个结果约为 61.7 的 F 值，这

好得多，并且提供了正交的好处。这些结果是把依存树和短语结构谓词-论元结构集成到一起的令人信服的证据。

从那时起，在依存分析树分析方面就有很多工作可做，并有了一系列的实验，举行了两个自然语言处理学习（Computational Natural Language Learning，CoNLL）共享任务[137，138]以推进依存分析和语义角色标注的结合，这些包括 Johansson 和 Nugues [139] 使用更丰富的句法依存表示，考虑树间隔（gap）和迹（trace），并把 PropBank 谓词和论元映射到这种表示。

基本短语块　一个常见的问题是，完整的语法表示对语义角色标注任务有帮助吗？换句话说，在对一个谓词的论元分类前，创建一个完整的句法树有多重要？块表示也许更快，对出现在语音数据中的短语重排序可能更鲁棒。Gildea 和 Palmer [131] 使用基于块的方式探讨这个问题，结果是，句法分析有助于填补这个大差距。Hacioglu [124] 进一步尝试基于块的语义标注方法，达到了较为乐观的结论。Punyakanok、Roth 和 Yih [140] 也报告了分块实验。一般来说，基于块的系统把每个基本短语（base phrase）分为 B（语义角色的开始）、I（语义角色内）、O（语义角色外，即无），这就是一个 IOB 表示，用另外一个 SVM 分类器为每个块赋予语义标签。图 4.14 显示分块流程的示意图。表 4-5 列
[132] 出语义分块程序的语义特征。



图 4.14　语义组块器

对于每个待标注的单位（基本短语），通过围绕着每块的固定大小的上下文创建一组特征。除了上述特征，分块程序使用先前已经被赋到语境中单位的语义标签。用 5 个单位的滑动窗口来表示语境。在识别和分类任务中的性能约为 70 的 F 值。

2. 分类范式

在上一节中，我们考察了多种句子层次的结构表示，用来解决语义角色标注问题，也考察了这些表示的特征，这些特征可训练一个模型从而实现自动识别。在这一部分中，我们专注在这个问题所使用的机器学习方法。这些方法有着不同的复杂性。最简单的方法是把语义角色标注当成一个纯粹的分类问题，其中每一个谓词的论元分类和该谓词其他论元的分类是独立的。其他的研究人员采取了这种基本范式，但是增加了一个简单的后处理除

去了一些明显不可能的分析结果，比如两个论元的重叠。一些更复杂的方法用针对具体论元的语言模型或框架元素组统计增强了后处理步骤。这些后处理在很大程度上解决了原有的独立性假设带来的问题。

也有一些更复杂的方法进行所有论元的联合解码，试图捕捉论元间的相互依赖关系。[133] 可惜这些方法到目前为止只产生了轻微的效果，部分原因在于使用一个单纯的分类器，再通过一个后处理器，已经可以取得较好的效果。在这一部分，我们不再提供所有方法的详细描述，而专注于一种当前有较高性能的方法，该方法能有效地利用多知识源。并且采用一种联合架构，当处理非训练类型的文本时，性能不会急剧下降。

表 4-5 基于组块的分类器使用的特征

词	块中的词
词偏原形	词词的原形
词性标记	块中词的词性
基本短语位置	基本短语中词元的位置，采用 IOB2 表示，如 B-NP，I-NP，O
子句标记	标记与子句相关的句中词元位置的标记串
词元位置	与谓词相关的短语位置，取 3 个值，"before"，"chic" 以及 "-"（对谓词自身）
路径	定义词元与谓词间的一个路径
子句括号模式	
子句位置	表明词元是否在包含谓词的子句之内或之外的二元特征
中心词后缀	中心词的长度为 3～4 的后缀
距离	词元与谓词间的距离，以基本短语个数或 VP 块的个数来度量
长度	块中词的个数
谓词词性标记	谓词的词性标签
谓词频率	常用或非常用（阈值为 5）
谓词基本短语上下文	以谓词为中心的左右窗口大小为 2 以内的基本短语链
谓词词性上下文	紧邻谓词前后的词的词性标记
谓词论元框架	谓词左右的核心论元模式
谓词个数	句子中谓词的个数

首先，Gildea 和 Jurafsky [113] 给出了改进的语义角色标注算法，包括两个步骤。首先，该系统基于两个特征 P(论元 | 路径，谓词) 和 P(论元 | 中心词，谓词)，计算成分作为论元的最大似然概率。其次，对每个其论元概率非 0 的成分，通过对条件依赖于[134] 各种特征集的概率分布进行插值，归一化概率，并选择最可能的论元序列。他们所用的概率分布如表 4-6 所示。

表 4-6 从 Charniak 分析树抽取的特征计算的用于语义论元分类的概率分布

分布
P(论元 | 谓词)
P(论元 | 短语类型，谓词)
P(论元 | 短语类型，位置，语态)
P(论元 | 短语类型，位置，语态，谓词)
P(论元 | 短语类型，路径，谓词)
P(论元 | 短语类型，路径，谓词，子类化架)
P(论元 | 中心词)
P(论元 | 中心词，谓词)
P(论元 | 中心词，短语类型，谓词)

Surdeanu 等人 [118] 使用决策树分类算法 C5[142, 143]，并采用和 Gildea、Jurafsky[113] 相同的特征。分类器内置的增强能力使性能略有改善。Chen 和 Rambow[130] 提出使用决策树分类器 C4.5[142]。Fleischman 和 Hovy[144] 给出了在 FrameNet 语料上使用最大熵分类器的结果。Pradhan 等人[120] 使用 SVM 在 PropBank 语料上获得了更好的性能。尽管如此，最大熵分类器和 SVM 的结果差异是很小的。

Pradhan 等人 [120] 给出了利用各种分类器在同一数据集使用相同特征集的结果，并进行比较。Gildea 和 Palmer [131] 的系统使用几种不同特征集估算后验概率并且进

行插值，这和 Gildea，Jurafsky [113] 系统的估计完全一样。而 Surdeanu 等人 [118] 使用决策树分类器。表 4-7 显示三种不同系统的论元分类性能。

表 4-7　不同的分类器使用相同的特征的论元分类

分类器	准确率 (%)
SVM (Pradhan 等 [120])	88
决策树 (Surdeanu 等 [118])	79
Gildea 和 Palmer [113]	77

在实验中，我们使用 TinySVM[⊖] 和 YamCha[⊜] 一起作为 SVM 训练和测试软件 [145，146]。SVM 的参数，如核函数的类型及其他参数的值是使用开发集根据经验确定的。选择一个度为 2 的多项式核函数，其违反边界的单位成本 $C=1$，终止条件公差 $\varepsilon=0.001$。

[135]

SVM 在文本分类任务中的性能良好，在这些任务中，使用高维稀疏特征向量来表示数据 [147，148]。受到使用 SVM 成功为短语块进行标注的启发 [145]，Pradhan 等人 [149，126，120] 把语义角色标注问题看为一个使用 SVM 的多元分类问题。

SVM 本质上是一个二元分类器，但多元分类问题也可以归为多个二元分类问题，可以采用一对一的成对方式或者一对多的 (One Versus All，OVA) 方式 [150]。对于使用成对方法的 N 类问题，为每对可能的类别训练 $\frac{N(N-1)}{2}$ 个二元分类器。而对于 OVA 方式，N 个二元分类器被训练为区别每个类和由剩余类组合创建的元类。比较这两种方式，主要是要训练的分类器数目及训练分类器使用的数据规模的权衡。虽然有些实验的结论是成对的方法优于 OVA 方式 [151]，但 Pradhan 等人 [120] 的初步实验显示 OVA 有更好的性能。因此，他们选择了 OVA 的方法。

SVM 输出从最大间隔超平面到一个特征向量的距离。为了便于产生概率阈值并生成 n-best 假设格，他们通过将 sigmoid 函数拟合到得分，把距离转换为概率。如 Platt [152] 所述。

这个系统包括两个阶段：训练阶段和测试阶段。我们首先讨论 SVM 是如何进行训练的。因为支持向量机的训练时间随着训练实例数目的增长而呈指数增长，而在句法树中，90%的节点都具有空论元标签，将训练过程分为两个阶段是更有效的：

1）过滤掉高概率为空的节点。在整个数据集上训练一个二元的空或非空分类器。如 Platt [152] 所描述，将 sigmoid 函数拟合到原始得分以转换为概率。分类器处理所有的训练实例，空角色和非空角色的相应得分被 sigmoid 函数转换为概率。最有可能为空的节点 (概率>0.90) 被从训练集中剪掉。这可使空节点的数目减少 90%，节点总数减少约 80%。这可能会剪掉一些非空节点，但可忽略不计 (约 1%)。

2）其余的训练数据被用来训练包含一个空类的所有类的 OVA 分类器。

采用这个策略，只有一个分类器 (空或非空) 在所有数据上训练。其余的 OVA 分类器在过滤节点上训练 (约为总数的 20%)，从而能够大大地节省时间。

在测试阶段，所有的节点被分类器直接归为空角色或在第 2 阶段训练的分类器训练的角色。如果第一遍时我们对测试集采用空或非空的 OVA 分类器来进行过滤，就像我们在训练阶段所做的那样，则召回率会有一点下降。这很少的性能提升只需要很少的计算成本，这是因为在测试阶段，SVM 的速度是非常快的。测试算法的伪代码如图 4-3 所示。其中一个变化是，这种策略将在第一阶段过滤所有的空实例而不是只剪掉高概率的空节

[136]

⊖ http://chasen.org/~taku/software/TinySVM/。

⊜ http://chasen.org/~taku/software/yamcha/。

点。但这种方法会使性能显著下降。

对于标准的树库分析树，这样的系统对论元识别和分类联合任务的性能是 90%以上，而对于自动生成的分析树，性能是接近 80%。

3. 克服独立性假设的问题

正如前面所提到的，已经提出各种后处理方法，它们作为一系列独立的论元分类步骤，用来克服语义角色标注的限制。我们现在来看这样的一些策略。

不允许重叠 由于每个成分的分类和其他成分的分类是独立的，可能两个重叠的成分被赋给同一个论元类型。因为我们的处理对象是分析树，树中节点的重叠总是有一个祖先-后代的关系，因此重叠被限制为如例 4-1 所示的包容。

例 4-1 [ARG0 nobody] [predicate knows] [ARG1 at what level [ARG1 the futures and stocks will open today]]

在这里出现了问题，因为重叠的论元在 PropBank 中是不允许的（或者，更具体地说，一个动词谓词的任意两个论元在 FrameNet 中是不会重叠的）。解决这个问题的一个方法是在重叠的论元中进行选择，仅保留在 SVM 上获得最高可信度（基于分类概率）的一个，而其他的设置为空角色。应用 sigmoid 函数对 SVM 原始得分进行转换得来的概率可用作可信度的度量。

论元序列信息 另一种方法是使用已有的信息，如 Gildea 和 Jurafsky (2002) [113] 提出，一个谓词可能包括一组特定的论元类型，从而提高统计论元标注性能。一个类似的但更有条理的方法包括增加一些额外的约束，论元顺序信息被保留，谓词也被看作为论元，是序列的一部分。可以这么做来实现这个想法：首先像前面介绍的将原始的 SVM 分数转化为概率，在论元序列中训练 trigram 语言模型。然后，对于每个正在分析的句子，对语法树上的每个节点使用 n-best 假设产生论元格，通过格（该格使用由 sigmoid 得到的概率作为观察概率以及语言模型概率）执行 Viterbi 搜索，找出最大似然路径，这样每个节点被分配一个属于 PropBank 的论元或空值。

该搜索被限制为这样的一种方式：没有两个非空节点重叠。为了简化搜索，Pradhan 等人 [120] 只允许空角色被赋予其空概率高于阈值的节点。而通过训练语言模型，我们可以用实际的谓词估计转移到谓词或从谓词转移出的概率，或者我们可以对所有的谓词执行一个联合估计。Pradhan 等人发现合并识别和分配语义论元可以提高核心论元的精确性，而附加论元的精确度稍微降低了。这是合乎逻辑的，对附加论元的顺序和数量的约束比较宽松。因此，使用这种策略对于核心论元是有益的。同时也使用了一些其他的使用论元上下文信息的策略。Toutanova，Haghighi 和 Manning [153] 提出对于给定谓词用对数线性模型来预测语义角色的一种全局模型。而 Punyakanok 等人 [154] 用一种基于整数线性规划为基础的推理框架来提高语义角色标注的性能。这两种方法与 Viterbi 方法相比有一些提高。 137

4. 特征性能

在每个任务中，并不是所有的特征都有用。某些特征在有些上下文中加入了更多的噪声而不是有用信息。特征的功效取决于它们所使用的分类范式。表 4.8 显示了把每个特征分别加到论元分类和论元识别的任务中所获得的对于基线系统的效果。此外，将命名实体的特征加入到空、非空分类器会使性能下降，这种效应主要归因于两个问题：1）很多包括命名实体的成分不是谓词的论元（一个论元的父节点可能包括相同的命名实体），在空或非空的分类中这成为一个噪声特征；2）SVM 不能较好处理无关的特征 [155]。当这些从树库抽取的特征被单独用在代表论元的成分分类时，整体分类准确率从 87.9%上升到 88.1%，而

添加中心词的词性作为一个特征将显著地提高论元分类和论元识别任务的性能。

表 4-8　把每个特征加到基线系统，对论元分类和论元识别中特征的影响。星号表示改进在统计上是显著的

特　征	论元分类	论元识别		
	准 确 率	精 确 率	召 回 率	F_1
Baseline [120]	87.9	90.7	88.5	91.8
+ Named entities	88.1	90.8	88.9	91.9
+ Head POS	*88.6	94.4	90.1	*92.2
+ Verb cluster	88.1	94.1	89.0	91.5
+ Partial Path	88.2	93.3	88.9	91.1
+ Verb sense	88.1	93.7	89.5	91.5
+Noun head PP (only POS)	*88.6	94.4	90.0	*92.2
+Noun head PP (only head)	*89.8	94.0	89.4	91.7
+Noun head PP (both)	*89.9	94.7	90.5	*92.6
First word in constituent	*89.0	94.1	91.1	*92.7
+Last word in constituent	*89.4	93.8	89.4	91.6
+First POS in constituent	88.4	94.4	90.6	*92.5
+Last POS in constituent	88.3	93.6	89.1	91.3
+Ordinal const. pos. concat.	87.7	93.7	89.2	91.4
+Const. tree distance	88.0	93.7	89.5	91.5
+Parent constituent	87.9	94.2	90.2	*92.2
+Parent head	85.8	94.2	90.5	*92.3
+Parent head POS	*88.5	94.3	90.5	*92.3
+Right sibling constituent	87.9	94.0	89.9	91.9
+Right sibling head	87.9	94.0	89.9	*92.1
+Right sibling head POS	88.1	94.1	89.9	92.0
+Left sibling constituent	*88.6	93.6	89.5	91.6
+Left sibling head	86.9	93.5	88.1	89.9
+Left sibling head POS	*88.8	93.5	88.5	91.4
+Temporal cue words	*88.6			
+Dynamic class context	88.4	—	—	—

5. 特征显著性

在对系统性能的分析中，估计所用的各种特征集的相对贡献是有用的。表 4-9 显示了各种特征组合对使用树库分析树在训练集和测试集上对所有 PropBank 论元进行论元分类的准确率。

在表 4-9 的上半部分，我们看到，在一次去掉一个特征后，性能下降。特征按照显著性增加的顺序进行排序。去掉所有与中心词相关的信息，性能将显著降低。表 4-9 的下半部分显示了一些特征组合的表现。表 4-10 显示在论元识别中特征的显著性。对于论元分类，去掉路径对性能有最小的影响，而在论元识别中，去掉路径信息将导致 SVM 训练的收敛速度变慢，并且对性能有最不利的影响。

表 4-9　各种特征组合在论元分类任务的性能

特　征	准 确 率
所有特征 [120]	91.0
除了路径外的所有特征	90.8
除了短语类型外的所有特征	90.8
除了中心词及其词性外的所有特征	90.7
除了所有短语外的所有特征	*83.6
除了词外的所有特征	*82.4
除了中心词、并来词和 LW 信息外的所有特征	*75.1
只用路径和词	74.6
只用路径和短语类型	47.2
只用中心词	37.7
只用路径	28.0

表 4-10 各种特征组合在论元识别任务的性能

特 征	精 确 率	召 回 率	F1
所有特征 [120]	93.2	92.5	93.8
除了中心词外的所有特征	93.3	92.3	93.7
除了谓词外的所有特征	94.3	91.9	93.2
除了中心词、外来词和 LW 向量外的所有特征	91.8	88.5	*90.1
除了路径和部分路径外的所有特征	88.4	88.9	*88.6
只使用路径和中心词	86.5	84.5	86.3
只使用路径和谓词	89.3	81.2	85.1

6. 特征选择

事实上，加入命名实体识别特征对空或非空分类器的性能产生了不利影响，而相同的特征集显示论元分类任务的识别性能有了显著的改善，这表明特征选择策略是有益的。一种策略是进行去一法（leave-one-out）实验，即每次我们从总的特征集中分别去掉一个特征，根据性能下降的程度来决定是保留还是删掉该特征。这是一种较为简单的特征选择策略，它假设特征之间是相互独立的。我们可以使用更复杂的特征选择策略。如采用 SVM 分类，根据每个论元类型选择特征的一个缺点是：SVM 输出的是距离，而不是概率。不同分类器类距离可能是不可比的，特别是当采用不同的特征来训练一个二元分类器时。一种解决方法是使用 Platt [152] 提出的算法通过拟合 sigmoid 函数将 SVM 得分转化为概率。Foster 和 Stine [156] 展示的 Pool-Adjacent-Violators（PAV）算法 [157] 提供了一种当 Platt 算法失效时能够更好地把原始分类器的得分转变为概率的方法。转换的概率可能不准确，在这种情况下，概率可被分组，并且可以训练一个扭曲（warping）函数来进行校准。

7. 训练语料的规模

任何监督学习方法的一个很重要的问题是训练高性能分类器所需训练数据的数量。为了检验这个学习问题，Pradhan 等人 [129] 用不同数目的训练数据来训练分类器。结果如图 4-15 所示。从上往下的第一条曲线为单独进行论元识别的 F_1 值的变化曲线。第三条曲线为结合论元识别和分类任务所得到的 F_1 值的变化曲线。可以看到，从 10 000 个实例后，识别性能趋向于稳定，这表明简单地使用更多的数据不是一个好的策略。一个更好的策略是只标注适当数目的新数据。此外，事实上第一和第三条曲线（第一条是进行论元识别任务的 F 值，第三条曲线是同时进行论元分类和识别的 F 值）几乎平行，这说明对整个数据的分类存在一定的错误。解决这个问题的一种方法是找出更好的特征。

138 / 140

8. 克服句法分析错误

在进行详细的错误分析后，Pradhan 等人 [129] 发现由于识别问题，进一步提高整体系统的性能存在着瓶颈。基线系统在已知是论元的情况下进行分类的准确率可达到 90%。另一方面，该系统的识别性能稍低，只得到 80% 的召回率和 86% 的准确率。这些识别错误的原因主要有两个。一个是当语法成分已经在分析树中时，系统并不能识别出所有有语义角色的成分，还会把没有语义角色的成分误识别出来。另一个错误是句法分析器根本没有提供与正确论元相对应的成分。用 Charniak 分析器的分类性能比用树库分析树的 F 值差了 3 个点。另一方面，用 Charniak 分析器进行论元识别的性能 F 值差了 12.7 个点。错误的一半，大约 7 个点，主要是因为成分缺失，另外的大约 6 个点主要是因为分类错误。

因为采用自动句法分析器的论元识别性能的严重下降，有必要检查两种技术以改进论元识别：结合不同的句法表示的分析结果，在相同的表示中采用 n best 分析或者分析森林。

141

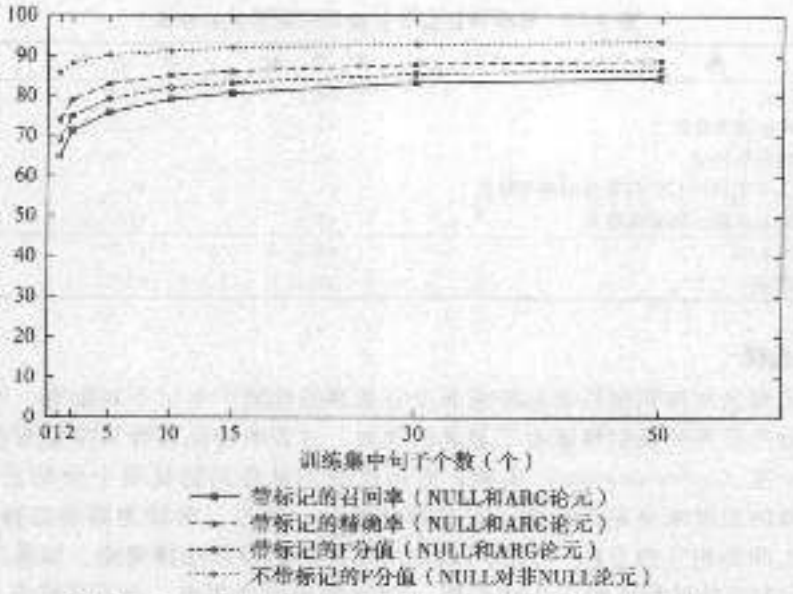


图 4-15　基于宾州树库的论元分类和论元识别任务的学习曲线

多视图　Pradhan 等人 [129] 报告了解决给定的句法分析中论元缺失问题的实验。他们探讨如何结合语义角色标注使用不同的句法视图：一种用 Charniak 分析器 [158] 进行训练；另一种用 Minipar [135] 的基于规则的依存分析树；第三种方法基于扁平的浅层句法块表示 [159]。他们发现，这三种视图互为补充，可提高性能。

我们已经讨论过的一些系统使用的特征是基于语法分析器产生的句法成分 [149, 126]，其他只使用句法分块程序所产生的扁平的语法表示 [160, 159, 124]。后一种方法缺少由层次句法结构所提供的信息，而前一种方法增加了一个约束，即可能的候选角色必须是在语法树中已经出现的节点中的一个。虽然基于块的系统是高效和鲁棒的，但使用基于完整语法分析特征的系统一般更准确。对于基于分析树成分系统的错误分析表明，分析错误是系统错误的主要原因。句法分析器经常没有产生对应于语义论元正确片段的任何成分。Pradhan 等 [129] 第一次尝试通过合并从不同的句法分析树产生的语义角色标注来处理这个问题。这种方法基于的假设是不同的句法分析器可能产生不同的错误，结合它们的输出将可能比任何单个系统都有所提高。这个初步的尝试使用的特征来自于 Charniak 分析器、Minipar 分析器和基于块的分析器。它显示出这些结合确实带来了一些提高，但是合并信息的方法是启发式和次优的。研究者提出了一种结合不同句法视图的改进框架，目标是保持基于短语的分块程序切分的鲁棒性和灵活性，同时能够利用完整语法分析树中的特征。他们也想把从不同的句法分析中得到的特征进行组合以得到额外的鲁棒性。为此，他们使用从 Charniak 分析器和 Collins 分析器得到的特征。结合基于 Minipar 和基于 Charniak 的语义角色标注的好处是能够显著提高 ARG1 的性能，而其他的一些论元略有改善，参见图 4-15。

语义分析的合并方式如下：论元的得分转换为校准的概率，得分低于阈值的论元将被删除。每个语义角色标注使用独自的阈值。对于剩下的论元，如果任何一组论元重叠，概率最低的将被删除直到没有重叠为止。在基于块的系统中，一个论元可能包括若干块，赋给论元的 BEGIN 标记的概率被用作形成论元的块序列的概率。

整体框架是对每个句法分析树视图训练独立的语义角色标注系统，然后用这些系统输

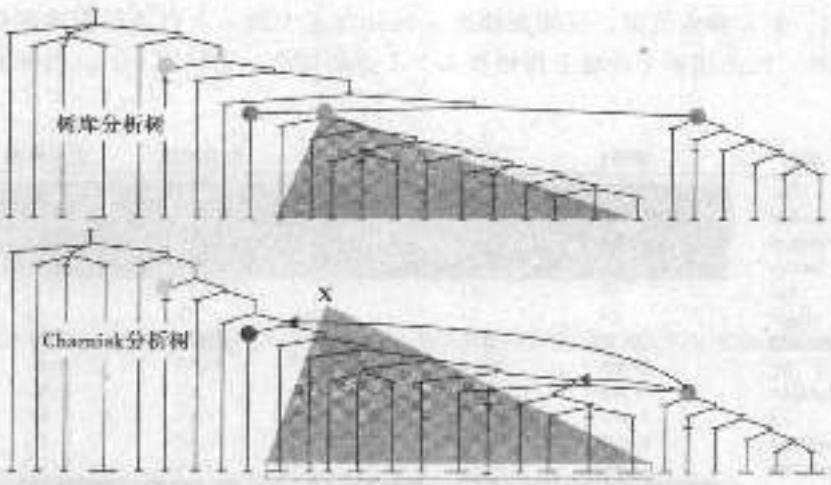


图 4-16 分析错误导致的论元删除

出的论元角色作为使用扁平句法视图的语义角色分类器的额外特征。基于成分的分类器遍历语法分析树并为每个节点分类为空（没有角色）或者其中的一个语义角色。正如我们在4.5.2节所看到的，基于块的系统使用IOB表示基本短语。成分级的角色被映射到分块程序的IOB表示。然后这些IOB标记作为另外的基本短语语义角色标注程序（分块程序）的特征，同时作为分块程序使用的标准特征集的补充。n倍交叉验证用来训练基于成分的 142
角色分类器和基于块的分类器，参见图4-17。

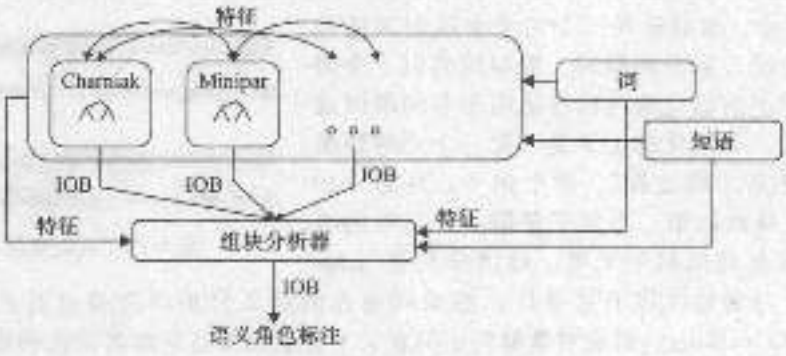


图 4-17 新的体系结构

结合所有特征的基于块的系统进行了4次迭代训练。每一次训练独立的SVM分类器，使用75%的训练数据。剩下的25%的训练数据被每一个系统标注，该过程迭代4次，得到分块程序的训练集。当分块程序训练完成，基于PSG和Minipar的语义角色标注系统用所有的数据重新训练。一旦完成再训练，SVM为所有的论元训练begin（B）和inside（I）类，以及onside（O）类。如图4-18所示，这种结构的一个特别的优点是，得到的最终切分并不一定和输入切分一致。根据提供的与特征相关的信息，分类器能够产生新的、更好的切分。 143

这是一种结合多视图的方法。另一个由Surdeanu等[161]提出的组合策略也显示了比用单一视图的性能改进。

扩大搜索范围 另一种方法是用n-best分类器[153, 154]选择成分或者通过压缩森

林表示［162］扩大搜索范围，压缩森林比 n-best 在更大的 n 上可表示更多的变化。通过使用分析森林，比使用单个最佳分析树有 1.2 个点的提高，比使用 n-best 分析树有 0.5 的提高。



图 6.18　用新的体系结构进行分类的例子

9. 名词论元

到现在为止，我们只对一个句子中动词谓语的论元进行了识别和分类。要产生一个句子级的语义表示，有必要在一个句子中找到其他可能为谓词的论元，如名词谓词、形容词谓词、介词谓词。本章讨论的语义角色标注适用于名词谓词或名词化的谓词。名词化的定义是"把一个动词转换成一个抽象化名词的过程"。举个例子，在图 4.19 的句子对中，每对的第二个句子是第一个句子的名词化。注意在名词化的句子里，动词分别是 made 和 took。一个分析这些句子并寻找这些动词论元的语义分析器会错过真正的事件，即 complaining 和 walking，而这对理解句子的意义十分重要并且是由名词化的谓词 complain 和 walk 分别表示。

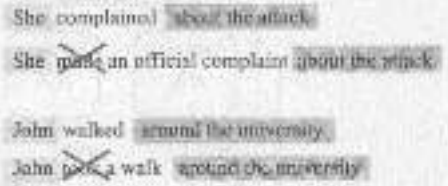


图 4-19　名词化的例子

大量有关自动语义角色标注的文献中，很少涉及名词的语义角色标注，仅有的那么几篇，也只处理名词化。然而，因为缺少带有名词谓词和相应论元的标注语料，对能够自动识别并标注名词论元的统计算法进行的研究很少。据我们所知，只有 Hull 与 Gomez［163］给出的基于规则的系统及 Lapata［164］在解释转换名词与其周边修饰名词关系的工作与之比较接近。随着 FrameNet 项目已经提供的手工标注名词化谓词的论元信息的数据，采用该数据进行自动识别名词化谓词的可行性实验可以做了。

在本节中，我们将讨论来源于动词的特征对于名词的适应性。大部分适应是直接的。我们还研究了利用这些转换的特征识别名词论元的语义属性的性能。换句话说，我们研究了这些特征在名词论元识别和分类上有多大作用，进一步，是否存在与名词化相关的新特征集合，且有较好的效果呢？

以下是 Pradhan 等人 [165] 给出的一些新特征，也给出了理由。其中的一些特征对于一些成分是不存在的。在这种情况下，相关的特征值被置为 UNK。几乎所有我们为动词谓词所使用的新特征，除了 CCG 特征，被加入到了基线系统。

中介动词特征——支持动词在识别名词谓词论元中起重要作用。使用 3 类中介动词：1) be 动词；2) 给动词（一个很小的动词集合，如 make、take、have）；3) 其他词性以 VB 开始的动词。对每一类增加 3 个特征：1) 说明在谓词和论元之间动词存在性的一个二元特征；2) 词本身作为一个特征；3) 在分析树上，成分到动词的路径。下面的例子显示了在这些中介动词特征后面的直觉：

[Speaker Leapor] makes general [Predicate assertions] [Topic about marriage]

谓词的 NP 扩展规则——这是 Gildea 和 Jurafsky [113] 提出的动词次范畴化特征的名词等价物。它代表了语法分析器为树中最低的 NP（包含谓词）实例化的扩展规则。该特征可以把具有相似内部结构的名词短语聚类到一起，从而有助于找到论元修饰符。

谓词单复数——这个二值特征说明这个谓词是单数或复数，因为单数或复数往往有不同的论元选择属性。

成分是否包含所有格——这是一个二值特征。如果在成分中有一个所有格的单词（具有词性 POS、PRP、PRP$ 或 WP$ 之一），因为这些对名词论元而言，往往是主语或宾语标记。以下的例子可以用于说明这个概念：

[Speaker Burma's] [Possessum oil] [Predicate search] hits virgin forests

支配谓词的动词——谓词的第一个 VP 祖先的中心词。

最近，Jiang 和 Ng [165] 使用这些特征在最大熵的分类器中对 NomBank 语料进行论元标注。另外，在最近的 CoNLL 评测中 [137, 138]，NomBank 论元被加到集成的句法语义依存树中。

10. 多语言问题

由于早期的研究语义角色标注的系统主要是在英语语料库中进行，各种核心特征和学习机制主要是针对英语。大部分针对英语的核心特征能够较好地转到其他语言 [167, 168]。但一些专门特征对特定的语言是重要的，同时也能够提高英语系统的性能。比如 Xue 和 Palmer [127] 介绍的针对中文的谓词框架特征也能够改善英语的性能。一些特征是针对语言的，从而导致在英语中没有对应特征。这些特征对特定的语言是唯一的。例如中文需要一个更复杂的分词过程——而英语用一些非常简单的算法就可进行。因此，在中文的情况下必须训练特殊的分词模型，才可能开始进行句法分析或语义角色标注。

另一方面，缺乏形态的中文模糊了动词、名词、形容词的区别，使这些谓词和它们论元的形成了更紧密的联系。这使得对所有类型的谓词要训练一个统一的模型。然而，中文另一个影响自动语义角色标注性能的特点是：中文比英语有更多的动词类型——至少多 4 倍。因此在相同大小的语料库中，每个动词实例的数目对中文而言少得多。这加重了已经存在的严重的数据稀疏问题。同时，这意味着有更少的多义性需要处理，从性能角度而言这倒是个不错的性质。创建一个特定的聚类特征，在一定程度上克服了这个问题。所以，在某种意义上说，虽然一组相似的特征对于不同的语言都有用，但在一些具体的实例中，可以有很大的不同，每个特征的相对好处随着语言而不同。Xue [167] 介绍了一些新特征以提高中文语义角色标注系统的性能。另一个要注意的问题是中文的句法分析器的性能比英语差，所以基于块的浅层语法分块 [169] 的结果与基于完整语义分析的结果相比也有竞争力。

146

与中文相反，阿拉伯语的特点是其有丰富的形态。这意味着在分析树中，阿拉伯语有许多语法 POS 类，这比英语或者中文几乎多一个数量级。到目前为止，在阿拉伯语的语
[146] 义角色标注系统的文献报道中还没有利用其特定形态的丰富性 [168]。

与英语的另一个显著的区别是，不管是阿拉伯语还是中文都有很多隐含或省略的主语。在 Penn Treebank 上省略成分用“迹”来标记，在 PropBank 中用论元来标记。和英语不同，中文和阿拉伯语要求特殊的模型来训练识别省略的主语，才能进行谓词-论元结构的完整识别。

11. 跨体裁的鲁棒性

这些方法一个可能的缺点是，所有的训练都在同一个宾州树库上进行。当在其他的非 WSJ 语料上评测时，性能将降低。Carreras 和 Màrquez [17] 显示在 Brown 语料库中的性能比在 WSJ 测试语料库中的性能下降 10 个点的 F 值。在 WSJ 语料上进行训练和测试，语法分析器的识别性能是主要的错误来源，分类性能是相当好的。但是，Pradhan、Ward 和 Martin [172] 给出了当我们在 WSJ 语料库上进行训练，在 Brown 语料库上进行测试时，分类性能和识别性能都受到了同样程度的影响。可见需要更多的词法语义特征来弥补不同体裁的语料间的性能差距。Zapirain [173] 指出，增加选择性偏好的特征提供了一个好的词汇－语义泛化方法。

4.5.3 软件

下面列出了一些可用的语义角色标注软件包：

- ASSERT（Automatic Statistical SEmantic Role Tagger，自动统计语义角色标注器）[http://www.cemantix.org/assert.html]，在英文 PropBank 数据基础上训练的语义角色标注工具。
- C-ASSERT [http://hlt030.cse.ust.hk/research/pc-assert]，ASSERT 的中文扩展版。
- SwiRL [http://www.surdeanu.name/mihai/swirl/]，另一个在 PropBank 基础上训练的语义角色标注工具。
- Shalmaneser（A Shallow Semantic Parser，浅层语义分析器）[http://www.coli.uni-saarland.de/projects/salsa/shal/]，基于 FrameNet 数据的浅层语义分析工具链。

4.6 意义表示

我们现在转向第三专题——更深层次的语义解释，其目标是接受自然语言输入并将之转换成一个无歧义的、可供计算机进一步操作的表示。这种形式对人而言可能是不能理解的，但对机器而言却是可理解的。一种可以类比的情况是，高级语言更接近人类处理信息
[147] 的方式而低级语言代码则更适合计算机执行。虽然编译器和解释器对高级语言所写的程序强加了各种特别的句法和语义限制，但自然语言可采取的形式却没有强加这类限制。为定义领域并消除歧义，人工语言需要具有精确性；而自然语言则依靠接收者（即听者或读者）利用上下文以及一般的世界知识来实现消歧。研究人员已经花了几十年设法赋予人们如何解释、对上下文进行编码，并使用世界的知识，以便让机器能毫不费力地理解人类所能理解的东西。然而，还有很长的路要走，到目前为止已开发的技术只能用于特定的领域和问题，还不能扩展到任意领域。这通常称为**深层语义分析**（deep semantic parsing），与

包括词义消歧、语义角色标注的浅层语义分析相对。

4.6.1 资源

一些项目已创建了表示方式和相关资源，使得在本领域可以进行更多的实验。让我们来看看其中的一些资源。

1. ATIS

航空旅游信息系统（Air Travel Information System，ATIS）项目[174]被认为是最早致力于构建将自然语言转换为终端应用决策所需知识表示的系统之一，尽管该项目并不是很关注形式化知识表示。该任务中有一台机器负责对用户的自然语言查询进行转换，该查询是有关航班信息并只允许使用受限词汇。然后，该知识表示被编译为SQL查询，并用于从航班数据库中提取答案。在编码中间的语义信息时该系统使用了一个分层的框架表示。图4-20显示了一个用户查询样本及其相应的框架表示。训练语料库中包含了137个人完成的超过774个场景，共生成了超过7300个口语语句。所有的语句都已被转录，其中的2900份则被归类并使用规范参考答案进行标注[⊖]；大约600份被转换为树库[⊖]。



图4-20 一个用户查询样本及其在ATIS程序中的框架表示

2. Communicator

Communicator是ATIS的后续者，ATIS侧重于**用户发起的对话**（user initiated dialog），而Communicator则采用**混合发起对话**（mixed-initiated dialog）。所谓用户发起对话指用户提出问题，机器负责提供答案；而在混合发起对话的过程中，计算机将提供实时航空信息并帮助用户一起协商出首选行程。在此期间，程序收集了成千上万的对话。这些对话目前都可以通过语言数据联盟（Linguistic Data Consortium，LDC）获得。卡内基-梅隆大学则收集更多的数据，其中有一部分可用于研究[⊖]。大约有一百万个词，其中约1600个对话标注了对话行为[⊖]。

[168]

3. GeoQuery

在美国地理学领域，有一个被称为Geobase[175]的地理数据库的自然语言接口（Natural Language Interface，NLI）。该接口包含了大约800条Prolog事实。这些事实与诸如人口、邻国、主要河流及国内各大城市等的地理信息一起存放在关系数据库中。下面是一些查询样本及其知识表示：

1) What is the capital of the state with the largest population?

```
answer (C, (capital (S, C), largest (P, (state (S), population (S, P)))))
```

⊖ http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S26。

⊖ http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC95T26。

⊖ http://www.speech.cs.cmu.edu/Communicator/Corpus/。

⊖ http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2002S56。

2) What are the major cities in Kansas?

```
answer (C, (major (C), city (C), loc (C, S), equal (S, stateid (kansas))))
```

这就是 GeoQuery 语料，也已被翻译成日语、西班牙语和土耳其语。

4. 机器人世界杯：CLang

机器人世界杯（RoboCup，www.robocup.org）是由人工智能学界倡导的国际赛事，以机器人足球作为其领域。它有一个专门的形式语言——CLang，该语言用于对教练的意见进行编码，使用 if-then 规则来表示行为。下面是本领域的一个范例表示：

If the ball is in our penalty area, all our players except player 4 should stay in our half.

```
((bpos (penalty-area our)) (do (player except our 4) (pos (half our))))
```

4.6.2　系统

149 正如我们可以看到的，根据终端应用的不同，上述这些例子中的意思表示可能是 SQL 查询、Prolog 查询或特定领域的查询表示。现在我们来看看将自然语言映射到这类意义表示的各种解决方法。

1. 基于规则的方法

ATIS 和 Communicator 项目中表现非常出色的部分语义分析系统是基于规则的系统，为了能更鲁棒地应对语音识别错误，他们使用一个基于手工制作语义文法的解释器。其基本理念是，句子的传统句法解释比其内含的语义信息复杂得多，因此将句子中的意义单元直接分析为语义结构被证明是一个更好的方法。此外，在处理自然语音时，系统则不得不考虑非语法的标志、口吃、停顿等现象，此时，词序变得不那么重要，散落在句子或话语中的意义单位无须按其句法意义的次序排列。Ward [176, 177, 178] 的系统 Phoenix 则使用递归转移网络（Recursive Transition Network，RTN）[179] 和一个手工制作的语法来提取层次框架结构，并依据每个新获得的信息片断重新评估和调整这些框架的值。该系统对自然语音输入的错误率为 13.2%（语音识别词错误率为 4.4%），而对录音脚本输入的错误率则为 9.3%。

2. 有监督的方法

虽然基于规则的技术开始是比较容易构建，也较好地为各种任务目标定制了解决方案。但它们仍有几个缺点：1）需要一些前期的努力创建规则；2）时间以及书写规则的特殊需求通常将所开发的系统限制到特定领域；3）当问题较复杂且与领域无关时，这类系统就很难维护和扩展；4）它们往往很脆弱。另一种方法是使用由手工标注数据训练而得的统计模型。然而，统计模型不能被用来处理未知现象，除非可以得到一些手工注释的数据。对 ATIS 进行评估时，创建了一些包含语义信息的手工标注数据。Schwartz 等人 [180] 以此为契机，创建了也许是第一个针对 ATIS 领域的端到端（end-to-end）有监督统计学习系统。他们的系统有 4 个组成部分：1）语义分析；2）语义框架；3）语篇；4）后端。该系统使用有监督的学习方法，为提高监督系统性能，该系统结合了一种快速的训练数据扩充方式——人工干预校验以生成质量稍差但数量更大的数据。Miller 等人 [181] 更详细地描述了该算法。他们的系统对整个测试集的错误率是 14.5%，而在上下文无关的句子子集上的错误率则是 9.5%。此后，还有很多对该模型的改进，比如 He 和 Young [182]。

Zelle 和 Mooney [183] 延续了现在通常所谓数据库自然语言接口（Natural Language
150 Interface for Databases，NLIDB）的研究，将自然语言形式的 GeoQuery 领域问题转换为

Prolog 查询并在 Prolog 数据库中检索答案。他们介绍了一个称为 CHILL（Constructive Heuristics Induction for Language Learning，语言学习的建构性启发式归纳）的系统，该系统基于归纳逻辑编程语言的关系学习技术，采用移进-归约分析器将输入句子映射为 Prolog 程序形式的分析表示。他们首选的语义表示是形式逻辑而不是 SQL，因为一旦获得该语义表示，就可以很容易地将它翻译成其他等价表示形式。他们在不同数量的查询上测试了系统的性能并与名为 GeoQuery 的基于规则的系统对比。这里，GeoQuery 是和 Geobase 一同发布的系统。当使用大约 175 条查询进行训练时，CHILL 的性能就和 Geobase 系统相当。而当增加更多的查询时该系统就超越了 Geobase。在对新查询进行测试时准确度达到了 84%，有时会归纳出 1100 行 Prolog 代码。

从那时起，机器学习和句法分析都有了进展。研究人员确定了新的方法也细化了现有方法。例如，SCISSOR（集成语法和语义的语义组合以获得最佳表示，Semantic Composition that Integrates Syntax and Semantics to get Optimal Representation）系统使用统计句法分析器来创建语义增强的分析树（Semantically Augmented Parse Tree，SAPT）[184，185]。SCISSOR 的训练包含 3 项（自然语言、SAPT、意义表示），使用了标准语法分析器，并用语义标签进行增强。该系统接着使用递归过程来实现对树中每个节点意义表示的构建，构造过程中则利用其子节点信息。SCISSOR 系统较之前的方法显示出了显著的性能提升。KRISP（基于核的鲁棒解释语义分析，Kernel based Robust Interpretation for Semantic Parsing）[186] 使用字符串核和支持向量机来改善底层学习技术。WASP（基于词对齐的语义分析，Word Alignment-based Semantic Parsing）[187] 将一种激进的方法引入语义分析——它使用最先进的机器翻译技术来学习语义分析器。Wong 和 Mooney 将意义表示语言看成是自然语言的转化形式并用 GIZA++ 来生成自然语言和意义表示语言间的对齐，最后使用同步 CFG（Synchronous CFG，SCFG）框架来将这些对齐的串组合成完整的意义表示。SCISSOR 比 WASP 和 KRISP 更加准确一些，它们自己也从 SAPT 中的信息获益 [188]。KRISP、CHILL 和 WASP 也都用于学习西班牙语、土耳其语和日语等语言的意义分析器，并具有类似的精度。还有另外一种方法是由 Zettlemoyer 和 Collins 所提出的 [189]，他们通过学习概率组合范畴语法（Probabilistic Combinatory Categorial Grammar，PCCG）训练了一个用于自然语言接口的结构分类器。分类器基于对数线性模型，该模型用于计算在给定自然语言输入的条件下获得该句法和语义分析的概率分布。

4.6.3 软件

早期基于规则的系统并没有很多可用的软件程序，下面是一些可供下载的软件：

- WASP [http://www.cs.utexas.edu/~ml/wasp/]。
- KRISPER [http://www.cs.utexas.edu/~ml/krisp/]。
- CHILL [http://www.cs.utexas.edu/~ml/chill.html]。

4.7 总结

本章我们通过各种不同的视角考察了语义分析问题。对于意义表示和语言理解目前还没有捷径，所以，多年来，研究人员所处理的任务，要么是在领域相关的情况下解决更大问题中的一部分，要么是在非常受限的领域中解决完整问题。上述第一种情形是浅层语义解释，处理了语言问题的 4 个主要方面：结构歧义（这本质上是语法问题，因而是单独一章的主题）、词义、实体和事件识别以及谓词-论元结构识别，其中，后三个组件已被广泛

称为浅层语义分析。正如我们已经看到的，此过程中语法起着非常重要的作用，并不能完全与语义脱离。第二种情形是深度分析，或称语义分析，包括输入自然语言以及将输入的自然语言转换成某个意思表示，该表示往往针对特定任务并能让最终应用无歧义地执行。

我们了解到，所有这些方法的各种前沿都取得了进展。在该领域的早期时代，很少有手工标记语料库和成熟的学习技巧。即使是现在，针对资源贫乏的语言，仍然没有足够的数据来训练先进的学习算法。在这种情况下，研究人员只能诉诸于将领域信息编码入规则系统，通常这种系统只运用于特定领域。对于有足够人工标注数据的语言，更多的统计方法会成为主导。考虑到即使有足够的标注，数据还是很稀疏（要学习语言所有的细微差别，任何数量的人工标注都不是足够的），研究人员纷纷使用半监督或无监督的方法，后者比起有监督的方法或基于规则方法而言通常都是不太准确的。

4.7.1 词义消歧

词义消歧是语言理解的一个组成部分，由于受使用受限的词义或已包含了隐式的消歧，在信息检索、语音理解以及受限领域的应用中，词义消歧模块并不是很重要。然而，对于处理文本深入理解的应用，词义消歧可能还是至关重要的。这方面的研究一开始使用字典中定义的词义，因为字典是一开始的主要资源。一般认为，Lesk算法是第一个基于字典的词义消歧算法，其消歧过程依赖于对给定单词的语篇上下文与其字典注释之间重叠程度的计算。Roget辞典的建立引导了更多英语专用算法按其中所定义的类别来为词语分类。“一个语篇一个词义”的概念引出了一个重要的半监督算法：Yarowsky算法。随着类似WordNet的更丰富的词典以及和基于其词义的标注语料库（SEMCOR）的出现——有趣的是，与机器学习的进展同时——多数研究人员开始转向将它们用作标准，直到后来的研究表明WordNet词义的粒度可能过于精细了。如果连
[252] 人都不能在一定程度上认同词义上的区别，则更不用指望机器能做到了。这导致WordNet词义被合并成更粗糙的单位，以便更适合生成一致的人类标注，这同时也提供了更好的、实现高精度自动消歧的方法。WordNet仍然还是本领域的重要资源，该资源显著地推进了本领域的发展，最先进的消歧系统还在使用它。

在另外一条发展线上，随着互联网的发展及诸如维基百科等资源的广泛可用（维基百科已成为一种代替用标注资源），利用互联网资源成为了主流的追求之一。越来越多的语言理解领域都采用新颖的方式利用这种资源。主动学习是另一个发展方向。虽然目前可能更像是一门艺术而不是科学，但它对于积累词讲标注一直是非常有用的，这些词要么是罕见（低频）的且高词义困惑度（多义）的，要么是由于这样或那样原因没有足够的标注（可能的原因是贫乏资源语言，但原因不限于此[190]）。对于没有手工标注数据的语言，各种无监督的方法被开发出来，其中一些可利用不同的词义粒度和跨平行语料的实例。

4.7.2 谓词-论元结构

和词义消歧不同，标注文本中话题角色的系统很少是基于规则的。随着诸如FrameNet和PropBank等标注了谓词-论元结构的语料库的出现，促使一个巨大的研究浪潮聚焦到构建在文本中标注这些结构的系统，主要用于动词和名词谓词。在各种句法框架下引入了许多新的特征，其中一些甚至不需完整句法分析而只诉诸于基本短语组块（base phrase chunk），事实证明，对于存在树库的体裁，语法分析还是能起到很大作用的。词汇化是和语义结合较好的句法表示，但基于该表示方式的语义角色标注器往往会犯原先使用

自底向上的方法可避免的错误。此外，在第一轮就使用丰富的特征代价太高，因此，先生成 n-best 结果再使用更全局的特征集对之进行重排序（reranking）的组合方法通常性能更好。另外，自顶向下和自底向上相结合的方法由于能同时整合各种句法和非句法信息，因而也能提高性能。目前一个大的瓶颈问题是，当训练与测试语料在文本体裁和/或有些不同（语法风格、词的用法或实体与事件结构等的差异）时，系统性能往往比训练和测试语料匹配时要差很多。目前的状态是，句法信息已被利用并显著地有利于语义分析，但词汇和词义级的泛化还严重缺乏，从而使现有的方法对跨体裁或跨领域的文本鲁棒性较低。我们也看到针对英语（恰好是手工标记的语料库首先创建的语言）开发的基础技术很好地转移到了其他语言。当然，每一个新的语言都有自己的特质并会导致一些新特征的定义，这些新特征可能反过来改善了原有的英文系统。许多标注工作正在全世界范围内开展，我们还有很多东西要学。

4.7.3 意义表示

最后，我们考察了意思表示问题。这是一个较少研究的课题，尤其针对跨语言的情 153
形。意义表示是一个转换过程，它将自然语言输入转换为一种无歧义且容易由机器或终端应用理解的格式，机器或终端应用则可以利用该输入执行某操作。到目前为止，没有一个通用的表示方式，因此，这些系统以及它们所采用的表示方式往往是面向特定领域的。

新的研究计划将不断拓展现有技术的可能性，并创造出一些新技术。有朝一日，我们可以利用这些技术形成一个更丰富、更深层次且领域无关的意义表示形式。

参考文献

[1] N. Chomsky, *Syntactic Structures*. The Hague: Mouton, 1957.

[2] J. Katz and J. Fodor, "The structure of a semantic theory," *Language*, vol. 39, pp. 170–210, 1963.

[3] V. H. Yngve, "Syntax and the problem of multiple meaning," in *Machine Translation of Languages* (W. N. Locke and A. D. Booth, eds.), pp. 208–226, Cambridge, MA: MIT Press, 1955.

[4] N. Ide and J. Véronis, "Introduction to the special issue on word sense disambiguation: The state of the art," *Computational Linguistics*, vol. 24, no. 1, pp. 2–40, 1998.

[5] E. Agirre and P. Edmonds, eds., *Word Sense Disambiguation: Algorithms and Applications*. Dordrecht: Springer, 2006.

[6] M. Palmer, H. Dang, and C. Fellbaum, "Making coarse-grained and fine-grained sense distinctions, both manually and automatically," *Natural Language Engineering Journal*, vol. 13, no. 2, pp. 137–163, 2007.

[7] P. Resnik and D. Yarowsky, "Distinguishing systems and distinguishing senses: New evaluation methods for word sense disambiguation," *Journal of Natural Language Engineering*, vol. 5, no. 2, pp. 113–133, 1999.

[8] R. Krovetz and W. B. Croft, "Lexical ambiguity and information retrieval," *ACM Transactions on Information Systems*, vol. 10, no. 2, pp. 115–141, 1992.

[9] S. Katz, "Estimation of probabilities from sparse data for the language model component of a speech recognizer," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 35, no. 3, pp. 400–401, 1987.

[10] L. Bahl, F. Jelinek, and R. Mercer, "A maximum likelihood approach to continuous speech recognition," *PAMI—IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, no. 2, pp. 179–190, 1983.

[11] P. F. Brown, V. J. Della Pietra, P. V. deSouza, J. C. Lai, and R. L. Mercer, "Class-based n-gram models of natural language," *Computational Linguistics*, vol. 18, no. 4, pp. 467–479, 1992.

[12] W. Gale, K. W. Church, and D. Yarowsky, "Estimating upper and lower bounds on the performance of word-sense disambiguation programs," in *Proceedings of the 30th annual meeting on Association for Computational Linguistics*, pp. 249–256, 1992.

[13] G. Hirst, *Semantic Interpretation and the Resolution of Ambiguity*. Cambridge: Cambridge University Press, 1987.

[14] P. Procter, *Longman Dictionary of Contemporary English (LDOCE)*. Harlow: Longman Group, 1978.

[15] R. Chapman, *Roget's International Thesaurus*. New York: Harper and Row, 1977.

[16] G. A. Miller, "WordNet: An on-line lexical database," *International Journal of Lexicography*, vol. 3, no. 4, pp. 235–312, 1990.

[17] H. Kučera and W. N. Francis, *Computational Analysis of Present-Day American English*. Providence, RI: Brown University Press, 1967.

[18] G. A. Miller, C. Leacock, R. Tengi, and R. T. Bunker, "A semantic concordance," in *Proceedings of the Workshop on Human Language Technology*, pp. 303–308, 1993.

[19] D. I. Moldovan and V. Rus, "Logic form transformation of WordNet and its applicability to question answering," in *Proceedings of the Association for Computational Linguistics*, pp. 394–401, 2001.

[20] H. T. Ng and H. B. Lee, "Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach," in *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pp. 40–47, 1996.

[21] SIGLEX, "SENSEVAL: Evaluation Exercises for the Semantic Analysis of Text," 2011. http://www.senseval.org.

[22] R. Weischedel, E. Hovy, M. Palmer, M. Marcus, R. Belvin, S. Pradhan, L. Ramshaw, and N. Xue, "OntoNotes: A large training corpus for enhanced processing," in *Handbook of Natural Language Processing and Machine Translation* (J. Olive, C. Christianson, and J. McCary, eds.) New York: Springer, 2011.

[23] S. Pradhan, E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel, "OntoNotes: A unified relational semantic representation," *International Journal of Semantic Computing*, vol. 1, no. 4, pp. 405–419, 2007.

[24] E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel, "OntoNotes: The 90% solution," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 57–60, 2006.

[25] S. Pradhan, E. Loper, D. Dligach, and M. Palmer, "Semeval-2007 task-17: English lexical sample, srl and all words," in *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, pp. 87–92, 2007.

[26] D. Lenat, "Cyc: A large-scale investment in knowledge infrastructure," *Communications of the ACM*, vol. 38, no. 11, pp. 33–35, 1995.

[27] Z. Dong and Q. Dong, *HowNet and the Computation of Meaning*. Hackensack, NJ: World Scientific, 2006.

[28] A. D. de Ilarraza, A. Mayor, and K. Sarasola, "Semiautomatic labeling of semantic features," in *Proceedings of the International Conference on Computational Linguistics*, 2002.

[29] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *Proceedings of the 14th conference on Computational linguistics*, pp. 539–545, 1992.

[30] R. Snow, D. Jurafsky, and A. Y. Ng, "Semantic taxonomy induction from heterogenous evidence," in *ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pp. 801–808, 2006.

[31] T. Chklovski and P. Pantel, "Verbocean: Mining the web for fine-grained semantic verb relations," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2004.

[32] A. Thanopoulos, N. Fakotakis, and G. Kokkinakis, "Automatic extraction of semantic relations from specialized corpora," in *Proceedings of the International Conference on Computational Linguistics*, pp. 836–842, 2000.

[33] N. Calzolari and E. Picchi, "Acquisition of semantic information from an on-line dictionary," in *Proceedings of the 12th Conference on Computational Linguistics*, pp. 87–92, 1988.

[34] S. McRoy, "Using multiple knowledge sources for word sense disambiguation," *Computational Linguistics*, vol. 18, no. 1, pp. 1–30, 1992.

[35] M. E. Lesk, "Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone," in *Proceedings of the SIGDOC Conference*, 1986.

[36] A. Kilgarriff and J. Rosenzweig, "English framework and results," *Computers and the Humanities*, vol. 34, no. 1–2, 2000.

[37] S. Banerjee and T. Pedersen, "An adapted lesk algorithm for word sense disambiguation using wordnet," in *Proceedings of the 3rd International Conference on Computational Linguistics and Intelligent Text Processing (CICLing-02)*, pp. 136–145, 2002.

[38] D. Yarowsky, "Word-sense disambiguation using statistical models of Roget's categories trained on large corpora," in *Proceedings of the 14th Conference on Computational Linguistics (COLING-92)*, pp. 454–460, 1992.

[39] R. Navigli and P. Velardi, "Learning domain ontologies from document warehouses and dedicated web sites," *Computational Linguistics*, vol. 30, no. 2, 2004.

[40] R. Navigli and P. Velardi, "Structural semantic interconnections: A knowledge-based approach to word sense disambiguation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 7, pp. 1075–1086, 2005.

[41] B. Magnini and G. Cavaglia, "Integrating subject field codes into wordnet," in *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, 2000.

[42] S. Patwardhan, S. Banerjee, and T. Pedersen, "Using measures of semantic relatedness for word sense disambiguation," in *Proceedings of the 4th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-03)*, 2003.

[43] M. Strube and S. P. Ponzetto, "Wikirelate! Computing semantic relatedness using Wikipedia," in *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI'06)*, pp. 1419–1424, 2006.

[44] S. P. Ponzetto and M. Strube, "Knowledge derived from Wikipedia for computing semantic relatedness," *Journal of Artificial Intelligence Research*, vol. 30, pp. 181–212, 2007.

[45] R. Navigli and S. P. Ponzetto, "Babelnet: Building a very large multilingual semantic network," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 216–225, 2010.

[46] S. P. Ponzetto and R. Navigli, "Knowledge-rich word sense disambiguation rivaling supervised systems," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 1522–1531, 2010.

[47] P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer, "Word-sense disambiguation using statistical methods," in *Proceedings of the 29th annual meeting on Association for Computational Linguistics*, pp. 264–270, 1991.

[48] D. Yarowsky, "Homograph disambiguation in text-to-speech synthesis," in *Progress in Speech Synthesis* (J. Hirschberg, R. Sproat, and J. van Santen, eds.), pp. 159–175, New York: Springer, 1996.

[49] Y. K. Lee and H. T. Ng, "An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation," in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing (EMNLP '02)*, pp. 41–48, 2002.

[50] H. T. Ng, "Exemplar-based word sense disambiguation: Some recent improvements," in *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing*, pp. 208–213, 1997.

[51] J. Chen and M. S. Palmer, "Towards robust high performance word sense disambiguation of english verbs using rich linguistic features," in *Proceedings of 2nd International Joint Conference on Natural Language Processing*, pp. 933–944, 2005.

[52] D. Dligach and M. Palmer, "Novel semantic features for verb sense disambiguation," in *Proceedings of the Conference of Association of Computational Linguistics*, pp. 29–32, 2008.

[53] R. Rada, H. Mili, E. Bicknell, and M. Blettner, "Development and application of a metric on semantic nets," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 1, pp. 17–30, Jan 1989.

[54] P. Resnik, "Using information content to evaluate semantic similarity in a taxonomy," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pp. 448–453, 1995.

[55] E. Agirre and G. Rigau, "Word sense disambiguation using conceptual density," in *Proceedings of the 16th conference on Computational linguistics*, pp. 16–22, 1996.

[56] P. Resnik, "Selectional preference and sense disambiguation," in *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, 1997.

[57] P. Resnik, "Disambiguating noun groupings with respect to WordNet senses," in *Proceedings of the 3rd Workshop on Very Large Corpora*, pp. 27–38, 1995.

[58] C. Leacock, G. A. Miller, and M. Chodorow, "Using corpus statistics and WordNet relations for sense identification," *Computational Linguistics*, vol. 24, no. 1, pp. 147–165, 1998.

[59] I. Dagan and A. Itai, "Word sense disambiguation using a second-language monolingual corpus," *Computational Linguistics*, vol. 20, no. 4, 1994.

[60] M. T. Diab, "An unsupervised approach for bootstrapping Arabic sense tagging," in *Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages*, pp. 43–50, 2004.

[61] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the 33rd Annual Meeting of the ACL*, 1995.

[62] M. Galley and K. McKeown, "Improving word sense disambiguation in lexical chaining," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pp. 1486–1488, 2003.

[63] R. Mihalcea and D. I. Moldovan, "An automatic method for generating sense tagged corpora," in *Proceedings of the 16th National Conference on Artificial Intelligence and the 11th Conference on Innovative Applications of Artificial Intelligence (AAAI '99/IAAI '99)*, pp. 461–466, 1999.

[64] R. Mihalcea, "Using Wikipedia for automatic word sense disambiguation," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pp. 196–203, 2007.

[65] J. Grimshaw, *Argument Structure*. Cambridge, MA: MIT Press, 1990.

[66] M. Baker, "Thematic roles and syntactic structure," in *Elements of Grammar: Handbook of Generative Syntax* (L. Haegeman, ed.), New York: Springer, 1997.

[67] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The Berkeley FrameNet project," in *Proceedings of the International Conference on Computational Linguistics (COLING/ACL-98)*, pp. 86–90, 1998.

[68] C. J. Fillmore and C. F. Baker, "FrameNet: Frame semantics meets the corpus," poster presentation at the *74th Annual Meeting of the Linguistic Society of America*, Chicago, Jan. 6–9, 2000.

[69] C. Fillmore, C. Johnson, and M. R. L. Petruck, "Background to FrameNet," *International Journal of Lexicography*, vol. 16, no. 3, 2003. [158]

[70] C. J. Fillmore, C. Wooters, and C. F. Baker, "Building a large lexical databank which provides deep semantics," in *Proceedings of the Pacific Asian Conference on Language, Information and Computation*, 2001.

[71] M. Palmer, D. Gildea, and P. Kingsbury, "The Proposition Bank: An annotated corpus of semantic roles," *Computational Linguistics*, pp. 71–106, 2005.

[72] C. J. Fillmore, "Frame semantics," in *Linguistics in the Morning Calm*, pp. 111–138, Seoul: Hanshin; Linguistics Society of Korea, 1982.

[73] D. R. Dowty, "Thematic proto-roles and argument selection," *Language*, vol. 67, no. 3, pp. 547–619, 1991.

[74] C. Barker and D. Dowty, "Non-verbal thematic proto-roles," in *Proceedings of North Eastern Linguistics Conference (NELS-23)*, pp. 49–62, 1992.

[75] M. Marcus, G. Kim, M. A. Marcinkiewicz, R. MacIntyre, A. Bies, M. Ferguson, K. Katz, and B. Schasberger, "The Penn Treebank: Annotating predicate argument structure," 1994. http://clair.si.umich.edu/clair/anthology/query.cgi?type=Paper&id=H94-1020.

[76] O. Babko-Malaya, A. Bies, A. Taylor, S. Yi, M. Palmer, M. Marcus, S. Kulick, and L. Shen, "Issues in synchronizing the English treebank and PropBank," in *Proceedings of the Workshop on Frontiers in Linguistically Annotated Corpora 2006*, July 2006.

[77] M. Palmer, O. Babko-Malaya, and H. T. Dang, "Different sense granularities for different applications," in *Proceedings of the HLT NAACL 2004 Workshop: 2nd Workshop on Scalable Natural Language Understanding*, pp. 49–56, 2004.

[78] A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman, "The nombank project: An interim report," in *Proceedings of the NAACL/HLT Workshop on Frontiers in Corpus Annotation*, 2004.

[79] C. Macleod, R. Grishman, A. Meyers, L. Barrett, and R. Reeves, "Nomlex: A lexicon of nominalizations," http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.12.1452, 1998.

[80] B. Levin, *English Verb Classes And Alternations: A Preliminary Investigation*. Chicago: University of Chicago Press, 1993.

[81] K. Kipper, A. Korhonen, N. Ryant, and M. Palmer, "A large-scale classification of English verbs," *Language Resources and Evaluation*, vol. 42, no. 1, pp. 21–40, 2008.

[82] H. T. Dang, K. Kipper, M. Palmer, and J. Rosenzweig, "Investigating regular sense extensions based on intersective Levin classes," in *COLING/ACL-98: Proceedings of the 36th Conference on Computational Linguistics*, pp. 293–299, ACL, 1998.

[83] C. F. Baker and J. Ruppenhofer, "FrameNet's frames vs. Levin's verb classes," in *Proceedings of the 28th Annual Meeting of the Berkeley Linguistics Society*, 2002.

[84] K. Erk and S. Pado, "Towards a resource for lexical semantics: A large German corpus with extensive semantic annotation," in *Proceedings of Association for Computational Linguistics*, 2003.

[85] A. Burchardt, K. Erk, A. Frank, A. Kowalski, S. Pado, and M. Pinkal, "Using FrameNet for the semantic analysis of German: Annotation, representation, and automation," in *Multilingual FrameNets in Computational Lexicography: Methods and Applications* (H. C. Boas, ed.), New York: Mouton de Gruyter, 2009.

[86] K. H. Ohara, "Lexicon, grammar, and multilinguality in the Japanese FrameNet," in *Proceedings of the 6th International Conference on Language Resources and Evaluation*, 2008.

[87] C. Subirats, "Spanish FrameNet: A frame-semantic analysis of the Spanish lexicon," in *Multilingual FrameNets in Computational Lexicography: Methods and Applications* (H. C. Boas, ed.), pp. 135–162, Mouton de Gruyter, 2009.

[88] C. Subirats and H. Sato, "Spanish FrameNet and FrameSQL," in *Proceedings of the 4th International Conference on Language Resources and Evaluation, Workshop on Building Lexical Resources from Semantically Annotated Corpora*, 2004.

[89] L. D. Borin, F. Dana, T. G. Markus, and K. D. Maria, "The past meets the present in the Swedish FrameNet++," in *Proceedings of the 14th Euralex International Congress*, (Leeuwarden), 2010.

[90] H. C. Boas, ed., *Multilingual FrameNets in Computational Lexicography: Methods and Applications*. New York: Mouton de Gruyter, 2009.

[91] N. Xue and M. Palmer, "Adding semantic roles to the Chinese treebank," *Natural Language Engineering*, vol. 15, no. 1, pp. 143–172, 2009.

[92] M. Palmer, O. Babko-Malaya, A. Bies, M. Diab, M. Maamouri, A. Mansouri, and W. Zaghouani, "A pilot Arabic propbank," in *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, 2008.

[93] W. Zaghouani, M. Diab, A. Mansouri, S. Pradhan, and M. Palmer, "The revised Arabic propbank," in *Proceedings of the 4th Linguistic Annotation Workshop*, pp. 222–226, July 2010.

[94] M. Palmer, S. Ryu, J. Choi, S. Yoon, and Y. Jeon, "Korean propbank," 2006.

[95] M. Taulé, M. Martí, and M. Recasens, "Ancora: Multilevel annotated corpora for Catalan and Spanish," in *Proceedings of Language, Resources and Evaluation, LREC*, 2008.

[96] M. Palmer, R. Bhatt, B. Narasimhan, O. Rambow, D. M. Sharma, and F. Xia, "Hindi syntax: Annotating dependency, lexical predicate-argument structure, and phrase structure," in *Proceedings of the 7th International Conference on Natural Language Processing (ICON-2009)*, 2009.

[97] J. Hajic, B. Vidova-Hladka, and P. Pajas, "The Prague Dependency Treebank: Annotation structure and support," in *Proceedings of the IRCS Workshop on Linguistic Databases*, pp. 105–114, 2001.

[98] R. Iida, M. Komachi, K. Inui, and Y. Matsumoto, "Annotating a Japanese text corpus with predicate-argument and coreference relations," in *Proceedings of ACL Linguistic Annotation Workshop*, 2007.

[99] N. Sondheimer, R. Weischedel, and R. Bobrow, "Semantic interpretation using KL-ONE," in *Proceedings of the 10th International Conference on Computational Linguistics and 22nd Annual Meeting of the Association for Computational Linguistics*, pp. 101–107, 1984.

[100] N. Calzolari, "Acquiring and representing semantic information in a lexical knowledge base," in *Proceedings of the ACL SIGLEX Workshop on Lexical Semantics and Knowledge Representation*, pp. 188–197, 1992.

[101] R. long Liu and V. wun Soo, "An empirical study on thematic knowledge acquisition based on syntactic clues and heuristics," in *Proceedings 31st Annual Meeting of the ACL*, pp. 243–250, 1993.

[102] G. Hirst, "A foundation for semantic interpretation," in *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pp. 64–73, 1983.

[103] D. Dahl, M. Palmer, and R. Passonneau, "Nominalizations in PUNDIT," in *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, 1987.

[104] M. Palmer, C. Weir, R. Passonneau, and T. Finin, "The kernel text understanding system," *Artificial Intelligence*, vol. 63 (Special Issue on Text Understanding), pp. 17–68, October 1993.

[105] J. L. G. Rossi and E. Francozo, "Hybrid thematic role processor: Symbolic linguistic relations revised by connectionist learning," in *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 852–861, 1999.

[106] C. Ross, "A framework for robust semantic interpretation," in *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pp. 311–318, 2000.

[107] L. S. Peh and H. T. Ng, "Domain-specific semantic class disambiguation using wordnet," in *Proceedings of the ACL Workshop on Very Large Corpora*, pp. 56–65, 1997.

[108] C. D. Manning, "Automatic acquisition of a large subcategorization dictionary from corpora," in *Proceedings of the 31st Meeting of the Association for Computational Linguistics*, pp. 235–242, 1993.

[109] T. Briscoe and J. Carroll, "Automatic extraction of subcategorization from corpora," in *Proceedings of the 5th Conference on Applied Natural Language Processing*, March 31 – April 3 1997.

[110] J. Pustejovsky, "The acquisition of lexical semantic knowledge from large corpora," in *Proceedings of Speech and Natural Language Workshop*, pp. 311–315, 1992.

[111] R. Green, B. J. Dorr, and P. Resnik, "Inducing frame semantic verb classes from WordNet and LDOCE," in *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pp. 375–382, 2004.

[112] R. S. Swier and S. Stevenson, "Unsupervised semantic role labelling," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 95–102, 2004.

[113] D. Gildea and D. Jurafsky, "Automatic labeling of semantic roles," *Computational Linguistics*, vol. 28, no. 3, pp. 245–288, 2002.

[114] D. Magerman, "Natural language parsing as statistical pattern recognition," PhD thesis, Stanford University, 1994.

[115] M. J. Collins, "Head-driven statistical models for natural language parsing," PhD thesis, University of Pennsylvania, Philadelphia, 1999.

[116] D. Lin, "Automatic retrieval and clustering of similar words," in *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association of Computational Linguistics (COLING/ACL)*, 1998.

[117] T. Hofmann and J. Puzicha, "Statistical models for co-occurrence data" (memo), Massachusetts Institute of Technology Artificial Intelligence Laboratory, Feb. 1998.

[118] M. Surdeanu, S. Harabagiu, J. Williams, and P. Aarseth, "Using predicate-argument structures for information extraction," in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2003.

[119] M. Fleischman, N. Kwon, and E. Hovy, "Maximum entropy models for framenet classification," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2003.

[120] S. Pradhan, K. Hacioglu, V. Krugler, W. Ward, J. Martin, and D. Jurafsky, "Support vector learning for semantic argument classification," *Machine Learning Journal*, vol. 60, no. 1, pp. 11–39, 2005.

[121] D. M. Bikel, R. Schwartz, and R. M. Weischedel, "An algorithm that learns what's in a name," *Machine Learning*, vol. 34, pp. 211–231, 1999.

[122] R. Girju, D. Roth, and M. Sammons, "Token-level disambiguation of VerbNet classes," in *Proceedings of the Interdisciplinary Workshop on the Identification and Representation of Verb Features and Verb Classes*, 2005.

[123] K. Daniel, Y. Schabes, M. Zaidel, and D. Egedi, "A freely available wide coverage morphological analyzer for English," in *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, 1992.

[124] K. Hacioglu, S. Pradhan, W. Ward, J. Martin, and D. Jurafsky, "Semantic role labeling by tagging syntactic chunks," in *Proceedings of the 8th Conference on Computational Natural Language Learning (CoNLL)*, May 2004.

[125] D. Vickrey and D. Koller, "Applying sentence simplification to the CoNLL-2008 shared task," in *Proceedings of the 12th Conference on Computational Natural Language Learning (CoNLL 2008)*, pp. 268–272, 2008.

[126] S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky, "Shallow semantic parsing using support vector machines," in *Proceedings of the Human Language Technology Conference/North American Chapter of the Association of Computational Linguistics (HLT/NAACL)*, 2004.

102

[127] N. Xue and M. Palmer, "Calibrating features for semantic role labeling," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2004.

[128] D. Gildea and J. Hockenmaier, "Identifying semantic roles using combinatory categorial grammar," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2003.

[129] S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky, "Semantic role labeling using different syntactic views," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2005.

[130] J. Chen and O. Rambow, "Use of deep linguistic features for the recognition and labeling of semantic arguments," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2003.

[131] D. Gildea and M. Palmer, "The necessity of syntactic parsing for predicate argument recognition," in *Proceedings of the 40th Annual Conference of the Association for Computational Linguistics (ACL-02)*, 2002.

[132] A. Moschitti, D. Pighin, and R. Basili, "Tree kernels for semantic role labeling," *Computational Linguistics*, vol. 34, no. 2, 2008.

[133] K. Hacioglu, "Semantic role labeling using dependency trees," in *Proceedings of Coling 2004*, pp. 1273–1276, 2004.

[134] R. Hwa, A. Lopez, and M. Diab, "engconst2dep program for converting Treebank trees to dependency trees," 2011.

[135] D. Lin, "Dependency-based evaluation of MINIPAR," in *Workshop on the Evaluation of Parsing Systems*, 1998.

[136] D. Lin and P. Pantel, "Discovery of inference rules for question answering," *Natural Language Engineering*, vol. 7, no. 4, pp. 343–360, 2001.

[137] M. Surdeanu, R. Johansson, A. Meyers, L. Màrquez, and J. Nivre, "The CoNLL 2008 shared task on joint parsing of syntactic and semantic dependencies," in *CoNLL 2008: Proceedings of the 12th Conference on Computational Natural Language Learning*, pp. 159–177, 2008.

[138] J. Hajič, M. Ciaramita, R. Johansson, D. Kawahara, M. A. Martí, L. Màrquez, A. Meyers, J. Nivre, S. Padó, J. Štěpánek, P. Straňák, M. Surdeanu, N. Xue, and Y. Zhang, "The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages," in *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pp. 1–18, 2009.

[139] R. Johansson and P. Nugues, "Extended constituent-to-dependency conversion for English," in *Proceedings of 16th Nordic Conference on Computational Linguistics (NODALIDA)*, pp. 105–112, 2007.

[140] V. Punyakanok, D. Roth, and W. tau Yih, "The necessity of syntactic parsing for semantic role labeling," in *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.

[141] L. A. Ramshaw and M. P. Marcus, "Text chunking using transformation-based learning," in *Proceedings of the 3rd Annual Workshop on Very Large Corpora*, pp. 82–94, 1995.

[142] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.

[143] R. Quinlan, "Data Mining Tools See5 and C5.0," 2003. http://www.rulequest.com.

[144] M. Fleischman and E. Hovy, "A maximum entropy approach to framenet tagging," in *Proceedings of the Human Language Technology Conference*, 2003.

[145] T. Kudo and Y. Matsumoto, "Use of support vector learning for chunk identification," in *Proceedings of the 4th Conference on Computational Natural Language Learning (CoNLL)*, 2000.

[146] T. Kudo and Y. Matsumoto, "Chunking with support vector machines," in *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2001.

[147] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proceedings of the European Conference on Machine Learning (ECML)*, 1998.

[148] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "Text classification using string kernels," *Journal of Machine Learning Research*, vol. 2, no. Feb, pp. 419–444, 2002.

[149] S. Pradhan, K. Hacioglu, W. Ward, J. Martin, and D. Jurafsky, "Semantic role parsing: Adding semantic structure to unstructured text," in *Proceedings of the International Conference on Data Mining (ICDM 2003)*, 2003.

[150] E. L. Allwein, R. E. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifiers," in *Proceedings of the 17th International Conference on Machine Learning*, pp. 9–16, 2000.

[151] U. H. G. Kressel, "Pairwise classification and support vector machines," in *Advances in Kernel Methods* (B. Scholkopf, C. Burges, and A. J. Smola, eds.), Cambridge, MA: MIT Press, 1999.

[152] J. Platt, "Probabilities for support vector machines," in *Advances in Large Margin Classifiers* (A. Smola, P. Bartlett, B. Scholkopf, and D. Schuurmans, eds.), Cambridge, MA: MIT press, 2000.

[153] K. Toutanova, A. Haghighi, and C. D. Manning, "A global joint model for semantic role labeling," *Computational Linguistics*, vol. 34, no. 2, 2008.

[154] V. Punyakanok, D. Roth, and W. tau Yih, "The importance of syntactic parsing and inference in semantic role labeling," *Computational Linguistics*, vol. 34, no. 2, 2008.

[155] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, "Feature selection for svms," *Advances in Neural Information Processing Systems (NIPS)*, vol. 13, pp. 668–674, 2001.

[156] D. P. Foster and R. A. Stine, "Variable selection in data mining: Building a predictive model for bankruptcy," *Journal of American Statistical Association*, vol. 99, pp. 303–313, 2004.

[157] R. E. Barlow, D. J. Bartholomew, J. M. Bremner, and H. D. Brunk, *Statistical Inference under Order Restrictions*. New York: Wiley, 1972.

[158] E. Charniak, "A maximum-entropy-inspired parser," in *Proceedings of the 1st Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pp. 132–139, 2000.

[159] K. Hacioglu, "A lightweight semantic chunking model based on tagging," in *Proceedings of the Human Language Technology Conference/North American Chapter of the Association of Computational Linguistics (HLT/NAACL)*, 2004.

[160] K. Hacioglu and W. Ward, "Target word detection and semantic role chunking using support vector machines," in *Proceedings of the Human Language Technology Conference*, 2003.

[161] M. Surdeanu, L. Màrquez, X. Carreras, and P. R. Comas, "Combination strategies for semantic role labeling," *Journal of Artificial Intelligence Research*, vol. 29, pp. 105–151, 2007.

[162] X. Hao, H. Mi, Y. Liu, and Q. Liu, "Forest-based semantic role labeling," in *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI) Conference*, 2010.

[163] R. D. Hull and F. Gomez, "Semantic interpretation of nominalizations," in *Proceedings of the 13th National Conference on Artificial Intelligence*, pp. 1062–1068, 1996.

[164] M. Lapata, "The disambiguation of nominalizations," *Computational Linguistics*, vol. 28, no. 3, pp. 357–388, 2002.

[165] S. Pradhan, H. Sun, W. Ward, J. Martin, and D. Jurafsky, "Parsing arguments of nominalizations in English and Chinese," in *Proceedings of the Human Language Technology Conference/North American Chapter of the Association of Computational Linguistics (HLT/NAACL)*, 2004.

[166] Z. P. Jiang and H. T. Ng, "Semantic role labeling of nombank: A maximum entropy approach," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (EMNLP '06), pp. 138–145, 2006.

[167] N. Xue, "Labeling chinese predicates with semantic roles," *Computational Linguistics*, vol. 34, no. 2, pp. 225–255, 2008.

[168] M. Diab, A. Moschitti, and D. Pighin, "Semantic role labeling systems for Arabic language using kernel methods," in *Proceedings of Association for Computational Linguistics (ACL)*, 2008.

[169] W. Sun, Z. Sui, M. Wang, and X. Wang, "Chinese semantic role labeling with shallow parsing," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pp. 1475–1483, 2009.

[170] Y. Yang and N. Xue, "Chasing the ghost: Recovering empty categories in the Chinese Treebank," in *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pp. 1382–1390, 2010.

[171] X. Carreras and L. Màrquez, "Introduction to the CoNLL-2005 shared task: Semantic role labeling," in *Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL)*, 2005.

[172] S. Pradhan, W. Ward, and J. H. Martin, "Towards robust semantic role labeling," *Computational Linguistics*, vol. 34, no. 2, 2008.

[173] B. n. Zapirain, E. Agirre, L. Màrquez, and M. Surdeanu, "Improving semantic role classification with selectional preferences," in *Proceedings of the 8010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 373–376, 2010.

[174] P. J. Price, "Evaluation of spoken language systems: The ATIS domain," in *Proceedings of the 3rd DARPA Speech and Natural Language Workshop*, 1990.

[175] Borland, *Turbo Prolog 2.0 Reference Guide*, 1988.

[176] W. Ward, "Understanding spontaneous speech," in *Proceedings of the Workshop on Speech and Natural Language*, pp. 137–141, 1989.

[177] W. Ward, "The CMU Air Travel Information Service: Understanding spontaneous speech," in *Proceedings of the Workshop on Speech and Natural Language*, pp. 127–129, 1990.

[178] W. Ward and S. Issar, "Recent improvements in the CMU spoken language understanding system," in *Proceedings of the Workshop on Human Language Technology*, pp. 213–216, 1994.

[179] W. A. Woods, "Transition network grammars for natural language analysis," *Communications of the ACM*, vol. 13, no. 10, pp. 591–606, 1970.

[180] R. Schwartz, S. Miller, D. Stallard, and J. Makhoul, "Hidden understanding models for statistical sentence understanding," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-97)*, pp. 1479–1482, 1997.

[181] S. Miller, R. Bobrow, R. Ingria, and R. Schwartz, "Hidden understanding models of natural language," in *Proceedings of the 32nd Meeting of the Association for Computational Linguistics*, pp. 25–32, ACL, 1994.

[182] Y. He and S. Young, "Semantic processing using the hidden vector state model," *Computer Speech and Language*, vol. 19, pp. 85–106, 2005.

[183] J. Zelle and R. Mooney, "Learning to parse database queries using inductive logic programming," in *Proceedings of the Association for the Advancement of Artificial Intelligence*, pp. 1050–1055, 1996.

[184] R. Ge and R. Mooney, "A statistical semantic parser that integrates syntax and semantics," in *Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL-2005)*, pp. 9–16, 2005.

[185] R. Ge and R. Mooney, "Learning a compositional semantic parser using an existing syntactic parser," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 611–619, 2009.

[186] R. J. Kate and R. J. Mooney, "Using string-kernels for learning semantic parsers," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 913–920, 2006.

[187] Y. W. Wong and R. Mooney, "Learning synchronous grammars for semantic parsing with lambda calculus," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 960–967, 2007.

[188] R. J. Mooney, "Learning for semantic parsing," in *Proceedings of the 8th International Conference on Computational Linguistics and Intelligent Text Processing*, 2007.

[189] L. Zettlemoyer and M. Collins, "Online learning of relaxed CCG grammars for parsing to logical form," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 678–687, 2007.

[190] J. Chen, A. Schein, L. Ungar, and M. Palmer, "An empirical study of the behavior of active learning for word sense disambiguation," in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pp. 120–127, 2006.

第5章

Multilingual Natural Language Processing Applications: From Theory to Practice

语言模型

Katrin Kirchhoff

5.1 概述

人类语言技术的很多应用涉及统计语言模型的使用。该模型给出我们感兴趣的语言词序列的先验概率。给定字母表或基本单元的集合Σ和序列$W=w_1w_2\cdots w_n\in\Sigma^*$，语言模型可以根据从训练集中预先估计的参数，计算$W$的概率值。最为常见的$\Sigma$（也称为词汇表）是包含在训练数据中所有不同词构成的列表。然而，正如我们将在本章看到的，选择符合语言模型定义的词元是相当困难的，特别是对非英语的语言。

通常，一个语言模型和另外一个或者多个可预测的可能词序列的模型一起使用。在语音识别中，一个语音识别系统将声学模型的分数（也可能是其他分数，比如发音模型分数）和语言模型分数融合起来，用于对声音信号的口语词序列进行解码打分。在机器翻译中，语言模型用于为翻译模型产生的机器译文打分。语言模型在信息抽取[1]、作者身份识别[2]和文档分类[3]中已经作为标准工具。在其他相关领域，语言模型定义在声音单元或者孤立的文本字符上，而不是单词上。其中，一种用于语言识别的核心方法是依赖于基于音或音素构建的语言模型[4]。在光学字符识别中，语言模型被用于预测字符序列[5，6]。本章我们关注的是在自然语言词汇或类词汇单元建立的语言模型，我们现在把用空白符号隔开的内容作为基本单元。在讨论特定语言产生的问题之前，比如词形丰富的语言或者没有明显分隔符的语言，我们首先提出了基本的n元模型方法来统计语言模型建模
[169] 和一系列更加高级的建模技术，本章最后给出了多语言和跨语言的语言模型方法。

5.2 n元模型

由于自然语言没有限制，它允许词序列无限长，因此很长的词序列W的概率是无法直接进行计算的。$P(W)$的概率可以根据链式规则分解成各个部分概率的乘积：

$$P(W)=P(w_0\cdots w_i)=P(w_1)\prod_{i=1}^{i}P(w_i\mid w_{i-1},w_{i-2}\cdots w_2w_1) \tag{5.1}$$

因为乘积中的每一项仍然很难直接计算，所以统计语言模型采用了n元近似，这也是为什么它们被称为n元模型，它们假定只有最近的前$n-1$个词与当前词的预测有关，而在此之前的词就与当前词不相关了，或者说它们是等价的。给定这个“历史等价类”的假设，该n元模型可以定义为：

$$P(W)\approx\prod_{i=1}^{i}P(w_i\mid w_{i-1},\cdots w_{i-n-1}) \tag{5.2}$$

根据n的长度，我们可以分别定义1元（$n-1$）、2元（$n-2$）、3元（$n-3$），或者4元、5元等。一个n元模型也称为$n-1$阶马尔可夫模型。因为式（5.2）的概率估计体现了马尔可夫假设，当前词只与前面$n-1$个词有关，与其他词无关。

5.3 语言模型评价

在描述参数估计的方法和基本 n 元模型方法的各种细化之前，我们先来谈谈一个语言模型的性能评价。根据前面给出的定义，语言模型计算词序列 W 的概率。怎么能够知道一个语言模型是否成功估计词序列的概率呢？一般来说有两个标准：在保留的测试集上语言模型的覆盖率和困惑度，这里测试集不属于训练数据的一部分。覆盖率计算测试集中 n 元组在语言模型中的比例。一种特殊的情况是未登录词率（Out-Of-Vocabulary rate，OOV rate），也就是 100 减去 1 元的覆盖率，或者说没有被语言模型覆盖的单个词类型的比例。第 2 个标准即困惑度是一个信息论的度量。给定一个离散的概率分布模型 p，困惑度可以定义为 2 的指数次方，这里的指数是 p 的熵：

[170]

$$PPL(p) = 2^{H(p)} = 2^{-\sum_x p(x)\log_2 p(x)} \tag{5.3}$$

在语言模型中，我们经常对语言模型 q 在包含 s 个词（$w_1\ w_2 \cdots w_s$）的测试集上的性能更加感兴趣，因此语言模型的困惑度可以定义为：

$$PPL(p,q) = 2^{H(p,q)} = 2^{\sum_{i=1}^{s} p(w_i)\log_2 q(w_i)} \tag{5.4}$$

或者简化为：

$$2^{+\sum_{i=1}^{s}\log_2 q(w_i)} \tag{5.5}$$

其中 $q(w_i)$ 是计算第 i 个词的概率，如果 $q(w_i)$ 是 n 元概率，该公式变成了

$$2^{-\frac{1}{s}\sum_{i=1}^{s}\log_2 p(w_i|w_{i-n+1}\cdots w_{i-1})} \tag{5.6}$$

比较不同的语言模型，特别是使用不同方法来把文本分解为不同的语言模型单元（如词或者词素），我们必须根据相同的单元数目对困惑度归一化，这样比较的结果才有意义。

困惑度可以理解为由当前词来预测下一个词时下一个词的可能数量。如果一个模型没有任何的预测能力，那么困惑度等于词汇集合的大小。相反，如果一个模型有完美的预测能力，则它的困惑度为 1。语言模型的研究主要是最小化代表目标领域的保留数据集（held-out data set）的困惑度。

然而，需要注意的是有时语言模型的目标并不是预测词序列的概率，而是用于区分来自于诸如机器翻译系统或语音识别系统这样的前端系统产生的词序列的“好”和“坏”。在这种情况下，语言模型需要为那些错误的、不符合语法或者无法接受的词序列给出一个与正确的序列相比有最大可区分性的分数。最小化困惑度的优化并不是这里的目标，我们将会在 5.6.3 节来讨论这个问题。

5.4 参数估计

5.4.1 最大似然估计和平滑

标准的 n 元模型训练是采用最大似然估计和参数平滑算法对 n 元概率进行估计。最大似然估计可以通过简单的计算相对频率来获得：

$$P(w_i \mid w_{i-n+1}, w_{i-n+2}) = \frac{c(w_{i-n+1}, w_{i-n+2}, w_i)}{c(w_{i-n+1}, w_{i-n+2})} \tag{5.7}$$

[171]

其中 $c(w_{i-n+1}, w_{i-n+2}, w_i)$ 是三元 w_{i-n+1}、w_{i-n+2}、w_i 在训练集中出现的次数。很明显可以发现该方法对没有出现在训练数据中的词序列分配了零概率；另外，在训练集中出现的词序列的概率可能会过度估计。对 n 元中的大概率进行削减并将其分配到零概率的 n 元组中的处

理过程称为平滑。最常见的平滑技术称为回退（backoff）。该方法将 n 元组的计算分为两种，一种是在训练集中频次低于预设的阈值 τ，另外一种是在训练集中频次超过预设的阈值。对于第一种情况，对 n 元的最大似然估计是用低阶的 $n-1$ 元的概率和回退权重来计算的。对于第二种则保留原来的最大似然估计的方法，并用一个打折因子将概率重新分配给低阶的分布。因此，在给定 w_{i-1}、w_{i-2} 的情况下 w_i 的回退概率 P_{BO} 可以根据下式计算：

$$P_{BO}(w_i \mid w_{i-1}, w_{i-2}) = \begin{cases} dP(w_i \mid w_{i-1}, w_{i-2}) & \text{若 } c > \tau \\ \alpha(w_{i-1}, w_{i-2}) P_{BO}(w_i \mid w_{i-1}) & \text{否则} \end{cases} \tag{5.8}$$

其中 c 是 (w_i, w_{i-1}, w_{i-2}) 的频次，d 是高阶分布的打折因子，归一化因子 $\alpha(w_{i-1}, w_{i-2})$ 保证了整个分布的和为 1。它可以由下式进行计算

$$\alpha(w_{i-1}, w_{i-2}) = \frac{1 - \sum_{w_i : c(w_i, w_{i-1}, w_{i-2}) > \tau} dP(w_i \mid w_{i-1}, w_{i-2})}{\sum_{w_i : c(w_i, w_{i-1}, w_{i-2}) \leqslant \tau} P_{BO}(w_i \mid w_{i-1})} \tag{5.9}$$

打折因子的计算方法确定了平滑技术。众所周知的技术包括 Good-Turing、Witten-Bell、Kneser Ney 和其他方法。参见 Chen 和 Goodman [7] 的详细描述以及对不同平滑技术的比较。例如，在 Kneser Ney 平滑中，在概率估计之前，一个固定的打折参数 D 被应用于原始 n 元组频次：

$$P_{KN}(w_i \mid w_{i-1}, w_{i-2}) = \begin{cases} \dfrac{\max(c(w_i, w_{i-1}, w_{i-2}) - D, 0)}{\sum_{w_i} c(w_i, w_{i-1}, w_{i-2})} & \text{若 } c > \tau \\ \alpha(w_{i-1}, w_{i-2}) P_{KN}(w_i \mid w_{i-1}) & \text{否则} \end{cases} \tag{5.10}$$

修正的 Kneser Ney 平滑是被广泛应用的技术，不同的打折因子 D_1、D_2、D_{3+} 被用于出现 1 次、2 次、3 次或更多的 n 元组平滑。

$$Y = \frac{n_1}{n_1 + 2 \times n_2} \tag{5.11}$$

$$D_1 = 1 - 2Y\frac{n_2}{n_1} \tag{5.12}$$

$$D_2 = 2 - 3Y\frac{n_3}{n_2} \tag{5.13}$$

$$D_{3+} = 3 - 4Y\frac{n_4}{n_3} \tag{5.14}$$

其中 n_1、n_2、…是出现 1 次、2 次、…的 n 元组。

172 另外一类常见的语言模型平滑技术是线性插值模型 [8]。在线性插值中，M 个模型通过下式进行融合：

$$P(w_i \mid w_{i-1}, w_{i-2}) = \sum_{m=1}^{M} \lambda_m P(w_i \mid h_m) \tag{5.15}$$

其中 λ 是特定模型的权重。每个模型可能使用不同的条件变量，比如不同长度的历史信息或者来自不同数据集的参数估计，比如大规模的通用领域数据或者是小规模特定领域的数据（参见 5.5 节）。模型的权重受到 $0 \leqslant \lambda \leqslant 1$ 和 $\sum_m \lambda_m = 1$ 的约束。权重通过在不同于模型使用的训练集（并且也不是用于最后的评估或测试集）的保留的数据集上最大化对数似然（最小化困惑度）来进行估计。一般通过期望最大化（Expectation-Maximization，EM）算法来实现 [9]。

5.4.2 贝叶斯参数估计

贝叶斯概率估计是另外一种可选择的参数估计方法，模型的参数被看做是一组受到先验分布控制的随机变量。给定一个训练样本 S 和一组参数 θ，$P(\theta)$ 表示不同 θ 值的先验分布，并且 $P(\theta \mid S)$ 是后验分布，可以通过贝叶斯法则表示为：

$$P(\theta \mid S) = \frac{P(S \mid \theta)P(\theta)}{P(S)} \tag{5.16}$$

在语言模型中，这组参数是词概率的向量，也就是 $\theta = (P(w_1), \cdots, P(w_K))$，（其中 K 是词汇个数），或者更一般化，$\theta = (P(w_1 \mid h_1), \cdots, P(w_K \mid h_k))$是一个包含 K 个 n 元和给定长度为 h 的历史信息的 n 元模型。训练样本 S 是词序列 $w_1 \cdots w_k$，我们要求在给定先验分布和训练样本的情况下对 θ 进行点估计。这个可以通过最大后验（Maximum A Posteriori，MAP）准则或者是贝叶斯准则来实现。前者是根据式（5.16）找到最大后验概率：

$$\theta^{MAP} = \underset{\theta \in \Theta}{\operatorname{argmax}}\, P(\theta \mid S) = \underset{\theta \in \Theta}{\operatorname{argmax}}\, P(S \mid \theta)P(\theta) \tag{5.17}$$

其中 Θ 是 θ 所有可能取值构成的空间。贝叶斯准则找到在给定样本 S 的情况下估计 θ 的期望值：

$$\theta^{B} = E[\theta \mid S] = \int_{\theta} \theta P(\theta \mid S)\mathrm{d}\theta \tag{5.18}$$

$$= \frac{\int_{\theta} \theta P(S \mid \theta)P(\theta)\mathrm{d}\theta}{\int_{\theta} P(S \mid \theta)P(\theta)\mathrm{d}\theta} \tag{5.19}$$

假定先验分布是一个均匀分布，那么对词 w 的最大后验估计也就等价于最大似然估计，而贝叶斯估计等价于拉普拉斯（Laplace）平滑的最大似然估计。 173

$$\theta_w^B = \frac{c(w)+1}{\sum_w c(w) + K} \tag{5.20}$$

对先验分布的不同选择将产生不同的估计函数。语言模型中最常用的先验分布是狄利克雷（Dirichlet）分布。狄利克雷分布是多项分布的共轭先验分布（也就是先验和后验分布有着相同的函数形式）。它可以定义为：

$$p(\theta) = D(\alpha_1, \cdots, \alpha_K) = \frac{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \tag{5.21}$$

其中 Γ 是伽马函数，α_1，…，α_K 是狄利克雷分布的参数（或称超参数），也可以被认为是从一个先验训练样本中得到的计数。在狄利克雷先验下的最大后验估计是：

$$\theta^{MAP} = \underset{\theta \in \Theta}{\operatorname{argmax}} \frac{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{(v_k - 1)} \tag{5.22}$$

其中 v_k 是词 k 在训练样本中出现的次数，它的结果是另外一个狄利克雷分布，参数为 $v_k + \alpha_k$。$P(\theta \mid W, \alpha)$ 的最大后验估计等价于加 m 平滑的最大似然估计，其中 $m_k = \alpha_k - 1$，也就是说，大小为 $\alpha_k - 1$ 的伪计数加到了每一个词（或 n 元组）的计数中。超参数提供了一个便利的方式来集成不同的信息来源，可用于语言模型的参数估计。该方法在语言模型适应中有非常成功的应用（例如 [10]），先验通过大规模领域外的数据集来获得，而观察的频次通过小规模领域内的数据集来计算。参见 5.5 节来获取贝叶斯语言模型适应的细节。早

期对语言模型的构建完全依赖于贝叶斯估计[11]，其性能比不上用5.4.1节描述的技术估计得到的标准n元模型。然而，随着最近贝叶斯统计的发展，其他可选模型已经得到了发展，产生的结果已经超过用Kneser-Ney平滑的n元模型。这里特别要说明的是专业模型包含了假定文档的潜在主题结构并用贝叶斯参数估计技术对结构进行建模。这类模型的全面讨论参见5.6.8节。

5.4.3　大规模语言模型

近年来，人们对语言模型能够适应于大规模数据集变得很感兴趣。每天可用的单语
174 语料数量都在增加。对于很多语言，模型可以构建在几十亿或几万亿数量级的数据上。语言模型对这种规模数据集的适应需要重新语言模型训练、存储和集成到真实系统（例如语音识别解码器）的方式。这也影响到参数估计，因为精确的概率估计变得不可行。

几个站点[12，13]已经提出了使用分布式方法来实现大规模的语言模型建模。它们的共同特点是整个语言模型的训练数据划分为几个部分，并且每一个部分的频次或概率分别存储在不同的物理位置（也就是它们以客户-服务器体系结构分布存储在独立的计算机节点群中）。在运行时，客户端能够从一个语言模型服务器上请求获取数据块集合的统计信息，如此可以实时产生概率估计（可能是以插值的形式）。分布式语言模型的优势是它能够应付超大规模的数据和大规模的词汇量，并且允许数据动态地加入而不用重新计算静态的模型参数。需要的参数如n元模型的阶数或者不同数据块的混合使用方式可以在运行时被选择或指定，这使得动态解码方法可以被使用。然而分布式方法的缺点是网络请求的速度慢。

Brants等人[13]提出一种回退的非归一化形式，这种形式不同于标准的回退（参见公式（5.8））。因为如果在n元组频次超过最小阈值（在这里是0）时，它使用原始的相对频率估计而不是打折概率。

$$S(w_i \mid w_{i-1}, w_{i-2}) = \begin{cases} P(w_i \mid w_{i-1}, w_{i-2}) & \text{若 } c > 0 \\ \alpha S(w_i \mid w_{i-1}) & \text{否则} \end{cases} \tag{5.23}$$

α参数对于所有上下文都是固定的，而不依赖于低阶n元组，如公式（5.8）所示。其结果不再是一个归一化的概率分布而是一组非归一化分数（用S而不是P来标记概率），这组分数和标准概率的使用方式一样。这种方法的优点是非归一化分数在分布式框架下容易计算，因为不再需要对所有n元上下文（保存在不同的物理位置因此查询代价高）求和。有趣的是，作者发现该模型在大规模数据上的性能和用标准的Kneser-Ney平滑训练的模型几乎一样好。

另一种可行方法是在使用小规模语言模型[14，15]产生初次输出后，在第二阶段使用大规模分布式语言模型进行重打分。还有一种方法是在单独机器的工作内存中存储大规模语言模型，但是用不精确的数据结构来提高使用效率。基于该目的，Talbot和Osborne[16]研究布隆（Bloom）过滤器来实现该目标。在这种方法下，语料统计信息（n元频次，上下文频次）用高内存效率、随机数据结构（一个布隆过滤器）这样的量化方式来表示。如果$c(w_1, \cdots, w_n)$是n元$w_1 \cdots w_n$的频次，那么量化频次$q(w_1 \cdots w_n)$定义如下：

$$q(w_1 \cdots w_n) = 1 - \lfloor \log_b c(w_1 \cdots w_n) \rfloor \tag{5.24}$$

175 在测试的时候，通过过滤器查询需要的统计信息。给定量化频次，通过期望频次来估计真实的频率：

$$E[c(w_1 \cdots w_n) \mid q(w_1 \cdots w_n) = j] = \frac{b^{j-1} - b^j - 1}{2} \tag{5.25}$$

在这种框架下，频率将不会被低估但是有可能被高估，尽管高估的概率随着估计错误的大小以指数级别下降。该方法的优点是尽管丢给频率频次可能并不准确，但是数据结构的查询却很快，因此能够使模型即时计算平滑概率。在实践中，基于布隆过滤器的语言模型和基于精确参数估计的语言模型在机器翻译任务的性能上很接近，而内存可节省4~6倍[16]。

大规模语言模型继续的总体趋势是丢弃前面部分提到的精确的参数估计，支持近似估计，随着搜集到的文本数据数目和大小的继续增长，这种发展趋势看起来将会继续，并产生更强和更完善的估计技术。

5.5 语言模型适应

语言模型训练数据不足是一种常态，特别是将一个语音或语言处理系统迁移到新的领域、主题或语言时。基于这种原因，人们对语言模型适应做了很多努力，也就是说设计和调整语言模型使得在只有少量训练数据可用的情况下语言模型在新的测试集上表现得好。

混合语言模型或者模型插值是最常使用的适应方法。一般来说，一个本领域的语言模型可以通过使用小规模的本领域数据来训练，一个大规模背景或通用模型可以通过大规模非本领域的数据来训练。这些模型根据式(5.15)进行插值并在小的开发集上做插值权重的优化。自然地，该方法可运用于多个语言模型，并且已经发展出多种基本模型的插值方法。

一个流行的方法是依赖于主题的语言模型适应。Seymour和Rosenfeld[17]表明文档首先可根据很多个不同主题进行聚类，对于每一个主题类可构建不同的语言模型。目标模型则是选择少量的、特定主题相关的语言模型进行插值来生成的。

一个动态的适应语言模型可以通过**触发器**(trigger)模型来实现。它的想法是根据文档的潜在主题，某些词的组合较其他更经常共现，一些词触发了其他词，例如在财政新闻文档中词stock和market就是如此。潜在语义分析(Latent Semantic Analysis，LSA)[18]和概率潜在语义分析(Probabilistic Latent Semantic Analysis，PLSA)[19]都已经使用[20，21，22]，这些模型根据主题对词进行聚类并用它们作为触发对。LSA最初在 176
信息检索中被形式化，它用文档—词的共现矩阵来表示一组文档，其中行表示不同的词，列表示不同的文档。矩阵的每一个元素表示词出现在文档的频率(可能的权重)。对矩阵做奇异值分解能够将矩阵映射到低阶的连续向量空间，在该空间下可以用诸如余弦距离的方法来计算对应的词向量的语义相似度。语言模型可以动态调整如下：

$$P(w_i \mid h_i, \tilde{h}_i) = \frac{P(w_i \mid h_i)\rho(w_i, \tilde{h}_i)}{Z(h_i, \tilde{h}_i)} \tag{5.26}$$

其中$\tilde{h}_i$表示LSA空间中到词w_i为止的全局文档历史，ρ表示一个相似度函数，用于计算当前词和语义历史的相容性。其想法是语义相近的词的概率将会通过一个因子被加强，该因子与这些词和全局文档的历史信息的相似度成正比关系。触发关系也能通过约束模型框架以约束的形式融入到语言模型中(例如，5.6.5节[23]讨论到的最大熵(MaxEnt)模型或者5.6.3节[24]讨论到的判别式语言模型)。

PLSA扩展了基本的、非概率的LSA模型，它通过假设一个使用更加复杂的潜在类别模型来分解词-文档共现矩阵，而不是简单地采用奇异值分解。给定一个潜在类别c，每一对词—文档共现(w, d)的概率可以表示为：

$$P(w,d) = \sum_c P(c)P(w \mid c)P(d \mid c) = P(d)\sum_c P(c \mid d)P(w \mid c) \tag{5.27}$$

然而，PLSA 的一个潜在问题是它容易对训练数据产生过拟合。最近的一个基于主题的聚类形式是潜在狄利克雷分配（Latent Dirichlet Allocation，LDA）[25]，它可以理解为 PLSA 的正则化版本。基于 LDA 的主题模型和它的扩展形式在 5.6.8 节中讨论。

标准适应框架的进一步变种是无监督适应（unsupervised adaptation），主要与语音识别应用相关。除了使用书写的文档或者没有噪声的转录语音作为适应数据，直接使用语音识别器的输出结果也是一种选择 [26]。各种研究（比如 [27，28]）已经表明这种方法能够达到使用没有噪声的转录信息获得的改进的大约一半的效果。

最近，使用互联网资源作为额外的语言模型数据十分常见。如果特定主题、领域或语言的可用数据不充分，则可以通过网络查询来获取额外的领域相关数据。通过预处理和可能的数据过滤，它们或者加入到已经存在的数据池中，或者根据这些网络获得的数据直接训练一个独立的模型，随后与已存在的基线语言模型进行插值。[29，30，31，32] 对基于这种通用步骤的几种快速适应方法进行了讨论。

最后，人们也研究了针对语言模型适应的其他概率估计方法。其中之一就是**最大后验自适应法**（maximum a posteriori adaptation）[10]，这里的计数分别来自于通用的领域外（Out-of-Domain，OD）和领域内（In-Domain，ID），这些数据的综合如下所示：

177

$$P(w \mid h) = \frac{c_{OD}(w,h) \cdot \varepsilon_{ID}(w,h)}{c_{OD}(h) \cdot \varepsilon_{ID}(h)} \tag{5.28}$$

其中 h 和 w 分别是历史和待预测的词。c_{OD} 是来自于领域外的计数，而 c_{ID} 则是来自领域内的计数。ε 参数的范围是 0～1，它表示分配给适应数据的权重，因为领域外数据的数量一般要超过可用的适应数据，这两类数据的贡献度可以通过近似地设置 ε 这个参数来平衡。最大后验和混合模型 [33] 的比较表明混合模型在适应数据变化时没有最大后验适应的鲁棒性高。

尽管目前大量的语言模型适应工作都是在语音识别的背景下进行的，但有一部分工作则是在机器翻译的背景下进行的。在 Eck、Vogel、Waibel [34] 和 Zhao、Eck、Vogel [35] 的工作中，由初始译文构建的查询将用于从大规模的目标语言数据语料中选择一些额外的句子作为附加的训练数据。根据这些数据构建的模型与基线语言模型进行插值，用来对输入文本的源语言句子再次翻译。在 5.8.2 节中，我们还将讨论可运用跨语言数据用于语言模型适应的其他技术。

5.6　语言模型的类型

尽管到目前为止统计语言模型中最广泛使用的仍然是 n 元模型，但很多其他模型得到了发展并在实际应用中显示出了更多的好处。它们经常和 n 元模型联合使用。

5.6.1　基于类的语言模型

基于类的语言模型 [36] 是解决语言模型数据稀疏的一种简单方法。方法首先根据自动的方式 [37] 或语言学标准将词聚到不同的类别，例如不同的词性类别。该统计模型假定在给定当前词类别的情况下词条件独立于其他词。如果 c_i 是词 w_i 的类别，则一个基于类的二元模型可以定义如下：

$$P(w_i \mid w_{i-1}) = \sum_{c_i, c_{i-1}} P(w_i \mid c_i) p(c_i \mid c_{i-1}, w_{i-1}) P(c_{i-1} \mid w_{i-1}) \tag{5.29}$$

$$= \sum_{c_i} P(w_i \mid c_i) P(c_i \mid c_{i-1}) P(c_{i-1} \mid w_{i-1}) \tag{5.30}$$

在这样的假定下 c_i 在给定 c_{i-1} 的条件下独立于 w_{i-1}。通常一个词只有一个类[⊖]，因此模型可以简化为：

$$P(w_i \mid w_{i-1}) = P(w_i \mid c_i)P(c_i \mid c_{i-1}) \tag{5.31}$$

178

Goodman [38] 将上式的分解和下面的模型做了比较：

$$P(w_i \mid w_{i-1}) \approx P(w_i \mid c_{i-1})P(c_i \mid c_{i-1}) \tag{5.32}$$

当前词不仅条件依赖于当前词的词类，也依赖于前面词的词类。在 North American Business News 语料库中（训练集的数据大小在 10 万个词到 28 400 万个词）的实验中使用了 25 000个测试句子，词汇量为 58 000 个，结果表明式 (5.32) 的模型性能更好，训练数据在 10 万个词附近的情况除外，基于类的模型已经成功地降低了语言模型的困惑度，并对各种不同的语言处理系统的性能提升有帮助。然而它们仍然需要和基于词的语言模型进行插值。

5.6.2 变长语言模型

在标准的语言模型中，词汇单元根据简单的标准来定义，例如空格分隔符。对下一个词出现概率的预测是基于固定长度的历史信息（除了回退），当前已经发展出了该方法的很多变种，旨在以数据驱动的方法重新定义词汇单元，从而产生了由不固定个数的基本单元合并的单元。这些方法称之为变长 n 元模型。这些模型面临的挑战除了要估计语言模型的概率，还要在语言建模单元中找到最佳的词序列 $w_1 w_2 \cdots w_t$ 切分方法。Deligne 和 Bimbot [39] 把词序列的切分看作是一个隐变量，并使用 ML 过程来寻找最佳切分。一个 7 阶的变长模型相比于标准的基于词的二元模型在困惑度上有轻微改进，但是并没有说明在实际应用中的效果。

一个更简单的方法是根据语言的标准书写法，用空格切分词，不重新分词，而是对原来分词结果中的单元进行合并。短语中频繁出现的有限个合并单元可加入到语言模型的词汇表中。一个用于识别潜在短语候选单元的常用标准是相邻词的互信息（例如 [40]）。短语单元实际的选择是使用贪心的迭代算法：每一轮迭代都选择那些能够最大程度降低开发语料困惑度的候选词。在 Zitouni、Smaili 和 Haton [42] 中，词类信息用于识别候选短语单元，因为互信息是在类间而不是在词间进行计算的。相比于基于词的候选对选择，这种方式能够降低大约 10% 的困惑度。该模型也能在中等规模法语自动语音识别（Automatic Speech Recognition，ASR）任务上降低 18% 的相对词错误率。

5.6.3 判别式语言模型

标准的 n 元模型是一个生成模型，对给定词序列 W 分配一个概率。然而，在诸如机器翻译或语音识别这样的实际应用中，语言模型的任务是将好的句子译文和坏的句子译文区分开。基于这个原因，判别式的语言模型参数训练更加适合，这使得不同质量的词串获得最大的区分性概率估计。最近，Roark 等 [43]、Collins、Saraclar 和 Roark [44]、Shafran 和 Hall [45] 以及 Arisoy 等 [46] 对这样的判别式语言模型建模进行了尝试。这里，语言模型应用在已经存在的候选句子译文集合 Y 中，该集合是由一些生成函数 $\mathrm{GEN}(x)$ 对输入 x（例如语音识别中的声音序列或者是机器翻译中的源语言串）产生的，可对于输入 x 和任一输出 $y \in Y$，定义任意的特征函数，并用于一个全局线性模型，通过下述公式来选

179

⊖ 原文为“通常一个类包含不止一个词”，疑错。——译者注。

择最佳译文：

$$F(x) = \underset{y \in GEN(x)}{\operatorname{argmax}} \phi(x,y)\alpha \tag{5.33}$$

其中 α 是一个权重向量，在最基础的情况下，特征函数是来自训练数据的原始 n 元组的计数。然而，模型也可以融合其他特征函数，如表示词类或者是比词小的单元的统计数据（参见 5.7.1 节）。参数向量 α 可以通过感知机算法［47］或者条件对数模型［43］来训练。感知机算法迭代遍历所有训练样本（若干轮）并为每一个样本选择当前最高分的假设，如果与正确的参考假设不同，就通过增加正确假设的特征计数并减去所选假设的特征计数来更新当前的权重。此训练步骤直接对像语言识别系统中的词错误率这样的目标函数最小化。如此，通过优化系统性能而不是最小化 5.3 节所提到的困惑度来对最终模型的不同 n 元特征的权重进行调整，Roark、Saraclar 和 Collins［48］说明在大词汇量语音识别任务中单遍解码降低 1.8%的词错误率（从 39.2%到 37.4%），在多遍解码识别器中降低词错误率 0.9%，最近，判别式语言模型已经应用在统计机器翻译中［49］，相比于最新的基线系统有 1～2 个点 BLEU 值的改进。就如我们看到的，判别式语言模型也提供了一种方便的方法来融合额外的语言学信息，比如形态学特征。

5.6.4　基于句法的语言模型

n 元语言模型一个众所周知的缺点是它不能考虑最近前 $n-1$ 个词之外的历史信息，然而，长距离依存现象在自然语言中普遍存在，当前词的选择依赖于距离句子位置很远的词。在下面的例子中，复数名词 Investors 触发了复数动词 were，但是 n 元模型在这种情况下没有将其作为条件变量而是忽略了，这里的 n 一般不超过 4 或者 5。

Investors, *who still showed confidence in financial markets last week*, **were** *responsible for today's downturn.*

为了解决这个问题，研究者们提出了几种基于句法的语言模型方法，它们的目标是对
180 句法关系进行直接建模，并利用它们提高概率估计的准确度。大多数这些方法使用统计句法分析器构建一个句子的句法表示 S，并且定义一个融合 S 的概率模型。Chelba 和 Jelinek 的**结构语言模型**［50］计算了一个词序列和它的句法分析 S 的联合概率 $P(W,S)$，并将其分解为部件概率的乘积，这些部分涉及词序列的不同元素、句法结构的头节点和句法结构的词性标记。在［50］中的结果表明集成与 3 元模型结合的结构语言模型能够在华尔街日报连续语音识别（Continuous Speech Recognition，CSR）和 Switchboard 语料上使困惑度降低 8%，在语音识别系统中使用词图重打分技术能够使华尔街日报语料的困惑度下降 6%，并在 Switchboard 语料上下降 0.5%（从 41.1%到 40.6%）。

Wang 和 Harper［51］提出了另外一种 "almost-parsing" 语言模型（也称为 SuperARV 模型），该模型基于带依存限制的语法。这里，句子使用 SuperARV 模型来进行标注，它包含丰富的标记组合了词（词典中的实体）的词汇化特征和句法信息。词序列和标记序列的一个联合语言模型（SuperARV 语言模型）可定义如下：

$$P(w_1,\cdots,w_N,t_1,\cdots,t_N) = \prod_{i=1}^{N} P(w_i t_i \mid w_1\cdots w_{i-1},t_1\cdots t_{i-1}) \tag{5.34}$$

$$= \prod_{i=1}^{N} P(t_i \mid w_1\cdots w_{i-1}) P(w_i \mid w_1\cdots w_{i-1},t_1\cdots t_{i-1}) \tag{5.35}$$

$$\approx \prod_{i=1}^{N} P(t_i \mid w_{i-2},w_{i-1},t_{i-1},t_{i-1}) P(w_i \mid w_{i-2},w_{i-1},t_{i-2},t_{i-1}) \tag{5.36}$$

通过对高阶和低阶模型的语言线性差值来实现模型的平滑。SuperARV 模型在华尔街日报宾州树库和 CSR 任务中进行测试，并且和其他基于句法分析的语言模型进行比较，包括前面提到的结构语言模型、标准的二元模型和基于词性的模型，结果表明 SuperARV 模型困惑度分数最低。在 CSR 任务中 SuperARV 模型使用词图重打分使词错误率相对下降了 3.1%~13.5%，再一次超过了其他的模型。

5.6.5 最大熵语言模型

基于最大似然估计的语言模型的缺点之一是语言模型参数估计仅来源于训练集数据，这使得这种估计受训练数据的影响太大。最大熵模型给出了另外一种思路使得这种限制变得更缓和一些。最大熵语模不是根据训练数据的 n 元频次来计算它的概率（可进行平滑），而是认为模型预测的频次平均等同于事件发生的观察次数。最大熵模型的公式如下所示：

$$P(y \mid x) = \frac{1}{Z(x)} \exp\left(\sum_i \lambda_i f_i(x,y)\right) \tag{5.37}$$

其中 $f(x, y)$ 是在输入和预测变量中定义的特征函数，λ_i 是特征函数的权重，$Z(x)$ 是一个归一化因子，根据下式计算：

$$Z(x) = \sum_{y \in Y} \exp\left(\sum_i \lambda_i f_i(x,y)\right) \tag{5.38}$$

一旦合适的特征函数已经定义，那么 f_i 的期望值就是：

$$E(f_i) = \sum_{x \in X, y \in Y} \tilde{p}(x)\, p(y \mid x) f_i(x,y) \tag{5.39}$$

其中 $\tilde{p}(x)$ 是 x 在训练数据中的经验分布，f_i 的经验期望（来自于训练数据）为：

$$\tilde{E}(f_i) = \sum_{x \in X, y \in Y} \tilde{p}(x,y) f_i(x,y) \tag{5.40}$$

要求训练使得期望值与经验期望值相等，并同时最大化 $p(y \mid x)$ 分布的熵。这等价于最大化训练数据的条件对数似然率。

$$E(f_i) = \tilde{E}(f_i) \quad \forall i \tag{5.41}$$

Rosenfeld [52] 最早将最大熵框架应用到语言模型中，在语言模型的上下文中，y 表示预测词，x 表示历史信息，或者更一般地说，x 是用于预测的条件变量。注意到在这种情况下，一个比最近的前 $n-1$ 个词更大的上下文可能允许被包含进来。特征函数可以定义在整个句子 [53] 或者甚至更大的范围中。通常来讲特征函数可以简单地定义在 n 元组上。例如给定词 w_1 和历史 h_1，则一个二元特征函数就定义如下：

$$f_{w_1,w_2}(h_1,w_2) = \begin{cases} 1 & \text{若 } h_1 \text{ 在 } w_1 \text{ 中且 } w_2 = w_2 \\ 0 & \text{否则} \end{cases} \tag{5.42}$$

模型的训练可以使用迭代的方法，比如广义迭代演算（generalized iterative scaling）[54] 或改进的迭代演算（improved iterative scaling）[55]，或更快的伪牛顿方法（quasi-newton approach）（参见 [56]）。然而，最大熵语言模型的训练需要大量的计算，原则上公式 (5.38) 中的归一化因子需要计算所有不同的 x 值，特征期望的计算要求定义特征的所有 (x, y) 对进行求和。Wu 和 Khudanpur [57] 提出了高效的训练方法使得训练速度得到相当大的提高。首先，根据词是否受边缘或条件限制进行划分，并且在两个集合中分别计算归一化因子的和。其次，在重复使用部分求和（比如以相同后缀结尾的历史）时提出了层次的归一化算法。这样使得速度提高了 15~30 倍。

最大熵模型另外一个存在的问题是它容易出现过拟合，特别是使用了大量与样本数目

有关的特征函数时，该问题可能的解决方案是特征选择［55］、正则化［59］，或者是向特征函数引入先验值，例如 Chen 和 Rosenfeld［59］提出了使用高斯先验，该方法并不是简单的最大化训练数据的条件对数似然率，

$$\operatorname*{argmax}_{\Lambda}\sum_{i=1}^{M}\log P_{\Lambda}(y_i \mid x_i) \tag{5.43}$$

而是最大化条件对数似然与包含对所有特征函数的零均值高斯的乘积。

$$\operatorname*{argmax}_{\Lambda}\sum_{i=1}^{M}\log P_{\Lambda}(y_i \mid x_i)\times\prod_{k=1}^{n}\frac{1}{\sqrt{2\pi\sigma_k^2}}\exp\left(\frac{\lambda^2}{2\sigma_k^2}\right) \tag{5.44}$$

其中 σ_k^2 是第 k 个高斯变量的方差。Goodman［60］建议使用指数先验作为一个可选项，有时可以获得更好的性能。

Wu 和 Khudanpur［61］表明了在 Switchboard 任务上对采用最大熵模型的语言识别的结果，这里的 Switchboard 任务引入了主题限制。模型使困惑度下降了 7%，绝对词错误率下降了 0.7%（从 38.5%到 37.8%）。单独集成句法限制可以使困惑度下降 7%，词错误率下降 0.8%。联合这两种类型的限制表明性能有相加的效果，困惑度下降了 12%，绝对词错误率下降了 1.3%。

5.6.6　因子化语言模型

因子化语言模型（Factored Language Model，FLM）的方法［62，63］建立在如下观察的基础上：词的预测依赖于前面词的表层形式，通过考虑增加诸如词的词性或形态类别等额外信息可以使得模型具有更好的泛化能力。特别是，我们可能无法用 n 元频次估计给定 w_{i-1} 的情况下 w_i 的概率，但如果知道词 w_{i-1} 属于特定的类别，假设属于限定词（determiner），就可以对 $P(w_i \mid \text{determiner})$ 获得一个好的概率估计。这使我们想起前面章节提到的基于类的模型。然而，FLM 通过泛化回退策略将很多这样基于类的估计联合起来并进行层次结构化。FLM 假定每个词都有一个因子化表示形式，即词由特征向量而不是单独的表面形式来表示。也就是 $W=f_{1:K}$，一个例子如下：

WORD_i	*Stock*	*prices*	*are*	*rising*
STEM_i	Stock	price	be	rise
TAG_i	Nsg	N3pl	V3pl	Vpart

[185]

词的表面形式可以是其中的一个特征。在这种表达方式下统计模型可以定义如下（使用三元估计）：

$$p(f_1^{1:K},f_2^{1:K},\cdots,f_t^{1:K})\approx\prod_{i=3}^{t}p\left(f_i^{1:K}\mid f_{i-1}^{1:K},\cdots,f_{i-2}^{1:K}\right) \tag{5.45}$$

因此，每个词不仅依赖于按时间排列的词变量的单个数据流，也依赖于同时出现的特征变量。

在标准回退的定义中（公式（5.8）），模型从高阶回退到低阶分布。在 FLM 中，回退过程则不是那么直接明显，因为条件变量不仅包含词序列，也包含了出现的平行特征。因此，我们需要确定哪个特征子集可以向它的低阶回退。原则上有几种不同的方式来选择回退路径：

1）基于语言学知识选择一个固定预定义的回退路径（例如先用形态学特征，后用句法特征）。

2）在运行时基于统计标准选择路径。

3）选择多个路径并组合它们的概率估计。

第三种选择称为平行回退（parallel backoff）。它通过一个新的泛化回退函数来实现（这里以三元来说明）：

$$P_{GBO}(f \mid f_1, f_2) = \begin{cases} d_r P_{ML}(f \mid f_1, f_2) & \text{若 } c \gg \tau \\ \alpha(f_1, f_2) g(f, f_1, f_2) & \text{否则} \end{cases} \tag{5.46}$$

与公式（5.8）类似，c 是 (f, f_1, f_2) 的频次，$P_{ML}(f \mid f_1, f_2)$ 是最大似然估计，τ 是计数的阈值，$\alpha(f_1, f_2)$ 是归一化因子（它保证了产生的分数满足概率分布要求）。函数 $g(f, f_1, f_2)$ 决定了回退策略。在典型的回退过程中，函数 $g(f, f_1, f_2)$ 等价于 $P_{ML}(f \mid f_1)$。在泛化平行回退中，g 可以是 f，f_1，f_2 的任意非负函数，可实例化为均值、加权均值、乘积或最大化函数。例如，均值函数可以利用单独的估计：

$$g_{mean}(f, f_1, f_2) = 0.5 P_{ML}(f \mid f_1) + 0.5 P_{ML}(f \mid f_2) \tag{5.47}$$

除了对 g 可以有不同选择，回退图中的不同层可以选择不同的折扣参数。

我们没有先验知识知道哪一种回退策略有明显优势，最佳的策略高度依赖于特定的语言模型建模工作。因为可能的因子化语言模型结构空间和回退参数空间都很大，一种可取的方式是使用自动的、数据驱动的方法来找到最佳的设置。Duh 和 Kirchhoff [64] 提出了基于遗传算法的 FLM 自动最优化方法。

FLM 已经作为被广泛使用的 SRILM（Stanford Research Institute Language Modeling）工具包 [65] 的一个新增功能，并且成功地应用在基于词素的语言模型 [62]、多说话人语言建模 [66]、对话行为标注 [67] 和语音识别 [68, 69] 中，特别是一些数据稀疏的应用场景。例如，高度屈折变化的语言建模（参见 5.7.2 节）。 184

5.6.7 其他基于树的语言模型

另外几种语言建模方法利用树结构，例如其中一种是 Zitouni [69] 提出的基于层次类的回退模型。这里回退过程是按照词类的层次树形结构来完成的，其中越靠近树的顶端，类别越抽象，越靠近树的底部，类别越明确。回退过程沿着类层次以自底向上方式进行，也就是说，比起抽象类别，优先考虑更为具体的回退类别。与 FLM 的主要不同点是回退路径是固定和预先定义的，然而 FLM 允许合并回退图中不同路径的概率估计以及运行时进行路径的动态选择。Zitouni 发现当测试集包含大量之前未出现事件时，基于层次类的语言模型的作用最大：在语音识别的语言建模任务中，当词汇量为 5000 时，未出现词的困惑度下降 10%，然而在词汇量为 20 000 时，其困惑度则下降了 26%，词错误率下降了 12%。Wang 和 Vergyri [70] 提出了对该层次类的 n 元语言模型做了一些扩展。具体地说，把词性信息加入到词聚类的过程，根据不同的词性种类，层次类树结构被分开定义。在埃及口音阿拉伯语（Egyptian Colloquial Arabic）语音识别任务（其中测试集包含 18 000 个词）中，该模型相比与标准的 n 元模型，降低了 8% 的困惑度；而相比与 Zitouni [69] 提出的模型，降低了 3% 的困惑度。

随机森林语言模型（Radom Forest Language Model，RFLM）由 Xu 和 Jelinek 提出 [71]，该模型对训练集数据中所有的词历史信息看做是一个随机增长的决策树集合（随机森林）。决策树的节点与历史集合有关，根节点包含所有的历史信息。根据词在历史信息中特定位置的身份，树将历史集合分成两个子集，实现树的增长。在所有可能的分割方式中，我们选择最大化训练数据的对数似然率的分割方式。采取两种措施向该过程引入随机性：首先父亲节点的历史集合最初被随机分配给两个子节点，其次用于对数似然率测试的分割方式也是随机选择的。在增长过程结束之后，决策树中的每一个叶子节点都可以看做

是类似词历史信息聚成的等价类，等价类在这里不再是根据最近的 $n-1$ 个词来定义（如传统的 n 元模型），而是基于历史信息中词的从属关系集合来定义的。

具体实现中，决策树的生成过程需要运行多次，每一次运行产生的决策树会加入到随机森林中。假设我们获得了 M 棵决策树，则 RFLM 的概率就是每一个独立的决策树概率的平均值：

$$P_{RF}(w_i \mid w_{i-n+1},\cdots w_{i-1}) = \frac{1}{M}\sum_{j=1}^{M} P_{DT_j}(w_i \mid \phi_{DT_j}(w_{i-n-1},\cdots,w_{i-1})) \tag{5.48}$$

这里 ϕ_{DT_j} 是第 j 个函数，它将历史 $w_{i-1+n+1}$，…，w_{i-1} 信息映射到第 j 棵决策树的叶子节点。决策树 M 的数量一般从几十个到几百个。在华尔街日报语料库的宾州树库部分上
185 的测试结果表明，使用随机森林的三元语言模型的困惑度和词错误率相比于使用 Kneser-Ney 插值平滑的三元语言模型降低了 10.0%。然而，将 RFLM 和 Kneser-Ney 模型进行插值没有使困惑度进一步改善。用 RFLM 的 n-best 列表重打分的方法在华尔街日报 DARPA'93 HUB1 基准任务中使相对词错误率改进了 11%。从这以后，RFLM 一直应用于结构语言建模 [71] 和韵律建模 [72] 中。在多语言建模中，该技术也应用于形态丰富的语言之中（参见 5.7.1 节）。

5.6.8　基于主题的贝叶斯语言模型

最近统计语言建模的一个明显的趋势是对文档的潜在主题结构进行贝叶斯建模。该类最早的潜在狄利克雷分配（LDA）模型是由 Blei、Ng 和 Jordan [25] 提出的。LDA 模型假定一个文档有 K 个主题构成，它们标记为 z_1，…，z_K，每个主题根据该主题下的词分布来生成词（也就是用词袋子模型对主题进行建模；没有考虑 n 元组）。主题 $k=1$，…，K 下的词概率向量用 ϕ_k 来表示，每一个主题有一个先验概率，用 α 表示。主题分布的狄利克雷先验 θ_1，θ_2，…，θ_K 受超参数 α_1，…，α_K 控制（参见 5.4.2 节对狄利克雷分布的解释）：

$$P(\theta_1,\cdots,\theta_K) = \frac{\Gamma\left(\sum_k \alpha_i\right)}{\prod_k \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \tag{5.49}$$

这种方法下的生成模型是由狄利克雷分布采样生成的先验 θ_1，θ_2，…，θ_K 的集合。给定的主题 z_K 以概率 θ_k 来选择，词 w 在该主题下则以概率 $\phi_k(w)$ 来选择。文档包含 i 个词构成的序列 W，其概率计算如下：

$$p(W \mid \alpha,\phi) = \int p(\theta \mid \alpha)\left(\prod_{i=1}^{i}\sum_{z_i} p(z_i \mid \theta)\, p(w_i \mid z_i,\phi)\right) d\theta \tag{5.50}$$

LDA 面临的主要挑战是无法通过精确的推导计算潜在变量 θ 和 z 的后验分布 $p(\theta, z \mid W,\alpha,\phi)$。一般采用像马尔可夫链蒙特卡罗（Markov chain Monte Carlo）（例如 [73]）或变分推理（variational inference）[25] 之类的采样技术来实现。

因为 LDA 模型是一个一元模型，它在具体的应用中需要和一个 n 元模型组合。Wang 等 [74] 将 LDA 和一个三元模型、一个概率上下文无关文法进行组合，在华尔街日报语料中与用 Kneser-Ney 平滑的三元模型相比，困惑度下降 9%～23%。Hsu 和 Glass [75] 采用 LDA 和马尔可夫模型结合用于口语演讲识别任务。在一个已经适应的三元模型上，结合 LDA 模型提供的主题标签所训练的语言模型，困惑度可降低 16.1%，词错误率减少 2.6%。

LDA 模型已经有很多种不同的扩展。第一，LDA 可以推广到利用狄利克雷过程
186 [76]，这是非参数化的先验模型，可以处理无限个主题，因此不假定固定的 K 个主题，

其主题数目可以根据训练数据的属性来调整。第二，潜在的主题变量可以层次结构化，每一个主题可以包含若干个子主题，不同的数据组可以共享同一个主题。这些通过层次狄利克雷过程（Hierarchical Dirichlet Process，HDP）[77]来建模。Huang 和 Renals 利用 HDP 将主题和参与者角色集成到语言模型中来处理会议类型会话语音识别。HDP 自适应语言模型与标准的自适应模型相比，使词错误率稍微下降（0.3%），其中基线系统有 39% 的词错误率。Teh [78] 报告了基于 Pitman-Yor 过程的贝叶斯语言模型和用 Kneser-Ney 平滑的三元模型（没有和基线模型插值）性能相当。

5.6.9 神经网络语言模型

除了基于 LSA 的语言模型，前面提到的语言模型建模方法都是在离散空间中估计事件的概率。神经网络语言模型（Neural Network Language Model，NNLM）[79] 采用了不同的策略，离散的词序列首先映射到连续空间中，然后在这个连续的空间中对 n 元概率进行估计。我们假定具有相似分布属性的词具有相似的连续表示，反过来将产生更平滑的概率估计。

神经网络是典型的多层感知机，其中包含节点的输入层、映射层、隐层和输出层。NNLM 的结构图表达如图 5-1 所示。相邻的层通过带权重的边完全相互连接。词汇量如果为 V，输入则用 $n-1$ 个 V 维的二元特征向量来表示 $n-1$ 个历史词（例如三元组的前面两个词）。维度固定为 d 的映射层 i 在训练时对共享词（该词在训练中学习而得）的连续空间进行编码。隐层 h 包含固定的 J 个节点，每一个节点计算一个阈值，该阈值是一个由输入触发的非线性组合，比如用下面的正切函数来计算：

$$h_j = \tanh\left(\sum_{k=1}^{d} w_{jk}^{c} i_k + b_j^{h}\right) \quad \forall j, j = 1, \cdots, J \tag{5.51}$$

[187]

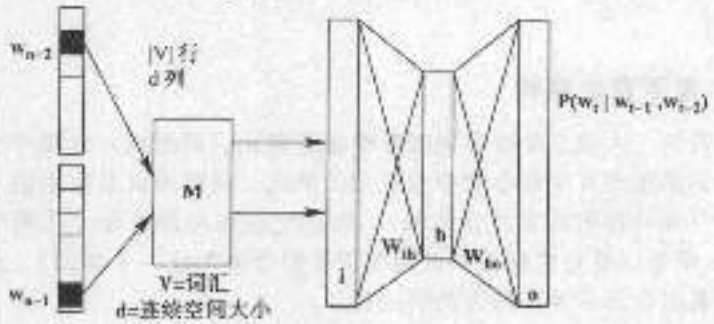


图 5-1 神经网络语言模型

其中 w^{c} 表示连接映射层和隐含层的边的权重，b^{h} 是隐层节点的偏差值。最后，输出层 o [188] 计算 V 的后验概率分布：

$$o_j = \frac{a_j}{\sum_{k=1}^{V} a_k} \quad \forall j, j = 1, \cdots, V \tag{5.52}$$

和

$$a_j = \sum_{k=1}^{J} w_{jk}^{o} h_k + b_j^{o} \tag{5.53}$$

在训练阶段，输出层被赋予二值目标标签信息，其中，1 表示被预测的词，0 表示所

有其他词。通过后向传播算法对神经网络训练，目标是最大化训练数据的对数似然，一般还会加入一个正则项 R 来限制参数值 θ：

$$L = \frac{1}{T}\sum_{t=1}^{T}\log(P(w_t \mid h_t);\theta) - R(\theta) \tag{5.54}$$

正则项可以有不同的形式，一个通用的方法是计算权重的平方和：$\sum_k w_k^2$。这样做可以降低神经网络的复杂度，并减少权重过大而导致的过拟合。因此，历史中一个特别的词首先映射到所有词共享的连续空间。在给定历史的情况下，将该空间作为给定历史信息下估计被预测词概率的一个基础。

Schwenk 和 Gauvain [80]，Schwenk [81]，以及 Schwenk、Déchelotte 和 Gauvain [82] 已经成功地将 NNLM 应用在语音识别任务中，Alexandrescu 和 Kirchhoff [83] 则成功用于机器翻译中。尽管他们仅使用了 m 个使用最多的词汇，但在和标准的 n 元模型组合后也起到了很好的效果，Schwenk [81] 报告了在法语广播新闻识别任务中，语言模型的困惑度下降了 8%，词错误率有 0.5% 的降低。Emami 和 Mangu [82] 在阿拉伯语语音识别任务中词错误率有 0.8% 的绝对降低（3.8% 的相对提高）。

Emami 和 Jelinek [84] 将基于神经网络的概率估计和结构语言模型组合；Alexandrescu 和 Kirchhoff [85] 在阿拉伯语中将 NNLM 和因子化词表示相组合。这样不仅可以利用分布性质，还可以利用形态学和词性类别信息来探索词的相似度。

5.7　特定语言建模问题

对语言进行建模的研究主要针对英语，然而语音和语言处理技术也涉及其他的语言。
[189] 标准的 n 元建模方法对一些语言存在很大的问题，因此有必要对传统的语言模型框架进行调整。在这个部分，我们对 3 类特定语言问题进行探讨：语言形态复杂性、无分词、口语和书面语的比较。

5.7.1　形态丰富语言的建模

形态丰富语言的一大特点是很多词能够根据形态化（词生成）过程产生很多不同且独一无二的词形。词素是语言中最小的承载语义的单元。词素可以是自由的（即单独存在），或者是受约束的（即和其他词素组合出现）。形态化过程包含复合（从两个独立已存在的自由词素生成）、派生（将自由词素和受约束词素组合来生成一个新词）、屈折变化（将自由词素和复合词素组合表示某一特定的语法特征）。

例如德语，以高复合性著称，特别是名词。土耳其语是黏着语，将几个词素组合作为一个词，因此，在英语中用句法短语表示的内容在土耳其语中仅用一个由空格分开的单元表示就可以了。比如 *görülmemeliydik* 等于 "we should not have been seen"。

因此，土耳其语词数量非常庞大。很多语言有丰富的词形变化，像芬兰语和阿拉伯语，根形式（基本形式）可以有几千个不同的词形表现。表 5-1 给出两种现代标准阿拉伯语（Modern Standard Arabic，MSA）屈折变化范式。一种是对词根 skn（基本意义：live）的现在时的动词屈折变化，另一种是词根 ktb（基本意义：book）的代名词所有格屈折变化。

由于较高的词型词例比，形态多样性在语言模型建模中产生了严重的问题，这使得训练集数据存在数据稀疏问题，很多测试数据中的 n 元组没有在训练数据中出现，或者是出现的次数不够多，因此对概率估计不准确。另外一个问题是未登录词（OOV）的高出现

率。在一定程度上，这种负面影响可以通过搜集更多的训练语料来避免。然而，随着使用越来越多的文本，形态丰富的语言并没有和词形较单一的语言一样，词汇增长有明显的下降趋势。这种趋势是由语言形态复杂程度所决定的。表 5-2 表明了不同语言的词项和词例的关系，以及在保留的测试集中不同语言的未登录词率。

表 5-1 现在时动词形式和所有格代词的 MSA 屈折范式（后缀和词干用连字号分开）

词	意思	词	意思
a-skun (u)	I live	kitaabi-y	my book
ta-skun (u)	you (单性) live	kitaabu-ka	your (单性) book
ta-skun-ina	you (阴性) live	kitaabu-ki	your (阴性) book
ya-skun (u)	he lives	kitaabu-hu	his book
ta-skun (u)	she lives	kitaabu-haa	her book
na-skun (u)	we lives	kitaabu-na	our book
ta-skun-uwna	you (阳性复数) live	kitaabu-kum	your book
ya-skun-uwna	they live	kitaabu-hum	their book

表 5-2 不同语言的词例、词型的数目以及未登录词率（非分解形式）

语言	风格	词例数	词型数	N个词中的未登录词率	来
英语	新闻文本	19M	105k	1% (60K)	[86]
阿拉伯语	新闻文本	19M	590k	11% (60K)	[86]
捷克语	新闻文本	16M	423k	8% (60K)	[87]
俄语	新闻文本	15.5M	1.5M	25% (100K)	[88]
土耳其语	混合文本	8M	440k	12% (300K)	[89]
芬兰语	新闻文本、书	100M	4M	1.5% (4M)	[90]

在处理形态丰富的语言时，需要确定具体应用的词汇表是否可以用一个完全词形的列表来表示（例如训练集中出现频率最高的形式），或者是否把比词更小的单元（亚词，subword）选作基本的语言建模单元。该选择依赖于可用计算资源的限制，例如语音识别应用中解码器的高效性、内存和速度要求以及训练数据的规模。将词分解为更小单元的优点是降低了词汇量，反过来也降低了不同 n 元的数目，除了改进了速度和降低了内存损耗外，亚词单元还在多个词中出现，因此每个单元的训练词例数目增加了，这样使得概率估计更具鲁棒性。最后，基于亚词单元的建模使得语言模型把非零概率分配给那些没有在训练数据中出现的词。另外，如果一个词被统性分解，一个固定的 n 元上下文仅提供了一个词不同部分的关系，没有提供词之间的依存关系。因此语言模型的预测能力下降了。并且，当语言模型的词汇量和用于语音识别的词汇量相等时，需要注意定义的单元不能太小，如果太小在声学上容易混淆。

阿拉伯语通常被认为是一种形态丰富的语言，它可以作为一个有趣的例子来强调在不同的情况下是否需要分解成亚词单元。多项研究 [68, 63, 91] 已经表明引入形态信息到语言模型中对带阿拉伯语方言的建模有帮助。尽管阿拉伯语的方言较现代标准阿拉伯语（书面标准）在形态上更为简单，但它的训练数据非常稀疏，因为它本质上是口语并且需要人工转录成可获取的语言建模数据。现代标准阿拉伯语有大规模的数据可用，但在使用大规模数据时 [91]，语言模型中的形态分解并没有产生明显的改进，并且像现代标准阿拉伯语的语音识别这种大规模应用需要的词汇数量为 60 万～80 万个词，这样的数量级是当前解码器能够处理的 [92]。

对于具有特别高的词型词例比的语言（芬兰语或土耳其语），形式分解是需要的；对于大型任务，如果需要充分覆盖测试集的数据，所需词汇量很可能超过当前解码器的处理

能力，并且相应的语言模型概率估计也不够准确。在下面的部分中，我们讨论词分解问题最近的几种处理方法。

5.7.2　亚词单元的选择

我们可以用数据驱动、无监督的方式来对亚词单元进行识别；也可以基于语言学信息（例如形态学分析器）；或者是二者的结合。基于语言学的方法主要涉及手写形态分析工具，比如为阿拉伯语而开发的 Buckwalter 形态分析器，该分析器把每个词转化为不同形态部件。每一个词形在这种情况下会有几种可能的分析，因此后续阶段需要执行统计消歧这个步骤（例如 [34]）。针对考察的特定语言，数据驱动方法融入了不同粒度的信息，并且优化标准可以有很大的不同。一些方法致力于发现语言学上定义的语素相对应的单元，然而其他方法则专注于选择一个最适合当前任务或应用的基本单位。

识别语言学语素的自动算法最早是 Zellig Harris 在 1995 年提出的方法，它估计词里每一个字母后面接不同字母的困惑度 [85]。如果某个转移的困惑度很高（即后面的字母很难估计），那么在这种情况下可以假设一个词素边界。Adda-Decker 和 Lamel [96] 对上面的方法做了修改，用来分解德语复合词，从而使一个规模为 3 亿词、固定词汇量在 65 000~100 000 之间的德语语料，未登录词率相对下降 23%~50%。

一般来讲，简单的基于频率的方法容易出现对训练数据的过拟合并且产生比预期更多的词素。因为拟合的数据相对于整个词素数目是不平衡的。解决这个问题的办法是在建模时显式包含对词素集合大小的惩罚项，最近开发的 Morfessor 工具包 [97] 就是这样实现的，它通过最大化语料 C 的后验概率来获得词素集合 M：

$$M = \arg\max_{M} P(M \mid C) = P(C \mid M)P(M) \tag{5.55}$$

这也等价于最小化描述长度的方法。通过贪心算法实现对可能词素的搜索，也就是尝试所有可能的分割方式，将每一个词递归分割成两个部分。我们选择那些能够改进概率 $P(M \mid C)$（减少编码长度）的分割方式。这个方法的后续版本 [98, 99] 包含一个建模的形态类别模型和不同的概率估计技术。Morfessor 模型在一个涉及芬兰语、土耳其语和英语 [100] 的基线评测任务中超过了其他自动分词算法。使用 Morfessor 分解单词的语言模型已经应用到芬兰语、爱沙尼亚语、土耳其语和阿拉伯语的语音识别中，并在前面 3 种高黏着性语言的测试中获得好成绩 [90]。

191

与尝试匹配预定义语言单元集合不同的另外一种方法是推出一个直接优化诸如困惑度或未登录词率这些用于评价语言模型性能标准的单元集合。这对那些没有严格黏着性的语言可能更为适合，这些语言包含一定数量的屈折变化，例如由组合两个或两个以上语素产生的词性有不透明的变化。Whittaker 和 Woodland [101] 采用了该方法，将其运用于俄语的建模。在此，一个基于小品词的模型定义为：

$$P(w_i \mid h) = \frac{1}{Z(h)} P\left(u^{w_i}_{L(w_i)} \mid u^{w_i}_{L(w_i)-1}\right) P\left(u^{w_i}_{L(w_i)-1} \mid u^{w_i}_{L(w_i)-2}\right), \cdots, P\left(u^{w_i}_{1} \mid u^{w_{i-1}}_{L(w_{i-1})}\right) \tag{5.56}$$

其中词 w_i 可以根据一些分解函数 L 分解为 $L(w_i)$ 个小品词 $u_1, \cdots, u_{L(w_i)}$。小品词语言模型计算在给定历史下小品词的概率，其中历史包含到上一个词中最后一个小品词的所有小品词。比较两种推出小品词的数据驱动方法：一种是对固定长度下所有可能单元的贪心遍历，保留那些能够最大化数据似然的小品词，另一种是小品词增长技术，小品词初始化为所有的单字符单元，然后连续加入附近的字符来扩展，使得最后产生的单元能够获得最小的困惑度。

Kiecza、Schultz 和 Waibel [88] 提出了一种韩语建模的方法，该方法将基本音节单元组合为比音节大，但是比韩语词小的单元，称为 eojols，它和土耳其词的复杂度相近。通过最小化未登录词率来对音节进行组合。在这两种方法里，困惑度和未登录词率都有了很大的改进，但在最后的系统评估（语音识别词错误率）中，系统性能保持不变，或没有少量提升。

在最近的研究中，亚词单元的选择都根据最终系统性能来进行优化。一般的做法是尝试所有的切分方法，并评估它们对系统性能的影响。Arisoy、Sak 和 Saraçlar[132] 在土耳其语音识别中，比较了采用下列 4 种单元：词、随机切分的单元、语言学上定义的语素，词根加后缀的语言模型建模对系统性能的影响，结果表明词根加后缀的词错误率效果最好，优于其他 3 种。

5.7.3 形态类别建模

语言模型中针对亚词单元的大多数工作主要关注黏着语中词的线性分解。结果产生的亚词单元最常在标准的 n 元模型中使用。就如前面提到的，一个问题是 n 元组的上下文需要扩展，除了要对亚词单元的依存关系建模，还要完成词间依存关系的建模，这也相对应需要增加训练数据的规模。

然而，在研究者们提出的几种方法中，词仍然是建模的基本单元，但是概率分配上考虑了亚词部件或者形态类的统计信息。Arisoy 等 [46] 提出了针对土耳其语的判别式语言模型（参见 5.6.3 节）。该模型引入了定义在词素上的特征函数，例如词根的 n 元频次或 192
者屈折变化类别频次。语言模型为整个词序列分配概率（未分解的 n-best 假设），但考虑基于词素特征函数给出的限制。实验结果表明该模型相对于只使用基于词特征的判别式语言模型有少量的改进（在广播新闻识别任务中绝对词错误率降低了 0.3%）。Shafran 和 Hall [45] 用相同的方法处理捷克语，取得了类似的结果。

Kirchhoff 等 [63] 和 Vergyri 等 [68] 在因子化语言模型中对阿拉伯语使用形态类（词干、词根等）和词作为条件变量。尽管模型对整个词形的概率进行预测，但是在概率回退过程使用了形态学成分信息。在阿拉伯语的语音识别任务中，使用有限训练数据训练的 FLM 语言模型可以略微降低系统词错误率（绝对错误率，0.5%～1.5%）。FLM 已经成功应用于形态多变语言的语音识别，例如，爱沙尼亚语 [103]，也用于机器翻译 [104]。

形态化特征（除了词本身）在阿拉伯语和土耳其语 [85] 中也作为神经网络语言模型（参见 5.6.9 节）额外的输入特征，创建了一个因子化神经网络语言模型。相比于基于词的神经网络语言模型，该模型的困惑度有实质性的改进（对于阿拉伯语为 10%，对于土耳其语为 40%），但目前没有具体应用结果的相关报告。

Sarikaya 和 Deng [105] 提出了面向阿拉伯语的形态和词汇的联合语言模型。这里，句子用一个表示形态、句法、语义和其他属性信息的句法树来标注。语言模型使用最大熵概率（参见 5.5.5 节）估计，同时预测词串及其对应的树的概率。在英语到阿拉伯语的翻译任务评测中，相比于基于词的三元语言模型，该模型提升了 0.3 个点（绝对值）的 BLEU，相比于基于语素的三元语言模型，提升了 0.6 个点的 BLEU。

Oparin 等将 RFLM（参见 5.6.7 节）用于形态语言模型建模 [106]，和标准的基于词的 RFLM 不同，随机森林模型使用的决策树不仅可以查询不同词的从属关系，也可以查询关于形态特征（曲折变化或形态标记）、词干、原形和词性的信息。通过一个包含 24 万词汇的捷克口语演讲识别任务，研究者们对模型进行了评测。尽管基于词的 RFLM 相对于 Kneser-Ney 类型的三元语言模型没有实质性的提高，但是形态 RFLM 在困惑度上有 19.4% 的相对提高，

在词准确性上有3.4%的相对提高。与前面的研究结果不同，将RFLM和标准的n元模型插值能够进一步提升系统性能（困惑度提升了15.6%）。除了产生不同的词历史聚类（通过形态特征而不是词特征来导出），形态RFLM具有更大的随机性，因为每一个决策树节点潜在的分割方式大幅度增加，在这种情况下可能使形态RFLM获得较好的性能。

5.7.4　无分词语言

尽管新著性在很多语言中生成了很多很长并且复杂的词形，但其他语言仍然没有显式
[193] 地对字符串进行分词。在中文或日语这样的语言中，句子写作一连串字符，用标点符号隔开，但是内部没有空格来表示词的边界。对这些语言有背景知识的读者能够马上用最正确的理解方式对字符序列进行分词。尽管统计语言模型对这种语言可以基于字符进行建模，但是先对它们进行分词，然后再训练语言模型这样更为合适。和将词分解为亚词单元（参见5.7.1节）类似，使用字符作为基本的建模单元可能无法正确表达词间的关系。并且，分词能够决定字符如何发音，这对语音识别系统中语言模型和发音模型使用相同的建模单元来说很重要。最后，实验表明，在中文[107]和日语[108]中，在自动分词的文本上构建的语言模型比基于字符的语言模型的困惑度更低。

自动分词算法主要将词典信息、统计搜索、额外特征，例如外来字母、字符共现次数和字符的位置等进行融合。这些算法大多根据统计解码框架，使用诸如Viterbi搜索算法生成最有可能的分词结果。此外，研究者们还探索了包含条件随机场[109，110，111]、最大熵建模[112，113]和感知机[114]判别式模型等其他方法。很多中文分词的工作是在从2003年起由ACL（Association for Computational Linguistics）举办的SIGHAN中文分词比赛中做的。这个比赛已经成为评测不同分词系统的基准平台。通过精确率P（也就是正确分词占分词结果的比例）、召回率R（所识别的正确分词占所有正确分词的比例）以及它们的组合指标F值（$F=2PR/(P+R)$），自动分词结果和语言学上真实的分词结果进行比较。这些可以对未登录词和词汇表中的词分别计算。目前，最好的分词系统在最近的评测中F值为0.96。然而，对于未登录词，F值偏小，大约为0.76[115]。

除了尝试匹配语言学上定义的词，优化分词能够直接提高语言模型的性能。Sproat等[108]证明用于中文分词的词典对在已分词文本上训练的二元语言模型的困惑度有很大的影响。通过合并频繁共现的词对来迭代优化字典，使得每一次迭代困惑度都下降。注意，该方法和前面5.7.1节中使用数据驱动算法来生成类词表的亚词单元的方法很像。另外一个数据驱动方法的例子是日语，使用字符块来对语言模型建模[107]。具体实现中，通过选择最高频的n元组和与之相类似的模式来生成块。因此，基本的模型单元既不是字符也不是词，而是中间的单元。

5.7.5　口语与书面语言

[194] 统计语言建模很大程度上依赖于大规模的书写文本数据，并且语言建模研究的一个明显趋势是研究如何调整当前的语言建模技术去适应更大的数据库。然而，世界上6800种语言中，很多语言是口语，也就是没有书写系统的语言。它们要么是土著语言，没有文字传统，要么是语言变种，比如地方方言，每天作为口语来交流，而很少应用在写作中。举个例子，阿拉伯语的很多方言用在日常的交际中，但是几乎没有书写形式。其他语言可能既有口语又有书写，但是可能没有标准的正确拼写。

这两种情况说明了语言建模的困难。对于第一种情况，获得语言建模训练数据的唯一

方法是手工转录语言或方言。这样做的代价很高，并且整个过程非常消耗时间，因为它涉及：1）写作标准的制定。2）训练母语者让他们使用写作系统并保持一致性和准确性。3）在数据转录中的实际投入。对于第二种情况，从那些正在考察的语言中获取的文本资源（例如通过互联网）需要进行标准化，这是一个相当耗费劳力的过程。因此，对这些资源稀缺的语言，几乎没有相关的语言模型建模工作。很多研究集中在如何通过网络来快速收集这些资源稀缺语言的语料。Le 等 [116] 及 Ghani、Jones 和 Mladenic [117] 描述了该过程中面临的一些内在的挑战。对口语和缺乏标准的语言，可能快速进行语言模型建模的方法包含基于文法或基于类别的方法，并结合有限的转录数据。对于一个受限的应用，例如对话系统的开发，可能的言语结构可以通过任务语法或基于类的语言模型来预定义，然而更细粒度的词序列概率或给定词类下的词概率则由小规模数据训练得到的语言模型来完成。一个有趣的研究方向是使用与考察的语言相近的语言或者虽不相近但资源丰富的语言数据来改善目标语言模型。下面章节对这些方法做了一些描述。

5.8 多语言和跨语言建模

5.8.1 多语言建模

至此，我们已讨论了直接将统计语言模型应用于特定语言或语言类型所引起的问题，例如黏着语或无分词的语言。我们一直默认假设语言模型只在与目标语言相关的应用中使用。然而，在很多情况下，一个系统可以顺序地面临多种语言（例如不同的用户使用不同的语言，没有预先告知随后的文本中会出现什么语言），或在诸如**编码切换**（code switching）中同时出现多种语言。这里说话人在同一句话中可能同时使用多种语言或方言。编码切换的现象存在于各种各样的双语或多语言社区，或使用两种语言或者方言的场景，例如除了口语或方言变种，还使用正式标准语言。在美国“西班牙语”的使用（混合了西班牙和英语）就是编码切换的一个例子。下面，我们通过 Franco 和 Solorio [118] 的例子来说明问题： [195]

I need to tell her que no voy a poder ir.

'I need to tell her that I won't be able to make it.'

为了处理口语间语言动态切换的多语言输入，可以根据单语语料对语言模型进行分别建模，使用了这些模型的系统（例如一个基于语音的搜索或基于电话的对话系统）可以基于第一步的语言识别结果来选择语言模型，或者基于在初始处理之后产生最高分数的语言模型（在语音识别中有时会结合发音模型）来进行动态选择。

Fugen 等表明如何通过上下文无关文法将几个单语语言模型合并成一个多语语言模型，其中文法的非终结符包含语言信息，终结符状态与单语 n 元模型一致。使用明确的文法规则来对现有状态进行扩展（只用匹配语言中的 n 元组），以避免不合时机的语言切换。构建单个多语语言模型的可选方法是在包含多个单语语料的数据池中训练一个单独多语言模型或训练多个单语语言模型，然后以插值方式来使用。第一种技术降低了系统性能，特别是语料大小不平衡的时候 [120，121]。第二种技术则有轻微的提高，但仍然比不上前面提到基于文法的方法 [119]。

对第二种情况（句内语言转换）的语言建模十分困难，因为几乎没有或根本没有相关的训练数据可用。Wang 等 [122] 通过引入一个暂停单元形式的通用回退节点来构建 4 种语言的语言模型，也就是语言在出现暂停后允许以某种概率进行切换。

5.8.2 跨语言建模

另外一个问题是一种语言的数据是否可以帮助改进另一种语言的语言模型，假定风格

或领域非常接近。如果目标语言的可用数据不足，但存在大量的外语文本，从中可抽取足够的信息，那么原来不精确的概率估计有望借此得到改善。

这个思想最直接的方法是自动将其他语言文本翻译成目标语言，然后将它（尽管有错误）作为额外的语言训练语料。Khudanpur、Kim [123] 和 Jensson 等 [124] 采用了这种方法。

在早期的研究中，用于语音识别的中文新闻文本语言模型的训练数据，就是通过添加同领域自动翻译的英语文本译文来进行扩充的。从翻译文本中抽取的一元和在可用的中文
196 文本数据上训练的三元基线语言模型进行插值。用于翻译的英语文本的选择和插值系数λ根据每一个新的具体场景指定，如此同时假定了一个隐式主题适应形式。在语音识别中，产生的模型的字符困惑度有大约 10%的相对降低，词错误率有 0.5%的绝对降低（对于不同的系统，基线字符错误率大约为 26%）。作者也注意到英语文本相对中文文本更新，因此也许对系统性能的提高有帮助。这在调查潜在的外语言（out-of-language）的数据资源时是一个重要的考虑因素。

Jensson 等 [124] 为冰岛的天气预报开发了一个语言模型，使用小规模的内语言（in-language）数据，以及用来训练机器翻译系统的有限数量的英语—冰岛语平行语料。通过大规模自动翻译的数据训练一个语言模型和基线语言模型插值，在冰岛语音识别系统中有积极的作用，困惑度有 9.2%的降低，词错误率相对降低了 1.9%～9.8%。

然而，如果在开始没有充分的语言数据来训练机器翻译系统，则用机器翻译技术处理其他语言数据可能会失败。尽管上述冰岛语实验中表明在严格受限的领域中，有限的平行语料仍然可能是足够的。也可以不使用完全成熟的机器翻译系统，而依赖于高质量的基于词的翻译词典。Kim 和 Khudanpur [125] 表明高质量的翻译字典可以从文档对齐平行语料的词对中通过计算互信息统计来获得，而并不需要句子对齐的平行语料。在中文广播新闻识别的实验中，他们发现通过基于词典的翻译结果和基线语言模型插值生成的一元模型能够达到和跨语言语言模型相似的性能。另外一种通过文档对齐数据构建翻译词典的可行方法是使用跨语言的潜在语义分析 [126]。在这种方法中，两种语言的词都映射到同一个语义空间中，在该空间中不同语言的词间相似度用于构建词翻译概率。

前面提到的方法的一个缺点是产生模型的质量很大程度上依赖于翻译准确度。Tam 等 [127] 最近提出了另外一个模型，在翻译之前使用双语潜在语义分析（bilingual Latent Semantic Analysis，bLSA）进行适应。方法要求源语言和目标语言都分别使用一个 LSA 模型，因此需要一个平行训练语料。LSA 模型在主题上引入了狄利克雷风格的先验分布（参见 5.6.8 节），混合主题权重通过源端 LDA 模型来确定，并映射到目标 LSA 模型中，它们可以用于计算目标语言的边际分布。假定源语言是中文（Ch），目标语言是英语（En）。在英语中词的边际概率分布为

$$P_{\mathrm{En}}(w) = \sum_k \phi_k^{\mathrm{En}}(w)\theta_k^{\mathrm{Ch}} \tag{5.57}$$

其中 θ_k 是第 k 个主题的先验，$\phi_k(w)$ 是根据第 k 个潜在主题生成词 w 的概率。就如我们
197 看到的，主题先验是由源语言决定的，然而主题相关的词概率分布由目标语言决定。目标语言的边际概率以下列方式融入到目标语言模型中，如下：

$$P_{\mathrm{target}}(w \mid h) \propto \left(\frac{P_{\mathrm{bLSA}}(w)}{P_{\mathrm{base}}(w)}\right)^{\beta} P_{\mathrm{base}}(w \mid h) \tag{5.58}$$

其中 P_{bLSA} 是适应概率，P_{base} 是基线模型概率。这个方法在跨语言映射中强制双语词语的主题一一对应，在新领域的汉英统计机器翻译评测任务中表明 bLSA 适应语言模型困惑度降低了 9%～13%，并且 BLEU 值提升了 0.3 个点。

5.9 总结

统计语言模型在最近几年得到了很大的发展。尽管采用平滑的最大似然估计的经典 n 元模型仍然是主流的方法，但是，很多新的模型，例如从神经网络模型到判别式语言模型，都已经和标准的 n 元模型一起被使用。

很多语言建模技术，比如语言模型适应，已经被证明适用于很多不同形态的语言，并且核心技术可以说是语言独立的。如语言有丰富的形态变化，特别是黏着性的语言，也就是每个语素可以产生很多不同的词形，则会存在关键的差异。这种情况下，在 n 元语言建模前进行词分解，或把基于亚词部件的统计信息融入到判别式、因子化、神经网络语言模型中是非常有帮助的。

一个最近的趋势是使用超大规模分布式语言模型，它不使用传统的概率估计方法，而是使用近似分数或计数。考虑到语言建模数据的数量在每天增加，这种趋势肯定会在近期产生很大的影响，这对大词汇量的语言（例如形态多样化的语言）来讲也很有意义，因为这使实际系统使用大型语言模型更方便。

相比较而言，几乎没有研究是针对资源匮乏的语言的，这些语言包含了大多数口语和方言，没有大规模可用的文本数据。在这点上，从这些语言种类中获得数据的标准方法是手工转录，而这只能获得有限数量的数据。将举衍（bootstrapping）技术和语音识别技术结合可以以一种增量方式来自动转录更多的数据。然而，这是一个鸡和蛋的问题，因为足够精确的语音识别系统需要足够的文本和声音数据来训练初始模型。语言模型中跨语言的适应技术是未来重要的发展方向，这将对把人类语言技术应用到资源匮乏语言产生重要影响。 [198]

参考文献

[1] J. Ponte and B. Croft, "A language modeling approach to information retrieval," in *Proceedings of the ACM Special Interest Group on Information Retrieval (SIGIR)*, pp. 275–281, 1998.

[2] F. Peng, D. Schuurmans, S. Wang, and V. Keselj, "Language independent authorship attribution using character level language models," in *Proceedings of the European Chapter of the Association for Computational Linguistics (EACL)*, pp. 267–274, 2003.

[3] F. Peng, D. Schuurmans, and S. Wang, "Language and task independent text categorization with simple language models," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 110–117, 2003.

[4] M. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4(1), pp. 31–44, 1996.

[5] M. Nagata, "Japanese OCR error correction using character shape similarity and statistical language model," in *Proceedings of the Association for Computational Linguistics*, pp. 922–928, 1998.

[6] I. Bazzi, R. Schwartz, and J. Makhoul, "An omnifont open-vocabulary OCR system for English and Arabic," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, pp. 495–504, 1999.

[7] S. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," Tech. Rep. TR-10-98, Harvard University, 1998.

[8] F. Jelinek and R. Mercer, "Interpolation estimation of Markov source parameters from sparse data," in *Proceedings of the Workshop on Pattern Recognition in Practice*, 1980.

[9] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39(1), pp. 1–38, 1977.

[10] M. Federico, "Bayesian estimation methods for n-gram language model adaptation," in *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP)*, pp. 240–243, 1996.

[11] A. Nadas, "Estimation of probabilities in the language model of the IBM speech recognition system," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 859–861, 1984.

[12] A. Emami, K. Papineni, and J. Sorensen, "Large-scale distributed language modeling," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 37–40, 2007.

[13] T. Brants, A. Popat, P. Xu, F. Och, and J. Dean, "Large language models in machine translation," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 858–867, 2007.

[14] Y. Zhang, A. Hildebrand, and S. Vogel, "Distributed language modeling for n-best list reranking," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 216–223, 2006.

[15] H. Schwenk and P. Koehn, "Large and diverse language models for machine translation," in *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)*, pp. 661–668, 2008.

[16] D. Talbot and M. Osborne, "Smoothed Bloom filter language models: tera-scale LMs on the cheap," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 468–476, 2007.

[17] K. Seymour and R. Rosenfeld, "Using story topics for language model adaptation," in *Proceedings of Eurospeech: European Conference on Speech Communication and Technology*, pp. 1987–1990, 1997.

[18] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, pp. 391–407, 1990.

[19] T. Hofmann, "Probabilistic latent semantic analysis," in *Proceedings of the 22nd Annual International SIGIR Conference on Research and Development in Information Retrieval (SIGIR-99)*, pp. 35–44, 1999.

[20] J. Bellegarda, "Exploiting latent semantic information in statistical language modeling," *Proceedings of the IEEE*, vol. 88(8), pp. 1279–1296, 2000.

[21] D. Mrva and P. Woodland, "A PLSA-based language model for conversational telephone speech," in *Proceedings of the 8th International Conference on Spoken Language Processing (ICSLP)*, pp. 2257–2260, 2004.

[22] S. Bai and H. Li, "PLSA based topic mixture language modelling approach," in *Proceedings of the 6th International Symposium on Chinese Spoken Language Processing*, pp. 1–4, 2008.

[23] R. Lau, R. Rosenfeld, and S. Roukos, "Trigger-based language models: a maximum entropy approach," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 45–48, 1993.

[24] N. Singh-Miller and M. Collins, "Trigger-based language modeling using a loss-sensitive perceptron algorithm," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 25–28, 2007.

[25] D. Blei, A. Ng, and M. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[26] R. Gretter and G. Riccardi, "On-line learning of language models with word error probability distributions," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 557–560, 2001.

[27] M. Bacchiani and B. Roark, "Unsupervised language model adaptation," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 224–227, 2003.

[28] G. Tür and A. Stolcke, "Unsupervised language model adaptation for meeting recognition," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 173–176, 2007.

[29] I. Bulyko, M. Ostendorf, and A. Stolcke, "Getting more mileage from web text sources for conversational speech language modeling using class-dependent mixtures," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 7–9, 2003.

[30] T. Ng, M. Hwang, M. Siu, I. Bulyko, and M. Ostendorf, "Web-data augmented language models for Mandarin conversational speech recognition," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 589–592, 2005.

[31] A. Sethy, P. Georgiou, and S. Narayanan, "Building topic-specific language models from webdata using competitive models," in *Proceedings of Eurospeech: European Conference on Speech Communication and Technology*, pp. 1293–1296, 2005.

[32] V. Wan and T. Hain, "Strategies for language model web-data collection," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1069–1072, 2006.

[33] L. Chen, J. Gauvain, L. Lamel, G. Adda, and M. Adda, "Language model adaptation for broadcast news transcription," in *Proceedings of the ISCA ITR Workshop on Adaptation Methods for Speech Recognition*, 2001.

[34] M. Eck, S. Vogel, and A. Waibel, "Language model adaptation for statistical machine translation based on information retrieval," in *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pp. 26–28, 2004.

[35] B. Zhao, M. Eck, and S. Vogel, "Language model adaptation for statistical machine translation with structured query models," in *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pp. 411–417, 2004.

[36] P. Brown, V. D. Pietra, P. de Souza, J. Lai, and R. Mercer, "Class-based n-gram models of natural language," *Computational Linguistics*, vol. 18(4), pp. 467–479, 1992.

[37] S. Martin, J. Liermann, and H. Ney, "Algorithms for bigram and trigram word clustering," in *Speech Communication*, pp. 1253–1256, 1998.

[38] J. Goodman, "A bit of progress in language modeling," *Computer Speech and Language*, pp. 403–434, 2001.

[39] S. Deligne and F. Bimbot, "Language modeling by variable length sequences: Theoretical formulation and evaluation of multigrams," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 169–172, 1995.

[40] F. Jelinek, "Self-organized language modeling," in *Readings in Speech Recognition* (A. Waibel and K.-F. Lee, eds.), pp. 450–506, San Mateo, CA: Morgan Kaufman, 1990.

[41] K. Ries, F. Buo, and A. Waibel, "Class phrase models for language modeling," in *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP)*, pp. 389–401, 1996.

[42] I. Zitouni, K. Smaili, and J. P. Haton, "Statistical language modelling based on variable length sequences," *Computer Speech and Language*, vol. 7, pp. 27–41, 2003.

[43] B. Roark, M. Saraclar, M. Collins, and M. Johnson, "Discriminative language modeling with conditional random fields and the perceptron algorithm," in *Proceedings of the Association for Computational Linguistics*, pp. 47–54, 2004.

[44] M. Collins, M. Saraclar, and B. Roark, "Discriminative syntactic language modeling for speech recognition," in *Proceedings of Association for Computational Linguistics*, pp. 507–514, 2005.

[45] I. Shafran and K. Hall, "Corrective models for speech recognition of inflected languages," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 390–398, 2006.

[46] E. Arisoy, B. Roark, I. Shafran, and M. Saraclar, "Discriminative n-gram modeling for Turkish," in *Proceedings of Interspeech: Annual Conference of the International Speech Communication Association*, pp. 825–828, 2008.

201

[47] M. Collins, "Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 1–8, 2002.

[48] B. Roark, M. Saraçlar, and M. Collins, "Discriminative n-gram language modeling," *Computer, Speech and Language*, vol. 21(2), pp. 373–392, 2007.

[49] Z. Li and S. Khudanpur, "Large-scale discriminative n-gram models for statistical machine translation," in *Proceedings of the Association for Machine Translation in the Americas (AMTA)*, pp. 133–142, 2008.

[50] C. Chelba and F. Jelinek, "Structured language modeling," *Computer, Speech and Language*, vol. 14, pp. 283–332, 2000.

[51] W. Wang and M. Harper, "The SuperARV language model: Investigating the effectiveness of tightly integrating multiple knowledge sources," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 238–247, 2002.

[52] R. Rosenfeld, "A maximum entropy approach to adaptive statistical language modeling," *Computer, Speech and Language*, vol. 10, pp. 187–228, 1996.

[53] R. Rosenfeld, "A whole sentence maximum entropy language model," in *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 230–237, 1997.

[54] J. Darroch and G. Ratcliff, "Generalized iterative scaling for log-linear models," *Annals of Mathematical Statistics*, vol. 43(5), pp. 1470–1480, 1972.

[55] S. D. Pietra, V. D. Pietra, and J. Lafferty, "Inducing features on random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 380–393, 1997.

[56] R. Malouf, "A comparison of algorithms for maximum entropy parameter estimation," in *Proceedings of the Conference on Computational Linguistics (COLING)*, pp. 1–7, 2002.

[57] J. Wu and S. Khudanpur, "Efficient training methods for maximum entropy language modeling," in *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP)*, pp. 114–118, 2000.

[58] S. D. Pietra, V. D. Pietra, and J. Lafferty, "Inducing features of random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19(4), pp. 1–13, 1997.

[59] S. Chen and R. Rosenfeld, "A survey of smoothing techniques for ME models," *IEEE Transactions on Speech and Audio Processing*, vol. 8, pp. 37–50, 2000.

[60] J. Goodman, "Exponential priors for maximum entropy models," in *Proceedings of the North American Chapter of the Association for Computational Linguistics*, pp. 305–312, 2004.

[61] J. Wu and S. Khudanpur, "Maximum entropy techniques for exploiting syntactic, semantic and collocational dependencies in language modeling," *Computer, Speech and Language*, vol. 14, pp. 355–372, 2000.

[62] J. Bilmes and K. Kirchhoff, "Factored language models and generalized parallel backoff," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 4–6, 2003.

[63] K. Kirchhoff, D. Vergyri, K. Duh, J. Bilmes, and A. Stolcke, "Morphology-based language modeling for Arabic speech recognition," *Computer, Speech and Language*, vol. 20(4), pp. 589–608, 2006.

[64] K. Duh and K. Kirchhoff, "Automatic learning of language model structure," in *Proceedings of the Conference on Computational Linguistics (COLING)*, pp. 148–154, 2004.

[65] A. Stolcke, "SRILM - an extensible language modeling toolkit," in *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP)*, pp. 901–904, 2002.

[66] G. Ji and J. Bilmes, "Multi-speaker language modeling," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 137–140, 2004.

[67] G. Ji and J. Bilmes, "Dialog act tagging using graphical models," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2005.

[68] D. Vergyri, K. Kirchhoff, K. Duh, and A. Stolcke, "Morphology-based language modeling for Arabic speech recognition," in *the 8th International Conference on Spoken Language Processing (ICSLP)*, pp. 2245–2248, 2004.

[69] I. Zitouni, "Backoff hierarchical class n-gram language models: effectiveness to model unseen events," *Computer Speech and Language*, vol. 21, pp. 88–104, 2007.

[70] W. Wang and D. Vergyri, "The use of word n-grams and parts of speech for hierarchical cluster language modeling," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 321–324, 2006. 203

[71] P. Xu and F. Jelinek, "Random forests in language modeling," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 325–332, 2004.

[72] Y. Su and F. Jelinek, "Exploiting prosodic breaks in language modeling with random forests," in *Proceedings of the ISCA Workshop on Speech Prosody*, pp. 91–94, 2008.

[73] C. Andrieu, N. de Freitas, A. Doucet, and M. Jordan, "An introduction to MCMC for machine learning," *Machine Learning*, vol. 50, pp. 5–42, 2003.

[74] S. Wang, R. Greiner, D. Schuurmans, L. Cheng, and S. Wang, "Integrating trigram, PCFG and LDA for language modeling via directed Markov random fields," in *Proceedings of the NIPS Workshop on Bayesian Methods for Natural Language Processing*, 2005.

[75] B. Hsu and J. Glass, "Style and topic adaptation using HMM-LDA," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 373–381, 2006.

[76] T. Ferguson, "A Bayesian analysis of some nonparameteric problems," *Annals of Statistics*, vol. 1(2), pp. 209–230, 1973.

[77] Y. Teh, M. Jordan, M. Beal, and D. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, pp. 1566–1581, 2006.

[78] Y. Teh, "A hierarchical Bayesian language model based on Pitman-Yor process," in *Proceedings of the Association for Computational Linguistics*, pp. 985–992, 2006.

[79] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," in *Proceedings of Neural Information Processing Systems (NIPS) Conference*, vol. 13, 2000.

[80] H. Schwenk and J. Gauvain, "Neural network language models for conversational speech recognition," in *Proceedings of Interspeech: Annual Conference of the International Speech Communication Association*, pp. 2253–2258, 2004.

[81] H. Schwenk, "Training neural network language models on very large corpora," in *Proceedings of Human Language Technology Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pp. 201–208, 2005.

[82] A. Emami and L. Mangu, "Empirical study of neural network language models for Arabic speech recognition," in *Proceedings of IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, pp. 147–152, 2007.

[83] H. Schwenk, D. Déchelotte, and J. Gauvain, "Continuous space language models for statistical machine translation," in *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pp. 723–730, 2006.

[84] A. Emami and F. Jelinek, "A neural syntactic language model," *Machine Learning*, pp. 195–227, 2005. 284

[85] A. Alexandrescu and K. Kirchhoff, "Factored neural language models," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 1–4, 2006.

[86] S. Khudanpur, "Multilingual language modeling," in *Multilingual Speech Processing*, pp. 169–205, 2006.

[87] W. Byrne, J. Hajic, P. Ircing, and F. Jelinek, "Large vocabulary speech recognition for read and broadcast Czech," in *Text, Speech and Dialog. Lecture Notes in Computer Science*, vol. 1692, pp. 235–240, 1999.

[88] D. Kiecza, T. Schultz, and A. Waibel, "Data-driven determination of appropriate dictionary units for Korean LVCSR," in *Proceedings of the International Conference on Speech Processing (ICSP)*, pp. 323–327, 1999.

[89] H. Dutağacı, *Language Models for Large Vocabulary Turkish Speech Recognition*. PhD thesis, Boğaçizi University, 1999.

[90] M. Creutz, T. Hirsimäki, M. Kurimo, A. Puurula, J. Pylkkönen, V. Siivola, M. Varjokallio, E. Arisoy, M. Saraçlar, and A. Stolcke, "Analysis of morph-based speech recognition and language modeling of out-of-vocabulary words across languages," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 380–387, 2007.

[91] G. Choueiter, D. Povey, S. Chen, and G. Zweig, "Morpheme-based language modeling for Arabic LVCSR," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1053–1056, 2006.

[92] H. Soltan, G. Saon, D. Povey, L. Mangu, B. Kingsbury, J. Kuo, M. Omar, and G. Zweig, "The IBM 2006 GALE Arabic ASR system," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 349–352, 2007.

[93] T. Buckwalter, "Buckwalter Arabic morphological analyzer version 2.0." Linguistic Data Consortium (LDC) catalog number LDC2004L02, ISBN 1-58563-324-0, 2004.

[94] O. Rambow and N. Habash, "Arabic diacritization through full morphological tagging," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 117–120, 2007.

[95] Z. Harris, "From phoneme to morpheme," *Language*, vol. 31(2), pp. 190–222, 1955.

[96] M. Adda-Decker and L. Lamel, "Multilingual dictionaries," in *Multilingual Speech Processing*, pp. 305–322. Amsterdam: Elsevier, 2006.

[97] M. Creutz and K. Lagus, "Unsupervised models for morpheme segmentation and morphology learning," *ACM Transactions on Speech and Language Processing*, vol. 4, no. 1, pp. 1–34, 2007.

[98] M. Creutz and K. Lagus, "Induction of a simple morphology for highly-inflecting languages," in *Proceedings of the 7th Meeting of the ACL Special Interest Group in Computational Phonology (SIGPHON)*, pp. 43–51, 2004.

[99] M. Creutz and K. Lagus, "Inducing the morphological lexicon of a natural language from unannotated text," in *Proceedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*, pp. 106–113, 2005.

[100] M. Kurimo, M. Creutz, M. Varjokallio, E. Arisoy, and M. Saraçlar, "Unsupervised segmentation of words into morphemes: Challenge 2005. An introduction and evaluation report," in *PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*, 2006.

[101] E. Whittaker and P. Woodland, "Particle-based language modeling," in *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP)*, 2000.

[102] E. Arisoy, H. Sak, and M. Saraçlar, "Language modeling for automatic Turkish broadcast news transcription," in *Proceedings of Interspeech: Annual Conference of the International Speech Communication Association*, pp. 2381–2384, 2007.

[103] T. Alumae, "Sentence-adapted factored language model for transcribing Estonian speech," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2006.

[104] K. Kirchhoff, M. Yang, and K. Duh, "Statistical machine translation of parliamentary proceedings using morpho-syntactic knowledge," in *Proceedings of the TC-STAR Workshop on Speech-to-Speech Translation*, pp. 57–62, 2006.

[105] R. Sarikaya and Y. Deng, "Joint morphological-lexical language modeling for SMT," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 145–148, 2007.

[106] I. Oparin, O. Glembek, L. Burget, and J. Cernocky, "Morphological random forests for language modeling of inflectional languages," in *Proceedings of the IEEE Spoken Language Technology Workshop*, pp. 189–192, 2008.

[107] A. Ito and M. Kohda, "Language modeling by string pattern n-gram for Japanese speech recognition," in *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP)*, pp. 490–493, 1996.

[108] R. Sproat, T. Zheng, L. Gu, J. Li, Y. Zheng, Y. Su, H. Zhou, P. Bramsen, D. Kirsch, I. Shafran, S. Tsakalidis, R. Starr, and D. Jurafsky, "Dialectal speech recognition: Final report," Tech. Rep., CLSP, Johns Hopkins University, 2004.

[109] H. Tseng, P. Chang, G. Andrew, D. Jurafsky, and C. Manning, "A conditional random field word segmenter for SIGHAN bakeoff 2005," in *Proceedings of the 4th SIGHAN Workshop on Chinese Language Processing*, pp. 168–171, 2005.

[110] H. Zhao, C.-N. Huang, and M. Li, "An improved Chinese word segmentation system with conditional random field," in *Proceedings of the 5th SIGHAN Workshop on Chinese Language Processing*, pp. 162–165, 2006.

[111] X. Mao, Y. Dong, S. He, S. Bao, and H. Wang, "Chinese word segmentation and named entity recognition based on conditional random fields," in *Proceedings of the 6th SIGHAN Workshop on Chinese Language Processing*, pp. 90–93, 2008.

[112] Y. Song, J. Guo, and D. Cai, "Chinese word segmentation based on an approach of maximum entropy modeling," in *Proceedings of the 5th SIGHAN Workshop on Chinese Language Processing*, pp. 201–204, 2006.

[113] A. Jacobs and Y. Wang, "Maximum entropy word segmentation of Chinese text," in *Proceedings of the 5th SIGHAN Workshop on Chinese Language Processing*, pp. 185–188, 2006.

[114] D. Song and A. Sarkar, "Training a perceptron with local and global features for Chinese word segmentation," in *Proceedings of the 6th SIGHAN Workshop on Chinese Language Processing*, pp. 143–146, 2008.

[115] G. Jin and X. Chen, "The fourth international Chinese language processing bakeoff: Chinese word segmentation, named entity recognition and Chinese POS tagging," in *Proceedings of the 6th SIGHAN Workshop on Chinese Language Processing*, pp. 69–81, 2008.

[116] V. B. Le, B. Bigi, L. Besacier, and E. Castelli, "Using the web for fast language model construction in minority languages," in *Proceedings of Eurospeech: European Conference on Speech Communication and Technology*, pp. 3117–3120, 2003.

[117] R. Ghani, R. Jones, and D. Mladenic, "Building minority language corpora by learning to generate web search queries," *Knowledge Information Systems*, vol. 7, no. 1, pp. 56–83, 2005.

[118] J. Franco and T. Solorio, "Baby steps towards a language model for Spanglish," in *Proceedings of the 8th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2007)*, 2007.

[119] C. Fügen, S. Stüker, H. Soltau, and F. Metze, Efficient handling of multilingual language models," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, pp. 441–446, 2003.

[120] T. Ward, S. Roukos, C. Neti, M. Epstein, and S. Dharanipragada, "Towards speech understanding across multiple languages," in *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP)*, 1998.

[121] Z. Wang, U. Topkara, T. Schultz, and A. Waibel, "Towards universal speech recognition," in *Proceedings of the IEEE International Conference on Multimodal Interfaces (ICMI)*, pp. 14–16, 2002.

[122] F. Weng, H. Bratt, L. Neumeyer, and A. Stolcke, "A study of multilingual speech recognition," in *Proceedings of the Eurospeech: European Conference on Speech Communication and Technology*, pp. 359–362, 1997.

[123] S. Khudanpur and W. Kim, "Contemporaneous text as side information in statistical language modeling," *Computer Speech and Language*, vol. 18(2), pp. 143–162, 2004.

[124] A. Jansson, E. Whittaker, K. Iwano, and S. Furui, "Language model adaptation for resource deficient languages using translated data," in *Proceedings of Interspeech: Annual Conference of the International Speech Communication Association*, pp. 1329–1332, 2005.

[125] W. Kim and S. Khudanpur, "Cross-lingual lexical triggers in statistical language modeling," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 17–24, 2003.

[126] W. Kim and S. Khudanpur, "Cross-lingual latent semantic analysis for language modeling," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 257–260, 2004.

[127] Y. Tam, I. Lane, and T. Schultz, "Bilingual-LSA based LM adaptation for spoken language translation," in *Proceedings of the Association for Computational Linguistics*, pp. 520–527, 2007.

文本蕴涵识别

Mak Sammons, V. G. Vinod Vydiswaran, Dan Roth

6.1 概述

从2005年开始，研究人员就开始广泛地对文本蕴涵识别（Recognizing Textual Entailment，RTE）任务进行研究，这个任务在没有约束参与者使用特定表达或推理方法的情况下专注于提高通用文档推理的能力。这个自然语言处理的子领域已经获得了很好的发展前景，因为很多系统性能有稳定的提升并且该问题得到了广泛的研究。大量研究人员对问题本身定义的一些性质和解决这些问题的方法的特点展开了研究。RTE的解决方案在其他NLP应用和更具挑战性的自然语言理解（Natural Language Understanding，NLU）任务上有实际的使用价值。例如阅读学习[1]和机器阅读[2]已经出现，它们同样有相似的问题需要解决。让我们开始研究这个领域吧，这真让人激动。

在众多NLP任务中，特别是对于那些能够从集成的背景知识中获益的任务，文本推断能力的高低对系统性能起关键作用。问答系统有潜在可能成为下一代搜索引擎，但是它本身具有局限性，特别是在处理非事实性问题的时候。并且，对人类而言，从一系列纯文本文档（例如新闻类文章）中提取出感兴趣的事实（例如，"某个在公司X工作的员工"）包含深度抽象、综合以及常识的应用三个过程。因此，对于软件而言，它也同样需要执行这些过程。

在这一章中，我们会明确一个框架，在这个框架下我们能设计和构造一个RTE系统。首先，我们定义一个RTE问题，然后概要说明它在NLP其他任务上的应用。接着我们为RTE定义一个框架，并且展示它是如何融入那些已经在成功的RTE系统上使用的技术。我们还会描述在RTE领域的关键研究（主要集中于系统开发），并且还会展示每个系统如何关联到我们所定义的框架上。最后，我们陈述了在RTE研究上的紧迫挑战和一些有用的资源。

我们假定读者已经熟悉机器学习和它的一套训练、开发和测试方法的基本思想；我们关注的焦点在于为RTE开发一个应用时遇到的实际困难。

我们为RTE框架的所有关键步骤提供一些简单的算法。尽管它们被故意简化了，因此可能效率不是特别高，但用来构造一个基本的RTE系统还是足够的，并且我们可以对它进行多个维度的扩展。6.4节讨论了针对RTE不同方法的关键研究，我们在一个高层面上将每一个研究方向都映射到我们的框架上（完整的应用细节超出了本章的范围，请参考引用文献的工作）。这种映射允许我们开发与系统相关的其他方面，从而实现对我们最感兴趣的方法的跟踪研究。

6.2 文本识别蕴涵任务

这一小节定义RET任务，解释这一定义的优缺点，并表明这个问题重要的原因。我们展示了RTE如何应用在一系列的NLP任务中，并介绍这些应用的一些具体例子。

6.2.1　问题定义

在这一章中，我们所解决的 RTE 任务的形式化描述是由 Dagan、Glickman 和 Magnini [3] 定义的，具体如下：

定义 6-1　文本蕴涵（textual entailment）　是指文本对之间的指向关系，用符号 T 表示蕴涵的文本，用 H 表示被蕴涵的文本，也就是假设。如果 H 的意思能够从 T 中推导出来，那么就认为 T 蕴涵 H，因为这样很容易被人们理解。

研究人员指出这个不太正式的定义是基于（和假定）普通人既理解语言又知道一些常用的背景知识。

一个蕴涵对由一个文本 T 和一个假设 H 组成；通常，H 是一个短文本，而 T 是一个跨度更大的文本。图 6-1 展示了一个实例文本和三个假设。每个蕴涵对的标签是由多个人工标注完成的；在标注过程中我们并不明确要求有背景知识，不过它仍然是一个潜在因素。通常我们获取的知识是静态的，例如因果关系或著名城市和地标的位置（这些知识不会随着时间变化），而不是一些会随着时间变化的事实，例如现任美国总统的名字。

RTE 任务的规范也要求将文本作为推断一个假设是否为事实的固有组成部分；尽管背景知识可以扩充文本表示的内容，但不能完全取代它。我们举个例子，如果一个 RTE 系统使用了从维基百科抽取的一些事实，这些事实可能包含一个声明，该声明确定了一个当红电影明星的国籍，这其实就等同于一个假设声明。然而，如果在文本中没有出现支持这一事实的证据，那么即使假设本身是一个“真实”的事实，蕴涵标签也会被标记为非蕴涵。

一个二元的 RTE 任务要求系统将每个蕴涵对标注为蕴涵或不蕴涵，也就是说标注 T 蕴涵 H，或者 T 不蕴涵 H。在图 6-1 中，文本蕴涵假设 1，但是既不蕴涵假设 2 也不蕴涵假设 3。

210

> 文本：BMI 以 20 亿美元购买总部位于休斯顿的 LexCorp 公司的行为引起了证券交易商大规模抛售，因为他们寻求将风险降至最低。自 2008 年以来，LexCorp 一直运营工持股的公司。
> 假设 1：BMI 收购了一家美国公司。
> 假设 2：BMI 花费了 34 亿美元购买了员工持股的 LexCorp 公司。
> 假设 3：BMI 是一家员工持股的公司。

图 6-1　一些有代表性的 RTE 实例

三元 RTE 任务引入了矛盾的概念。我们基于 Marneffe、Rafferty 和 Maning [4] 来定义蕴涵中矛盾的概念。

定义 6-2　如果一个人能够根据文本 T 表述的关系或事件推出 H 描述的关系或事件有很大可能是假的，那么我们认为一个蕴涵对中的假设 H 与文本 T 是矛盾的。

一个三元的 RTE 任务要求系统将每个蕴涵对标注为蕴涵、矛盾或未知，也就是说，或者 T 蕴涵 H，或者 H 与 T 相矛盾，或者给定 T 时 H 是否为真未知。在图 6-1 中，文本 T 蕴涵假设 1；假设 2 与 T 相矛盾；从 T 所给出的信息中，无法得知假设 3 是否为真。

任务的难易取决于所选定的蕴涵对，并且设计一个合适的语料库并不容易。PASCAL（Pattern Analysis，Statistical Modelling and Computational Learning）[1] 和美国国家标准及技术研究所（National Institute of Standards and Technology，NIST）[2] 在语料的制作过程

---

[1] http://pascallin.ecs.soton.ac.uk/Challenges/RTE3/。

[2] http://www.nist.gov/tac/2010/RTE/index.html。

中面临很大的挑战。除了 RTE 4 外所有的语料都有独立的开发和测试部分，每一个部分都包含 600~800 个蕴涵对；RTE 4 只有一个部分，包含 1000 个蕴涵对。所有这些语料都是均衡的，大约 50%的标签属于蕴涵，50%的标签不属非蕴涵。在 RTE 4 和 RTE 5 中，非蕴涵的实例进一步被划分为两类：未知的和矛盾的（分别占全部实例的 35%和 15%）。

每个语料定义了 3~7 个任务来进一步划分这些数据。每个任务与一个领域相关联，该任务的实例从这个领域中获得（例如，问答系统 QA、信息抽取 IE）。一些文献描述了面对每个挑战的更多细节，例如 Bentivogli 等 [5]）。系统的性能随着任务的变化而变化，表明不同任务的实例之间有显著的质量差别，但是因为任务标签在部署的 RTE 应用中不可用，所以我们在这里不考虑它。（如果存在任务信息，那么利用它来扩展这里描述的框架实现就很容易，扩展要么引入一个特征来表示该任务，要么在每一个任务中分别使用调参和训练过的推理组件。）

除此之外，在 RTE 3（解释，矛盾）和 RTE 5（搜索）中引入了一些试点任务，矛盾任务是 RTE 4 和 RTE 5 中主要任务的一部分。RTE 5 同样引入了一个搜索试点任务，这里我们就不详述了（更多细节请参考 Bentivogli 等 [5]）。 211

在这些语料上性能较好的系统可以认为是能够很好地“理解”自然语言文本[1]。针对两个最新的挑战赛（RTE 4 和 RTE 5），性能最好的系统在二元任务（蕴涵与非蕴涵）上能达到 74%的正确率，在三元任务中达到 68%。

这章的剩下部分我们要确定 RTE 任务中涉及的挑战，定义一个通用的框架来处理它，并描述 RTE 中的相关研究，显示它如何融入这个框架。

6.2.2 RTE 的挑战

考察人类在决定蕴涵对的蕴涵标签时所采取的不同步骤是非常有启发性的，如图 6-1 所示。

为了识别文本蕴涵假设 1，人类读者必须识别以下 4 点：1）假设中提到的公司能够匹配 LexCorp；2）位于休斯顿略指美国；3）识别象征性关系购买；4）判断“A 被 B 购买”暗指“B 拥有 A”。

要识别出假设 2 与文本矛盾，需要与上述类似的步骤。不同的地方在于，读者必须整合以下信息，首先 LexCorp 是一个员工控股的公司，其次必须能够推理出尽管文本和假设中的购买价格不同，但是它们指向同一笔交易的概率是非常高的，因此假设 2 与文本矛盾。

假设 3 包含了文本的全部文字，但是断言了一个无法从已知的证据中识别出来的关系，所以它的标签是未知：有可能 BMI 是一家员工控股的公司，但也有可能不是。

这些步骤中，有些与 NLP 或者计算语言学社区定义的其他任务相关，例如命名实体识别（识别 LexCorp 和 BMI 是公司）、共指（LexCorp 的不同表示指的是相同的内在实体）、语义角色标注（是 BMI 购买，而不是 LexCorp）。其他的步骤可能不相关。相关任务还没有独立地取得好的进展，尽管它们与已有问题的定义相关。文本推理步骤可能是所有步骤中最难的，因为它需要利用我们对这个世界的理解来识别因果关系、蕴涵关系，并将多个语句抽象成一个通用的原则。

尽管同计算机的方法来应对 RTE 的挑战并不需要按照这些步骤或模仿这种能力，但由于不使用人类处理过程的系统只取得了有限的成功，这激励研究者尝试由人类处理过程

[1] 这个假设基于标准的机器学习评测方法，即评测系统性能时只使用不在训练及开发集内的数据。

的直觉激发的分治方法。在分离一些特别能力上，研究者已经获得了一些成功，例如归一化数字数量（日期、速率、比例、计数），从语言学的角度借助一些解决方案来促进问题的解决，例如语法分析器和诸如命名实体识别等浅层语义分析工具。

212 也许有人争论图 6-1 所示的例子有可能通过简单的词汇匹配来解决，但是很明显，文本可变为和假设 1 在用词上截然不同，同时假设 1 仍保持蕴涵关系。相反地，可使文本和假设 2 在词汇上重叠非常多，同时保持它们之间互相矛盾的关系。这种直觉由 RTE 挑战赛的结果所证实，即使用其他的、更结构化分析的系统的性能可以超过基于词汇相似度的系统，细节将在 6.2.3 节介绍。

6.2.3　评估文本蕴涵系统性能

定义 6-1 被 PASCAL 和 NIST 作为 6 项研究挑战赛的基础。这些语料库向公众开放，其中前 3 项不含限制，接下来的 3 项需要接受用户许可（详情查看先前类到的网址）。第 3 次挑战赛，即 RTE-3，按照定义 6-2 设置了一个试验任务。RET-4 和 RTE-5 主任务的语料库中都引入了矛盾信息，因此同时按照二元和三元预测任务进行标记[⊖]。

这些研究挑战赛提高了人们对 RTE 问题的兴趣度并使对该问题的研究有了长足的发展。我们在 6.4 节中给出了一些有用的例子，现在我们对最高水平系统的性能做一般介绍。

图 6-2 表明 2005 年对所有 5 个 RTE 挑战赛的二元蕴涵任务的结果所做的概要。为了便于比较，对于每一个数据集的词法基准系统（来自 Mehdad 和 Magnini [6]）的性能也一同给出。

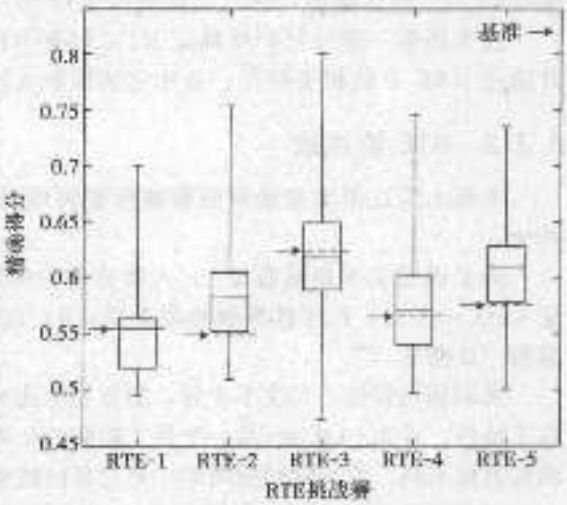


图 6-2　2005—2009 年的 PASCAL 文本蕴涵识别挑战赛二元任务的结果。在每一年的挑战赛中，我们给出了参与系统准确度的五点统计结果，另外也给出了词法基准系统的结果

由于每年的语料（来自于不同的领域或按照不同的大纲）不同，所以很难比较不同年份的结果。RTE-4 和 RTE-5 的平均文本长度有显著的增加，各自增加了 40 和 100 个单词，这相对于短文本的蕴涵对更具有挑战性。词法基准系统使用了基于假设和文本词之间重合度的阈值，在到目前为止的四～五个挑战赛中，它获得了 55%～58%的结果。这些结果相当一致。RTE-3（2007）的结果明显很高，并且所有的参赛系统与其他年份相比都有提高，表明这是一个“更容易”的蕴涵语料库。在所有的情况里，基准分数小于或等于每一个挑战赛的中等分数。

213

系统性能的上限也相当一致。RTE-4 和 RTE-5 中使用的更长的文本增加了任务的难度。更长的文本会引入更多无关信号（额外的与蕴涵决策无关的单词、短语和句子）增加了 RTE 系统的处理负担，并且扩大了蕴涵样例的范围，更广的范围需要整合多个句子的信息。

由于 RTE 实验数据集的不兼容性，除了根据相对较好的系统的性能来估计评测任务的难度以及观察一些系统是否明显超过基准系统外，我们很难在数字上得出强有力的结论。

⊖　在本章写作的时候，RTE-6 正在进行。

6.2.4 文本蕴涵解决方案的应用

许多NLP问题可以按照文本蕴涵识别的方式形式化。RTE与自动摘要有明显的相关性[7]，自动摘要系统需要从一个或多个文档中摘要出人类可读的信息。在判断一个新句子包含的信息是否已经被正在进行的摘要表达过（冗余检测）时，该子任务可以被认为是一个蕴涵对，它的当前摘要可以理解为文本，新的句子可以理解为假设。如果T没有蕴涵H，那么该句子就包含新的信息并且会集成到摘要中去。

信息提取的任务是在一组自然语言文本文档中识别一个固定关系集合中的实例，例如“为什么工作”和“在哪里出生”。如果我们用较短的句子表达关系，例如“一个人为某组织工作”和“一个人出生于某地”，那么源文档中的文本便成为了蕴涵对里面的文本，重新表达的关系便成为了蕴涵对里面的假设。这样RTE系统就可以直接应用了。类似地，要求系统自动地发现候选答案（来自于固定的文档集合中的文档章节）的问答系统，同样可以用相同的方法来重新表达：问题如“美国南部最大的城市是什么？”可以重新表达为一个短的陈述句：“美国南部最大的城市是一个城市”。这句话变成了假设，并且文档集合的部分内容——典型的是段落——成为了包含该假设的蕴涵对集合的文本。一个RTE系统能够直接用于识别真实的答案。

当然，在信息提取和问答任务中，这些外素的重新形式化的方法和问答任务是不充分的，因为RTE的解决方案通常要求密集型的数据。然而，直觉是可行的，研究人员已将RTE应用到其他NLP任务上。 [214]

1. 问答系统

Harabagiu和Hickl[8]直接应用RTE系统来对问答系统中的候选答案进行重排序，基本思想很简单：在一个已经存在的问答系统中返回最佳的候选答案。虽然最佳候选可能不是正确答案，但是在许多情况下正确答案位于返回的候选答案集合中。

Harabagiu和Hickl使用RTE系统来评估每一个候选答案。如前面所说的那样，他们的系统首先采用一个基于规则的方法将输入的问题转换为一个简短的陈述句。然后创建一个蕴涵对集合，其中将系统返回的候选答案合并作为蕴涵对的文本，将转换的问题作为蕴涵对的假设。RTE系统接下来依次应用在每一个蕴涵对上，那些能够蕴涵转换文本的候选蕴涵对移到列表的顶端，非蕴涵的则移动到底部。研究表明，添加文本蕴涵的组件能够将系统的准确率从30.6%提升至42.7%。

将查询用类似Harabagiu和Hickl的方法进行转换后，Celikyilmaz、Thint和Huang[9]使用类似蕴涵的组件来提取候选问题-回答对基于特征的表示。他们使用从蕴涵对比中得到的实值特征向量来计算大集合中问题-回答对间的相似度值，这些值用作图中连接表示问题-回答对的节点的边的权重。问题-回答对的（少量）子集拥有正确答案的标签，剩余节点的标签需要使用半监督学习方法得到。

2. 关系的穷举搜索

在许多信息采集任务中，如专利搜索、事故报告挖掘、在与合作者分享的没有经过整理的文档中检测秘密信息，有必要找到所有与给定的概念相关的文本片段。这涉及寻找直接或间接讨论该概念的所有段落，并过滤掉那些表面上相似，但实际上有着不同意义的段落。

这种信息需要直接映射为从大规模文本语料库中识别蕴涵的段落。然而，它需要扩充文本蕴涵系统，从而将成对的文本假设决策转换为基于搜索的蕴涵框架。因为大多数成功的RTE系统使用大量的NLP资源和计算复杂度高的推理算法，朴素的方法（对每个文档

的每个段落，测试它是否蕴涵表示目标信息陈述的集合中的任何一个）是不切实际的。

Roth、Sammons 和 Vydiswaran [10] 定义了一个集中的文本蕴涵的方法，SEER（Scalable Entailment Relation Recognition，可扩展蕴涵关系识别），分为两个阶段：语义检索和蕴涵识别。图 6-3 为该方法的示意图。

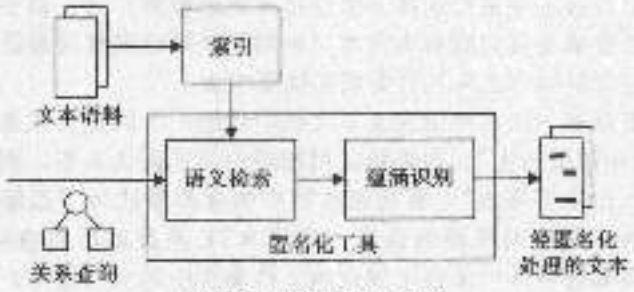


图 6-3　SERR 框架 [10]

该算法在图 6-4 中简述。在这个方法中，文本语料库先进行预处理，以发现语义成分，如命名实体（人、地点、组织、数字数量等），为了方便快速检索，它们作为语义单元被索引。用户表达的信息需求作为一个关系查询，可以利用同义词、替代名称以及其他类似的关键词的语义来扩展。这个查询接下来用于从语料库检索文本段落。根据蕴涵模块的处理结果来判断文本是否蕴涵给定的查询，并将蕴涵文本片段作为输出的结果，语义检索有助于提高蕴涵文本片段的召回率，RTE 模块对结果进行过滤从而提高整体的准确率。

将 RTE 1、2 和 3 中的信息检索和信息抽取的子任务的所有假设作为我们实验评测的语料，作为信息的需求。来自相同的蕴涵对的所有文本构成一个文档集合。检索组件找到每一个假设最相关的文档，并且为了识别相关的文档，RTE 模块将这些返回的结果标记为蕴涵或者非蕴涵。

当使用与 RTE 挑战赛中实际样例相同的假设-文本对来评估整体分类性能时，结果表明系统在每个挑战赛中都能位列公布结果中的前三名。这个体系结构也降低了比较的计算次数，相对于朴素方法的 3 800 000 个（使用 RTE 模块对所有的假设和文本进行比较），SERR 系统降低到了仅有 40 000 个。

```
SERR 算法
设置：
输入：文本集 D
输出：D 上的索引 {I}
对每个文本 d∈D
用命名语义内容标记 d
构造 D 上的索引 {I}
输入：信息需求 S

扩展的词汇化检索（ELR）(s)：
  R←∅
  用语义相似的词来扩展 s
  从 s 建立检索查询 q_s
  R←q_s 的前 k 个初文本的索引 {I}
  返回 R

SERR：
  答案集 A←∅
  for each 查询 s∈S
    R←ELR(s)
    答案集 A_s←∅
    for each 结果 r∈R
        使用 NLP 资源标记 s_r+
        如果 r 蕴含 s
            A_s←A_s∪r
A_s←A∪{A_s}
```

图 6-4　SERR 算法 [10]

3. 机器翻译

由 RTE 研究人员开发的技术也应用到了机器翻译（MT）方面的评测任务中。Padó 等 [11] 借鉴了文本蕴涵的思路提出了新的侯选翻译质量的自动评估方法。机器翻译评测使用统计方法来评估机器翻译系统给出的翻译结果和人工给出的参考译文的相似度，因为利用人工去评估大量的机器译文需要耗费巨大的资源，这样系统在大规模语料上的性能很难得到快速评价。主流的相似度计算方法是基于 n 元组的方法，这种方法与人工判断的结果有合理的

215
|
216

相关性，但距离完美还有很长的距离，尤其是因为该方法没有考虑待评测译文的全局结构。

Padó 等想出了一种能够考察结构特点的新方法，该方法使用与 Chambers 等 [12] 开发的文本蕴涵系统中相似的特征，他们的直观想法是：候选译文与参考译文应该是复述关系，因此二者应该互为蕴涵关系。在候选译文中丢失了信息就意味着它不能蕴涵参考译文，而候选译文中添加的信息就意味着参考译文不能蕴涵候选译文。质量差的翻译会引起两个方向上的蕴涵匹配失败。他们使用基于对齐分数、形态、极性和时态不匹配、语义关系、实体和日期兼容性以及其他信息的特征。

为了评估这种新方法，他们使用了来自 MT 研讨会上面的数据。根据标准度量来进行人工判断，结果表明 Spearman 相关系数有了显著提升。

Mirkin 等 [13] 使用蕴涵来翻译未知的术语。当一个术语相对不常见，抑或是从资源稀缺的语言进行翻译时，那么这个词可能不会出现在机器翻译系统使用的短语表中。Mirkin 等利用词法蕴涵规则将源译文转换成更加通用的形式来解决这个问题，他们证明了这个方法的可行性，采用的方法是使用了一个在法语或英语的平行语料中训练得到的机器翻译模型。接下来他们将这个模型应用到包含很多未知术语的英文新闻的句子中，这些句子来自于不同的领域，与训练模型的数据领域不同。

使用英语作为源语言使得他们能够使用 WordNet [14]，它是一个大规模的英语本体库，通过同义关系、上位关系和很多其他词法关系建立词的连接。他们使用同义关系来生成未知词的复述并利用上位关系从英文句子中生成蕴涵（更加通用）文本。然后将这些带有未知词的句子的不同版本的法语译文质量与仅使用更加标准的复述资源的译文质量做对比。

结果表明基于文本蕴涵的方法比基于复述的方法在对未知术语的覆盖方面有高达 50% 的提升；翻译质量相比于忽略未知术语的时候也有所提升，其中有额外的 16.8% 的机器译文能够被用户接受，而正确翻译的数目仅下降了 2.7%。

6.2.5 其他语言中的 RTE 研究

当今，很少有非英语的蕴涵语料库，其中两个非英语 RTE 数据源是 EVALITA[㊀] 和 CLEF（Cross-Language Evaluation Forum）[㊁]。意大利特兰托的 FBK-Irst 举行的自然语言处理评测程序评测了包含 RTE 在内的多种意大利语问题的自然语言处理技术。CLEF 的问答验证测试使用 RTE 的形式化来推动问答系统技术。CLEF 开发了一个语料库，将候选回答以及表达成陈述句的问题组成对，其思想是 RTE 系统可以根据每个候选答案是否蕴涵重新表达的问题来检测到正确答案。CLEF 有德语、英语、西班牙语、法语、意大利语、荷兰语和葡萄牙语的语料库。

NLP 社区在为其他语言开发的、能与英语比美的 NLP 资源中取得了稳定的进展。欧洲语言资源协会（European Language Resources Association）[㊂] 以及亚洲自然语言处理联盟（Asian Federation of Natural Language Processing）[㊃] 提供了很好的信息源。但是也有不少语言尚未建立自己的词性标注器和语法分析器，这需要研究人员在研究蕴涵时使用更浅的信息。

我们在 6.3 节中提到的框架需要一个特定的假设：当有多个语料资源时，我们假设它们在确定单词边界之间时是一致的。但是在现实中，即使是英语也可能在原始输入文本的

217

㊀ http://evalita.fbk.eu/te.html。
㊁ http://nlp.uned.es/clef-qa/ave/。
㊂ http://www.elra.info/。
㊃ http://www.afnlp.org/。

分词中出现不一致，形态丰富的语言，比如阿拉伯语，同一个词可以分割出不同的前缀和附着语素。德语把各种词拼成复合词也给单词边界的确定带来困难。中文不使用空格进行分词，但是机器翻译系统这样的 NLP 应用把它们组织成类似词的形式。

没有一成不变的解决方案，在下文所述的框架中，开发者必须按需要来确定分词策略，并且确定不同级别的表示能够与所选择的分词方法相适应。如果使用了带有冲突的分词方法的资源，开发者必须令人满意地解决这个问题。假设满足需要的条件，开发者可以根据已有的资源，按照我们的框架来实现合适的解决方案。

6.3 文本蕴涵识别的框架

本节我们将定义一个灵活的 RTE 程序的框架，我们将借鉴 Rath 和 Semmons [15] 的观点。详细地描述任何一个真实 RTE 系统的实现都要用一整章，因此，我们只描述系统，在合适的时机给出样例算法。在 6.4 节，我们会给出相关的研究和已经公布的具体实现的细节，并且讲解这些实现如何适应我们的框架。

我们在这里讲的框架旨在用统一的方法融合已有的（以及新的）NLP 资源，能够系统地开发直接且复杂的 RTE 系统。框架的另一目标是可以直接支持多种 RTE 方法的实验，如在 6.4 节中描述的研究人员提供的方法。

在本章的最后，我们提供一些有用的下载地址的不完全列表（大部分都有非商业用途的许可）。但是，我们避免讲解任何一种具体的实现。我们关注在已有任务中表现良好的系统，并且期望这些系统能够输出一致的结果。因此你可以选择使用最适合你需求的特定系统。

6.3.1 要求

在真正开始设计 RTE 系统的框架之前，先考虑有什么已有的自然语言处理组件能够对识别蕴涵有好处。我们关注能够对 RTE 系统有明显帮助或者能够作为 RTE 系统有用功能的基础 NLP 模块，我们只考虑具有广泛认同的输出格式的 NLP 组件。

我们对两种资源特别感兴趣：能够为文本添加语义信息的资源，如命名实体标注器与语法分析器。另一种是能够比较文本区间，如词、名称和短语等的资源，并且可以给出一些相似的度量。我们将前者称为**标注器**（或者分析器），后者为**比较器**，或者**度量**（参见 6.3.3 节）。

再次考虑图 6-1 中的例子以及人类推理时遵循的步骤，以此来指导我们的系统所具备的能力。第一步，要识别出词“BMI”和“LexCorp”是两家公司的名字，这个信息要交给命名实体识别器来做（参见第 8 章）。第二步，将“位于休斯顿”联系到“美国”，这需要一个至少在词级别的事实知识库。第三步，要识别规范化关系“购买”，首先要将该词识别为名词（通过词性标注器），需要一个词典来把任意的动词映射到它们的一般形式上。词典的一个可能选择便是流行的本体词典 WordNet [16]，用“派生于”这种关系来识别动词“购买”。下一步，为了与结构“BMI 收购了一个美国公司”相比较——这个结构也可以用句法或依存分析器获得（参见第 8 章），需要解释句法结构来识别主语、宾语以及直接宾语——规范化动词的论元。另一种选择，可以通过浅层语义分析器或语义角色标注器（参见第 4 章）来进行该步骤。最后在第四步，为了识别文本蕴涵假设，必须将两个句法（或浅层语义）结构进行比较。

当然其他资源也会很有用，一个成功的 RTE 系统中还包括：

- 识别以及规范化数词。
- 识别同一给定的命名实体的不同表达（比如国际商业机器公司指的是 IBM 而不是 BMI）。

• 确定文本中有哪些实体实际指向同一实体（也称为共指消解，在第8章中描述）。

每个RTE实现基本上都采用已有系统以及处理同样问题的自建模块的混合。

这就意味着一般的RTE框架要在各种粒度（从单词、短语到句子）上支持各种文本蕴涵对的标记，还要使用特定资源来比较这些标注。

6.3.2 分析

之前描述的大量NLP资源都假定自然语言理解任务（大体上）是可以分解的：我们可以分离成单独的问题，然后一个一个处理这些问题。我们从手工标注者解决语言蕴涵的步骤中得到启示，正如6.2.2节所述。计算机科学中其他领域的经验已经表明了分治思想的强大功能，因此将用这种思想来引导我们解决RTE问题。

6.3.3 有用的组件

在这里我们描述一些对RTE框架十分有用并且广泛使用的组件。

1. NLP分析的多视图表示

我们用分析器得到的输出来定义处理文本的成分，这些成分可以通过关系相互连接。我们把成分和关系的任意一种模式称为**结构**，每个成分也可以算作一种简单的结构。

我们用每个分析器资源来定义它自己的处理文本的**视图**。当然最基本的视图就是Word视图。我们要求Word视图由词元表示而非原始文本，其他的视图都按照Word视图进行规范化。因此，每个成分必须精确对应一个词索引集合，这对于检测不同视图间的对应关系而言是非常方便的（比如，识别语义角色标记论元也是一种命名实体）。 220

图6-5表示了系统中输出的结合命名实体（NE）、数词（NUM）、语义角色标记（SRL）

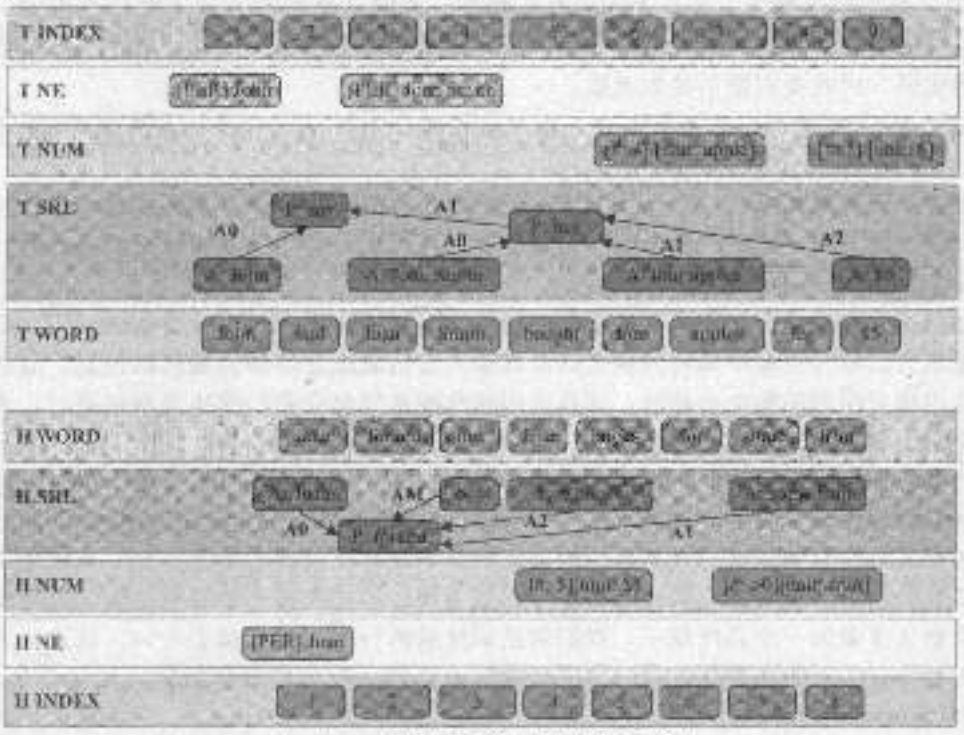


图6-5 文本蕴涵对的多视图表示示例

分析、单词以及它们的索引生产的数据结构。每一个成分都对应着原文中的单词，并且包含对应单词的索引表。

一般来说，成分指定了类型（从已有的相似度度量进行选择），一个或多个属性-值对表明了感兴趣的信息——如词的词性以及原形，它们可以用于相关相似度度量。属性-值对还表明了成分所对应的原始文本中词的索引。在这个例子里，NE 成分的类型和值都是从一个命名实体识别器的输出中得出来的。

NUM 成分是对数量以及其单位的规范化表示，还带有原文中对应词元的索引。

SRL 视图中有谓词（P）和论元（A）成分，通过代表谓词相关论元角色的关系来进行连接（A0 是 agent，或者语义主语，A1 是 patient，或者语义宾语）。这些角色与句法上的主语或宾语角色不同，因为它们不受如被动语态等的影响（详细的语义角色标注见第 4 章）。
221 注意在我们的表示中，嵌套是允许的。一个谓词可以作为另一个谓词的论元（在这种情况下，比如谓词 say 以谓词 buy 作为它的语义宾语）。论元成分没有分配角色，这是因为一个论元可能与多个谓词有谓词-论元结构关系，而且在不同的关系中它的角色也可能不同。

多视图表示比统一、单视图的表示有许多优势。每个资源都可以独立于其他资源进行处理并且可以增量式添加。这种表示十分灵活：如果我们想因不同目的而使用不同信息源——正如过滤时的情形（参见 6.3.7 节）——这很直接。也使得编写处理多视图的一般算法成为可能，而无须知道存在哪种视图。最后，多视图表示法延迟了规范化阶段：把不同的视图融合到一种数据结构中，就要解决边界和关系结构不一致的问题，将这些决策延迟到后面的阶段是不错的选择——例如，在推理阶段，当有足够的证据来支持一个决策时。如果需要，可以在预处理的最后一步将多视图结构收缩成一个图结构。

2. 比较标记成分

RTE 的关键步骤之一就是比较文本和假设。给定已经合并的大量不同信息源时，我们需要特殊的资源来比较某些类型的成分。如果将这些资源用统一的方法对待，就可以简化我们的实现。因此我们使用抽象度量：

定义 6-3　度量会比较两个成分，比较后返回一个在［－1，1］上的实数，1 表示相同，－1 表示相反，0 表示不相关。

度量是比较器概念的一个具体化。比较器能比较两个结构并且返回任意信息。一个度量比较两个成分而只返回一个分值。比较器更加专门化，是为特定的结构而设计的（比如语义角色标注导出的谓词-论元结构）。

注意这里度量的定义限制了上下文的使用，只使用要比较的成分类型的知识。生成用于创建成分的解析资源中编码的信息以及将输入分析成成分的算法编码的信息。我们把度量看作为相对简单的集中的资源，并且选用抽象度量以允许我们描述简单的接口，因此可以简化图生成代码。

当我们考虑将一个新的信息源添加到已有的，或许复杂的 RTE 系统之中时，这种设计选择的原因便非常显然：理想情况下，我们希望能够避免重写我们的图生成和对齐算法。如果我们编写的能够处理新标记的新比较器与原有比较器的规范一致，我们便无须重写这些算法。另一个局部化的原则在于可以以方便的形式来提升领域相关知识的封装性。

为给出度量的一个具体例子，我们描述词度量的行为（参见算法 6-1）。给定一对原形为 rise 和 increase 的词成分，我们的度量应该会返回一个比较高的数值，比如 0.8，因为
222 rise 和 increase 在某种上下文中是同义词。给定 paper 和 exterminate，它应该返回一个接近 0 的值，因为这两个词没有什么关系。如果给出 rise 和 fall，就应该返回一个接近－0.7

的值，因为它们是反义词（我们使负数的绝对值较小，这样我们在对齐阶段就更可能选中正数，但是这仅仅是出于我们对期望行为的直觉而非经验）。

分值的决定现在更多的是一种艺术而非科学，我们将它们用实数来表示，这样在推理中能够保持灵活性。例如，我们发现，使用不完美的实值词相似度分值的基准系统，相比于使用按阈值进行分配 1.0 和 0.0 的相似度分值的基准系统，能够在实验中有更好的结果。

算法 6-1 词度量的算法。函数 levenshteinDistance() 计算两个词之间的编辑距离。函数 isSynonym() 查询 WordNet，并且如果两个词是同义词就返回 true，否则返回 false。函数 isHypernym() 查询 WordNet，并且返回将这两个词分隔开的上位关系层数（如果没有相同的上位就返回无穷）

```
// 假定：词条为小写

compare( firstWordC, secondWordC )
  score ← 0
  firstWord ← getAttribute( firstWordC, WORD )
  secondWord ← getAttribute( secondWordC, WORD )

  if ( firstWord == secondWord )
      score ← 1.0
  else
      levDistance ← levenshteinDistance( firstWord, secondWord )
      numChars ← max( firstWord.length, secondWord.length )

      if ( ( numChars − levDistance ) / numChars > 0.9 )
          score ← 0.8
      else if ( isSynonym( firstWord, secondWord ) )
          score ← 0.8
      else if ( isAntonym( firstWord, secondWord ) )
          score ← −0.7
      else
          numHypernymLinks ← isHypernym( firstWord, secondWord )
          if ( numHypernymLinks < 4 )
              score ← (0.9/numHypernymLinks)
  return score
```

一般情况下，一些度量分值需要进行调整。例如，有些命名实体之间的相似度使用字符串编辑距离的变种，它们对于非常不同的名字倾向于返回一个适中的正值。然而，基于 WordNet 的词相似度度量，当两个词通过许多上位步骤才能关联时，会返回一个相对低的正值。这种情况与两个词之间具有蕴涵关系时的情况相近，而不同于具有相近字符串编辑距离分值时两个实体的情况。 223

一般地，度量并不对称，因为蕴涵关系不对称。考虑如下情况，该度量应用图 6-1 中蕴涵例子的名词短语进行比较。必须比较的名词短语对之一是由文本中的 a company（一个公司）以及假设中的 an American Company（一个美国公司）组成。在这种情况下，文本并不包含足够的信息，因此名词短语度量的返回值应该为 0。然而，如果 an American Company（一个美国公司）在文本中，a company（一个公司）是在假设中，那么度量的返回值应该接近于 1.0，因为前者蕴涵后者[⊖]。

命名实体度量应该能够识别 John Q. Smith 有很大的可能蕴涵 John Smith 和 Mr. J. Smith，而非蕴涵 Ms. J. Smith。同样，这种关系并不一定对称，如 John Smith 并不一定蕴涵 John Q. Smith。

⊖ 额外数据对（在这种情况下是文档）对蕴涵的作用称为单调性。关于蕴涵和单调性的讨论，参见 MacCartney 与 Manning [17]。

6.3.4　通用模型

图 6-5 是一个典型的 RTE 系统的框图，蕴涵关系是一次或者批量进行处理。为了简单起见，我们描述一次处理一对的过程，而某些特定情况下需要批处理模式。我们用系统的评测来描述系统（这与使用的 RTE 系统的行为相对应）。我们单独处理训练机器学习组件的过程，尽管该过程通常多次使用相同的步骤。

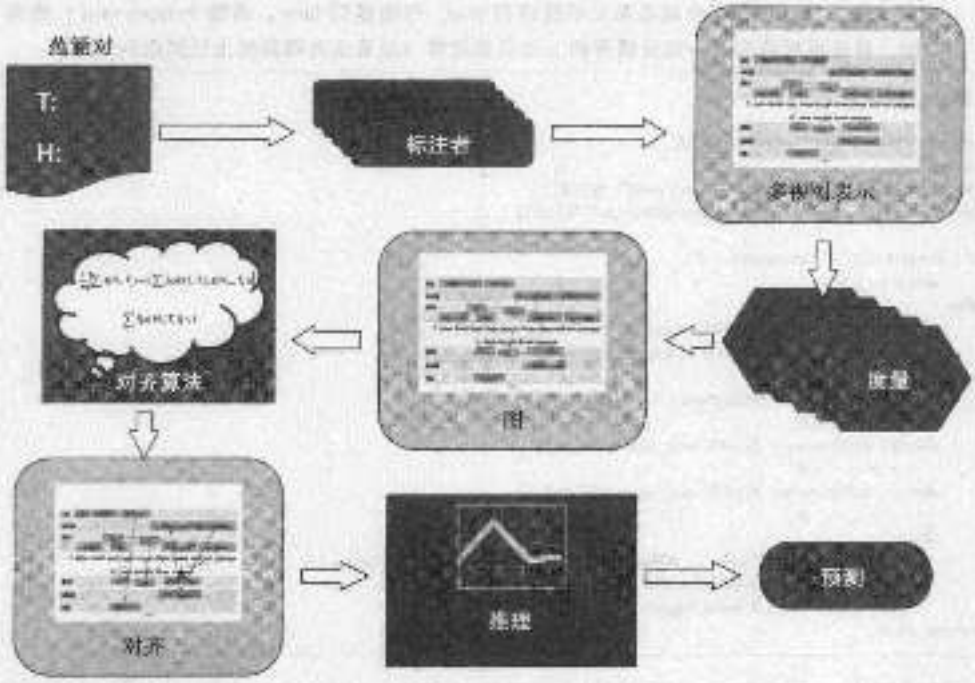


图 6-5　通用 RTE 框架的框图

1. 预处理

假定在 RTE 过程的第一步，我们要用一套现成的标注器[⊖]来标注蕴涵对的文本。尽管资源很多，但常用的资源包括句子和词的分割（确定句子边界、词以及标点符号词元）、POS 标记、依存分析或句法分析、命名实体识别、共指消解以及语义角色标记。这些不同的资源用于富化文本[⊖]。

我们在 6.3.3 节中描述了一种适于融合多种标注的数据结构。并且，在合适的时候，我们会说明它是如何对应到某些特定 RTE 系统使用的表示类型上的。

取决于使用的现有组件，在应用这些资源之前，可能需要清理输入，而据我们所知，没有现成的解决方案。例如，一些旧包可能无法处理多字节字符，它们必须被替换掉或忽略掉。清理步骤也可以对拼写进行规范化，这对诸如句法分析器或 POS 标记器等有很大的影响。

224

⊖　包或组件可以从一个或多个开源或学术软件中立即得到。

⊖　这里 RTE 的术语有些过载。我们用文本区别标示一般句子，段落及其部分，使用文本表示蕴涵对中的与假设对的那个输入的部分。

2. 富化

我们使用术语富化（enrichment），以与预处理相区分。富化指的是对已有视图进行操作，从而扩充视图或者产生新的视图。这与分析资源有本质上的不同，分析资源直接处理文本，生成标注，直接分析为成分、关系和视图。富化资源有两个功能：通过将文本或标注模式映射到结构的封闭集来进行抽象，或者通过识别输入文本或标记中的隐式内容并将其显式化为新结构来扩充已有标注。

抽象的例子是把动词的修饰语，如句子 *Attackers failed to enter the building* 中的 *failed to*，以及图 6-3 例子中的 *said that*，表示为动词的属性，或在相应的谓词-论元结构中表示为关系节点。在后者的情形中，我们可以编写代码来识别那种结构，将诸如 *buy* 这样的内嵌谓词用一个表示不确定的属性来标记。

规则应用（参见 6.4.3 节）是扩展的一个例子，将处理文本的隐式内容显式化或者生成文本的显式复述。RTE 系统可能用它们来生成额外的表示处理文本的复述的句法分析树，或者生成语义角色标注信息中的谓词-论元结构。

225

3. 图生成

在识别文本和假设中的多种句法和语义结构以后，必须比较假设中和文本中的这些结构。在最简单的系统中，只比较词。在更为成功的系统中，比较的范围为更广泛的标注类型。通常，文本与假设用图来表示，其中节点对应标注单元（如词、命名实体、分析树、标记了动词论元的语义角色），边对应标记类型间的联系（例如通过共指边把一个实体的不同提及连起来，或用依存树中的有类型依存边来联系词）。然后，根据相似度度量（可能是简单的相等判断）来将假设和文本中的成分连接起来，进而组成一个有别于文本和假设结构的二分图。

我们假定在图生成阶段中必须比较的成分类型都对应一个比较器（或者度量，如 6.3.3 节定义的那样）。多种成分可以共享同一种比较器，复杂成分的比较器（有结构的成分，如数字数量结构或谓语-论元结构）也可以调用其他更基本的比较器。

4. 对齐

对齐这个步骤后面的直觉是 de Marneffe 等人 [4] 提出来的。他们认为，文本中只有很少一部分成分是与假设相关的。对齐这个步骤就是要确定这些相关的部分，进而简化下一个步骤。

很多 RTE 系统有显式的对齐步骤；其他系统把对齐及推理集成为一步。一般来说，对齐将假设中的每个成分映射到文本中的一个成分上。该启发式方法基于以下观察：假设一般比文本要短，在蕴涵例子的正例中，人类读者经常使用文本的一部分来生成对假设的“分段”（piecewise）解释。

大多数 RTE 系统先把所有成分集成为单个图结构——也就是我们所说的一个视图——然后将图上的每个成分进行对齐。其他方法使用词来进行对齐，并且在推理步骤中，分析对应于对齐词的其他视图的结构。在我们自己的工作 [18]（在 6.4.6 节描述）中，我们对不同的视图组进行多重对齐，在推理步骤中比较它们，来分辨蕴涵与非蕴涵。

5. 推理

所有 RTE 系统都必须有一个决策组件来标记出所有的蕴涵关系对。它可以是一个相对简单的有阈值的重叠的度量，也可以非常复杂，比如从对齐图中抽取特征，然后应用机器学习分类器来确定最终的标签。有人使用定理证明系统来处理从蕴涵对中导出的逻辑表示和来自预处理步骤的分析。我们将于 6.4 节讨论一些不同的方法。

226

6.3.5 实现

本节我们讲解 RTE 系统中的不同部分，关注于成功 RTE 系统中共有的功能。案例分析将在第 6.4 节进行讲解，这些系统都来自最近的 RTE 挑战赛，并对应到我们的描述中。对于这个通用框架，我们使用一个使用基于 WordNet 的词成分相似度度量以及简单的基于命名实体的过滤规则的简单词法蕴涵算法（Lexical Entailment Algorithm，LEA）来作为我们的运行实例。

我们使用图 6-7 中的蕴涵对作为输入举例（略有人造痕迹），这个例子可以展示在 LEA 系统的上下文中，RTE 框架的每一步。

文本：John 说 Joan Smith 用 5 美元买了 3 个苹果。
John said Joan Smith bought three apples for five dollars.
假设：Joan Smith 为 3 个苹果支出了 $5。
Joan Smith forked out $5 for three apples.

图 6-7 一个文本蕴涵对实现的例子

1. 预处理

我们需要编写一个模块来控制不同分析资源产生的数据流，并将每个资源的输出翻译成成分、关系、视图数据结构。词级标注如词性以及原形可以整合到词成分中。将诸如命名实体等浅层标注分析为它们自己视图中的成分是很直接的。结构化标注，如共指或语义角色谓词和论元，需要为表示形式做决定，如是否为谓词和论元建立单独的视图，或者是否创建一个对应完整语义角色标注结构的额外成分。

预处理的一般顺序如下：

1）句子分割。

2）词边界检测。

3）POS 标记。

4）依存或句法分析。

5）命名实体识别。

6）共指消解（确定代词和其他实体提及的指向）。

7）语义角色标注（动词与名词性动词）。

这个顺序反映了一般的依赖性：比如，很多 NLP 应用需要 POS 标记作为信息源；大 [227]
多数语义角色标注需要依存或句法分析信息。有些工具允许甚至要求用户提供这种输入，而另外一些工具在内部完成所有的工作。自己提供这些数据可以避免重复应用具有相似功能的工具，进而提升效率。为方便起见，诸如词性和原形等词级标注可以加到词成分中。

注意，如果你使用不同源的工具，那么它们可能对输入有不同的要求。例如，许多应用使用未分割的文本作为输入，在内部进行分割。这里的问题是，没有明确“正确”分割的指南，因此不同源的输出在句子和词边界上可能会不一致。例如，用连字符连接的单词应该分开吗（如，American-led 还是 American-led）？表示货币的符号应该和数字在一起吗（如，$12M 还是 $12M）？在这种情况下你必须亲自解决这些差异。当然，你可以使用一个完整的工具集合来提供所有你需要的不同标注，或者使用接受已分割输入的工具。但每个任务中最好表现的工具都来自同一出处的这种情况是很少见的。如果一个特定的工具按照特定的分割标准进行开发，那么当给定另一个分割标准的输入时，它的表现或许不好。

2. 运行实例：词法蕴涵算法

对于我们的 LEA RTE 系统，我们需要两个视图：Word 视图和 NE 视图。Word 视图将包含蕴涵对成员（即文本或假设）所对应的每个词元的词成分，它将包括原始词和词的原形（如果有的话）。NE 视图将包含蕴涵对成员中每个命名实体的成分，包括实体的原始表示（原始文本的词元序列）以及实体类型。我们一开始并不使用所有信息，但是它使我们扩展原始算法成为可能。得到的多视图数据结构与图 6-5 中所示相同，除了没有 SRL 与 NUM 视图。

有的 NLP 应用提供编程接口，有的则没有。但是几乎所有应用都生成有标记文本的输出。对于那些不熟悉解析 NLP 工具输出任务的读者，我们在算法 6-2 描述了一个解析命名实体识别（Name Entity Recognition，NER）输出的算法。算法 6-2 也展示了一个 NER 输出的样例。我们假定 NER 使用与生成 Word 视图一样的方法来切分输入文本，或者 NER 使用分词的文本作为输入。我们同样假定在 NER 的输出中没有重叠的命名实体。那么在我们使用的 NER 标记器上该假设为真，尽管将算法进行扩展，使其能处理产生重叠的命名实体工具的输出，这并不是很难的事情。

3. 富化

为了扩充我们的 LEA 系统，我们富化处理的文本，添加与习语用法一致的简单表达。这个简单的资源用的是一个从习语短语到等价表达的人工书写的映射，比如 *kick the bucket* 映射到 *die*。假设只考虑能够映射到相同或更少数目的词的表达，我们也可以简单添加另一个词成分，该成分对应原始习语表达中相同的索引（一个替换词成分能够覆盖原始句子中的多个索引）。算法 6-3 描述了一个简单习语映射算法。 228

算法 6-2 这个算法解析了 NER 式的标注。函数 getNextWord(nerOutput) 分割出 nerOutput 的第一个单词并返回之。函数 peekNextChar(aWord) 返回 aWord 的第一个字母。函数 concatenate(startString, nextWord) 把 nextWord 附加到 startString 后，以一个空格隔开

```
// nerOutput 样例: "[PER Joan Smith ] bought apples."

// 假定：没有重叠的实体，输入中的中括号已经替换

CreateViewFromNerOutput( String nerOutput )

    neView ← ∅
    neType ← null
    neValue ← null
    indexSet ← ∅
    isInNe ← false

    while ( nextWord ← getNextWord( nerOutput ) )

        firstChar ← peekNextChar( nextWord )
        if ( firstChar == '[' )
            isInNe ← true
        getFirstChar( nextWord )
        neType ← nextWord
    else if ( firstChar == ']' )
        neConstituent ← { neType, neValue, indexSet}
        neView ← neView ∪ neConstituent
        indexSet ← ∅
        neType ← null
        neValue ← null
        isInNe ← false
```

```
    else if ( isInNe )
        wordIndex ← wordIndex + 1
        indexSet ← indexSet ∪ index
        neValue ← concatenate( neValue, nextWord )
    else
        continue

return neView
```

算法 6-3 生成习语视图的简单算法

```
// 假定：annotationGraph已具备词视图
// idiomList是从习语单词单个词的映射
// 例如forked out →pay ⊖

AddIdiomView( annotationGraph )
  maxWordsInIdiom ← 3
  indices ← getOrderedWordIndices( annotationGraph )
foreach index ( indices )
    indexSet ← 0
    offset ← 0
    sequence ← ""
    replacement ← null

    do
        offsetIndex ← index + offset
        word ← findWordWithIndex( annotationGraph, offsetIndex )
        sequence ← concatenate( sequence, word )
        replacement ← findIdiomMatch( sequence )
        indexSet ← indexSet ∪ offsetIndex
        offset ← offset + 1
    while ( ( replacement != null ) AND ( offset < maxWordsInIdiom ) );

    if ( replacement != null )
        idiomConstituent ←generateIdiomConstituent( replacement, indexSet )
        idiomView ← idiomView ∪ idiomConstituent

if ( idiomView != 0 )
    addView( annotationGraph, idiomView )

return
```

富化后的多视图数据结构如图 6-8 中所示，原始假设是 "Mr. Smith forked out $5 for three oranges"。在多视图表示中，每一个词元，包括标点符号，都有一个词成分。

习语映射模块（IdiomMapper）加入了一个新的词成分 *pay*，注意这个成分覆盖了原文中的 *forked out* 所对应的两个索引，这点在确定最优对齐时很重要（参见 6.3.6 节）。

230

4. 图生成

在图生成阶段，比较资源（度量）应用到从文本和假设中抽取的相关成分对。这可以用一种很直接的方法实现：遍历假设和文本的视图，然后遍历每个视图中的成分，最后应用合适的度量。

但是度量的代码本身可能很复杂，如对依存分析（子）树等高度结构化成分的度量。我们在算法 6-4 中提供了一个简单的图生成算法。

⊖ 原文误写为 buy。——译者注

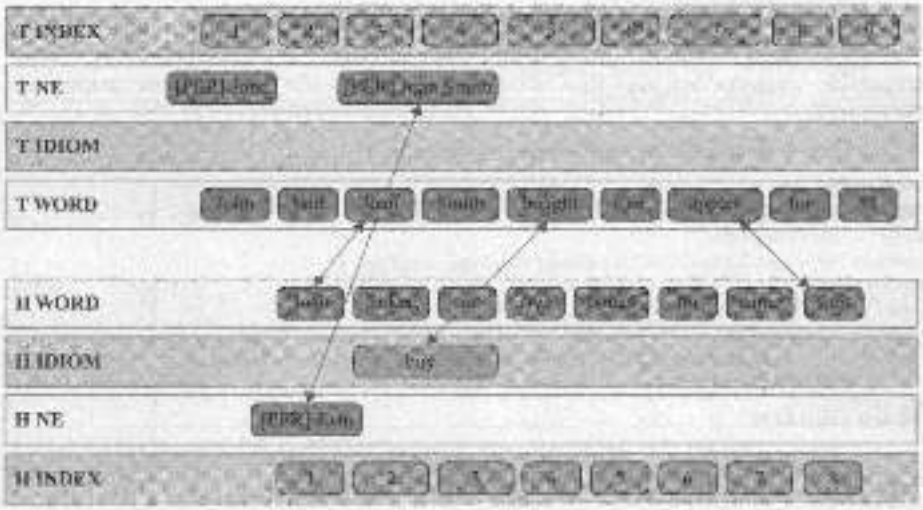


图 6-8 对于样例蕴涵对，LEA 得到的最优对齐。绿色的箭头用来连接对齐的部分

运行实例

在我们的例子中，文本中有命名实体 *John* 和 *Joan Smith*，假设中有 *Joan*、*John* 和 *Joan* 有很小的编辑距离（为 1），但是人类读者会明白，除非是拼字错误，否则这两个名字指的是不同的人。我们假定我们的命名实体度量足够聪明，可以明白这种情况，则会返回一0.7 的相似度分值。

我们另外的文本-假设命名实体对，即字符串 *Joan Smith* 与 *Joan*，应该返回一个高的分值，尽管它们的编辑距离很大（为 6）。假定我们的 NER 度量能够返回 0.9 的分值，因为尽管这两个字符串不相同，但是它们很可能指的是同一个体。

假定我们的词相似度度量使用 WordNet，并且应用如下启发式方法：如果词与同义词或上一级的上位词连接，那么分值为 0.9。如果词通过两级上位进行连接，那么分值为 0.6，三级上位的话分值为 0.3。如果词与反义词相连，那么分值为一0.5。这些行为在算法 6-1 所述算法中都有描述。 231

算法 6-4 匹生成阶段的算法（比较蕴涵对成员图）。这里假定系统存储一个从配对成分类型到兼容比较器的映射，并且比较器与度量类似，会返回一个分值

```
CompareHypothesisToText( hypGraph, textGraph )
  edgeList ← ∅
  foreach view hypV in hypGraph
      viewEdgeList ← ∅
      foreach view textV in textGraph
          if ( isCompatible( hypV, textV ) )
              viewPairEdgeList ← CompareViews( hypV, textV )
              viewEdgeList ← viewEdgeList ∪ viewPairEdgeList
      edgeList ← edgeList ∪ viewEdgeList
  return edgeList


CompareViews( hypView, textView )
  edgeList ← ∅
  foreach constituent hypC in hypView
      hypEdgeList ← ∅
      hypId ← getIdentifier( hypC )
      foreach constituent textC in textView
```

```
            textId ← getIdentifier( textC )
            score ← CompareConstituents( hypC, textC )
            matchEdge ← ( ViewType, hypId, textId, score )
            hypEdgeList ← hypEdgeList ∪ matchEdge
        edgeList ← edgeList ∪ hypEdgeList
    return edgeList


CompareConstituents( hypC, textC )
    hypType ← getType( hypC )
    textType ← getType( textC )
    comparatorSet ← getCompatibleComparator( hypType, textType )
    matchScore ← 0
    foreach ( comparator ∈ comparatorSet )
        score ← comparator → compare( hypC, textC )
        if ( score > matchScore )
            matchScore ← score
    return matchScore
```

232

6.3.6　对齐

大多数对齐算法背后的思想是，有的对齐比其他对齐要好，而直接为每个假设成分选择最相近的文本成分太过简化，因为这没有考虑句子结构。

给定蕴涵图生成算法以及比较器（度量）的表达，我们可以将选择最优对齐问题视为一个优化问题。我们将视图组一起对齐：例如，我们可以在一个对齐中结合 NE 与 NUM 视图。我们可以同时将所有视图一起对齐，或者单独对齐，取决于我们想要运行的推理类型。

我们通过限制对齐来允许假设中的每个索引映射到文本中的至多一个目标，因此覆盖多于一个词元的成分不会有重叠。这里的目标是识别解释假设词元的文本片段，进而简化推理问题。

总的来说，我们的直观想法是一些视图需要竞争：当同一词元有多种可选表示时，如习语的替代，我们希望互斥地进行选择，在这种情况下视图应该在对齐前组成组；我们希望其他视图能够单独处理，因为它们可能会提供有用的信息，如果组成组以后，这些信息可能会丢失。例如，假设命名实体度量只返回区间 [0，1] 内的值，并且没有实体成分匹配。如果我们将 NE 与 Word 视图合并，则会得到错误的部分实体匹配，比如共享称呼、姓或者碰巧出现在其他蕴涵对成员中的名与姓中的常规名词。当我们使用有不兼容输出（即它们的分值不能用同样的方法进行解释）的度量合并视图时，也会遇到相似的问题。另外，合并命名实体与词可能会产生问题，因为对于正向匹配，词相似度度量一直返回较低的分值。

在最优解中，不同粒度的成分可能均有对齐边，条件是它们没有重叠。

由于度量可能会返回负值，所以目标函数必须考虑这些。负值代表矛盾；如果没有更好的正向匹配，该信息会与随后的蕴涵决策高度相关。所以在目标函数中，要使用边权重的**绝对值**（magnitude）。边仍然保留表示负向的标签，在推理阶段使用。

对于浅层成分的对齐，我们需要在深层结构上进行猜测，因为我们在目标函数中添加局部性，对假设中相邻成分连到文本中分离成分的对齐进行惩罚。我们忽略相交的边，因为我们不认为这是判断蕴涵的可靠信息。

所以目标函数为：

$$\frac{\sum_i e(H_i, T_j) + \alpha \cdot \sum_i \Delta(e(H_i, T_j), e(H_{i+1}, T_k))}{m} \tag{6.1}$$

约束为：

$$\sum_j I[e(H_i, T_j)] \leqslant 1 \tag{6.2}$$

233

其中 m 是假设中词元的个数，$e(H_i, T_j)$ 是比较假设词元 i 与文本词元 j 的度量分值的绝对值，α 是距离惩罚的权重参数，$\Delta(e(H_i, T_j), e(H_{i+1}, T_k))$ 计算对齐到假设词元 i 的文本成分及对齐到假设词元 $i+1$ 的文本成分间的距离。对于覆盖多个词元的成分，该值是所有 T_j 中覆盖词元与所有 T_k 中覆盖词元的距离的最小值。这个距离函数可以通过多种方式来计算：例如，在词元或者依存分析树的路径中，$I[e(H_i, T_j)]$ 是指示假设词元 i 映射到文本词元 j 的指示函数。

对于融合不同粒度的对齐，前述表达使用覆盖当前词元对的映射成分的边的分值的绝对值作为一个词元级边权重。例如，对假设中覆盖的每个词元，两个命名实体间分值为 1.0 的边会计数为 1.0，覆盖两个索引的命名实体会生成分值为 2.0 的边。这避免了惩罚高于一个成分匹配的情况。

在我们的 RTE 系统 [18] 中，我们没有对齐训练数据，因此我们人工选择对齐参数 α（一个接近于 0 的正数，足够打破平衡），并且使用穷举搜索来找到最优对齐。搜索时间有上限，超过之后使用一个贪心的从左往右对齐来代替最优解。我们使用词元的个数作为距离度量 Δ。

算法 6-5 显示了我们使用的搜索算法。EdgeSetList 按照如下添加：对于假设文本区间内的每个索引，所有以该索引为起始的成分的边都收集到一个集合中，然后添加到 EdgeSetList 中。所有可能的对齐都要考虑并计算分值，返回最高的对齐分值。

函数 getNextAlignment 用来遍历所有可能的边的集合，这些边满足“每个假设词元一条边”的约束。为了实现这一点，需要使用 CounterSet：这个对象存储覆盖假设的每个索引的成分的总边数，以及指示在 EdgeSetList 中对应的索引的那条边用于之前的对齐的索引。为了生成下一个对齐，它增加不在对于相应假设索引的边集合中的最后一条边上的首个 EdgeSet 索引。如果一个计数器到达了最大的索引，它将重设为第一个索引，然后处理下一个计数器。如果所有的计数器都处于最大索引，那么所有的对齐已被考虑。

为了生成对应当前 CounterSet 值的对齐，要遍历 EdgeSetList。从以最小索引起始的成分的边集合开始，选择对应到相应索引的计数器的边。边假设成分的最后索引一个一旦找到，会跳过中间索引。接下来处理下一个没有被假设成分覆盖的索引，直到所有的假设索引都遍历过为止。

（按照目前所写，当 CounterSet 增加时，算法可能会生成重复对齐。但是增加的计数器是在成分覆盖的区间内，这里成分对应到一个通过较小假设索引计数器所选择的边上。为了节省空间和保持简洁，这里省略掉重复检测。然而，这里的算法是正确的，只是不够高效。）

234

算法 6-5　对于视图集合，寻找最优的对齐算法。函数 getIndices() 为图返回一个有序的词索引列表

```
findBestAlignment( edgeSet, hypGraph, textGraph )
  bestScore ← 0.0
  edgeSetList ← ()
  foreach index ( getIndices( hypGraph ) )
      currentEdgeSet ← findEdgesWithStartIndex( hypGraph, index )
      edgeSetList ← edgeSetList ∪ currentEdgeSet
  counterSet ← getCounterSet( edgeSetList )
  bestAlignment ← ()
  do
      currentAlignment ← getNextAlignment( edgeSetList, hypGraph, textGraph,
      counterSet )
```

```
        score ← scoreAlignment( currentAlignment )
        if ( score > bestScore )
            bestAlignment ← currentAlignment
            bestScore ← score
    while ( currentAlignment != 0 );
    return bestAlignment

getNextAlignment( edgeSetList, hypGraph, textGraph, edgeSetCounters )
    currentAlignment ← 0
    if ( incrementCounters( edgeSetCounters ) )
        position ← 0
        maxPosition ← sizeOf( edgeSetCounters )
        nextUncoveredIndex ← 0
        while ( position < maxPosition )
            position ← position + 1
            if ( nextUncoveredIndex <= position )
                currentEdgeSet ← edgeSetList[ position ]
                currentPositionCounter ← edgeSetCounters[ position ]
                currentEdge ← currentEdgeSet[ currentPositionCounter ]
                currentAlignment ← currentAlignment ∪ currentEdge
                hypConstituentId ← getHypConstituentId( currentEdge )
                hypConstituent ← findConstituent( hypGraph, hypConstituentId )
                lastIndex ← getLastIndex( hypConstituent )
                nextUncoveredIndex ← lastIndex + 1
    return currentAlignment

incrementCounters( edgeSetCounters, edgeSetList )
    index ← 0
    while ( index < sizeOf( edgeSetList ) )
235     counter ← edgeSetCounters[ index ]
        edgeSet ← edgeSetList[ index ]
        maxCount ← sizeOf( edgeSet )
        if ( counter < maxCount )
            counter ← counter + 1
            return true
        counter ← 0
        index ← index + 1
    return false
```

运行实例

在我们的 LEA 系统的对齐步骤中，我们结合单词和习语视图，并单独对 NE 视图进行对齐。其基本原理是，我们可以对习语成分和词成分使用相同的词度量，并且我们认为习语替换相当于生成一个新句子，其中替换项与原始项竞争。部分匹配的习语没有意义。图 6-8 描述了 LEA 系统生成的对齐。

式（6.1）是 LEA 实现的距离函数，简单起见，它总是返回 0。尽管当文本非常长时，但为倾向于聚类边的距离添加惩罚是可能的，而且对于假设中的某些词，在文本中可能会有多个匹配的词。

这个简单的 LEA 系统使用一个贪心的对齐方法，选择每个单独假设词匹配的最大值。在成语和词视图对齐中，习语替换计数两次，因为它覆盖了两个词索引。虚词如冠词（a、the 等）和介词（on、of 等）通常携带比名词，动词和形容词更少的语义内容，因此 LEA 使用包含这样的项的停用词表并忽略边的分值。

最优对齐的总对齐分数（图 6-8 中有显示）为 0.43。

NE 视图也进行对齐。假设视图中只有一个 NE 成分，并通过最高分值边进行对齐。

6.3.7 推理

RTE 系统的推理组件最后决定每个蕴涵对的标签（以及分值）。尽管在这里我们将它与对齐步骤区分，在某些方法中，二者是紧密联系的。

在一些系统中，推理仅仅是简单地将对齐分值与阈值相比。在这种二元 RTE 任务中，如果分值高于阈值，那么蕴涵对就标记为蕴涵，否则标记为非蕴涵。在三元任务中，有些系统进行两个连续的分类：一个用来区分未知的例子和其他，第二次分类将其他分为蕴涵与矛盾（参见 Wang、Zhang 和 Neumann [19]）。其他系统对一个对齐分值使用两个阈值：低的阈值用来区分未知与矛盾（参见 Iftene 和 Moruz [20]）。 [236]

其他系统在对齐步骤后进行特征提取（比如 Chambers 等人 [12]）。例如，这些特征能够刻画连接假设中每对词的依存分析连接与对齐到文本词的相应连接之间的对应关系。随后这些特征可以作为机器学习分类器的输入，分类器可以使用它们来预测蕴涵对的标签。

一些系统会根据全局特征来改变对齐分值。这些特征可能是过滤规则：例如，如果假设中有命名实体，但在文本中没有找到匹配，那么这个例子很可能是非蕴涵。另外一些特征的例子是否定特征：一般地，否定和其他项或结构影响极性，比如 "failed to"，它们在预处理或富化阶段被识别，并且在图结构中编码。然后它们用于影响最终决策，当文本中有否定词而假设中没有时，或许会将蕴涵标记为矛盾。相反的情况也成立，这时有其他的因素表明文本蕴涵假设。为了允许度量返回负值，通过边标签跟踪并且使用边分值的绝对值决定对齐，这种特征已符合我们提出的框架：在富化步骤做抽象，解释相关相似性度量的富化表示（允许它返回负分值），然后确定负边是否在最终对齐中存在，也是可能的。

运行实例

命名实体的对齐用作一个过滤器：如果在假设中有一个命名实体没有匹配文本中的任何对象，则 LEA 自动判断为非蕴涵。我们也可以通过限定单独边的分值以及将预测标签设置为非蕴涵来达到这种效果，如果任何单独的假设 NE 成分分值都比阈值低。

如果假设中命名实体全部匹配，则会咨询词与习语对齐。

由于假设包含一个命名实体并且和文本中的实体有正值的对齐，所以 LEA 不会将标签设置为非蕴涵，并且会咨询词与习语对齐。

对于词和习语对齐，LEA 使用一个简单的阈值，它只在二元任务中使用。假设词阈值为 0.67，LEA 会根据词和习语对齐将该例子的标签预测为非蕴涵。

注意 LEA 在这个例子上出现了错误，要做得更好，它需要有识别 "$5" 和 "five bucks" 是相同的能力，这种能力由数量分析以及对应的相似度度量来提供。这种资源同样会识别 some fruit 与 four apples 之间的映射，尤其当充分利用词相似度度量时。

如果在文本和假设中有反义项，比如 love 和 hate，则我们的词度量会返回一个负分值。如果在文本中没有对 hate 有更好的匹配（非反义词），对齐工具则会选择反义匹配边，因为它忽略边分值的负号。在推理步骤中，负值会保留并且会自动惩罚分值。我们可以通过改变分值函数，使用规则（如 "如果文本和假设中两个对齐的动词是反义词，则预测为矛盾"）或者以乘法来累积分值（一个负值边会导致全局为负值）来加强推理算法。这些启发式方法有时很有效，但是一般会带来新的错误源。尽管如此，考虑到这种效果，一些成功的 RTE 系统用它来提高性能。 [237]

6.3.8　训练

在那些最成功的系统中，对齐或推理组件必须通过使用开发数据集来进行调节，以适应蕴涵语料库。在使用机器学习组件的系统中，该过程就称作训练（training）：机器学习算法处理开发语料库中的蕴涵例子，计算相关的统计量，根据接收的输入特性来生成问题的模型。输入特性通常称为特征（feature）：表达式和函数接收输入中的特定部分，并且为每个例子计算一个值。

在基于非机器学习的组件中，也可能会有一个使用开发集调节相似度函数的过程，一般是在参数空间中使用尝试或穷举方法进行搜索。

我们在6.4节描述一些系统的训练过程。

运行实例

对于LEA系统，我们需要计算在推理步骤用于决定蕴涵标签的阈值。我们通过计算每个例子在开发语料库中的最好对齐分数，依据对齐分数对例子进行排序，并且将每个分数当作一个可能的阈值来测试。最后选取那个能将最多例子正确分类的阈值。

你也许在公式6.1中已经观察到了，我们用假设中词元的数目来归一化对齐边分值之和。我们这样做后，在推理（以及训练）的步骤中，决策对于假设的长度是无偏的。例如，考虑两个不同的例子，一个的假设长度是4而另一个的长度是12。如果每个例子中有4个相似的部分，我们直觉上选择不同的蕴涵标签，因为前者比后者更可能被标记为蕴涵。

6.4　案例分析

在这个章节中我们综述了一些最新系统，将它们作为案例学习。在每个案例中，我们描述方法的主要特征、方法使用的预处理模块以及用来预测蕴涵决策的方法（在相关的地方）。许多开源的资源被多个系统使用；我们不对每个这种资源进行重复引用，只列出名字，并且将所有信息放到本章的最后（详见6.6节）。我们的目标是在这里描述RTE中有趣的研究并且将不同的方法关联到我们的框架中。对于具体的实现细节，请参考原始文献。

要注意的是，只要可能，我们包含在RTE 5数据集上评测的系统。然而，一些有趣
[238] 的系统只在早期的RTE数据集上评测，因此它们精确度的结果不具有直接可比性。

6.4.1　抽取语篇约束

Hickl和Bensley提出了一个识别文本蕴涵的框架，该框架基于提取篇式信念或语篇约束。假设如下：文本包含了许多简单的构造，即使不蕴涵某些特殊的文本-假设对，它们也是对的。图6-9描述了一个蕴涵对以及所有的语篇约束实例；图6-10是描述这个系统的框图。

预处理的步骤包括句法分析和语义依存分析、命名实体识别、共指消解和数词识别。这些系统的输出可以用一个图表示来统一。

在简化步骤中，文本和假设会被分解成简单的句子集，这些句子本身是真的，与蕴涵对的真值无关。一个关系抽取器被用来辨别已知关系，如“拥有”、“位置接近”、“属性于”等，以及识别补充表述，例如括号、as从句和同位语。

在对齐步骤中，应用一个基于词元的对齐工具，这个对齐工具使用多种相似度度量，
[239] 例如基于WordNet的词相似度、Levenshtein字符串编辑距离和命名实体相似（相等）度量。这些度量用于将假设约束中的词对齐到文本约束中。

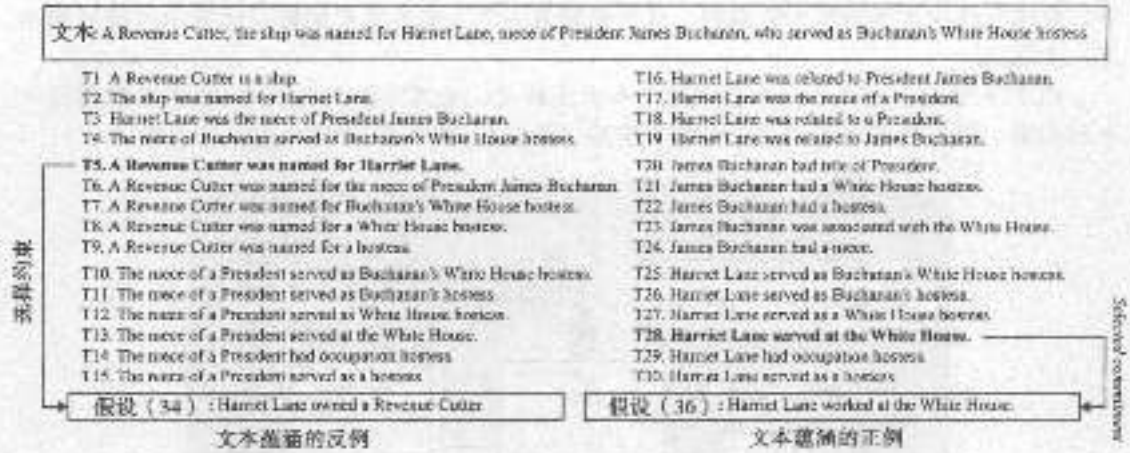


图 6-9 文本中蕴涵约定的例子 [21]

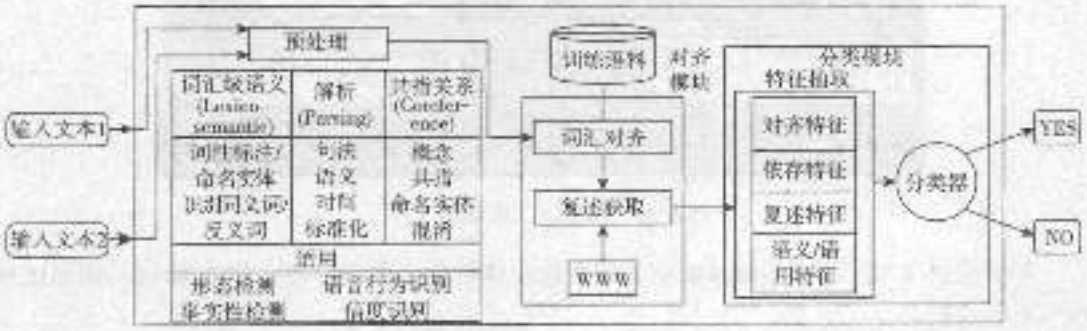


图 6-10 文本蕴涵框架 [22]

在推理步骤中，系统根据实体和论元的匹配情况来提取特征，并且使用决策树分类器来判断一个约束对是否代表一个有效的蕴涵实例。

这个分类器是用标准的方法训练的，使用从开发语料库中的每个例子提取的特征。

这个系统在 RTE 3 测试集上达到了 80.4%的准确率，并且修改的系统在 RTE 4 数据集上的得分为 74.6%（见 Hickl [23]）。

尽管系统表现得很好，但它依赖于一个专有的额外训练数据的大语料库，并且大部分所使用的预处理工具也是专有的。但是，本质概念和许多其他方法是类似的，即将表面文本分解成简单的单元并且匹配这些单元，而不是匹配原来的词和句子。

6.4.2 基于编辑距离的 RTE

尽我们所知，树编辑距离（一般基于依存分析结构）由 Punyakanok、Roth 和 Yih [24] 首次在文本推理中用于问答系统任务中选择答案。一些团队也在后来将树编辑距离应用于文本蕴涵的任务中去，（如 Kouylekov 和 Magnini [25] 在 RTE 1 中的工作）。

Mehdad 等人 [26] 提出了一个开源的关于文本蕴涵的框架，称为“编辑距离文本蕴涵套件”（Edit Distance Textual Entailment Suite，EDITS）[27]，它提供了一个基本的、可自定义的框架，可以系统地开发与评测基于编辑距离方法的 RTE。这个框架允许计算编辑距离，通过在字符串、词元、树级别上使用编辑操作来将文本转换成假设。除此之

外，它也允许包含蕴涵和矛盾规则，这类规则为一个从文本元素到假设元素的转换规则加上一个分值。

EDITS 框架也定义了一个通用的文本标注格式，来表示输入的文本-假设对和蕴涵与矛盾规则。训练数据用于学习一个距离模型，图 6-11 展示了 EDITS 的工作流程。

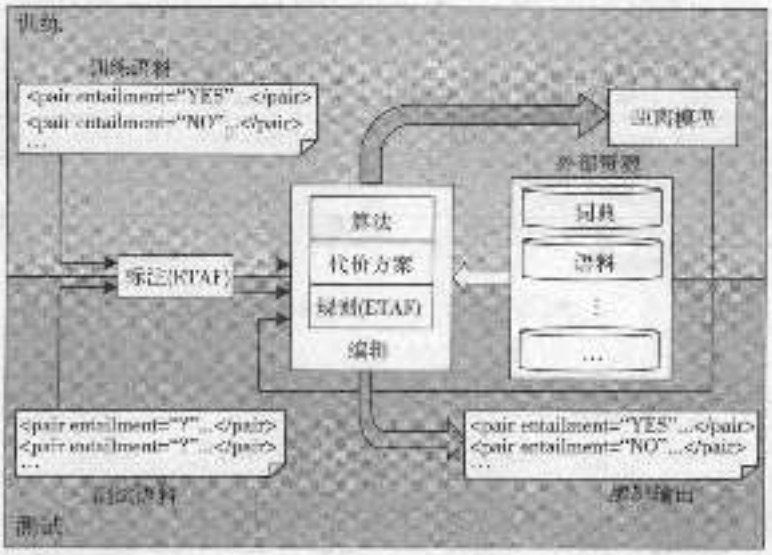


图 6-11　EDITS 工作流程 [27]

在提交到 TAC RTE 5 的系统中，预处理步骤使用了依存分析、POS 标记、词形还原和形态学分析。

图生成和对齐步骤结合在了一起。最小的编辑距离代价是通过一个操作集（插入、删除和替代）来决定的，每个操作都有一个相应的代价。这些代价是通过一个优化算法和一个阈值来学习的，阈值通过最大化开发集的性能来得到。词级的替换资源来自 VerbOcean [28]、WordNet [14] 以及维基百科的潜在语义分析。

通过学习到的阈值，提取步骤可以比较计算出来的编辑距离：如果蕴涵对的编辑距离大于阈值，系统就分配“非蕴涵”标签，反之分配“蕴涵”标签。

基于 EDITS 的 RTE 系统在 RTE5 的分数达到了 60.2%，但是可以通过探索新的替换资源来提高分数，也可以通过富化输入结构来提高，如使用命名实体的信息（在推理步骤 246
中要用专门的相似性度量）。

6.4.3　基于转换的方法

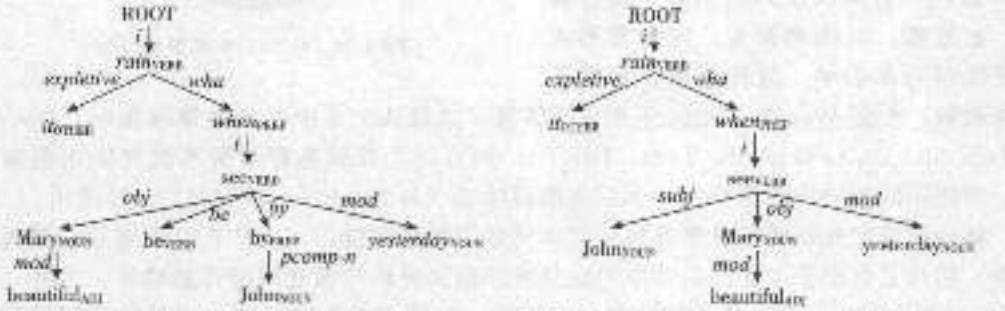
Braz 等人 [30] 描述了一个扩展的蕴涵对中文本和假设的基于图的表示，它通过使用为短语提词、短语、句法以及词间-论元级信息的同义表达设计的手写规则来进行扩展。他们使用模型论方法来论证他们的系统，当规则用于蕴涵对文本时，扩展表示有对该文本的一个可能（正确）的解释（理想情况下，当文本蕴涵假设时，使得文本与假设更相近）。如果任何文本的表达包含了该假设，那么该文本蕴涵了这个假设。

包容（subsumption）以整数线性规划问题来表达，根据文本决定最小代价的假设包容。规则也有代价，这些代价进一步根据表达规则的表示级别来加权（直观上看是这样的：匹配关系——因此动词——比匹配如限定词等单独项要重要）。

系统的预处理步骤用浅层分析、句法分析、命名实体以及语义角色标注来标记蕴涵对。富化步骤试图匹配每条规则的左部到文本图上。如规则匹配，规则的右部则用来扩充文本图。迭代多次，以使得规则可串起来应用几次。

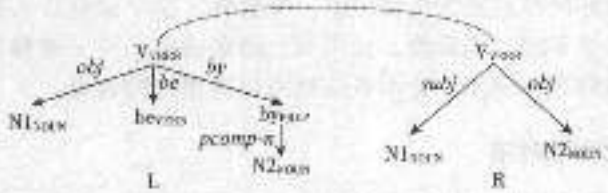
这里没有显式的对齐步骤，推理步骤视为整数线性规划问题，根据文本决定包容假设的最小代价。如果代价过高，蕴涵对则标记为非蕴涵，否则标记为蕴涵。这个系统已知在一个 RTE 1 开发集的子集上能够超过智能的词法基准系统，在 RTE 1 测试集上达到 56.1%的准确率（两个最好的系统准确率均为 58.6%）。

Bar-Haim 等人 [29] 描述了一个转换基于句法分析表示的蕴涵对文本的构架。使用规则来表示句法分析树中的片段。手工编码规则被用于抽象大量的句法变体。这些规则使用占位符来把两个句法树结构片段组成一对，占位符表示子树在转换中不发生变化。图 6-12 给出了一个例子。



原始句：It rained when beautiful Mary was seen by John yesterday 推导句：It rained when John saw beautiful Mary yesterday

a) 被动式到主动式转换的应用实例



b) 被动式到主动式的转换（置换规则）。虚线弧代表对齐

图 6-12 一个推导规则的应用实例，来自 Bar-Haim 等人 [29]

在 RTE 过程的富化步骤中，规则的头与文本的结构进行对比。如果它们匹配，则通过规则体来生成新的句法分析树。由规则占位符识别的原始文本结构的子树复制到新分析树的对应位置。

推理步骤从最匹配的文本—蕴涵表示对（由距离度量定义）中提取特征，分类器使用这些特征来预测蕴涵标签。

为了训练分类器，使用相同的步骤。首先抽取特征，然后按照标准有监督学习的方式来使用蕴涵对的特征表示以及蕴涵对的标签。系统的一个版本在 RTE 4 上获得了 60.5% 的准确率（Bar-Haim 等人 [31]）。

这些方法的一个缺点是需要很多的规则来获得很多可能的句法变体；制作此类人工编写的规则是一个高成本工作，这无疑是个问题。然而，直接融合世界知识的方法是很有吸引力的，因为要使 RTE 取得显著的提升，必须克服融合背景知识这个难题。

6.4.4　逻辑表示及推理

Clark 和 Harrison [33] 提出的 BLUE (Boeing Language Understanding Engine) 系统是一种基于形式逻辑的 RTE 方法。BLUE 先把文本转换成一种基于逻辑的表示，然后使用定理证明系统从这种表示中推出假设。

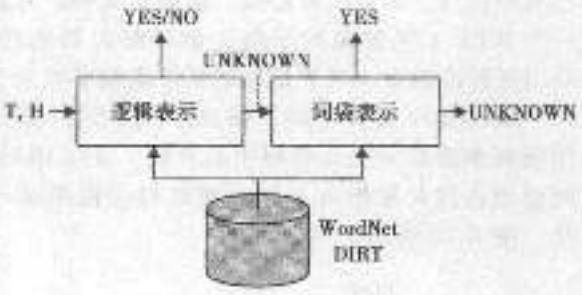


图 6-13　BLUE 系统架构 [32]

BLUE 系统是一个两级流水线，如图 6-13 所示。最开始，文本和假设使用自底向上的线图分析算法来分析成逻辑表示 [34]。这个逻辑形式是一个有逻辑类型元素的简化树状结构，它融合了一些预处理中的步骤，如依存分析、POS 标注、名词以及共指消解。情态属性，如复数、时态和否定，用逻辑形式的特殊谓词来表示。这种逻辑表示用于推断蕴涵，根据 WordNet 的包容及等价信息，以及从文本中发现的推理规则 (Discovery of Inference Rules from the Text，DIRT)。如果这个逻辑推理步骤不能判断出蕴涵或矛盾，则使用词袋对齐模型作为一个后备推理模块（与 WordNet 和 DIRT 一起使用）。

242
|
243

通过使用定理证明系统来寻找从文本到假设的推理链，BLUE 正试图寻找蕴涵决策的解释。但是它受限于知识源以及如句法分析和语义分析等预处理阶段的错误。另外，根据 Clark 和 Harrison [33] 提出的分析，文本中一些隐含信息的存在，以及缺乏能够填补文本和假设间语义差距的知识，限制了系统的表现（在 RTE 5 上分值为 61.5%）。

这个系统一个非常好的特点是为标注产生一个解释，这个解释让人们可以确定错误的根源，并且可以评价这个系统的可靠性：如果对于给定的蕴涵例子解释是可取的，我们就可以确信对于相似领域的未知例子，这个系统也可以有很好的表现。

6.4.5　独立于蕴涵学习对齐

De Marneffe 等人 [35] 独立于 RTE 来研究对齐，他们提出对齐可以认为是识别文本中相关的部分，以及这比确定文本中的哪部分蕴涵假设要简单的想法。他们把对齐形式化为一个最优化问题，考虑假设中单个词元以及由依存边相连的假设词元对的对齐。他们使用人工标注对齐数据来训练他们的对齐工具，用他们自己的方法来评测对齐工具。在 MacCartney、Grenager 和 DeMarneffe [36] 所描述的蕴涵系统中，自动对齐工具是对齐步骤的基础，它用作全局分类器特征源。这些学者提出一个有用的构想，通过目标函数来表达对齐。但是他们的方法有一个缺点，即需要标注的对齐数据去训练这个系统，这需要很多的时间和资源来构建。

MacCartney、Galley 和 Manning [37] 将对齐问题扩展到短语级别（在这里短语仅仅意味着连续的文本区间），并且通过对假设文本中短语的相等、替换、插入和删除操作来形成对齐分值。他们用 Brockett [38] 生成的词法对齐标记来训练模型。尽管他们表示该方法比基于两个词法级别对齐的基准系统有提升，但他们在同一系统的短语级和词元级对齐中并没有观察到显著差异（即短语的大小固定为一个词元）。

244

这种方法的局限是，它似乎无视已知的成分边界，而且并没有提供一个明晰的机制来应用专门的相似资源，只能对连续文本区间进行统一处理。此外，它需要有标记的对齐数

据，这只存在有限数据，而且都在同元级。然而，他们对于训练对齐工具以及运行时探索对齐的可能空间问题的解决方法是十分优雅与清晰的。

6.4.6 在RTE中利用多对齐

RTE系统开发者希望使用更深层的NLP分析器，他们面临的两个难点是：整合运行在不同粒度上（词、短语、语法和谓词-论元级别）的NLP分析器，以及用一种跨越不同表示层的一致方法来应用相似度量以及其他知识资源（如规则）。在基于对齐和全局相似度的RTE方法中，在试图整合多个知识资源时会出现问题，因为资源是为不同的任务而准备，即使当它们都返回实值分数时，也可能有不兼容的输出。例如，一个命名实体度量可能会返回一个0.6的分值，这表示相对较低的相似度，而一个基于WordNet的度量可能返回相同的值以表示相对较高的相似度；他们的分值是不兼容的，因为返回相同的分值并没有等价的意义。

Sammons等人[18]试图解决这两个问题，他们描述了一个多视图方法，该方法中不同的NLP分析源呈现于不同的数据视图中，尽管可比的表示层次可能被融合在同一视图中。专业知识资源被编码为对这些个体视图进行操作的度量。他们的系统对每个蕴涵对中的文本和假设使用多对齐，根据不兼容的度量将视图区分成不同的对齐。

特征在单一对齐和多个对齐上进行定义，根据如下的观察，（例如）如果词汇级的对齐或基于语义角色的谓词-论元结构的对齐表示蕴涵，但使用数量度量的对齐不表示蕴涵，这是一个很好的迹象，显示出该文本并不蕴涵假设。这些特征用来训练一个分类器。

多视图、多对齐的模型允许以一种模块化的方法来融合新的NLP分析方法和知识资源，基于机器学习的推理组件允许系统确定来自不同分析数据源的线索的可靠性。

该系统与其他基于对齐的系统相比有竞争力，在RTE 5二元任务中得到66.6%的分值。

6.4.7 自然逻辑

MacCartney和Manning[17]提出一个基于自然逻辑的表示与推理过程的框架来应对文本蕴涵的挑战。在这种方法中，用接近原来表面形式、不涉及完备的语义解释的句法形式来刻画有效推理模式。

基本思想是将蕴涵过程分解成一系列较小的蕴涵决策，将部分文本与部分假设进行比 245
较，并关联到封闭操作集中的一个操作，这个操作表明了两者之间的语义关系。例如，语义包含（semantic containment）能识别出何时一个概念推广另一个概念，而语义排除（semantic exclusion）指示当一个概念为真时，则排除另一个为真。

他们还对上下文结构进行分类，在承认文本蕴涵假设时，上下文结构会影响给定关系的有效性，这依据极性和单调性进行表示。极性必须兼容才能允许蕴涵，极性还需考虑蕴涵对中表达的动词的否定和情态修饰。单调性说明了一个文本概念是否比它在假设中相应的部分更一般或者更特殊，往往出现在结构中一些特别类型上，例如全称量化的陈述句。

为了确定蕴涵，该文本首先被表示为一个基本语义关系（前提），然后用一系列的编辑操作来将这个前提转换为假设。对于每个编辑操作，使用一个统计分类器来预测一个词汇蕴涵关系，这些关系根据中间节点的语义属性通过句法树向上传播。最后一步根据编辑序列写出蕴涵关系结果。

此方法适合简单的句子，例如那些在FraCaS语料库中的句子（Cooper等人[39]），但是要从蕴涵对的文本中抽取可靠的基础前提是非常困难的，因为这往往需要世界性的知

识来推断那些能紧密反映假设中结构的关系。为了将自然逻辑推理应用到 RTE 任务上，Pado 等人 [40] 用一个直接的线性函数来融合之前描述的对齐系统以及 NatLog 编辑距离，并在 RTE 4 上获得了 62.7%的分数。

6.4.8　句法树核

Mehdad、Zanzotto 和 Moschitti [41] 提出的 SemKer 系统，使用句法树核来定义文本树对和假设树对之间的相似度，这些树对是从每对蕴涵例子中取得，并通过基于维基百科的相似度度量来扩展模型。系统使用基于依存树的表示，通过词汇或语义匹配抽象节点。SemKer 通过句法语义树核（Syntactic Semantic Tree Kernel，SSTK）[42] 来计算词项间的相似度，这种方法用片段（子树）匹配来编码词汇相似度。

系统有一个初步的词汇对齐阶段，在这个阶段建立潜在的子树-匹配位置，称为锚点（anchor）。这些关注子树-匹配组件的应用，为每个蕴涵对确定最终的文本和假设之间的对齐。

为了训练推理模型，这些锚点被抽象成一般占位符，另一个基于树核的相似度函数被应用于比较蕴涵对之间的对齐模式。目标是学习更一般的结构对应，以适用于多个蕴涵对。使用对间（interpair）距离度量和蕴涵实例标签来训练一个支持向量模型，这种模型用于他们 RTE 系统的推理步骤中。

在 RTE 5 上这个系统表现良好，在二元标注任务中达到 66.2%的准确率（前 5 名）。为
246 了能获取自然语言文本中所允许的更大范围的句法信息，进而提升性能和泛化能力，该方法似乎需要更多的训练数据。如果使用 Hickl [23] 描述的专有语料库进行训练，系统的性能会如何变化令人非常感兴趣。

6.4.9　使用有限依存上下文的全局相似度

Ifrene 和 Moruz [20] 开发的系统在 RTE 5 上执行二元或三元蕴涵任务时表现均为最佳。他们系统的结构，和许多其他成功的系统一样，非常接近我们在 6.3 节所描述的系统。

在预处理步骤中，蕴涵对中的文本首先通过扩展缩写（如将 isn't 替换为 is not）以及替换某些标点符号来进行规范化。这提高了他们所使用的现成包的性能。蕴涵对的导出表示基于依存分析树，由命名实体信息来扩充。预处理步骤也应用了一些自定义的数据源，这些数据源标注特定的关系（如 "work-for"）、数量词和语言。

对齐步骤有局部和全局的分值函数。首先，每个假设成分被映射到最优的候选文本成分。此过程包括应用从 WordNet、维基百科、VerbOcean 和其他自定义资源中导出的规则来识别不相似的文本-假设词项对间可能的映射，这些映射具有相应的分值。这些局部适应性分值也考虑了被比较节点的父节点和连接它们的依存边的类型。

接着这些局部对齐分值被整合，并根据对齐的全局特征做了一些调整，如假设中的命名实体是否匹配文本中的实体，以及对齐的谓词是否在文本和假设之一中被否定，而在另一个中则没有。

这个推理步骤在最终分值中使用了两个阈值：一个更高的阈值用来区分蕴涵和非蕴涵，而一个较低的阈值用来区分未知和矛盾。对这些阈值运行调整，使得系统在开发集上执行三元任务时达到最佳性能；通过把未知和矛盾标签合并为非蕴涵标签，二元标注可以直接从二元标注中导出。

该系统在 RTE 5 三元任务和二元任务中分别达到了 68.5%和 73.5%的准确度。

6.4.10 RTE的潜在对齐推理

Chang等人[43]开发了一个联合学习方法，这种方法学习蕴涵决策伴随着学习一种中间表示，这种中间表示能对齐文本和假设。中间对齐级别无须假定为有监督学习。他们为RTE和需要通过中间表示进行学习的其他问题提出了一种通用的学习框架。

这个框架使用声明性整数线性规划（Integer Linear Programming，ILP）推理公式（参考Chang、Ratinov和Roth[44]），可以很容易地用二元变量来定义中间表示，知识可以作为模型的约束进行添加。这个模型假定所有正例至少有一个好的中间表示（对齐），而反例没有好的中间表示。

在训练过程中，如果模型产生一个很好（有效）的对齐，意味着基于由这个对齐触发的特征而产生的蕴涵决定是正确的，学习阶段使用这样的对齐作为正例来训练蕴涵分类器，并且还对这个对齐模型提供反馈。

用图来表示文本和假设，其中单词和短语是节点，词间的依存关系是边。此外，有向边将动词连接到它们语义角色标记论元的中心词上。在文本图和假设图中，这种节点和边的映射定义了对齐。使用词映射和边映射之间的关系来约束对齐变量；例如，仅当相应的词映射激活时，边映射才激活。

这种方法的一个关键是对齐步骤没有被要求作为一个单独、独立的任务；相反，定义一个对齐结构的空间，目标应用的标准答案训练标签与优化方法一起使用，来确定目标任务的最优中间表示，即能最大限度地提高目标任务的性能表示。这省略了中间结构所需的昂贵的标注工作。

Chang等人[43]在音译发现（transliteration discovery）、复述识别和文本蕴涵识别方面应用他们的框架。对于RTE任务，预处理步骤使用命名实体、依存分析、语义角色标注以及共指分析，将它们合为一个单一的、规范的图形结构。图生成步骤使用词和命名实体的相似性度量（请参考Do等人[45]），但也计算文本和假设边之间的对齐边，这里边的源端和尾端也进行对齐。

对齐和推理步骤整合为一步，并且最优的对齐和最优的蕴涵决策是基于训练过程学习到的特征权重。Chang等人[5]的系统在RTE 5语料库上执行二元任务获得了66.8%的准确率。

6.5 RTE的进一步研究

图6.2中的结果表明，在RTE能够解决问题之前还有很长的路要走。从这一章中给出的各种例子可以很明显地看出，可靠识别文本蕴涵需要处理许多更小的蕴涵现象，例如确定两个字符串指向相同的底层实体，或者应用背景知识来推断文本中没有明确注明的东西。在本节中，我们展示一些特别重要的能力以此对未来研究提供可能的关注点，这些能力目前还没有（足够的）开发。

6.5.1 改进分析器

所有成功的RTE方法都取决于其他NLP工具的输入。标注越复杂，相应工具性能越差。提高资源的性能，例如命名实体识别器和句法分析器，就能有助于提高依赖于它们的RTE组件的性能。这点在Bar-Haim等人[29]的RTE系统上表现特别明显，因为它使用基于分析结构的规则来富化输入信息。

学者们普遍认为，对文本推断的一个很重要的功能是共指消解。尽管共指消解系统在按某种目标建造的语料库上取得了不错的表现，但它们（与其他 NLP 应用一样）在其他领域的原文本上的表现就差了很多。性能下降部分原因是过拟合评测领域，以及评测本身所做的假设。特别地，系统在将共指短语（即非指代）提及指向正确实体方面表现非常差。

6.5.2 发明或解决新问题

有很多语言学现象似乎和 RTE 相关但却没有现存的 NLP 资源，甚至自然语言处理社区都没有把它们看作必要的任务。某些问题即使被认为是潜在有用的，但还可能缺乏相关的语料库。

一个较为相关例子是迹（trace）或省略（parasitic gap）恢复：识别出句子中作者隐指他物，需要读者通过上下文语境补足空缺的部分；例如，在句子“John Sold apples, Jane oranges”里，人们会推断出“Jane”和“oranges”之间的关系是“Sold”。通过句法分析器来填补空缺的尝试，正如 Dienes 和 Dubey [46] 所做的，只取得了有限的成功，部分是因为句法分析器并不是完全正确的，还因为原始标注（见 Marcus, Santorini 与 Marcinkiewicz [47]）不一致。

还有一个相关问题是零形回指解析。例如，在句子“一个雨天就够糟的了，一连三个简直无法忍受。”中，人会认为“三个”指的是“三个雨天”。虽然有些文献讨论了这个问题，但是到目前为止，还没有被社区广泛使用的应用程序存在。

NLP 工具通常只标记显式内容，文中所描述的问题需要额外的大量处理。如果这些问题能够得到解决——例如，通过识别出内容缺失的地方，或能更好，添加缺失内容——一些 NLP 分析器能为 RTE 和其他 NLP 任务产生更有用的输出。

还有一个值得长期关注的话题是篇章结构。更困难的 RTE 例子需要综合分散在多个句子中的信息。在 RTE 5 试验的搜索任务中，Mirkin 等人 [48] 观察到在一些新文章中，需要标题中的信息才能完全理解文章中的句子。一些事件中的关系，如因果及时间，可能通过不限于一个句子的结构来传达——而单句通常为许多 NLP 工具的处理边界。篇章结构能够指出长距离依存，但篇章结构还是 NLP 研究中的一个开放性话题。随着 Penn Discourse Treebank [49] 的公布，已经出现了一些适于开发某类长距离依存分析的资源。

6.5.3 开发知识库

249 还有许多已知的蕴涵现象没有在 RTE 语料库中被明显表现出来，但对能进行自然语言理解的系统而言是必要的。特别是，某些人类无意识的推理对自动系统而言是一个很大的挑战，因果和空间推理便是很好的例子（图 6-14 及图 6-15 展示了 RTE 5 语料库中的样例）。

因果推理与人们应用的表达领域相关的因果关系的世界知识有关；例如，炸弹会爆炸，而爆炸会造成死伤。

在图 6-14 的蕴涵对里，人必需推断出人对所站的结构上施力（重力），桥上的重量越大就暗示着人越多。因此假设中桥坍塌的原因表达为“重量太大”而非“人太多”是合理的。

在图 6-15 的蕴涵对里，文本表达了巴格达和华盛顿的政治领袖都很关心爆炸案。然后详细描写了三个爆炸案，读者必须推断出“巴格达南部”（*south of Baghdad*）意味着“在巴格达地区”（*in the Baghdad area*），并且因为地处伊拉克（这能通过地理背景知识推断出），Abu Gharib 也可以被当作“在巴格达地区”，至少在给出的上下文中如此。

文本：Local health department officials were quoted as saying that the bridge over the Santa Barbara river, in southern Peru's Ayacucho province, "broke in two" as students and teachers from four rural schools were crossing it while going home … Local police said the 120-meter bridge, made of wooden boards and slats held together by steel cables, collapsed because too many people were on it.

假设：The Peruvian bridge in Ayachuco province broke because of the weight on it.

图 6-14 需要理解因果关系的 RTE 5 例子（开发集，文本有截断）

文本：Three major bombings in less than a week will be causing some anxiety among political leaders in Baghdad and Washington. Last Thursday 15 people were killed by a car bomb at a crowded cattle market in Babel province, south of Baghdad. On Sunday more than 30 died when a suicide bomber riding a motorbike blew himself up at a police academy in the capital. Tuesday's bombing in Abu Ghraib also killed and wounded a large number of people — including journalists and local officials.

假设：Some journalists and local officials were killed in one of the three bombings in the Baghdad area.

图 6-15 需要理解空间关系的 RTE 5 例子（开发集，文本有截断）

250

其他类型的推理，如需要从图 6-16 中的蕴涵对中识别多种亲属关系，进而确定蕴涵对间的各种强联系。这些推理不够一般，但是在 NLP 任务中有代表性。在这里，用一种一致的、足够无歧义并且 RTE 系统可使用的方法来表示知识是一个挑战。CYC 数据集 [50] 是一个大知识库，以一种一致的逻辑形式精心编码。由于表示的限制，这个数据集没有被大规模使用。然而，Lin 以及 Pantel 的 DIRT 规则 [51] 被广泛认为是一种可用形式（带有实体位置的依存树路径），但是对于实际使用而言，噪声过多（见 Clark 与 Harrison [33] 以及 Bentivogli 等人 [5] 的研究来找到一些例子），也类似 TextRunner [52] 的 OpenIE 方法识别的“事实”也有很多噪声，在 RTE 中的用处仍有待证实。

文本：British newsreader Natasha Kaplinsky gave birth to a baby boy earlier this morning at around 08:30 BST. She had been on maternity leave since August 21. Kaplinsky had only been working with Five News just over a month when she announced she was pregnant. Her husband of three years, investment banker Justin Bower announced "We're absolutely thrilled."

假设：Natasha Kaplisky and Justin Bower got married three years ago.

图 6-16 需要理解亲属关系的 RTE 5 例子（开发集，文本有截断）

一种以合适表示呈现的适用领域的无噪声规则集将会是很有价值的财富。Szpektor 等人 [53] 提出了一个很有前景的表示。

6.5.4 更好的 RTE 评价

现有的对 RTE 的评价主要集中于绝对性能上，给定系统，在二元任务中报告预测两个标签之一（蕴涵与非蕴涵）的准确度，或在三元任务中报告预测三个标签之一（蕴涵、矛盾与未知）的性能准确度。从人类推理的角度，RTE 研究者所面临的一个问题是，预测标签需要涉及做其他的蕴涵决策，而单个标签并没有给我们提供系统如何处理这些更小决策的信息。在图 6-15 的例子中，一个人必须推断出文本中报道了三件爆炸案，短语“包括记者和当地官员”（*including journalists and local officials*）是“一大群人”（*a large number of people*）所指的实体，并且文本中提及的三处地点全部位于巴格达地区。如果不知道系统实际上是如何解决这些问题的，我们就无法侦知系统所使用的方法在处理需要相似推理的新蕴涵问题时是否可靠；例如，系统可能预测错了蕴涵标签，但是可能正

确解决了需要空间知识的推理问题，如果开发蕴涵子问题的可靠解决方案，RTE 社区可
251 通过认识并重用这些方案来避免重复劳动并且将注意力转向其他所需的能力。

针对这个问题有两个明显的解决方案：要求系统来为它们的答案提供解释并且使用比当前二元或三元标签更多的信息来标注 RTE 实例。

至少有一个 RTE 系统（Clark 与 Harrsion [33]）已经可以生成解释，这对识别知识资源里的缺陷是很有帮助的。尽管该系统严重依赖于其形式逻辑推理过程，而该过程在处理有噪声的输入时是很脆弱的。尽管有了该系统，解释里的步骤也不总是很清晰，并且无法证实人类推理的步骤能够完全契合这个形式体系。

一个解释的标准格式——以及相应蕴涵实例的标注——能进一步使得 RTE 系统开发者用一种系统的、合作的方法来生成解释成为可能，而不是大家各行其是。

第二种选择是更完整地标记 RTE 实例，而不使用特定的解释表示。作为一个部分措施，需要一个决定以及记录蕴涵判断蕴涵标签所需的蕴涵现象的标注标准，这个标准至少能够对给定 RTE 系统特性有大致的理解，这是通过检查正确标记的实例以及活跃的蕴涵现象之间的联系来完成的。

另外，这样的标记能让研究者快速抽出带有特定特性的蕴涵语料库，可以根据 RTE 的性能来评价特定现象的资源。Sammons、Vydiswaran 与 Roth [54] 提出了这些问题以及标注标准。

6.6　有用资源

本节将给出一些在 RTE 挑战赛评测中的 RTE 系统所使用的资源的信息。

6.6.1　文献

NIST TAC RTE 挑战赛在其官网发布数据集以及参加 RTE 系统的说明，而许多 RTE 研究者均参与这项挑战赛[①]。你可以在 ACL RTE 门户网站[②]找到更多有关 RTE 研究文献的链接。其他与 RTE 相关的文献出现在诸如 ACL、EMNLP、COLING 和 AAAI 等会议上，ACL 和 EMNLP 的论文也可以在 ACL authology[③] 上获得。

6.6.2　知识库

ACL RTE 门户也能为一些有用的知识库提供链接[④]，比如规则集合，我们在 6.4 中
252 的案例分析中涉及了其中一些内容。

ACL RTE 门户网站还能提供一些完整 RTE 系统的下载。

6.6.3　自然语言处理包

一些流行的自然语言处理框架包括 LingPipe[⑤]、UIMA[⑥]、NLTK[⑦] 和 GATE[⑧]，当

① http://www.nist.gov/tac/。
② http://www.aclweb.org/aclwiki/index.php?title=Textual_Entailment。
③ http://www.aclweb.org/anthology-new/。
④ http://www.aclweb.org/aclwiki/index.php?title=RTE_Knowledge_Resources。
⑤ http://alias-i.com/lingpipe。
⑥ http://incubator.apache.org/uima/。
⑦ http://www.nltk.org/。
⑧ http://gate.ac.uk/。

然除此之外还有一些其他可为公众所利用的 NLP 框架。一些 NLP 框架还能提供命名实体识别、共指关系、切分等 NLP 模块。我们还发现，如把 NLP 工具分配于多台电脑，Thrift[㊀] 和 XML-RPC 库（比如 Apache 库[㊁]）都是很有用的资源。

许多研究小组成功研发了 NLP 标注工具。斯坦福大学[㊂] 提供了词性标注器、语法分析器和命名实体识别器，以及一些能够简化某些 NLP 编程任务的相关资源。认知计算小组（cognitive computation group）[㊃] 提供了大量的 NLP 工具程序，包括领先水平的 Illinois 命名实体标注器（Illinois named entity tagger）、共指消解器、词性标注器、组块器（chunker）以及语义角色标注器（semantic role labeler）。此外，他们还发布了他们的命名实体相似度和词汇相似度度量（Illinois-NESim 以及 Illinois-WNSim）。他们还提供了 Learning Based Java（LBJ），这是对 Java 编程语言的一种扩展，不仅简化了 Java 应用中不可缺失的一环——机器学习方法的开发与部署，而且还包含了一些有用的 NLP 工具，比如句子和单词层级的切分器。许多研究者使用了由 Michael Collins[㊄]、Dan Bikel[㊅] 和 Eugene Charniak[㊆] 开发的语法分析器。

上述提及的 NLP 工具和其他 NLP 工具还有更多的实现，有更多的文献在描述未发布的应用。本章列举的内容只是其中广受欢迎的一小部分，作为你开始深入探索的敲门砖。

6.7 总结

RTE 任务为语义推理在文本处理方面提供了一个广泛适用、与表示方法无关的框架，使得研究者能够采取许多不同方法解决实际问题。NLP 社区解决包括命名实体识别和消解在内的其他文本推断问题的方法是处理“组件”推理任务，这些任务被认为是某个全面而未指明的推断过程的一部分。一种流行的 RTE 研究方法就是把 RTE 看成是一种融合众多组件的框架，这些组件的组合方式则填补了这个总体过程的空白；正是在这种观念的指导下，我们提出了本章所阐述的 RTE 体系。 253

我们致力于寻找对于多种互不调和的需求的应对之策：

- 整合能力——对现有 NLP 资源能够随意整合，尽管这些资源在不同语言间存在粒度（词、短语、谓语结构）、形式或者实用性的不一致。
- 灵活性——能够灵活适应开发者限制条件，比如工程量和运行时复杂度等。
- 模块化——能够以一种模块化的方式添加新的 NLP 分析器和知识资源。
- 通用性——开发人员可以使用多种不同的方法进行推断。

思考 RTE 难题的一种自然的方式就是对齐，因为这一概念允许通过多重视角表现富化的文本以及融合特定的、成分级的相似度度量，进而实现知识资源模块化。在系统层次，对齐的概念能简单直接地拓展不同步骤以适应新的资源。

我们提出的框架根源发展多种语言的 NLP 资源的主流方法设计而成，旨在当拥有合适的资源时，能够开发任何语言的系统。这个框架还允许在表示的丰富性和计算速度之间做出权衡：如果使用浅层次（不够结构化）的知识库和 NLP 分析器，我们将获得一种更

㊀ http://incubator.apache.org/thrift/.
㊁ http://ws.apache.org/xmlrpc/.
㊂ http://nlp.stanford.edu.
㊃ http://L2R.cs.uiuc.edu/cogcomp.
㊄ http://people.csail.mit.edu/mcollins/code.html.
㊅ http://www.cis.upenn.edu/~dbikel/software.html.
㊆ ftp://ftp.cs.brown.edu/pub/nlparser/.

为简单的推理算法和一种更为迅捷的计算过程。用户工作时也能够使用 NLP 资源更少的语言；尽管复杂推理可能受限于 NLP 资源的可获得性，但是在更浅显的表示层次上开发出 RTE 系统仍然可能。

在我们对这一领域有前景的研究综述中，我们阐明了解决 RTE 难题不同方面的不同方法，包括表示法、背景知识资源应用、对齐方法以及推理技术。为了使读者能够把从这些内容中获得的启发融入自己的 RTE 体系中，我们指出了每种方法的执行是如何与我们的框架相匹配的。

RTE 是一个复杂的问题，解决方法则需要严密的计划和辛勤的付出。我们的目标是为你提供一种工具，通过这种工具，你可以用一种模型迅速上手，该模型可以运行扩展，从而在特定的子问题上取得提升。我们还提供了相关研究和有用资源的介绍。

参考文献

[1] E. Hovy. "Learning by reading: An experiment in text analysis." In *Text, Speech and Dialog*, vol. 4188 of *Lecture Notes in Computer Science*, pp. 3–12, Berlin: Springer, 2006.

[2] Homeland Security Newswire, "DARPA awards BBN $30 million in machine reading project," 2009. http://www.homelandsecuritynewswire.com/darpa-awards-bbn-30-million-machine-reading-project.

[3] I. Dagan, O. Glickman, and B. Magnini, "The PASCAL Recognising Textual Entailment Challenge," *Lecture Notes in Computer Science*, no. 3944, pp. 177–190, 2006.

[4] M.-C. de Marneffe, A. N. Rafferty, and C. D. Manning, "Finding contradictions in text," in *Proceedings of ACL-08: HLT*, pp. 1039–1047, 2008.

[5] L. Bentivogli, I. Dagan, H. T. Dang, D. Giampiccolo, and B. Magnini, "The fifth PASCAL Recognizing Textual Entailment Challenge," in *Proceedings of the 2nd Text Analysis Conference (TAC)*, 2009.

[6] Y. Mehdad and B. Magnini, "A word overlap baseline for the recognizing textual entailment task," 2009. http://hlt.fbk.eu/sites/hlt.fbk.eu/files/baseline.pdf.

[7] H. Dang and K. Owczarzak, "Overview of the TAC 2009 summarization track," in *Proceedings of the 2nd Text Analysis Conference (TAC)*, 2009.

[8] S. Harabagiu and A. Hickl, "Methods for using textual entailment in open-domain question answering," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 905–912, 2006.

[9] A. Celikyilmaz, M. Thint, and Z. Huang, "A graph-based semi-supervised learning for question-answering," in *Proceedings of the Annual Meeting of the Association of Computational Linguistics*, pp. 719–727, 2009.

[10] D. Roth, M. Sammons, and V. Vydiswaran, "A framework for entailed relation recognition," in *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL)*, 2009.

[11] S. Padó, M. Galley, D. Jurafsky, and C. D. Manning, "Robust machine translation evaluation with entailment features," in *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 297–305, 2009.

[12] N. Chambers, D. Cer, T. Grenager, D. Hall, C. Kiddon, B. MacCartney, M.-C. de Marneffe, D. Ramage, E. Yeh, and C. D. Manning, "Learning alignments and leveraging natural logic," in *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pp. 165–170, 2007.

[13] S. Mirkin, L. Specia, N. Cancedda, I. Dagan, M. Dymetman, and I. Szpektor, "Source-language entailment modeling for translating unknown terms," in *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 791–799, 2009.

[14] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, 1998.

[15] D. Roth and M. Sammons, "A unified representation and inference paradigm for natural language processing." Tech. Rep. UIUCDCS-R-2008-2969, UIUC Computer Science Department, 2008.

[16] G. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller, "Wordnet: An on-line lexical database," *International Journal of Lexicography*, vol. 3, no. 4, pp. 235–312, 1990.

[17] B. MacCartney and C. D. Manning, "An extended model of natural logic," in *The 8th International Conference on Computational Semantics (IWCS-8)*, 2009.

[18] M. Sammons, V. Vydiswaran, T. Vieira, N. Johri, M.-W. Chang, D. Goldwasser, V. Srikumar, G. Kundu, Y. Tu, K. Small, J. Rule, Q. Do, and D. Roth, "Relation alignment for textual entailment recognition," in *Proceedings of the 2nd Text Analysis Conference (TAC)*, 2009.

[19] R. Wang, Y. Zhang, and G. Neumann, "A joint syntactic-semantic representation for recognizing textual relatedness," in *Notebook Papers and Results, Text Analysis Conference (TAC)*, pp. 133–139, 2009.

[20] A. Iftene and M.-A. Moruz, "Uaic participation at RTE5," in *Notebook Papers and Results, Text Analysis Conference (TAC)*, pp. 367–376, 2009.

[21] A. Hickl and J. Bensley, "A discourse commitment-based framework for recognizing textual entailment," in *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pp. 171–176, 2007.

[22] S. Harabagiu and A. Hickl, "Methods for using textual entailment in open-domain question answering," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 905–912, 2006.

[23] A. Hickl, "Using discourse commitments to recognize textual entailment," in *Proceedings of the 22nd International Conference on Computational Linguistics (COLING)*, 2008.

[24] V. Punyakanok, D. Roth, and W. Yih, "Natural language inference via dependency tree mapping: An application to question answering," 2004. http://hdl.handle.net/2142/11100.

[25] M. Kouylekov and B. Magnini, "Recognizing textual entailment with tree edit distance algorithms," in *Proceedings of RTE 2005*, 2005.

[26] Y. Mehdad, M. Negri, E. Cabrio, M. Kouylekov, and B. Magnini, "Edits: An open source framework for recognizing textual entailment," in *Notebook Papers and Results, Text Analysis Conference (TAC)*, pp. 169–178, 2009.

[27] Y. Mehdad, M. Negri, E. Cabrio, M. Kouylekov, and B. Magnini, "EDITS: An open source framework for recognizing textual entailment," in *Proceedings of the 2nd Text Analysis Conference (TAC)*, pp. 169–178, 2009.

[28] T. Chklovski and P. Pantel, "VerbOcean: Mining the web for fine-grained semantic verb relations," in *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*, pp. 33–40, 2004.

[29] R. Bar-Haim, I. Dagan, I. Greental, I. Szpektor, and M. Friedman, "Semantic inference at the lexical-syntactic level for textual entailment recognition," in *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pp. 131–136, 2007.

[30] R. Braz, R. Girju, V. Punyakanok, D. Roth, and M. Sammons, "An inference model for semantic entailment in natural language," in *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 1678–1679, 2005.

[31] R. Bar-Haim, I. Dagan, S. Mirkin, E. Shnarch, I. Szpektor, J. Berant, and I. Greenthal, "Efficient semantic deduction and approximate matching over compact parse forests," in *Proceedings of the 1st Text Analysis Conference (TAC)*, 2008.

[32] P. Clark and P. Harrison, "An inference-based approach to recognizing entailment," in *Proceedings of the 2nd Text Analysis Conference (TAC)*, pp. 63–72, 2009.

[33] P. Clark and P. Harrison, "An inference-based approach to recognizing entailment," in *Notebook Papers and Results, Text Analysis Conference (TAC)*, pp. 63–72, 2009.

[34] P. Harrison and M. Maxwell, "A new implementation of GPSG," in *Proceedings of the 6th Canadian Conference on AI (CSCSI'86)*, pp. 78–83, 1986.

[35] M.-C. de Marneffe, T. Grenager, B. MacCartney, D. Cer, D. Ramage, C. Kiddon, and C. D. Manning, "Aligning semantic graphs for textual inference and machine reading," in *AAAI Spring Symposium at Stanford 2007*, 2007.

[36] B. MacCartney, T. Grenager, and M. de Marneffe, "Learning to recognize features of valid textual entailments," in *Proceedings of RTE-NAACL 2006*, 2006.

[37] B. MacCartney, M. Galley, and C. D. Manning, "A phrase-based alignment model for natural language inference," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2008)*, 2008.

[38] C. Brockett, "Aligning the RTE 2006 corpus," Tech. Rep. MSR-TR-2007-77, Microsoft Research, 2007.

[39] R. Cooper, D. Crouch, J. V. Eijck, C. Fox, J. V. Genabith, J. Jaspars, H. Kamp, D. Milward, M. Pinkal, M. Poesio, and S. Pulman, "Using the framework," Tech. Rep., The FRACAS Consortium, 1996.

[40] S. Padó, M.-C. de Marneffe, B. MacCartney, A. N. Rafferty, E. Yeh, and C. D. Manning, "Deciding entailment and contradiction with stochastic and edit distance-based alignment," in *Proceedings of the 1st Text Analysis Conference (TAC)*, 2008.

[41] Y. Mehdad, F. M. Zanzotto, and A. Moschitti, "SemKer: Syntactic/semantic kernels for recognizing textual entailment," in *Notebook Papers and Results, Text Analysis Conference (TAC)*, pp. 259–265, 2009.

[42] S. Bloehdorn and A. Moschitti, "Combined syntactic and semantic kernels for text classification," in *Proceedings of the 29th European Conference on IR Research (ECIR)*, 2007.

257

[43] M.-W. Chang, D. Goldwasser, D. Roth, and V. Srikumar, "Discriminative learning over constrained latent representations," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 429–437, 2010.

[44] M. Chang, L. Ratinov, and D. Roth, "Constraints as prior knowledge," in *ICML Workshop on Prior Knowledge for Text and Language Processing*, pp. 32–39, July 2008.

[45] Q. Do, D. Roth, M. Sammons, Y. Tu, and V. Vydiswaran, "Robust, light-weight approaches to compute lexical similarity," Computer Science Research and Technical Reports, University of Illinois, 2010. http://L2R.cs.uiuc.edu/~danr/Papers/DRSTV10.pdf.

[46] P. Dienes and A. Dubey, "Antecedent recovery: Experiments with a trace tagger," in *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pp. 33–40, 2003.

[47] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of English: The Penn Treebank," *Computational Linguistics*, vol. 19, no. 2, pp. 313–330, 1994.

[48] S. Mirkin, R. Bar-Haim, E. Shnarch, A. Stern, and I. Szpektor, "Addressing discourse and document structure in the RTE search task," in *Proceedings of the 2nd Text Analysis Conference (TAC)*, 2009.

[49] R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi, and B. Webber, "The Penn Discourse Treebank 2.0," in *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, 2008.

[50] C. Matuszek, J. Cabral, M. Witbrock, and J. DeOliveira, "An introduction to the syntax and content of CYC," in *Proceedings of the 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, 2006.

[51] D. Lin and P. Pantel, "DIRT: Discovery of inference rules from text," in *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining 2001*, pp. 323–328, 2001.

[52] A. Yates, M. Banko, M. Broadhead, M. Cafarella, O. Etzioni, and S. Soderland, "TextRunner: Open information extraction on the web," in *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pp. 25–26, 2007.

[53] I. Szpektor, I. Dagan, R. Bar-Haim, and J. Goldberger, "Contextual preferences," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) with the Human Language Technology Conference (HLT) of the North American Chapter of the ACL*, pp. 683–691, 2008.

[54] M. Sammons, V. Vydiswaran, and D. Roth, "Ask not what Textual Entailment can do for you...," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 1199–1208, 2010.

第7章

Multilingual Natural Language Processing Applications: From Theory to Practice

多语情感与主观性分析

Carmen Banea, Rada Mihalcea, Janyce Wiebe

7.1 概述

主观性（subjectivity）分析和情感（sentiment）分析以自然语言中的个人陈述，例如意见（opinion）、感情（emotion）、情感（sentiment）、评价（evaluation）、信念（belief）以及推测（speculation）等，为主要研究目标。主观性分析对文本进行主观和客观的分类标注，而情感分析则更进一步地将主观性文本划分为正向文本、负向文本以及中性文本。

到目前为止，大量的文本处理应用程序已经使用自动情感和主观性分析技术，包括自动的有表现力的语音合成（text-to-speech synthesis）[1]，在网上论坛和新闻中跟踪情绪时间表[2, 3]，以及对于产品评论的情感挖掘[4]等。在许多自然语言处理任务中，主观性分析以及情感分析已经作为生成更有效数据的第一层过滤，许多研究工作，如问答系统[5]、对话摘要（conversation summarization）[6]以及文本语义分析[7, 8]等，均可从中受益。

目前，大多数的情感分析以及主观性分析研究都是以英语为研究目标的。然而，对于其他语言，包括日语[9, 10, 11, 12]、中文[13, 14]、德语[15]以及罗马尼亚语[16, 17]等的研究也日益引起研究者的重视。另外，在NTCIR-6[18]的"中日观点提取"任务中，一些与会者进行了英语以外的其他语言的主观性分析和情感分析[1]。

由于互联网使用者中仅有29.4%的人使用英语[2]，对于构建英语以外的其他语言的主观性和情感分析的资源和工具的需求日益增大。本章我们将回顾多语言主观性分析以及情感分析的主要研究方向，重点关注资源以及工具的发展。我们将特别介绍并综述了三大类方法：1）7.4节简要阐述了基于词语和短语的标注方法；2）7.5节描述句子标注的方法；3）7.6节将对基于文本层次标注的方法进行说明。

我们将阐述多语言以及跨语言的方法。对于多语言的方法，我们回顾除英语以外其他语言的相关工作，在这些工作中，资源和工具是为了特定的目标语言开发的。在7.3节中，我们还将简要介绍一些关于这个类别的、基于英文数据的主要研究方向，同时特别强调那些可用于其他语言的方法。而对于跨语言的方法，我们描述几种已经提出的方法，这些方法通过语言映射的方式来利用现有的英文资源和工具。

7.2 定义

在各种书面或者口头的论述中，作者或者演讲者的一些思想或情感上的陈述，以及一些有关实体引用的论述是一种重要的信息。例如，新闻通常除了报道事实外还带有感情倾

---

[1] NTCIR是一系列由日本学术振兴会资助的评测研习会，目标任务有信息提取、文本摘要、信息抽取以及其他任务。NTCIR-6、7和8包含对汉语、英语及日语的多语观点分析。

[2] www.internetworldstats.com/stats.htm，2008年6月20日。

向，社论、评论、博客以及政治演讲传达着作者或者演讲者的意见、信仰以及意图。一个参加补习班的学生可能表达他的理解或者疑惑。Quirk 等人对此定义了一个专用术语：**私人状态**（private state），用于表明思想上或情感上的状态 [19]。用他们的话说，私人状态是一种并非可客观地观察或验证的状态。“有时会看到一个人断言上帝存在，但他未必相信上帝存在。信念在这种意义上是私人的”。**主观性**是表示私人状态的语言学术语，是从文学理论中改编而来 [20]。**主观性分析**是识别一个私人状态什么时候被表达出来以及该状态相关属性的工作。这里的属性主要包括谁表达了这个私人状态、表述的态度的类型、私人状态的对象是谁或者表达了什么，以及私人状态的倾向性（例如其是否是正向的或者是负向的）等。例如，考虑如下一个句子：

The choice of Miers was praised by the Senate's top Democrat, Harry Reid of Nevada.

（迈尔斯的决定被来自于内华达州的民主党领袖哈利·里德议员表扬了。）

在这个句子中，短语“被表扬了”（was praised by）表明了该句表述了一个私人状态。这个私人状态，根据这个句子的作者所说，是由里德（Reid）所表达的，并且它是有关于迈尔斯（Miers）的选择，他在 2005 年 10 月由布什（Bush）总统提名到最高法院。态度的类型是一种情感（评价、感情或者判断），它的倾向性是正向的 [21]。

本章主要关注的是主观性存在与否的检测，以及更进一步地判断它的倾向性。这些判断可以通过许多维度得来。其中之一是上下文。一方面，没有上下文，我们也可能通过词语来判断文本的主观性以及倾向性：“爱”（love）是一个主观的、正向的词语，而“恨”（hate）是一个主观的、负向的词语。另一个极端是，我们拥有语言“完全”的上下文信息，如在文本或对话中使用的语言。事实上，从无上下文到有上下文是一个连续的状态，我们可以在这个连续的状态上定义许多自然语言处理任务。

首先是构建一个词语级别的主观性词典。这里的词典是一个包含了许多带有主观性特征的关键词列表，倾向性信息常被添加到这样的词典当中。除了“爱”和“恨”之外，还有例如“杰出”（brilliant）、“兴趣”（interest）（正向倾向性）以及“警告”（alarm）（负向倾向性）等。 [250]

我们也可以根据词语的主观性以及倾向性对它们的词义进行分类。考虑以下两个来自于 WordNet [22] 的有关“Interest”的解释：

- 兴趣，涉及——一种对某人或者某物关心或好奇的感受，如“an interest in music”中的 interest。
- 利率——借钱的固定开销，通常是借款总额的百分数，如“how much interest do you pay on your mortgage?”中的 interest。

第一个解释是主观的，带有正向倾向性。但是第二个解释则不是（非主观的解释被称作是客观解释）——它并没有涉及私人状态。例如，再考虑名词“difference”的意思：

- 不相同——一种不相似的性质，如“there are many differences between jazz and rock”中的 difference。
- 偏差、偏离、差异（一种偏离于标准的差异），与“the deviation from the mean”中的 deviation 词义相同。
- 争议、不同看法、冲突（对一些重要事情的不同看法以及争议），与“he had a dispute with his wife”中的“dispute”词义相同。
- 差别（一种重要的变化），如“his support made a real difference”中的 difference。
- 剩余、差（减法后剩余的数）。

第一个、第二个以及第五个定义是客观的，而其他几个定义则是主观的。有趣的是，第三个解释是带有负面倾向性的（表明两个人之间发生冲突），然而第四个解释却是带有正面倾向性的。

词和词义级别的主观性词典是非常重要的，因为对上下文主观性分析（contextual subjectivity analysis）[23]——从一个具体的文本或者对话中识别并提取私人陈述来说，它们是非常有用的资源。我们可以从多个不同的层面来判断文本的主观性以及倾向性。在文档层面，我们可以考虑文本是否具有观点倾向，如果是的话，那么判断它主要是正向的还是负向的。我们可以进行更为细致的分析，并判断句子是否表达了主观性。例如，考虑以下来自于 Wilson [23] 的例子。第一个句子是主观的（具有正向倾向性），而第二个句子则是客观的，因为它并不带有任何主观性的描述：

- He spins a riveting plot which grabs and holds the reader's interest.
- The notes do not pay interest.

更进一步，可以对每个表达进行判断，例如，第一个句子中的 "spins"、"riveting" 和 "interest" 可能被判断为主观性的表达。一个更有意思的例子如下，"Cheers to Timothy Whitfield for the wonderfully horrid visuals"。虽然在一个词级别的主观性词典中，"horrid" 应当被当做一个负向倾向性词，但是，在这个上下文中，它具有正向倾向性。"wonderfully horrid" 表达了一种对于 "visuals" 的正向情感。（相似地，"Cheers" 表达
251 了对于 Timothy Whitfield 的正向情感）。

7.3　英语中的情感及主观性分析

在描述现有多语言情感及主观性分析的工作之前，我们将简要地概述英语研究工作的主线，同时介绍一些在英语分析中最常用的资源。通过跨语言映射或者单语（monolingual）与多语（multilingual）的举引（bootstrapping）方法，上述的一部分资源和工具已成为建立其他语言资源的基石。正如即将详细描述的那样，在跨语言映射方法中，标注好的资源可以通过平行语料库映射到另外一种语言，以生成面向该语言的资源。在多语言学习方法中，除了通过跨语言映射所获得的标注数据之外，源语言以及目标语言的单语语料库也可以通过例如协同训练的学习方法一起使用，以改进方法的效果。

7.3.1　词典

一个最经常使用的词典是 OpinionFinder 系统 [21] 中提供的主观性以及情感词典。该词典包含人工标注的资源，以从语料库中学习的条目作为扩展，共包含有 6856 个不同的条目，其中 990 个是多词表达。词典中，每个条目都被标注了词性以及可信度：最经常出现在主观性文本中的词语有较强的主观性可靠性，而那些较少出现，但出现次数仍然高于 "偶然" 的词语则被标注为较弱的主观性可靠性。此外，每个条目还被标注了极性，表明与之相符的词语或者短语是正向的、负向的还是中性的。例如，以下一个条目来自于 OpinionFinder 词典：type＝strongsubj、word1＝agree、pos1＝verb、mpqapolarity＝weakpos，这表明词语 agree 作为动词来使用的时候带有很强的主观性，并且它带有弱正向倾向性。

另一个经常用于极性分析的词典是 General Inquirer 所提供的词典 [25]。该词典包含 10 000 个词，并将这些词分为 180 个类别，这些类别信息广泛用于内容分析。词典包含语义类别（例如 animate、human）、动词类别（例如 negatives、becoming verbs）、认识取向类别（例如 casual、knowing、perception）以及其他。在 General Inquirer 中，最大的两

个类别是 valence 类，它是一个包括 1915 个正向词和 2291 个负向词的词典。

SentiWordNet [26] 是一个基于 WordNet 的资源，主要用于观点挖掘。它为 WordNet 中的每一个同义词集分配了一个量化的三元组（正向、负向以及客观），表明同义词集中的词语在这三个属性上的强度。SentiWordNet 的标注是从一系列人工标注的同义词集中自动生成的。如今，SentiWordNet 中包含了自动标注的 WordNet 的所有同义词集，总和超过了 100 000 个词语。

7.3.2 语料库

标注了主观性以及情感的语料库不仅用于训练情感自动分类器，它们也可以用于观点 252
挖掘词典的抽取。例如，在上述 OpinionFinder 词典中，大量的条目都是从一个大规模标注观点的语料库中挖掘出来的。

MPQA 语料库 [27] 最作为 2002 年“多角度问答”（Multi Perspective Question Answering，MPQA）研讨会任务的其中一部分收集并标注而来。它包括 535 条来自于各种新闻资源的英语新闻文章，文中标注了意见以及其他的私人状态（信仰、情绪、情感、推测等）。该语料库最早仅在从句和短语级别上进行标注，但是与数据集有关的句子级别的信息可以通过简单启发式方法获得 [24]。

另一个关于情感文本的人工标注的语料库是在近期的 SEMEVAL 任务 [28] 中创建并使用的。它是一个新闻标题集，共包含 1000 条用于测试的标题以及 200 条用于开发的标题，每一条标题都被标注了 6 种 Eckman 情感（生气、厌恶、悲伤、欢乐、悲伤、惊讶）以及它们的倾向性（正向或者负向）。

另外的两个数据集均为电影评论领域。一个是包含 1000 条正向评论和 1000 条负向评论的极性数据集，而另一个是包含 5000 个主观句子和 5000 个客观句子的主观性数据集。这两个数据集都是由 Pang 和 Lee 所创建的 [29]，并且已经被用于训练观点挖掘的分类器。这些领域相关的数据集有助于提高给定领域数据分类器的性能。

7.3.3 工具

目前，研究者们已经提出了大量的方法用于英语情感分析及主观性分析。这些方法大致可以分为两类：基于规则的系统，依赖于人工或者半自动建立的词典；机器学习分类器，通过有标注的语料库训练而成。

在基于规则的系统中，最常用的是 OpinionFinder [24]。基于大词典中词语或者短语的存在与否，它可以对新文本自动地进行主观性标注。简单地说，OpinionFinder 的高精确率分类器主要依靠以下三条启发式规则来进行句子的主客观标注：1）如果两个或以上的强主观性表述出现在一个句子当中，那么这个句子被标注为主观。2）如果没有强主观表述出现在句子中，并且至多有两个弱主观表述出现在前句、当前句以及下一个句子当中，那么这个句子被标注为客观。3）否则，如果前面两条规则都不适用，则这个句子被标注为未知。这个分类器利用主观性词典提供的信息以及上述规则来从大量未标记的文本中获得主客观数据。之后，这些数据用于自动抽取出一个模式集合，这些集合将以迭代的方式用于识别一个更大集合中的主客观句子。

除了这个高精确率分类器外，OpinionFinder 还包括一个高覆盖率分类器。高精确率分类器被用来自动生成有标注的英语数据集，然后数据集用于训练一个高覆盖率主观性分类器。

在 MPQA 语料库中进行评测，上述高精度分类器拥有 86.7% 的准确率和 32.6% 的召

[263] 回率，而高覆盖率分类器拥有 79.4%的准确率和 70.6%的召回率。

另一个值得一提的无监督系统是由 Turney [30] 提出的，该工作以 Hatzivassiloglou 和 McKeown [31] 的早期工作为基础，目前该系统基于自动标注的词和短语进行训练。例如，以参考词 excellent 和 poor 开始，Turney 依据当前词或短语的点互信息（Pointwise Mutual Information，PMI）与正向参考（excellent）PMI 及负向参考（poor）PMI 之比[⊖]，来区分该词或短语的极性。根据这种方法得到的极性评分用于自动标注产品、公司或者电影评价的极性。注意到这个系统是完全无监督的，因此对于其他语言的应用来说特别有吸引力。

最后，当存在可用的标注语料的时候，使用机器学习方法来建立主观性分类器及情感分类器是很自然的。例如，Wiebe、Bruce 和 O'Hara [32] 利用一个非人工标注主观性信息的数据集来训练一个机器学习分类器，这使得结果相较于基准系统有了显著的提升。类似地，从半自动建立的数据集出发，Pang 和 Lee [29] 创建了句子层次的主观性标注分类器以及文档层次的情感标注分类器。当存在标注数据时，上述的机器学习分类器也可以很好地用于其他语言。

7.4　词级和短语级标注

对于情感分析以及主观性分析工具和资源的开发往往从词典的构建开始，词典中包含标注过情感或主观性的词或短语。通过考虑文本是否包含词典中的条目，这类词典已成功地应用于基于规则的自动观点标注分类器的构建中。

到目前为止，主要存在三类词级和短语级标注的方法：1）人工标注，这涉及人对选定词和短语的判断；2）基于如字典等知识源的自动标注方法；3）基于语料库导出的信息的自动标注。

7.4.1　基于字典的方法

建立面向新语言的观点词典的一种最简单的方法是使用双语字典对已经存在的源语言词典进行翻译。Mihalcea、Banea 和 Wiebe [16] 通过使用一个英语-罗马尼亚语的双语字典，翻译了来自于 OpinionFinder（于 7.3.1 节描述）的英语主观性词典，生成了一部罗马尼亚语的主观性词典。

在翻译过程中会遇到许多挑战。首先，英语主观性词典中有屈折变化词。但为了能够利用双语词典对条目进行翻译，词形必须还原。然而，词形还原可能会导致词主观性的丢失。例如，memories 的原形是 memory，一旦翻译成罗马尼亚语（翻译为 memurie），它
[264] 的主要意思便是客观的，表示记录信息的能力。

其次，无论是词典还是双语字典都无法提供单个条目的义项信息，因此翻译只能选择目标语言中最可能的义项。幸运的是，一些双语字典将翻译使用的频率以反序列出，这是一个启发式信息，可用于部分地解决这个问题。而且，词典中有时会包含几个相同的条目，但它们表达了不同的词类，例如，grudge 有两个独立的条目，分别表示名词和动词两个角色。

使用这种直接翻译过程，Mihalcea 等人得到了一个包含 4983 个条目的罗马尼亚语主观性词典。表 7-1 列出了该词典中的一些样例条目，以及它们原本的英语形式。这个表还列出了情感的可靠性（强、弱）以及词性——由英语主观性词典提供的属性。

⊖ 两个词 w_1 和 w_2 的 PMI 定义为这两个词共同出现的概率除以每个词单独出现的概率。$\mathrm{PMI}(w_1,w_2)=\frac{P(w_1,w_2)}{P(w_1)P(w_2)}$。

表 7-1 罗马尼亚主观词典中条目的样例

罗马尼亚语	英语	属性	罗马尼亚语	英语	属性
mirunmiezy	beautifying	强，动词	plin de regret	full of regrets	强，形容词
nutaSi	nurshe	弱，形容词	sclav	slavus	弱，名词

为了评估该词典的质量，两个母语为罗马尼亚语的标注者分别标注了 150 个随机挑选的条目的主观性。每一个标注者独立地阅读了大约 100 个从网页当中抽取出来的例子，其中包括大量来自于新闻的资源。词语的主观性最终由它最常出现位置的上下文决定，并且考虑了它在网页当中最经常出现的意思。在经过讨论解决分歧以后，最终的翻译集里包含 125 个正确的翻译条目，其中包含 49.6%（61 个）的主观性条目，然而有 23.6%（29 个）的条目主要用在客观陈述中（其他的 26.8%是混合的）。

Mihalcea 等人 [16] 的研究表明，从翻译中衍生的罗马尼亚语的主观信息的可靠性比原来英文集合中信息的可靠性要弱。在许多情况下，主观性信息在翻译的过程当中丢失了，这种现象发生的主要原因是词在源语言、目标语言或两者中只有歧义。例如，词 "Iragile" 准确地翻译成罗马尼亚语是 "Iragil"，这个词通常用于指代那些易碎品，而这样的翻译就使得这个词丢失了关于 "易损坏" 的主观性信息。而有的词一旦被翻译，就将完全失去它的主观性。例如，"one-sided" 翻译成罗马尼亚语是 "cu o singura latură"，意思是 "只有一面"（用于描述物体）。

Kim 和 Hovy [15] 从英文词典出发，使用类似的翻译方法创建了一个德语词典。该词典主要关注极性而非主观性。他们使用的英文极性词典是一个通过使用少量种子词以及 WordNet 结构 [22] 而半自动生成的词典。简而言之，对于给定的种子词，可以从 WordNet 中抽取它的同义词集和同义词，然后计算该词属于二类中某一类的概率，概率是根据特定类中的种子在该词语的扩展范围内出现的数量以及频率来计算的。因而，这种计算方式代表了词与种子的相近程度。使用该方法，Kim 和 Hovy 生成了一个包含约 1600 个动词以及 3600 个形容词的英文词典，并将这些词语根据它们的极性分类为正向词和负向词。 265

该词典之后被翻译为德文，使用了一个自动生成的翻译字典。翻译字典是根据词对齐从欧洲议会语料库中得到的 [33]。为了评价该德语极性词典的质量，词典中的条目在基于规则的系统中使用，该系统随后用于标注 70 封德语邮件的极性。整体上，该系统在标注正向极性上能取得了 66%的 F 值，在标注负向极性上的 F 值为 50%。

Banea、Mihalcea 和 Wiebe [34] 提出了另一种建立主观性词典的方法，根据一些人工选择的种子，使用繁衍方法来建立主观性词典。在每一次迭代过程中，方法根据在线词典中得到的相关词语来扩展种子集合，这些相关词语通过使用一种词相似度度量来进行过滤。繁衍方法的过程如图 7-1 所示。

上述方法从一个包含主观性词的种子集合开始，均匀地从动词、名词、形容词和副词中进行采样，以条目是否出现在词典中为基础，将新发现的相关词添加到词典中。对于每一个种子词，收集所有出现在它定义中的开放类词。如果种子存在同义词和反义词，那么这些词也一并收集。值得注意的是，在这里，词的歧义并不是一个问题，因为对于每个候选词，方法都将扩展它所有可能的意思。随后，使用从目标语言语料库中训练的潜在语义分析系统来计算词与种子词的相似度，根据这个相似度度量，不正确的词义会被过滤掉。

在罗马尼亚语实验中，以 60 个种子词为基础，Banea 等人创建了一个包含有 3300 个条目的主观性词典。然后，通过将一个基于规则的分类器嵌入该词典中，并且对 504 个人工标注的句子进行主观性分类，从而对词典的质量进行了评价。上述分类器的最终 F 值为 266

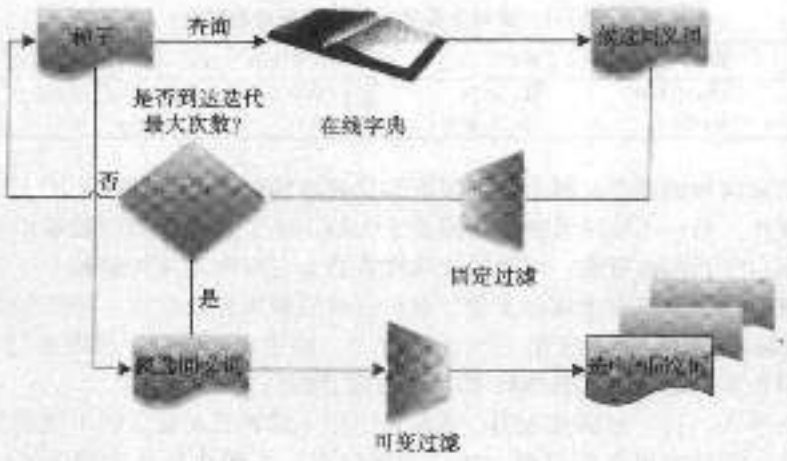


图 7-1　举例过程

61.7%，显著高于基于默认分配多数类的、F 值为 54% 的简单基准系统。

Pitel 和 Grefenstette [35] 使用了一个相似的举例方法来建立一个法语情感词典。他们将词分为了 44 个情感类（例如道德（morality）、爱（love）、犯罪（crime）、不安全（insecurity）），每个类都与一个正向倾向或负向倾向相联系。以一些种子词（每个类中有 2~4 个种子词）为起始，他们使用了同义词扩张方法来自动添加每一类的新候选词。然后，通过一种相似度计算的方法来对候选词语进行过滤，这里的相似度是通过从种子数据训练得来的潜在语义分析及机器学习系统计算的。使用这种方法，Pitel 和 Grefenstette 得到了一个包含 3500 个词的法语情感词典，并通过对比包含人工标注项的标准数据集来评估该词典。结果表明，随着训练词典中可用训练样本的增多，对于给定的类，分类的 F 值从 12% 提高到了 17%，并最高达到了 27%。

7.4.2　基于语料库的方法

除了词典以外，研究者也发现文本语料库可用于挖掘词和短语的主观性和极性信息。迄今为止，大多数基于语料库的研究方法都延续 Turney 的工作 [30]（见 7.3.3 节），他提出了一种基于正面或者负面种子（比如 "excellent" 和 "poor"）的 PMI 相关度的词的极性度量方法。

Kaji 和 Kitsuregawa [36] 提出了一种通过测量自动收集来自网页的正面和负面数据的关联强度来构建日语情感词典的方法。首先，通过使用 HTML 网页布局的结构化信息（例如，列举能明确地指示评论中评价部分存在的标记表，如优点、缺点、减、加等），以及日本独特的语言结构（例如，助词被用作主题标记），该方法自动从网络中挖掘出一个具有正面和负面陈述的语料库。以 10 亿 HTML 文档为起点，上述方法收集了大约 500 000 个极性句子，其中有 220 000 个是正面的，其余的是负面的。由两个人人工验证 500 个句子，结果表明，该方法的平均精确率为 92%，这表明通过这种方法可以构造具有相当质量的语料库。

接下来，Kaji 和 Kitsuregawa 使用该语料自动获取一个包含极性短语的集合。开始时，他们把所有的形容词和形容词短语作为候选，测量了这些候选与正面和负面数据之间的卡方和 PMI 值，然后从中选择超过一定阈值的词和短语。实验表明，PMI 比卡方的效果好。基于 PMI 的词或短语的极性值定义为：

$$PV_{PMI}(W) = PMI(W, pos) - PMI(W, neg)$$

其中

$$PMI(W, pos) = \log_2 \frac{P(W, pos)}{P(W)P(pos)} \quad PMI(W, neg) = \log_2 \frac{P(W, neg)}{P(W)P(neg)}$$

pos 和 neg 表示从网络中自动收集的正面和负面的句子。

通过使用一个含有 405 个形容词短语的数据集，其中包括 158 个正面短语、150 个负面短语和 97 个中性短语，Kaji 和 Kitsuregawa 构建了一个词典。条目的数量为 8166～9670 条，主要取决于选取候选时所用的阈值。当阈值为 0 时，正面短语的精确率为 76.4%（召回率 92.4%），当阈值上升到 3.0 时，精确率上升到 92.0%（召回率 65.8%）。在同样的阈值下，负面短语的精确率从 68.5%（召回率 94.0%）变化为 87.9%（召回率 62.7%）。

另一个基于语料库的、构建日语极性词典的方法是由 Kanayama 和 Nasukawa [12] 提出的，方法主要专注于特定领域的术语。这些研究人员提出了一种新方法，通过从建立的领域无关的词典中自动获取给定领域的极性原子，该方法能够进行无监督的领域相关的情感分析。在他们的工作中，极性原子定义为“能够被人们理解的、并可明确子句极性的最小语法结构”。通常代表一个由倾向性和一个动词或者形容词以及它们的可选论元组成的元组。系统根据句内和句间的共现来确定极性转换，并以增量方法自动产生了一个领域相关的极性词典。

首先，使用语法分析器的输出来确定候选命题。接下来，分两个阶段进行情感分析。以基于英语情感词典的已有极性原子词典为起始，该方法从早先抽取的命题中发现共现条目。这些命题被划分为正面或负面，依据是命题中包含的原子的类别标签，如果遇到否定则使用相反的标签。下一步涉及将初始的情感标识扩展到未标记的命题上。为此，方法考虑了上下文的共现，这假定在一个给定的上下文中极性并不会发生改变，除非遇到转折连词。最后，对于每个新极性原子，根据它在正面和负面上下文中分别出现的总数目来计算术语的置信度。

上述方法对从 4 个领域中抽取的日文产品评论进行评价，这 4 个领域分别为：数码相机、电影、手机和汽车。每个语料库中评论的数量从 155 130（手机）到 263 934（数码相机）不等。以这些数据集合为起点，该方法可以从每个领域中抽取出 200～700 个极性原子。经过人工评价，这些原子的精确率从 54%（手机语料）～75%（电影语料）不等。

Kanayama 和 Nasukawa 的方法在某种程度上与由 Kobayashi 等人先前提出的方法类似。后者从在网络上挖掘的日文产品评论上抽取观点三元组 [9]。情感三元组由以下域组成：产品（product）、属性（attribute）和值（value）。该过程涉及由两步组成的学习过程。第一步是基于一个共现模式集生成候选。这些集合应用于一组网络评论。此外还依赖于三个字典（主语、属性以及值的字典）。这三个字典在每次学习迭代结束时均会更新。一旦产生了候选的排序列表，采用人工判断的方式来对排序最高的候选词进行标注。这里人工参与的步骤涉及识别属性和值，以及用新抽取的实体来更新对应的字典。

对于实验，Kobayashi 等人使用了两个数据集，分别包括 15 000 条汽车评论和 10 000 条游戏评论。在学习方法开始时，他们使用一个包含 389 个车名和 660 个计算机游戏名的主语词典、一个包含 7 个属性描述的初始通用属性列表（例如，成本（cost）、价格（price）、性能（performance）），以及一个包含 247 个条目的值列表（例如，好（good）、漂亮（beautiful）、高（high））。每一个抽取的模式都根据抽取的表达式的频率及其可靠程度进行评分。对于评测，标注人员标注了 105 条汽车评论和 280 条计算机游戏评论，并识

别数据的属性以及相应的属性值。具体上，使用这个半自动系统，Kobayashi 等人发现构建观点三元组词典的速度要比纯人工创建的方式快 3 倍。此外，与人工抽取的表达相比，该半自动系统能够实现 35%~45%的覆盖率，这是非常高的。

日文短语的情感倾向性也是 Suzuki、Takamura、Okumura [10]、Takamura、Inui 和 Okumura [11] 工作的研究目标。两者都使用了从标注数据上训练的期望最大化模型。Takamura 等人考虑寻找像“轻笔记本电脑”之类短语极性的任务。这类短语不能直接根据单独词的极性来获得整个短语的极性（因为在这种情况下，“轻”和“笔记本电脑”均为中性）。在一个从日语报纸上抽取的包含 12 000 条日文形容词-名词短语的数据集上，他们发现基于三角形和 U 形图依赖的模型可以达到约 81%的准确率。

Suzuki 等人使用了与 Kobayashi 等人 [9] 类似的方法，但目标为评价性表达（evaluative expression）。他们使用期望最大化算法和朴素贝叶斯分类器来执行一个系统，进而对包含主语、属性和值的评价性表达进行极性标注。在一个包含 1061 个标注样本和 34 704 个未标注样本的数据集上，他们的方法获得了 77%的精确率。而根据 1061 个有标记样本中的最大类来进行标记的基准系统的精确率为 47%。与该基准系统相比，该结果有了显著的提升。

最后，Bautin、Vijayarenu 和 Skiena [37] 提出了另一类关于词和短语极性分析的工作。Bautin 等人工作的目标是衡量目标语言文本中给定实体（如，乔治·布什、弗拉基米尔·普京）的极性，而非获得新语言中主观性或情感词典。他们的方法先将给定语言的文档（如新闻专线、欧洲议会文档）翻译为英语，随后根据实体和英语情感词典中正面或负面单词间的关联度来计算目标实体的极性。

他们的实验考察了 9 个不同的语言（阿拉伯语、中文、英语、法语、德语、意大利语、日语、韩语以及西班牙语）以及覆盖国家和城市名的 14 个实体。他们发现在实体极性和主观性的衡量结果上，不同语言的差异很大，从非常弱的相关性（接近 0）到很强的相关性（0.60 及更高）。例如，累积不同语言中所有 14 个实体的极性分值，日语和中文文本中这些实体提及间分值的相关度只有 0.08，而法语和韩语文本收集到提及分值的相关度
259 却高达 0.63。

7.5 句子级标注

语料库标注一般是必不可少的，它要么作为多种文字处理应用的最终目标（例如，从网页挖掘意见，将评论分类为正面和负面），要么作为构建自动主观性和情感分类器的中间步骤。目前，这方面的主要工作都认为是句子级别或者文档级别的。标注的结果主要取决于最终应用（或分类器）的要求。标注过程通常使用下列两种方法：基于词典的方法，包括以规则为基础的分类器，依赖于用上一节描述的方法来构建词典；或者基于语料库的方法，需要通过已有标注数据进行训练，以获得机器学习分类器。

7.5.1 基于字典

基于规则的分类器，如由 Riloff 和 Wiebein 在 [38] 中提出的分类器，可与任何观点词典相结合来构建基于句子的分类器。这些分类器主要根据文本中词典信息的存在与否，相应地决定句子的分类，如主观/客观或正面/负面。

在上一节描述的词典中，有一个用于基于规则的分类器中，即罗马尼亚语的主观性词典。它是通过翻译英语词典的方式来构建的 [16]（见 7.4.1 节）。该分类器依赖三个启发式策略来标注主观和客观句子：1）如果两个或更多强主观表达出现在同一个句子中，该

句子则被标注为主观；2）如果句子中没有出现强主观表达，并且在之前、当前和之后句子中至多出现二个弱主观表达，那么该句子被标注为客观；3）否则，如果上述前面两条规则都不符合，该句子则被标注为未知。

为了对分类器的质量进行评价，使用了一个具有标准主观性标注的罗马尼亚语语料库。该语料库包括504个句子，这些句子来自于罗马尼亚语-英语平行语料库，并且根据[27]中的标注方案来进行标注。该分类器整体准确率为62%，召回率为39%；主观性标注的准确率为80%，召回率为21%。

另一个用于基于规则方法的主观性词典是Banea等人[34]提供的（见7.4.1节）。他们通过多次的拿衍迭代之后获得了一个包含3900个条目的罗马尼亚语词典，Banea等人根据它构建了一个基于规则的分类器，通过对上述有504句人工标注的罗马尼亚语数据集进行评估，该分类器的整体准确率和召回率为62%。这比从基于翻译的词典中获得的结果要好很多，表明特定的语言信息对主观性分析的重要性。

除了罗马尼亚语，词典方法也用于日语句子的极性分类[39]。Kanayama等人使用一种基于深度句法分析的机器翻译系统从日语产品评论中抽取“情感单元”（sentiment unit），具有很高的精确率。在这里，情感单元被定义成一个二元组，包含情感标签（正面或负面）和一个带有论元（名词）的谓词（动词或形容词）。该情感分析系统使用了基于转换的机器翻译引擎的框架，其中，产生式规则和双语词典分别被情感模式和情感词典所代替。 [270]

该系统最终不仅能够挖掘出关于产品属性的正面或负面评论，而且也提供了友好的用户界面用于浏览产品评论。使用来自目标语言句法分析器的信息，从日语中导出的情感单元可以用于句子极性的分类。系统使用大约4000个情感单元，在对200个句子进行评测时，在以降低召回率至44%为代价的同时，情感标注系统可以达到89%的高精确率。

7.5.2 基于语料库

一旦拥有带有主观性或极性标注的句子级语料库后，便可以训练一个分类器来自动标注额外的句子。

这就是Kaji和Kitsuregawa提出的方法[40，36]，他们在网络收集了大量的标注了情感的句子构成语料库，随后使用这些数据集来训练句子级的分类器。Kaji和Kitsuregawa使用了在7.4.2章节中描述的方法，该方法依赖于HTML网页布局的结构化信息以及日语的特有结构，从网络上收集到了一个约含500 000个正面和负面句子的语料库。接着，他们对标注的质量进行了评估，评估由两个人来完成。结果表明，当在随机抽取的500个样本句子上进行度量时，平均准确率可以达到92%。

Kaji和Kitsuregawa还使用了上述数据的一个子集来构建朴素贝叶斯分类器，该子集包含126 000个句子。通过选择由单句组成的人工标注评论，可以自动收集到三个领域相关的数据集（计算机、餐厅和汽车）。使用这三个数据集，分类器的准确率位于83%（计算机）~85%（餐厅）之间，这可以与从领域内数据训练而来的分类器的准确率相媲美。这些结果表明了自动构建的语料库的质量，它能够用来训练一个可靠的句子级分类器，并且分类器很容易移植到新领域。

另一个基于语料库的方法是由Mihalcea等人[16]提出的。在该方法中，通过对平行文本进行跨语言映射，能够建立一个句子级的有主观性标记的罗马尼亚语语料库。在具体实现过程中，Mihalcea等人使用了一个包含107个文档的平行语料库，文档从英语SemCor语料库[41]以及它的罗马尼亚语人工译文中获得。该语料库大约包含11 000个句子，每

部分约 250 000 个词元。此外，它是一个均衡的语料库，涵盖了体育、政治、时尚、教育以及其他领域的大量主题。

为了标注平行语料库中的英语部分，需要使用两个 OpinionFinder 分类器（在 7.3.3 节中描述）来对语料库中的句子进行标注。随后，将 OpinionFinder 标注信息映射到罗马尼亚语的训练句子中，这些句子可以用于训练朴素贝叶斯分类器，进而对罗马尼亚句子进行主观性自动标注。使用一个由 504 个人工主观性标注的句子组成的语料库（与先前节实验中使用的标准语料库一样），可以对分类器的质量进行评估。当使用高精确率分类器对英语语料库进行标注时，分类器的整体准确率为 84%。当使用高覆盖率分类器时，准确率上升到 68%。无论是哪种情况，准确率都显著高于使用主要类方法的基准系统的 54%，
271 这表明跨语言映射是一种能够构建新语言中主观性标注语料库集合的可靠方法。

使用同样的想法，即对平行文本进行跨语言映射，Banea 等人[17]提出了一种基于机器翻译的方法来生成所需的平行文本。通过将英语的句子级主观性标注自动映射到译文文本上，可以构建罗马尼亚语和西班牙语的主观性分类器。先使用罗马尼亚语作为目标语言，同样在之前描述的具有 504 个句子的标准数据集上进行测试，考虑不同的翻译情形会获得许多不同的结果。第一种情形，自动翻译人工标注的英语语料库（MPQA，参见 7.3.2 节），然后将人工标注映射到英文上，利用这些标注来训练一个分类器。如果分类器使用 SVM 分类器[42]，则可以得到 66% 的准确率。第二种情形，使用高覆盖率的 OpinionFinder 分类器对英语语料库进行自动标注，然后将标注结果映射到机器翻译的文本中，再一次，从新语言中得到的标注上训练一个 SVM 分类器，此次得到了 65% 的准确率。最后一种情形，将罗马尼亚语料库自动翻译为英文，随后使用 OpinionFinder 分类器对英语语料库进行标注，并将获得的主观性标签重新映射回罗马尼亚语语料库。在这个数据上训练的 SVM 分类器具有 67% 的准确率。

同样的实验也在西班牙语上进行。当源语言文本具有人工主观性标注时，能够得到 68% 的准确率。当标注信息是由 OpinionFinder 工具自动生成时，得到的准确率为 63%。总体而言，使用机器翻译文本获得的结果仅稍低于使用人工翻译文本时的结果，这表明机器翻译能够有效地用于生成跨语言映射技术所需的平行文本。

7.6　文档级标注

自然语言应用，如评论分类或者网页观点挖掘，往往需要具有主观性和极性标注信息的语料库。除了在以前章节中描述的句子级标注方法，研究者们也提出了许多标注整个文档的方法。与之前相同，我们主要考虑研究中的两个方向：基于字典的标注，它需要假设词典是可得的，以及基于语料库的标注，它主要依赖于通过有标记数据训练的分类器。

7.6.1　基于字典

根据特定语言词典中的已有线索，使用基于规则的系统来对文档进行标注，这也许是最简单的文档标注方法。其中一种方法是由 Wan [43] 提出，他的方法是通过使用一个极性词典以及一个带有负面词和强化成分（intensifier）的集合来标注中文评论。词典中含有 3700 正面词，3100 负面词和 148 个强调成分，所有的这些都是从 HowNet 发布的中文情感分析词汇表（vocabulary for sentiment analysis）上收集得到。此外，13 个负面词语也是从相关研究收集得到的。给定这个词典，文档的极性通过结合文档中句子的极性标注，而句子的极性则为句子中单词的极性之和。当在一个含有 886 句中文评论的数据集上

进行评测时，该方法的整体准确度为74.3%。

在Wan [43]提出的另外一种方法中，使用机器翻译将中文评论翻译为英语，紧接着 [272]
使用一个依赖于英语词典的规则系统来自动标注英语评论。使用两个商用机器翻译系统以及OpinionFinder极性词典（参见7.3.1节），他们进行了多组实验。同样使用前面提到的测试数据集，该翻译方法达到了81%的准确率，明显高于使用中文词典直接分析评论所达到的结果。此外，结合不同的翻译和方法能进一步将准确率提升为85%，这些表明不同知识源的融合可以获得比单个资源更好的性能。

此外，还有一种方法是由Zagibalov和Carroll提出的[14]。该方法是一种半监督方法，通过迭代地构建词典以及标注新文本，可以对中文本文进行极性标注。该方法首先要识别文本中的“词项”（lexical item）。词项是非字符符号间的汉字序列，并且包括一个否定词和一个状语。方法使用由人工挑选的6个否定词和5个状语组成的列表，这增加了该方法运用于其他语言的可行性。为了能成为添加到种子列表里的候选，词项需要在所考虑的数据中至少出现两次。

接下来，方法识别文本中的各种“区域”（zone），这里区域指的是标点符号之间的字符序列。整个文档的情感分数被计算为评论所包含的正面和负面区域情感分数的差值。而区域的情感分数由区域中词项的极性分数相加而来。最后，词项的极性分值与它的长度（字符数）的平方以及它的前一个极性分数成正比，而与包含该词项的区域长度成反比。当词项前面有否定词时，这个得分要乘以-1。

半监督过程由迭代的步骤组成，这些步骤会使得种子集合不断变大，标注文档的数量也不断增多。以仅包含一个形容词（good）的种子集合为起始，新的文档不断被标注为正面和负面，紧接着识别出现在文档中的能够添加到种子集合的新词项。根据词项的出现频率来决定它能否添加到种子集合中。若添加到种子集合中，词项在正向文档和负向文档中出现的频率必须相差3倍以上。当连续两轮没有找到新的种子时，半监督过程将停止。

在一个均衡的中文评论语料库上测试该方法，其中语料库是由十个不同的领域编集而来。文档级的平均准确度率为83%。此外，该系统也能在每一个领域中抽取出一个包含50～60个种子的集合，该集合可能对其他的情感标注算法有帮助。

另一个方法是由Kim和Hovy [15]提出的。该方法通过使用一个从英文翻译而来的词典对德语文档进行标注。使用了一个从7.4.1节中详细介绍的词典构造方法来生成一个具有约5000条目的英语词典。使用词对齐从欧洲议会语料库中自动生成一个翻译字典，进而可以将上述词典翻译为德语。该德语词典用于一个规则系统中，然后用系统对70封德语邮件进行极性标注。简言之，通过一个启发式方法来判断文档的极性：若文档中负面词的数目超过一定阈值，那么文档具有负面极性；反之，若文档中正面词的数目超过一定阈值，则为正面极性。总体上来看，该系统正面极性标注的F值为60%，负面极性标注的F值为50%。 [273]

7.6.2 基于语料库

假定已存在一个有标注的数据集，最直接的基于语料库的文档标注方法是训练一个机器学习分类器。Li和Sun [44]使用一个中文酒店评论数据集训练了包括支持向量机、朴素贝叶斯和最大熵的多个分类器。在包含6000个正面评论和6000个负面评论的训练集，包含2000个正面评论和2000个负面评论的测试集上，他们获得了高达92%的准确率，这取决于使用的分类器和使用的特征。这些实验证明如果存在足够的训练数据，那么构建精确的情感分类器是可行的。

Wan [45] 提出了一个相关但更为细致的方法，他使用一种能够充分平衡源语言和目标语言资源的协同训练方法。该方法在中文产品评论的自动情感分类上进行了测试。对于一种给定目标语言（中文）的产品评论，该方法通过机器翻译获得另外一种语言（英语）的评论。该算法然后使用两个支持向量机分类器，一个用于中文，另一个用于英语，进行协同训练以迭代方法来建立情感分类器。方法开始时，训练数据集包含中文及其英语翻译的标注样例集。接下来，执行第一次协同迭代训练，对未标注实例进行分类。如果两种语言分类器对未标注数据分类的标签一致，那么这些刚被标注的实例将被加入训练数据集中，用于在下次迭代中重新训练这两个分类器。在这过程中，方法不考虑分类标签不一致的评论数据。正如所预料的那样，随着迭代次数的增加，分类器的性能不断增强，之后当错误标记实例的数量超过了某个阈值时，分类器的性能开始下降。实验中，每次迭代加入5个正面和5个负面的评论，分类器在第40次迭代后达到最高性能，整体F值达到81%。由于能够充分利用跨语言和语言内的知识，该方法是一个成功的方法。

7.7　什么有效，什么无效

当面临一种新的语言时，我们在为这种语言创建一个情感或主观性分析的工具时，哪种方法最好？答案在很大程度上取决于该语言可用的单语资源和工具，如词典、大语料库、自然语言处理工具和一些与主要语言[⊖]，如英语，有跨语言关系的资源（例如，双语词典或平行文本）。

7.7.1　最佳情况：已有人工标注的语料库

当目标语言中存在人工标注的情感或主观性语料库时，是最好的情况，不幸的是，这种情况很少见，只有很少的语言才有大规模人工标注的语料库（如，英语中的MQPA语料库 [27]）。

274

另外一种可行的方案是从网络数据中得出上述语料库，比如电影或产品的评论集。对于网上存在大规模评论的语言来说这是可行的方案。例如，很多方法就依赖于网络评论源，包括英语的电影或产品评论 [29, 4]、日语的产品评论 [12, 9]，以及酒店的中文评论 [44]。

一旦拥有大规模的标注数据之后，不管语料库是人工标注的还是从公开的网站挖掘的，我们便可以通过训练一个机器学习系统来轻松地获得一个自动标注工具，该任务可以被认为是一个文本分类问题。像朴素贝叶斯、决策树和支持向量机[⊖]这样的学习算法可以用于标注新文本的主观性或情感。

7.7.2　次优情形：基于语料库的跨语言映射

次优选择是通过跨语言映射的方法从存在标注数据的主要语言中构建一个目标语言的标注数据集。这就假定在目标语言和主要语言（如英语）之间能够通过人工或自动翻译来构建两者之间的联系，联系以平行文本的形式存在。通过这种联系，主要语言的标注语料库自动迁移到目标语言中。该方法首先由 Mihalcea 等人 [16] 提出，通过罗马尼亚语-英语平行文本来映射主观性标签，并且随后和机器翻译技术一起运用，将主观性标签映射到罗马尼亚语 [17] 或将情感标注映射到中文 [43]。

⊖　即有许多已有资源和工具的语言。

⊖　通常在现有工具包，如 Weka [46] 中可以找到。

翻译可以在两个方向上进行。首先，研究人员可以使用主要语言的文本集，以人工或自动方式翻译成目标语言文本。在这种情况下，如果源端的文本已经由人工标注了主观性或情感信息（例如，MPQA），那么这些人工标注可以被映射到目标语言中。或者，主要语言的文本可以通过使用像 OpinionFinder [24] 这样的主观性或情感分析工具来进行自动标注。另一种选择是以目标语言中的文本开始，将它们翻译成资源丰富的主要语言文本。同样，翻译可以通过人工进行，也可以通过机器翻译系统来实现。

不论在哪个方向上使用翻译，不论使用的是人工创建的平行语料库还是使用机器翻译的文本，最终都可以获得有标注的主观性或情感的目标语言数据集。这些数据集用于训练一个如先前章节所述的自动分类器。

7.7.3 第三优情形：孳衍词典

很多方法依赖于主观性和情感词典来创建基于规则的分类器，以用于标注新文本。例如，最常用的英语主观性标注工具是 OpinionFinder [24]，该工具基于一个大规模的主观性词典 [47]。类似地，Turney [30] 建议的方法依赖于一个情感词字典，进而实现文本级性的自动标注。

对于建立主观性或情感词典，最成功的方法之一是以少量的人工选择的种子，通过使用孳衍方法来建立词典。在这过程中，如果该语言有可用的词性标注器和句法分析器，那么上述孳衍过程可以以信息抽取模板（extraction pattern）为基础 [48]。另一个方法是使用在电子字典中的同义词和定义信息来进行孳衍迭代 [34]。在这种情况下，除了一个目标语言的词典，不需要其他的高级语言处理工具。以一个包含所有开放类词（open-class word）的种子集合为初始数据，方法收集字典中所有相关词语，包括同义词、反义词和定义所用的词语。在这个候选词集合中，只有那些与种子词语密切相关的词语被保留下来，用于下一轮的孳衍迭代。这里的相关性是通过使用如潜在语义分析 [49] 那样的相似度指标来度量。在图 7.1 中给出了孳衍迭代过程。经过多次迭代，上述过程将产生包含几千个条目的词典。

将词典应用于基于规则的句子或文档级标注分类器，这是主观性或情感词典的典型应用。例如，在 Banea 等人 [34] 使用的分类器中，如果句子包含了 3 个或更多出现在主观性词典中的条目时，句子将被标记为主观的；如果句子含有两个或更少的条目时，则被标注为客观。文档级标注也可以通过文档中包含主观性或情感单词的频率来确定。

7.7.4 第四优情形：翻译词典

如果上述方法对于目标语言皆不可行，那么最后一种方法就是将已存在的主要语言的词典自动翻译为目标语言，以此来构建目标语言的词典。在此，唯一需要的资源为一个主要语言的主观性或情感词典和一个双语字典，用于将主要语言的词典翻译为目标语言。该方法最早被用于构建德语的情感词典 [15]，随后被应用到构建罗马尼亚语的主观性词典 [16]。

尽管非常简单和高效（几秒钟就可以创建一个超过 5000 个条目的词典），但是该方法的准确率很低，这主要是由于上下文无关翻译过程中面临的困难造成的：如何选择最合适单词翻译存在明显的困难，短语翻译的覆盖面小，词典中屈折形式和双语字典的原形形式之间并不匹配等。即便这样，这种方式构建的词典能够很容易地以人工的方式修正，因此可用于帮助创建给定目标语言的主观性或情感资源。

7.7.5 各种可行方法的比较

因为并不存在一种语言使得上述 4 种方法都可以适用，因此，要对上述 4 种方法做出

全面的比较是困难的。然而，目前，在罗马尼亚主观标注工具研究方面，研究者们做了很多实验，这些实验都是在公共数据集上进行的，因此可以让我们做部分比较。

表 7-2 给出了这几个实验的结果，实验均在一个具有 504 句人工主观性标注的数据集
276 上进行（见 7.4.1 节）。只有一个方法没有加入对比，因为它依赖于人工标注语料库，而罗马尼亚语没有人工标注的数据集。不出意外，以人工构建平行语料为基础进行跨语言映射的方法的效果最好，次优的方法是依靠从源端到目标端或者从目标端到源端的机器翻译的方法，方法的性能和前一方法很接近。当没有任何标注资料可用时，可以使用主观性词典构建一个基于规则的分类器。在这种情况下，享衍方法的效果最好，然后是基于双语词典进行跨语言映射的方法。

表 7-2　罗马尼亚语中不同主观性标记方法的对比

	精确率	召回率	F 值	准确率
平行语料	69.35	78.75	73.75	69.64
源端到目标端机器翻译	67.76	83.15	74.67	69.44
目标端到源端机器翻译	76.06	55.34	55.67	57.85
词典享衍	68.98	61.90	65.85	64.89
词典翻译	65.84	38.83	48.83	55.95

7.8　总结

情感分析和主观性分析是一个迅速发展的领域。虽然目前的工作以英文为主，但以其他语言为研究目标的工作也越来越多，目前，中文、德语、日语、罗马尼亚语、西班牙语及其他语种已经有许多可供主观性及情感分析研究的资源及工具可用。

本章介绍了多语言情感、主观性分析相关的一些最新方法。这些方法的区别主要在于它们所采用的途径：基于语料库的有监督方法或者基于规则的无监督方法，以及这些方法研究的文本范围：词、短语、整个句子，或者整个文档。虽然对现今所有方法进行对比是很困难的事情，但本章仍尝试对目前较为通用的方法进行了一个概述，并在罗马尼亚语主观性分析这个特定任务中来比较各种方法的不同。

虽然资源贫乏语种的性能与资源丰富语种（如英语）相比仍然存在较大的差距，但多语方法工作的日益增加，有希望为越来越多的语言带来资源与工具。

致谢

本章的工作部分基于自然科学基金项目 #0917170 和 #0916046。本章中表达的任何
277 观点、发现、结论以及推荐均是作者的个人观点，并不代表自然科学基金会的看法。

参考文献

[1] C. O. Alm, D. Roth, and R. Sproat, "Emotions from text: Machine learning for text-based emotion prediction," in *Proceedings of the Human Language Technologies Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP-2005)*, pp. 347–354, 2005.

[2] L. Lloyd, D. Kechagias, and S. Skiena, "Lydia: A system for large-scale news analysis," in *String Processing and Information Retrieval (SPIRE 2005)*, 2005.

[3] K. Balog, G. Mishne, and M. de Rijke, "Why are they excited? Identifying and explaining spikes in blog mood levels," in *Proceedings of the 11th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-2006)*, 2006.

[4] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining 2004 (KDD 2004)*, pp. 168–177, 2004.

[5] H. Yu and V. Hatzivassiloglou, "Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2003)*, pp. 129–136, 2003.

[6] G. Carenini, R. Ng, and X. Zhou, "Summarizing emails with conversational cohesion and subjectivity," in *Proceedings of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2008)*, 2008.

[7] J. Wiebe and R. Mihalcea, "Word sense and subjectivity," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2006.

[8] A. Esuli and F. Sebastiani, "Determining term subjectivity and term orientation for opinion mining," in *Proceedings the 11th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-2006)*, pp. 193–200, 2006.

[9] N. Kobayashi, K. Inui, K. Tateishi, and T. Fukushima, "Collecting evaluative expressions for opinion extraction," in *Proceedings of International Joint Conference on Natural Language Processing (IJCNLP)*, pp. 596–605, 2004.

[10] Y. Suzuki, H. Takamura, and M. Okumura, "Application of semi-supervised learning to evaluative expression classification," in *Proceedings of the 7th International Conference on Intelligent Text Processing and Computational Linguistics*, 2006.

[11] H. Takamura, T. Inui, and M. Okumura, "Latent variable models for semantic orientations of phrases," in *Proceedings of the 11th Meeting of the European Chapter of the Association for Computational Linguistics*, 2006.

[12] H. Kanayama and T. Nasukawa, "Fully automatic lexicon expansion for domain-oriented sentiment analysis," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2006.

[13] Y. Hu, J. Duan, X. Chen, B. Pei, and R. Lu, "A new method for sentiment classification in text retrieval," in *Proceedings of the International Joint Conference on Natural Language Processing*, pp. 1–9, 2005.

[14] T. Zagibalov and J. Carroll, "Automatic seed word selection for unsupervised sentiment classification of chinese text," in *Proceedings of the Conference on Computational Linguistics*, 2008.

[15] S.-M. Kim and E. Hovy, "Identifying and analyzing judgment opinions," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 2006.

[16] R. Mihalcea, C. Banea, and J. Wiebe, "Learning multilingual subjective language via cross-lingual projections," in *Proceedings of the Association for Computational Linguistics*, 2007.

[17] C. Banea, R. Mihalcea, J. Wiebe, and S. Hassan, "Multilingual subjectivity analysis using machine translation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, 2008.

[18] N. Kando, T. Mitamura, and T. Sakai, "Introduction to the NTCIR-6 special issue," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 7, no. 2, 2008.

[19] R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik, *A Comprehensive Grammar of the English Language*. New York: Longman, 1985.

[20] A. Banfield, *Unspeakable Sentences*. Boston: Routledge and Kegan Paul, 1982.

[21] T. Wilson, "Fine-grained subjectivity and sentiment analysis: Recognizing the intensity, polarity, and attitudes of private states," PhD thesis, University of Pittsburgh, 2007.

[22] G. Miller, "WordNet: A lexical database," *Communication of the ACM*, vol. 38, no. 11, 1995.

278

[23] T. Wilson, "Fine-grained subjectivity and sentiment analysis: Recognizing the intensity, polarity, and attitudes of private states," PhD thesis, University of Pittsburgh, 2008.

[24] J. Wiebe and E. Riloff, "Creating subjective and objective sentence classifiers from unannotated texts," in *Proceedings of the 6th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2005)*, 2005.

[25] P. Stone, *General Inquirer: Computer Approach to Content Analysis*. Cambridge, MA: MIT Press, 1968.

[26] A. Esuli and F. Sebastiani, "SentiWordNet: A publicly available lexical resource for opinion mining," in *Proceedings of the 5th Conference on Language Resources and Evaluation (LREC 2006)*, 2006.

[27] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," *Language Resources and Evaluation*, vol. 39, no. 2-3, pp. 165–210, 2005.

[28] C. Strapparava and R. Mihalcea, "Semeval 2007 task 14: Affective text," in *Proceedings of the 4th International Workshop on the Semantic Evaluations (SemEval 2007)*, 2007.

[29] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, 2004.

[30] P. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*, pp. 417–424, 2002.

[31] V. Hatzivassiloglou and K. McKeown, "Predicting the semantic orientation of adjectives," in *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, pp. 174–181, 1997.

[32] J. Wiebe, R. Bruce, and T. O'Hara, "Development and use of a gold-standard data set for subjectivity classifications," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pp. 246–253, 1999.

[33] F. Och and H. Ney, "Improved statistical alignment models," in *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, 2000.

[34] C. Banea, R. Mihalcea, and J. Wiebe, "A bootstrapping method for building subjectivity lexicons for languages with scarce resources," in *Proceedings of the Learning Resources Evaluation Conference (LREC 2008)*, 2008.

[35] G. Pitel and G. Grefenstette, "Semi automatic building method for a multidimensional affect dictionary for a new language," in *Proceedings of the 6th International Language Resources and Evaluation (LREC'08)*, 2008.

[36] N. Kaji and M. Kitsuregawa, "Building lexicon for sentiment analysis from massive collection of HTML documents," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2007.

[37] M. Bautin, L. Vijayarenu, and S. Skiena, "International sentiment analysis for news and blogs," in *Proceedings of the International Conference on Weblogs and Social Media*, 2008.

[38] E. Riloff and J. Wiebe, "Learning extraction patterns for subjective expressions," in *Conference on Empirical Methods in Natural Language Processing (EMNLP-03)*, pp. 105–112, 2003.

[39] H. Kanayama, T. Nasukawa, and H. Watanabe, "Deeper sentiment analysis using machine translation technology," in *International Conference on Computational Linguistics*, 2004.

[40] N. Kaji and M. Kitsuregawa, "Automatic construction of polarity-tagged corpus from HTML documents," in *Proceedings of the International Conference on Computational Linguistics / Association for Computational Linguistics*, 2006.

[41] G. Miller, C. Leacock, T. Randee, and R. Bunker, "A semantic concordance," in *Proceedings of the 3rd DARPA Workshop on Human Language Technology*, 1993.

[42] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer, 1995.

[43] X. Wan, "Using bilingual knowledge and ensemble techniques for unsupervised Chinese sentiment analysis," in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 2008.

[44] J. Li, , and M. Sun, "Experimental study on sentiment classification of Chinese review using machine learning techniques," in *International Conference on Natural Language Processing and Knowledge Engineering*, 2007.

[45] X. Wan, "Co-training for cross-lingual sentiment classification," in *Proceedings of the Joint Conference of the Association of Computational Linguistics and the International Joint Conference on Natural Language Processing*, 2009.

[46] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*. Boston: Morgan Kaufmann, 2005.

[47] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in *Proceedings of Human Language Technologies Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*, 2005.

[48] E. Riloff, J. Wiebe, and T. Wilson, "Learning subjective nouns using extraction pattern bootstrapping," in *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003)*, 2003.

[49] T. K. Landauer, P. Foltz, and D. Laham, "Introduction to latent semantic analysis," *Discourse Processes*, vol. 25, 1998.

281
/
282

实　践

第 8 章“实体检测和追踪”，探究确定文本中是否出现真实世界中各种类型实体的方法，这些实体包括人物、组构和地点，以及研究这些表达的边界是什么，在什么情况下这些实体是共指关系。

第 9 章“关系和事件”，侧重于从语料库中抽取文本的相关实体，相关属性及它们之间的关系，并以一种结构化的方式来存储这些信息。

第 10 章“机器翻译”，描述从一种人类语言到另一种人类语言的自动翻译方法。

第 11 章“跨语言信息检索”，探讨了根据用户的搜索查询，检索文档或部分文档的问题。

第 12 章“多语自动文摘”，讨论了自动总结文档摘要的问题。

第 13 章“问答系统”，根据语料库中的信息探索自动回答问题的方法。

第 14 章“投票”，描述了问答中一个相对较新的领域，依据语料库中文档的信息，处理有多个答案的复杂查询。

第 15 章“口语对话系统”，描述了如何建立一个能够处理人机对话的系统。

第 16 章“聚合自然语言处理引擎”，讨论了使用一个通用结构将多类自然语言处理引擎结合起来的方式。

第8章

Multilingual Natural Language Processing Applications: From Theory to Practice

实体检测和追踪

Xiaoqiang Luo, Imed Zitouni

8.1 概述

信息抽取（Information Extraction，IE）是指从自然语言文档中识别和抽取有用的文本信息。信息的“有用性”是由用户和应用所决定的。对于输入的文档，我们经常关心“谁在何时或由于什么原因（为什么）对谁做了什么”。很明显，信息抽取的范围可以是任意广泛的，甚至有时可能需要世界知识。为使问题简化，本章我们只关注以下两个子任务：

1）从文档中检测提及，并识别其属性；提及是指确定一个物理对象（如一个人物或一个组织机构）的文本块；

2）将指代相同对象的提及用实体来分组；实体是许多个指代相同对象的提及集合。

这两个子问题是对于文档理解至关重要的步骤，因为它们在语篇上确定了重要的概念对象和它们之间的关系。

第一个问题叫做**提及检测**（mention detection），包括检测某种提及的边界并有选择地确定其语义类型（如人物或组织机构）及其他属性（如名称、名词或者是代词）。第二个问题称为**共指消解**（coreference resolution），将指代相同实体的提及归结到一个等价类中。由于解决了这两个问题就可以识别一篇文档中的实体及其属性，因此这一章的内容是“实体检测和追踪”（Entity Detection and Tracking，EDT），该术语也用于 ACE [1] 项目中。

某种提及可以是名称、名词，或代词。例如，

President Ford said that he has no comments.

（福特 总统 说 他 没有 评论。）

这句话包含三个人物提及：President（总统）、Ford（福特）、he（他）。福特是一个名称，总统是一个名词，他是一个代词。显然，总统和福特指的是同一个人，我们说他们属于同一个实体。然而，由于上下文有限，他指向的是不是福特总统是有歧义的。若断章取义，福特也可能是一个组织，如“Ford sold 10 million cars in the first quarter.”（福特在第一季度卖出了 1000 万辆汽车。）正如自然语言处理中存在的许多其他问题一样，这种歧义是实体检测和追踪（Entity Detection and Tracking，EDT）面临的主要困难。

提及检测和共指消解最成功的方法就是数据驱动的统计方法。对于这种方法，训练数据集由人工标注，并且可以从数据中自动学习到统计模型。学习到的模型可以被应用到未知的文档。和一个基于规则的系统相比，统计的方法有许多优点：

- 数据驱动的方法能够迅速测试不同的算法和特征。
- 当通过添加新的数据训练集而有新的数据可用时，统计系统可以不断完善。
- 统计系统可以很容易地移植到其他语言。

本章中讨论的方法，按照把 EDT 核心算法与语言相关特性相分离的原则进行组织。事实上，要讨论的算法在不经过过多修改的情况下，便可为多种语言构建系统。这并不是

误。'我们应该忽略语言问题。相反，依赖于语言的现象可由预处理或从数据中提取特征的可配置模块来处理。例如，对于屈折变化非常丰富的语言，如阿拉伯语，空格分隔的词在EDT中可能不是一个很好的单位，而形元（morph）却往往可以解决数据稀疏问题。对于没有空格的书面语言，如中文、韩文、日文，必须要对输入文本切分为词语。另一个例子便是中文的"缩略词"：中文的新词是多个连续词语的首字、尾字或者首字尾字混合而构成的。在计算过程中，通过扩展缩略词的定义，从而可以包含这些情况，进而捕捉到这种语言学现象。

从系统结构的角度而言，有两种EDT系统：

1）级联系统：在这种系统中，一个提及检测组件后面串联一个共指消解组件。这种架构的优势是两个子系统之间存在一个清晰的界线，并且可以独立开发和改进。例如提及检测系统可以在一个数据集上训练，而共指消解系统可使用完全不同的另一套数据集进行训练。由于两者分离，系统可以很容易识别和纠正错误。级联结构的缺点是，这两个问题本是紧密相连的，但却被孤立地解决了。

2）联合系统：另一种架构是共同解决这两个问题[2]，换句话说，系统会尝试在进行提及检测的同时找到共指链；它先假定一个提及，然后寻找之前出现的可能指代；换句话说提及检测操作与共指消解操作交错进行。这种架构的优点是具有"全局"最优的系统参数，但它的时间和空间算法复杂度通常要比相应的级联系统大得多。

出于如下原因，本章中我们提出了一个样例级联系统：首先，提及检测和共指消解已足够复杂，进而它们值得单独处理；其次，级联系统使得对两个组件进行调试和错误分析比联合系统更容易；最后，这里描述的级联方法在实践中被证实具有非常好的表现[3]。 386

8.2 提及检测

提及检测任务与命名实体识别（Named Entity Recognition，NER）关系密切，命名实体识别最近在许多研究中[3，4，5，6]已成为人们关注的焦点并成为评测任务的重点：MUC-6、MUC-7、CoNLL'02和CoNLL'03都涉及命名实体识别的评测任务。在NLP文献中，一个命名实体代表了一个名称的实例，比如一个地点，一个人物或一个组织机构，而NER任务包括识别出这样一个实体的每一次独立出现[⊖]。这一章我们称具体对象或抽象物的文本引用实例为提及，它可以是人名（如*John Mayor*），名词（如*president*），或代词（如*she*，*it*）。这个任务自从在ACE 2003的竞赛中引入后，已经引起了人们极大的兴趣。

在CoNLL'03评测任务中，Florian等人提供的系统获得了最好的成绩[7]。该系统对3种不同分类器进行线性插值计算：1）隐马尔可夫模型（HMM）；2）最大熵模型（MaxEnt）；3）鲁棒的风险最小化模型（RRM）。他们获得的最终结果是英语有88.76的F值，德语有72.41的F值，在这两种语言的评测任务上都是最佳结果。[8]中描述的系统在CoNLL'03中名列第二。它是一种完全基于最大熵的方法，它使用了不同类型的特征：上下文特征和词汇化特征，如大写词。Klein等人在CoNLL'03中提出了另一种命名实体识别的方法[9]，是一种基于字符的HMM方法。这种方法非常依赖命名实体的内部特征。近期，Tran等人[10]在越南语的命名实体识别评测任务中显示，使用支持向量机方法的性能超过了使用条件随机场（Conditional Random Field，CRF）模型的方法

⊖ 在1995年的消息理解会议（Message Understanding Conference，MUC-6）中，命名实体的种类包括人名、机构名、地名、时间、百分比、货币量。

($F_{\beta=1}=87.75$ 比 86.48)。这个比较是在支持向量机和条件随机场中使用相同特征集的条件下，基于所获得的 F 值的平均值所得出的。Benajiba 与 Rosso [11] 和 Benajiba、Diab 与 Rosso 的研究表明对于提及检测与 NER，使用基于条件随机场的技术是有效的。他们报告了阿拉伯语在使用不同特征集下的结果，这些特征集包括上下文、形态、词汇特征，以及基于地名词典的特征。还有一些论文详述了自动内容抽取（Automatic Content Extraction，ACE）的结果。例如，Florian 等人 [12] 对提及检测提出了一个两步法：先边界检测，然后进行分类，与一个预测边界和提及类型的联合模型相比，这种技术会获得更好的效果。我们在下面将介绍一种用于提及检测的数据驱动方法，它使用了 MaxEnt 框架。这种方法在 ACE 评测竞赛中显示出了非常有竞争力的结果 [1]。

8.2.1　数据驱动的分类

通过为文本中每个独立词元赋予一个标注，提及检测问题可形式化为一个分类问题，这些标注编码指出一个词元是否为某个特定提及的开始符，或者某个特定提及的内部元素，或不属于任何提及。按照这样的形式化规则，提及检测便和许多其他 NLP 任务更加
[287] 相似，如基本名词短语的分块 [13]、文本分块 [14] 以及 NER [15]。

过去的研究结果表明能够融合多种信息资源 [3，4，5] 的模型框架对于获得良好性能是至关重要的。在本节中，我们介绍一个 MaxEnt 提及检测系统，在进行分类决策时可以整合任意类型的信息。当然你也可以用你最喜欢的机器学习方法取代 MaxEnt 模型，只要这些信息能够在系统中有效地运用起来。

形式上，令 $x_1^L=(x_1, x_2, \cdots, x_L)$ 是一个连续的词元序列（即一个句子或一个文档）。通过下述方法可将提及检测问题转化成一个序列的分类问题：通过分配 y_i 标签为每一个词元 x_i，其中 y_i 取自一个有限集：$y=\{t_1, \cdots, t_K\}$。例如，如果我们想要找到 PER（人物）和 ORG（组织），一个提及的潜在标注集编码可以是 $y=\{$PER-B，PER-I，ORG-B，ORG-I，O$\}$，当词元被标注为 PER-B 和 PER-I 时分别表示人物提及的开始及内部；词元被标注为 ORG-B 和 ORG-I 分别表示机构提及的开始及内部，词元被标注为 O 意味着该词元不是一个提及。注意到尽管一个合法的提及检测结果可以用唯一的 B、I、O 序列来编码，在解码阶段必须要注意排除非法标注序列：例如，如果不跟在 PER-B 标注之后，PER-I 标注是不允许单独出现的。

根据以上假设，提及检测系统的目标是当给定一个句子 x_1^L 时找到最可能的标注序列，即

$$\hat{y}_1^L=\underset{y_1^L}{\operatorname{argmax}} P(y_1^L \mid x_1^L) \tag{8.1}$$

在实践中，模型 $P(y_1^L \mid x_1^L)$ 中参数的个数通常是非常多的，以至于想要从有限的训练数据中得到一个较好的估计是不切实际的。所以该模型需由马尔式法则来分解，去掉基于较长的历史条件的假设，只考虑较短的历史条件：

$$\begin{aligned} P(y_1^L \mid x_1^L) &= P(y_1 \mid x_1^L)P(y_2 \mid x_1^L, y_1)\cdots P(y_L \mid x_1^L, y_1^{L-1}) \\ &\approx P(y_1 \mid x_1^L)P(y_2 \mid x_1^L, y_1)\cdots P(y_L \mid x_1^L, y_{L-k+1}^{L-1}) \end{aligned} \tag{8.2}$$

注意在构建基本的模型模块中只保留最近的 $k-1$ 个标注，$P(y_i \mid x_1^L, y_{i-k+1}^{i-1})$。

在这一章中，利用 MaxEnt 模型来计算 $P(y_i \mid x_1^L, y_{i-k+1}^{i-1})$：

$$P(y_i \mid x_1^L, y_{i-k+1}^{i-1})=\frac{1}{Z(x_1^L, y_{i-k+1}^{i-1})}\exp\left[\sum_{j=1}^{n}\lambda_j f_j(x_1^L, y_{i-k+1}^{i-1}, y_i)\right] \tag{8.3}$$

其中 $Z(x_1^i, y_{i-k+1}^{i-1})$ 为归一化因子，λ_j 是特征函数 $f_j(x_1^i, y_{i-k+1}^{i-1}, y_i)$ 相应的权重。对于给定的标注数据集合，已有好的学习训练算法 [16, 17, 18, 19] 可以找到最优参数，即最大化训练数据的对数似然[1]。 288

MaxEnt 方法可以无缝地将多种特征类型融合，但是它会高估低频特征的置信度。当我们对那些有着不够可靠参数估计的特征施加一些硬性约束时，这个问题会表现得更为明显。对模型作出某些调整能够解决这个问题，比如向模型添加高斯先验概率 [20] 或指数先验概率 [21]，利用模糊的 MaxEnt 边界 [22]，或者使用带有不等式约束的 MaxEnt [23]。

有各种各样的方法可用于估算 λ_j 的最优值，其中一种是顺序条件广义迭代演算（Sequential Conditional Generalized Iterative Scaling，SCGIS）技术，它能够既快速又鲁棒地进行提及检测（也能处理很多其他 NLP 问题）[18]。为了解决低频特征引起的置信度过高估计问题，我们建议从基于添加高斯先验概率的正则化方法开始 [20][2]。直观地说，这个措施使得模型的参数接近于 0 值除非有来自数据的证据表明其他。在计算类别的概率分布后，选择标准是挑出最大后验概率的那一个。8.2 节介绍的解码算法可通过动态规划算法完成序列分类。

8.2.2 搜索提及

现在，我们拥有了自己的模型，我们将用它寻找句子中的提及。这些提及有很强的交叉依赖性，如果为每一个词元进行独立的分类就无法正确地建模。

在公式（8.2）中，我们进行标注序列分类时，限制其只和前 $k-1$ 个标注有关，但我们并不对词元强加任何限制条件：概率是通过计算整个词元序列 x_1^i 而得。在实际情况中，尽管特征只检测我们感兴趣的特定词元的有限上下文，但是它们可以“向前看”，即检测当前词元后一个词元的特征。

根据公式（8.2）所描述的约束条件，公式（8.1）中的序列可以有效地被识别。为了达到这一目标，我们创建一个**分类标签格**（classification tag lattice）（也称为一个**架**（trellis））。如下所示：

- 记 x_1^i 为词元输入序列，$S=\{s_1, s_2, \cdots, s_m\}$ 为 $y^k(m=|y|^k)$ 枚举集合。我们称元素 s_j 为一个状态。每一个这样的状态对应于连续 k 个后继词元标注过程。当将元素 s_i 视为 k 个元素的向量时，我们发现它是非常有用的。我们使用 $s_i[j]$ 符号来表示 s_i 向量中的第 j 个元素（即 $x_{i-k+j+1}$ 的标注），$s_i[j_1 \cdots j_2]$ 代表该向量介于 j_1 和 j_2 之间的元素序列。
- 我们从概念上把每一个字符 x_i，$i=1, \cdots, L$ 与 S、$S^i=\{s_1^i, \cdots, s_m^i\}$ 联系起来：这个集合代表当 x_i 被检验时 x_{i-k+1}^i 所有可能的标注。 289
- 然后，我们创建从集合 S^i 到集合 S^{i+1} 的链接，其中 $i=1, \cdots, L-1$，链接具有如下式所示的属性

$$w(s_{j_1}^i, s_{j_2}^{i+1}) = \begin{cases} p(s_{j_1}^{i+1}[k] \mid x_1^i, s_{j_1}^{i+1}[1..k-1]) & \text{若 } s_{j_1}^i[2..k] = s_{j_2}^{i+1}[1..k-1] \\ 0 & \text{否则} \end{cases}$$

---

[1] 该方法的描述已超出本章范围，感兴趣的读者请自行阅读所引用的参考文献进行深入研究。

[2] 注意得到的模型并不是真正意义上的最大熵模型，它不是一个有最大熵的模型（乘积中的第二项）而是一个最大后验概率的模型。

这些权值与从 $s_{j_i}^i$ 状态到 $s_{j_{i+1}}^{i+1}$ 状态的转换概率相对应。如果状态不相容（即，没有可能的标注序列 Y，使得 Y［$i-k-1$，…，i］是词元 x_{i-k-1}^{i} 的标注序列，Y［$i-k+2$，$i+1$］是词元 x_{i-k+2}^{i+1} 的分类标注序列），那么其权值是 0。如果这两种状态是相容的，那么权值与在上下文标签 $s_{j_i}^{i-1}$［1…$k-1$］和观察到的词元序列 x_{i}^{i} 中预测标签 $s_{j_i}^{i-1}$［k］的概率成正比。

- 对于每个词元 x_i，我们递归地计算[⊖]

$$\alpha_0(s_j)=0, j=1,\cdots,k$$

$$\alpha_i(s_j)=\max_{j_i=1,\cdots,M}\alpha_{i-1}(s_{j_i})+\log w(s_{j_i}^{i-1},s_j^i)$$

$$\gamma_i(s_j)=\arg\max_{j_i=1,\cdots,M}\alpha_{i-1}(s_{j_i})+\log w(s_{j_i}^{i-1},s_j^i)$$

直观地理解，$\alpha_i(s_j)$ 代表在格中经过 i 步后，以 s_j 状态结束的最可能路径的对数概率，$\gamma_i(s_j)$ 代表在这条特定的路径上 s_j 之前的状态[⊜]。

- 在计算 (α_i) 的值时，寻找最优路径的算法与公式（8.1）的解相对应，即

1）确定 $\hat{s}_L^L=\text{argmax}_{j=1\cdots M}\alpha_L(s_j)$。

2）对于 $i=L-1\cdots1$，计算 $\hat{s}_i=\gamma_{i+1}(\hat{s}_{i+1}^{i+1})$。

3）给出公式（8.1）的解：

$$\hat{y}=(\hat{s}_1[k],\hat{s}_2[k],\cdots,\hat{s}_L^L[k])$$

完整的算法参见算法 8-1。该算法的时间复杂度为 $\Theta(|y|^k\cdot L)$，对于句子长度 L 为线性复杂度，但对马尔可夫依赖的长度 k 为指数级复杂度。为了减少搜索空间，我们使用柱搜索（beam search）。

[290]

算法 8-1　维特比（Viterbi）搜索

```
输入：词元 x_1^L
输出：最可能的标注序列（即模型）ŷ^L = argmax_{y^L} P(y_1^L|x_1^L)
创建 S = {s_1, ..., s_M}，这是 Y^k 的一个改革
for j = 1, M do α_j ← 0
for i = 1 − k, L + k do
    for j = 1, M do
        γ_i = 1, b_j = −∞
        for j' = 1, M such that s_{j'}[2..k] = s_j[1..k − 1] do
            a ← α_{j'} − log w(s_{j'}^{i−1}, s_j^i)
            if (a > b_j) then
                b_j ← a, γ_i ← j'
    α ← b
ŝ_{L+k} ← arg max_{j=1...M} α_j
j ← arg max_j γ_{i+1,j}
for i = L + k − 1 ... 1 do ŝ_i ← s_{i,j} ← γ_{i+1,j}
ŷ^L ← (ŝ_1[1], ŝ_2[1], ..., ŝ_L[1])
```

柱搜索

任何人实现算法 8-1 都面临着一个现实的挑战：即使 k 值很小，空间 y^k 也可能十分大，尤其是当分类空间很大的时候。这一问题的出现是由于算法的搜索空间与 $|y|^k$ 成正比。实际上，对于很多自然语言处理任务（NLP）而言，这也是柱搜索算法被认为更好的

⊖ 方便起见，和状态 s_j 相关联的下标 i 记做 a_i，并省略 i 函数 $\alpha_i(s_j)$ 为 $\alpha(s_j)$。

⊜ 因为数值稳定的原因，w 用对数函数来计算。因为如果使用本身数值计算那很长的句子就会导致结果过小而不精确。另一种方法，我们可以用 w 的系数进行规一化，即在每次计算时相乘中间的所有系数和来进行归一化。

原因。这一算法是基于以下理念而被构建出来的：在架中的许多节点有非常小的 α 值，而这些值将不会被包含在任何“较优”的路径中，因此这些值可以在计算中被跳过而几乎不会影响最终效果。为了达到这一目的，在架中的第 i 步时，算法仅保留很少的 $M=|y|^k$ 个状态。然后，在计算出第 $i+1$ 步时扩展的节点后，即可基于它们的 α_i 值过滤掉部分状态。我们可以采用多种过滤技术，而这其中最常用的两种是：

- 固定柱宽：在每步中仅保留前 n 个高分候选作为扩展。
- 可变柱宽：在第 i 步中仅保留和最高得分候选项相对距离为某一特定值（按照 α_i）之内的候选项。

这两种方法都是很好的选择。经验表明柱宽为 5 和相对距离为 30%的可变柱宽可显著加快计算速度（20～30 倍），并且几乎不会降低其性能。在特定任务中，应该使用相应的开发集来优化这些参数值，当然这也取决于研究人员如何在速度和准确性中进行取舍。

8.2.3 提及检测特征

如前文所述的最大熵框架，任意一种特征都可以被使用。这使得系统设计人员可以对感兴趣的特征类型进行试验，而不用担心特征间的相互作用。比较起来，在基于规则的系统中，系统设计人员不得不考虑在一个特殊的示例中字典信息与词性信息、组块信息的互相影响。这并非是说最终基于规则的系统在某些方面比统计模型差。基于规则的系统是基于宝贵的洞察构建的，如果我们把自己限制为仅采用统计方式进行建模，那么这些见解是很难得到的。事实上，基于规则的系统的输出很容易被整合为 MaxEnt 框架的输入特征之一，这也使得该框架在通常情况下能够获得比其他任何类型的系统都更好的性能。 291

在一个典型的提及检测系统中，使用的特征通常可以分为 5 大类：词汇特征、句法特征、从其他命名实体分类器获得的信息（具有不同的语义标注集合）、基于地名词典的特征和跨语言的提及传播所获得的特征。我们还使用了前文提到的分类标签作为附加特征。

1. 词汇化特征

当前词元（段）x_i 本身及其上下文很显然是判断 x_i 是否是一个提及的最重要的特征之一 [3，5]。词汇特征用跨越当前词元的 n 元组来实现，包括其前驱和后继部分。对于一个词元 x_i，n 元组特征包括前 $n-1$ 个词元（x_{i-n+1}，…，x_{i-1}）和后 $n-1$ 个词元（x_{i+1}，…，x_{i+n-1}），n 一般取 3 就会有很好的效果。

在一个形态学丰富的语言中，如阿拉伯语，我们应该考虑所实现的特征：是词干的 n 元组，同样包括前驱和后继部分 [4]。如果当前词元 x_i 是一个词干，词干的 n 元特征应包含前驱的 $n-1$ 个词干和后继的 $n-1$ 个词干。词干 n 元组特征由附着的前缀和后缀的单词的基本形式（即词干）构成，其表示了一个词汇的一般形式，这样便减少了数据稀疏。在我们的实验中，n 被设置为 3（词干的三元特征）。

2. 句法特征

句法特征包括词性标记和浅层句法分析信息。这使得在提及检测时引入了另一个层级的抽象性和普遍性。我们发现在当前词元的小窗口使用 POS 和浅层句法分析信息是很有效的。例如，在每个词元的大小为 5 的窗口中（当前词元，前驱的两个词元和后继的两个词元），我们可以在词性和分块信息的基础上计算特征[1]。词性信息有助于对某些词元消歧。浅层句法分析信息或文本块，有助于定义提及的边界。例如，如果两个相邻的词元

[1] 块是一个对应句法短语的较短的单词序列，消歧它不再存在任何子句法短语。

x_i、x_{i+1}属于两个不同的词组（例如，分别是名词短语和动词片语），并且 x_i 是提及 m_j 的一部分，那么 x_{i+1}不可能也是同一个提及 m_j 的一部分。

3. 来自其他命名实体分类器的特征

292 除了用丰富的词汇和句法特征之外，利用不同的提及标注器也是非常有用的。这些标注器在不同于“初级”标注器的数据集上训练。此外，这些标注器识别的提及类型和我们关注的可能并不相同。假设我们关注的是 ACE 任务。某一个标注器可以识别出很多类别，包括日期或职业（这并不是 ACE 的一部分）。它可能也会识别出人物类别，但根据各自的标注标准不同，可能不匹配我们标注任务中的人物概念。有一种假设——**组合假设**（combination hypothesis）——是将提及和不同来源的命名实体分类器相结合，将互补的信息加入到提及检测模型当中，从而提高性能。事实上，由 Borthwick 等的研究结论得出[24]，来自不同标注器的输出在作为提及检测模型的额外特征时是非常有效的。这种方法可以让系统将不同的提及类型自动关联到理想的输出，而不需要人工映射。

4. 基于地名词典的特征

一个**地名词典**（gazetteer）是一种特殊类型[㊀]的词典，它包含一些特定类型的词项。用于提及检测系统的地名词典通常包括人名、国家名和公司名。名称地名词典通常包含单词元的名，如 Daniel 或 Galsa，也有短语的名称，如 Ben Ali、Barak Obama 或 United States。这种提及检测系统通过一个简单的特征函数来进行检测，该函数返回一个特定词元是否在地名词典中。更正式地说，当处理词元 x_i 时，我们检查词元本身 x_i 或它周围的词元（x_{i-a}，…，x_{i+a}）是否属于一个地名。

5. 跨语言提及的传播特征

很少有英语以外的语言有规模较大并且质量较高的可用数据资源。大多数语言只有较少的数据资源可用。我们可以通过使用有大规模资源的语言来提高另一种语言的提及检测系统，从而减轻资源匮乏带来的差异性。该方法需要有建立在丰富资源语言上的检测系统，并且有从源语言到资源丰富语言同时含有词对齐的翻译。首先，我们使用的统计机器翻译（SMT）系统将源语言的单元（文档或句子）x_j^N 翻译为资源丰富的语言，以生成目标序列 $\hat{e}_j^M=(\hat{e}_1,\hat{e}_2,\cdots,\hat{e}_M)$。以词元序列 $\hat{e}_j^M$ 作为输入，一个建立在资源丰富语言上的提及检测系统为每个词元赋予提及标注，构造出标注序列 $\hat{\phi}_j^M=(\phi_1,\phi_2,\cdots,\phi_M)$。使用 SMT 产生的源文本 x_j^N 和翻译文本 $\hat{e}_j^M$ 间的词对齐[25]，我们将目标语言的标注 $\hat{\phi}_j^M$ 传播到源语言，建立标注序列 $y_j^N=(y_1,y_2,\cdots,y_N)$[㊁]。举个例子，如果一个资源丰富语言的词元序列 $\hat{e}_i\hat{e}_{i+1}\hat{e}_{i+2}$ 与源语言中的 x_jx_{j+1} 对齐，并且 $\hat{e}_i\hat{e}_{i+1}\hat{e}_{i+2}$ 被标注为一个地点提及，那么序列 x_jx_{j+1} 可以被标注为地点提及：B-LOC、I-LOC。因此，每个在 x_j^N 中的词元 x_i 的标注与传播源的 y^N 中 y_i 的标注相对应，这里 $y_i=\phi(i,A,\hat{\phi}_j^M)$，其中 A 是源语言和资源丰富语言之间的对齐信息。在我们使用 SMT 词对齐将目标语言 $\hat{e}_j^M$ 的标注序列 $\hat{\phi}_j^M$ 传播到相应的目标语言文本 x_j^N 上时，我们将得到一个标注序列 y_j^N，使得每一个在 x_j^N 中的 x_i

293 都赋予一个 y_j^N 中的 y_i。这个标注序列可以被用作 MaxEnt 框架中的一个额外的特征，也可以用来构建基于地名词典的特征。对于提及检测系统中跨语言提及传播方法的更多详细信息，请参阅 Zitouni、Florian [26] 和 Benajiba、Ztouni [27]。

㊀ 从技术上讲，地名词典特指一个地名列表，但是在 NLP 领域中其应用可能会更加广泛。

㊁ 如果你使用的 SMT 系统没有提供词对齐功能，也可以使用 Giza++。

8.2.4 提及检测实验

实验使用 ACE 2007 数据集[①]，包括 4 种语言：阿拉伯语、中文、英语和西班牙语。这些数据是从各种领域（广播新闻、广播谈话、新闻专线、博客、电话访谈）中挑选出来的。标注类型被分为 7 类：人、组织机构、地点、设施、GPE（Geopolitical Entity，地理政治实体）、交通工具、武器。除了提及信息，提及、关系、事件、时间之间的共指也会被标注出来。

因为用于评测的测试集答案是不公开的，我们将公开的训练语料按 85%和 15%的比例分开。为了便于将未来和现在的工作进行比较并模拟出一个实际的方案，划分原则基于文章的日期：测试数据是从训练集中按照年代顺序在每一个领域中挑出最后 15%的数据。这样测试集和训练集中的文档在时间上便不会有重叠了，而且测试集的数据相比训练集数据在时间上是更新的。表 8-1 列举出了每种语言中训练数据和测试数据集的文档数。

表 8-1 数据集的文档数

语 言	训 练 集	测 试 集
阿拉伯语	323	56
中文	538	95
英语	499	100
西班牙语	467	52

使用 ACE 数据的性能通常用一个特定的指标来评估，即 ACE 值[1]，因为我们只对提及检测任务感兴趣，所以我们使用更直观和更流行的 F 值（没有加权的）来进行估计，即召回率与精确率的调和平均数。

表 8-2 呈现了提及检测试验系统在使用了所有可用的语言学知识后对于 4 种语言的结果，包括通过词汇（3 个词窗口的词和形元，长度不大于 4 的前、后缀，英语还利用了 WordNet [28]）、句法（词性标记、文本块），和其他信息提取模型的输出。 [294]

表 8-2 阿拉伯语、中文、英语、西班牙语提及检测系统的结果。结果用精确率（P）、召回率（R）、F 值（F）来度量，列数 N 表示测试集中提及的数量

语 言	N	P	R	F	语 言	N	P	R	F
阿拉伯语	5455	85.5	76.8	80.9	英语	8170	86.6	80.8	83.7
中文	4761	82.1	71.3	75.8	西班牙语	2487	79.1	73.6	76.2

结果表明英语的提及检测系统和阿拉伯语、中文和西班牙语这些其他语言的系统相比，有更好的性能。这些结果基本是可以预想到的，因为英语的模型训练有规模更大的数据集，并且可以使用更丰富的信息比如 WordNet [28] 和建立在更大数据集上的信息抽取系统的输出。

另一个实验是利用高性能的英语提及检测系统，研究跨语言提及传播特征的影响，以进一步改进其他语言的系统，我们特别考虑改进阿拉伯语、汉语和西班牙语的系统。为了这个实验，我们使用了 3 个 BLEU [29][②] 分数非常有竞争力的 SMT 系统。阿拉伯语到英语的 SMT 系统与 Huang、Papineni [30] 的描述类似，在 NIST（National Institute of Standards and Technology）2003 的阿拉伯语到英语的翻译评测中，得到 0.55 的 BLEU 分数。汉语到英语的 SMT 系统与 Al-Onaizan、Papineni [31] 的架构相似。这个系统在 NIST 2003 汉语[③]到英语机器翻译评测中，获得了 0.32 的 BLEU 分数。西班牙语到英语

① 和 ACE 2005 的数据相同。

② BLEU 是一个使用多参考译文对翻译结果进行自动评测的指标。

③ 此处应该是原书错误，原书为阿拉伯语到英语。——译者注

的 SMT 系统与 Lee 等人 [32] 描述的系统相似，在 TC-STAR 2006 的最终版本的欧洲议会全会语言语料库评测中获得了 0.55 的 BLEU 分数。表 8.3 表明了当借助于英语提及检测系统抽取的特征时，阿拉伯语、中文和西班牙语的提及检测系统的性能都有所提升。提及检测系统的性能在阿拉伯语上增加了 0.9F（80.9 与 80.0），中文上增加了 2.3F（78.1F 与 75.8F），西班牙语上增加了 1.9F（78.1F 与 76.2F）。结果表明利用跨语言提及传播信息在提升性能方面是有效的。Zitouni、Florian [26] 认为随着语言的可用资源越来越多，性能的提升却有减缓的趋势，这个结果有助于回答一个很重要的问题：在一个资源贫乏的语言中，当我们想要提升提及检测系统的性能时，我们应该构建资源还是利用另一个资源丰富的语言的标注传播呢？答案似乎是后者。

295

表 8.3　完整的提及检测系统在阿拉伯语、中文、西班牙语上的结果，使用词汇、句法、信息提取模型的输出、跨语言传播特征（成熟的系统）。结果用精确率（P）、召回率（R）、F 值（F）来度量，列数（N）表示测试集中提及的数目

跨语言提及传播				
语言	N	P	R	F
阿拉伯语	3566	84.2	77.6	80.9
中文	4791	81.7	74.8	78.1
西班牙语	3487	80.1	76.2	78.1

8.3　共指消解

在某些自然语言应用中仅获得文档中一些独立的提及是不够的。例如，在基于下述段落的情况下，对于“When was John F. Kennedy assassinated?”（约翰 F. 肯尼迪何时遇刺身亡的？）这个问题，我们该如何回答：

John F. Kennedy was the thirty-fifth President of the United States. He was later assassinated on Friday, November 22, 1963.

（约翰 F. 肯尼迪是美国第三十五届总统，后来他在 1963 年 11 月 22 日星期五被暗杀。）

John F. Kennedy 被一个代词提及 He（他）所指，于是答案在后一句话中可以找到。因此为了正确回答这个问题，知道 He（他）指的是 John F. Kennedy 是至关重要的。

将那些指向同一个物理对象的提及链接到一个实体的过程叫做共指消解。共指消解和指代消解非常相近，指代消解指的是找到代词的正确先行词。我们使用共指消解是因为本节所讨论的问题范围较指代消解更为广泛，包括了解决所有类型名词短语的指代关系。

虽然基于规则方法 [33，34，35，36] 的共指消解已有很多研究，但本文关注的是基于机器学习的方法。我们知道有大量的研究工作是有关建立可学习的共指消解系统的 [37，38，39，40，41，42，43]。早期的系统 [37，38] 通过训练数据学习一个模型，对于一对提及指向同一实体的可能性赋予一个分数，然后根据提及对的分数对指向的实体进行聚类。这类系统的一个技术难点是传递性。例如，如果提及 A 与提及 B[a] 链接，提及 B 与提及 C 链接，但提及 A 没有与提及 C 链接，那么系统没有处理这种传递性的能力。在实际情况中，基于提及对的系统通常将提及链接到第一个高于指定阈值的候选先行词或候选集中最好的先行词来解决这个问题。为了克服这个缺点，一些研究者 [40，44，42] 采

296

[a] 这表明，分数高于预设的阈值。

用实体提及模型，即计算一个候选实体和当前提及的分数。Ng [45] 提供了在过去 10～15 年中有关基于机器学习的共指消解的研究进展。Ng 的研究工作还介绍了各种有用的资源，如已标注的共指语料库和有关的公开评测任务。本章的其余部分，我们重点关注一个基于 Bell 树算法的共指消解系统 [40]。我们的目标是，使各位想动手实践的读者在阅读完本章之后可以实现这个算法。

8.3.1 Bell 树的构建

Bell 树是一种数据结构，它表示在一篇文档中提及可能指代的实体的假设空间。众所周知，将 n 个对象划分成许多不相交的非空子集的方法数目就是 Bell 数 [46] $B(n)$。Bell 数有一个近似的公式：$B(n) \sim \frac{1}{e}\sum_{k=0}^{\infty}\frac{k^n}{k!}$，它随着 n 的增长急剧增加。比如 $B(20)\approx 5.2\times 10^{13}$，它已经是一个天文数字了！毫无疑问不可能完穷尽搜索整个空间，所以一个有效的搜索策略是必要的。

在我们解决搜索问题之前，首先描述一下通过文档中的提及和共指模型构建 Bell 树的过程。我们假设文档中若实体增量地由提及产生，并且同步地构建出 Bell 树。第一个提及用来生成树的根节点，后续的每个提及或者是开始指代一个新实体（添加），或者链接到已有实体。在这个过程结束的时候，每个叶节点代表一个可能的共指结果。这个过程称为提及同步：当加入一个新的提及，就会在树中每层创建节点。因为树叶的数量就是可能共指结果的数量，即等同于 Bell 数 [46]。这棵树就被称作 Bell 树。

图 8-1 说明了 Bell 树是如何依据下述三个提及创建的：

President Ben Ali said that his minister, Mohammed Ghannouchi, will present the case.

初始节点由第一个不完整实体 [*President*] 组成（即在图 8-1 中的节点 a）。接下来，提及 *Ben Ali*（参见图 8-1 顶部的线）活跃，可以链接不完整实体 "*President*"，并生成新的节点 b1，也可以引入一个新的实体并创建另一个节点 b2。活跃的提及可能链接的不完整实体称作受关注的实体（in-focus）。类似地，提及 *his* 在下一个阶段活跃，可以有 5 种可能的操作，也会产生 5 种可能的共指结果如图所示节点 c1～c5。

图 8-1 描绘的推导过程中，每个 Bell 树的叶节点对应一个可能的共指结果，不存在其他可能的实体结果。因此，Bell 树完全表示了共指消解问题的搜索空间。共指消解最终等同于找到最好的叶节点。由于搜索空间巨大，所以即使文档中只有中等数量的提及，也很难直接估计 Bell 树中树叶分布的情况。但是，我们可以考虑提及到实体的增量处理过程。观察实体创建的动态视图，共指消解问题自然地变成了对 Bell 树中竞争路径的评分。 297

Bell 树表示法有一个很好的特性，每次链接或引入的次数对于所有可能的假设共指结果都是相同的。这使得它对“局部”链接或引入概率的排序变得很容易，因为数量是相同的。

Bell 树表示也是增量式变化的，因为提及是逐渐添加的。这使得设计一个解码器和搜索算法很容易。共指模型在 8.3.2 节论述，但是我们暂时假设，有模型可以为链接和引入分支进行评分。那么解码一个文档就只是构建其中提及的 Bell 树，就像前文提到的，然后给树中的路径打分。在处理 n 个提及之后，我们得到一个深度为 $(n-1)$ 的树。然后在用下一个提及扩展树之前，我们先对累计分数低于某指定阈值（和最高的分数相比）的分支进行剪枝；换句话说，我们在剪枝的 Bell 树上做广度优先搜索。

8.3.2 共指模型：链接和引入模型

我们用一个二元条件概率模型来计算激活的提及链接到受关注不完整实体的概率。这

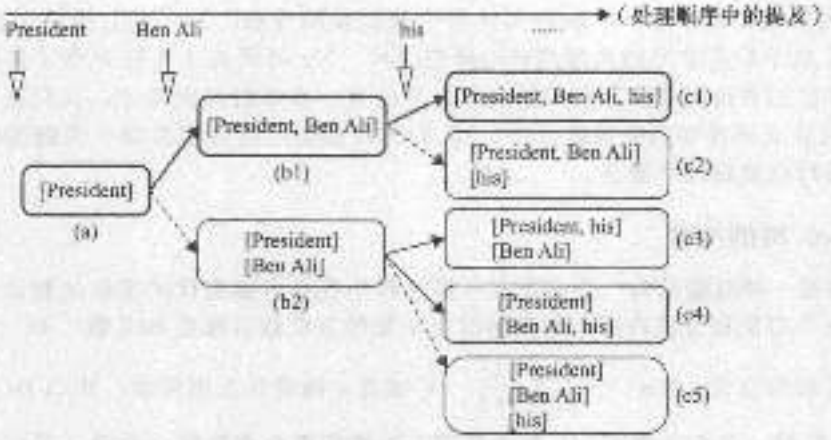


图 8-1　Bell 树代表实体从提及中形成的过程。提及在 [ ] 中是指一个部分实体。实线箭头意味着一个提及链接到一个受关注的实体，虚线箭头表示引入一个新的实体。这里的提及是按文中出现的顺序来处理的[1]

些条件包括所有已生成的不完整实体，受关注实体的下标和活跃提及[2]。

形式化来看，使 $\{m_i: 1\leqslant i\leqslant n\}$ 成为文档中的 n 个提及。提及的下标 i 代表它们在文档中被处理的顺序（文档顺序是无关紧要的）。e_j 作为一个实体，$g: i\to j$ 是从提及 i 到实体 j 的映射（多对一）。对于一个下标为 k 的活跃提及（$1\leqslant k\leqslant n$），先定义：

$$I_k=\{t: t=g(i),\quad 对于\quad 1\leqslant i\leqslant k-1\} \tag{8.4}$$

公式（8.4）为对应 m_k 的已建立实体的下标集合（注意 $I_1=\varphi$），并且

$$E_k=\{e_t: t\in I_k\}$$

上式为已建立的实体集合。链接的概率为

$$P(L\mid E_k, m_k, A_k=t) \tag{8.5}$$

这是活跃提及 m_k 与受关注实体 e_t 的链接概率。随机变量 A_k 从集合 I_k 中取值并指示哪一个实体是受关注的。L 是二值的，如果 m_k 与 e_t 链接，则取值为 1，否则为 0。

例如在图 8-1 中，从 b2 到 c4 的分支，活跃提及是 *his*，在处理 *his* 时已经建立的部分实体是 $E_3=\{[President], [Ben\ Ali]\}$，关注的实体是 $[Ben\ Ali]$。概率 $P(L=1\mid E_3, his, A_3=2)$ 来度量 *his* 链接实体 $[Ben\ Ali]$ 的可能性。

$P(L\mid E_k, m_k, A_k=t)$ 只度量 m_k 与 e_t 链接的可能性。但是它并未说明 m_k 引入一个新实体的概率。幸运的是，引入的概率可以通过公式（8.5）的链接概率计算得到，如下所示。

因为引入一个新实体意味着 m_k 与其他任何实体不存在链接 E_k 的关系。引入一个新实体的概率 $P(L=0\mid E_k, m_k)$，可以通过以下公式来计算

$$P(L=0\mid E_k, m_k) \tag{8.6}$$

$$=\sum_{t\in I_k}P(L=0, A_k=t\mid E_k, m_k)$$

$$=1-\sum_{t\in I_k}P(A_k=t\mid E_k, m_k)P(L=1\mid E_k, m_k, A_k=t) \tag{8.7}$$

公式（8.7）表明引入新实体的概率可以通过连接概率 $P(L=0\mid E_k, m_k, A_k=t)$ 来计

[298]

[1] 原书图 8-1 中 c2 箭头已改为虚线箭头。——译者注

[2] 文档本身一直都是上下文感条件的一部分，这里省略以简化讲解。

算，其中边际概率 $P(A_k=t \mid E_k, m_k)$ 是已知的。模型中的 $P(A_k=t \mid E_k, m_k)$ 也可以解释为候选实体选择模型。对于当前提及 m_k 和已创建的实体集合 E_k，$P(A_k=t \mid E_k, m_k)$ 则是实体 e_t 成为候选实体的概率。

为 $P(A_k=t \mid E_k, m_k)$ 直接训练模型是比较困难的，因为 I_t 随着提及持续的处理是不断增长的。所以我们用以下公式近似估计 $P(A_k=t \mid E_k, m_k)$。

$$P(A_k=t \mid E_k, m_k) \approx \begin{cases} 1 & \text{若 } t=\arg\max_{t \in I_k} P(L=1 \mid E_k, m_k, A_k=t) \\ 0 & \text{否则} \end{cases} \tag{8.8}$$

公式（8.8）并不是对 $P(A_k=t \mid E_k, m_k)$ 近似估计的唯一方法。例如，我们可以用一个 I_k 的均匀分布。我们试验了几种近似的方案，包括均匀分布，但公式（8.8）的计算效果最好，所以我们采用公式（8.8）的计算方法。我们也可以直接训练出 $P(A_k=t \mid E_k, m_k)$，然后用它为 Bell 树中的路径评分。问题在于 I_k 大小是可变的，而 A_k 从中取值，并且引入操作取决于 E_k 中所有的实体，这使得直接训练 $P(A_k=t \mid E_k, m_k)$ 比较困难。 299

用近似公式（8.8），公式（8.7）中的引入概率则变为：

$$P(L=0 \mid E_k, m_k) = 1 - \max_{t \in I_k} P(L=1 \mid E_k, m_k, A_k=t) \tag{8.9}$$

链接概率（公式（8.5））和近似引入概率（公式（8.9））用来为 Bell 树中的路径打分，比如，图 8.1 中的路径 a—b2—c4 就是从 a 到 b2 的引入概率和从 b2 到 c4 链接概率的乘积。

由于公式（8.9）只是一个近似计算，所以我们也可以引入一个常量 α 去平衡链接概率和引入概率，那么真实的引入分数就成为：

$$P_\alpha(L=0 \mid E_k, m_k) = \alpha P(L=0 \mid E_k, m_k) \tag{8.10}$$

如果 $\alpha<1$，则意味着添加新的实体会受到惩罚。因此，α 又叫引入惩罚（start penalty），α 一般使用开发集调整，并且用来平衡实体漏报和误报。

模型 $P(L \mid E_k, m_k, A_k=t)$ 依赖于所有未完成实体 E_k，这也有着非常高昂的复杂度。链接一个提及 m_k 和受关注实体 e_t 时，假定与其他实体无关也是合理的。

$$P(L=1 \mid E_k, m_k, A_k=t) \tag{8.11}$$

$$\approx P(L=1 \mid e_t, m_k) \tag{8.12}$$

$$\approx \max_{m \in e_t} P(L=1 \mid e_t, m, m_k) \tag{8.13}$$

从公式（8.11）和公式（8.12）可以看出，除了受关注实体 e_t 外的其他所有实体都可以假定为对 m_k 和 e_t 的链接没有影响。公式（8.13）进一步假设实体-提及的分数可以通过最大提及对的分数获得。公式（8.13）中的模型和 Morton [47]、Soon、Ng 与 Lim [37] 以及 Ng and Cardie [38] 所使用的模型比较相似，但公式（8.13）可以包含实体级的特征，因为 e_t 也是条件的一部分。

8.3.3 最大熵链接模型

最大熵模型采用公式（8.23）所述的模型

$$P(L \mid e_t, m, m_k) = \frac{1}{Z(e_t, m, m_k)} \exp\left[\sum_i \lambda_i g_i(m, m_k, L) + \sum_j \lambda_j h_j(e_t, m_k, L)\right] \tag{8.14}$$

其中 $g_i(m, m_k, L)$ 是提及对的特征，$h_j(e_t, m_k, L)$ 是实体级的特征，因为它是由实体 e_t 和提及 m_k 计算得来的。实体级的特征可以包含地捕捉到实体 e_t 和提及 m_k 之间的性和数的一致性；提及对特征，从另一种意义上讲，对于编码词汇特征是有用的，如字符串 m 303

和m_2是否字面上匹配。正如 Berger、Della Pietra、Della Pietra [17] 所描述的，一旦最大熵模型特征选定后，最佳的特征权重 $\{\lambda_i\}$ 和 $\{\lambda_i\}$ 可以高效地找到。

因为实体和提及之间的关系可以通过它们的特征来描绘，好的特征集对于系统的性能是十分必要的。共指模型所使用的特征被归纳为几组，绝大部分特征在不同语言上是通用且可移植的，然而某些特征比如词干匹配特征，是用来描述阿拉伯语形态学相似度的：

1）词汇特征只针对于非代词提及。它们包括两个提及的完整或部分的字面匹配、首字母缩略词、根据提及拼写的实际配对。

2）属性特征可以直接从训练数据中计算得到。比如 ACE 的训练数据包括了实体类型，实体子类型，以及那些可以用来描绘同一个实体中提及的提及类型信息。因为代词是一个紧密联系的范畴，我们提取一个代词的性、数、所有格和反身性并传播给它们所属的实体。

3）编辑距离用来计算两个字符串之间的距离（即两个提及的拼写），并量化这个距离。这也是另一种表征提及之间相似度的方法。

4）距离特征表示两个提及之间间隔多远，或者相隔词元、句子或提及的个数。

5）词干匹配特征比较两个提及之间的词干，这些特征是特别为阿拉伯语设计的。

6）一致性特征是在一个提及和实体对上计算的。它们用来检测提及和实体间性和数的一致性。需要注意的是，这个特征集合和代词的性和数的属性是不同的。

7）句法特征来自 [48] 在阿拉伯语的宾州树库 [49] 上训练的最大熵句法分析器自动生成的句法分析树。提及的词性标签也从分析树中抽取而来。我们还可通过检查它们在分析树中的结构关系来检测在同一个句子中的两个提及是否为同位语。因为约束理论 [50] 很好地说明了代词消解，所以一组特征用来计算代词提及在句子范围内的候选先行词是否在管辖语域内。Luo、Zitouni [44] 详细地介绍了这些特征。

除了这些最基本的特征，复合特征可由基本特征结合生成。比如，具有反身性的距离特征的代词提及可以帮助捕获到反身代词的先行词比非反身代词的先行词更接近。 [301]

8.3.4　共指消解实验

我们汇报了在最近的 ACE 数据上 [51] 的实验结果[⊖]。数据集包括了 599 个丰富多样化的文档，其中包括新闻专线的文章、博客、世界性新闻组网络的布告、广播新闻的文字稿、广播新闻的对话、电话交谈。我们保留了每个文档中最新的 16% 的内容作为测试集，其余的作为训练集。如此划分后的文档数、字数、提及数和实体数的统计信息制成表格，如表 8-4 所示。

表 8-4　ACE 数据的统计信息：训练集和测试集中的文档数、词数、提及数和实体数

数 据 集	文 档 数	词 数	提 及 数	实 体 数
训练集	499	253 771	46 646	16 162
测试集	100	45 659	8 178	3 709
所有数据	599	299 430	54 824	18 811

2008 年的 ACE 值是在评测任务中的官方评分，这里我们用来描述我们共指系统的性能。它的具体定义可以通过官方评测文档[⊜]找到。因为 ACE 值是一个有权重的评测指标，

⊖ 在 ACE 2007 和 2008 的评测中，没有新的训练数据发布，也就是说，2005、2007、2008 使用同样的训练数据。

⊜ 官方评测文档可在 http://www.itl.nist.gov/iad/mig/tests/ace/2008/doc/ace08-evalplan.v1.2d.pdf。

用来衡量共指系统的相对值，又因为在 ACE 程序中权重和它的公式每年都会变化，所以当比较两个系统时知道使用的是哪一年的评分程序是非常重要的。更值得一提的是，ACE 值对一些特定类型的错误是不敏感的，因为它们的权重非常低（比如代词错误只是命名错误的 1/10）。

表 8-5 包含了上述提及的测试集的结果，它使用最近 ACE08 的评分程序：http://www.itl.nist.gov/iad/mig//tests/ace/2008/software/ace08-eval-v17.pl。第二列是基于 ACE08 值的 F 测试，最后一列是官方的 ACE 值。第二行包含了作为共指系统输入的标准提及的结果，最后一行对应于使用提及检测系统的结果。我们可以看出，给定标准提及 [302] 时，共指系统可以得到非常高的 ACE 值。然而提及检测系统过多的噪声会大大降低系统性能，使 ACE 值从 79.8%降低到 60.3%。

表 8-5 使用 ACE08 评分的共指消解结果：第二列是 B3-F 值，ACE08 值是官方指标

提 及	B3-F 值	ACE08 值
标准	85.1	79.8
提及检测系统	80.2	60.3

8.4 总结

这一章，我们讨论了信息抽取中两个十分重要的任务：提及检测和共指消解。用一个例子详细介绍了级联系统的实现，这个系统包含了一个基于最大熵值模型的提及检测组件，并串联了一个基于 Bell 树算法的共指消解系统。提及检测组件在检测提及时把它看作一个序列标注问题，使用从训练数据中自动抽取的词汇、句法、语义特征。基于 Bell 树的共指消解系统通过在 Bell 树中找寻从根节点开始到叶节点的路径来寻找文档中最好的共指结果，Bell 树则描述共假设空间。一个实体-提及的二元模型用来对路径中的每一个分支评分。这样一个统计系统的好处就是它是数据驱动的并且只要获得语言特定的信息作为特征就可以很快地应用到其他语言。

我们指出提及检测和共指消解可以用一个联合方式来解决，这是因为在某些提及判断中需要共指消解来指导（相反也成立）。尽管在实践上，联合系统的复杂度经常掩盖了它的优势，但是联合系统仍旧值得未来更深入的研究。提及检测和共指消解的输出为未来深入的分析奠定了基础，比如说第 9 章中讨论的关系和事件抽取。它也可以直接被用于一些应用中，比如问答（第 13 章中讨论）或者机器翻译系统（第 10 章中讨论）。

参考文献

[1] NIST (National Institute of Standards and Technology). "The ACE evaluation plan." 2007. www.nist.gov/speech/tests/ace/index.htm.

[2] H. Daumé III and D. Marcu, "A large-scale exploration of effective global features for a joint entity detection and tracking model," in *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 97–104, 2005.

[3] R. Florian, H. Hassan, A. Ittycheriah, H. Jing, N. Kambhatla, X. Luo, N. Nicolov, and S. Roukos, "A statistical model for multilingual entity detection and tracking," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2004)*, pp. 1–8, 2004.

[303]

[4] I. Zitouni, J. Sorensen, X. Luo, and R. Florian, "The impact of morphological stemming on Arabic mention detection and coreference resolution," in *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, pp. 63–70, 2005.

[5] I. Zitouni, X. Luo, and R. Florian, "A cascaded approach to mention detection and chaining in Arabic," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 17, pp. 935–944, 2009.

[6] Y. Benajiba, M. Diab, and P. Rosso, "Arabic named entity recognition: A feature-driven study," *In the special issue on Processing Morphologically Rich Languages of the IEEE Transaction on Audio, Speech and Language*, 2009.

[7] R. Florian, Abe Ittycheriah, H. Jing, and T. Zhang, "Named entity recognition through classifier combination," in *Conference on Computational Natural Language Learning*, 2003.

[8] H.-L. Chieu and H. Ng., "Named entity recognition with a maximum entropy approach," in *Conference on Computational Natural Language Learning*, 2003.

[9] D. Klein, Joseph Smarr, H. Nguyen, and C. Manning, "Named entity recognition with character-level models," in *Conference on Computational Natural Language Learning*, 2003.

[10] Q. T. Tran, T. T. Pham, Q. Hung-Ngo, D. Dinh, and N. Collier, "Named entity recognition in Vietnamese documents," *Progress in Informatics Journal*, no. 4, 2007.

[11] Y. Benajiba and P. Rosso, "Arabic named entity recognition using conditional random fields," in *Workshop on HLT and NLP within the Arabic world. Arabic Language and local languages processing: Status Updates and Prospects, 6th International Conference on Language Resources and Evaluation* (LREC), 2008.

[12] R. Florian, H. Jing, N. Kambhatla, and I. Zitouni, "Factorizing complex models: A case study in mention detection," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 473–480, 2006.

[13] L. Ramshaw and M. Marcus, "Exploring the statistical derivation of transformational rule sequences for part-of-speech tagging," in *The Balancing Act: Proceedings of the ACL Workshop on Combining Symbolic and Statistical Approaches to Language*, pp. 128–135, 1994.

[14] L. Ramshaw and M. Marcus, "Text chunking using transformation-based learning," in *Proceedings of the Third Workshop on Very Large Corpora*, pp. 82–94, 1995.

[15] E. F. Tjong Kim Sang, "Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition," in *Proceedings the Conference on Natural Language Learning*, pp. 155–158, 2002.

[16] J. N. Darroch and D. Ratcliff, "Generalized iterative scaling for log-linear models," *The Annals of Mathematical Statistics*, vol. 43, no. 5, pp. 1470–1480, 1972.

[17] A. Berger, S. Della Pietra, and V. Della Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[18] J. Goodman, "Sequential conditional generalized iterative scaling," in *Proceedings of the Association for Computational Linguistics*, 2002.

[19] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming*, vol. 45, no. 3 (Ser. B), pp. 503–528, 1989.

[20] S. Chen and R. Rosenfeld, "A survey of smoothing techniques for me models," *IEEE Transactions on Speech and Audio Processing*, 2000.

[21] J. Goodman, "Exponential priors for maximum entropy models," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 305–312, 2004.

[22] S. Khudanpur, "A method of maximum entropy estimation with relaxed constraints," in *1995 Johns Hopkins University Language Modeling Workshop*, 1995.

[23] J. Kazama and J. Tsujii, "Evaluation and extension of maximum entropy models with inequality constraints," in *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pp. 137–144, 2003.

[24] A. Borthwick, J. Sterling, E. Agichtein, and R. Grishman, "Exploiting diverse knowledge sources via maximum entropy in named entity recognition," in *Proceedings of the 6th Workshop on Very Large Corpora*, 1998.

[25] P. Koehn, "Pharaoh: A beam search decoder for phrase-based statistical machine translation models," in *Proceedings of the Association for Machine Translation in the Americas*, 2004.

[26] I. Zitouni and R. Florian, "Mention detection crossing the language barrier," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, 2008.

[27] Y. Benajiba and I. Zitouni, "Using parallel corpora to enhance mention detection," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, 2010.

[28] G. A. Miller, "WordNet: A lexical database," *Communications of the ACM*, vol. 38, no. 11, 1995.

[29] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

[30] F. Huang and K. Papineni, "Hierarchical system combination for machine translation," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 277–286, 2007.

[31] Y. Al-Onaizan and K. Papineni, "Distortion models for statistical machine translation," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 529–536, 2006.

[32] Y.-S. Lee, Y. Al-Onaizan, K. Papineni, and S. Roukos, "IBM spoken language translation system," in *TC-STAR Workshop on Speech-to-Speech Translation*, pp. 13–18, 2006.

[33] J. Hobbs, "Pronoun resolution," Tech. Rep., Dept. of Computer Science, City University of New York, Technical Report TR76-1, 1976.

[34] S. Lappin and H. J. Leass, "An algorithm for pronominal anaphora resolution," *Computational Linguistics*, vol. 20, no. 4, 1994.

[35] R. Mitkov, "Robust pronoun resolution with limited knowledge," in *Proceedings of the 17th International Conference on Computational Linguistics*, pp. 869–875, 1998.

[36] R. Stuckardt, "Design and enhanced evaluation of a robust anaphor resolution algorithm," *Computational Linguistics*, vol. 27, no. 4, 2001.

[37] W. M. Soon, H. T. Ng, and C. Y. Lim, "A machine learning approach to coreference resolution of noun phrases," *Computational Linguistics*, vol. 27, no. 4, pp. 521–544, 2001.

[38] V. Ng and C. Cardie, "Improving machine learning approaches to coreference resolution," in *Proceedings of the Association for Computational Linguistics*, pp. 104–111, 2002.

[39] X. Yang, G. Zhou, J. Su, and C. L. Tan, "Coreference resolution using competition learning approach," in *Proceedings of the Association for Computational Linguistics*, 2003.

[305]

[40] X. Luo, A. Ittycheriah, H. Jing, N. Kambhatla, and S. Roukos, "A mention-synchronous coreference resolution algorithm based on the Bell tree," in *Proceedings of the Association for Computational Linguistics*, 2004.

[41] D. Zelenko, C. Aone, and J. Tibbetts, "Coreference resolution for information extraction," in *ACL 2004: Workshop on Reference Resolution and Its Applications*, pp. 24–31, 2004.

[42] X. Yang, J. Su, J. Lang, C. L. Tan, T. Liu, and S. Li, "An entity-mention model for coreference resolution with inductive logic programming," in *Proceedings of the Association for Computational Linguistics: Human Language Technology*, pp. 843–851, 2008.

[43] A. Rahman and V. Ng, "Supervised models for coreference resolution," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pp. 968–977, 2009.

[44] X. Luo and I. Zitouni, "Multi-lingual coreference resolution with syntactic features," in *Proceedings of Human Language Technology (HLT)/Empirical Methods in Natural Language Processing (EMNLP)*, 2005.

306

[45] V. Ng, "Supervised noun phrase coreference research: The first fifteen years," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 1396–1411, 2010.

[46] E. Bell, "Exponential numbers," *American Mathematical Monthly*, pp. 411–419, 1934.

[47] T. S. Morton, "Coreference for NLP applications," in *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, 2000.

[48] K. Toutanova, D. Klein, C. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 2003.

[49] M. Maamouri and A. Bies, "Developing an Arabic treebank: Methods, guidelines, procedures, and tools," in *Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages*, 2004.

[50] L. Haegeman, *Introduction to Government and Binding*, 2nd ed., Oxford: Basil Blackwell, 1994.

307
/
308

[51] NIST (National Institute of Standards and Technology), "ACE 2005 evaluation," 2005. www.nist.gov/speech/tests/ace/ace05/index.htm.

关系和事件

Daniel M. Bikel, Vittorio Castelli

9.1 概述

词语在世界上无处不在，并且这些词语越来越多地以电子形式储存。截至2008年，世界上存在了超过1万亿个互不相同的网页，这个数字还正在以每天超过十亿的数量增长[1]，而且每个网页中都至少包含一些文本。正如我们在之前的章节中所了解到的，自然语言文本充满歧义并且富含各种信息，实际上这两个属性也是互相补充的。有了这么多电子文本文档，人们对于能够处理这些自然语言文本的计算机系统有着越来越大的需求，这样的计算机系统需要能够自动地将含有自由形式的、便捷阅读的文本合成更加为准确、紧凑的结构化的表现形式，并且能够以更高效的方式来访问和处理大量的文档。比如，一个公司需要追踪用户对其产品的反馈信息；一个政治家需要了解他的选民对他的观点态度；一个智能的分析器需要记录特定、某个群体或者某个组织的人的行为与话语。

若使计算机能够接近于完全理解自然语言文本的内容，需要有一个能够包含句法、语义、语用，以及世界知识的模型，并且有适当丰富的意义表达方式。那种完整的理解程度超出了本章的范围。本章中我们探究更局限的问题，即抽取相关的信息来填充一个与每个特定任务相关的事实的“数据库”。更具体地说，我们将问题定义为寻找一个语料文本中的所有相关实体（在第8章中已经详细讨论），找到这些实体的所有相关属性以及实体间的所有相关关系，并且将这些信息以结构化的方式进行储存。直观地，一旦填充了事实，我们的数据库就可以通过非常简单的数据库查找来回答下面几类问题：

- 在特定的文档或文档集中提到的人或实体是谁？
- 一家特定的公司里有多少员工，并且他们的名字是什么？
- 一些人或实体之间的关系是什么？
- 在一个文档，或者一系列文档中提到了哪些事件？
- 一些特定的事件在什么时间发生？
- 一些特定类型的事件在哪里发生？

尽管从一系列文档中找出事实来填充数据库的目标看起来很适中，但两个广泛信息抽取项目——信息理解会议（Message Understanding Conference，MUC）和自动内容抽取（Automatic Content Extraction，ACE）[4，5，6，7]的有限的成功表明，这是一个令人却步的挑战。

9.2 关系与事件

在第8章中，我们了解了如何识别和查找文本中的提及类型（参见8.2节），以及如何寻找提及共指的内容（参见8.3节）。本章主要研究如何寻找实体间的语义关系。能够处理这种任务的系统通常被称为**关系抽取系统**（relation extraction system）。关系抽取这

个术语在自然语言处理的文献中有几种含义。从广义上看，我们能够区分两条研究的主线，第一条（[8，9]）主要涉及三种特殊类型的关系抽取：

- 抽取与词汇本体联系在一起的关系，比如部分-整体关系，上下位关系以及方式关系；
- 抽取本质上类似的关系，比如发现动词1和动词2表示的是相同的概念，但是动词1要更强一些；
- 查找**相似前提**（similarity enablement），即识别动词1表示的行为是动词2表示行为的先决条件。

研究的第二条主线解决识别潜在的异构实体之间更为普遍的语义联系，比如探究人和公司之间的雇佣关系，疾病和人之间导致死亡的关系，或者一个实体（比如一个公司）拥有者和另外一个拥有者之间的关系。本章主要研究第二条线，即广义类别的关系抽取。

比如，假设我们需要建立一个多语言的系统，这个系统每一次都能够识别出文本中PERSON实体被描述成一些其他实体的拥有者（*owner*）的情况。这种以及许多其他类型的语义关系通常被表述成一个句子，因此在文献中最为常见的研究方式是建立一个系统来寻找**句内**（within-sentence）关系。在这方面，我们希望构建一个能够分析已经识别出实体提及的句子的系统，当一对提及存在时，该系统能够识别两者间“拥有”类型的关系提及。作为一个更有雄心的目标，我们希望有一个系统可以识别实体之间的关系而不用考虑
310 两个实体是否是在同一个句子中被提到。然而，基于本章的目标，我们假设两个实体在同一个句子中的提及作为它们间关系的证据，即使其中一个或者两个实体都是以代词的形式出现（比如“*he owns it*”，其中“*he*”表示的是PERSON实体，而“*it*”表示的是PERSON实体所拥有的公司）。

事实上，我们已经见过一个关系抽取系统——**共指消解系统**（coreference resolution system），它能够找到文档中的共指实体提及间“同一实体”的关系。但是对于那些牵涉超过两个实体的关系要怎么处理呢？当关系包含一个或多个实体的状态改变时，我们称为事件。**事件抽取系统**（event extraction system）可以识别出具有状态改变的实体的集合。比如，“*Mary bought apples for* $20”这个句子中包含了事件“bought”以及三个实体“Mary”、“apples”、“$20”，通过使用谓词演算，我们可以用三元谓词来表示这样的事件。比如bought（Mary，apples，$20），或者二元谓词对bought（Mary，apples）和paid（Mary，$20）。当我们在之后的9.6节讨论如何设计一个事件抽取系统时，这两者的区别就显得很重要了。

9.3　关系类别

正如同提及检测和共指消解，目前许多关于关系抽取的工作都源于国家标准与技术研究所（National Institute of Standards and Technology，NIST）的ACE评测[7]。正如8.2节讨论的一样，ACE的任务包含7个主要类型的实体：设施（FAC）、地理政治实体（GPE）、地点（LOC）、组织机构（ORG）、人（PER）、交通工具（VEH）以及武器（WEA），每个类别又有许多子类别，总计45个实体类别。ACE的竞赛要求系统产生丰富的关系集合，并分成7个主要的类别和18个子类别：

- PHSY（physical）：一个空间关系，表示人处于或者靠近设施、一个地理位置或者一个地理政治实体；设施处于或者靠近一个地理位置或者一个地理政治实体；地理位置处于一个更大范围的地理位置或一个地理政治实体，或者也可能是一个设施；以及一个地理政治实体处于或者靠近另一个地理政治实体。它的子类别是LOCATED和NEAR。
- PART-WHOLE：一个空间关系，表示一个设施、地理位置、地理政治实体或者组织是另一个设施、地理位置、地理政治实体或者组织的一部分。关系子类别

GEOGRAPHICAL 体现了地理位置、设施以及 GPE 之间的 PART-WHOLE 关系；对于组织和有组织角色的 GPE，子类别 SUBSIDIARY 描述了论元间的组织 PART-WHOLE 关系。

- PER-SOC (personal-social)：个人-社会关系，体现了人与人之间的关系，关系可以是 BUSINESS 相关的或者基于 FAMILY 的，也可以是 LASTING PERSONAL 关系，比如友情。因此 PER-SOC 关系有三个子类别来区分这三种情况，偶然的个人与社会关系不在 ACE 的考虑范围之内。
- ORG-AFF (organization-affiliation)：这类关系是关于人与组织之间的关系。一个人可以受雇于一个公司（EMPLOYED 类型）也可以是其中的一个成员（MEMBER 类型）。一个特殊种类的成员或者雇员是从属于运动组织（SPORT-AFFILIATION 类型）。人与组织的关系可以既不是它的成员也不是它的雇员，这种情况一般是公司创始人（FOUNDER）、拥有者（OWNER）以及投资人（INVESTORS-SHAREHOLDER）。最后，当人们作为学生或校友时可以从属于一个教育机构（体现在 STUDENT-ALUMN 类别）。
- GEN-AFF (general-affiliation，GEN-AFF)：一些人与组织、地理政治组织的隶属关系或者组织与地理政治实体的隶属关系不属于之前提到的类别，在这些关系中，我们识别公民身份、国家的居住权、宗教隶属以及种族（所有这些都属于 CITIZEN-RESIDENT-RELIGION-ETHNICITY 的 ACE 子类别）。类似地，一个公司也可以在一个特定的地点或者特定的国家进行商业贸易，这些体现在了 ORG-LOCATION 的 ACE 子类别中。
- ART (artifact，ART)：描述人造制品的使用者、发明者、生产者以及这些人造制品之间的关系。
- METONYMY：同一实体的两个不同方面的关系。最常见的例子是用一个机构名来指代这个机构的设施。

311

当然，系统内部可能产生更精细的关系，并且映射成所需关系集合。在最近几年的 ACE 评测中，这是很常见的情况。除了关系子类别之外，ACE 定义了一些其他属性，比如模态（关系是肯定的还是否定的）、时态（关系是发生在过去、现在还是将来，或者这个关系没有特定的发生时间）。

9.4 将关系抽取视为分类

9.4.1 算法

在这一节中，我们将关系抽取看作一个多元分类问题，在算法 9-1 中简要地展示了该方法：

算法 9-1 关系抽取算法的初始版本，将其看作一个分类问题

```
1: procedure RelExtract d                    // d 是一个文档
2: R ← ∅                                     // R 是该过程输出的关系集合
3: foreach 句子 s ∈ d 具有提及 m_1 ... m_n do
4:   foreach 提及对 m_i, m_j, 1 ≤ i < j ≤ n, do
5:     R ← R ∪ CLASSIFY(m_i, m_j)
6:   end
7: end
8: return R
```

312

在这个简化的情景中，我们使可能的分类标签为：

$$S=\{\text{NONE},\text{Phys.located},\text{Phys.near},\cdots\}$$

随着这样的设计，我们也需要将分类器的输出集扩充为 S 与可能的模态集合 M 和可能的时态集合 T 的向量积，即 $S\times M\times T$。但是这样的联合模型尽管可能实现，也会使数据过于碎片化，使得模型很难得出高度相关的标签。意识到这个潜在的数据稀疏问题，IBM[10] 和其他地方的系统使用**分解**（factored）或**级联模型**（cascaded model），通过一系列的二元和多元分类来执行 CLASSIFY 函数（算法 9-1 中的第 3 行）：

存在	元性	在 m_i 和 m_j 之间是否有任意类型的关系
类别	多元	假设关系成立，是什么类型
子类别	多元	假设某类关系成立，是什么子类型
模态	多元	肯定、否定、可能或者未指明
时态	多元	过去、现在、将来或者未指明
顺序	二元	对于所谓的 p 描述的关系，顺序是 $p(m_i, m_j)$ 还是 $p(m_j, m_i)$

如果存在分类器返回假，则所有整个分类器流水线被短路，这正是所期望的。

最后一个分类器决定了两个提及的顺序，好像它们是谓词的论元。这是因为一些关系（如 BUY）的语义依赖于它们的论元顺序，顺序不同产生的关系可能就不同。考虑下面两个句子：

- Mary bought apples.
- Apples were bought by Mary.

两种情况的关系都是 bought（Mary，apples），其中 Mary 是购买者而 apples 是被买的物品，而不管文本中提及的顺序。然而，对于许多关系类型，论元的顺序是无关的，比如见面关系，只要两个人见面就成立。

将该问题组织为一个级联问题当然只是一个权宜之计，但这是否是唯一的方法？传递每个分类器产生的一个最优输出会产生一个问题，即错误传播，例如，一个 Type 分别器将 SUBSIDIARY 关系误分类，那么 Subtype 分类器也没有希望改正这个错误。解决这个问题的方法除了实现一个联合模型外，就是在过程的每一步中产生 k 个最优的假设，然后在最后选择分数最高的假设。

9.4.2　特征

基于分类的关系抽取器有几个主要的类别特征，包括结构、词汇、基于实体、句法以及语义。关系分类的特征一般编码正在分析的提及对的特定性质，或描述该提及对是如何
[313] 在句子上下文中出现的。

结构特征。考虑下面的句子：

In 1860 there was a four-way race between the Republican Party with Abraham Lincoln, the Democratic Party with Stephen Douglas, the Southern Democratic Party with John Breckenridge, and the Constitutional Union Party with John Bell.

在候选人和他们各自的党派中有 4 个 ORG-AFF 的 ACE 关系。在 *race* 和 1860 之间也存在一个 TimeOf 关系（非 ACE 关系）。这个例子描述了一个很直观的概念，即在某种意义上，如果提及对相隔很远，那么它们很难被一个关系联系起来，而两个相距很近的提及对通常会参与到同一个关系当中。结构特征的第一个类别体现的是提及的距离，可以用一些合适的方法来衡量，比如中间词元的数量、中间提及的数量，或者在句法树中两个提及之间的最短路径长度。当前考虑的提及对之一或两者与其他提及有关系时，会触发另一类的

结构特征。为了能在一次迭代的解码算法中使用，这些特征必须是具有因果关系的，即它们只有在其他关系存在时才能起作用，而这些关系又必须能由解码器在当前提及对间产生任何关系之前检测到的。比如，考虑句子：*Mary bought apples and pears*：解码器先考虑提及对（Mary，apples），然后考虑提及对（Mary，pears），当解码到后面的提及对时，会触发特征 FirstArgAppearsInBoughtRelation，而当解码第一个提及对时，不会触发相关特征。

词汇特征。*Bob married Mary* 和 *Bob called Mary* 这两个句子在结构上是相似的，但是传递了不同的信息。特别地，第一个句子包含的是 ACE 中 PER-SOC.FAMILY 关系的例子，而第二个句子不包含 ACE 关系 PER SOC.FAMILY、PER-SOC.BUINESS 或 PER-SOC.LASTING-PERSONAL 中的任意一个。为了检测出关系并进行恰当的分类，除了短句子本身外还需要更多的信息。结构特征还不足以来区分两种情况，而如果包含了词汇信息，系统就有可能做出正确的判断。词汇特征包含了当前分析的提及中一些或全部词的信息，如果一个实体是命名实体，那么它通常还会伴随着一个实体首词的特殊特征。这个类别中的其他特征所包含的词汇一般出现在分类出的提及的两端的小窗口中，以及在两个提及的左边、右边或者中间的所有动词中。不同于结构特征，词汇特征会大幅度地提高特征空间的维度。因此，对于有丰富词法的语言，即使是英语，也通常用形态分析器或词干还原器来确保忽略词缀。

基于实体的特征。考虑 *I went to France* 和 *I went to IBM* 这两个句子：第一个句子包含了第二个句子中没有的 PHYS-LOCATED 关系。除非 France 和 IBM 同时出现在训练集与上面相似的句子中，否则单纯依靠结构和词法特征的关系检测器很难区分出这两种情况。然而，在第一个句子中，提及 *France* 类型是 GPE，角色是 LOCATION，而 IBM 的类型是 ORG，并且没有特定的角色。这个例子暗示了提及对的特征属性的重要性，比如类别、子类别以及相应实体的角色，还有提及的等级（名称、名词、代词）。这个特征同样也不鼓励系统在句子 *France was ousted during the first round of the World Cup* 中寻找 LOCATION 关系，因为 France 这个 GPE 在句子中表示的是一个球队的角色。 314

句法特征。这个特征和共指消解系统中使用的十分类似。这些特征自身就被分为两种子类别：基于标记的和基于路径的。基于标记的特征会关注与当前分类的提及中的词相联系的非终结符标记，至多包括词性标记。词性标记通常是词的直接父节点的非终结符。基于路径的特征更加细化，它们表示正在进行分类的两个提及的中心词的最短路径的编码方式。基于路径特征的例子包括：

- 覆盖了提及对的最小子树的根的成分标记。
- 根节点的子女标记列表：两个提及间最短路径上所有或选定的成分标记。
- 当两个提及在同一个短语、名词短语、句子中时触发的指示函数。
- 检测特殊模式的指示函数，比如 Metion1-PP，其中 PP 包含了 Metion2。
- 一个用来显示是否有提及之一是或者包含最小覆盖子树的中心词的指示。

句法特征中一个重要的类别是从依存树中推导而来：它们包含了依存树连接提及范围（mention extent）或者提及中心词（mention head）的完整或者缩减的路径。这些路径都用成分标记进行表示，可以是词汇化标记，也可附带额外标记。

语义特征。在句子 *Both have since left the embattled company* 中，提及 *both* 和 *company* 通过 ACE 关系 ORG-AFF 相联系的证据由它们的语义角色简单体现[11]：第一个提及是动词 *left* 的 ARG0 角色，而第二个的角色是 ARG1。语义特征依靠语义角色标记来显示两个提及之间的连接类别，比如，当一些提及（被句法树节点标签覆盖）是同一个动

词的论元时会触发时；还有的是当提及之一是一个动词的论元而另一个不是时触发[⊖]。

可以设计额外的特征，同时属于多个类别。覆盖了提及对的最小子树的中心词便是一个基于句法信息的词汇特征的例子，覆盖提及对的最小 VP 的中心词也是一个例子。Mention1 是离 Mention2 最近的具有实体类型 Type1 的提及这一信息是一个混合的基于结构-实体的特征，该特征在预测关系的存在及属性时十分有用。

自然我们会问这些特征对于预测和标记关系能起到多大的帮助。尽管这个答案与关系
315 的特定分类方法以及应用的领域有关，但基于实体的特征和句法特征对于关系检测和关系分类都能起到很大的帮助。特别地，依存特征在关系抽取领域有着广泛的应用，甚至超过了后面将要描述的判别性分类的范围。Jiang 和 Zhai [12] 描述了一个能系统性评价英语中用于检测和分类关系的多种特征类别的有效性的方法。

9.4.3　分类器

前面章节定义的特征定义了一个相当大的特征空间，这会在学习统计分类器的时候产生维数灾难。选取一个能不受庞大的特征空间以及数据稀疏问题影响的分类器才是明智之选，因此，最近关于关系抽取的研究工作都依赖于最大熵（MaxEnt）[13]、支持向量机（SVM）[14]、条件随机场（CRF）[15]，依赖程度低一点的还有朴素贝叶斯 [16] 和机械抽取器（rote extractor）[17]。

最大熵分类器是一个简单的指数模型[⊖]。用 (x, y) 表示训练样本，其中 x 是证据（比如一个提及对和它们的支持），y 是独立的变量（指示函数，如指示检测关系时提及间关系存在与否，指示如关系类型这样的关系属性值）。$f(x, y)$ 表示二元的特征函数，例如，当第一个实体的实体类别是 PERSON，而第二个实体的实体类别是 ORGANIZATION，并且两个提及之间没有关系，那么 $f(x, y) = 1$。我们可以使用前一节描述的特征来建立大量的二元特征函数，对训练集抽取出来的每个特征及每个特征值和与该特征值相关的被预测的属性的每个观察值建立一个特征函数。使用递增索引 i（running index i）来任意地索引这些特征函数。最大熵模型估计给定 x 后 y 的条件概率：

$$p(y \mid x) = \frac{1}{Z_\lambda(x)} \exp\left(\sum_i \lambda_i f_i(x, y)\right)$$

其中分母是称为**配分函数**（partition function）的归一化常数，保证右边所有的 x 对于 y 的条件概率之和为 1。权重 λ_i 可以通过在给定约束下最大化训练集的概率获得。约束条件为：单个计数的边际要等于经验边际（即归一化的计数）。

有两个主要原因使得最大熵分类器很吸引人：首先，它们使用的是对数线性模型，它们极其简单的函数形式能使它们对于数据稀疏以及维数灾难具有鲁棒性；其次，学习一个最大熵分类器可以被看作一个概率单纯型上的约束优化问题，这是一个已经有深入研究的问题，并且存在高效解。最大熵分类器已经被成功使用，例如，Kambhatla [10, 18] 级联模式的每一阶段均使用最大熵分类器。Kambhatla [18] 的进一步评论指出，ACE 的语料库中进行关系抽取主要的错误源于遗漏；关系属性分类器的错误率远比检测阶段要小。
315 根据我们的经验，这并非是一个罕见的问题，并且很可能是因为大部分存在的提及对没有参与进同一个关系而造成了该问题。作者通过在最大熵分类器之上加入一个重抽样层（bagging layer）[19] 来解决这个问题：先从原始训练集中进行独立的置换采样获得几个

⊖ 参见第 4 章来获取对语义角色标注的全面认识。

⊖ 这里条件为分类器的最大熵模型的简单描述，对用于序列分类问题的最大熵模型也详细的概述请见 8.2 节。

不同的子训练集，然后再从这几个训练集中训练出25个最大熵分类器。当对一个提及对进行分析时，所有的分类器都参与，然后用它们的结果进行投票。如果至少3个分类器参与了，系统就接受某关系的存在。实验结果证明抽样法获得了7%的ACE分数提升。我们对最大熵分类器进行简短的总结：最大熵分类器适用于多语言的关系抽取，使用的是不局限于获取个别特征值的特征函数，在实际中，如9.4.2节描述的组合特征是很常见的情况。

SVM分类器是个二元分类器，通过在特征空间中使用超平面然后对不同类的样本进行分类。如训练数据是线性可分的（即被超平面分割），那就有无限多的超平面可以用来分割数据。SVM的一个明显特征就是选择与两个类别最近的样本都有最大距离的超平面。这个属性使得SVM有着比其他基于支持向量的方法有着更优的属性。SVM被用于关系抽取有两个主要原因。首先，学习SVM的问题可以被看作对偶的空间受限优化问题，它甚至可以在高维空间中高效解决；其次，SVM可以学习十分复杂的决策面（比超平面更加复杂），它将特征空间隐式地映射到更高维的空间中，在此空间中学习分割的超平面，再将这个超平面映射回原始的空间中。映射依赖于核函数，它本质上是一个满足一些数学性质的原始空间点的相似性度量。通过使用核函数，分类器能够隐式地解释特征之间的相互作用；比如，一个平方核使得SVM可以解释特征之间的所有成对的相互作用。

9.5 关系抽取的其他方法

9.5.1 无监督和半监督方法

基于特征的有监督判别式方法并不是关系抽取的唯一方法，本节我们将对适合进行多语言关系抽取的技术进行一个综述。

有监督方法的一个显著局限性就是需要一个人工构建的大训练语料。只有选定的语言才有公开的关系抽取训练语料，比如带标注的英语、中文、阿拉伯语语料是ACE评测提供的。构建一个带标注语料的代价如此之大，使得无监督和半监督方法变得更有吸引力。

到目前为止，很少有纯粹的，用来解决关系抽取问题的无监督方法。Gonzàlez 和 Turmo [20] 描述了一个基于应用二元特征的集成聚类方法来处理关系抽取问题。基于集成聚类的方法产生了一个混合的多元伯努利分布；每个分布都分配一个分数，使用协方差矩阵的特征值之和作为度量，越"紧凑"的分布分值越高；每个训练样本分配了一个分数，分数是通过该样本属于某个类别的概率的聚类权重计算得来的；通过分析训练样本分 317
数的直方图可以找到一个改变点（直方图中的一个拐点），从而可以用它作为一个阈值；新的提及对也用同样的方式进行分析，那些分数高于阈值的样本就会被认为有关系相连。在选定的ACE关系子集的ACE语料库中，与标准提及相对比，这种思想简明的方法获得了58的F值，和 [21] 有监督的方法的63.2的F值相比，已经很高了。

尽管纯粹的无监督方法具有成功用于关系检测的潜力，但它们不适用于关系分类。当只有小规模的带标记语料可用时，研究者可以通过半监督学习方法 [22] 将它们与无标记的语料一起使用。半监督的一个常见的方法是举行，过程是使用带标记的样本来猜测临近的未标记样本的标签，之后将其添加进训练语料中。举行应用于关系抽取的例子可以在Chen等人 [23] 的文章中找到。这几位研究人员使用了标记传播算法 [24]，它是一个基于图的算法，该算法将提及对表示成节点，边的权重则是用两个提及对的相似度来计算。标记从带标记的样本迭代地传递到最近的顶点，从而保证原始带标记的样本不会被重新进行标记。在关系检测任务中，该方法的性能在只有1%的标注数据的情况下可以得到F一

58.5；而当数据全部有标记时，性能会提升至 F=71.1。当 10%的数据有标记时，F=63.2，而对于关系检测和分类，F 值从只有 1%的带标记数据时的 39.0 上升到了所有数据都标记时的 54.6。当有 10%的标记数据时，F 值为 43.6。

而 Greenwood 和 Stevenson [25] 中也描述了类似的方法，它依赖于从依存树中获得的模式（基于依存树的模式的使用下面将会进行更详细的讨论），表示为三元组的链，其中每个三元组包含了一个词、该词的词性标注以及与其父动词的关系，和依存树中所定义的一样。作者提倡了一个方法，从语料中抽取大量模式，由标注人员提供有意义的标注模式作为初始种子集合。这些模式然后与一个相似性函数一起来查找一系列与已标记模式类似的未标记模式；然后用最接近的带标记样本来对这些未标记的模式进行标记，重复这个过程。在 MUC-6 数据集 [2] 上的实验结果显示，通过使用带标记的种子集合，半监督的方法能有效地提高了 F 值，并且随着迭代次数的增加 F 值还会提高（在本例中为 190），但是实验结果仍不如那些使用大量带标记训练集的实验。

Ravichandran 和 Hovy [17] 提出了一种很直观的过程，学习使用 Web 捕捉关系的表层模式。他们先提出了由已知存在目标关系的实体对组成的查询，进而识别出包含两个实体提及的句子。假设这些句子可以描述关系，则使用后缀树构造器来识别涉及这些提及的词法模式。为了在问答系统任务中评测该方法的性能，作者提出了一个过程。首先构建一个由实体对组成的查询，实体之一充当问题论元的角色，另一个实体充当问题答案的角色。然后，作者计算包含这两个查询论元的模式的出现次数，以及包含问题论元的模式出
[318] 现的次数，不考虑问题答案在这个模式中是否出现。最后，该方法的精确度可以用这两个数值之比来计算。Alfonseca 等人 [26] 描述了另外一个评价机械抽取器性能的方法，该抽取器是根据 Ravichandran 与 Hovy 所描述的无监督过程来进行训练的。

9.5.2　核方法

9.4 节中描述过基于最大熵分类器以及其他生成或判别式分类器。而最近受到广泛关注并且可能适用于多语言关系抽取的另一种方法是基于核的算法。这些方法的要点为：首先是通过抽取合适的模式来描述关系，其次是通过子模式的匹配次数来计算模式间的相似度。直观上看，如果句子中参与关系的提及是十分接近的，那么描述关系的模式很可能会很简单，而复杂度会随着提及间距离的增长而急剧上升。即便是对涉及提及间距离适中的“长距离”关系，匹配整个模式也会因此不可行。但是，需要指出具有足够多出现次数的子模式也是很有用的关系指示，并且匹配这些子模式对于检测和分类关系而言也许已经足够。根据子模式匹配的数量，核方法可以高效地描述模式间的相似度。它们的主要吸引力在于，可以用高效的计算方式来计算匹配的子类别数量（穷举方法随着模式长度的增加呈指数级增长），并且能够很容易地与强大的判别式分类器以及相关方法一起使用。SVM [14] 和投票感知机 [27] 都是很好的例子。这些方法对于资源贫乏的语言更具吸引力，因为它们一般只依赖于有限数量的特征，而不像传统的基于分类的方法那样需要大范围的异构特征（heterogeneous feature）。

Zelenko、Aone 和 Richardella [28] 描述了一个文本的浅层句法分析的核 [29] 来学习文本中的 PERSON-AFFILIATION 和 ORGANIZATION-LOCATION 关系。浅层句法分析中的节点有类别和角色属性，还有许多其他属性。关系核被定义为一个匹配函数，决定了两个节点是否匹配，也是一个相似度函数，使用这些节点以及他们子女的属性来递归地计算两个节点的相似度。

Culotta 和 Sorensen [21] 将训练集中的关系实例表示为扩展的依存树；他们通过丰

高句子表示扩展了 Zelenko 等人的工作，提出了一个更一般化的框架来调节特征权重，并且采用了复合核。复合核是树核和词袋核的组合，将树看作一个特征向量。

Bunescu 和 Mooney [30] 提出了一个关系核，计算了两个序列之间相同词的子序列的数量，并用第一个和最后一个词间的距离来加权。作者将该核与 SVM 学习包一起使用，并且在两个不同的数据集 Aimed 和 ACE 上，显示出该方法比已存在的基于规则的系统的效果有提升。作者在随后的论文 [31] 中使用弱监督设置（weakly supervised setting）来扩展核方法。 319

9.5.3 实体和关系检测的联合方法

联合推理是最近才研究的 NLP 领域，该方法可以同时处理多个问题。特别地，新兴工作的目标在于同时抽取提及和关系。直观上认为，文本中通常会传达能由关系或事件捕捉到的信息，因此，涉及候选实体提及关系的存在是该提及存在的证据指示。反之，假设实体提及被检测出来，一对实体提及间的关系会受包括提及检测算法生成的后验概率在内的实体提及属性的影响。尽管这些领域都仍在发展初期，然而对于多语言关系抽取却具备很大的吸引力，尤其是对于资源贫乏的语言，现有的研究方法得到的结果并不理想，但联合推理可以在缺乏额外资源的情况下提升性能。

9.6 事件

从广泛意义上来看，事件表示能用自然语言文本描述的世界中任何状态的变化。事件抽取是指通过使用任意算法来抽取出该状态变化的结构表示，尤其包含了参与的实体。典型地，一个词，一般为动词，表示了状态的变化，而动词的论元通常是参与事件的实体。因此事件可以被看作关系的一般化，是实体和单一触发器（典型地仍是动词）的关系集。

在 2004 年的 DARPA ACE 评测 [32] 上，参加的系统被要求找到事件的 5 个主要类别，识别出 7 个不同类别的实体，正如表 9-1 所示。重要的是，其中有可能有同个事件中同个类别中的多个实体。比如，句子 *The criminal destroyed the car and the building* 中，*car* 和 *building* 都应该被标记为对象。

事件的概念在 ACE 2005 的评测 [6] 中被精细化。事件触发器的概念被具体化了（尽管触发器检测从来不是评测度量标准的一部分），而抽取出的事件类别也更为具体，参与的实体类别也是一样，如表 9-2 所示。评测指南将事件限制为句子中使用动词的那些明确的提及。从这方面来看，ACE 事件抽取十分类似于第 4 章描述的语义分析的任务。实际上，这可以被认为是定向的、实际的语义分析应用。

表 9-1 ACE 2004 事件抽取任务

(a) 事件类别
破坏/毁灭（BRK）
创造/改造（MAK）
转让或占有或控制（GIV）
移动（MOV）
状态的变化（CNT）

(b) 事件参与角色	
角　色	描　述
Agent	事件发生的原因
Object	参与事件的实体
Source	原始位置（仅针对 MOV 和 GIV）
Target	最终位置（仅针对 MOV 和 GIV）
Time	事件发生的时间
Location	事件的位置
Other	其他参与角色

表 9-2 ACE 2005 中的事件类型和子类型

类　型	子 类 型
Life	出生、结婚、离婚、受伤、死亡
Movement	交通
Transaction	转让所有权、转让金钱
Business	开公司、合并公司、宣布破产、关闭公司

（续）

类　型	子类型
Conflict	攻击、示威
Contact	见面、打电话
Personnel	开始位置、结束位置、任命、选举
Justice	逮捕入狱、假释、审讯、辩护、控告、有罪、宣判、罚款、行刑、平冤、无罪、上诉

9.7　事件抽取方法

对于一个句子中的事件抽取有两种主要的方法。第一种方法，由 IBM、纽约大学［33］以及阿姆斯特丹大学的 David Ahn［34］所探索的基于流水线的方法：首先有一个触发器检测系统能够寻找和 8 个目标事件类别一致的动词，然后其他系统尝试寻找与事件角色相一致的提及。更具体地，典型的分类过程如下所示：

1）触发器识别。

2）论元识别。

3）论元归属分配。

4）事件共指。

320
|
321

在纽约大学的系统中，融合了手写的启发方法以及一些机器学习的元素。针对前三个任务，Ahn 对比了基于记忆的学习［35］以及最大熵分类器，得到了近乎相同的结果。IBM 的系统在前三个任务中使用了最大熵模型，在第四个任务中使用了启发式方法。

IBM、纽约大学、Ahn 和其他人使用的特征都与用来捕捉关系的特征很相似，原因也很合理，因为事件抽取任务中使用的流水线方法与关系寻找中的很类似，在关系寻找中每个关系的一端即为事件触发器。大部分的特征类别都是独立于语言的，当然基于英语的词汇层次体系 WordNet[㊀] 的特征是个显著的例外。然而，寻找触发器更类似于提及检测而不是关系检测，因此，更常见的方法是通过位置和词法特征使用提及检测系统。比如，IBM 系统简单地探索了带标记的数据并且运行了它的提及检测系统，将提及检测看作是使用 BIO（开始、内部、外部）标答集［37，38］的多类别标记问题。

论元识别和分类可以看作是两个分解的问题，正如 Ahn 的系统中实现的一样，或者看作单个的分类步骤，或者像 IBM 模型里的方法。在这两种方法中，每个提及被作为触发器可能的论元独立考虑。IBM 通过从左到右的方式分类，提及细化了这个方法（句子前面的提及比句子后面的提及先进行分类），后面的分类可将前面的作为条件。这就是**贪心优先**（greedy best-first）解码策略。

第二个主要的方法是由 BBN［39］发现的，使用初始的处理（使用机器学习、启发式或者两者进行训练）来提出整个事件——触发器和他们的已标记论元，然后使用分类器来决定输出的事件。这个方法受结构化预测问题的重排序方法［40］的启发，也与该方法很类似，正如在 3.5.3 节描述的。所有的事件分类器都是使用有监督方法训练而成的，对给定句子的给定触发器，通过使用一个单趟事件“提议”算法，能够生成许多事件候选。该单趟算法为每个事件生成一个分数，如果候选集中分数最高的事件不是该触发器的真实事件，那么真实事件就会被添加到训练集中的正类训练样本中（通常用 1.0 标记），而分数最高的候选就会被添加到负类训练样本中（通常用 −1.0 标记）。这样，就可以学习出一

㊀ 除了英语，其他语言可用的类似 WordNet 的资源是比较少的。如果你正在开发一个关系或者事件抽取系统，建议去寻找与你系统使用语言相关的词汇资源。

个超平面来区分单趟系统产生的候选事件。和很多结构预测重排序文献一样，BBN 使用了一个类似感知机的分类器来学习超平面。

这个方法的优势在于它考虑了事件实体相关的特征：触发器以及它的所有带标记的论元。相比之下，流水线方法也考虑了每个独立的论元或者只查找之前产生的论元，并且分类器只有一个最优输出。这个算法的缺点在于解码策略稍显复杂。 322

9.8 超句

抽取事件的初始工作集中于独立地处理每个句子，将事件提及归结成事件一般用一个简单的过程来解决，通过启发式地查找匹配的触发器和论元，并且充分利用已经实现的共指消解。随后，出现了一个研究思路，通过利用超句信息来帮助事件抽取，有两种策略。

Ji 和 Grishman [41] 从“每个语篇一个意义”的限制 [42] 引申出“每个主题类一个意义”的限制，其中主题类是同样主题的文档的集合。思想是，提到事件或者与主题相关的词语可以清楚地表示文档集中的事件。所以要分类的事件提及，包括触发器及其论元，要与其他在同一个主题类别中的触发器和论元相一致。Ji 和 Grishman 使用了一个开源的文档搜索引擎——INDIR [43] 来收集与目标文档主题相关的文档，然后再对跨文档频率使用人工调节权重来追求局部决策的一致性。在 ACE 事件抽取任务中，使用了统计一致性策略后，能在触发器分类上获得 7.6% 的绝对 F 值提升，而基准系统对于论元分类获得了 6% 的绝对 F 值提升。

在句子之外的相关工作中，Liao 和 Grishman [44] 发现在主题类文档中，不仅文档中的触发器对于文档和文档的主题分类是一致的，其他事件类型跟目标事件类型也有很强的关联，这样，文档中其他事件类型的存在就可以为目标事件类型提供很强的指示。比如，作者找到了 *Attack*、*Transport* 和 *Injure* 事件通常和事件 *Die* 同时出现，它们的相关系数超过了 0.3。利用这些事件之间的一致性，与基准系统比，它们在触发器分类上获得了 9.0% 的绝对 F 值提升，而对于论元分类则获得了 8% 的绝对 F 值提升。

9.9 事件匹配

信息抽取尤其是事件抽取的目标通常是把感兴趣事件记录到数据库中，但这不是信息抽取技术唯一的用途。对于尝试回答开放式问题，尤其是能够处理答案不止一个的问题的问答系统，信息抽取系统能够帮助产生有价值的句子级别的信息。这种类型的系统会在第 14 章中详细描述，这里我们会描述一些子问题，即在已有事件描述的情况下，在语料库的句子中查找同样包含那个事件的描述。

Bikel 和 Castelli [45] 开发了一个二类分类器，将事件和句子的描述作为输入，如果 323
句子中包含这个事件的描述则返回真，否则返回假[⊖]，他们选择使用了平均感知器算法 [27]，并且选择了两种类别的特征用于分类器的训练。

第一类特征是**低级特征**（low-level feature），其中包含**词汇特征**（lexical feature）和**提及匹配特征**（mention-matching feature）。词汇特征用来衡量事件描述和句子中同时出现的词汇的百分比。为了能够使用二元特征函数，这个百分比被放入 5 个区间中，[0，0]、(0，0.33]、(0.33，0.66]、(0.66，0.99]、(0.99，1.0]，并且在每个区间中创造一个特征函数。如果一个提及同时在事件描述和句子中出现，那么提及匹配特征就是

⊖ Bikel 和 Castelli 也是本章的作者。

二元特征函数。每个提及类别都创建一个这样的二元特征（了解更多提及检测的内容参见第 8 章）。

第二类的特征是**高级特征**（high-level feature），利用了语料库中句子和问答的依存句法分析（参见第 3 章）。就我们的目的而言，我们对句子 $w=\langle w_1, w_2, \cdots, w_k\rangle$ 定义了一个依存树 $\tau=\langle V, E, r\rangle$，其中 $V=\{1,\cdots,k\}$，$E=\{(i,j): w_i$ 是 w_j 的子节点$\}$，$r\in\{1,\cdots,k\}$：w_r 是根节点的词。不使用节点只是简单句子中的词的标准依存树，模式中每个单词 w_i 关联一个词性标注 t_i、一个形元或者词根 m_i（如果 w_i 没有变形就是它本身），一个非终结符标签集 N_i，该词的同义词集 S_i，以及一个规范提及 $cm(i)$。更形式地，我们让句子中的每个元素成为一个 6 元组 $w_i=\langle w_i, t_i, m_i, N_i, S_i, cm(i)\rangle$。在这种情况下的依存树是由中心词汇化（head-lexicalized）的成分树推导而来，中心词汇化意味着一个中心词可以关联多个非终结符标签，这也是为什么 N_i 是一个集合而不是单一的非终结符。单词 w_i 的规范提及 $cm(i)$ 是这个单词最长的名字提及，以防该词是一个代词而与同一个文档中的其他提及共指。

模型中高级特征利用了**依存关系**（dependency relation）的传递闭包，即由依存句法分析产生的中心词-修饰词关系。我们可以在理论上将这个中心词-修饰词关系看作一个集合论的关系，用 aRb 表示，句子中 a 是中心词 b 的修饰词，R 表示修饰关系。比如在一个短句 *John saw Mary* 中，我们可以有 Mary R saw。从理论上看，我们可以观察到所有关系的**传递闭包**（transitive closure），对于 $\forall a, b, c$：$(aRb \wedge bRc) \Rightarrow aRc$。假如我们有一个稍长的句子 *John saw someone talking to Mary*。在这种情况下，对它的句法分析是：

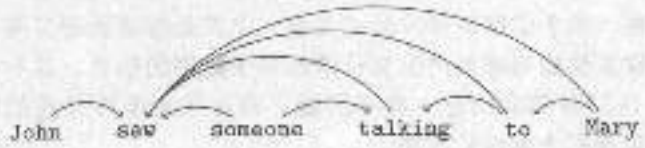


我们可以有中心词-修饰词的关系：

- talking R saw
- to R talking
- Mary R to

[324]

等，然而如使用传递闭包，



我们也可以包含 Mary R saw[⊖]。在两个句子中，John R saw 都存在，因此 John 也是主动词 saw 的一个修饰词。

更形式地讲，如果 E 是依存句法分析的子女（child-of）关系，那我们将 E' 作为 E 的传递闭包，它是后代（descendant-of）关系。因为我们最终尝试构建一个模型来决定一个事件描述 e 是否在句子 s 中提到。我们用 E'_e 表示事件描述中的后代关系，用 E'_s 表示句子中出现的后代关系，高级特征实质上计算 E'_e 和 E'_s 之间是否有重叠。为了实现这些，我们定义了一对匹配函数，使得我们可以决定一个依存对 $d_e=(d_e, d, d_e, a)\in E'_e$ 是否等价于 $d_s=(d_s, d, d_s, a)\in E'_s$，其中 d_s, d 表示依存对的后代而 d_e, d 表示的是祖先。第一个匹配函数是

⊖ 这种关系类型也是语言学中提到的简单支配关系（dominance relation）。

检测依存对 d_x 的两个后代是否相等而第二个匹配函数 match_a 是用来检测两个祖先是否相同。

$$\text{match}_d(d_x,d_y) = (m_{d_x,d} = m_{d_y,d}) \vee (cn(d_x,d) = cn(d_y,d))$$

$$\text{match}_a(d_x,d_y) = (m_{d_x,a} = m_{d_y,a}) \vee (cn(d_x,a) = cn(d_y,a))$$

这两个匹配函数表明，如果两个后代或者祖先有相同的形元或者有相同字串规范提及，那么它们就是相等的。最后，对于两个后代 d_x 和 d_y，我们定义了一个全局匹配函数：

$$\text{match}(d_x,d_y) = \text{match}_d(d_x,d_y) \wedge \text{match}_a(d_x,d_y)$$

如果 $\text{match}(d_x, d_y)$ 返回错误，则 d_x、d_y 用基于重叠的同义集进行更弱类别的等价性测试：

$$\text{synmatch}(d_x,d_y) = (S_{d_x,t} \cap S_{d_y,a} \neq \emptyset) \wedge (S_{d_x,a} \cap S_{d_y,a} \neq \emptyset)$$

表 9.3，源于 Bikel 和 Castelli [45]，展示了模型中对于例子事件描述 e = *Abdul Halim Khaddam resigns as Vice President of Syria* 和句子 s = *The resignation of Khaddam was abrupt* 使用的特征类别。

模型中最后一个特征类别是基于依存集合 E'_e 和 E'_s 上量化核函数的值：

$$K(E'_e,E'_s) = \sum_{(d_x,d_y)\in E'_e\times E'_s:\text{match}(d_x,d_y)} (\Delta(d_x)\cdot\Delta(d_y))^{-1}$$

325

其中 $\Delta((i,j))$ 是依存树 t 中节点 i 到节点 j 的路径距离。这个核函数实质上测量匹配的依存集 E'_e 和 E'_s 之间的整体距离。

表 9.3 依存匹配特征类型：实例特征里 $x\in\{m, s\}$，取决于依存匹配是因为 $\text{match}(d_x, d_y)$ 返回真还是因为同义词集匹配 $\text{synmatch}(d_x, d_y)$

特征类型	实例	评论
Morph bigram	x-resign-Khaddam	稀疏但有用
Tag bigram	x-VBZ-NNP	
Nonterminal	x-VP-NP	所有对来源于 $N_i\times N_j, (i,j)\in E_t$
Depth	x-eventArgHeadDepth=0	E'_x 深度为 0 且为 resign 事件的根

通过使用依存关系的传递闭包，模型获得了更强大、更一般的方法，能够匹配事件描述在句子中出现的方式。Bikel 和 Castelli 在一个只有 3546 个样本的训练集上训练了这个模型，并使用了一个有 485 个样本的小型开发测试集，获得了 66.5% 的 F 值，重要的是，他们可以通过调整模型来牺牲精确率提高召回率，反之亦然。

9.10 事件抽取的未来方向

正如本章刚开始描述的一样，事件抽取十分类似于语义分析和语义角色标注。这两项任务都热衷于抽取谓词-论元结构。但是对于事件抽取，目标限制于预先定义的谓词类别集。随着更加复杂的语义分析系统的出现，我们可能看到这两种关于谓词-论元提取方法的融合。特别地，带标记资源如 PropBank [46] 和 NomBank [47] 的可用性意味着我们可以建模更一般化的谓词-论元抽取，这样我们就可以把特殊事件抽取看作过滤问题，而不是单独的建模问题。此外，如果目标是回答问题，正如我们在前一节中见到的，那么这样的过滤可以当需要时才做。

9.11 总结

本章我们回顾了文本中关系和事件抽取的主要方法。通过依赖于低级组件如词性标注

器、句法分析器、提及检测组件以及共指消解系统等提取的特征。关系和事件能够组成信
326 息抽取工具中的高级组件。关系和事件抽取系统都依赖于词汇特征。这些特征提供了丰富但是潜在稀疏的、有用的区分性信息，并且基于句法分析树路径的特征通常能够更好地泛化以及处理长距离的依存。而且，两类系统都依赖于提及检测来识别关系和事件的参与者，并且提供容易泛化的关系，因为只有少量的提及类别。

事件抽取系统和关系抽取系统如此类似的一个主要原因是，事件本身可以被认为是围绕某个锚点的关系的集合。因此，正如我们所见到的那样，事件同样与语义角色标注系统十分相似。

关系和事件抽取的主要目标是结构化地表示文本中的信息，这样它可以进入数据库进行搜索，比简单地进行关键字搜索更容易、更有效。我们也看到了低级的信息抽取过程能为事件匹配提供基础。正如将在第 14 章看到的一样，这类方法也可以形成开放的问答系统的基础。

参考文献

[1] J. Alpert and N. Hajaj, "We knew the web was big," Blog post: http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html, July 2008.

[2] *Proceedings of the 6th conference on Message Understanding (MUC-6)*, Association for Computational Linguistics, 1995.

[3] *Proceedings of the 7th Message Understanding Conference (MUC-7)*, Association for Computational Linguistics, 1998.

[4] *Proceedings of ACE 2003 Workshop*, 2003.

[5] *Proceedings of ACE 2004 Workshop*, 2004.

[6] NIST (National Institute of Standards and Technology), "The ACE 2005 (ACE05) evaluation plan." http://www.itl.nist.gov/iad/mig/tests/ace/2005/doc/ ace05evalplan.v2a.pdf, 2005.

[7] NIST, "The ACE evaluation plan." www.nist.gov/speech/tests/ace/index.htm, 2007.

[8] T. Chklovski and P. Pantel, "Verbocean: Mining the web for fine-grained semantic verb relations," in *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*, 2004.

[9] T. Chklovski and P. Pantel, "Large-scale extraction of fine-grained semantic relations between verbs," in *International Workshop on Mining for and from the Semantic Web*, p. 12, 2004.

[10] N. Kambhatla, "Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations," in *Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions*, p. 22, 2004.

327 [11] D. Gildea and D. Jurafsky, "Automatic labeling of semantic roles," *Computational Linguistics*, vol. 28, no. 3, pp. 245–288, 2002.

[12] J. Jiang and C. Zhai, "A systematic exploration of the feature space for relation extraction," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 113–120, 2007.

[13] A. Berger, S. Della Pietra, and V. Della Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[14] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[15] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 282–289, 2001.

[16] P. Domingos and M. Pazzani, "On the optimality of the simple bayesian classifier under zero-one loss," *Machine Learning*, vol. 29, no. 2-3, pp. 103–130, 1997.

[17] D. Ravichandran and E. Hovy, "Learning surface text patterns for a question answering system," in *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 41–47, 2002.

[18] N. Kambhatla, "Minority vote: at-least-n voting improves recall for extracting relations," in *Proceedings of the COLING/ACL on Main Conference Poster Sessions*, pp. 460–466, 2006.

[19] L. Breiman, "Bagging predictors," in *Machine Learning*, vol. 24, p. 123, 1996.

[20] E. González and J. Turmo, "Unsupervised relation extraction by massive clustering," *Data Mining, IEEE International Conference on*, vol. 0, pp. 782–787, 2009.

[21] A. Culotta and J. Sorensen, "Dependency tree kernels for relation extraction," in *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, p. 423, 2004.

[22] O. Chapelle, B. Schölkopf, and A. Zien, eds., *Semi-Supervised Learning*. Cambridge, MA: MIT Press, 2006.

[23] J. Chen, D. Ji, C. L. Tan, and Z. Niu, "Relation extraction using label propagation based semi-supervised learning," in *ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pp. 129–136, 2006.

[24] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Local and global consistency," in *Advances in Neural Information Processing Systems 16*, pp. 321–328, Cambridge, MA: MIT Press, 2004.

[25] M. Greenwood and M. Stevenson, "Improving semi-supervised acquisition of relation extraction patterns," in *Workshop on Information Extraction beyond the Document*, pp. 29–35, 2006.

[26] E. Alfonseca, M. Ruiz-Casado, M. Okumura, and P. Castells, "Towards large-scale non-taxonomic relation extraction: Estimating the precision of rote extractors," in *Proceedings of the 2nd Workshop on Ontology Learning and Population: Bridging the Gap between Text and Knowledge*, pp. 49–56, 2006.

[27] Y. Freund and R. E. Schapire, "Large margin classification using the perceptron algorithm," in *COLT' 98: Proceedings of the 11th Annual Conference on Computational Learning Theory*, (New York, NY, USA), pp. 209–217, ACM, 1998.

[28] D. Zelenko, C. Aone, and A. Richardella, "Kernel methods for relation extraction," *Journal of Machine Learning Research*, vol. 3, pp. 1083–1106, 2003.

[29] S. Abney, "Parsing by chunks," in *Principle-Based Parsing* (R. Berwick, S. Abney, and C. Tenny, eds.), pp. 257–278, Boston: Kluwer Academic Publishers, 1991.

[30] R. Bunescu and R. Mooney, "Subsequence kernels for relation extraction," *Advances in Neural Information Processing Systems*, vol. 18, p. 171, 2006.

[31] R. C. Bunescu and R. J. Mooney, "Learning to extract relations from the web using minimal supervision," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL'07)*, 2007.

[32] NIST, "The ACE 2004 evaluation plan." http://www.itl.nist.gov/iad/mig/tests/ace/ace04/doc/ace04-evalplan-v7.pdf, 2004.

[33] R. Grishman, D. Westbrook, and A. Meyers, "NYU's English ACE 2005 system description," Tech. Rep., New York University, 2005.

328

[34] D. Ahn, "The stages of event extraction," in *Proceedings of the Workshop on Annotating and Reasoning about Time and Events (ARTE '06)*, pp. 1–8, 2006.

[35] W. Daelemans, J. Zavrel, K. van Der Sloot, and A. van Den Bosch, "TiMBL: Tilburg memory-based learner, version 5.1," Tech. Rep., University of Tilburg, 2004.

[36] H. Daumé III, "Notes on CG and LM-BFGS optimization of logistic regression." Paper available at http://pub.hal3.name#daume04cg-bfgs; implementation available at http://hal3.name/megam/, August 2004.

[37] L. Ramshaw and M. Marcus, "Exploring the statistical derivation of transformational rule sequences for part-of-speech tagging," in *The Balancing Act: Proceedings of the ACL Workshop on Combining Symbolic and Statistical Approaches to Language*, pp. 128–135, 1994.

[38] E. Sang and J. Veenstra, "Representing text chunks," in *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 173–179, 1999.

[39] L. Ramshaw, E. Boschee, M. Freedman, J. MacBride, R. Weischedel, and A. Zamanian, *Handbook of Natural Language Processing and Machine Translation*. New York: Springer, 2011.

[40] M. Collins and N. Duffy, "New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 263–270, 2002.

[41] H. Ji and R. Grishman, "Refining Event Extraction through Cross-document Inference," in *Proceedings of ACL-08: HLT*, pp. 254–262, 2008.

[42] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pp. 189–196, 1995.

[43] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft, "INDRI: A language-model-based search engine for complex queries," in *Proceedings of the International Conference on Intelligent Analysis*, 2005.

[44] S. Liao and R. Grishman, "Using document-level cross-event inference to improve event extraction," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 789–797, 2010.

[45] D. M. Bikel and V. Castelli, "Event matching using the transitive closure of dependency relations," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, HLT '08, pp. 145–148, 2008.

[46] M. Palmer, P. Kingsbury, and D. Gildea, "The proposition bank: An annotated corpus of semantic roles," *Computational Linguistics*, vol. 31, 2005.

[47] A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman, "Annotating noun argument structure for NomBank," in *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC-2004)*, 2004.

机器翻译

Philipp Koehn

机器翻译是自然语言处理领域的圣杯之一，其任务很明确：在保留语义的情况下，把某种语言的文字转换成另一种语言。机器翻译模仿人类的一项活动，像业余的和专业的双语人士的日常工作，但是同时它又显得困难重重，因此大多数的研究者并不期望近期内能达到人类的翻译质量，机器翻译的目标很中肯：生成足够好的或者说可用的翻译。

近年来，随着万维网和数据驱动方法的出现，机器翻译的研究有了一些新的动向，并吸引了更多的关注，越来越多的研究机构已经针对这个问题展开了研究。现在任何人都可以通过浏览 Google Translate 和 Systran's Babelfish[⊖] 等著名网站了解机器翻译。

机器翻译研究突出表现在以下两个方面：机器翻译系统已经达到了能为大量的人提供有用服务的程度；同时明显较低的准确率也表明还有大量的研究工作要做，也许不可能达到完美翻译的程度，但可达到更高的翻译质量。

10.1 机器翻译现状

机器翻译研究起源于 20 世纪 40 年代，但最近具有深远意义的变化可以追溯到 1988 年。那一年，一群 IBM 的研究人员从根本上改变了机器翻译的研究方法。传统的翻译系统需要大量的语言学家定义转换规则和词典，这项工作是长期的且非常辛苦，一种用于语言翻译的统计方法可以减轻这种需求。替代的方法是：与传统方法不同，该方法需要大量已翻译的包含成千上万个词汇的文本语料。同时，需要一个巧妙的统计模型，该模型有助于学习翻译规则，并且为寻找给定输入句子的最好翻译的解码算法提供依据。

最近 20 年来，当时 IBM 提出的相当简单的模型（10.3 节将详细讨论）已经发展为基于短语的模型（phrase-based model）（参见 10.4 节）和基于树的模型（tree-based model）（参见 10.5 节）。

目前机器翻译研究最主要的几个方向包括：

- 开发能更近似地从语言学的角度理解语言的模型；
- 新的机器学习方法用于从数据中学习翻译规则的估计问题；
- 尝试利用各种不同类型的数据源，这些数据源通常不属于期望的领域，甚至根本不是常用的句对句的翻译。

机器翻译被集成到各种应用中：跨语言的信息抽取、语言翻译和辅助翻译工具等。

这一章主要讨论现代机器翻译系统的基本技术和方法。我们首先确定翻译的目标是什么以及如何评价翻译质量获得了提高。

10.2 机器翻译评测

机器翻译被定义为：在保留意义（注意词“意义”）的情况下，把一种语言的文本转

⊖ http://translate.google.com/ 和 http://babelfish.yahoo.com/。

换为另一种语言。虽然意义也许可以激发哲学家的思想，但却是工程人员的噩梦。意义是什么？如何才能度量意义？怎么才能知道两个词、短语或者句子表达相同的意义？如果它们的意义是相似的，那么相似的程度如何？

事实上，关于意义的问题，几乎每个该领域的研究组织对机器翻译的评估都有自己的准则[2，3，4]，这也证明了意义实际上不是一个微不足道的问题。图 10-1 显示了 10 个不同的翻译人员把一个中文句子翻译成英语句子的结果。即使是一个如此简短的句子，每个翻译人员都给出了不相同的结果。这种不一致性并不是中文的性质——其他语言也存在同样的现象。

这个 机场 的 安全 工作 由 以色列 方面 负责 。
Israeli officials are responsible for airport security.
Israel is in charge of the security at this airport.
The security work for this airport is the responsibility of the Israel government.
Israeli side was in charge of the security of this airport.
Israel is responsible for the airport's security.
Israel is responsible for safety work at this airport.
Israel presides over the security of the airport.
Israel took charge of the airport security.
The safety of this airport is taken charge of by Israel
This airport's security is the responsibility of the Israeli security officials.

图 10-1　同一个中文句子来自不同翻译人员的 10 种不同的英语译文（NIST 2001 评测集中的一个典型例子）

因此，如果用机器翻译系统翻译一个中文句子，得到的结果即使是一个很完美的翻译，也很可能不匹配任何一个人为给出的翻译结果。那么，如何才能知道这是一个正确的翻译呢？

因为不能期望翻译的结果匹配任何一个参考译文，所以需要有某种方法来比较机器翻译系统的输出结果的意义与原文的意义，或者更常用的是，和与人为参考译文的意义作比较。

10.2.1　人工评测

我们并不相信计算机可以处理好意义的问题，所以通常把这个任务交给人工评测。给定原文和机器翻译系统的输出结果，让评测人员来判断是否是一个正确的翻译。

图 10-2 显示了在最近一次研究中，4 个不同的评测人员对一个法语句子的英语翻译的评测结果[5]。虽然对有些翻译结果的评测结果是相同的，但大部分情况下是不同的。所
332 以，即使是翻译结果的评测这样一项简单的任务，也没有明确的答案。

correct	Sans se démonter, il s'est montré concis et précis.
1/3	Without fail, he has been concise and accurate.
4/0	Without getting flustered, he showed himself to be concise and precise.
4/0	Without falling apart, he has shown himself to be concise and accurate.
1/3	Unswayable, he has shown himself to be concise and to the point.
0/4	Without showing off, he showed himself to be concise and precise.
1/3	Without dismantling himself, he presented himself consistent and precise.
2/2	He showed himself concise and precise.
3/1	Nothing daunted, he has been concise and accurate.
3/1	Without losing face, he remained focused and specific.
3/1	Without becoming flustered, he showed himself concise and precise.

图 10-2　人类对译文的评测结果。当译文是正确的，4 个评价结果常常并不相同。例如，对法文句子的第一个译文，一个评测人员认为正确而其他三个认为错误

这是问题吗？不是的，以统计机器翻译中采用的概率论的观点来看，任何一个原文并没有确定的译文，只能说某些译文相比其他译文正确的可能性更高。对于每个译文，可能的评测结果表现为一个概率分布，如果有足够的样本，其统计结果将收敛于真实的概率分布，因此评测是有效的。在意义世界里，并没有绝对的正确或绝对的错误，总是有人能找出译文的缺点。

实际上，机器翻译系统会生成有些错误的译文，特别是对于一个有30个词的长句，不能期望得到没有任何瑕疵的译文。此外，我们对绝对的评价并不感兴趣（有多少句子的翻译是正确的呢？），而关注系统的相对评价（系统A是否比系统B更好？）。因此，通常情况下，不是说某个译文是否正确，而是说某个译文是否比另外一个更好。

如图10-3所示，一个人造的例子，5个翻译系统给出5个不同的译文。每个译文都有不同的错误：漏译了一个词，错译了一个词，多加了一个词 *not*，错误的标点符号以及拼写错误。哪个译文更好？ 333

Reference:	Israeli officials are responsible for airport security.
System A:	Israeli officials are responsible for security.
System B:	Israeli officials are responsible for rail security.
System C:	Israeli officials are not responsible for airport security.
System D:	Israeli officials are responsible. For airport security.
System E:	Israeli officials are responsible for aeport sequrity.

图10-3 5个不同的有错误的译文，如何比较它们

再次重申，译文的评测不是一项简单的任务。人工评测会有不同的偏好，有些评测者也许会注重标点符号[6]，而其他评测者完全不在乎。添加一个简单的功能词会带来多坏的影响？如果这个词是 *not*，情况又如何？

也许可以避免简单地回答译文是否正确，而是使用更加粒度的衡量标准。译文是否流利——也就是说，译文是否符合目标语言的语法？译文是否忠实——也就是说，在抛开语法的情况下，译文是否传达与原文相同的意思？即便使用上述标准，不同的评测人员也会有不同的偏好。

也许我们过虑了。我们为人工评测者设计了一个很不自然的任务。除了为考试结果打分的语言老师外，没有人会看译文并孤立地评估其质量。人类用译文是为了满足获取信息的需要，如果某个外文文本的译文正是他们在寻找的答案，那么译文就是正确的。

为了真实地评价机器翻译的质量，需要把译文放在能够使用它们的环境下考察。最近有些工作尝试建立基于任务（task-based）的评价方法。例如，给评测人员一个译文，然后问一些内容相关的问题。如果能够回答这些问题，那么译文就是正确的[7]。在另一个相似的方法中，可以在不给出原文的情况下，要求评测人员编辑译文从而得到流利的译文；然后，通过检查编辑后的译文是否正确来判断他对译文的理解[3]。

10.2.2 自动评测

机器翻译系统的开发过程中需要频繁的评测——太频繁以致人工评测的代价非常高。在机器翻译的研究中替代的方法是建立一种被广泛接受的机器自动评测标准。事实上，学术论文中关于机器翻译质量的提高很少包括人工评测的结果，而几乎都是基于当前最流行的自动评测标准，BLEU。

能够期望计算标准分值的计算机程序可靠地评价机器翻译的质量吗？如果计算机程序能够判断一个译文是否正确，那为什么不能首先就产生正确的译文呢？实际上，自动评测

采用一种有窍门的回避方法。

窍门就是评测中不仅使用原文和系统给出的译文，而且使用一个或多个由可靠的翻译人员给出的参考译文。前面已经详细地讨论了人或机器翻译系统都可能得出正确的、但是与已有参考译文不同的译文，因此，这就是使用回避方法的地方，也是争论不休的地方：如果机器译文跟已有的参考译文是相似的，那么就很有可能是正确的。虽然很容易就能找到一个简单的例句推翻这种观点，但自动评测事实上是基于包括几百条、甚至几千条句子的测试集的，在大规模的测试集上，与参考译文更相似的译文可以认为是更好的。

机器翻译自动评测标准的研究人员不仅支持此观点，还通过进行相关性研究证实此观点：对译文按自动评测标准排序的结果几乎与人工评测的结果一致。甚至举办对评测标准的评测活动，通过比较与人工评测结果的相关性，不同评测标准的开发人员展开竞争[2, 3, 4]。

在机器翻译系统的开发过程中，已经形成了一套测试翻译质量的体系。首先，选择测试集，让翻译人员给出一个或多个参考译文；然后，运行机器翻译系统并度量输出结果与参考译文的相似度；最后，调整机器翻译系统并在相同的测试集上再次运行，再次度量相似度，判断翻译质量是否有提高。

现在的任务就是如何度量机器翻译结果与参考译文之间的相似度。这个词是和“意义”类似的可怕的词汇之一，但是我们将从简单的方法开始。

10.2.3　WER、BLEU、METEOR 等

语言是由单词构成的，如果两个句子有很多共同的词，那么可以说它们是相似的。因此，比较机器翻译的输出结果与参考译文的时候，可以统计 1）匹配数（match）：在输出结果和参考译文相同单词的个数；2）插入数（insertion）：仅在输出结果中出现的单词的个数；3）删除数（deletion）：仅在参考译文中出现的单词的个数。

给定上面三个计数结果，可以计算许多指标：

$$\text{精确率} = \frac{\text{正确数}}{\text{匹配数} + \text{插入数}} \tag{10.1}$$

$$\text{召回率} = \frac{\text{正确数}}{\text{匹配数} + \text{删除数}} \tag{10.2}$$

$$\text{PER} = 1 - \frac{\text{匹配数}}{\text{正确数} + \max(\text{插入数}, \text{删除数})} \tag{10.3}$$

$$\text{F 值} = \frac{2 \times \text{精确率} \times \text{召回率}}{\text{精确率} + \text{召回率}} \tag{10.4}$$

$$\text{加权 F 值} = \frac{(1+\alpha) \times \text{精确率} \times \text{召回率}}{\alpha \times \text{精确率} + \text{召回率}} \tag{10.5}$$

上面这些指标是过去这些年提出的机器翻译评测标准的基础。精确率和召回率哪个更重要一直存在争议，这也关系到如何惩罚太短或太长译文的问题。位置无关错误率（Position-independent Error Rate，PER），是最早提出的评测标准之一。

在机器翻译输出和参考译文之间进行单词匹配这种简单的方法有很多改进的措施。

第一种改进措施是利用多个参考译文（multiple reference translation）。考虑到译文可以有一定的偏差，那么仅使用一个参考译文作为最佳标准可能是太强的约束。如果使用多个参考译文，那么正确的机器翻译输出与其中一个参考译文具有更高相似度的概率就会提高。这一措施将减少正确译文得分较低的情况。如何在评测标准中考虑多个参考译文，可以总是选择与任何参考译文相比得分最高的，也可以采用更复杂的方式。例如，只要输出

结果中的某个单词与任意参考译文中某个单词匹配，我们就认为匹配。

第二种改进措施是不仅匹配单词，也可以匹配多个单词组成的 n 元组，这种方法尝试考虑单词的顺序。不能期望输出结果与参考译文中的所有匹配单词的顺序相同，但是如果输出结果与参考译文中的多个相邻单词是按序匹配的，那么这种情况当然更好。

以上两种改进措施是计算 BLEU 值[8]的基础。BLEU 值是在机器翻译领域最常用的自动评测标准，非常值得详细讨论，其正式定义为：

$$\text{BLEU} = brevity\text{-}penalty \times \exp\left(\frac{1}{4}\sum_{i=1}^{4}\log\ precision_i\right)$$

$$brevity\ penalty = \min\left(1, \frac{output\text{-}length}{reference\text{-}length}\right) \tag{10.6}$$

BLEU 实质上是 n 元组精确率的几何平均数，通常使用长度为 1~4 的 n 元组（$precision_i$ 表示长度为 i 的 n 元组的精确率）。因为是一种基于精确率的度量标准，所以有必要避免选择太短的译文。这个问题通过引入长度惩罚因子解决，长度惩罚因子仅当输出译文比参考译文的长度短的情况下才起作用。使用多个参考译文的时候允许输出译文中的 n 元组与任何一个参考译文匹配。如果某个 n 元组在输出译文中出现多次，那么出现次数必须等于该 n 元组在某个参考译文中出现的最大次数才会被认为是匹配成功。在多个参考译文的 BLEU 指标中，与输出译文长度最接近的那个参考译文的长度就被确定为参考译文的长度。

BLEU 值是基于整个文档或测试集计算的，而不针对单个句子。实际上，对于单个句子而言其并不是一个较好的衡量标准，因为句子级的四元组的精确率经常为 0，或者四元组的匹配结果对最终结果有太强的影响。当把 BLEU 用于句子级的评测时，精确率通常在实际匹配次数的基础上加 1 进行平滑。

自 2002 年 BLEU 值被提出，已经有很多改进的方法。一种方法是不对 n 元组基于表层形式进行严格匹配，而至少把那些出自同一个原形而仅是形态不相同的单词看作是部分匹配的。也可以使用像 WordNet [9] 那样的资源，对同义词进行匹配。一种最近比较被看好的评测标准是 METEOR（Metric for Evaluation of Translation with Explicit Ordering）[10]，支持上面提到的匹配方法，同时更强调召回率而不是精确率。

一种较早的想法是不仅把句子看作单词或者 n 元组的集合，而且显式地计算输出译文和参考译文之间的词对齐关系。词错误率（Word-Error Rate，WER）是一种来源于语音识别的评测标准，检查词对齐关系，并且不允许句子之间单词的位置变化。因为存在许多 236
单词的顺序发生变化但意义不变的情况，WER 已经被改进为允许词序的变化，但通过附加错误（类似于插入数和删除数）惩罚这种现象。TER 被称为翻译错误率或翻译编辑率。在允许移动的情况下，计算输出译文和参考译文进行词对齐的最小代价。不幸的是，找到最小代价的对齐在计算上是很复杂的，因此这种评测指标在实际中计算起来非常慢，并且通常情况下只能粗略地计算。

最后，我们已经具备了所有的要素，可以把机器翻译的评价问题看作一个机器学习问题。多年以来，评测活动已经构建了机器译文及其人工评测结果的训练语料，从而有了一个明确定义的目标：优化自动评测指标与人工评测结果之间的关联。因此，可以在机器学习方法中使用任意特征，例如，近年来有些研究人员利用了诸如句法关系和语义角色这样的语言学特征。

10.3　词对齐

统计机器翻译的思想是从一个句子对齐的双语平行语料中学习翻译规则，首要的工作是从语料库中抽取单词的翻译。找到单词的翻译是建立词对齐关系的前提，而建立词对齐关系是任何统计机器翻译模型的基本步骤之一。

10.3.1　共现

假设已经有了一个句子对齐的平行语料库，语料库中外文句子 f 与它的英语翻译 e 成对出现。这样的语料库可以从互联网上获取（例如，欧洲议会语料[⊖]或者语言数据联盟（Linguistic Data Consortium，LDC）[⊜]）或者从翻译机构的翻译记忆库中收集。原始的语料需要进行基本的预处理，典型的如词的切分（分离标点符号），数据整理（删掉非常长的句子或者相对长度不匹配的句子），删除大小写（例如，把所有单词变为小写），语料预处理完成后就可以开始下一步工作。

然后我们需要从语料中学习一种语言中的单词能被翻译成另一种语言中的哪些词。以词汇化概率分布 $t(e \mid f)$ 的形式表示词之间的对应关系，$t(e \mid f)$ 表示外语单词 f 被翻译为英语单词 e 的概率。例如，对于德文单词 *Haus*，期望学习到以下的关系：

$$t(e \mid Haus) = \begin{cases} 0.8 & 若\ e = house \\ 0.16 & 若\ e = building \\ 0.02 & 若\ e = \text{home} \\ 0.015 & 若\ e = household \\ 0.005 & 若\ e = shell \end{cases} \tag{10.7}$$

正如统计机器翻译这个名字所示，需要从统计数据中学习一个模型，这些统计数据就 337
是双语语料库中单词出现的频次。遍历语料库中的所有包含外语单词 f（如 *Haus*）的双语句对，可以统计出其对应哪些英语单词及对应的次数。基于这些统计结果，可以估计条件概率分布：

$$\hat{t}(e \mid f) = \frac{count(f,e)}{\sum_{e'} count(f,e')} \tag{10.8}$$

在统计计数时必须小心谨慎。比如说，考虑一个包含德语单词 f 的句子 f，在对应的英语句子 e 中有 5 个单词。能把单词 f 与 5 个英语单词 $e \in$ e 的重现关系都当作一次计数吗？

如果这样计数的话，对于长句和短句将导致不一样的计数结果。在一个有 5 个英语单词的句对中，对外语单词 f 计数的结果是 5 次；但是如果英语句子的长度是 10，那么计数的结果就是 10。实际上，外语单词 f 的对应翻译在每个句子中只出现一次。

因此，替代的方法是使用分数计数，如果英语句子有 5 个单词，既然不知道哪个单词是外语单词 f 的翻译，那么对每个词的计数就是 1/5。

这种计数方法的效果怎么样？直觉上，一个外语单词 f 与其常用的翻译 e 之间的共现频率应该比较高，所以可以期望估计出相对高的 $\hat{t}(e \mid f)$。但是，在英语中几乎每个句子的结尾都是句号，那么句号与外文单词 f 的共现概率就比任何 f 的真实翻译都高。能断定每个外语单词最可能的翻译真的是句号吗？

⊖ http://www.statmt.org/europarl/。
⊜ http://www.ldc.upenn.edu/。

有的地方肯定出错了。在统计中使用单词 f 出现的次数而不是 e 出现的次数归一化 (f, e) 的共现次数。现在概率估计也可以使用其他的统计方法，例如，通过互信息的方法。实际上，在文献中若干种这样的统计方法已经用于共现的统计。

10.3.2 IBM 模型 1

IBM 模型 1 是第一个统计机器翻译模型，使用一种不同的方法解决概率估计问题。不改变条件概率模型，而是为每个句对寻找一个词对齐关系。一个外语句子 $\mathbf{f}$ 以对齐关系 a 被翻译成英语句子 $\mathbf{e}$ 的概率定义如下：

$$p(\mathbf{e}, a \mid \mathbf{f}) = \frac{1}{Z}\prod_{j=1}^{\ell_e} t(e_j \mid f_{a(j)}) \tag{10.9}$$

上述公式包括对齐函数 a，其功能非常直观：英语句子中位置为 j 的单词匹配外语句子中位置为 $a(j)$ 的单词。值得注意的是，这个公式是原始的 IBM 模型 1 的一个稍微简化的公式，在这里没有引入噪声信道模型[1]。通过归一化常数 Z 来确保 $p(\mathbf{e}, a|\mathbf{f})$ 是一个正确的概率分布。

假如已经完成了上一节所述的估计过程，并最终得到条件概率 $t(e|f)$。再次观察平行语料的第一个句对，并期望找出它们之间最可能的词对齐关系。

338

仔细观察等式（10.9）会发现要使 $p(\mathbf{e}, a|\mathbf{f})$ 最大化，就意味着分别使每个 $t(e_j|f_{a(j)})$ 最大化。换句话说，需要找出外语句子 $\mathbf{f}$ 中能最好解释单词 e 的单词 f。

正如上一节提到的，也许在你的心里依然会存在一些困惑，每个外语单词 f 都可能偏向于对应英语中的句号。但这并不是我们在此要讨论的问题，因为只有一个外语单词会对应句号，但是也需要考虑外语单词与其他英语单词的对应关系。与英语单词 *house* 对应最好的外语单词是哪一个呢？当然不是外语句号，句号与相当多的单词存在对应概率，这使情况变得非常混乱。我们期望德语单词 *Haus* 有非常大的机会对齐到英语单词 *house*。这也是令人困惑的，因为 *Haus* 可能仍然偏向于对齐到英语中的句号，但可以期望 $p(house \mid Haus)$ 大于 $p(house \mid .)$。

更进一步：如果遍历双语平行语料库中的每个句对，并为每个句对找出最可能的词对齐关系，然后仅计数对齐的词，我们可以期望更好地估计词的翻译概率分布 $t(e \mid f)$。下一节将讨论使用巧妙的 EM 算法更好地处理这个问题。

10.3.3 期望最大化

因为语料的不完备性，从平行语料库中学习单词的翻译概率将会遇到一些困难。不错，我们有平行语料库，并且所有的英语句子都能匹配它们的外语翻译，但是我们只有句子的对齐而没有词的对齐。

如果有真实的词对齐语料，那么统计计数并估计单词的翻译概率 $t(e|f)$ 是非常简单直观的。另一方面，如果能得到真实的词对齐概率 $t(e|f)$，那么很容易就可以找出每个句子对最有可能的词对齐关系。但是两种信息都没有，那么能做什么呢？

期望最大化（Expectation Maximization，EM）算法的基本思想如下：首先假设已经有词的概率分布信息 $t(e|f)$，然后就可以找出最好的词对齐关系。通过得到的词对齐关系

---

[1] 噪声信道模型利用贝叶斯规则 $\arg\max_e p(\mathbf{e}|\mathbf{f}) = p(\mathbf{e})p(\mathbf{f}|\mathbf{e})$ 将成语言模型 $p(\mathbf{e})$。因此把翻译模型方向从 $p(\mathbf{e}|\mathbf{f})$ 转换为 $p(\mathbf{f}|\mathbf{e})$。

可以重新建立一个更好的模型，有了新的模型后重复上述过程。

简单地说，期望最大化算法的流程如下：

1）初始化模型，通常从均匀分布开始。

2）将模型应用于数据：计算每个可能的词对齐关系的概率。

3）从数据中学习模型，基于词对齐计数，重新估计词的翻译概率分布。

4）重复迭代步骤 2 和 3 直到收敛。

实际上，在上一节中已经运行了简化的 EM 算法的两次迭代。在 EM 算法中，必须考虑每一种可能的对齐——而不仅仅是最有可能的对齐——基于给定句对对齐的条件概率计数（在算法的第一次迭代中，通过收集分数计数隐含地完成这一操作）。

339

考虑所有可能的词对齐关系是一项非常困难的任务：因为每个英语单词都可以对齐到任意的外语单词，因此每个句对全部可能的词对齐关系是指数级的。在 IBM 模型 1 中，采用了一种巧妙的方法在多项式的时间内准确地估计概率。但是在改进的模型中，这是不可能的。代替的方法是，通过从对齐空间中采样，从而找出最有可能的对齐关系，并仅在采样的子空间中计数。

10.3.4　对齐模型

对于词对齐和统计机器翻译来说，IBM 模型 1 是很简单的，IBM 最初的研究人员也仅仅是把模型 1 当作构建更复杂模型的中间步骤。因为有太多的对齐关系可选择，所以在词的数量比较少时，模型 1 的效果并不好。如果一个外语句子中相同的单词出现多次，那么模型 1 就不能处理这种情况，因为这些单词具有相同的概率，英语单词将对齐到多个相同词中的哪个呢？

一种扩展模型的方法是引入对齐概率组件，IBM 模型 2 提出了一个基于词的绝对位置 $a(i \mid j, l_e, l_f)$ 的模型。基于英语句子和外语句子的长度 l_e、l_f 和英语单词在句子中的位置 j，可以预测对应的外语单词的位置 i。

综合这些因素，就得到 IBM 模型 2：

$$p(\mathbf{e}, a \mid \mathbf{f}) = \frac{1}{Z} \prod_{j=1}^{l_e} t(e_j \mid f_{a(j)}) a(a(j) \mid j, l_e, l_f) \tag{10.10}$$

毕竟，通常情况下单词仅在短语内部移动，所以并不映射词的绝对位置关系，而是更趋向于使用相对于前一个词的位置关系。IBM 模型 4 和隐马尔可夫模型（HMM）都把相对对齐模型应用到词对齐中[11]。

模型的进一步扩展：虽然严格限制一个英语单词只能对齐到一个外语单词，但是一个外语单词却可以对齐到多个英语单词。对于这个问题，IBM 模型 3 引入繁衍率（fertility）的概念，并增加另一个条件概率预测一个外语单词生成多少个英语单词。

10.3.5　对称化

现在谈到了统计机器翻译中最棘手的问题。虽然目前还是经常把 IBM 的这些模型应用到词对齐中，但它们存在基本的缺陷。使 EM 算法非常有效的巧妙方法是限制是一个英语单词仅与一个外语单词对齐[⊖]。

从语言学的角度来看，这样的限制没有道理，严格的一对多对齐限制也是一种奇怪的

⊖　实际上，也允许英语单词对齐到人造的空（null）词，但不允许一个英语单词对应到多个外语单词。

不对称。那么，还能做什么呢？也许可以从两个方向运行 IBM 模型中的 EM 训练（得到一个一对多的对齐和一个多对一的对齐），然后合并两个方向的对齐关系，这个粗略的处理过程称为**对称化**（symmetrization）[12]。 340

一旦得到了两个方向的同对齐结果，就可以对对齐点求并集和交集。一种比较通用的方法是，最终的对齐关系中包括交集中所有的对齐点和并集中的某些对齐点，通常添加的是并集中与已经建立对齐关系的点相邻的对齐点，如图 10-4 所示。

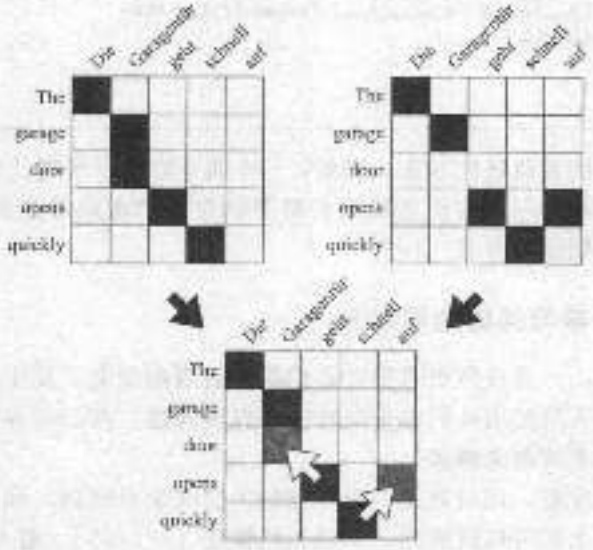


图 10-4 克服 IBM 模型的假设一对多对齐的缺陷（反方向运行时是多对一对齐）。启发式方法如“grow-diag-final”从双向运行模型得到的两种对齐结果的交集中的对齐点开始（黑色箭头所示），然后从两种对齐结果的并集中选择相邻的对齐点（白色箭头所示）

例如，“grow-diag-final”是最常用的一种方法，该方法已经被集成到开源的 Moses 系统中。实现时，增加与已经对齐的点直接或者对角相邻的点，直到收敛。在最后一步执行时，增加迄今为止还没有对齐的点。这种启发式方式的伪代码如算法 10-1 所示。

算法 10-1 启发式对称化算法伪代码，该算法从由两个 IBM 模型对齐的交集与并集所组成的点集合中挑选点，以完成对称化

输入： 句对 (e, f)，以及对齐 $e2f$ 和 $f2e$
输出： 句对齐 a

```
grow-diag-final(e2f, f2e);
邻居 = {(-1,0),(0,-1),(1,0),(0,1),(-1,-1),(-1,1),(1,-1),(1,1)}
对齐 = intersect(e2f, f2e);
GROW-DIAG();
FINAL(e2f);
FINAL(f2e);

function GROW-DIAG();
while 迭代直到没有新点可添加 do
  for all 英文词 e ∈ {e_1...e_n} do
    for all 外文词 f ∈ {f_1...f_m} do
      if e 与 f 对齐 then
        for all 邻居点          e_new, f_new do
          if {e_new 未对齐 | f_new 未对齐}
          & {e_new, f_new} ∈ union(e2f, f2e) then
```

```
14:                 添加对齐点 (e_new, f_new)
15:               end if
16:             end for
17:           end if
18:         end for
19:       end for
20: end while

21: function FINAL(a):
22: for all 英文词 e_new ∈ {e_1...e_m} do
23:   for all 外文词 f_new ∈ {f_1...f_n} do
24:     if {e_new 未对齐 | f_new 未对齐} & (e_new, f_new) ∈ union(e2f, f2e) then
25:       添加对齐点 (e_new, f_new)
26:     end if
27:   end for
28: end for
```

对称化的过程有很多改进的方法，例如，EM 训练的每一次迭代后都可以执行对称化[13]。也可以使用机器学习的方法迭代地向对齐的交集中增加对齐点[14，15]，或者迭代地从对齐的并集中删除对齐点[16]。

10.3.6　作为机器学习问题的词对齐

与评价标准一样，一旦自然语言研究者们设法适当地定义了某个问题，不久就会有一大群机器学习的研究人员使用他们偏爱的算法研究该问题。在词对齐方面也发生了同样的事情，近年来这样的事情越来越多。

341
~
342

从机器学习的角度看，词对齐是一个有趣的无监督学习问题，列出最近应用到的所有相关方法并不重要。大家可以猜想到，如感知机算法[17，18]、最大熵模型[19]、神经网络[20]、最大边界方法[21]、boosting[22，23]、支持向量机[24]、条件随机场[25，26]和 MIRA 算法（Margin Infused Relaxed Algorithm）[27]。

机器学习方法在词对齐上成功突破的关键是测试集的建立，这里是指人工标注的作为最佳标准的词对齐资料。对许多语言对存在几个那样的测试集，通常可以通过 LDC[⊖] 获取。

如何评测词对齐的质量存在一些争议，对齐错误率（Alignment Error Rate，AER）是一种较早的评价标准，但受到了严厉的批评[28]。因为词对齐多数情况下只是统计机器翻译的一个中间过程，那么最终的评价准则应该是使用某种词对齐关系能够得到怎样的翻译质量。当然，这种评测指标在计算上是非常耗时的。

10.4　基于短语的翻译模型

当前，在统计机器翻译领域占主导地位的方法是基于较短文本块（通常只包括 1～3 个单词）之间的映射关系而建立的模型。既然这些文本块不必是语言学上的短语（例如，语法分析中的成分），那么称为短语就有点误导作用。

与基于词的翻译模型相比，基于短语的模型克服了单词之间必须一一对应这样一个根本的缺点。虽然在处理实际问题的时候，基于词的模型引入了诸如繁衍率和生成空词等方法，但是后果是参数训练和解码算法都更加困难。基于短语的模型的优势还包括：可以从更多的训练语料中学习到越来越长的短语。最极端的情况下，一个句子也许可以在训练语料中找到完整的翻译。

⊖ http://www.ldc.upenn.edu/。

10.4.1 模型

基于短语的模型的优点是它相当简单，因此存在很直观的训练方法和高效的解码算法。如图 10-5 所示，输入的外语句子被切分成多个短语，每个短语一对一地对应到英语短语，并且英语短语之间可以进行调序。

现在从数学意义上给出基于短语的统计机器翻译模型的定义。首先，运用贝叶斯法则，对翻译方向进行转换，并引入语言模型 p_{LM}。对于给定的外文句子 $\mathbf{f}$，其最优的英文翻译 $\mathbf{e}_{best}$可以定义为：

$$\begin{aligned}\mathbf{e}_{best} &= \operatorname{argmax}_{\mathbf{e}} p(\mathbf{e} \mid \mathbf{f}) \\ &= \operatorname{argmax}_{\mathbf{e}} \frac{p(\mathbf{f} \mid \mathbf{e}) p_{LM}(\mathbf{e})}{p(\mathbf{f})} \\ &= \operatorname{argmax}_{\mathbf{e}} p(\mathbf{f} \mid \mathbf{e}) p_{LM}(\mathbf{e})\end{aligned} \tag{10.11}$$

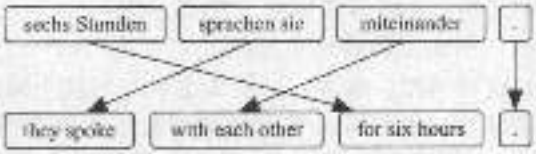


图 10-5 基于短语的机器翻译，输入的句子被切分成若干个短语（不必是语言学意义上的），然后将短语一对一地翻译成英语短语，顺序可能调整

值得注意的是可以忽略外语句子 $\mathbf{f}$ 的概率 $p(\mathbf{f})$，因为对于其所有可能的译文 $\mathbf{e}$，它是一个常量。可以进一步把条件概率 $p(\mathbf{f}|\mathbf{e})$ 分解为：

$$p(\mathbf{f} \mid \mathbf{e}) = \prod_{i=1}^{I} \phi(\overline{f}_i \mid \overline{e}_i) d(start_i - end_{i-1} - 1) \tag{10.12}$$

其中，外语句子 $\mathbf{f}$ 被切分成 I 个短语 $\overline{f}_i$，每个外语短语 $\overline{f}_i$ 被翻译成一个英语短语 $\overline{e}_i$。因为数学意义对翻译方向进行了转换，所以短语翻译概率 $\phi(\overline{f}_i \mid \overline{e}_i)$ 被建模成从英语到外语的翻译。

调序问题由**基于距离的调序模型**（distance-based reordering model）实现。参考前一个短语，判断当前短语是否需要调序。定义 $start_i$ 是翻译成第 i 个英语短语的外语输入短语中第一个词所在的位置，end_i 是该外语短语中最后一个单词所在的位置。通常情况下，这种模型不是根据语料来估计概率，而是使用一个基于相对移动距离的固定代价函数：$d(x) = \frac{1}{Z^{\alpha|x|}}$。

一些其他的组件也可以引入到模型中。典型的是，为每个产生的单词附加一个因子 ω，并在模型中引入词语惩罚 $\omega^{-|\mathbf{e}|}$，从而可调节模型以产生更长的输出或更短的输出。

10.4.2 训练

基于短语的模型的最主要的知识源是大规模的短语翻译表。短语翻译表中包括输入短语和它们可能的翻译，以及相应的频率值。

可以从词对齐的双语平行语料库中学习获得短语翻译表。给定一个词对齐和句对，就可以抽取出与词对齐一致的所有短语对。与词对齐一致指的是短语对中的单词相互对齐，但不对齐到短语对外的单词。

图 10-16 给出了一个实例。假定有词对齐点 (*opens*, *geht*)、(*opens*, *auf*) 和 (*quickly*, *schnell*)，可以抽取出短语对 (*opens quickly*, *geht schnell auf*)。

短语抽取时会有一些具体的规定，如最大短语长度（通常为5~7）、在短语的边界上是否可以包括未对齐的词（通常情况可以，但有时不可以）、是否采用分数计数、是否句子中一个源语言短语可对应到多个目标语言短语（相反方向也一样）。

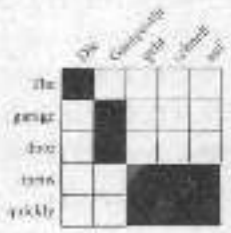


图 10-6　短语抽取：给定图中的词对齐，抽取出短语对 (*opens quickly*, *geht schnell auf*)

抽取短语对时，通过统计得到累计次数，再基于相对频率就可以直接估计短语翻译的条件概率：

$$\phi(\bar{f}\mid\bar{e})=\frac{count(\bar{e},\bar{f})}{\sum_{\bar{f}'}count(\bar{e},\bar{f}')} \tag{10.13}$$

短语翻译的条件概率估计经常会遇到数据稀疏的问题。在极端的情况下，一个英语短语 $\bar{e}$ 在语料库中只出现一次，那么唯一对应的外文短语 $\bar{f}$ 的短语翻译概率为 $\phi(\bar{f}\mid\bar{e})=1$。

有几种方法可以缓解这种情况。通常，可以加上额外的基于词汇的翻译概率的评分函数，例如 IBM 模型 1；使用 Good-Turing 平滑方法对初始计数进行折扣也是有效的[29]。

在 10.4.5 节中将继续改进模型，重新形式化为对数线性模型 (log-linear model)，这种模型可以很方便地集成额外的评分函数。现在首先考虑一个实际的问题：对一个新的、从未出现过的输入句子，如何产生它的译文。

10.4.3　解码

假设希望翻译如下的德语句子：

Sechs　Staden　sprachen　sie　miteinander.

six　hours　spoke　they　with each other.

英语句子通常以主语开头，因此在翻译成英语句子时，要先找出德语句子的主语 *sie* 并翻译为 *They*，然后找出动词 *sprachen* 并以其对应的 *spoke* 扩展翻译。从左到右地构建翻译，可以得到英文句子：

They spoke with each other for six hours.

在解码算法中，我们希望机器也从能左到右进行翻译，然而这并不简单，因为在短语翻译表中有太多的选项可以选择。如图 10-7 所示，一个真实例子的短语表选项的部分摘录（使用从欧盟「欧洲议会」语料库获取的短语翻译表），翻译时，仅当句子全部构建完成后才能计算整个翻译结果的概率。

Sechs	Stunden	sprachen	sie	miteinander	.
six	hours	it would be		with each other	.
six ,	hours ,	it would		to each other	.
6	hours of	they spoke		together	.
for six	few hours	spoke	they	with	
	time	talked	she	each other	

图 10-7　一个德语短句的翻译选项

解码算法开始时可以选择图中任意的翻译选项，随后就不能再选择已经翻译过的词或短语，但还剩下几乎同样多的选项。一种朴素的算法尝试找出翻译选项所有可能的组合，

这种算法的时间复杂度是句子长度的指数级。实际上，机器翻译的解码已经被证实是一个NP完全问题[30]。

在常用的柱搜索栈解码算法中，我们通过保存最有希望的局部翻译，并使用新的翻译选项进行扩展直到覆盖整个输入句子的方法来搜索句子可能的翻译。部分翻译（称为翻译假设，hypotheses）基于已经翻译过的外语单词的数量被组织在不同的栈中。例如，栈1保存所有已经翻译了一个外语单词的翻译假设。为了限制栈中翻译假设的数量，必须丢掉那些看上去希望不大的假设。

扩展一个栈中的翻译假设将产生新的翻译假设，并把它们存放在后续的栈中，然后继续处理下个栈。算法10-2是这种解码算法的伪代码，图10-8描述了该过程。

算法10-2 启发式栈解码的伪代码

```
输入：外文句 f = f_1 ... f_J
输出：英文翻译 e
1: 将所有空的翻译假设放到栈0
2: for all 栈 0...n − 1 do
3:   for all 栈中的翻译假设 do
4:     for all 翻译选项 do
5:       if 可以应用 then
6:         创建新的翻译假设
7:         将它放到栈中
8:         如果可能，与已存在的翻译假设重合并
9:         如果栈太大，对栈空间进行剪枝
10:      end if
11:    end for
12:  end for
13: end for
```

术语柱搜索意味着使用“光柱”的方法搜遍整个搜索空间中最有可能的那部分，但“光柱”并不是足够明亮到能探索到所有可能的路径，因此只能探索到一些可供选择的路径。

如上所述，在解码的过程中必须清理掉每个栈中希望不大的候选假设。注意当产生部分翻译后，就可以按照公式（10.13）算出短语翻译概率，基于到目前为止被选择的翻译选项计算部分翻译的得分。然后可以根据计算出的翻译得分对栈中的候选翻译排序，摒弃最差的那些。

特别地，同一栈中的翻译假设可能覆盖不同的外语单词，如果仅根据当前翻译得分做出判断，对那些先翻译句子中较难部分的候选翻译是不公平的。因此，除了考虑当前得分外，还应考虑**未来代价估计**（future cost estimate）。

还有一个非常重要的方法：**重合并**（recombination），在搜索过程中，可能会存在两条不同的解码路径导致基本相同的状态。例如，可以从单个词的短语对 *sie→they* 开始翻译，然后通过短语对 *sprachen→spoke* 扩展。但我们也可能简单地使用包括两个单词的短语对 *sprachen sie→they spoke*，两种翻译假设中，其中一个当前得分更高（根据短语翻译的代价计算），那么就可以安全地移除较差的那个翻译假设。

值得注意的是，重合并时翻译假设并不需要完全匹配，只需要保证它们对后续扩展是不可区分的。虽然两个翻译假设必须覆盖相同的输入单词（这将影响后续搜索），但只要超出了 n 元组模型的窗口，对应的输出结果就可以不相同。

346

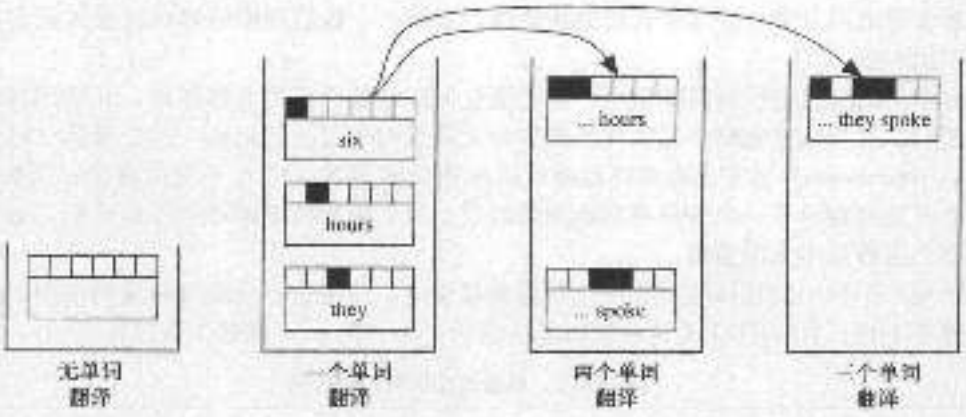


图 10-8　启发式找解码搜索

10.4.4　立方剪枝

一种常用的启发式解码的新变体称为**立方剪枝**（cube pruning），虽然它在基于短语的解码中跟立方体和剪枝都没关系，也许**有序扩展**（sorted expansion）是个更好的名字。因为产生的绝大多数翻译假设都被丢弃，立方剪枝单点扩展最有希望的那些翻译假设。为此，需要对已有的候选翻译和可用的翻译选项排序，然后用最有希望的翻译选项扩展最有希望的翻译假设，如图 10-9 所示。

假如希望使用覆盖了第一个单词的翻译选项去扩展覆盖了第一个词的翻译假设。在这个例子中有 4 个这样的翻译假设和 5 个翻译选项——事实上真正的数量会大得多，原始的朴搜索解码尝试所有 20 种可能——我们希望只是集中在某个子集上。

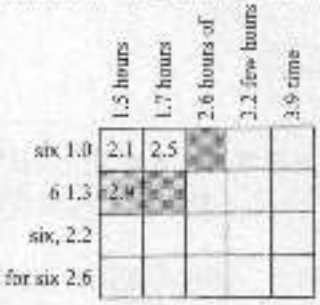


图 10-9　立方剪枝，对翻译假设（y 轴）和翻译选项（x 轴）进行排序，只扩展生成最有可能的翻译假设

最有希望的新的翻译假设是最好的已有翻译假设和最好的翻译选项的组合，所以从图的左上角开始扩展。

基于已有翻译假设和可用翻译选项的生成代价，可以产生最好的 n 个新的翻译假设。然而，新的翻译假设的代价并不是把原有假设的代价和扩展选项的代价简单相加，只有当把它们组合在一起后才能计算真正语言模型的得分，从而得到翻译假设的真正代价。

[317]

替代的办法是：根据代价函数，我们挑选最有希望的翻译假设和翻译选项来进行扩展，并总是挑选已扩展生成的最佳翻译假设的相邻未扩展翻译假设进行扩展。

在上面的例子中，最有希望的翻译假设位于图的左上角，其真实代价是 2.1；搜索相邻的选项，其代价分别是 2.5 和 2.9；然后使用代价为 2.5 的选项扩展翻译假设；依次类推。

10.4.5　对数线性模型和参数调节

前面的章节已经介绍了一些可以改进机器翻译模型的组件，如词汇化概率和词惩罚。从数学上严格构建一个包括众多组件的模型是棘手的，这些组件包括句子的翻译概率 $p(e|f)$、单词添加概率、独立性假设和同现等。因此还是放弃这种想法比较好，而把模型清楚地表

示为各种特征函数 h_i 的加权组合，根据特征的重要性分配权值 (λ_i)：

$$p(\mathbf{e}\mid\mathbf{f}) = \prod_i h_i(\mathbf{e},\mathbf{f})^{\lambda_i}$$

$$\log p(\mathbf{e}\mid\mathbf{f}) = \sum_i \lambda_i \log h_i(\mathbf{e},\mathbf{f}) \tag{10.14}$$

这些特征函数包括在 10.4.1 节介绍的基于短语的翻译模型的组件：如语言模型 $h_{\mathrm{LM}}(\mathbf{e},\mathbf{f}) = p_{\mathrm{LM}}$ 和短语翻译模型 $h_\phi(\mathbf{e},\mathbf{f}) = \sum_i \phi(f_i\mid e_i)$。

既然为特征函数 h_i 引入了权重 λ_i，那么如何设置这些权重呢？对于一个外语句子 $\mathbf{f}$，每个特征函数都与英语句子 $\mathbf{e}$ 是否是一个好的翻译有关联。通过衡量这些特征函数的重要性可以优化整体翻译的质量。

这正是结束自动评价指标讨论的时候：给定一个由外语输入句子和它们的参考译文构成的**调参集**（tuning set），基于任意给定的权重集合 (λ_i)，使用我们的模型和解码器翻译 348
这个调参集，并自动计算输出结果的 BLEU 值。然后改变权重集合并重新解码，判断 BLEU 值是否有提高。在这里，面对的是一个定义良好的多维参数优化问题，通常称为**调参或最小错误率训练**（Minimum Error Rate Training，MERT）。

因为解码是非常耗时的工作，所以采用一种快捷的方式：为每个输入的句子产生 n-best 译文，然后基于这些 n-best 译文优化权重。一种常用的方法 [31] 是一次只优化一个权重。当固定其他权重的时候，最有可能找到这个权重的最优解的。然而，这种方法被限制在一个区域内搜索，很可能陷入局部最优，因此随机化权重并重新开始是有必要的。另外，也可以重新运行解码过程，从而避免基于不能代表整个搜索空间的 n best 列表进行优化。

10.4.6 控制模型的大小

基于短语的翻译模型构建的短语翻译表远远大于双语平行语料本身的大小，这种情况并不是很直观。设想一下，长度为 n 的句子包含的短语数是 $O(n^2)$。

典型地，训练语料有数百万的句对，构建的短语翻译表通常会达到千兆字节级的规模。即使有摩尔定理，也经历了很长一段时间才使基于短语的翻译模型进入实用阶段，但是即使现在也不能在内存中保存大规模的模型。如果尝试把统计机器翻译系统应用到掌上设备，那么这种情况就变得更严重。

已经提出了很多种解决办法，从短语翻译表的有效存储到过滤和剪枝。下面了解一下这些解决办法。

上面已经提到短语翻译表比原始语料库要大得多，一种令人感兴趣的方法是根本不存储短语表，而仅存储原始语料库。当然，给定一个源语言句子，必须能够快速找到与之匹配的源语言短语（和它们的翻译），因此有人提出使用后缀数组的方法 [32]。

后缀数组是一种包含语料中所有后缀的有序列表的数据结构。可以把后缀看作是一个很长的短语，这个短语从语料的任何一个位置开始到语料的最后位置结束。所有后缀的数量与语料中单词的个数是一致的，因此有序的索引也与单词的个数保持一致。当查找输入句子的某个后缀时，可以使用索引找到任意多的匹配，然后根据（也被保存的）词对齐和语料的目标端语句在线抽取匹配的短语。

然而，如果语料的规模实在太大，就需要对保存在内存中的数据做出更多的限制。值得注意的是，对于单个句子的翻译，仅会用到短语翻译表中很小的一部分，可以不用加载整个短语表到内存中，而是过滤到只剩下需要的部分。过滤经常用在实验中，因为在实验

中会重复使用包括几千条句子的测试集。

如果想要开发一个提供在线服务的机器翻译系统，就没有时间过滤千兆级的数据，除非能够以一种非常有效的数据结构把短语翻译表存储在磁盘上，这种数据结构适合快速查找短语，如前缀树 [33]。

最后，在认真观察短语翻译表后会意识到它们中的大部分并不起作用：很长的短语对和数以千计包含句号的短语对（包括逗号的更多），被使用的可能性都比较低。那么，为
[349] 什么不清理短语翻译表呢？基于显著性的测试可以忽略掉一些短语对，如测试高于随机产生的次数（more-than-random occurrence）[34] 或者对数似然率（log likelihood ratios）[35]。在第二遍抽取短语对的阶段也许可以考虑上面要到的那些因素，这一阶段并不抽取质量差的短语对 [36]。

也许只需要抽取能够解释每一个训练句对的最短的短语对 [37]，这也是 n 元组翻译模型的基本思想 [38, 39]，它是基于短语翻译模型的一个变体。或者，通过察看一个短语对在解码过程中被使用的频繁程度或出现在最佳翻译结果中的频繁程度，从而对短语翻译表进行剪枝 [40, 41]。最后，Kotsom 等人 [42] 使用支持向量机的方法清理短语表。

10.5 基于树的翻译模型

任何有一些语言学背景的读者都会认为我们的模型是粗糙、毫无希望的。语言的最基本的概念是递归，句子是由从句构成的，从句是由动词、名词短语等组成。名词短语也可能包括从句，同样由动词和诸如此类的成分构成。事实上，所有现代的句法理论都把句子看作有层次的树结构，而不是由单词组成的串。

对于统计机器翻译的研究者来说，上面提到的任何一件都不是令人意外的事。句法树的使用——不管是使用句法分析器还是从语料库中自动学习树结构的方式——从 20 世纪 90 年代中期开始就是统计机器翻译研究领域中一直被关注的方向。然而，直到最近基于树的翻译模型在正面的交锋中都没有胜过更简单的基于短语的翻译模型。

原因之一是基于树结构的操作更复杂，因此需要计算上更耗时的学习方法，同时也使得解码过程中的搜索更困难。约束源语言句法树和目标语言句法树具有某种形式的同构关系（如仅允许子节点之间的调序并且没有重大的重构）可以简化模型，但这种约束也被证实是太强的约束。

基于句法的方法的另一个问题是：首先假设句法树是正确的，但到目前为止还没有可用的足够好的句法分析器。

当前的基于树的方法借鉴基于短语模型的成功之处，可以看作是基于短语方法的扩展。

10.5.1 层次短语翻译模型

正如定义所示，基于短语的模型的一个限制是不允许有不连续的短语。例如，可能期望映射如下的英语和法语对：

$$does\ not\ X \rightarrow ne\ X\ pas$$

然而基于同步上下文无关文法可以表达上面的映射关系，这种方法区分终结符（单词）和非终结符（X）。一条文法规则可能包括多个非终结符：

$$X_1\ of\ X_2 \rightarrow X_2 X_1$$

我们已讨论过基于短语的模型，可以把上面的规则理解为在已经被抽取了子短语对的
[350] 短语之间映射。如果允许这样的规则，就能更好地解释某些特定的调序现象，如能够 of

的角色和不连续的短语。

从同对齐的双语平行语料库中抽取层次短语对（hierarchical phrase pair）[43] 的方法是很直观的。除了所有原始的词汇化短语对外，还必须检查每个短语对中是否有子短语对，并把它们替换为非终结符。那么，就可以把层次短语对添加到短语翻译表中。请看图 10-10 中的例子。

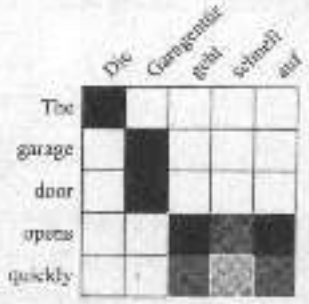


图 10-10 学习层次短语翻译规则。从短语对开始（*geht schnell auf*，*opens quickly*），抽取子短语对（*schnell*，*quickly*）就可以得到翻译规则（*geht X auf*，*opens X*）

抽取子短语对可能会迅速增大短语对的数量，因此必须引入一些合理的约束，如短语对中最少要包括一个单词、短语最多能包括的单词个数等。

很明显，增加层次短语对是有利的，但是会使在 10.4.3 节中介绍的解码算法失效，这种算法要求从左到右构建翻译结果。当增加了诸如 *ne X pas* 之类的短语后，如何从左到右构建一个句子？

有一个源自句法文法的很直接的解决办法：这是一个句法分析问题，必须使用句法分析算法，如线图分析。

10.5.2 线图解码

线图解码不是从左到右进行解码，而是从底向上进行解码。首先找出每个单词的翻译，然后找出跨度为 2 的短语的翻译，再找出跨度为 3 的短语的翻译，依次类推直到覆盖整个句子，如图 10-11 所示。

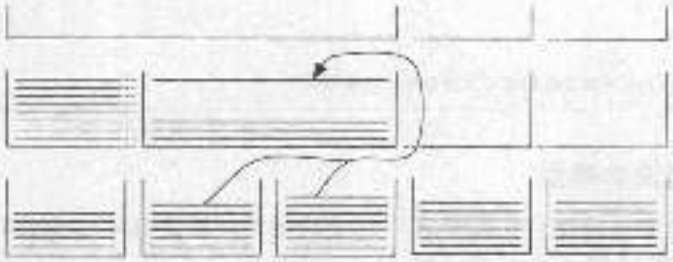


图 10-11 基于树的模型的解码，一个栈代表一个输入单词（底层行），更高层的栈代表连续的块

例如，把下面的句子翻译为英语：

Je ne parle pas anglais.

首先可以使用许多传统的短语翻译规则，这些规则构成线图的条目：

I *speak* *English*

je *ne* *parle* *pas* *anglais*

然后使用层次短语规则（这里 *X* 匹配 *speak*）：

ne X pas→*do not X*

这样就可以添加线图条目 *do not speak*：

do not speak

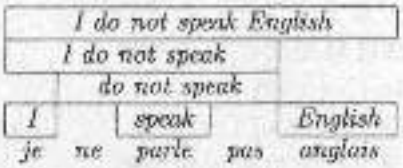
I *speak* *English*

je *ne* *parle* *pas* *anglais*

最后，使用黏合（glue）规则：

$$X_1 X_2 \rightarrow X_1 X_2$$

两次使用上面的规则就可以得到完整的输出：



算法 10-3 给出了线图解码算法的大概步骤。实际上，需要许多改进的措施，如避免遍历所有可能的序列（从第 4 行开始）和所有规则（从第 5 行开始）的循环，全部遍历计算代价太高。当为一个跨度增加新的线图条目时，需要采用能有效地搜索底层线图条目和可用规则的方法，例如，使用 Earley 句法分析。

算法 10-3　线图解码算法的核心代码

```
输入：外文句子 f = f_1, ... f_J
输出：英文翻译 e
1: for span 长度 l = 1 ~ J_f do
2:   for start = 0 ~ J_f - l do                    // 跨度的开始
3:     end = start + l
4:     for all 由 span [start,end] 中短语条目和前后构成的序列 s do
5:       for all 规则 r do
6:         if 规则 r 适用于该序列 s then
7:           建立新的线图条目 c
8:           将线图条目 c 加入图中
9:         end if
10:      end for
11:    end for
12:  end for
13: end for
14: return        [0,J_f] 中最长线图条目所对应的英文翻译 e
```

10.5.3　基于句法的模型

在发展为层次短语模型后，构建基于句法的模型并不算很大的跨越，这种模型使用句子构成成分的真实标记如 VP 和 NP。除了词对齐的平行双语语料库外，还需要源语言、目标语言或者双方进行句法标注，如图 10-12 所示。

252

源语言端的句法作为选择规则的约束，目标语言端的句法要求翻译输出结果的句法分析结果是一棵树，因此除了通过 n 元组语言模型保证输出结果的流畅外，还保证了输出结果具有良好的句法结构。

举个例子，法语动词的否定形式如下所示：

$$\text{VP}_1 ne\ \text{V}\ pas \rightarrow \text{VP}_1 do\ not\ \text{V}$$

增加句法标记时应保持几分警惕，

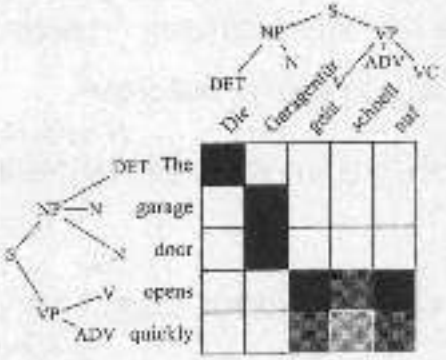


图 10-12　抽取句法翻译规则：从图 10-12 中层次短语对的例子，获取短语对 (*geht x auf*, *opens x*)，句法标记用来区别各种非终结符，得到规则 VP_1 *geht* ADV *auf* → VP_1 *opens* ADV

因为基于短语的模型中的短语不必匹配句法树中的成分。既然每个线图条目都需要一个成分标记，那么有两种选择：1）目标端的短语必须是单一的句法成分；2）创造一个人工标记。

举个例子，在短语模型中，可能有以下的规则：

$$der\ große \rightarrow the\ big$$

当在目标端增加句法标记的时候，选择1必须扩展规则使其覆盖包括 *the big* 的整个名词短语：

$$der\ große\ X \rightarrow \mathrm{NP}:the\ big\ \mathrm{N}$$

选择2必须创建新的标记，如：

$$der\ große \rightarrow \mathrm{DT}+\mathrm{J}:the\ big$$

两种选择都有缺点：选择1放弃了许多作为潜在规则的短语对，限制了从平行语料库中抽取到的知识。选择2导致了非终结符的快速增长，增加了解码的难度。 353

在基于句法的模型中使用的第三种选择是二叉化句法树，从而使得对符合句法成分的短语的限制不是那么严格。

10.6 语言学挑战

到目前为止，并没有对翻译本身的性质给予足够的关注。大多数读者，特别是学习过第二种语言的读者，对于是什么导致翻译困难会有一个直观的理解：在源语言中的单词有多个不同的意思，因此对应不同的翻译结果；两种语言的词序也可能不同；在句子中词之间的关系也可能以不同的方式表示——形态学标记、功能词或词序。

所有这些问题都是统计机器翻译系统应该解决的问题。虽然我们声称书中介绍的机器翻译方法是语言无关的，但实际上如果两种语言之间有几乎相同的词序、相似的概念和隐喻，并且目标端语言形态学变化简单，那么翻译的效果会更好。例如，机器翻译系统把法语翻译成英语时效果较好，但把中文翻译成土耳其语时效果就比较差。

10.6.1 译词选择

计算语言学中一个常见的问题是词义消歧，像 *interest* 和 *bank* 这样的单词有多个意思。这个问题在机器翻译中就表现为译词选择，也就是说在翻译成德语时，是把 *bank* 翻译成 *Bank* 还是 *Ufer*，是选择与财富相关还是与河流相关的意思。

词义消歧的研究表明：局部上下文（相邻的单词或者词性标记）、在更大范围内的功能词、词的语义角色和句法上相关的词都能够指示词的意义。

实际上，使用 n 元语言模型已经能够有效地捕捉到局部上下文的信息，这些信息非常 354
有利于翻译时选择正确的词汇。词的先验概率也是有用的：*bank* 表示财富相关的词义比表示为与河流相关的词义更频繁。因此，统计机器翻译系统在译词选择上处理得相当好，显然要好于传统的基于规则的系统。

然而，近年来许多研究人员针对统计机器翻译中的词义消歧问题展开研究，通过在模型中集成一些前面提到的额外的特征，取得了一定的效果。把传统的条件概率分布——基于单词的或基于短语的——转换为更复杂的分类器是比较直观的方法。

一种常用的方法是使用最大熵方法集成源语言句子中的任意特征。集成目标语言端句子的特征是比较难的，因为在柱搜索解码算法中句子通常都处于分裂的状态。如果假设一个单词的翻译取决于目标端句子的第一个单词，那么就不能重合并第一个单词不同的翻译假设。

10.6.2　形态学

目前已经介绍的翻译模型都是基于单词的表层形式的。例如，它们认为单数形式的 *house* 与复数形式的 *houses* 是没有关联的。因为统计机器翻译中的绝大多数研究都把英语作为目标语言，而英语的形态变化又相对简单，所以形态学一直不被认为是应该优先考虑的问题。诚然，把 *house* 和 *houses* 当作两个完全不同的单词会丢失一些泛化信息，但是这样可以使模型比较简单，或许能够较好地区分单数形式和复数形式的翻译。

然而，当把输入句子翻译成形态丰富的语言的时候，如土耳其语、匈牙利语、捷克语和德语，形态学就变成了一个非常重要的问题。首先应该想到的就是丰富的形态学将导致更多的词汇量，因此在模型估计时会有严重的数据稀疏问题。

其次，当翻译成形态丰富的语言时，从局部上下文通常很难区分应该选择哪种形态变体。例如，当把 *the man* 翻译成德语时，可以选择 *der Mann*、*des Mannes*、*dem Manne* 和 *den Mann* 作为它的翻译。哪个是正确的翻译取决于这个名词短语与其句法中心词的关系，例如，是主语还是宾语？

因子化翻译模型（factored translation model）[44] 提出不是把单词看作简单的词元，而把它看作各种因子组成的矢量，如原形、词性标记、性和数等。在模型中包含这样的附加信息有两个好处：首先，在原形之间而不是词的表层形式之间翻译有利于泛化；其次，丰富了模型能够利用的信息，也就是说，可以基于词性标记调序，或基于形态学标记检查语法的一致性。

在基于短语的模型中增加因子化的表示，丰富了源语言端的输入信息，从而在目标端可以更好地选择形态，提高翻译输出的语法一致性，较好地翻译少见的形态学变化。这种方法同时也带来了风险，因为假设这些过程之间是相互独立的，从而把短语翻译分解成几个独立的映射过程。如果形态丰富的短语在语料库中出现的频率较高，那么就可以较好地翻译它们，此时把这种短语的翻译分解为多个更细粒度的步骤只会带来坏处　　就像一次
[355] 性翻译一个长的短语（如果可能的话）的效果比逐词翻译的效果更好。

10.6.3　词序

句子由一个或多个从句组成，每个从句以动词为中心，同时包括动词的论元和修饰语，用来描述一个动作。像英语这样的语言利用语序来确定句子中的实体哪个是主语、哪个是宾语，以及它们的角色是什么。

英语是一种 SVO 语序的语言（*English is an SVO language*），意味着一个从句通常是以主语（*English*）开头、随后是动词（*is*）和若干宾语（*an SVO language*）。其他的语言或许有不同的词序规定，如 VSO 或 SOV。这为机器翻译提出了一个直接的问题。单词在翻译成目标语言时需要重新排列。

调序是因为句法的不同而产生的，这个见解是 10.5 节讨论的基于树的翻译模型的主要驱动力之一。如果能够获得输入句子的句法树，或在翻译时构建输出句子的句法树，那么表层的任意移动（例如，一个单词向左移动 3 个位置）体现在句法树上仅仅是子节点的重排序。

一般来说，使用基于树的模型是相当复杂的，因此提出了一些简化的方法用来在统计机器翻译中集成句法树信息。一种思想是，在实际翻译前对输入句子**预排序**（pre reorder）。预排序的目的是：在保留所有单词的情况下，按照输出译文的期望顺序对输入句子重排序。预排序可以根据手写的规则（因为我们主要担心众所周知的长距离的移动），或

者从词对齐的源端已标注的平行语料库自动学习到的规则，甚至仅仅使用词性标记。预排序后可以决定唯一的输入序列，或者保留多个潜在的选择。这样做的目的是期望基于短语的模型能够更容易地翻译重排序后的输入，有些输入甚至在翻译中不再允许调序。

另一类语言中词的顺序是自由的，不能被简单地归类为 SVO 或 VSO。回顾一下，固定词序的目的是定义句子中不同成分之间的关系，例如名词短语与动词之间的关系。有些语言使用不同的手段来定义这种关系：标记或名词格，例如，在日语中就使用了标记。一个说英语的人应该也熟悉这种概念：介词扮演了几乎同样的角色（*from the house* 与 *to the house*）。名词的格能改变单词的表面形式，例如：*der Mann* 是主语，而 *dem Manne* 是宾语。

在统计机器翻译中，如何翻译使用不同手段定义句法关系的两种语言并没有得到充分的研究。一部分原因是因为大多数研究把英语作为目标输出语言，而通过 n 元语言模型可以较好地处理英语中固定的词序。

10.7 工具和数据资源

虽然构建机器翻译系统是一项复杂的任务，但使用一些可获得的开源软件和数据资源使得完成该任务比较方便。应该留意任何最新的发展情况，这里只列出一些最常用的软件和资源。 356

10.7.1 基本工具

除了句子对齐和词对齐这两项不平凡的任务外，统计机器翻译系统的其他训练过程的实现也是相当直观的。

在原始资料中（例如，一本书及其翻译或多语言的网站）找到的已经翻译的文本很少是句子对齐的格式，而这种格式又是必需的。因此，第一步工作就是为每个句子找到对应的翻译。

最简单的方法是基于句子的长度进行相似度度量；更复杂些的方法还可以利用双语词典的信息。一种广泛使用的用于句子对齐的工具是 Hunalign[1]，这种工具利用上面的两种信息来确定最好的对齐关系，也具有过滤掉落在不匹配句对的功能。

在 10.3 节中详细讨论了词对齐的问题。GIZA++ 工具包[2] 是较早提出的通用 IBM 模型的开源实现，使用范围很广。最近，词对齐的问题再次受到研究机构的重视。Berkeley word aligner[3] 是受重视的一个成果，这种工具把对称化思想（回顾 10.3.5 节）更紧密地融入到词对齐方法中。

对机器翻译来说，语言模型的使用是必需的。大多数情况下，机器翻译系统集成现有的语言模型工具和库，而不是再次开发。最流行的工具是开源的 SRILM 工具包[4]，已经使用了十多年的时间。更新的工具是 IRSTLM 工具包[5]，使用压缩表示和可扩展训练方法构建大规模的语言模型（数以亿计的单词）。还值得一提的是 randLM 工具包[6]，使用一种有损数据结构来更有效地存放如此大规模的语言模型。

---

① http://mokk.bme.hu/resources/hunalign/。
② http://www.fjoch.com/GIZA++.html。
③ http://nlp.cs.berkely.edu/Main.html#WordAligner。
④ http://www.speech.sri.com/projects/srilm。
⑤ http://hlt.fbk.eu/en/irstlm。
⑥ http://sourceforge.net/projects/randlm/。

10.7.2 机器翻译系统

整个机器翻译系统——包括训练程序和解码器——均能通过开源许可的方式获得。

最常用的工具是 Moses[㊀]，实现了本章中介绍的大部分方法，并利用现有的工具完成了词对齐和语言模型的建立。较新的工具如 Joshua[㊁] 解码器，主要着重于基于层次的模型和基于句法的模型的开发。

尽管在本章没有提及基于规则的机器翻译，但目前为止还有很多商用翻译系统基于手写的规则。典型地，这些系统在翻译决策时可以利用更详细的知识，但不能使用语言模型
357 和其他概率加权的决策过程。无论如何，这仍然是机器翻译领域一个活跃的方向。开源的 Apertium 项目[㊂] 旨在为许多语言对建立基于规则的机器翻译系统。

10.7.3 平行语料

最后，但也是最重要的是，必须有已经翻译好的文本作为统计机器翻译系统的训练语料——越多越好，与你感兴趣的领域越接近越好。

实际上机器翻译系统中使用的所有平行语料都是现成的语料，也就是说，它们是因为别的目的而建立的，然后被应用到机器翻译的研究中。这些语料的最主要来源是政府（如加拿大的法语-英语语料）和国际组织（联合国、欧洲议会）。当今多数的翻译产生于经济领域（如产品文档、营销材料），但其所有者严格地保护着这些材料。一个有希望的新方向是利用网络合作的方式创建翻译语料——流行的方式是维基翻译（wiki translation）和众包（crowd sourcing）。

下面是一些常用语料的简单列表：

- Canadian Hansards[㊃] 由加拿大议会记录组成，包括英语和法语。
- 欧洲议会语料[㊄] 由已翻译的欧洲议会记录组成，包括 11 种语言，每种语言大约有 4000 万单词。
- Acquis corpus[㊅] 由欧盟成员国必须提交的法律文档组成。这些语料包括 22 种语言，每种语言达到 4000 万单词。
- OPUS 项目[㊆] 收集了很多来源的平行语料，包括开源的文档和电影对白。
- LDC[㊇] 是计算语言学领域最主要的语料来源。该组织也发布平行语料，特别是阿拉伯语-英语对和中文-英语对，这两种语言对是最近美国资助的研究计划的目标。

10.8 未来的方向

358 尽管统计机器翻译已经有 20 多年的历史，但它仍然非常活跃。这个领域非常注重评测活动，因此更关注性能而不是新奇的想法，这导致了被证明的有效新方法的快速采用。下面简要介绍当前研究的一些主要问题。

㊀ http://www.statmt.org/moses/。

㊁ http://sourceforge.net/projects/joshua。

㊂ http://www.apertium.org/。

㊃ 部分语料可从 http://www.isi.edu/natural-language/download/hansard/得到；更多语料可通过 LDC 获取。

㊄ http://www.statmt.org/europarl/。

㊅ http://wt.jrc.it/lt/Acquis/。

㊆ http://urd.let.rug.nl/tiedeman/OPUS/。

㊇ http://www.ldc.upenn.edu/。

统计机器翻译模型中存在大量的参数，这些参数值的估计是一个核心问题。虽然长久以来都是基于训练语料中体现的概率分布估计参数，但研究者们还是有浓厚的兴趣尝试使用更先进的机器学习方法。当前的系统依靠两种方法的混合：生成模型（例如，短语翻译概率）和判别式训练（参数调节，参见10.4.5节）。

基于句法的模型的研究也非常活跃，有许多待解决的问题：是用短语结构文法还是依存文法来表示句法，特定的文法形式体系和有效的解码算法等。

统计模型的训练对平行语料的依赖性使得大家非常关注语料资源缺乏的情况。如何才能使用可比语料或者纯粹的单语语料？如何处理小规模的领域内语料和大规模的领域外语料？可以以用户与机器翻译系统交互的形式作为额外的训练语料，从而提高系统的效果吗？

最后，因为机器翻译与其他的信息处理应用有密切关联，所以把统计机器翻译集成到这些应用中也是很吸引人的。

最近的研究已经尝试了两种应用。第一种，语音翻译系统的目的是集成语音识别、机器翻译和语音合成。第二种，因为高质量的翻译最终还是需要人的参与，最新的计算机辅助翻译工具利用了统计翻译中的方法。

10.9 总结

数据驱动方法的应用使得机器翻译这一领域非常活跃。

机器翻译似乎是一项很直观的任务：在不改变意义的情况下，把一种语言的文本翻译成为一种语言的文本。但如何准确地度量一个句子的翻译是否是正确的仍然是一个尚未解决的问题，如10.2节中关于评测的讨论。

从双语平行语料库中学习翻译模型的一个重要的步骤是词对齐（参见10.3节）。基于短语的模型（参见10.4节）和基于树的模型（参见10.5节）的构建都依赖于一个词对齐的平行语料库，这是当前最常用的方法。

统计机器翻译已经取得了很大的进步，例如，使用当前的系统把新闻报道从法语翻译成英语，可以得到可读性和准确率都较高的输出。但是仍然存在很多挑战，尤其是对于具有不同词序和形态丰富的语言对（参见10.6节）。

大量的开源工具和资源方便了研究者对该领域的研究（参见10.7节），许多未来研究方向仍需探索（参见10.8节）。 [359]

参考文献

[1] P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, R. L. Mercer, and P. Roossin, "A statistical approach to language translation," in *Proceedings of the International Conference on Computational Linguistics (COLING)*, 1988.

[2] C. Callison-Burch, C. S. Fordyce, P. Koehn, C. Monz, and J. Schroeder, "Further meta-evaluation of machine translation," in *Proceedings of the NAACL 3rd Workshop on Statistical Machine Translation*, pp. 70–106, 2008.

[3] C. Callison-Burch, P. Koehn, C. Monz, and J. Schroeder, "Findings of the 2009 Workshop on Statistical Machine Translation," in *Proceedings of the NAACL 4th Workshop on Statistical Machine Translation*, pp. 1–28, 2009.

[4] M. Przybocki, K. Peterson, and S. Bronsart, "Official results of the NIST 2008 metrics for machine translation challenge (MetricsMATR08)," 2008. http://nist.gov/speech/tests/metricsmatr/2008/results/.

[5] P. Koehn and B. Haddow, "Interactive assistance to human translators using statistical machine translation methods," in *Proceedings of the 12th Machine Translation Summit (MT Summit XII)*, 2009.

[6] L. Truss, *Eats, Shoots & Leaves—The Zero Tolerance Approach to Punctuation*. London: Profile Books, 2003.

[7] D. A. Jones, T. Anderson, S. Atwell, B. Delaney, J. Dirgin, M. Emots, N. Granoien, M. Herzog, T. Hunter, S. Jabri, W. Shen, and J. Sottung, "Toward an interagency language roundtable based assessment of speech-to-speech translation capabilities," in *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas (AMTA)*, 2006.

[8] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics (ACL)*, 2002.

[9] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, "Introduction to WordNet: An online lexical database," Tech. Rep. CSL 43, Cognitive Science Laboratory Princeton University, 1993.

[10] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72, 2005.

[11] S. Vogel, H. Ney, and C. Tillmann, "HMM-based word alignment in statistical translation," in *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*, 1996.

[12] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, 2003.

[13] E. Matusov, R. Zens, and H. Ney, "Symmetric word alignments for statistical machine translation," in *Proceedings of the Conference on Computational Linguistics (COLING)*, pp. 219–225, 2004.

[14] D. Ren, H. Wu, and H. Wang, "Improving statistical word alignment with various clues," in *Proceedings of the 11th Machine Translation Summit (MT Summit XI)*, 2007.

[15] Y. Ma, S. Ozdowska, Y. Sun, and A. Way, "Improving word alignment using syntactic dependencies," in *Proceedings of the ACL-08: HLT Second Workshop on Syntax and Structure in Statistical Translation (SSST-2)*, pp. 69–77, 2008.

[16] V. L. Fossum, K. Knight, and S. Abney, "Using syntax to improve word alignment precision for syntax-based machine translation," in *Proceedings of the 3rd Workshop on Statistical Machine Translation*, pp. 44–52, 2008.

[17] R. C. Moore, "A discriminative framework for bilingual word alignment," in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 81–88, 2005.

[18] R. C. Moore, W.-T. Yih, and A. Bode, "Improved discriminative bilingual word alignment," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 513–520, 2006.

[19] A. Ittycheriah and S. Roukos, "A maximum entropy word aligner for Arabic-English machine translation," in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 89–96, 2005.

[20] N. F. Ayan, B. J. Dorr, and C. Monz, "NeurAlign: Combining word alignments using neural networks," in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 65–72, 2005.

[21] B. Taskar, L.-J. Simon, and D. Klein, "A discriminative matching approach to word alignment," in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 73–80, 2005.

[22] H. Wu and H. Wang, "Boosting statistical word alignment," in *Proceedings of the 10th Machine Translation Summit (MT Summit X)*, 2005.

[23] H. Wu, H. Wang, and Z. Liu, "Boosting statistical word alignment using labeled and unlabeled data," in *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pp. 913–920, 2006.

[24] C. Cherry and D. Lin, "Soft syntactic constraints for word alignment through discriminative training," in *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pp. 105–112, 2006.

[25] P. Blunsom and T. Cohn, "Discriminative word alignment with conditional random fields," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 65–72, 2006.

[26] J. Niehues and S. Vogel, "Discriminative word alignment via alignment matrix modeling," in *Proceedings of the 3rd Workshop on Statistical Machine Translation*, pp. 18–25, 2008.

[27] S. Venkatapathy and A. Joshi, "Discriminative word alignment by learning the alignment structure and syntactic divergence between a language pair," in *Proceedings of SSST, NAACL-HLT 2007/AMTA Workshop on Syntax and Structure in Statistical Translation*, pp. 49–56, 2007.

[28] A. Fraser and D. Marcu, "Measuring word alignment quality for statistical machine translation," *Computational Linguistics, Squibs & Discussion*, vol. 3, no. 33, pp. 293–303, September 2007.

[29] G. Foster, R. Kuhn, and H. Johnson, "Phrasetable smoothing for statistical machine translation," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pp. 53–61, 2006.

[30] K. Knight, "Decoding complexity in word-replacement translation models," *Computational Linguistics*, vol. 25, no. 4, pp. 607–615, 1999.

[31] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* pp. 160–167, 2003.

[32] C. Callison-Burch, C. Bannard, and J. Schroeder, "Scaling phrase-based statistical machine translation to larger corpora and longer phrases," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pp. 255–262, 2005.

[33] R. Zens and H. Ney, "Efficient phrase-table representation for machine translation with applications to online MT and speech translation," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pp. 492–499, 2007.

[34] H. Johnson, J. Martin, G. Foster, and R. Kuhn, "Improving translation quality by discarding most of the phrasetable," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 967–975, 2007.

[35] H. Wu and H. Wang, "Comparative study of word alignment heuristics and phrase-based SMT," in *Proceedings of the 11th Machine Translation Summit (MT Summit XI)*, 2007.

[36] L. Zettlemoyer and R. C. Moore, "Selective phrase pair extraction for improved statistical machine translation," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pp. 209–212, 2007.

[37] C. Quirk and A. Menezes, "Do we need phrases? Challenging the conventional wisdom in statistical machine translation," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, Main Conference*, pp. 9–16, 2006.

[38] J. B. Mariño, R. E. Banchs, J. M. Crego, A. de Gispert, P. Lambert, J. A. R. Fonollosa, and M. R. Costa-jussà, "*N*-gram-based machine translation," *Computational Linguistics*, vol. 32, no. 4, 2006.

[39] M. R. Costa-jussà, J. M. Crego, D. Vilar, J. A. R. Fonollosa, J. B. Mariño, and H. Ney, "Analysis and system combination of phrase- and *N*-gram-based statistical machine translation systems," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pp. 137–140, 2007.

[40] M. Eck, S. Vogel, and A. Waibel, "Estimating phrase pair relevance for translation model pruning," in *Proceedings of the 11th Machine Translation Summit (MT Summit XI)*, 2007.

[41] M. Eck, S. Vogel, and A. Waibel, "Translation model pruning via usage statistics for statistical machine translation," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pp. 21–24, 2007.

[42] T. Kutsumi, T. Yoshimi, K. Kotani, I. Sata, and H. Isahara, "Selection of entries for a bilingual dictionary from aligned translation equivalents using support vector machines," in *Proceedings of the 10th Machine Translation Summit (MT Summit X)*, 2005.

[43] D. Chiang, "A hierarchical phrase-based model for statistical machine translation," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pp. 263–270, 2005.

[44] P. Koehn and H. Hoang, "Factored translation models," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 868–876, 2007.

第11章

Multilingual Natural Language Processing Applications: From Theory to Practice

跨语言信息检索

Philipp Sorg, Philipp Cimiano

单语言信息检索（Information Retrieval，IR）的研究始于20世纪60年代以前，到了现在已经形成了成熟的研究领域和体系。基于领域跨语言信息检索也在近些年来引起了大量的兴趣。造成这个现象的原因是多方面的。首先是相关学科和技术尤其是机器翻译的进步促使了跨语言信息检索系统的发展。其次，我们在过去的几年发现，非英语的互联网用户的数量增长得越来越快[3]，他们使得越来越多的非英语信息内容出现在网络上。再者，Web 2.0时代的到来使得跨语言检索的需求增加。尽管专业的网站通常都有针对主要语种的翻译，但诸如Flickr，Yahoo! 问答，Facebook和Twitter等的Web 2.0应用中的大量用户生成的内容却没有翻译。最后，对于那些跨国公司或者国际组织团体而言，跨语言信息检索也是基本需求。

本章我们将介绍跨语言信息检索（Corsslingual Information Retrieval，CLIR）和多语言信息检索（Multilingual Information Retrieval，MLIR）的相关技术。其中CLIR涉及两种语言——查询中（query）语言和文档集合（collection）语言，依据语言同质的（language-homogeneous）文档集以查询的语言来回答查询。与此相对照，MLIR则涉及任意多种语言，依据包含不同语言的文档集，系统会以支持的任何语言回答查询。

因为CLIR和MLIR所用到的主要技术是传统信息检索的基本技术，所以我们也会介绍信息检索的一些基本技术。本章涉及开发CLIR和MLIR系统可供选择的一系列相关技术以及实践中最佳的方法。想要了解CLIR和MLIR所用到的主要方法和技术框架的研究者，或者是正在实现一个多语言信息检索系统的开发者可以在本章有所收获。本章探讨了文档模型（document model）、检索函数（retrieval function）和多语言信息检索系统中的翻译手段等，具体包括特定语言下的文档预处理、统计信息检索模型、机器翻译系统和IR系统评测等。

11.1 概述

信息检索主要处理信息项（information item）[1]的表示、存储、组织以及存取。通过合理的信息项表示与组织，信息检索系统可以使想得到特定信息的用户得到他们感兴趣的信息。

最典型的方式就是：用户把自己的信息需求用查询关键字表示，这些关键字通常是一些关键词的集合。IR系统用这些关键字作为输入，把系统认为和用户信息需求最相关的信息项作为结果返回。同问答系统（Question Answering，QA）不同的是，QA返回的是对用户提交问题的直接回答，而IR系统则返回一个按与用户查询中的相关度排序的文档列表。IR系统的目标是将相关的文档尽可能排在靠前，而不相关的文档则尽可能排在后面。

[3] http://www.internetworldstats.com/stats7.htm。

本章我们主要介绍信息检索的多语言方面。在 CLIR 和 MLIR 中，用户查询信息所使用的语言和相关信息的语言可能是不同的。虽然相关性概念原则上说是和语言无关的，但用户还是需要理解所检索出的项，所以多语搜索系统必须保证仅返用户所支持语言的检索结果或所返回的结果已被翻译为用户所支持的语言。

11.2　文档预处理

本节我们介绍文档的预处理工作。预处理的输入是原始文档，输出则是词元集合。词元是术语（类）在文档中的具体出现，表示最小的意义单元。预处理的输出定义了一个词汇表（vocabulary），可以用于对文档集合进行索引，如 11.3 节所述。

大部分 IR 模型和系统都简单地假设文档中词元间的顺序是无关紧要的。**短语索引**（phrase indices）和**位置索引**（positional indices）则是用到词元间次序的两个例子（相关介绍请见 Manning，Raghavan 和 Schtze [2]）。

根据语言、文字以及其他因素的差异，识别术语的过程可能差别很大。对于西欧语言来说，语言中的词语就可以用作 IR 系统的术语。而用于亚洲语言的 IR 系统中术语则常定义为固定数量的连续字符序列。以中文为例，该语言的词语之间通常没用空格进行分隔，因此将术语定义为字符序列就可以绕过词语识别的问题。

接下来的几节将介绍文档预处理的一些常用技术手段。我们将着重强调对于不同语言和文字而言预处理的差异，特别包括关于文档结构（document syntax）、编码格式（encoding）、词元化（tokenization）以及词元标准化（normalization of token）等的不同。算法 11-1 描述了完整的文档预处理流程，此流程显示了各不同的预处理步骤之间的依赖关系，
365 这些步骤将随后介绍。

算法 11-1　对文档 d 的预处理流程，结果是词元集合 T

```
d ← INPUT
T ← ∅
[c1, c2, ...] ← character-stream(d)
B ← tokenize([c1, c2, ...])
while B ≠ ∅ do
  t ← POLL(B)
  if is-compound(t) then
    B ← B ∪ compound-split(t)
  end if
  if not is-stop-word(t) then
    t = normalize(t)
    T ← T ∪ {t}
  end if
end while
return T
```

11.2.1　文档句法和编码

预处理流程的第一步是从给定数据流（data stream）中识别出文档集合 [2]。在很多情况下，研究者都可以直接把一个特定的文件或者网页当作一个文档。但是，对一些应用场景而言，文件可能包含很多文档（例如，XML 检索）或者文档分散在多个文件中（例如，网页集合）。因此，开发者需要根据具体搜索任务中需要检索的信息项类型来对文档究竟由什么构成给出定义。

接下来的步骤就是把这些文档转换成表示其内容的字符流。这个步骤的目标是把不同编码、文字和文字方向的文档转换成统一的表示。这个步骤完成后两个讲种相同而且内容

相同的文档应该转化成相同的字符流。需要处理如下挑战：

文档句法（document syntax） 文档的内容通常会按给定文件类型（file type）的句法进行编码。在对文件建立索引之前，要先根据文件类型规范抽取出文件中的文本内容，以避免那些包含格式说明或元数据的词汇元素被索引。

需要从中抽取内容的文件格式实例有 PDF 文件或 Web 页面。现有的很多函数库都支持对 PDF 或 HTML 文件类型的解析以及其中文本内容的抽取。

在很多情况下，一篇文档中只有一部分的文本内容表达了文档的语义信息（semantic content），其他的文字部分则对所有文档都是一样的，例如页眉（header）和页脚（footer）等部分。在这种情况下，对所有内容建立索引也会引入噪声。我们需要根据特定的文档格式，依据其结构设计具体的抽取算法以识别出其中的重要文本内容。以网页为例，如何抽取信息依赖于页面的结构。页眉中诸如标题、关键字等信息描述了网页的内容，应该被抽取，而所有页面都一样的顶部条（top bar）和菜单栏（menu）则应该被忽略。 367

编码（encoding）和**文字**（script） 计算机系统中字符的底层表示方法叫做编码。历史上，ASCII 的字符编码方案被广泛用于编码英文文档。这是一种很早的英文字符和一些符号的编码方式，只对拉丁字母表中的字符进行了编码，不能编码超出此范围的字符集合。作为一个能支持多数通用语言的编码规范，Unicode [3] 已成为国际化应用的事实标准。所有语言的每个字符都对应唯一的数值，这保证了很高的跨平台可移植性也避免了因转换带来的错误。Unicode 也支持从右向左书写的文字并可用于编码诸如阿拉伯语或者希伯来语等语言。由于多数操作系统和现代编程语言都支持 Unicode，所以强烈推荐使用 Unicode 作为默认的字符编码方式[⊖]。

IR 系统的查询和文档集通常会用同一种文字书写系统进行表示，在某些程度上，检索可归结为字符匹配，如果查询和文档文字不兼容则匹配不成功。韩语拥有两种通用文字系统，即谚文（Hangul）和韩文汉字（Hanja）。从一种文字系统转换到另外一种文字系统叫做音译（transliteration），请不要将此概念与翻译混淆，因为它所涉及的语言并未改变。音译一般是通过另外一种语言的拼读方式模仿源语言的发音方式，这种模仿是典型的可逆语音转换的过程。

为了对包含多种文字文档的数据集进行预处理，一般会采用罗马化（romanization）技术以获得一个统一的表示。罗马化就是把任何的文字音译成拉丁（罗马）字母。对检索系统而言，这种音译方式在搜索通名（common name）时特别有用。通过罗马化音译，这些不同语言和文字文档中所使用的通名大多被统一地映射到了相同的字符串序列。作为联合国地名专家组（United Nations Group of Experts on Gengraphical Names，UNGEGN）的一部分，罗马化工作组（Working Group on Romanization System）[4] 为多种语言提供了罗马化转换资源。这个工作组的目的是引入统一的地理名称表示，然而，他们提供的资源可以用于任意文本的罗马化。

文字的方向（direction of script） 文字记录了人的口语，所以文档中的词和字母的顺序同人们语言流的顺序是一致的 [2]。文档中的字节序列也一般会反映此自然顺序。文字的实际方向会由应用程序的可视化层处理，该层通常是用户界面的一部分。预处理的主要问题之一是文档集中包含了多种具有不同文字方向的语言文本。一个例子是阿拉伯语文本中包含英文通名。由于本章我们讨论的主要问题是 MLIR 的核心功能和模型，所以我们

⊖ 从技术上说，Unicode 是一种字符到码点的映射，Unicode 编码包括 UTF-8 和 UTF-16。

不进一步研究上述难点问题，而只关心文档的数据层，这些层不涉及文字的方向。但是如
368 果设计用户界面，这就是一个更为重要的问题了。

11.2.2　词元化

词元化就是把字符流分割成词元串的过程。词元是术语的实例，也对应了最小的索引单元。所有术语的集合称为词汇表。接下来将介绍3种需要不同词元化方法的通用词汇表类型。对于信息检索系统来说词汇表构造是很重要的。相关设计指南可以参考11.2.4节。

我们以如下句子为例来说明不同的词元化方法：

`It is a sunny day in Karlsruhe.`

分词　最常用的词元化方式就是利用词边界来分割文本。这样，词元就对应语言的词，而词汇表则等价于词典（包括词条）。

对于采用空白字符来划分单词的语言，这种方式在多数IR系统中已被成功使用。空白字符和标点符号因此成为将文本划分为词元的线索。这类语言的例子是西欧语言。这种方式的问题是，简单地在空白和标点符号处分割文本将可能把表示为单个词元的文本分割开。这类错误源的例子是连字符（*co-education*）、专名中的空格（*New York*）、日期（*April 28, 2010*）和电话号码等[2]。多数情况下可以采用启发式规则判断是否应该拆分。也可以训练分类器来处理这类不确定的情况。对于上面的例子来说按空白字符分词的结果如下：

`"It", "is", "a", "sunny", "day", "in", "Karlsruhe"`

对于中文等没有空白字符的文字来说，在词语边界词元化是非常难的问题。处理方法可分成两类：词汇的和语言学的。词汇方法利用词典在词元流中匹配术语，以获得一个完全覆盖。通常这种匹配是不确定的。为了得到最准确的结果，我们常常采用一些启发式的规则，例如最大匹配原则等。此方法的一个问题是不在词典中因而本会被匹配但又应该被检测出的未知术语。语言学方式则利用了包括已分词的文本在内的背景知识。使用基于词元频率的统计指标的目标是找出当前文本最可能的分割结果。隐马尔可夫模型（Hidden Markov Model，HMM）可以有效地实现此计算[5]。类似条件随机场（Conditional Random Field）的机器学习方法也已被成功用于此问题[6]。由于没有方法能获得完美的分词结果，错误的词元会被用于索引和检索，从而影响系统的性能。

短语索引（phrase index）　短语索引是以分词为基础的，此时词元的含义不再是单个的词汇而是一些词汇的元组。短语索引就是通常所说的n元模型（n-gram model），n定义了词元中的单词数量。通过迭代地在已分好词的字符流中移动长度为n的单词窗口，字符流将被映射为词元序列。这些词元保留了单词的上下文信息，但是通过这种方法构造出的词汇表会非常大。搜索过程中利用短语形式的词元的另一个问题是术语的数据稀疏性，
369 即查询中的许多术语根本不在文档集中出现。为了避免这个问题，可以在基于单个词语作为分词单元的检索方法之上使用短语索引。对于给出的例子来说，3元模型的分词结果如下：

`["It is a", "is a sunny", "a sunny day", "sunny day in", …`

基于字符的n元模型（character n-gram model）　基于字符的n元模型把术语项定义成连续的n个字符。构造的方法是在字符流上移动n个字符大小的窗口。这种切分方式得到的术语不是词语。词汇表则定义为包含n个字符的字符序列集，这里的字符也包括空白字符和标点符号。术语长度取4或5是较为合理的。对于上述例子来说，一个4元的字符模型的词元化结果如下所示：

`["_It_", "It_i", "t_is", "_is_", "is_a", "s_a_", "_a_s", …`

这种方法可应用于任意字符流，并不依赖诸如空白字符等词语边界线索。这种方式可以用于对任何文字的文本进行词元化，因为不需要进行词语的切分，所以也不会引入分词错误。这种方法已被证明在一些场景中优于基于词语划分的模型[7]。它也被应用到了拼写检查问题[2]。在多语言信息检索中，仅当无须把术语映射到不同语言时才能使用基于字符 n 元组的词元化，因为此时的术语并不与单词对应，因而无法进行跨语言映射或翻译。字符 n 元组的另一个问题是，检索结果将更难可视化。因为搜索过程中匹配的是字符 n 元组，所以我们很难在搜索结果中将匹配单词高亮显示。

11.2.3 规范化

规范化的目标是为了把描述相同概念的不同词元映射为同一个术语。一个英文规范化的例子是将复数形式映射为其单数形式，如把 cars 转换成 car。规范化可以看成是建立术语的等价类。在搜索过程中，规范化可以增加检出相关文档的数量从而提高系统的召回率。在建立索引前须对文档集进行规范化，在查询前则须对查询进行规范化，而且这两个规范化模型必须是相同的，这样可以保证所有词元被映射到其等价术语，这在把检索和文档匹配时是很重要的。

不同的语言有不同的规范化方式。对于那些词语具有很多形态的语言来说，一个普遍的做法就是把（复合）术语映射为对应的原形。例子是罗马语和德语。针对此问题有两个主要方法，一是**原形化工具**（lemmatizer），该工具利用词汇信息将术语映射为其原形。这种方式需要丰富的语言学资源。第二种方法是**词干化工具**（stemmer），该工具利用一些简单的规则将术语映射为其词干。对于复数的转换，可以通过删除词尾的 's' 来进行词元化。这种方式不需要丰富的语言资源。这种方式的缺点是术语不是被映射为原形而被映射为词干，这里词干并不一定对应单词。在多数情况下，许多不同概念的术语会被映射成同样的词干。例如，术语 organize、organizing 和 organization 都会被转换成 organ。这样在索引中就无法区分这些术语了。反过来，原形化工具则可以正确地把术语 organize 和 organizing 转换成原形 organize 而术语 organization 则保持不变。

370

对于使用变音符的语言来说，规范化更有用。如果变音符的使用不一致，那么在规范化步骤中将它们删除是有益的。例如，如果用户在查询中没有给出变音符，那么规范化时在索引前就需要删除变音符。删除变音符可以利用简单的基于规则的方法。

对于一些屈折语（例如德语、荷兰语、意大利语）来说，**复合词拆分**（compund splitting）是另一种规范化方式。这些语言中复合术语通常被拆分成组合的原形以增加系统的召回率。这种复合词拆分的方式十分类似于前面处理亚洲语言的分词过程。可以利用词典的方式匹配出复合词中的术语，也可以利用语言学方式来使用背景知识。许多现有的方法，通过比较复合词的频率和内部成分词的频率来决定是否要对复合词进行拆分。在应用复合词拆分方式的时候，通常复合词和拆分的成分都会被加入词元流，这样在检索的过程中也允许对复合词进行匹配。

删除停用词（stop-word）也是规范化步骤，该步骤会从词元流中删除常用术语。几乎所有文档中都包含的术语对于判别文档的相关性起不了作用。停用词一般都是冠词、介词或者连词。很多语言已经有可用于匹配和过滤词元的停用词表。

就上述例子来说，经过词干化并删除停用词后的结果是：

[sunny], [day], [karlsruh]

11.2.4　预处理最佳实践

在上节介绍完不同的预处理方法之后，我们提供了不同类型语言的预处理方法指南。

使用拉丁语和斯拉夫字母表语言的处理　这些语言都是利用空白字符分隔词语，所以分词是较好的词元化过程。根据搜索任务的需要，我们可以对通名、日期和电话号码等进行特殊处理以增强这一过程。词干化和原形化虽然通常可以增强这些语言检索的效果，但是增强的效果并不总是很明显[1]。词干化的实现代价不高，因而也值得做一做。如果对检索结果的精确率要求较高，那么规范化反而可能会降低搜索的质量。对于屈折语言来说，复合词分割将性能提高了25%[8]。

阿拉伯语、梵语、希伯来语的处理　用空白字符作为词边界的分词方法可以用于这些语言的处理（而且也建议使用），因为这些语言的词语形态变化不是很多，词语形态分析（词干化和原形化）不是必需的，但是变音符的处理还是需要注意。在建立索引和查询之前需要把带变音符的词语转换成规范的表示。

使用象形文字或音节文字的语言的处理　诸如韩语和日语之类的语言使用了多个文字书写系统，所以对应的查询串和文档都会用多种文字书写方式书写。在这种情况下，在检索和文档处理前需要对查询或文档进行音译以确保搜索过程的兼容性。这些文字系统中，词语通常不是以空白字符来分割的。如果某语言（如中文）已经有丰富的语料，那么基于启发式方法或者机器学习方法的分词模型已证明会取得很好的效果[6]。如果没有这些语料，也可以采用基于字符 n 元组的词元化模型。这种方法是和语言无关的且避免了复杂的词语边界检测方法。该方法已被证明很鲁棒并在处理欧洲语言时可取得与基于分词的系统相当的结果[7]。

371

11.3　单语信息检索

多数 MLIR 的实现方法要么直接基于单语 IR 技术，要么使用标准 IR 模型。MLIR 可以看成是多个不同语言单语信息检索系统的聚合。除了聚合的技术之外，基于特定语言检索的预处理，特别是翻译（11.5 节中说明）也是必要的。通常情况下，MLIR 采用和单语信息检索相同的索引结构以及类似的文档与检索模型。本章对单语信息检索进行了综述，包括文档表示、索引结构、检索模型以及文档先验模型等。我们关注 MLIR 和 CLIR 都会用到的信息检索的重要方面。如果想更多地了解单语信息检索模型，可以参考 Manning 等人[2]以及 Baeza-Yates 和 Ribeiro Neto[1]的论文。

11.3.1　文档表示

在 11.2 节中，我们介绍了文档的预处理工作，其结果是文档用词元流表示。词元是术语的实例，由词、词干或词的原形或者字符 n 元组来定义。本章介绍的信息检索模型独立于所用到的词汇表，并可以适用于任意术语模型。为了便于解释，我们简单假设本章所用到的术语是口语中的词汇，这种表述符合人们直观认为的词汇表。

现阶段大部分的信息检索方法都是用到了基于术语的独立性假设（independence assumption）的文档模型，这就意味着文档中术语的出现独立于相同文档中其他术语的出现。尽管这种独立性假设过于简单但用在信息检索模型中所取得的效果还是可以接受的。

在此独立性假设下，文档可以用向量空间模型（vector space model）表示，向量空间是由词典中的词构成，向量的每一维对应着词汇表中的一个术语。文档经由一个映射函数 f 而表示成向量，这个函数可以把文档 d 的词元流映射成术语向量 $\vec{d}$。有许多不同的映射

函数 f，最著名的是：

- 布尔文档模型（boolean document model）：如果某个术语在文档中出现至少一次，那么术语对应的向量维被设置成 1，否则设置为 0。
- TF 文档模型（TF document model）：向量每个维度的值依赖于该维度所对应术语在文档词元流中出现的次数——术语的频率。在术语向量中可以直接使用术语的频率，一种可能的变体是文档长度进行归一化后的术语频率。
- TF.IDF 文档模型（TF.IDF document model）：这类模型在术语频率值基础上额外再乘以术语的逆文档频率（inverse document frequency），所谓术语的文档频率指的是文档集合中包含此术语的文档个数。因而，逆文档频率将对不常见术语给予更高的权重，面对那些无法很好区分集合中文档的高频术语以更低的权重。大多数情况下 TF.IDF 模型中会使用取对数后的逆文档频率值。

给定一个文档集合，每个文档的术语向量可以组合起来形成一个术语-文档矩阵（term-document matrix）。这个矩阵的每一行表示一个具体的词项，每一列表示一个文档。我们将使用如下文档来解释不同的文档表示：

```
Doc1: It is a sunny day in Karlsruhe.
Doc2: It rains and rains and rains the whole day.
```

上述讨论的不同文档模型所表示出的术语-文档矩阵如下所示：

词	Boolean		TF		TF.IDF	
	文档 1	文档 2	文档 1	文档 2	文档 1	文档 2
sunny	1	0	1	0	$1\log 2/1=0.7$	0.0
day	1	1	1	1	$1\log 2/2=0.0$	$1\log 2/2=0.0$
Karlsruhe	1	0	1	0	$1\log 2/1=0.7$	0.0
rains	0	1	0	3	0.0	$3\log 2/1=2.1$

11.3.2 索引结构

信息检索系统的一个重要方面是时间性能。用户希望检索的结果可以实时获取，如果延迟了 1 秒就会认为检索系统响应过慢。显然，对给定查询简单地遍历全部文档的方式无法用于大规模的文档集。目前信息检索系统迅速的响应速度得益于倒排索引（inverted index）。其基本思路是，将各术语所出现的文档信息存储下来。这一思想为每个术语存储了它出现的文档的信息。这种术语到文档的对应关系即为倒排表（posting list）。具体例子可见 Manning 等人 [2] 的文章。在检索的过程中只需要对查询术语涉及的倒排表进行处理即可，由于用户的查询字符串通常只包含少量术语，所以检索分值只需要很小的平均时间复杂度即可计算。

对于上面给出的文档集例子，我们可以得到对应的倒排表：

```
sunny     -> doc1(1x)
day       -> doc1(1x), doc2(2x)
Karlsruhe -> doc1(1x)
rains     -> doc2(3x)
```

使用倒排索引的一个瓶颈是内存耗费问题。将倒排表加载到内存中是一个很慢的过程，应该避免这种情况经常发生。一些启发式的方法可以帮助决定哪个部分的倒排表应该驻留内存并且哪个部分应该被替换。减少倒排表所占内存的常用方法是压缩或者利用后缀树的技术手段，这些方法在 Baeza-Yates 和 Ribeiro-Neto [1] 的论文中有所介绍。对于超

大规模语料库来说，可以使用分布索引。倒排表通常会分布式地存储在几个服务器上。每个服务器只存储倒排表一部分子集。

要降低检索的时间复杂度，还可以利用非精确检索模型（inexact retrieval model）或Top-k模型，这些模型不会对所有的文档进行比对，只处理那些相关程度高的文档。通过这些方法，信息检索系统的时间复杂度可以在不明显降低检索性能的前提下进一步降低[9]。

11.3.3　检索模型

检索模型用于评价用户查询和文档之间的相关度。相关函数可以用不同的理论计算模型推导出来。接下来将介绍3类主要的检索模型：布尔模型（boolean model）、向量空间模型（vector space model）和概率模型（probabilistic model）。用户查询常根据所用模型的不同采用不同的方式表示。布尔模型的用户查询中被表示成二值的术语向量（binary term vector）。基于我们先前的独立性假设，这种方式的表示会丢失查询串中术语间的次序信息，而只反映了术语是否出现或者缺失。对向量空间模型和概率模型来说，查询串被表示成实值的向量空间，其中每个查询术语的分值将被累计[2]。

布尔模型　布尔模型是信息检索中出现最早的检索模型。在布尔模型中，查询串和文档的相关程度计算结果也是一个布尔值，是通过匹配两个分别代表查询串和文档的术语二值向量的方式来进行。因为向量空间模型和概率模型的效果要好于传统的布尔模型，所以我们在本章着重介绍这两个模型。如果读者还有兴趣深入了解布尔模型，可以参考Manning等人[2]的论文。

向量空间模型　向量空间模型基于文档的向量空间表示。通过前面的描述我们可以知道，向量空间通过词汇表信息来构造，术语-文档矩阵中的单元（entry）通常由相应的术语频率信息定义。为了计算文档集和给定查询之间的相关度可以采用多种不同的计算策略：

1）累积模型（accumulative model）：此检索函数为每一个查询术语计算出分值，这些查询术语分值则按文档被分别累加以获得每个文档的累积分值。单个术语t的分值计算函数可基于如下指标：

- $tf_d(t)$：术语在文档中的频率。
- $|d|$：文档长度。
- $df(t)$：查询术语的文档频率。
- $tf_D(t)$：文档集合中包含的查询术语的词元数目。
- $|D|$：文档集合中文档的数量。

374

例如，基于术语频率和逆文档频率的简单检索模型的累积分数计算方式如下：

$$score(q,d) = \sum_{t \in q} tf_d(t) \log \frac{|D|}{df(t)}$$

2）几何模型：查询串q的向量空间表示可以表示为术语向量$\vec{q}$。在这种情况下，检索模型可以采用术语向量空间模型中的几何相似度计算方式[2]。比如，余弦相似度（cosine similarity）就是在检索中成功应用的计算方式：

$$score(q,d) = \cos(\vec{q},\vec{d}) = \frac{(\vec{q},\vec{d})}{\|\vec{q}\| \|\vec{d}\|}$$

概率模型　概率模型的基本思想是估计文档与给定查询串相关的可能性。因此相关度被建模成取值为{1, 0}的随机变量R。我们说给定文档d和查询串q相关，当且仅当$P(R=1|d,q) > P(R=0|d,q)$[2,203页]。已经证明，当给定一个二值损失函数和基于全

都可用信息的所有概率的最准确估计，概率模型可获得最好的结果[10]。但是，在实际项目中不可能得到准确的概率估计。在向量空间模型中也可以利用概率空间模型调整启发函数的选择，利用逆文档频率就是一个这样的例子（详细信息请参见Manning等人[23]）。

BM25模型[21]是概率模型的一个例子，在实际应用已被证明非常成功。它的评分函数定义如下：

$$score(q,d)=\sum_{t\in q} idf(t)\frac{tf_d(t)}{k_1((1-b)+b\frac{|d|}{\sum_{d'}|d'|/|D|})+tf_d(t)}$$

$$idf(t)=\log\frac{|D|-df(t)+0.5}{df(t)+0.5}$$

此模型的参数取值一般是$k_1=2$，$b=0.75$，但是也需要根据具体的搜索任务和数据集进行调整。

语言模型（language model）　最近几年里，语言模型也被证实是一个很有力的替代检索模型。语言模型是概率模型的子类，文档、查询或整个集合都由生成模型表示。这些模型由术语的概率分布表示，如文档、查询或文档集生成某个特定术语的概率[12]。

最大似然估计经常被用来定义文档模型。文档d生成一个特定术语t的概率定义为：

$$P(t\mid d)=\frac{tf_d(t)}{|d|}$$

375

在信息检索中一般通过语言模型估计$p(d\mid q)$的概率，也即相关度分值。利用贝叶斯定理可以转换成：

$$P(d\mid q)=\frac{P(q\mid d)p(d)}{P(q)}$$

因为$P(q)$对于特定的查询串来说是常量，$P(d)$可以假定是均匀分布，所以文档的排序主要基于$P(q\mid d)$的值，当将查询串视为独立的术语集合时，$P(q\mid d)$的值可以用文档语言模型进行估计：

$$P(q\mid d)=\prod_{t\in q}P(t\mid d)$$

因为如果所有文档都不包含查询串中的词项，通过上式计算出来的值可能是0，所以需要采用平滑的方法。利用术语先验概率$P(t)$可以构造一个用于检索的混合模型：

$$P(q\mid d)=\prod_{t\in q}((1-a)P(t\mid d)+aP(t))$$

通常情况下可以利用文档中所有术语的集合估计术语的先验概率：

$$P(t)=\frac{\sum_{d\in D}tf_d(t)}{\sum_{d\in D}|d|}$$

11.3.4 查询扩展

查询扩展（query expansion）是一种用来提高检索效果的常用技术手段。在CLIR和MLIR中也引起了特殊关注，通常可以用添加了一些额外术语集合，进一步反映用户可能的信息需求。查询扩展的目标是通过额外扩展的术语信息描述相关内容，使得更多相关文档被检索出来。

上面已经介绍的所有信息检索模型都可以用到查询扩展，通常被扩展的术语被赋于的

权重要比查询中原有术语的权重低。具体权重值依赖于各扩展术语的置信度和查询扩展的整体权重。在概率检索模型中，查询扩展也可以用来改善概率的估计，例如可用于改善查询语言模型的估计。我们接下来要区分扩展术语的两种不同来源：

背景知识（background knowledge）　可利用一些外部的知识资源来寻找给定查询的扩展术语。例如，可以利用辞典以及查询术语的同义词来扩展查询。对于 CLIR 或者 MLIR 来说，对查询串的翻译也是一种特殊形式的查询扩展。在这种情况下，查询将依据其不同语言的翻译中的术语进行扩展。

相关反馈（relevance feedback）　用于查询扩展的相关反馈是一个两步的检索过程。首先，根据原始查询得到一个文档集合。然后，对这些文档集合进行相关性评估并在查询结果中识别出相关的文档集合。通过在扩展文档集合上利用术语频率和文档频率等扩充模型手段，可以得到一些可能有用的术语集合，然后把这些术语集合用于查询中扩展的第二步。

476

在相关性文档的选择步骤中可以利用自动的或者人工的选择方法。在第一种情况中采用人工选择的方式从第一步得到的检索结果中手动挑选出相关的文档。如果采用**伪相关反馈**（Pseudo-Relevance Feedback，PRF）方法那么在第一步检索中最好的 k 个检索结果被认为是最相关的，这种方式可以在无须人工参与的情况下进行自动查询扩充。出于这个原因，PRF 也叫做**盲目相关反馈**（blind relevance feedback）。

11.3.5　文档先验模型

在上述已经介绍的检索模型中，文档的先验概率被认为是均匀分布的（uniform），也就是说，检索文档的先验概率取值都相同而与特定查询无关。但是，在大多数的应用场景中这种假设是不成立的。例如，文档具有不同的质量和流行度。这些因素必定会影响文档的先验概率，高质量和非常流行的文档直觉上应该被赋予更高的相关度似然率。

不同的检索模型有不同的融合文档先验概率的方法。当使用向量空间模型时，文档的先验概率可以和每一个文档的 IR 分数相乘［1］，也可以和 IR 分数进行线性组合（在此特殊的应用情形下，我们还需要优化线性组合的权重）。在概率模型和语言模型中，文档的先验概率 $P(d)$ 估计作为模型参数的一部分。在没有任何背景知识的情况下，文档的先验概率则假设为所有文档的平均分布。然而，如果具备了一个文档的先验概率模型，我们可以直接用这个模型替换原先使用的统一概率的文档模型。

显然文档的先验概率估计建模依赖于特定的目标应用。例如，在 Web 检索中，包含网页及它们之间超链接关系的 Web 图可用于计算网页的权威度分值，该分值就可以用做文档的先验概率。PageRank［13］和 HITS［14］这两个成熟的算法可以用来计算 Web 图中的权威度分值。另外一个例子是社区门户搜索。用户、使用模式以及其他证据的等级可用于计算文档的先验概率［15］。

11.3.6　模型选择的最佳实践

选择一个检索模型的主要度量方式是检索的效果。不同模型的索引代价和搜索代价都是基本类似的。它们都基于类似的倒排索引并使用同样复杂度级别的算术运算来计算文档分数。

在参考数据集上进行性能比较时，向量空间模型、概率模型和语言模型之间并没有表现出显著差别。2009 跨语言评测论坛（Cross Language Evaluation Forum，CLEF）（关于评测比赛的详细信息请见 11.6 节）中 TEL 数据集上最好的结果由各种不同检索模型获得。例如，在英语文档的检索任务中，开姆尼兹大学（University of Chemnitz）实现的向

量空间模型的平均精确率均值（Mean Average Precision，MAP）[⊖]性能只比都柏林大学（University of Dublin）的语言模型高 0.4%，在法语文档的检索任务中，卡尔斯鲁厄大学（University of Karlsruhe）的概率模型系统则比最好的向量空间模型高出 1.4%［16］。从不同语言上的结果可以看到，差异性也与特定数据集有关。所以，我们无法判定总体上哪个模型的效果是最好的。

377

信息检索模型的选择也依赖于可用的训练数据集（比如，以样例查询相关性估计的形式）以及文档模型的丰富程度。当没有训练数据集的时候，建议采用成熟的标准模型和标准参数，例如 BM25（在 11.3.3 节中介绍）等。这将保证在新的数据集上获得基线效果。当具备了训练数据集后，这些模型的参数可以进行动态的优化和调整。当我们具有丰富的文档模型时，语言模型的加入也可以提供灵活适应方式。在性能不错的搜索系统上，我们还可以利用采用 PRF 的查询扩展手段进一步提升检索质量。查询扩展可以进一步增强模型的召回率。但是如果将模型的精确率作为评测的第一要素，那么查询扩充技术就需要有选择地使用。

11.4 CLIR

CLIR 的任务是从一系列用其他语言（文档集语言）表示的文档集中检索出与某种语言表示的给定查询串（查询语言）相关的文档。

定义 11-1 跨语言信息检索（crosslingual information retrieval） 给定一个文档集合 D，文档集的语言是 l_D（文档集语言），CLIR 的任务是检索出与语言 l_Q（查询语言）所表示的查询串相关的文档列表，并对这些文档列表进行排序。这里 D 是一个单语言的文档集合，即所有在 D 中的文档都是用相同的语言书写。

本质上，我们可以区分 CLIR 的两种不同形式。其一，我们用基于翻译的方法把查询或者文档集合翻译成检索系统支持的语言。这种方法采用标准检索技术把跨语言的信息检索简化为单语言的信息检索任务。第二，可以把文档和查询都映射到一个中间语（概念）空间。相关度函数可以基于这个中间语言的空间来定义。

11.4.1 基于翻译的方法

基于翻译的方法把查询或者文档集合翻译成检索系统所支持的语言。基于翻译的方法可以有不同的方式，一是使用不同的翻译技术，二是翻译的对象可以只是用户查询串或者文档集合，或者是对两者都进行翻译。对于后者我们将介绍几种可供选择的方法。所有，翻译可以包括人工翻译或者 MT 技术的应用。

378

翻译查询串（translating query） CLIR 的默认策略是把查询串翻译成文档集合语言。这种方法可以有效地把 CLIR 问题简化为单语言信息检索。接下来我们罗列了这种方法的一些优点（PRO）和缺点（CON）。

优点：

- 只翻译查询串，而查询串通常是一小段文本。
- 已经建立好的索引可以用于支持任意语言的查询串，只要这些查询串可以被翻译成文档集所使用的语言。

⊖ 平均精确率均值是衡量 IR 系统性能的一种标准评价指标，会偏爱高级查询者把关键词的文档的排序单位。MAP 的严格定义见 11.6 节。

缺点：

- 必须要有一个在线查询翻译。因为检索系统的响应时间是翻译时间和检索时间的叠加，因此，我们需要一个高效的 MT 系统以把系统性能维持在合适的水平。
- 检索系统的准确性依赖于所采用的 MT 系统的质量。

翻译文档（translating document）　一个更进一步的策略是把整个文档集合翻译成查询串的语言，并且为这种语言创造一个倒排索引。这在搜寻有固定查询串的语言的场景中可能会有用，比如用户在只有一种使用语言的门户网站中。下面我们也总结了这种方法的优点和缺点。

优点：

- 这种翻译是预处理的一部分，因为索引将会建立在翻译后的文档之上。在翻译步骤上几乎没有时间的限制，如果对翻译的质量有过高的要求可以采用人工翻译。

缺点：

- 我们必须事先知道并固定查询语言。因为这种语言的索引是特定的，所以不支持在其他语言上进行检索。
- 整个文档集合都需要被翻译，代价比较高。

枢轴语言（privot language）　作为上面两种方法的结合，查询和文档在这种方式中被翻译成枢轴语言。枢轴语言可以是自然语言也可以是人工语言，而且对很多语言都有相应的翻译系统能翻译到该语言。英语最常被当成一种中枢语言，因为大量翻译系统都可以把英语作为翻译的目标语言。因为不需要把查询语言直接翻译成文档语言，所以在没有支持查询串语言和文档集合语言互译的语言资源时，枢轴语言的方法就很有用了。

使用枢轴语言会把 CLIR 精简为标准的单语言信息检索问题，因为针对枢轴语言的 IR
[379] 系统将应用于任意查询语言和文档语言对。然而，检索效果依赖于把查询语言和文档集语言翻译成枢轴语言的质量好坏。这种方法的优点和缺点可以总结如下。

优点：

- 如果不能把查询串语言直接翻译成文档集合语言，可以采用枢轴语言来转换。
- 针对枢轴语言的 IR 系统可以处理任意查询和文档集合语言对。

缺点：

- 查询串的在线翻译和文档的离线翻译（作为文档预处理的一部分）是必需的。

查询扩展　查询扩展技术也能按照以下方法被应用于 CLIR 技术中：

- 翻译前扩展（pretranslation expansion）。查询串在被翻译之前进行拓展，然后被翻译系统所处理。这里的优点是更多的上下文信息可用于翻译。在 CLIR 系统中，这种扩展方式被证明能够提高检索结果的准确性 [37]。
- 翻译后扩展（posttranslation expansion）。等同于在单语言信息检索中使用的查询扩展。在一个 CLIR 系统中，翻译后再拓展查询串能够减少翻译误差，因为错误的翻译能在查询结果的局部分析中被发现（例如，使用 PRF）[17]。

11.4.2　机器翻译

正如前文所述，翻译步骤是基于翻译的 CLIR 所必需的。在使用（特定语言的）倒排索引进行检索之前，需要把查询串或者文档集事先进行翻译。请专业翻译人员进行人工翻译的成本通常比较高。文档的人工翻译不适用于大规模的语料库，并且不可能对查询串进行实时翻译，所以这种方式在反应时间需要以秒来计算的检索系统中（如 Web 检索）可行性不高。这也促使机器翻译在 CLIR 系统中的应用。

在这一章，我们介绍了在 CLIR 系统中使用的两种主要机器翻译技术——基于字典的翻译（dictionary-based translation）和统计机器翻译（statistical machine translation）。

基于字典的翻译 查询翻译的一个直接方法是使用双语词典进行逐项翻译。这里处理候选术语翻译有多种不同策略，包括采用最常见翻译到考虑所有可能翻译等。有趣的是，Oard [18] 的工作表明在 CLIR 系统中不同策略并没有显著差别。当使用所有候选翻译时，各查询术语会利用其翻译概率进行加权。

Ballesteros 和 Croft [17] 提出采用翻译后扩展的方式能减少基于字典查询翻译的翻译错误率。他们的实验使用 PRF 进行查询扩展以剔除由翻译而引入的无关术语，从而提高了检索效果。 [380]

统计机器翻译 与基于词典的翻译相对比，统计机器翻译（SMT）旨在翻译出完整的句子。因此 SMT 原则上能被用于查询串和文档集合的翻译（参见第 10 章对机器翻译的详细讨论）。

大部分目前的 SMT 系统都基于 Brown 等人 [19] 提出的 IBM 模型。这种翻译模型率先在双语平行句对语料库上通过迭代循环训练出一些双语词语对的概率模型。在训练过程的两个子步骤中，双语句对的术语对齐模型和术语翻译模型不断被优化。最终的翻译模型是上述两步迭代优化的产物。通过把短语作为翻译单元，可以进一步提高模型的性能。这种方式不但可以学习和利用单个术语的翻译和对齐，而且还可以学习和利用像 *New York* 这样的短语。采用额外的语言模型（在大型单语语料库上训练出来的）等其他的相关语言学知识，可以提高翻译的效果。

应用 SMT 系统的缺点是，查询串的在线翻译需要消耗比较长的时间，同时也需要一个训练语料库去训练统计翻译模型。但是随着电脑硬件水平的不断提升和在大型分布式在线翻译系统的应用，这个时间上的瓶颈也不存在很大的问题。实际上，最新系统已经足以应付实时查询翻译。但是对于某些特殊的语言对来说，训练语料的缺少也是一个不容忽视的问题 [20]。

11.4.3 中间语言文档表示

另外一种与基于翻译的 CLIR 是利用中间语言表示文档。本质上这种技术是把查询串和文档集合都映射到一个中间语言的概念空间（concept space）。与基于术语的文档表示不同的是，概念表示语义单元（units of thought），因此可以认为是语言无关的。但是把文档集合映射到中间语言所表示的概念空间则需要特定语言的映射函数。例如，这样的映射会依赖于不同语言术语与某一种中间语言的概念术语之间关联程度的量化值（聚合语义相似知识）。通过把查询串和文档都映射到同一个概念空间，信息检索问题被简化为比较查询和文档概念向量。因此我们可以采用一些标准的相似度计算方法，如向量（代表查询和文档的向量）角度的余弦，去计算这两个向量的相似度，进而可以按照相关性分数对检索结果进行排序。下面我们介绍两种已被应用到 CLIR 中的中间语言概念空间方法。 [381]

潜在语义索引（Latent Semantic Indexing，LSI） 在单语情形中，LSI 可以用来识别文本语料库中的潜在主题信息。这些主题与先前描述的概念相对应，通过利用文档中同时出现的术语而被提取出来。这可以通过术语-文档矩阵进行奇异值分解而获得 [21]。潜在主题就对应具有最大奇异值的特征向量，这样就得到了术语向量到**主题向量**（topic vector）的一个映射函数。LSI 最初被应用于文本表示中的降维，以及实现同义词或者相近术语的检索。通过使用平行训练语料库进行训练，LSI 可以应用于 CLIR [22]。在这种情形下，提取的主题

跨越不同语言的术语，而映射函数把所有语言的文档映射到相同的潜在主题空间。

显性语义分析（Explirt Semantic Analysis，ESA）　最近，ESA 被用做为一种基于概念的检索模型[23]。概念的定义一般是明确的而且与某些外部知识资源相关联。通过使用每个概念文本化的描述，可以把文档映射到概念空间。已用于 ESA 的这类资源实例（包括概念及其描述）有维基百科和维基词典。如果概念的文本化描述对对于检索系统所支持的语言都可用，则 ESA 就能够被应用到 CLIR。当使用维基百科作为一种多语言的资源时，跨语言的链接可以被应用于建立多语言的概念定义。Ciniano 等人[24]已经证明 ESA 可以扩展到跨语言的信息检索。

11.4.4　最佳实践

在大多数情况下，对查询串进行翻译是构建 CLIR 系统最灵活的方法。不但能支持任意查询语言，在能翻译为文档或索引语言的任意语言中检索时也可使用同样的索引。然而，它的成功依赖于可用的能实时翻译查询串和文档集的机器翻译系统。假设只有有限的资源，把查询串和文档翻译成具有大量翻译资源的中间语言，可能是最好的方法。

Oard 的工作表明，对查询串的翻译而言，SMT 系统优于基于字典系统。因而推荐使用 SMT 翻译系统（无论商业还是开源系统）。然而，如果检索系统针对特定领域，那么采用领域术语词典进行翻译的系统，其效果要好于采用那些一般性 SMT 的系统[38]。如果我们已经有了领域相关的语料，那么采用基于术语词典的翻译策略再加上检索后扩展技术（postretrival expansion）对于 CLIR 系统来说是最好的选择。

11.5　多语言信息检索

与 CLIR 对比 MLIR（Multilingual Information Retrieval，多语言信息检索）考虑了包含有用不同语言书写的文档的语料库。其定义如下：

定义 11-2　多语言信息检索　给定一个文档集合 D，其所含文档语言为 l_1，…，l_n，MLIR 的任务是检索出与用语言 l_0 表示的查询串 q 相关的文档集合，并且对检索结果进行相关性排序。这些相关的文档可能分布在所有的语言 l_1，…，l_n 中。

如果文档集合由不同语言的文档组成，并且检索系统的使用者至少对部分文档语言中具有一些知识，则 MLIR 就可以应用。在大多数情况下，有些用户的确拥有除了母语（查询所用的语言）以外的某些语言基本的阅读和理解技能。特别是针对那些跨国企业中的用户和一些网络用户而言。如果用户不能理解检索结果的语言，那么可以利用机器翻译的手
[382] 段把检索结果翻译成用户的母语。

大体来说，MLIR 系统是建立在和 CLIR 系统相类似的技术以及 CLIR 系统所采用的相同翻译方法的基础之上的。当然多语言信息检索用到的索引结构和相关度的计算方法相比单语言与跨语言检索所使用的还是有所不同。下面，我们简要描述这些不同，从统一索引到多语言相关的索引均有涉及。如果文档所用的语言事先不知道，那么在预处理阶段就需要语言识别算法识别这些文档所用的语言。

11.5.1　语言识别

语言识别就是标记出文档内容书写所用的语言。下面，我们假设文档是单语言的，也就是文档所包含的内容只涉及一种语言。在这一节的最后我们也会简要地介绍一下关于混合语言类型的文档等更为复杂的情形。

语言识别问题可以简单看成一个标准的离散类别分类问题。其目标种类是一个语言集合，其任务则是把文档分类为这些语言种类中的一种。如果已经有了每种语言的单语言训练语料库，我们可以采用有监督的机器学习方法。基于文档字符 n 元组表示的方法是目前效果最好的分类方法，Cavnar 和 Trenkle [25] 给出了在 14 种语言的语言识别任务上的结果，达到了 93%的准确率。他们通过提取字符 n 元模型（$n=1, \cdots, 5$）为每个文档建立术语向量。这里一个很重要的方面是，我们可以用各种语言的单语的语料去训练各个语言识别任务的分类器而不需要对齐的语料库，因此，这种方法理论上可适用于任意的语言集合。进一步说，因为字符 n 元组建立在字符流的基础之上，并且不存在词分割，因而这种方法也不需要对文档进行预处理工作。这种语言识别方法的准确性依赖于文档的长度，因为越长的文档提供越多该语言的证据。Cavnar 和 Trenkle 的实验表明对于超过 300 个字符的文档而言训练出的分类器，能够获得 99%甚至更高的精确率。

把上述的分类器应用在包含多语言的混合型文档上，会导致无法预测的分类结果。因为这种情况下，由于各个语种语言学特征的相互重叠导致术语或者 n 元组（恰是分类器所用的信息）的语言相关分布会被丢失，所以这些文档不得不事先被分割为单语言组成部分。但是随着对章节、段落以及句子的分割，就会产生较短的文档，在这些短文档上训练分类器的效果就会下降。

11.5.2 MLIR 的索引建立

MLIR 有两种主要方法建立倒排索引，这些方法的主要区别在于，建立一个索引（单一索引方式）还是分别为多个语言建立不同的索引（多索引方式）。单一索引方式为内含多种语言的文档集合建立一个索引。对于构建这样的单一索引我们介绍如下三种技术。

文档翻译（document translation） 首先把所有文档都转换为一种枢轴语言，MLIR 的问题就被精简为 CLIR 问题，此单一索引将包含所有翻译后的文档。 383

语言标记前缀（language token prefix） Nie [26] 建议为每个词元添加语言前缀信息以建立统一索引。这保证具有相同字符的不同语言的术语可以被区分开来。统一索引的术语词典包含了所有语言的术语，Nie 主张这种统一索引可以保留诸如术语频率和文档长度等的术语分布。

概念索引（concept index） 如前讨论，语言无关的概念索引也可用于 MLIR。不同语言的文档被映射到同样的中间语言概念空间，在这种方式下，多语言语料库只需要创建单个的概念索引。

多索引方式为语料库中的每种语言文档都建立了对应的索引，有两种不同的技术：

特定语言索引（language specific index） 在多语言集合中的每种语言的文档被添加到其对应语言的索引中，在这种情况下需要事先识别出各种语言以使应用特定语言预处理。对于单语言文档而言，包含在每个索引中的文档集合是不相交的。对于多种语言表示的混合型文档，不同语言的内容会被添加到各自语言的索引中，这种情况下，文档可能会出现在许多个特定语言索引中。

特定预处理（specific preprocessing） 对于每种语言，一个包含语料库中所有文档的索引会被创建出来。但是，这些文档的预处理方式将随着索引语言的不同而不同。因此对于每个索引，文档被认为与索引的语言相同。这样，每个文档都包含于所有的索引之中。

11.5.3 翻译查询串

索引策略的不同也导致了查询串翻译策略的不同。对于建立在文档翻译或者概念索引基础

上的单一索引结构，查询串的翻译类似于在 CLIR 中提到的查询串翻译策略，请参考 11.4 节。

对于多索引结构，查询串需要被翻译成所有文档语言。根据所用索引不同，翻译的应用方式也不同：

语言标记前缀（language token prefix）　若每个术语都有语言前缀的统一索引，查询将通过连接所有查询翻译并为各查询词元添加语言前缀而形成。标准的 IR 模型可以查询统一索引。

多索引（multiple index）　在多索引的结构上进行检索，需要把查询串翻译成不同的语言，然后再应用到对应语言的倒排索引上进行检索。这种方法检索出来的结果将包含各语言的特定语言排序，还需进一步合并为一个聚合分值并确定出一个聚合排序。接下来我
384 们讨论最重要的聚合模型。

11.5.4　聚合模型

基于多索引结构的检索需要分值聚合模型，因为依据各语言证据而形成的排序必须合并以生成最终排序。给定语言集合 $L=\{l_1,\cdots,l_n\}$，查询 q 和各文档的特定语言分值 $score_i(d,q)$，一个直接的方法是将所有语言分值求和：

$$score(q,d)=\sum_{i\in L} score_i(q,d)$$

然后可以利用这个聚合分值产生文档的一个总体排名。

这一聚合策略的主要问题是分值的潜在非兼容问题。简单对分值进行累加，事实上是假定在各排序中绝对分值表示了同样的相关度水平。但是对大多数检索模型来说，情况并非如此。分值的绝对值将依赖于文档集合的统计数据和术语权重，如文件数量、词元数量、平均文档长度或者文档频率等。对于每个索引这些值都是不同的，因此这些分值未必是可比的。为了解决这一问题，通常在排名模型的聚合前对每个分值进行归一化处理。MLIR 常用的一个标准方法是 Z-score 归一化 [27]，每个排名都使用统计指标对它们的分值进行归一化处理：最小分值、平均分值和标准差。给定以查询和文档相关度评判形式呈现的训练数据，我们可以应用机器学习技术来计算合并分值的最优权重（参见 Croft [28]）。

完整的 Z score 归一化的聚合步骤展示在算法 11-2 中。给定一个排序集 $R=\{r_1,\cdots,r_n\}$，算法对这些排名进行综合计算得到合并排序 r_*。第一步，用最小值、平均值和数值标准差来对每个排名 r_i 进行归一化。第二步，对每个文档在所有排名中的分值进行累加。最后，利用此聚合分值对这些文档进行降序重排并获得合并的排序。

11.5.5　最佳实践

相比 CLIR，将文档按其原始语言进行索引并使用翻译后的查询进行检索，这对 MLIR 而言是很一种最灵活的方法，因为它直接支持新语言的查询串并且可用于新检索和聚合模型。这种索引还具备一个有趣的特性：如果翻译系统被改变或者被更新也没有必要重新建索引。一个对 MLIR 的最新评估结果（即 CLEF workshop 2008 [16] 的结果）显示，目前最好的 MLIR 系统就采用了多语言索引结构，SMT 系统因此被用来进行查询串的翻译，聚合分值的计算用的是 Z-score 归一化策略 [29]。

使用带语言前缀的统一索引对在现成 IR 系统基础上建立起来的 MLIR 系统而言是较好的选择，因为这种方式不影响索引的建立和检索等步骤，只影响了文档的预处理（在词
385 元上增加语言前缀标记）和检索中的调整（翻译及增加语言前缀）。

算法 11-2 基于 Z-score 归一化方法对多非序结果 r_1，…，r_n 进行聚合。对于给定排序 r，$r[i]$ 定义了第 i 名的分值；$score_r(d)$ 则定义了文档 d 的分值；MIN、MEAN 以及 STD-DEVIATION 也都在排序 r 的分值基础上定义

```
R ← {r1, ..., rn}
for all r ∈ R do                          // 归一化
  μ ← MEAN(r)
  σ ← STD-DEVIATION(r)
  δ ← (μ − MIN(r)) / σ
  for i = 1..|r| do
    r[i] ← (r[i] − μ) / σ + δ
  end for
end for

r_c ← {}
for all d ∈ D do                          // 聚合
  s ← 0
  for all r ∈ R do
    s ← s + score_r(d)
  end for
  score_{r_c}(d) ← s
end for
r_c ← DESCENDING-SORT(r_c)
return r_c
```

11.6 信息检索的评价

任何 IR 系统的根本目标是满足用户的信息需求。但是使用者的满意度是非常难以量化的。因此 IR 系统的评价通常是建立在相关度这一概念基础上的，其中相关度由执行系统评价的团队来评估。我们可以采用一个二值的评价指标去定义相关性，文档和查询要么相关，要么不相关；也可以采用一个实值定义相关性，评价系统主要评价检索得到的文档和查询串的相关程度。IR 系统的评估中最常应用的是前者。给定文档和查询间相关与否的规范，IR 系统的目标是最大化所返回文档中相关文档的数量同时将返回的不相关文档的数量最小化。如果 IR 系统对这些结果进行相关程度的排序，则其目标就是把相关度最高的文档排在前面而把不相关的排后面。现在已提出了多个把握这些直觉的评价方法。此外，这几年来几个包含人工相关度评判的参考集也被开发出来了，在这些数据集上的结果是可重现的，因而这些数据允许不同的系统开发者相互比较，并可找出在某个特定任务下表现最好的检索模型、预处理模块、索引策略等。

本节中，我们首先介绍**道格克兰菲尔德范式**（Cranfield paradigm）的实验环境，然后介绍各种评价指标及采用它们的动机，这些指标是基于相关性评估方法的。我们介绍了创建相关性评估的人工和自动方法，最后，我们还概述了能用于评价 CLIR 和 MLIR 系统的数据集。

11.6.1 建立实验环境

用来评价 IR 系统效果的实验设置必须确保该实验是可以重现的，这也是开发克兰菲尔德评价范式的主要初衷。根据这一范式，我们需要一个固定的语料库，还需要设置能够描述需要的信息的最少数量的检索主题以及一个用作 IR 系统输入的查询串。期望被评估的系统能够对文档集建立索引，并返回每个主题（查询）的（排序的）检索结果。为了减少偶然误差，要设置一定数量的话题以便产生稳定的统计结果，通常建议设置至少 50 个话题。

每一话题都有一个标准答案，定义了集合里的相关文档 [2]。这里相关性的概念通常

286

是二值的，即文档和给定查询是否相关。使用该标准答案，我们可以通过检查 IR 系统所返回的文档是否与该话题相关以及是否所有相关文档都被该系统检索出来而实现对其评价。这些概念可通过特定的评价指标实现量化，通常应该是希望被最大化的（参见 11.6.3 节）。由于使用者的满意度通常很难被量化，相应的评价实验也很难重现，所以使用标准答案（包含预定义的话题和给定的相关度评估方法）的评价策略是我们经常采取的策略。

11.6.2　相关性评估

信息检索实验通常需要所谓的相关性评估（relevance assessment）以建立标准答案。虽然对诸如原始 Cranfield 语料库的较小数据集，有可能采用人工评估者穷举对各话题的所有文档进行检查的方式，但这种方式对大文档集合就难以实施了[2]。因此所谓的结果汇集（result pooling）技术就被采用以避免评估者浏览每个话题的所有有关文档集合。其思路是，汇集多个 IR 系统排名靠前的文档并进行评价，对于不同系统排名在前 k 位的文档才会被考虑，k 通常是 100 或者 1000。由于相关性评估对于不同的评估者差异较大，每个文档/主题对通常都会由多个评估者来评判。标准答案中所包含的最终相关性决策是一个聚合值，例如可以用基于大多数表决投票机制来确定。标注者间的一致性指标，例如 kappa 统计量（kappa statistic）[2]，则是实验有效性的标志。低一致性可能来源于信息需求本身定义的模糊。

387

通过在汇集中引入多个系统，研究者试图减少针对单个系统的相关性评分的偏见。另外，测试集合应该是充分完备的，以便此相关评价能够重用于未在初始汇集中出现的 IR 技术或者系统。

CLIR 和 MLIR 系统的另外一种评估方法是**配对检索**（mate retrieval）设置。该设置通过使用平行或对齐数据集（包含文档及其到所有相关语言的译文），避免了对相关性评判的需求。评测所用的主题同语料库的一个数据集合相对应。该主题的配对（mate），即文档在不同语言中的等价物，则被认为是唯一相关的文档。如此，各系统的目标就是恰好检索出这些配对。因此，这种标准答案能够被自动构建。使用此标准答案的评价指标明显被低估了，因为其他文档也有可能是相关的文档。

11.6.3　评价指标

在给定了一个包含相关评价方法的标准答案和确定的数据集后，我们需要一些评价指标去衡量信息检索系统的效果。信息检索中最常用的评价指标是**精确率**和**召回率**。精确率测量检索系统的结果文档集中相关文档所占的比重，召回率则测量相关文档被检索系统实际检索出来的比重。

精确率和召回率的计算方式可以依据表 11-1 给出的针对单个查询的检索结果列联表（contingency table）来解释。精确率 P 和召回率 R 的定义如下：

表 11-1　针对单个查询的检索结果列联表

	相关	不相关
检出	TP	FP
未检出	FN	TN

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN}$$

要选择一个合适的评估方法，了解信息检索系统的应用场景是关键。有些场景下，例如编写报告等，用户可能需要读完全部的文档；而另外一些场景下，比如 Web 上的即时搜索，用户可能只查阅排在最前面的文档集。这些特别极端的例子清楚地说明了评价指标的选择与 IR 系统的使用方式密切相关。

对精确率和覆盖面（即需要返回所有的相关文档）都同样重要的检索系统（未必就是 Web 搜索）来说，平均精确率（Average Precision，AP）就是一个合理的选择。平均精确率对排序中各特定位置的精确率进行平均，具体说来，这些位置就是所找到的相关文档的位置。

假设对 n 个文档中的 r 个进行排序，相关文档集为 REL，$rel:D\rightarrow\{0,1\}$ 是判断文档是否相关的二值函数（相关就是 1，不相关就是 0），P_k 表示截断等级（cutoff level）为 k 时的精确率，则 AP 可按如下公式计算： 388

$$AP(r)=\frac{\sum_{i=1}^{n}P_i\times rel(r_i)}{|REL|}$$

平均精确率均值（Mean Average Precision，MAP）在所有的文档主题上对 AP 求平均值，可以用来评价 IR 系统的整体性能。

MAP 和其他的评价指标，例如 bpref [31] 或者 infAp [32] 的一个共同特性是它们主要关注对每个查询（最大到预先确定的上限，通常是 1000）所检出的所有文档的整体检索性能。前已提及，在用户需要系统检索出尽可能多的相关文档的这种应用场景下，这种评价方式是比较合理的。但是用户可能不会浏览 IR 系统所检索出的所有这 1000 篇文档。所以其他评价指标也被提出以针对用户仅检查有限（排名最前的）文档子集的情况下评估检索系统的正确性。例如，可以在一个给定的排名上计算精确率（记为 P@r），排名 10 以上的精确率（P@10）通常用于测量首先检出的文档集的准确率。为使排名最靠前的文档正确，我们经常会使用最相关文档的平均排名倒数。

有时，相关性评估包含多个相关性等级并结合了类似归一化折扣累计增益（Normalized Discounting Cumulative Gain，NDCG）[33] 的指标，该指标将高相关度文档的排名先于低相关度文档的偏好纳入考虑。

11.6.4 已有数据集

通过重用包含公共文档语料库、话题、查询和相关性评估方法的共享数据集，IR 实验才变得可重现，结果才变得可比较。信息检索领域中，已经出现了一些初步定义检索任务并提供相关数据集的评测。

除了评测任务提供的数据集之外，平行语料库也是 CLIR 和 MLIR 信息检索系统所需要的。平行语料库资源可以用来训练统计模型，例如训练 SMT 系统或者训练 LSI 模型来识别跨语言潜在概念。此外平行语料库也可以被当作测试集，例如，在配对检索的应用场景中。

1. 评测任务

文本检索会议（Text REtrieval Conference，TREC）每年都举行，其目标是建立对 IR 系统进行系统评价和比较的平台。TREC 组织了不同的 Track（表示不同的信息检索任务，例如特定搜索、实体搜索或特殊领域搜索等）。TREC 为每个 Track 都提供了数据集和话题/查询（以及相关性评价），以便参与者可以用它们进行系统的开发和调节。TREC 从 1992 开始采用汇集技术以支持使用测试集不完全评估方式对 IR 系统交叉对比。TREC 389
等相关会议的举办是为了弘扬竞争精神，不同的开发组在共享的任务和数据集上开发出系统来相互竞争，这种方式可以使得竞争的结果具可比性。这些评测任务使研究者能够了解在特定任务中哪种信息检索模型或参数调节方法等效果更好，因而真正从本质上起到了推动科学研究进步的作用。当然，TREC 的主要目标不只是系统的竞赛，它也为研究社区提供了共享的数据集以便用于系统实验、对比并重现结果。TREC 最主要的评测项目是英文文档

的单语检索任务，所以 TREC 公布出来的相关数据集合只包含英文的主题和文档。

跨语言评测论坛（Crosslingual Evaluation Forum，CLEF）由欧洲地区的 TREC 建立，重点关注多语言信息检索。2000～2009 年，CLEF 的特定检索 track 中使用了不同的数据集，例如欧洲新闻语料和拥有 14 种语言的通讯社文档，也包含用英语、法语和德语表示的欧洲图书馆著录条目和波兰新闻语料库的 TEL 数据集。所有数据集都给出了不同语言的主题，从而使它们适合用于 CLIR 和 MLIR 任务。TEL 数据集也包含不同语言字段的混合文档。

IR 系统的 NII 测试集（NII Test Collection for IR Systems，NTCIR）主要举办一系列亚洲语言（包括日语、汉语和韩语等）的信息检索评测会议。已经发布了包含日语和英语表示的科学文摘数据集以及用中文、韩语、日语和英语表示的新闻文章（包括不同语言的主题）的数据集。此外，日-英专利检索数据集也已经公开。

信息检索评测论坛（Forum for Information Retrieval Evaluation，FIRE）专注印度语的评测。它已经发布了由孟加拉语、英语、印第语和马拉地语的网络论坛和相关邮件列表构建而来的语料库。主题使用的语言包括孟加拉语、英语、印第语、马拉地语、泰米尔语、泰卢固语和古吉拉特语。

2. 平行语料库

JRC-Acquis 是一个从欧盟现行法（Acquis Communautaire），即欧盟成员国使用的欧盟法律文献中提取出来的文档集合。它由以下 22 种语言的平行文本组成：保加利亚语、捷克语、丹麦语、德语、希腊语、英语、西班牙语、爱沙尼亚语、芬兰语、法语、匈牙利语、意大利语、立陶宛语、拉脱维亚语、马耳他语、荷兰语、波兰语、葡萄牙语、罗马语、斯洛伐克语、斯洛文尼亚语和瑞典语。

http://langtech.jrc.it/JRC-Acquis.html

Multext Dataset 是源于欧洲共同体官方期刊（Official Journal of European Community）的文档集，包括如下 5 种语言：英语、德语、意大利语、西班牙语和法语。

http://aune.lpl.univ-aix.fr/projects/multext/

Canadian Hansards（加拿大美国国会议事录）包含用英语和法语表示的第 36 届加拿大议会（Canadian Parliament）的官方记录（Hansards，国会议事录）的对齐文本块（句子或更小的片段）。

http://www.isi.edu/natural-language/download/hansard/

Europarl（欧洲议会）是一个平行语料，包含了用以下语言表示的 1996～2009 年的欧洲议会（european parliament）文集：丹麦语、德语、希腊语、英语、西班牙语、芬兰语、法语、意大利语、荷兰语、葡萄牙语和瑞典语。

http://www.statmt.org/europarl/

11.6.5　最佳实践

我们已经明确提出，对于 CLIR 和 MLIR 系统的理想评估方法和指标依赖于系统的应用场景。

如果将系统设计为以研究为目的，并且希望在一个特定研究问题上提高目前最好的信息检索系统，那么最好使用公开的数据集和标准的评价指标以确保系统与现存的系统具有可比性。许多时候，存在标准答案可用于计算评价指标。

如果检索系统涉及真实用户应用程序的一部分，那么数据集通常就该根据具体的任务进行定制。为了评估系统好坏，需要定义能覆盖用户预期信息需求的所有主题，并且要确

定相关性评估以创建合适的标准答案。如前所述，汇集技术可以有助于减少用于制定相关性评价的标准答案的工作量。

在一般情况下，我们提倡使用 MAP 或者平均排名倒数（mean reciprocal rank）等标准评估方法来评估检索系统的性能。当然也需要针对不同的情形制定不同的评价标准以达到预期的结果，比如要求排名靠前的文档具有高精确率这种情况。

11.7 工具、软件和资源

开发一个完整的 IR 系统包含许多不同的方面，如对预处理步骤的实现，倒排索引的文件结构和有效的检索算法。因此从零开始建立一个 IR 系统需要巨大的努力。使用已有的工具以降低构建检索系统的成本是非常必要的。

在一个具体的项目中，可能是这样的情况：只有检索模型或者排序函数需要改写，而系统其他组件则使用现成的。幸运的是，有几个库提供了标准 IR 组件甚至某些组件可被替换的完整框架。 [391]

下面我们挑选了一些支持开发 IR 系统的工具和软件库。我们重点关注那些广泛使用并有社区支持的成型工具。最受欢迎的 IR 框架是 Lucene，它也封装了许多本文介绍的其他工具。

1. 预处理

内容分析工具（Content Analysis Toolkit，Tika）是一个用 Java 实现的，用来从不同文件类型（例如 PDF 或者 DOC）的文档中提取内容的工具。它也支持文件类型的检测。Tika 源于 Lucene 项目。

http://lucene.apache.org/tika/

雪球词干分析器（Snowball Stemmer）是几种欧洲语言的词干分析器，它运行得非常快并且支持停用词去除。其所支持语言的停用词列表在项目网站上可以下载。

http://snowball.tartarus.org

HTML 分析器（HTML Parser）是用于解析 HTML 文件的工具，它能够忽略网页中的标签以及与语义内容无关的部分，从而提取出文本内容。

http://htmlparser.sourceforge.net/

BananaSplit 是一个基于词典资源的德语复合词拆分工具（compound splitter）。

http://www.dmi.de/niels/s0y/pages/bananasplit.html

翻译 http://www.statmt.org 门户网站是一个获取统计机器翻译系统信息的极佳入口。它提供了软件和用于训练翻译模型的数据集。

谷歌翻译服务[9]，作为一个商业 SMT 系统的例子，提供、支持多种语言间翻译的 API。然而，因为机器翻译在信息检索框架中只是预处理的一部分并且一般情况下没有深度集成到检索系统中，所以任何商业翻译系统都可以被嵌入到 CLIR 或者 MLIR 系统。

2. IR 框架

Lucene 是用 Java 实现的、广泛应用的 IR 系统。它是遵循 Apache 许可证的开源软件，因此能够被用在商业应用和开源项目中。Lucene 已经是成熟系统并且被应用到各种应用中。它的主要特征是可伸缩性和可靠性，这是以降低灵活性为代价的，使得更改其程序组件变得困难。举例来说，在 Lucene 中索引的建立依赖于所选择的检索模型，所以在不重建索引的情况下检索模型是不能更改的。

---

⑨ http://translate.google.com

http://lucene.apache.org

Terrier 和 Lemur 都是用于研究目的的检索工具。Terrier（由 Java 实现）和 Lemur（由 C++实现）都是灵活的 IR 框架，很容易拓展和修改。因为考虑的侧重点不同，所以它们在稳定性和查询效果上不能和 Lucene 相比。

http://terrier.org

582 http://www.lemurproject.org

3. 评价系统

trec_eval（源于 TREC）是一个用于依据标准答案为给定文档排序计算各种评价指标的工具。它的输入为有着简单语法的纯文本文件。按照 TREC 的格式标准输出文件使 trec_eval 能用于任何 IR 系统。前面介绍的 IR 框架也支持 TREC 格式的输出。

http://trec.nist.gov/trec_eval/

11.8 总结

本章我们对实现可用于访问不同语言文档的 IR 系统的方法进行了综述。我们区分了两类问题：跨语言和多语言信息检索。CLIR 用于在给定某特定语言话题的基础上检索出另外一种语言的文档，而 MLIR 则可应用于多语言文档集和不同语言的话题。

根据所支持的语言不同，我们分别讨论了所需的预处理方法（词元化，词干化等）。我们还进一步讨论了在开发 CLIR 和 MLIR 系统时能重用的信息检索基本方法。我们还特别证明了大部分信息检索的标准文档模型都可以在 CLIR 和 MLIR 中重用。我们讨论了 CLIR 和 MLIR 的两种主要方法：基于翻译的方法和基于中间语言表示的方法。我们对此还讨论了不同的机器翻译技术以及它们应如何应用于 CLIR 和 MLIR 系统中。统计机器翻译的发展使我们可以利用它把查询甲实时翻译成多种其他语言，从而把 CLIR 和 MLIR 转换为一个标准单语言检索任务。我们还讨论了识别文档语言的方法，如果需要建立和维护多种特定语言的索引结构，这种识别就是非常重要的一步。我们还简要讨论了将不同特定语言索引下获取的检索结果分值进行聚合以得到全局分值和排序的问题。此问题在多语言检索中是一个重要的问题。

因为系统评测在信息检索领域也是至关重要的，所以我们介绍了 IR 系统是怎样进行评测的。我们特别介绍了人工和自动的相关性判定方式，并介绍了标准的 IR 系统评测指标。最后我们概述介绍了关于标准数据集、评测竞赛、软件库以及一般资源。

致谢

本工作得到德国研究基金（German Research Foundation, DFG）Multipla 项目（项目编号：38457558）和欧洲委员会（European Commission）Monnet 项目（项目编号：
583 FP7-ICT-4-248458）的资助。

参考文献

[1] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. Boston: Addison-Wesley, 1999.

[2] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. New York: Cambridge University Press, 2008.

[3] The Unicode Consortium. *The Unicode Standard, Version 5.2.0*. Mountain View, CA: The Unicode Consortium, 2009.

[4] Working Group on Romanization Systems. "United Nations Group of Experts on Geographical Names (UNGEGN)." updated 2011. http://www.eki.ee/wgrs/.

[5] H. Zhang, Q. Liu, X. Cheng, H. Zhang, and H. Yu, "Chinese lexical analysis using hierarchical hidden markov model," in *Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing*, vol. 17, pp. 63–70, 2003.

[6] F. Peng, F. Feng, and A. McCallum, "Chinese segmentation and new word detection using conditional random fields," in *Proceedings of the 20th International Conference on Computational Linguistics*, p. 562, 2004.

[7] P. McNamee and J. Mayfield, "Character *N*-Gram tokenization for European language text retrieval," *Information Retrieval*, vol. 7, no. 1, pp. 73–97, 2004.

[8] C. Monz and M. de Rijke, "Shallow morphological analysis in monolingual information retrieval for dutch, german, and italian," in *Evaluation of Cross-Language Information Retrieval Systems* (C. A. Peters, ed.), pp. 1519–1541, Berlin: Springer, 2002.

[9] V. N. Anh, O. de Kretser, and A. Moffat, "Vector-space ranking with effective early termination," in *Proceedings of the 24th International Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 35–42, 2001.

[10] C. J. van Rijsbergen, *Information Retrieval* (2nd ed.). London: Butterworths, 1979.

[11] S. E. Robertson and S. Walker, "Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval," in *Proceedings of the 17th International Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 232–241, 1994.

[12] J. M. Ponte and W. B. Croft, "A language modeling approach to information retrieval," in *Proceedings of the 21st International Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 275–281, 1998.

[13] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 107–117, 1998.

[14] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM*, vol. 46, no. 5, pp. 604–632, 1999. [39c]

[15] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne, "Finding high-quality content in social media," in *Proceedings of the International Conference on Web Search and Web Data Mining (WSDM)*, pp. 183–194, 2008.

[16] N. Ferro and C. Peters, "CLEF 2009 ad hoc track overview: TEL & Persian tasks," in *Working Notes of the Annual CLEF Meeting*, 2009.

[17] L. Ballesteros and W. B. Croft, "Phrasal translation and query expansion techniques for cross-language information retrieval," in *Proceedings of the 20th International Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 84–91, ACM, 1997.

[18] D. Oard, "A comparative study of query and document translation for cross-language information retrieval," in *Machine Translation and the Information Soup* (D. Farwell, L. Gerber, and E. Hovy, eds.), pp. 472–483, Berlin: Springer, 1998.

[19] P. P. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer, "The mathematics of statistical machine translation: Parameter estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

[20] P. Resnik and N. A. Smith, "The web as a parallel corpus," *Computational Linguistics*, vol. 29, no. 3, pp. 349–380, 2003.

[21] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society of Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[22] S. T. Dumais, T. A. Letsche, M. L. Littman, and T. K. Landauer, "Automatic cross-language retrieval using latent semantic indexing," in *Proceedings of the AAAI Spring Symposium on Cross-Language Text and Speech Retrieval*, pp. 15–21, 1997.

[23] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using Wikipedia-based explicit semantic analysis," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1606–1611, 2007.

[24] P. Cimiano, A. Schultz, S. Sizov, P. Sorg, and S. Staab, "Explicit versus latent concept models for cross-language information retrieval," in *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1513–1518, 2009.

[25] W. Cavnar and J. M. Trenkle, "*N*-gram-based text categorization," *Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval (SDAIR)*, pp. 161–175, 1994.

[26] J. Nie, "Towards a unified approach to CLIR and multilingual IR," in *Proceedings of the Cross-Language Retrieval Workshop at SIGIR*, pp. 8–14, 2002.

[27] J. Savoy, "Data fusion for effective European monolingual information retrieval," in *Multilingual Information Access for Text, Speech and Images*, pp. 233–244, 2005.

[28] W. B. Croft, "Combining approaches to information retrieval," in *Advances in Information Retrieval*, pp. 1–36, 2000.

[29] J. Kraten, "Chemnitz at CLEF 2009 Ad-Hoc TEL task: Combining different retrieval models and addressing the multilinguality," in *Working Notes of the Annual CLEF Meeting*, 2009.

[30] C. Cleverdon, "The Cranfield tests on index language devices," *Aslib Proceedings*, vol. 19, no. 6, pp. 173–194, 1967.

[31] C. Buckley and E. M. Voorhees, "Retrieval evaluation with incomplete information," in *Proceedings of the 27th International Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 25–32, 2004.

[32] E. Yilmaz and J. A. Aslam, "Estimating average precision with incomplete and imperfect judgments," in *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, p. 112, 2006.

[33] K. Jrvelin and J. Kekliinen, "IR evaluation methods for retrieving highly relevant documents," in *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 41–48, 2000.

多语自动文摘

Frank Schilder, Liang Zhou

12.1 概述

自动文摘已经成为计算语言学中一个十分活跃的领域，研究者们从多种角度研究了这个问题。过去的研究主要关注单语文本，但最近几年里，多语自动文摘吸引了不少研究兴趣，多语文本也被用在了文摘系统中。

自动文摘可以划分为单文档和多文档两种。摘要可能是特定查询驱动的，也可能是为了提供文档（或文档集）的主要内容。根据不同的目的就有不同的摘要，例如，**信息型**（informative）摘要是输入文本中重要事实的一个压缩版本（如期刊论文的摘要），摘要也可能仅**指示**（indicative）了输入文本中的主题而未提供更多的细节（如科技论文的关键词）。另一种类型的摘要以评论的形式出现，这样的一个**评论**（evaluative）摘要一般会通过比较和输入文档相似的文档来给出观点。**详细摘要**（elaborative）则会提供一个人型文档或者是多个相关文档中比较多的细节，这样能帮助这类文档或相关文档的导航，例如维基百科[1]。

更基本地，我们可以通过自动文摘不同的实现方式将文摘分为文档的**摘录**（extract）或者文档的**摘要**（abstract）。摘录通过提取文档中最重要的部分，可能也会包含少量次要的部分来进行文摘。摘要则描述了对文档内容的总结，未必直接包含文档内容的原句。大多现今的自动文摘系统是通过摘录来实现的，但是也有一部分系统试图产生摘要[2]或者通过句子压缩以保留一个句子（或更多内容）中的重要部分来做文摘[3]。

最近的一些研究细节包括对书籍目录信息的摘要、更新式文摘（即只报告发展中事件的最新变化）或导引式文摘，目标是根据文档的类型从源文档中提取语义信息（例如事故或自然灾害）。

多语自动文摘继承了单语自动文摘的特征和挑战，并增加了一个维度。按粗略的定义，所谓多语自动文摘就是涉及超过一种语言的自动文本文摘。

具体说来，文摘系统能对一种源语言（例如阿拉伯语）进行处理，并用目标语言（例如英语）来呈现摘要结果。我们把这种特别的多语文摘叫做**跨语际文摘**（translingual summarization）。

更复杂的文摘叫做**跨语言**（crosslingual）文摘，这种文摘任务的源语言为多种语言，摘要结果用一种（或多种）目标语言呈现。

跨语言文摘是一种更具有挑战性的任务，因为它要聚合来自不同语言的多个源文档。所有的多语文摘，不管是否涉及两种或多种更多源语言或目标语言都面临许多问题。

第一个问题就是跨文档共指消解。命名实体在不同的语言中常被翻译成不同的结果。例如，Al-Qaida、al-Qa'ida、el Qaida、al Qaeda 是德语词 El Kaida、英文 Al-Qaeda 的不同翻译。文摘系统必须对这些变体进行规范化并把它们映射到同一个实体。

相似地，多语种指代消解问题也需要进行处理。语言对数和性一致性的编码是不同

的，英语就没有语法性的概念，但是其他印欧语系却不一样。例如，其他印欧语言会用具有不同性的代词对不同的先行词进行指代（例如，法语：la lune (FEM)-elle；德语：der Mond (MASC)-er）。

多语文摘系统可能会遇到的另一个问题是，不同语言通常使用不同的语篇结构。不同语言的篇章关系也许也是不同的，因此，用目标语言生成连贯的摘要也是困难的。

一个更复杂的问题是如何对语言相关的概念进行摘要。例如，在不同语言中要对法律的概念进行摘要，是十分困难的，甚至是不太可能的。

上述许多问题在单语自动文摘中已经存在（例如指代消解），但是由于不同的语言有不同的指代、篇章结构、概念，导致这些问题在多语文摘里更加严重。多语自动文摘系统的质量也因此取决于机器翻译系统的质量，目前还远不能达到完美的水平。在机器翻译中最小错误率策略用来最小化前述这些问题的影响。自动文摘系统也可以减小这些问题的影响，例如，包含基于容易提取特征（easy-to-extract）的知识贫乏（knowledge-poor）方法来处理指代消解问题 [4] 或基于图的方法来根据基于词的相似度指标对相似的句子进行聚类 [5]。

历史　最早的自动文摘系统之一是 1998 年 Ed Hovy 和 Chin-Yew [6] 开发的 SUMMARIST，它能生成不止英语一种语言的摘要。该系统可以从英语、西班牙语、法语、德语以及印度尼西亚语报纸中生成摘录[①]。

398

2001 年，SummBank 第一个面向研究的基于跨语言文摘框架的系统被开发出来。资源来自 Johns Hopkins Research Workshop [7]，包含中、英文 360 个文档和 40 个人工新闻聚类[②]。

2002 年，欧盟资助的项目 MLIS MUSI（Multilingual Summarization for the Internet）可以对英语和意大利语科技文章进行多语自动文摘 [8]。

几年以后，哥伦比亚大学开发了 NewsBlaster 自动文摘系统，它能使用户用不同的语言浏览互联网上多个网站的新闻 [9][③]。

2005 年，语言资源联盟（Linguistics Data Consortium，LDC）的多语自动文摘评测（Multilingual Summarization Evaluation，MSE）项目进一步促进了研究[④]。该评测使用了哥伦比亚大学的 NewsBlaster 主题聚类系统所生成的 25 个新闻话题，文摘任务是获取英语和阿拉伯语的新闻信息。但是，标注者通过英文新闻文档所生成的 100 词摘要并未直接来自阿拉伯语源文档，而仅来自它们的英文翻译。

多语文摘研究领域的另一个里程碑事件是 2006 年由 Horacio Saggion 发布的基于 GATE 的摘要系统 SUMMA [10，11]。利用开放架构的 GATE 系统 [12]，该系统直接把一些语言工具（例如分词程序、分句程序）集成一体化并可支持拉脱维亚语、瑞典语和芬兰语[⑤]。

最近的论文主要研究多语言文摘的特定问题或跨语际和跨语言文摘系统的方法，包括 Mani，Yeh 和 Condon 提出的通过不同语言寻找名字的系统，该系统把英语的名字匹配到中文，F 值可达到 97.0 [14]。Mille 和 Wanner [15] 则描述了另一个需要处理多语言的摘要问题并提出了一个处理不同语言专利的系统。

---

① 该系统的早期版本还能生成阿拉伯语和日语文摘。

② http://www.summarization.com/summbank/。

③ http://newsblaster.cs.columbia.edu 网站从来自英语新闻网站的新闻和图片进行了摘要。

④ http://projects.ldc.upenn.edu/MSE/。

⑤ 这些语言资源由 Charry 项目开发 [13]。

最近几年，自动文摘系统生成的摘要不仅是英语摘要。Lcuski 等人 [16] 提出的系统能把英语新闻头条翻译成北印度语。Orasan 和 Chiorean [17] 则使用最大边缘相关（Maximal Marginal Relevance，MMR）方法对罗马利亚新闻进行摘要。

12.2 自动文摘方法

12.2.1 传统方法

自动文摘是为了满足用户的信息需求，通过抽取并修改源文档的材料，创造一个更简洁的反映源文档内容的短文。如果该短文是一字不差地提取（或仅有最小化的修改），这样的文摘就叫做摘录；如果该短文是在摘要层次上获取内容的主旨，这样的文摘就叫做摘要。现今的大多自动文摘系统是摘录而非摘要。 [359]

大量的研究关注如何解决用户的需求，这导致了不同类型的文摘任务。例如，多文档自动文摘和基于查询的自动文摘。多文档自动文摘的摘要来自涉及相同主题的多文档，基于查询的自动文摘是根据用户的查询申来进行摘要而不提供通用目的的摘要，基于查询的自动文摘既可以基于单文档也可以基于多文档。

一般而言，每一个自动文摘系统都可分为三个步骤：

分析（analysis） 分析源文本，生成一些内部表示。该表示可以是一个特征向量的集合（例如句子中最常见词的计数），也可以是描述其内容的逻辑表示。对于一个跨语种的系统，这部分特别重要，因为这种表示必须对不同语言有一定的兼容性。

转换（transformation） 对这种内部表示进行修剪和压缩（例如，根据某个分值函数将句子进行排序）。同样，依赖内部表示选择的方式不同，该转换可能是语言相关的。

实现（realization） 文摘的目的是生成一个应源文档更短的文本。一种简易的方法是根据得分函数输出 n 个最高得分的句子，但是要生成一个连贯的摘要，其他的操作必不可少（例如共指消解）。如果多语自动文摘前期没有处理多语问题，则为了使文摘能用目标语言表示，它必须采用机器翻译部件，或者，直接从摘要语义表示生成目标语言文摘。

自动文摘的研究最早能追溯到 20 世纪 50 年代末 Luhn 的工作 [21]，Luhn 调查了句子中常见术语的影响，并提出了一个用于计算文档中每个句子得分的得分函数。

其他早期的自动文摘系统主要都是基于表面特征来提取文本中的重要句子，一般来说，文档开始（或者结尾）的句子经常是十分重要的 [22]。因此，在文档中句子的位置常常是决定句子重要性的一个很好的特征，因为作者喜欢把重要的句子放在文章中显著的位置。

许多早期的方法，也包括一些近期的做法，经常用下面的特征来进行句子的提取 [21, 22, 23]。

- 诸如 in summary（总之）这样的指示性短语。
- 术语的分布。 [403]
- 与标题重叠的词。
- 句子在文本、段落中的位置等。

这些通用特征也可以很容易地应用到多语自动文摘上。术语的分布和位置多半是语言无关的特征，其中位置信息是体裁相关的。例如，一般在新闻里重要的句子都在篇首，而法律文本一般都在篇尾进行信息总结。

通过基于特征提取的摘要方法生成的摘要并不总是连贯的。为了处理这一问题，自动文摘系统会结合可预测连贯性的语篇理论。由于语篇常常可看作一个图结构，所以下一小

节将讨论基于图的方法。

12.2.2　基于图的方法

这一节讨论基于图的文本模型以及这种表示方法如何提高文本自动文摘的质量。一方面，诸如修辞结构原理（Rhetorical Structure Theory，RST）[24] 的语篇理论通过树结构对文本连贯进行了建模；另一方面，诸如 PageRank [25] 的基于图的排序方法已被证明有助于根据句子的重要性计算句子的得分。

前一种方法需要各种语言的深度语言知识，基于图的方法则可以将文本转换为图表示，图中的节点是文本中的句子，节点的连线是句子间相似度的权重。本节的第二部分主要关注使用类似 PageRank 这样的计分机制进行摘要提取的方法。

1. 连贯与衔接

从源文档自动摘录形成的自动文摘在语言学质量上往往是比较糟糕的。因为从文本中摘录出句子形成的文摘，句子之间的指代关系（比如代词）和篇章结构（比如诸如 therefore 的语篇标记）都会被破坏，这使得文摘不连贯，很难读懂。

提高语言学质量的方法有好几种。我们先讨论两个重要的概念。第一，衔接（cohesion），它是句子之间的语义关系 [26]。典型的衔接是通过句子之间的指代关系（包括前指和后指）来表示的。其他支持衔接的语言学现象包括替换、省略、词语搭配。

[422] John went to the bank. He wanted to swim in the river.

这两句话的联系是十分紧密的，因为我们能通过指代关系知道 he 就是 John，也能通过词语搭配，从 swim 和 river 两个词的词汇搭配准确地推测出 bank（即河岸）的意思。

与衔接概念相关但范围小于句子间联系的概念就是连贯（coherence）。连贯常用在语篇概念建模中，表示整个文本中句子是如何联系的。自动文摘需要是连贯的，这样有助于用户的理解。因此，摘录的句子如何排序，如何修改来提高文摘的可读性是一个很重要的问题。

为了最大化文摘的连贯性，目前已经提出了几种方法 [27, 28, 29]。它们几乎都是建立在类似 RST 的语篇理论上 [24]。RST 的主要假设是基于通过修辞关系来联系的文本段的观察。段间的修辞关系可以经由语篇标记（例如，“因为”）明确标记出来，也可以根据上下文推导出来。修辞关系能表示事件的因果关系、描述某情景，或者把叙述移前。

Marcu 和 Echihabi [30] 提出了一个基于 RST 的修辞分析器并用于自动文摘。RST 的一个核心思想就是语篇树。树的节点可以合并文本段，有下面两种类型的节点：

- “核与卫”（主从关系）：核包含比较重要的信息，卫提取的信息支撑核。例如，阐述关系（ELABORTION）就是主从型的核卫关系，比如：

  Lactose is milk sugar; the enzyme lactase breaks it down.

- “核心与核”（并列关系）：另一方面，多核关系 CONTRAST 在两个同等重要的事实间形成对比，比如：

  For want of lactose, most adults cannot digest milk. In populations that drink milk, the adults have more lactase, perhaps through natural selection.

图 12-1 就是下面有关火星探测例文的 RST 语篇树，由 Marcu [31] 分析得到：

[With its distant orbit[1]] [—50 percent farther from the sun than Earth —[2]] [and slim atmospheric blanket,[3]] [Mars experiences frigid weather conditions.[4]] [Surface temperatures typically average about 60 degrees Celsius (-76 degrees Fahrenheit) at the equator[5]] [and can dip to -123 degrees C near the poles.[6]] [Only the midday sun at tropical latitudes

is warm enough to thaw ice on occasion,[7]] [but any liquid water formed in this way would evaporate almost instantly[8]] [because of the low atmospheric pressure.[9]] [Although the atmosphere holds a small amount of water,[10]] [and water-ice clouds sometimes develop,[11]] [most Martian weather involves blowing dust or carbon dioxide.[12]] [Each winter, for example, a blizzard of frozen carbon dioxide rages over one pole,[13]] [and a few meters of this dry-ice snow accumulate[14]] [as previously frozen carbon dioxide evaporates from the opposite polar cap.[15]] [Yet even on the summer pole,[16]] [where the sun remains in the sky all day long,[17]] [temperatures never warm enough to melt frozen water.[18]]

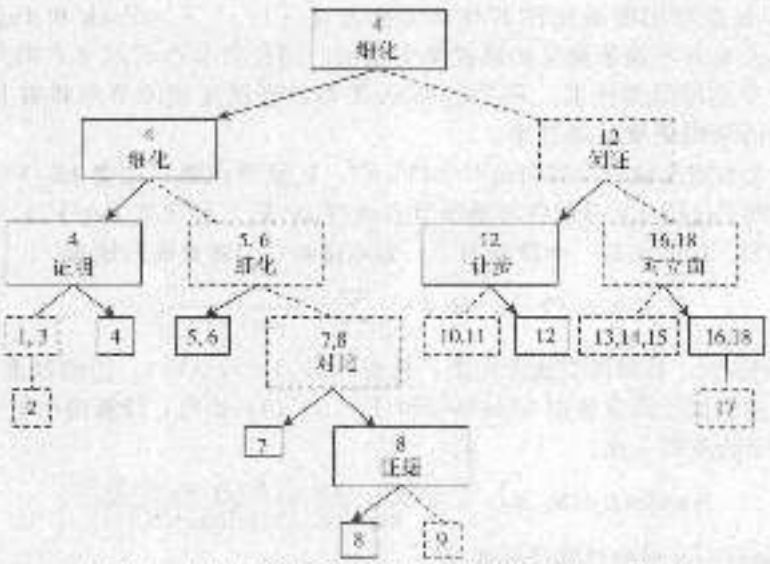


图12-1 火星探测例文的RST结构（来源：Marcu [31]）

这个文本段用方括号分成子句。例如，子句12（即 most Martian weather involves blowing dust or carbon dioxide.）描述了子句4（即 Mars experiences frigid weather conditions.）的一个例子。这里子句4是核，子句12是卫。"核与卫"节点定义为，如果卫节点从语篇树上删除，整个语篇仍然保持连贯性。这个特征也可以用于自动文摘，一棵语篇树可以通过剪枝使该文本生成一个更简明的短文 [31，28，29]。火星探测的全文可以形成下面的文摘：

Mars experiences frigid weather conditions. Surface temperatures typically average about 60 degrees Celsius (76 degrees Fahrenheit) at the equator and can dip to 123 degrees C near the poles. Only the midday sun at tropical latitudes is warm enough to thaw ice on occasion, but any liquid water formed in this way would evaporate almost instantly.

Most Martian weather involves blowing dust or carbon dioxide. Yet even on the summer pole, temperatures never warm enough to melt frozen water.

即使这些基于语篇理论（例如 RST）的方法能保持文摘的连贯性，但是它们要移植到其他语言还是比较困难的 [31] [32]⊕。

⊕ Marcu 和 Echihabi [30] 所建议的通过语篇标记学习语篇分析器的做法已由 Sporleder 和 Lascarides [33] 证明条件不成熟。

英语[31]、日语[28]和德语[29]语篇分析器已被用于自动文摘系统中了，但其他很多语言并没有开发出相应的语篇分析器。

403 甚至翻译语篇标记也是十分困难的事，因为它们含有很多不同的语义。例如，英语的标记 since，可以表原因，也可以是纯粹的时间含义。要正确地将它翻译成德语，我们必须在 weil（原因）或者 seit（时间）之间做出选择。

尽管如此，研究者们必须考虑在不同的源语言间连贯性是如何保持的。这个领域很可能会成为一个可探索的新领域。

2. 文本的图表示法

TextRank 也是利用图表示法的自动文摘方法[19]。TextRank 和 PageRank 相似，但它依据文本关系而不是依据文档链接来生成图。图中的节点表示文本的句子，节点之间的边是两个节点相似度权重。和 PageRank 类似，高度连接的节点将被其他句子"推举"出来，从而获得更靠前的排序。

形式上，文本定义成一个有向图 $G=(V,E)$，V 是节点集，边集 $E\subset V\times V$ 表示句子之间的联系。则 PageRank 分值将可基于节点入度 $in(V_i)$ 和出度 $out(V_i)$ 进行计算。具体计算公式如下，其中 d 是一个抑制因子，表示跳到一个新页面的概率[㊀]。

$$S(V_i)=(1-d)+d*\sum_{j\in in(V_i)}\frac{1}{|out(V_j)|}S(V_j) \tag{12.1}$$

对于 TextRank，有向图变成无向图，从而 $in(V_i)=out(V_i)$。边的权重 w_{ik} 计算句子相似度得到。相似度计算方法由 Mihalcea 和 Tarau [19] 给出，计算两个句子中间现词的个数并通过句子长度归一化。

$$Similarity(S_i,S_j)=\frac{|\{w_k \mid w_k\in S_i \wedge w_k\in S_j\}|}{\log(|S_i|)+\log(|S_j|)} \tag{12.2}$$

加权的 PagePank 得分计算公式如下：

$$WS(V_i)=(1-d)+d*\sum_{V_j\in in(V_i)}\frac{1}{\sum_{V_k\in out(V_j)}w_{jk}}WS(V_j) \tag{12.3}$$

表 12-1 是一个新闻报纸文章的样例。根据公式（12.2）可以计算出所有句子间的相似度得分，并建立出图 12-2 所示的样本图，图中包含了得分及权重。

图 12-2 中的每个句子都可以计算 PagePank 得分，得分高的句子有很多高权重的边指向它。

TextRank 用 2002 年文档理解会议（Document Understanding Conference，DUC）的数据进行评测，结果表明该系统与该评测中最好的系统相当。考虑到此方法是无监督的并且不需要任何语言相关的工具（除了相似度策略），它也可以用于处理其他语言。

434

表 12-1　用作 TextRank 输入的一个新闻文章样本，输出图为图 12-2

1. BC-Hurricane Gilbert, 0348
2. BC-Hurricane Gilbert, 0-11 [illegible]
5. Hurricane Gilbert heads toward Dominican Coast
6. By Ruddy Gonzales
7. Associated Press Writer
8. Santo Domingo, Dominican Republic (AP)
9. Hurricane Gilbert Swept toward the Dominican Republic Sunday, and the Civil Defense started its heavily populated south coast to prepare for high winds, heavy rains, and high seas

㊀ 参数 d 通常设置为 0.85。

（续）

10：The Storm was approaching from the southeast with sustained winds of 75 mph gusting to 92mph.
11："There is no need for alarm," Civil Defense Director Eugenio Cabral said in a television alert shortly after midnight Saturday.
12：Cabral said residents of the province of Barahona should closely follow Gilbert's movement.
13：An estimated 100，000 people live in the province，including 70，000 in the city of Barahona，about 125 miles west of Santo Domingo.
14：Tropical storm Gilbert formed in the eastern Caribbean and strengthened into a hurricane Saturday night.
15：The National Hurricane Center in Miami reported its position at 2 a.m. Sunday at latitude 16.1 north，longitude 67.5 west，about 140 miles south of Ponce，Puerto Rico，and 200 miles southeast of Santo Domingo.
16：The National Weather Service in San Juan，Puerto Rico，said Gilbert was moving westward at 15 mph with a "broad area of cloudiness and heavy weather" rotating around the center of the storm.
17：The weather service issued a flash flood watch for Puerto Rico and the Virgin Islands until at least 6 p.m. Sunday.
18：Strong winds associated with Gilbert brought coastal flooding，strong southeast winds，and waves up to 12 feet to Puerto Rico's south coast.
19：There were no reports on casualties.
20：San Juan，on the north coast，had heavy rains and gusts Saturday，but they subsided during the night.
21：On Saturday，Hurricane Florence was downgraded to a tropical storm，and its remnants pushed inland from the U.S. Gulf Coast.
22：Residents returned home，happy to find little damage from 80 mph winds and sheets of rain.
23：Florence，the sixth named storm of the 1988 Atlantic storm season，was the second hurricane.
24：The first，Debby，reached minimal hurricane strength briefly before hitting the Mexican coast last month.

405

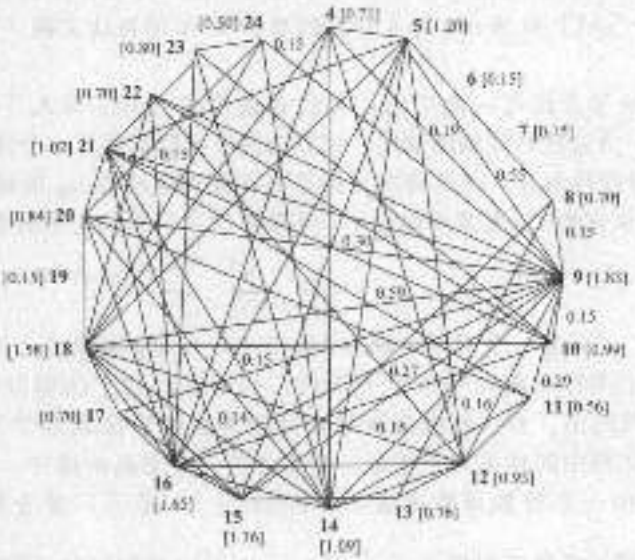


图12-2 通过文本生成的一个样本图（来源：Mihalcea 和 Tarau [18]）

另一个名为 LexPageRank 的类似方法，由 Erkan 和 Radev[5] 提出。LexPageRank 也利用了 PageRank，但是其相似度分数通过余弦相似度计算而来，并引入了权重阈值。他们将 LexPageRank 用于 MEAD 自动文摘系统，更详细的描述见 12.2.4 节。

12.2.3 学习如何做摘要

Kupiec、Pedersen 和 Chen 的工作 [23] 开始引入训练分类器以决定哪些句子应该被包含在摘要中的思路。过去的十多年中，许多方法都把这个问题看作分类问题，它们基本都有表 12-1 所列的如下部件 [34，35]：

1）一个包含文档和其摘要的对齐语料库。一些方法（例如词重叠）必须用于将摘要的句子和来自原始文本的句子匹配起来以进行摘要。

2）一个用于为每个句子生成特征向量的特征抽取器。特征可以是句子的长度、文本或段落中的位置、标题或篇首句子中词的重叠或者是句子中的词在文章中的词频。

3）一个用于对句子进行分类的机器学习算法。分类器可以是一个二元分类器、一个
[406] 多元分类器或者是一个回归模型，其中每个句子会得到一个总分数。

最近几年，出现了几种句子排序的方法。它们可以根据下面三种策略进行分类：

打分（score）　训练集中的每个句子都会获得一个分数。这个分数可以通过计算文档中句子与模型摘要中句子中词重叠得到。有了句子-分数组合，我们就可以学习一个回归模型。使用支持向量回归（Support Vector Regression，SVR）就可以用来为每个句子学习一个分数［38，37］。

偏序（partial order）　每对句子都进行排序以便获得句子偏序。句子偏序可用于训练排序算法。例如，Svore、Vanderwende 和 Burges［38］就使用 RankNet［39］，以成对交叉熵（Pairwise Cross-Entropy）为损失函数。这类似于 Amini 等人使用指数损失函数进行 XML 自动文摘的工作［40］。

等级（rank）　另一个学习摘要句子的方法是学习如何将句子排序成有序表。与成对排序不同，句子可排序为全序表（或者至少分成几“桶”并在桶中全排序）。这种方式的代表工作有 ListNet［41］和 Wang 等人开发的基于 Web 的自动文摘（Web-based summarization）［42］。

作为 3 种机器学习方法之一的举例，我们在这里对 Amini 等人［40］的基于偏序的（partial order-based）方法进行详细的描述。该方法的学习框架用了一个得分函数 $h:R^n \to R$ 来反映句子特征的最好线性组合。分类器的目标是将排序损失函数 L_R 的错误率最小化。对于每个文档 $d \in D$，损失函数 L_R 是其中得分小于不相关句子的相关句子的平均数。

$$L_R(h,D) = \frac{1}{|D|}\sum_{d\in D}\frac{1}{|S_d^{pos}|\,|S_d^{neg}|}\sum_{s\in S_d^{pos}}\sum_{s'\in S_d^{neg}}[[h(s)\geqslant h(s')]] \tag{12.4}$$

其中 $[[h(s)\geqslant h(s')]]$ 是一个谓词，如果 $h(s)\geqslant h(s')$ 则其值为 1，否则为 0。$L_R(h,D)$ 函数不断在所有的正例和反例句子组合中迭代，如果正例句子的得分小于反例句子的得分，则增加损失函数的值。有了此损失函数，排序算法的目标就是学习一个得分函数 h，使得该函数为同一文档中的相关句子指派一个比无关句子更高的得分。

损失函数应该用一个指数函数来表示，因为 $[[\,]]$ 是不可微分的（differentiated）。不同句子 s 和 s' 的得分差异可根据公式 $\sum_{i=1}^{n}\beta(s_i - s'_i)$ 以该句子的不同特征表示的差异进行计算：

$$L_{exp}(D,B) = \frac{1}{|D|}\sum_{d\in D}\frac{1}{|S_d^{pos}|\,|S_d^{neg}|}\sum_{(s,s')\in S_d^{pos}\times S_d^{neg}} e^{\sum_{i=1}^{n}\beta_i(s_i-s'_i)} \tag{12.5}$$

如果考虑到学习算法的计算复杂度，使用指数损失函数就是有好处的。它能很容易改写成一个计算指数损失函数的线性时间复杂度的公式：

$$L_{exp}(D,B) = \frac{1}{|D|}\sum_{d\in D}\frac{1}{|S_d^{pos}|\,|S_d^{neg}|}\sum_{s\in S_d^{pos}} e^{\sum_{i=1}^{n}\beta_i s_i}\sum_{s'\in S_d^{neg}} e^{\sum_{i=1}^{n}\beta_i s'_i} \tag{12.6}$$
[407]

算法 12-1 基于排序的可训练摘录式自动文摘算法 LinearRank 的伪代码

输入： $\bigcup_{d\in D} S_d^{pos}\times S_d^{neg}$，其中 D 是文档的集合，S^{pos} 是正例（摘要）句子的集合，S^{neg} 则是反例（非摘要）句子的集合

输出： 每个句子向量 s 相被用一化以便使 $\sum_{i=1}^{n}=1$；特征权重 $F=\{\beta_1\ldots\beta_n\}$ 被设置为任意值；t=0.

```
repeat
  for i = 1 to n do
    β_i^(t+1) = β_i^(t) + Σ^t
  end for
  t = t+1
until L_exp(D,F) 收敛
return B^F
```

使用每个新文档 d 的第 n 句为 d 创建一个新的摘要。排序的依据是以 B^F 为权重的句子特征的线性组合。

Amini 等人选择了一个称为 LinearRank（算法 12-1）的线性排序函数 $h(s, B)$，其中 $B=(\beta_1,\cdots,\beta_n)$ 是特征权重向量代表特征列表。算法通过更新规则 $B^{(t+1)}=B^{(t)}+\sum^{t}$ 不断迭代调整特征向量权重以优化公式 12.6 所描述的损失函数。更新函数可更精确地描述如下（更详细的信息请见 Amini 等人的工作 [40]）：

$$\beta_i^{(t+1)}=\beta_i^{(t)}+\frac{1}{2}\log\frac{\sum_{d\in D}\frac{1}{|s_d^{neg}||s_d^{pos}|}\sum_{s'\in S_d^{neg}}e^{h(s',B^{(t)})}\sum_{s\in S_d^{pos}}e^{h(s,B^{(t)})}(1-s'_i+s_i)}{\sum_{d\in D}\frac{1}{|s_d^{neg}||s_d^{pos}|}\sum_{s'\in S_d^{neg}}e^{h(s',B^{(t)})}\sum_{s\in S_d^{pos}}e^{-h(s,B^{(t)})}(1+s'_i-s_i)} \tag{12.7}$$

用三种方法中的一种生成训练集以后，每个句子都需要生成对应的特征，特征工程是很重要的，因为决定了分类器能学习的程度。

假设我们要实现一个基于查询的多文档自动文摘，如参加 DUC 或者 TAC 的自动文摘评测，我们可以利用整体主题、查询、文档甚至聚类中其他文档的频率信息。DUC/TAC 任务至少包括了一组25～50 篇依据主题分组的文档（例如，steps toward introduction of the Euro）和一个查询（例如，describe steps taken and worldwide reaction prior to introduction of the Euro on January 1, 1990）。有了这些信息，我们就可以使用下面这些在以往系统中用过的特征（例如，Schilder 与 Kondadadi [37]）：

主题标题频率（topic title frequency）：句子 s 中出现在话题标题 T 中的词 t_i 的个数与句子 s 的总词数 $t_{1,\ldots,|s|}$ 之比：

$$\frac{\sum_{i=1}^{|s|}f_T(t_i)}{|s|},\text{其中 } f_T=\begin{cases}1, & t_i\in T\\ 0, & \text{否则}\end{cases}$$

408

主题描述频率（topic description frequency）：句子 s 中出现在主题描述 D 中的词 t_i 的个数与句子 s 的总词数 $t_{1,\ldots,|s|}$ 之比：

$$\frac{\sum_{i=1}^{|s|}f_D(t_i)}{|s|},\text{其中 } f_D=\begin{cases}1, & t_i\in D\\ 0, & \text{否则}\end{cases}$$

实词频率（content word frequency）：句子 s 中所有实词 $t_{1,\ldots,|s|}$ 的平均实词概率 $p_c(t_i)$。实词概率定义为 $p_c(t_i)=\frac{n}{N}$，n 是词 t_i 出现在聚类中的次数，N 是聚类中词的总数：

$$\frac{\sum_{i=1}^{|s|}p_c(t_i)}{|s|}$$

文档频率（document frequency）：句子 s 中所有实词 $t_1, \ldots, t_{|s|}$ 的平均文档概率 $p_d(t_i)$。文档概率定义为 $p_d(t_i) = \frac{d}{D}$，d 是词 t_i 出现在给定聚类中的文档的个数，$p_c(t_i)$ 是聚类中文档的总数：

$$\frac{\sum_{i=1}^{|s|} p_d(t_i)}{|s|}$$

还有其他很多特征，包括标题频率（句子 s 中所有实际词的平均标题概率）、句子长度、句子位置、词的 TF-IDF 值、n 元组频率以及句子中的命名实体频率等都被证明是有效的。

如果涉及的所有语言都存在对齐语料，或者可以很容易获得对齐语料，这些机器学习的方法在多语自动文摘中也能很好地工作。但现实并非如此，围绕数据问题的工作也有不少，目前已提出了很多方法用来解决不同语言所写的文本之间的鸿沟。Li 和 Zha [29] 提出了一个算法，对多语言文档对的（子）话题进行对齐并通过提取句子来实现摘要。他们使用加权的二分图来实现这样的对齐，这些二分图代表了两个文档中的句子。然后句子通过机器翻译系统翻译成另一种语言，并根据计算翻译后句子和原始语言句子的相似度分数生成一个权重矩阵。注意，机器翻译所得的句子并不需要是最佳翻译，因为这里的目的是为了获取两个句子的相似度。

根据该加权图，我们可以找出高相关的句子，这些句子给出了两个文档共享的主要主题。此外，双聚类算法用来进一步找出每一个文档已聚类句子的子主题。

12.2.4　多语自动摘要

1. 挑战

我们回顾那些最重要的自动文摘方法是如何处理多语摘要的，现今的大多数自动摘要方法还是要依赖语言相关的资源和工具（例如修辞分析器、提示短语词典），一些方法从与源语言提取的表示中生成文摘，因而可用于独立于语言的自动文摘。

[435] 下面的列表总结了多语自动文摘系统需考虑的特征摘要。这些都是我们在做多语时必须面对的挑战。

词元切分（tokenization）　由于不同的语言有不同的词边界表示，所以词元切分是我们搭建多语自动文摘第一个应该克服的问题。比如英语通过空格和标点符号作为一个词元的分界，但其他语言比如中文，就需要一个更复杂的分词器来从一连串输入中提取词元，因为它们之间没有空格。英语中一个词元就是一个词，但是在不同的语言中并不一定相同。其他语言（例如，阿拉伯语）需要处理十分丰富的语法形态，因而要求能处理形元层的精细处理。

指代表达（anaphoric expression）　指代关系的识别（例如，代词、语篇标记、限定性名词短语）能帮助文摘结合更加紧密。单语自动文摘中已经存在一些技术，但是多语文摘面临着许多挑战，比如，不同语言中名字可能会被写成不同形式，语篇标记也会有不同的语义。

Mitkov[4] 提出了一种知识贫乏的指代消解方法。除了使用性、数一致性，还使用了许多简单的标识（例如，限定性、给予性、动词类）并为可能的先行词汇总出一个分值。

篇章结构（discourse structure）　文档结构的识别有助于提高摘要的连贯性。但是不同的语言，文本表达的结构也不相同。

机器翻译　机器翻译技术的现有水平还无法达到一个高质量实用的水平。当设计一个

多语自动文摘系统，设计者必须回答机器翻译应该在系统中何时使用的问题。如果一开始就翻译，源语言的组件能被重用（比如分词）。如果在识别完摘要句子后再进行翻译，相应的语言相关系统就必须用于对文档进行预处理。

2. 系统

现在有三个重要的自动文摘系统是具有多语能力的，它们分别是：MEAD、Summa 和 NewsBlaster。

MEAD[⊖]平台是多语自动文摘并带有开源的平台，提供了几种不同的文摘算法，有基于位置的，基于质心的，基于最长公共子序列的以及基于关键词的。它是用 Perl 编写的可以公开获得的平台，该系统的框架既可适用于之前讨论过的基于表层特征的方法也可用于训练侦测可摘要句子的分类器。它提供了诸如决策树，支持向量机以及最大熵等的机器学习算法，允许用户训练自己的自动文摘方法。 410

MEAD 的核心框架是基于质心的（centroid-based）文摘方法。质心是对某文档聚类起重要作用的词的集合。这些文档簇的相关文档和文摘句子均根据它们包含的质心进行抽取。

这个聚类算法叫做 CIDR [43]。CIDR 产生的文档簇共享相同的词。从一个文档开始，算法比较其他簇与它的相似度。文档用词向量表示，各词的值则为该词的 TF-IDF 值（文档频率和逆文档频率）。

每个簇都有一个质心，可以描述为一个仅包含最重要词（具最大 TF-IDF 值）的伪文档。质心的词向量是簇内所有文档词向量的平均值。

这个算法从一个文档开始，该文档被放进第一簇（仅包含一个文档），然后新的文档先计算它和每个簇代表词向量的余弦相似度，找到最高相似度的那个簇，如果该相似度不低于事先设好的阈值，那么就把该文档划分到这个簇内。

余弦相似度是这两个（词）向量的余弦值：

$$\text{sim}(A,B) = \cos\theta = \frac{A \cdot B}{|A||B|} \tag{12.8}$$

如果一个文档不能划分到任何簇，那么就以这个文档成立一个新的簇。

另一个基于图的算法也和 MEAD 一起出现：LexPageRank [5] 计算词汇连通矩阵（根据一个特定的阈值）中句子的 PageRank 分值。LexPageRank 和 TextRank 很类似，都利用 PageRank 来计算句子的分值。它与 PageRank 的不同在于图权重的生成方式不同，LexPageRank 用余弦相似度来计算句子间相似度而生成权重。它还允许定义余弦相似度阈值：如果两个句子的余弦相似度大于某特定阈值（例如，0.1），它们之间才会产生一条边。

另一个自动文摘系统是 SUMMA——该系统可以作为 GATE 的插件也可以独立运行。该系统提供基于位置的和基于质心的得分函数，还允许用户添加自己的得分函数。系统还包含了多种相似度的计算方法，例如余弦相似度和 n 元组相似度。

最后是由哥伦比亚大学开发的 NewsBlaster 自动文摘系统（参见图 12-3），可以处理多语扩展 [9][⊖]并且能让用户从互联网上的多个网站浏览多语新闻。这个系统对不是英语的文本进行机器翻译，然后在机器翻译的英语文本上利用了经充分测试的（well-tested）方法进行文本聚类 [44]。文摘则使用哥伦比亚文摘系统 [45] 生成，该系统聚类是由非英语文本翻译而来的文本。网上在线提供该系统的英语版本，但是似乎没有多语版本。 411

⊖ http://www.summarization.com/mead/。

⊖ http://newsblaster.cs.columbia.edu 网站可对（英语）新闻网站的新闻（和图片）进行自动文摘。

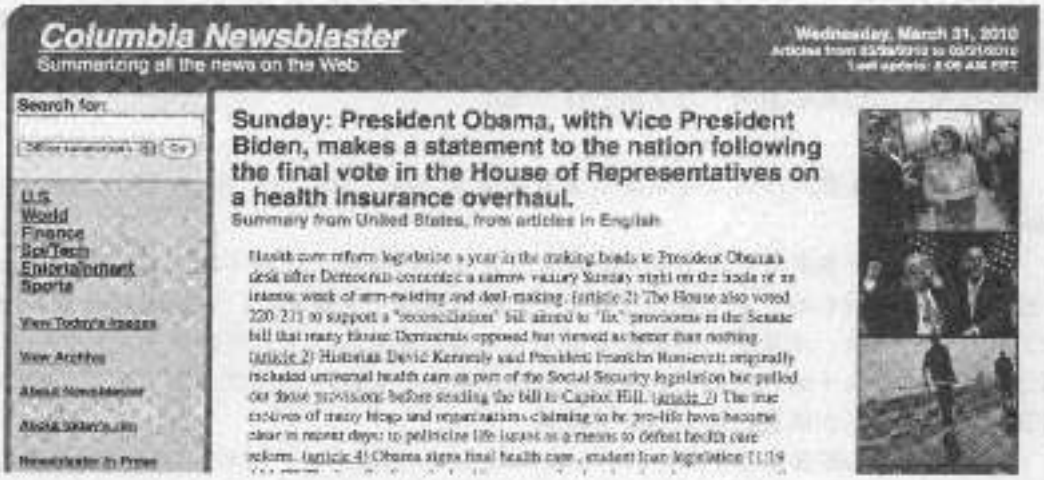


图 12-3 NewsBlaster 所生成的页面举例（摘自 Columbia 大学）

12.3 评测

确定一个自动文摘系统生成的文摘的质量也是自动文摘研究领域的一个主要挑战。文摘评价的方法可以分为两类，**外部评价**（extrinsic evaluation）和**内部评价**（intrinsic evaluation）。外部评价方法的基本思想是借助于自动文摘系统完成一些别的信息处理任务，通过该任务完成的性能指标来对文摘系统作间接评价。内部评价通过直接分析摘要的质量来评价文摘系统，并且可用于摘要开发周期的各个阶段。一般会通过比较待审摘要和参考摘要的覆盖率来评价。**参考摘要**（reference summary）一般由人工提供，作为评测比较的标准摘要。如果是在进行自动文摘系统评测，那**待审摘要**（peer summary）就是系统产生的摘要；如果我们是在分析参考摘要的质量，那它也可以是人工编写的摘要。

按照对文摘系统的评价是由人工完成还是机器完成，还可将评价方法分为人工评价和自动评价方式。自然，人工评价在很大程度上是可信的，因为人可以推理、复述并使用世界知识将具有类似意思但形式不同的文本单元关联起来。如果评价容易实施、管理且无须反复执行，则人工评价就是上佳之选。但是如果人力资源有限，则应该采用自动评价方法。要创造一种对一般自动文摘任务都能适用的自动评测方法是十分复杂的。这一节我们将详细讨论人工评价和自动评价方法。

412

12.3.1 人工评价

不管是对单文档还是多文档，在自动文摘中，摘要或者信息压缩都有很高的自由度。信息选取很大程度上取决于任务的定义、问题所属的话题、领域以及先验知识。就算是人工进行文摘任务，不同的人也会对原文中哪些句子需进入摘要会有不同意见。人们在进行人工评价练习时发现并分析了这一奇怪的现象。

有三种较为流行的人工评测方法，它们分别是 Lin 和 Hovy（[46] 和 [47]）提出的文摘评测环境（Summary Evaluation Environment，SEE），Van Halteren 和 Teufel [48] 的 Factoid 方法以及 Nenkova 和 Passonneau [49] 的金字塔（Pyramid）方法。

1. 文摘评测环境

文摘评测环境（Summary Evaluation Environment，SEE）提供了用户友好界面，评

审员可通过比较待审摘要和参考摘要进行质量评测。将参考文摘和待审文摘都分割成若干反映基本信息（例如句子、子句等）的单元，评审员可以依据各待审摘要单元与对应参考摘要单元的比较为其设定完全或部分内容匹配分值。各单元的语法合法性也可分别打分。此工作独特的特点是它既可以对摘要单元整体识别也允许部分匹配。

2. Factoid 方法

Factoid 工作的目标就是比较同一文本不同摘要的信息内容，并确定要在人工编写的摘要中达到稳定一致需要的摘要的最小数目。Van Halteren 和 Teufel 研究了一个基于单一文本创建的 50 个摘要，每个句子都有一系列称为 Factoid 的原子语义单元来表示。这里，语义原子性意味着每个 Factoid 所关联的语义信息可以小到一个词语也可以大到整个句子，系统将收集并分析所有摘要的 Factoid 集。那些在多个摘要间表示同样意义或携带同样信息量的 Factoid 将会由人工标识为语义相似。随着手工创建摘要的增长并达到一个特定水平，Factoid 集合也将稳定（新摘要以及其对应的 Factoid 的加入基本不会影响该集合）。理想情况是，在我们开始通过内容对比评价系统生成摘要之前，人工撰写的摘要标准集已经是一个稳定的集合。Factoid 方法显示，要在参考摘要间达到稳定一致，至少需要 15 个摘要。实际上，由于信息处理任务的资源密集本质，人工撰写摘要的量往往远小于这个数。 413

3. 金字塔方法

金字塔（Pyramid）方法是 Factoid 方法的一个更大规模的扩展。Nenkova 和 Passonneau 表明，只需 6 个摘要就可以达到参考摘要的稳定一致。作者在经验上已证明这降低了对参考摘要数目的需求，并且能达到可靠鉴别摘要计分的主要目标。文摘内容单元（Summarization Content Unit，SCU）原来被定义为一个不大于子句的单元，后面被重新定义为大于一个词但小于一个句子的单元，因为一个子句还可能会包含多个语义单元。

查找相似 SCU 的过程开始于对相似句子的查找，然后开始更精细地检查更紧密相关的子部分。在所有 SCU 都被找到并比较后，它们就可以被划分为如图 12-4 的金字塔结构。一个 SCU 被越多的参考文摘包含就越重要。将所有 SCU 按照重要程度排序，同等重要的 SCU 排列在同一行，由上向下重要程度逐行递减。金字塔的层数和 SCU 出现在摘要中的数量是有关系的，在所有摘要中都出现过的 SCU 肯定在金字塔的顶层，因为它们是最少的。在较底层出现的 SCU 可用来表现人工摘要撰写者在理解、兴趣以及摘要主题知识上的差异。在底层大量的 SCU 也说明了摘要的困难性。下面用一个来自 Nenkova 和 Passonneau [49] 的文摘例子来展示摘要的多样性，其中画线的部分就是共享的 SCU（这四个句子来自四个不同的文摘）：

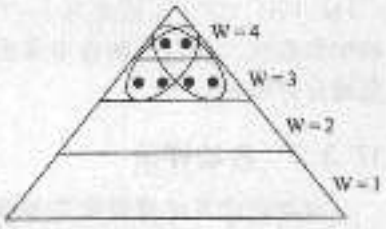


图 12-4 4 层金字塔（复制自 Nenkova 和 Passonneau [49]）

A. In 1988 two Libyans indicted in 1991 for the Lockerbie bombing were still in Libya.

B. Two Libyans were indicted in 1991 for blowing up a Pan Am jumbo jet over Lockerbie, Scotland, in 1988.

C. Two Libyans, accused by the United States and Britain of bombing a New York-bound Pan Am jet over Lockerbie, Scotland, in 1988, killing 270 people, for 10 years were harbored by Libya who claimed the suspects could not get a fair trial in America or Britain.

D. Two Libyan suspects were indicted in 1991.

我们可以获得两个 SCU：

SCU1（权重=4）：two Libyans were officially accused by the Lockerbie bombing

A. [two Libyans] [indicted]

414 B. [Two Libyans were indicted]

C. [Two Libyans,] [accused]

D. [Two Libyan suspects were indicted]

SCU2（权重=3）：the indictment of the two Lockerbie suspects were in 1991

A. [in 1991]

B. [in 1991]

C. [in 1991]

待审摘要的得分是基于精确率的，待审摘要 SCU 的权重总和与具相同数量 SCU 的最佳摘要的权重总和之间的比例将作为该待审摘要的得分。假设我们有一个已经由参考摘要中的 SCU 构建好的金字塔，在待审摘要中包含了 10 个 SCU，而仅仅只有一个 SCU 出现在金字塔里，并且这个 SCU 在金字塔中的权重是 1（在所有的参考摘要里仅出现过一次），那么这个待审摘要的得分就是 1/10=0.1。

虽然金字塔方法和其他人工评价一样需要很大的花费，但它还是被学术界广泛接受为首选手工评价方法。在 DUC 和 DUC 之后的 TAC 任务中都用到了这个评测方法。

金字塔在大规模数据上实验的另一个好处是可以获得大规模的人工撰写的摘要和对应的语义单元。在自动文摘系统的设计和调整中，以语义单元为标准数据将有助于句子级摘录的研究。

4. 响应度（responsiveness）

除了通过人工进行内容覆盖率的评测，TAC 还为每个待审摘要给出了一个范围为 1～5（从 TAC 2009 开始改为 1～10）的响应度分值。这个分数并不反映待审摘要和参考摘要的相似程度，而仅反映待审摘要的内容覆盖率和语言学质量（在以前的 DUC 中这两个质量被分开评估）。

12.3.2　自动评价

自动评价系统就是把参考摘要和待审摘要作为输入，通过执行文本比较以产生一个评价结果的系统。为了测试和证实自动评价的效果，研究者必须证明该自动评价的结果和人工评价相高度、正向、一致相关 [50]。

1. ROUGE

415 ROUGE（Recall-Oriented Understudy for Gisting Evaluation）[50，51] 是第一个自动文摘评测系统。它的基本思想是将待审摘要和参考摘要的 n 元组共现统计量作为评判待审摘要的依据。ROUGE 受到了机器翻译自动评价方法 BLEU [52] 的启发，但和 BLEU 面向精确率不同，它是一个面向召回率的方法。

ROUGE 不是用一个指标来进行评测，而是有一系列的标准对机器生成的摘要进行打分：

- ROUGE-N：计算待审摘要和与其相应的所有参考摘要的 n 元组召回率。下列公式显示如何计算待审摘要匹配的 n 元组占参考摘要中所有 n 元组的比例：

$$\text{ROUGE-N}=\frac{\sum_{S\in\{\text{Reference Summaries}\}}\sum_{\text{gram}_n\in S}\text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{S\in\{\text{Reference Summaries}\}}\sum_{\text{gram}_n\in S}\text{Count}(\text{gram}_n)} \tag{12.9}$$

- ROUGE-L：匹配两个文本单元之间的最长公共序列（Longest Common Subse-

quence，LCS)。注意是最长的序列匹配，不是连续匹配。不需要像 n 元组匹配一样预定义一个长度限制。这个指标也反映了比较的单元之间可能的句法结构差异。为了估计长度为 m 的摘要 X 和长度为 n 的摘要 Y 之间的相似度，基于 LCS 的 F 值可定义如下：

$$R_{lcs} = \frac{\mathrm{LCS}(X,Y)}{m} \tag{12.10}$$

$$P_{lcs} = \frac{\mathrm{LCS}(X,Y)}{n} \tag{12.11}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs}+\beta^2 P_{lcs}} \tag{12.12}$$

$\mathrm{LCS}(X, Y)$ 是 X 和 Y 的最长公共子序列，$\beta = P_{lcs}/R_{lcs}$ 当 $\partial F_{lcs}/\partial R_{lcs} = \partial F_{lcs}/\partial P_{lcs}$。

- ROUGE W：计算加权的 LCS。在 ROUGE-L 中我们考虑的公共子序列并不要求是连续的，对那些不连续的匹配并没有惩罚。ROUGE-W 对词序列的连续匹配和非连续匹配区别对待，并对非连续匹配给予一个间隔惩罚（gap penalty）。该惩罚用一个权重函数来表示，对于连续匹配将返回比非连续匹配更高的奖励值。
- ROUGE-S：计算跳二元组（skip-bigram）同现统计量。一个跳二元组是句中两个有序的词，中间允许任意长度的间隔。当间隔等于 0，则等价于 ROUGE-N 中 $n=2$ 的情形。基于跳二元组的 F 值计算公式如下：

$$R_{skip2} = \frac{\mathrm{SKIP2}(X,Y)}{C(m,2)} \tag{12.13}$$ 416

$$P_{skip2} = \frac{\mathrm{SKIP2}(X,Y)}{C(n,2)} \tag{12.14}$$

$$F_{skip2} = \frac{(1+\beta^2)R_{skip2}P_{skip2}}{R_{skip2}+\beta^2 P_{skip2}} \tag{12.15}$$

- ROUGE-SU：它是对 ROUGE-S 的补充，增加单个词（unigram）的匹配以处理两个句子没有任何跳二元组匹配的情况，否则 ROUGE-S 将会对那些可能有相同内容但是词序列不同的句子进行惩罚。

ROUGE 产生的评测结果和诸如响应度这样的人工评测有很高的相关性。相关性可以通过斯皮尔曼等级（Spearman ranking）和皮尔逊相关系数（Pearson correlation coefficient）来表达。斯皮尔曼相关系数（Spearman correlation coefficient）用于表明两个等级次序的相关性：

有 n 个等级对 (x_i, y_i) 时

$$\rho = 1 - \frac{6\sum (x_i - y_i)^2}{n(n^2-1)} \tag{12.16}$$

皮尔逊相关系数则直接在原始分值对 (X_i, Y_i) 而不是等级上进行计算，如下：

$$r = \frac{\sum_{i=1}^{n}(X_i-\overline{X})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\overline{Y})^2}} \tag{12.17}$$

ROUGE 自动评测方法的另一个优点是它不依赖其他语言的处理工具，例如各种句法分析器等，但它还是提供了选项以使用户在需要时可以激活词干化和词性标注。

2. 基元

基元（Basic Elements，BE）[53] 是基于最小语义单元概念的自动评测方法。一个基元是从句子里提取出来的一个类似 subject-object 关系和 modified-object 关系的语义单元。我们可以利用很多不同的句法和依存分析器得到 BE：Charniak 分析器 [54]、Collins 分析器 [55]、Minipar [56] 和 Microsoft Logical Forms [57]。BE 分割模型接受一个句法分析树并利用一些启发式函数从中提取出 BE。Collins 和 Charniak 的句法分析树是短语结构树，其中并不包含中心词和修饰词间的语义关系。而 Minipar 则是依存树，它会自动产生<*head*；*modifier*；*relation*>这样的三元组。BE 工具包默认是用 Minipar 的依存树来
417 生成 BE。系统通过测量待审摘要的 BE 和参考摘要的 BE 之间的重叠来进行打分。

为了证明 BE 在评测自动文摘中是有效的，它使用了 2003 年 DUC 的结果进行测试，比较系统产生的待审摘要和作为标准的参考摘要。相关系数通过比较 DUC 系统（待审摘要和基线）和 BE 产生的等级和平均覆盖率得分进行计算。在验证测试集上同时使用了斯皮尔曼相关系数和皮尔逊相关系数。虽然在运行 BE 包时有多个选项，但作者表明，在评测多文档结果时，运行 BE-F（其中使用 Minipar 来抽取 BE，不区分中心词和修饰词间的关系，使用所有词的原形）可以获得最高的斯皮尔曼和皮尔逊相关系数。

在进行自动文摘系统开发和调试时，通常会同时运行 ROUGE 和 BE 方法以对系统生成的结果进行整体分析。然而，由于 BE 需要依赖分析器，所以如果某语言没有依存分析器，研究者们可能无法在多语言场景下使用 BE 包。

3. 相关工作

ROUGE 和 BE 比较流行是因为它们的简单性以及和人工判断的高一致性。但是待审摘要和参考摘要的文本单元之间的比较还仅限于词汇标识的匹配。利用复述（paraphrase）和同义词测量语义相近性的研究也已经出现。ParaEval [58] 方法整体是 3 层比较策略，其中利用了复述匹配。最顶层通过贪心算法在参考摘要（通常是人工撰写的）和待审摘要（系统生成的）之间、在短语级别上寻找多词复述的重叠数，并选择具有高覆盖率的摘要。那些不匹配的部分将进入到下一层，该层使用用贪心算法寻找单个词的复述或同义词匹配。最后一层（最底层）则将前两层比较剩下的文本用 ROUGE-1 评测。这个多层设计可以保证在没有复述被发现时，在 ROUGE-1 的层面上也可以进行摘要内容匹配。和原来的 ROUGE 相比，ParaEval 在相关性上有略微的提高。我们可以应用机器翻译对齐数据来产生复述，其中假设那些经常可互换翻译的短语很可能互为复述 [59]。这个方法对机器翻译评测是很有效的 [60]，因为翻译的目标是从原始文档中产生一个没有压缩和冗余的目标语言文档。

12.3.3　自动文摘评测系统的近期发展

2004 年，Filatova 和 Hatzivassiloglou [61] 将原子事件定义为文本描述活动的主要成分，这些活动由动词和动作性名词关联起来。他们认为文本中事件的主要成分可以标记为命名实体，一个原子事件是一个三元组，包含同一个句子中动词或者动作性名词以及由它联系起来
418 的两个命名实体。原子事件用于创建基于事件的摘要，并且尚未在任何评测方法中使用。

Tratz 和 Hovy [62] 提出了一种对原始 BE 方法的改进，该改进有助于将表层文本转换为基元文本单元，该方法称为 BEwTE（BE with Transformations for Evaluation）。这个工作的基本思路是简单的词汇识别匹配，未考虑句法或语义结构不同但意义相同或相似的文本单位的等价性。为了进行自动转换，该方法提出了一系列的转换启发函数并定义了

它们的执行次序。这一方法的创建过程是人工执行的。待审摘要的 BE 得分是通过贪心匹配算法计算的，同时也使用对应各参考 BE 的总权重进行归一化。该方法在各 DUC 和 TAC 数据上的相关性检测效果比原始的 BE 方法和 ROUGE 都要好。除了依赖语言处理工具外，BEwTE 在处理多语的时候需要大量人工劳动和语言学知识以编写转换规则并规定其执行次序。

Louis 和 Nenkova [63] 提出了一种自动文摘的评测方法，该方法不需要用到人工撰写的参考摘要。该工作假设标准摘要在文本的词频率分布上有较低的散度，低散度意味着高相似度。参考摘要和待审摘要之间的 Kullbank Leibler（KL）散度和 Jensen Shannon（JS）散度被用作摘要的得分。话题信号 [64] 这一十分重要的摘要特征也证明在摘要评价中有很好的指导意义，高话题集中度表明更高的摘要内容质量。

另一个创新的工作，AutoSummENG（基于 n 元组图的自动摘要评价，Automatic Summary Evaluation based on n-gram Graphs [65]）则创建摘要图，其中 n 元组是节点，n 元组之间的关系是边。该工作中，摘要间的比较就变成了待审摘要图与参考摘要图间的比较了，关系用 n 元组周围固定长度的上下文窗口信息进行建模。边的权重则用 n 元组节点之间的距离和文本中的同现次数来表示。这个方法比其他自动方法效果要好。此外，它还不需要语言相关的分析工具，因而具有语言中立的优点。

12.3.4 多语自动文摘的自动评测方法

自动文摘是一个复杂的自然语言处理任务，它的评测是一个挑战，这也促进了领域的发展。虽然大多数自动评测方法都是基于词汇识别匹配的，但它们通过统计方法为自动文摘质量提供了一个可靠的评测方法。通过这些方法我们可以识别一个系统是好是坏，但是对于接近的系统，要识别出它们的细微差别是有困难的。考虑到这些评测方法的缺点，理解任务并进行错误分析对于摘要系统的设计者来说是十分重要的。而能用于多语自动文摘的自动评测方法就更少了，表 12-2 总结了本节所讨论的自动评测方法的语言无关情况。 419

表 12-2 自动文摘所使用的评价指标以及它们需要的语言相关处理工具

方法名	是否依赖语言处理工具	备注
ROUGE	否	
BE	是	需要短语树和依存树分析器
ParaEval	否	机器翻译对齐数据
BEwTE	是	基于依存和语言学知识
Divergence ([63])	否	
AutoSummENG	否	

12.4 如何搭建自动文摘系统

本节给出如何搭建自动文摘系统的一个蓝图。我们不指定任何特殊的编程语言和开发框架，因为一个自动文摘能用任何编程语言搭建。本节包含了用不同的工具和框架从零开始或在已有框架上搭建一个多语自动文摘系统的指南。

一个多语自动文摘系统的一般流程如图 12-5 所示。这个一般流程反映了自动文摘的 3 个普遍步骤（我们在 12.2.1 节做过介绍）。首先，文档必须经过分析。根据想要搭建的多语自动文摘系统类型，我们的输入文档可能是一种语言（即跨语际摘要）或者多种语言（即跨语言摘要）。12.5 节列出了多个多语语料库。语言的选择将影响到我们后面要选的工具（参见 12.4.2 节）。

收集好输入数据以后，就开始使用分词工具了。分词可能不只是简单的空格分词，一些特别的语言（例如中文）就不是用空格或者标点符号作为词的界限；如果某种语言有丰富的形态（比如阿拉伯语），或者能生成大量的复合表达（如德语的复合名词），也可能需要比词更精细的分类。这个分析步骤也包含其他的一些划分和组块技术，例如句子划分、组块分析和句法分析等。在分词处理中，词元、n 元组、组块等单位的频率也都需要统计。

在对文本进行词元划分后，下一步是把这些词元联系起来，可以通过共指关系（例如，Microsoft-the company）或者前后上下文（例如，Today-Microsoft-announced）加以实现。

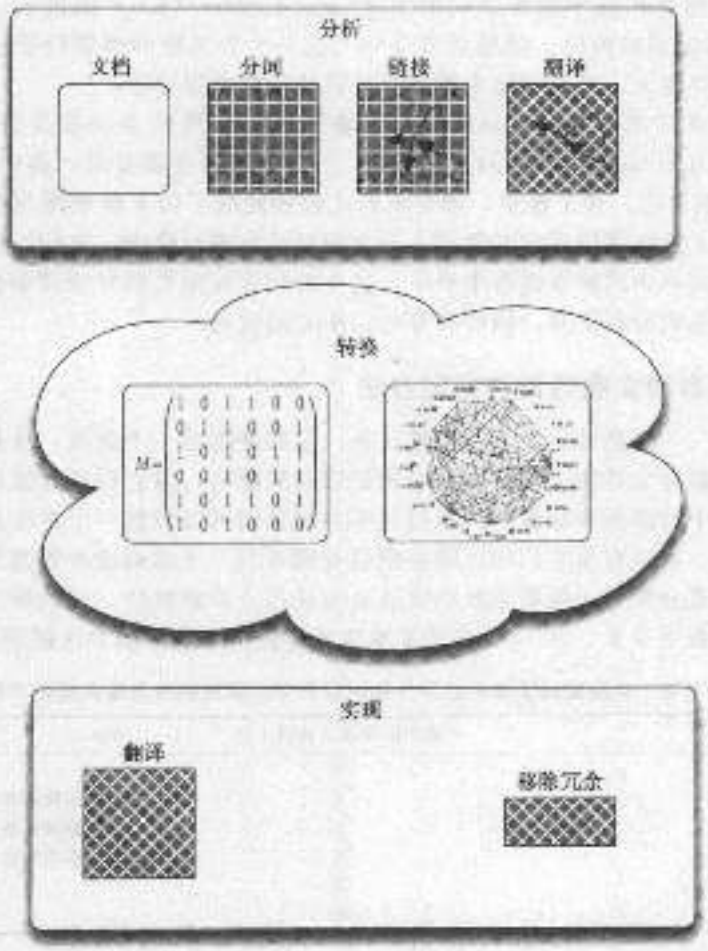


图 12-5　多语文摘系统的蓝图

分析阶段的最后步骤是对输入内容进行翻译。这可以在分词和词元联系前、后做，甚至可以推迟到转换步骤以后才做。

[420]

自动文摘系统的第二个阶段是对分析后的文本进行转换。文本分析中所做的选择决定了什么样的表达会成为转换模块的输入。12.2 节提供了文摘系统可选择的不同方法。本步骤的输出是一个根据摘要重要性降序排列的句子或语块列表。

最后的实现步骤将生成摘要。通过应用各种去冗余技术（例如 QR，余弦相似度），我们可以把要输出文本中的冗余移除。如果在输入文本中并没有运行机器翻译，那么现在就需要把生成的文本翻译成目标语言。

12.4.1 材料

开发一个多语自动文摘系统，各种形式的数据是首要条件，理想情况是，我们可以从NIST[㊀]和LDC[㊁]上获取各种摘要语料库。

然而，大多情况下，我们并没有某领域的可用数据，而创造标准数据的代价又是十分昂贵的。这种情况下，我们可以通过在可用的数据上训练和测试系统，然后将该系统迁移到新领域或者使其对新领域进行自适应。很明显，领域适应问题本身就是一个很有趣的研究方向，请参考最近的一些工作，如Daumé和Marcu [66]所描述的或ACL 2010领域自适应Workshop上所报告的[㊂]。

训练自动文摘工具的一个好的起点是NIST为DUC和TAC所提供的数据集。

系统可以用不同的框架实现。我们在这里列出一些可能的建议，但并不是表示这是一个完全的列表：

- UIMA[㊃]表示无结构信息管理架构（Unstructured Information Management Architecture），是IBM开发的一个Apache项目。它有一个组件架构和软件框架，可以实现对无结构内容的分析，例如文本、视频和音频数据。这个框架是基于Java的，但也可用于C++。
- GATE[㊄]是一个文本工程的通用架构（General Architecture for Text Engineering），于1996由谢菲尔德大学开发并发布了第一个版本。GATE是自然语言处理的一个通用框架，它包含了很多主要由Java实现并对开发文摘系统很有用的语言处理工具，而且Horacio Saggion所开发的SUMMA工具包就可以作为GATE的插件。
- NLTK[㊅]，自然语言处理工具包（the Natural Language ToolKit），提供了很多用Python写的自然语言处理工具。这个工具包的开发是为了指导如何用Python处理自然语言，其中包含了类型多样的包，包括各种不同的标注器、词干还原器、句法分析器以及语料处理工具和分类与聚类算法等。
- R[㊆]不是一个自然语言处理工具，它是一个用于统计计算和图处理的免费软件环境，使用它我们可以很容易实现12.2节中讨论过的一些技术。其中还包含了很多机器学习的包[㊇]和进行图处理的工具[㊈]，包括PageRank的实现。

421 / 422

这些通用的框架为我们编写自己的文摘系统提供了支持，你也可用从我们前面介绍过的开源的自动文摘系统开始：MEAD和SUMMA。

12.4.2 工具

搭建一个自动文摘系统需要很多工具。特别地，如果处理的是多语言，则一个机器翻

㊀ http://www.nist.gov/tac/data/index.html。
㊁ http://www.ldc.upenn.edu/。
㊂ http://sites.google.com/site/duuldp2010/home。
㊃ http://uima.apache.org/。
㊄ http://gate.ac.uk。
㊅ http://www.nltk.org/。
㊆ http://www.r-project.org/。
㊇ http://cran.r-project.org/web/views/MachineLearning.html。
㊈ http://igraph.sourceforge.net/。

译程序就是必要的。

下面是一个“材料”列表，提供了很多关于如何用一个材料替代另一个的建议。

分词工具（tokenizer）或**句子划分工具**（sentence splitter）　前面提到的工具（例如NLTK，GATE）中都包含了分词和句子划分工具。下面还有一些提供相似功能的自然语言处理工具：

- LingPipe　提供了几个不同的Java包来对人类语言进行分析，包括句子划分工具，中文分词工具以及英文分词工具。
- openNLP　合并了几个开源的自然语言处理项目，也提供句子划分和分词工具。

机器翻译程序（machine translation program）可用的机器翻译工具有好几个，可以让研究者训练自己的统计机器学习模型：

- Giza++　http://fjoch.com/GIZA++.html。
- Thot　http://sourceforge.net/projects/thot/。
- Moses　http://www.statmt.org/moses/。
- Joshua　http://www.cs.jhu.edu/ccb/joshua/index.html。

另一种方式是通过Google翻译API：

http://code.google.com/p/google-api-translate-java/。

特征选择工具（feature extractor）　为了运行机器学习实验或生成图表示，研究者必须从文档的句子（或者词）中提取特征。有一些可用工具可以实现这些处理并提供直接可用（out-of-the-box）的特征选择器（例如基于n元组的特征）。科罗拉多博尔德大学开发的与UIMA框架一起运作的工具称为ClearTK[⊖] [67]。

12.4.3　说明

前面两小节已经指出了搭建自己的自动文摘系统的必要材料和工具。在本章的最后部分，我们讨论如何利用自动文摘系统的蓝图（参见图12.5）。

[423]

首先，必须确定在何处使用机器翻译。可以先翻译也可以后翻译，后翻译有两个优点：当对大量的文档进行摘要时，文摘系统的速度会比较快，因为只有摘要的句子需要翻译。还有如果后翻译，翻译错误对文摘处理的影响会比较小，如果文摘要使用一个高层次的语言学特征，例如句法分析树，那么翻译错误对其影响将会是很大的，如果聚类或基于图的方法是基于词袋或者n元组特征的，那么翻译错误也许就不会有影响。根据你的选择，其他组件也应可获得并产生出有一定质量的输出（例如分词程序、组块分析器等）。

然后，你需要确定整体方法。12.2节我们总结了很多不同方法，特别介绍了聚类和基于图的方法，它们在跨语言或跨语际文摘上也能很好地工作。在做这部分决定时，你应该考虑到可用的语言资源以及机器翻译组件的质量。输出将会是一个摘要句子的排序列表，其中最好的句子就是最应该入选摘要的句子。

系统的生成部分有一个可选的模块。对于多文档自动文摘，这个模块是为了确保没有冗余的句子被选择，而对于多语自动文摘系统，则它必须确保实体和概念被正确地翻译。一些系统提供了去冗余（例如Carbonell和Goldstein [18]）或选择其他语言的名字（例如Mani、Yeh和Candon [14]）的解决方法。

⊖ http://code.google.com/p/cleartk/。

最后，最重要的是你必须决定用哪种评价方法。根据你使用的系统选择是内部评价还是外部评价。使用不同的参数进行评测，根据评测结果决定最好的系统参数。

12.5 评测竞赛和数据集

12.5.1 评测竞赛

DUC（Document Understanding Conference，文本理解会议） 由美国国家标准化局（National Institute of Standards and Technology，NIST）从2001年发起一直举办到2007年，会议致力于推动自动文摘研究的发展并提供了一个让研究者能参与到大规模文本测试中来的论坛。会议的任务有单文档文摘也有多文档文摘，除了2003年涉及对阿拉伯语到英语翻译的文摘外，此会议基本只涉及英语。

TAC DUC会议从2008年改名为文本分析会议（Text Analysis Conference），任务包括基于查询的多文档文摘、基于观点的自动文摘，还有更新式文摘。2011年TAC包含了多语任务，2012年的实体链接任务将真正是多语言的，涉及英语、中文和西班牙语。

MSE 多语自动文摘评测（Multilingual Summarization Evaluation，MSE）2005年和2005年都关注多文档自动摘要，语料是TDT-4语料库中的英语和阿拉伯语部分，包含了41 728篇阿拉伯语文档和23 802篇英语文档。和DUC 2003的自动摘要任务类似，自动文摘是在原始英文新闻文章的阿拉伯语翻译上进行的。首先通过哥伦比亚大学聚类算法对TDT4语料进行聚类，然后使用ISI的机器翻译系统来翻译阿拉伯语。有趣的是2005年的最好系统只用英语句子作为输入。

12.5.2 数据集

- SummBank（粤语、英语）有18 147篇双语平行文章，来自中华人民共和国香港特别行政区信息服务部（Information Services Department of the Hong Kong Special Administrative Region of the People's Republic of China）：
  http://clair.si.umich.edu/clair/CSTBank/
  http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogID=LDC2003T16
- 文档理解会议（Document Understanding Conference）（英语、阿拉伯语 _DUC 2003_）[⊖]。
- 文本分析会议（Text Analysis Conference）（英语）[⊜]。
- 多语自动文摘评测（Multilingual Summarization Evaluation）（阿拉伯语、英语）。
- 跨文档结构理论库（Crossdocument Structure Theory Bank，CSTBank）（英语）：数据用跨文档结构理论（Crossdocument Structure Theory，CST）进行标注，CST是一个与修辞结构理论相关的描述多文档语篇结构的功能理论。
  （http://clair.si.umich.edu/clair/CSTBank/）
- 纽约时报标注语料库（The New York Times Annotated Corpus）（英语）包含了从1987年1月1日到2007年6月19日纽约时报的超过180万篇文章，以及由纽约时报编辑室（New York Times Newsroom）提供的文章元数据。该语料库包含了由图书馆科学家所撰写的超过650 000篇文章摘要。虽然是单语语料库，但是其中提

[24]

⊖ http://www-nlpir.nist.gov/projects/duc/data.html。
⊜ http://www.nist.gov/tac/data/index.html。

供了对人物、组织、地点以及话题描述等的规范化索引，对跨文档的实体映射很有帮助。

http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2008T19

- 语言理解标注语料库（The Language Understanding Annotation Corpus）（阿拉伯语、英语）包含3000词的已标注英文文本（6949词）和阿拉伯语文本（2183词），标注包括：承诺信度（committed belief）、事件与实体共指关系（event and entity coreference）、对话行为（dialog act），以及时间关系（temporal relation）等。

  http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2009T10

- 话题检测与跟踪（Topic Detection and Tracking，TDT）语料库包含了多年创建的多语言数据（英语、阿拉伯语、中文普通话）。TDT2多语文本（TDT2 Multilan-
[425] guage Text）语料库包含了来自两种语言（美式英语和中文普通话）9个新闻源超过6个月（1998年1～6月）的新闻数据。

  http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDA2001T57

  TDT 3的数据除了上面的9个来源外还增加了两个英语电视源。这个语料库包含了3个月期间每日收集的数据（1998年10月到12月）。

  http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDA2001T58

  最后，TDT4包含了在2002年和2003年TDT技术评测中使用的英语、阿拉伯语和中文（广播新闻脚本和新闻数据）的完整数据集。

  http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDA2005T16

12.6 总结

这一章我们讲述了进行自动文摘的主要方法并展示了如何将它扩展到多语环境。多语自动文摘与为单个源或目标语言设计的自动文摘系统相比，更为复杂。

在对多语自动文摘的历史进行简单介绍后，我们综述了自动文摘的主要方法。大多数单语自动文摘又分3个阶段：分析、转换和生成。对多语自动文摘也一样。

1）在分析阶段，文摘系统可以以图的形式来表示一个文本。这可能是语言学的语篇树或者是基于句-句相似度的矩阵表示。

2）诸如PageRank的基于图的算法或根据相关性对句子进行分类的基于机器学习的分类器会执行转换处理。

3）在生成摘要时，多语文摘面临许多语言相关问题例如分词、指代消解和文摘实或的语篇结构等问题。

在自动文摘处理中涉及很多自然语言处理领域的研究，例如语言模型、理解、共指消解、指代消解和表层实现等。每个任务都有很多问题和解决方法，这就增加了自动文摘问题的复杂性和变化性。

这些复杂性也使其评测过程（任务的定义以及解决方案的比较）变得复杂。作为研究的一部分，已经有很多人工评价和自动评价方法被提出，针对不同的任务，例如基于查询的自动文摘、单文档自动文摘和多文档自动文摘，有不同的评价方法。我们讨论了几乎所
[426] 有的主流评价方法。这些方法涉及的指标从人工标注到自动生成的都有，有时是语言无关的，有时则需要诸如分析器等与特定语言相关的资源。

然后我们介绍了用一些可用的工具资源来搭建一个自动文摘系统。许多自然语言处理

工具都可以获得（尽管不是所有语言都有）。机器翻译系统能解决一些语言障碍问题，但在那些没有被广泛研究的语言中要搭建一个自动文摘系统还是需要开发很多新的资源。

最后，我们给出了一个数据集列表，这些数据可以让我们为英语以外的语言开发和训练我们自己的自动文摘系统。

参考文献

[1] S. Wan and C. Paris, "In-browser summarisation: Generating elaborative summaries biased towards the reading context," in *HLT '08: Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies*, pp. 129–132, 2008.

[2] U. Hahn and I. Mani, "The challenges of automatic summarization," *Computer*, vol. 33, no. 11, pp. 29–36, 2000.

[3] K. Knight and D. Marcu, "Summarization beyond sentence extraction: A probabilistic approach to sentence compression," *Artificial Intelligence*, vol. 139, no. 1, pp. 91–107, 2002.

[4] R. Mitkov, "Robust pronoun resolution with limited knowledge," in *Proceedings of the 18th International Conference on Computational Linguistics (COLING'98)/ACL'98 Conference*, pp. 869–875, 1998.

[5] G. Erkan and D. R. Radev, "LexPageRank: Prestige in Multi-Document Text Summarization," in *Proceeding of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2004.

[6] E. Hovy and C.-Y. Lin, "Automated text summarization and the SUMMARIST system," in *Advances in Automated Text Summarization* (I. Mani and M. Maybury, eds.), Cambridge, MA: MIT Press, 1998.

[7] D. R. Radev, S. Teufel, H. Saggion, W. Lam, J. Blitzer, H. Qi, A. Çelebi, D. Liu, and E. Drabek, "Evaluation challenges in large-scale multi-document summarization: the mead project," in *Proceedings of the Association for Computational Linguistics 2003*, 2003.

[8] A. Lenci, R. Bartolini, N. Calzolari, A. Agua, S. Busemann, E. Cartier, K. Chevreau, and J. Coch, "Multilingual summarization by integrating linguistic resources in the MLIS-MUSI project," in *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC'02)*, 2002.

[9] D. K. Evans, J. L. Klavans, and K. R. McKeown, "Columbia NewsBlaster: Multilingual news summarization on the web," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL): Demonstration Papers at HLT-NAACL 2004*, pp. 1–4, 2004.

[10] H. Saggion, "Multilingual multidocument summarization tools and evaluation," in *Proceedings of the 5th International Conference on Language Resources and Evaluation*, 2006.

[11] H. Saggion, "SUMMA: A robust and adaptable summarization tool," *Traitement Automatique des Langues*, vol. 49, no. 2, 2008.

[12] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, "GATE: An architecture for development of robust HLT applications," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 168–175, 2002.

[13] G. Demetriou, I. Skadiņa, H. Keskustalo, J. Karlgren, D. Deksne, D. Petrelli, P. Hansen, G. Gaizauskas, and M. Sanderson, "Cross-lingual document retrieval categorisation and navigation based on distributed services," in *Proceedings of the First Baltic Conference. Human Language Technologies: the Baltic Perspective*, 2004.

[14] I. Mani, A. Yeh, and S. Condon, "Learning to match names across languages," in *Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, pp. 2–9, 2008.

[15] S. Mille and L. Wanner, "Multilingual summarization in practice: The case of patent claims," in *Proceedings of the 12th Annual Conference of the European Association for Machine Translation (EAMT)*, pp. 120–129, 2008.

[16] A. Leuski, C.-Y. Lin, L. Zhou, U. Germann, F. J. Och, and E. Hovy, "Cross-lingual C*ST*RD: English access to Hindi information," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 2, no. 3, pp. 245–269, 2003.

[17] C. Orăsan and O. A. Chiorean, "Evaluation of a cross-lingual romanian-english multi-document summariser," in *Proceedings of the Sixth International Language Resources and Evaluation*. European Language Resources Association (ELRA), 2008.

[18] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 335–336, 1998.

[19] R. Mihalcea and P. Tarau, "Textrank: Bringing order into texts," in *Conference on Empirical Methods in Natural Language Processing*, 2004.

[20] X. Ji and H. Zha, "Correlating summarization of a pair of multilingual documents," *Research Issues in Data Engineering, International Workshop on*, vol. 0, p. 39, 2003.

[21] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of Research Development*, vol. 2, no. 2, pp. 159–165, 1958.

[22] H. P. Edmundson, "New methods in automatic extracting," *Journal of the ACM*, vol. 16, no. 2, pp. 264–285, 1969.

[23] J. Kupiec, J. Pedersen, and F. Chen, "A trainable document summarizer," in *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 68–73, 1995.

[24] W. C. Mann and S. A. Thompson, "Rhetorical structure theory: Toward a functional theory of text organization," *Text*, vol. 8, no. 3, pp. 243–281, 1988.

[25] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," in *Computer Networks and ISDN Systems*, pp. 107–117, Elsevier Science Publishers B. V., 1998.

[26] M. A. K. Halliday and R. Hasan, *Cohesion in English*. London: Longman, 1976.

[27] D. Marcu, *The Theory and Practice of Discourse Parsing and Summarization*. Cambridge, MA: MIT Press, 2000.

[28] K. Ono, K. Sumita, and S. Miike, "Abstract generation based on rhetorical structure extraction," in *Proceedings of the 15th Conference on Computational Linguistics*, pp. 344–348, 1994.

[29] F. Schilder, "Robust discourse parsing via discourse markers, topicality and position," *Natural Language Engineering*, vol. 8, no. 2/3, pp. 235–255, 2002.

[30] D. Marcu and A. Echihabi, "An unsupervised approach to recognizing discourse relations," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 368–375, 2002.

[31] D. Marcu, "Discourse trees are good indicators of importance in text," in *Advances in Automatic Text Summarization* (I. Mani and M. Maybury, eds.), Cambridge, MA: MIT Press, 1999.

[32] H. Lüngen, C. Puskas, M. Bärenfänger, M. Hilbert, and H. Lobin, "Discourse segmentation of german written text," in *Proceedings of the 5th International Conference on Natural Language Processing (FinTAL 2006)*, 2006.

[33] C. Sporleder and A. Lascarides, "Using automatically labelled examples to classify rhetorical relations: An assessment," *Natural Language Engineering*, vol. 14, no. 3, pp. 369–416, 2008.

[34] C. Aone, M. E. Okurowski, and J. Gorlinsky, "Trainable, scalable summarization using robust nlp and machine learning," in *Proceedings of the 17th International Conference on Computational Linguistics*, pp. 62–66, 1998.

[35] C.-Y. Lin, "Training a selection function for extraction," in *Proceedings of the 8th International Conference on Information and Knowledge Management*, pp. 55–62, 1999.

[36] Y. Ouyang, S. Li, and W. Li, "Developing learning strategies for topic-based summarization," in *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, pp. 79–86, ACM, 2007.

[37] F. Schilder and R. Kondadadi, "Fastsum: Fast and accurate query-based multi-document summarization," in *Proceedings of the Association for Computational Linguistics: HLT, Short Papers*, pp. 205–208, 2008.

[38] K. M. Svore, L. Vanderwende, and C. J. C. Burges, "Using signals of human interest to enhance single document summarization," in *Proceedings of the 24rd National Conference on Artificial Intelligence*, pp. 1577–1580, 2008.

[39] C. J. C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. N. Hullender, "Learning to rank using gradient descent," in *Proceedings of the 22nd International Conference on Machine Learning*, pp. 89–96, 2005.

[40] M.-R. Amini, A. Tombros, N. Usunier, and M. Lalmas, "Learning-based summarization of xml documents," *Journal of Information Retrieval*, vol. 10, no. 3, pp. 233–255, 2007.

[41] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to rank: from pairwise approach to listwise approach," in *Proceedings of the 24th International Conference on Machine Learning*, (New York, NY, USA), pp. 129–136, ACM, 2007.

[42] C. Wang, F. Jing, L. Zhang, and H.-J. Zhang, "Learning query-biased web page summarization," in *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, pp. 555–562, 2007.

[43] D. Radev, V. Hatzivassiloglou, and K. R. Mckeown, "A description of the CIDR system as used for tdt-2," in *DARPA Broadcast News Workshop*, 1999.

[44] V. Hatzivassiloglou, L. Gravano, and A. Maganti, "An investigation of linguistic features and clustering algorithms for topical document clustering," in *Proceedings of the 23rd ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 224–231, 2000.

[45] K. R. McKeown, R. Barzilay, D. Evans, V. Hatzivassiloglou, J. L. Klavans, A. Nenkova, C. Sable, B. Schiffman, and S. Sigelman, "Tracking and summarizing news on a daily basis with Columbia's NewsBlaster," in *Proceedings of the 2nd International Conference on Human Language Technology Research*, pp. 280–285, 2002.

[46] C. Lin and E. Hovy, "Manual and automatic evaluation of summaries," in *Proceedings of the Document Understanding Conference (DUC-02)*, 2002.

[47] C. Lin, "Summary evaluation environment," 2001 http://www.isi.edu/ cyl/SEE.

[48] H. V. Halteren and S. Teufel, "Examining the consensus between human summaries: Initial experiments with Factoid analysis," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL) Workshop*, 2003.

[49] A. Nenkova and R. Passonneau, "Evaluating content selection in summarization: The Pyramid method," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2004.

[50] C. Lin and E. Hovy, "Automatic evaluation of summaries using n-gram co-occurrence statistics," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2003.

[51] C. Lin, "ROUGE: A package for automatic evaluation of summaries," in *The Workshop on Text Summarization Branches Out*, 2004.

[52] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "IBM research report BLEU: A method for automatic evaluation of machine translation," in *IBM Research Division Technical Report, RC22176*, 2001.

[53] E. Hovy, C.-Y. Lin, L. Zhou, and J. Fukumoto, "Automated summarization evaluation with basic elements," in *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, 2006.

[54] E. Charniak, "A maximum-entropy-inspired parser," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, 2000.

[55] M. Collins, "Three generative lexicalized models for statistical parsing," in *Proceedings of the Conference of the Association for Computational Linguistics*, 1997.

[56] D. Lin, "A dependency-based method for evaluating broad-coverage parsers," in *IJCAI-95*, 1995.

[57] G. Heidorn, "Intelligent writing assistance," in *A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text* (R. Dale, H. Moisl, and H. Somers, eds.). New York: Marcel Dekker, 2000.

[58] L. Zhou, C. Lin, D. Muntenanu, and E. Hovy, "Paraeval: Using paraphrases to evaluate summaries automatically," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2006.

[59] C. Callison-Burch, P. Koehn, and M. Osborne, "Improved statistical machine translation using paraphrases," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2006.

[60] L. Zhou, C. Lin, and E. Hovy, "Re-evaluating machine translation results with paraphrase support," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2006.

[61] E. Filatova and V. Hatzivassiloglou, "Event-based extractive summarization," in *ACL Workshop on Summarization*, 2004.

[62] S. Tratz and E. Hovy, "Summarization evaluation using transformed basic elements," in *Text Analytics Conference (TAC-08)*, 2008.

[63] A. Louis and A. Nenkova, "Summary evaluation without human models," in *Text Analytics Conference (TAC-08)*, 2008.

[64] C. Y. Lin and E. Hovy, "The automated acquisition of topic signatures for text summarization," in *Proceedings of the Conference of the Association for Computational Linguistics (HLT/NAACL)*, 2000.

[65] G. Giannakopoulos, V. Karkaletsis, G. Vouros, and P. Stamatopoulos, "Summarization system evaluation revisited: *N*-gram graphs," in *ACM Transactions on Speech and Language Processing*, vol. 5, no. 3, 2008.

[66] H. Daumé III and D. Marcu, "Bayesian query-focused summarization," in *Proceedings of the Conference of the Association for Computational Linguistics*, 2006.

[67] P. V. Ogren, P. G. Wetzler, and S. Bethard, "ClearTK: A UIMA toolkit for statistical natural language processing," in *UIMA for NLP workshop at Language Resources and Evaluation Conference (LREC)*, 2008.

第13章

Multilingual Natural Language Processing Applications: From Theory to Practice

问答系统

Nico Schlaefer, Jennifer Chu-Carroll

13.1 概述和历史

问答系统能从信息库中检索到用户所需的答案。大多数传统信息检索系统采用关键词的搜索范式。比起单纯使用关键词的搜索，问答系统用自然语言提问的方式，更加直观，表达也更清晰。除此之外，信息检索系统以文章或文档的形式回复用户的查询，而问答系统则能提供既准确又切合主题的答案。利用网络获得的信息源通常是巨大的、冗余的，也是最新的。高级检索技术使用本地文本语料库进行检索。这种技术要求信息源必须是预先加工过的，并且只限制在特定领域（比如医疗、法律或内网数据），在评测方面则要求结果是可以比较的并且可重复生成的。

近来研究最火热的问题类型是**事实型问题**，如命名实体类的问题就寻求精确的答案（例如，土耳其的首都是什么）。**列表型问题**则找到这类事实性问题的答案列表，（例如，北大西洋公约组织包括哪些国家？）研究者尝试处理具有复杂答案的问题，比如**定义性问题**、**关系问题**和**观点问题**。其中定义性问题要求系统给出特定话题的信息，人物传记也包括在内，例如，爱因斯坦是谁？关系问题，比如塔利班和基地组织的关系是怎样的？观点问题，比如人们中意宜家家居的什么？本章我们主要关注那些用于事实型问答系统的方法和算法，也可以使之适用于回答列表型问题。事实型问题适用于阐释现代问答系统的原理，用于事实型问答系统的算法解决方案和评测方法与解决具有复杂答案问题的系统相比更加成熟。参见13.11节中关于其他问题类型的介绍。

问答系统主要的挑战是自然语言的灵活性、丰富性和模糊性，这些都导致问题中包含的信息和文本的答案经常不匹配。尽管简单的关键词匹配可以成功地识别许多问题的正确答案，但是具备常识和逻辑推理的能力都是不可少的，比如在RTE（识别文本蕴涵）任务中开发的技术[1]。另外的挑战来自于时间表达和陈述，它们具有时间上的敏感性。当回答类似这种问题“Which car manufacturer has been owned by VW since 1998?”的时候会遇到一些困难。比如1998年的报纸文章中只包含了短文Volkswagen today announced the acquisition of Bentley。为了能够识别出正确答案，问答系统必须明确Volkswagen和VW指代同一实体，而且Bentley是car manufacturer，同时也需要推断出acquisition表示ownership的意思，时间表达today与1998是一致的。

进一步说，在单一的文件里可能没法找到问题的答案，在这种情况下，将多个资源中的信息相结合就变得十分必要了。比如像这样的问题：索尼公司总部设立在哪个国家？尽管在文件中没有明确指出索尼总部位于日本，但两个独立的文件可能提到总部在东京，东京又是日本的一座城市。另外一种情况是将一个问题分解成多个子问题，最后的答案由这些子问题的答案构成。例如，“哪一个国家赢得足球世界杯和欧洲杯桂冠”？这个问题的答案是这两次比赛结果的交集。

问答系统的研究可追溯到 20 世纪 60 年代，一些专家系统在受限领域内得以发展[2]。BASEBALL 系统[3]设计用来回答有关美国棒球联盟的问题。LUNAR 系统[4]则提供阿波罗号从月球带回的一些岩石样本的回答。这两个系统都依赖于结构化知识源，这些结构化知识源是由相关的领域专家人工构建的，它们还不太容易拓展到更普遍的领域。一些早期的自然语言对话系统也包含了基本的问答功能。比如，由 Winograd [5] 开发的 SHRDLU 系统可以处理玩具领域的自然语言对话，这个领域包括少数的物体，用户可以与该系统对话来操控它或探寻关于世界的各种状态。问答系统另外一个早期的应用为像 QUALM [6] 这样的阅读理解系统，该系统可以加工一篇文章，并回答有关它内容的问题。这些系统都不再依赖于手工的知识库，但是限制在一个相当狭小的领域。20 世纪 90 年代是一个转折点，开放领域的问答系统 MURAX [7] 几乎可以回答非结构化文本中有关任何话题的问题，它使用了在线百科知识回答有关一般常识的事实型问题。

英语问答研究的主要动力在于一年一度的评测。这项评测开始于 1999 年，由文本检索会议（Text REtrieval Conference，TREC）创办[8]。参评系统基于新闻专线语料库和其他非结构化文档集合来回答事实类、列表型、定义类和关系类问题。2008 年问答系统转到了文本分析会议（Text Analysis Conference，TAC）[9]，焦点随之转到观点问题。跨语言评测论坛（Cross-Language Evaluation Forum，CLEF）为其他欧洲语言[10]建立了一个类似的评测平台，NTCIR（NII Test Collection for IR Systems）研讨会每年为亚洲语言[11]举办一个相似的评测平台。TREC 和 TAC 关注单一语言的问答任务，并且问题和信息源都是英文的，CLEF 和 NTCIR 还引入了跨语言的任务，即问题语种和所给资源的语种不同。

434

如今许多问答系统都提供网络接口，并且可以进行在线测试。这些系统包括由麻省理工大学开发的 START 系统[a]、ASK.com 系统和 Wolfram Alpha[b] 系统，微软与谷歌都把基本的问答系统性能融合进搜索引擎。近年来，两个 TREC 的早先系统，卡内基梅隆大学的 OpenEphyra[c] 系统和麻省理工大学的 Aranea[d]，都进行了开源，并提供下载。

13.2　架构

在近几年显然许多 QA 架构被采用，但 QA 系统绝大部分是基于一套核心流水线，包括问题分析、查询生成、搜索、候选答案生成以及答案打分等组件。问题分析组件使用一些技术从问题中挖掘句法和语义信息，这些技术包括答案类型分类，句法和语义分析以及实体命名识别。在查询生成阶段，信息就被转化成一个搜索查询的集合，会有不同程度的查询扩展。这些查询被传递给搜索组件以从知识源中检索所需的信息。搜索结果由候选答案生成组件进行加工，得出或提取所需粒度的候选答案（比如事实型问题或者定义性问题）。回答评分组件为上个步骤得到的答案进行评估，并且通常会合并相似的候选答案。在这个阶段，知识源可以重用来为各个候选答案提供证据，最后得出的结果是一系列按照置信度排名的答案。

图 13-1 描述了这个典型架构，并通过例子展示了如何处理一个句子。在文本格式中输入：Which computer scientist invented the smiley? 在这个简单的例子中，QA 组件决定

---

a　http://start.csail.mit.edu/。

b　http://www.wolframalpha.com/。

c　http://sourceforge.net/projects/openephyra/

d　http://www.umiacs.umd.edu/~jimmylin/downloads/Aranea-r1.00.tar.gz。

了这个问题是要查找类型为 computer scientist 的答案，并提取关键词 invented 和 smiley。查询生成组件通过答案类型与提取的关键词为搜索引擎构建一个查询。有了这种查询，搜索组件检索从文本语料库（如 Web）中检索出段落，如图 13-1 所示。在候选生成阶段，命名实体作为候选答案被提取出来。最后，回答评分组件使用一些特征来估计每个候选的可信分值，这些特征包括检索排序，在搜索结果中候选出现的数量，以及与预测的答案类型是否匹配。得分最高的候选 Scott E. Fahlman 作为最有可能的答案被返回。

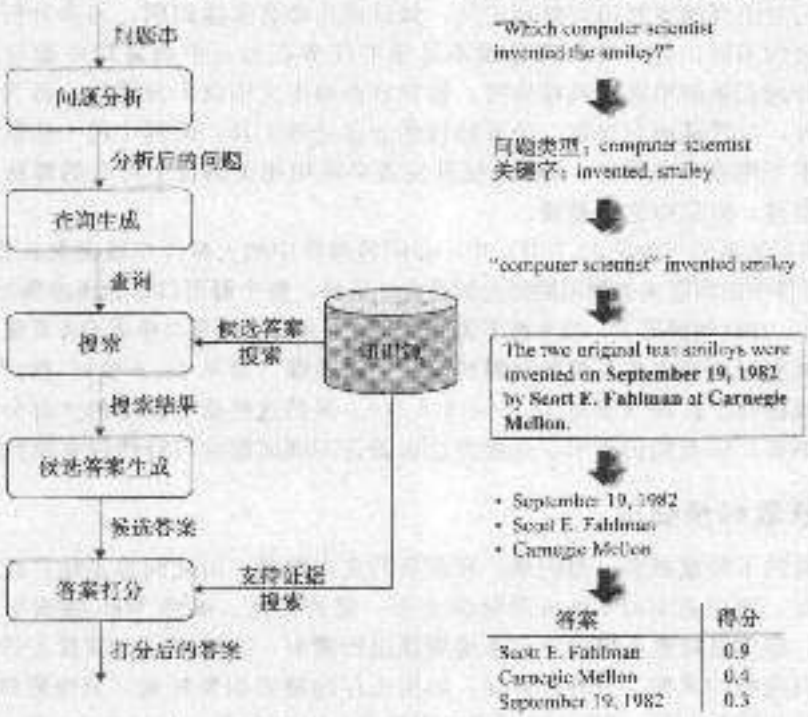


图 13-1 典型的 QA 架构（左）以及问题示例的处理流程（右）

大多数问答系统原则上遵循这一典型架构，虽然有些系统引入一些变化，包含额外的组件或改变系统组件之间的流向。例如，Harabagiu 等人 [12] 在他们的系统架构中引入反馈循环，从而可以能流尝试多个策略，例如当前面更精确的策略失败时采用具有较高的召回率策略。START 问答系统 [13] 与典型的 QA 架构不同，它分解复杂的问题并以嵌套的方式回答。例如，Where was the 20th U.S. president born? 可以这样回答，先检索总统的名字（James A. Garfield），然后使用此信息来找到他的出生地（Orange Township）。最后，现代问答系统对于一个问题往往采用多策略的方法来回答，其中几个独立的算法并行运行再将其结果合并（比如 Chu-Carroll 等 [14]，Nyberg 等 [15]）。这种并行方法计算代价可能非常高，但它也被证明是非常有效的，因为多个组件可以相互补充和加强，所以最终决定能被推迟到所有的执行路径的结果都是已知的之后。

435

QA 系统通常依赖于现有的信息检索引擎来搜索本地文件集或网络，以获得相关文档或段落。因此，问题分析组件和查询生成组件可以看作是将自然语言问题转变为抽象的查询集的预处理阶段。这些查询可用于潜在的搜索引擎。候选生成组件和答案评分组件可以视为从搜索结果中生成准确回答的后处理阶段。此处我们集中于这种基于搜索的问答方式。注意，有一些不需要传统文本搜索的系统，比如，Clifton 和 Teahan [16] 从文本资

436

源中自动提取知识关系并将其存储在知识库，更近一些的是 Wolfram Alpha 系统采用人类建造的知识库作为答案源。在运行时，QA 系统将问题与知识库中的项匹配，而不是在非结构化的文本中进行搜索。

这里概述的 QA 架构大部分都是不受语言限制的，亚洲语言和除英语外的欧洲语言的系统都采用相似的架构，而且也有一些系统可以用于其他语言，只需要在流水线布局上做少量改变。例如，JAVELIN 系统 [17] 最初是为英语开发的，后来被应用于中文和日语。然而，潜在的跨语言的自然语言处理工具，如分词和命名实体识别、句法分析器可能随不同语言有明显的不同，对一种语言是微不足道的任务在另一中语言却可能是具有挑战性的。比如，分词在英语中具有高精确度，但在日语和中文中就十分困难，因为单词之间缺少空格。此外，尽管英语有丰富、公开的自然语言处理工具，但其中的一些核心技术在其他语言中是不可用或不准确的。QA 系统开发者必须用相关语言中可行的算法实例化流水线中的每个组件，以适应这些差异。

在跨语言问答系统（参见 13.7 节）中，在问答架构中纳入翻译步骤成为必要，最常见的是，在 QA 组件中的问题或查询项翻译为源语言。另外，整个源可以作为离线预处理步骤被翻译到问题语言，在这种情况下，流水线不需要修改，但本质上还是与单语 QA 系统一样。

下面，我们讨论建立相关源的材料的离线处理步骤（参见 13.3 节），然后更详细地描述 QA 流水线的每个阶段（参见 13.4～13.6 节）。虽然这些章节提出的大部分算法和技术适用于各种语言，但我们仍指出了这些方法因语言不同而影响可行性和有效性的差异。

13.3 源获取和预处理

互联网提供了数量最多、范围最广和最新的文本数据，因此问答系统广泛利用了互联网信息。但是，使用更小的本地可用资源也有一定的优点。虽然 Web 搜索引擎的算法细节是未知的，但信息检索系统可以对本地资源进行索引，这样可以让开发人员对检索算法和搜索结果有完整的掌控。另外，Web 如用现存的搜索引擎搜索，只能原样使用，而本地可用资源可进行预处理、增强、并扩充有用信息。Web 内容、Web 搜索引擎用到的特征和算法都会随着时间的变化而变化，Web 的评价结果往往不具备可比性或可重现性。因此，如 TREC、TAC、CLEF、NTCIR 中的 QA 任务的比较评测，都采用静态的参考语料库。尽管
437 额外的资源，例如 Web 也允许被随意使用。实际应用中，QA 使用的技术还需要具备快速的响应时间和高可用性，这样会使实时 Web 搜索和随后检索相关网站变得不可能。如果知识领域含有不能被显示在 Web 上的机密数据或专业知识，那么就需要本地的索引和检索。

可以通过对知识领域的分析而选择初始的资源集。例如，新闻语料库对诸如政治事件、经济、体育等问题来说，是一个有用的资源；百科全书良好地覆盖了有关常识和知名实体的问题；博客资源可以用来处理意见问题。信息资源的可用性很大程度上取决于语言的种类，英语 QA 系统中频繁使用的一些资源可能在其他语言中使用得不广泛，或者不存在。比如 QA 系统普遍利用的在线资源，例如维基百科，则主要在英国使用。如果发现相关资源在开发集上可以提高搜索性能，则这些资源可以逐步增加到 QA 系统中，但需要注意的是，如果在小样本问题上选择资源，可能会导致过拟合。

文本文件集合被索引之前，通常需要一系列的预处理步骤。首先，大多数的问答系统会把字符转换成统一的编码方式并且替换符号，外文字符，也即对文本进行规范处理。这是改善搜索结果的一个必要步骤，以便于相似答案的合并，并支持基于答案（answer key）的自动评测。此外，低质量的源，如抓取得到的网页，可能需要进一步处理，包括

去除噪声（比如不支持语言的广告、文本）和校正拼写。接下来，文档往往被分割成句子来支持单个句子、句子边界对齐的段落的检索。根据源语言种类，可以进一步把句子分割成单词或者更小的文本单元，例如词素、字符 n 元组或者单个字符。在英语和其他欧洲语言中，词通常用于索引和检索的基本文本单元。通常需要把词语进行词干化来提高系统召回率并减少索引量，但这也会导致多义词的错误匹配。

在日文的 QA 系统中，文本通常被切分成词素。因为缺少单词之间的空格，所以日文分词比较困难，但是使用统计方法例如序列类型，可以相对有效地完成这一任务[18]。通常，一句话中包含多种字符类型的文字（片假名、平假名、汉字），那么字符类型间的变化可以对边界（识别）提供有用的线索。一些日文 QA 系统也对单个字符或者字符 2 元组进行索引，这样可以提高系统召回率，但是也会为搜索结果引入更多噪声。目前，还没有确定哪种方法效果最好。中文的词语也缺乏空格分隔，而且在一个文档中通常只包含一种字符类型，这就导致对于未在词典中出现的词语的识别和分隔比较困难。因此尽管目前中文分词[19]技术有提高，使得系统支持词语级别的索引，但大多数的中文 QA 系统都在字符级别进行索引和检索。

少量系统对源文本进行共指消解（也称为指代消解）。常见的指代类型是指向文本前面提到过的命名实体的代词和名词。例如，代词“he”可能代表一个特定的人，名词“city”可能代表前面提到的一个特定的城市。共指消解能确保相关术语的出现位置接近，从而改善段落的搜索，但这需要足够精确的算法。Hickl 等人[20]采用保守的方法，使用启发式方法解决代词和名词的共指问题。此外，包含时间约束的问题的回答性能，可以通过规范化问题和源中的时间表示提高。例如 Moldovan、Clark、Bowden[21]报告说，他们的系统把诸如“annually”和“each year”之类的表达替换为规范型（canonical form）这样使得系统在回答“How many grants does the Fullbright Program award each year”问题时，可以检索出包含“Fulbright awards approximately 4500 new grants annually”的候选段落。（TREC 15，Question 249.5）

一些信息资源提供了可以被 QA 系统利用的元数据（metadata）这些元数据也可以用到 QA 系统的多种处理流程中。例如，查询术语的扩充可以利用维基百科内部链接的锚文本和文章的自动聚定向提供的相关概念，这样可以提高搜索召回率[22，23]。此信息也可以在问答评分时，用于合并和加强相似的候选[24]。

通常利用多种类型的句法和语义标注对源进行标注，例如词性、命名实体类别、实体关系。如果这些标注信息合并到索引项中，就可以提供额外的信息明确表示更多的约束查询，全面提升搜索效果。词性和命名实体信息可以由检索组件利用，以确保查询与句法或语义相符的问题术语项实例进行匹配。例如，如果问题是关于“Washington”城市，那么提到“Washington”总统的信息是不相关的而且不应该被检索。此外，可以把搜索限制为包含希望得到的答案类型的段落。这种方法可以减少搜索结果中的噪声。但是，如果命名实体识别的召回率不高，那么相关段落可能会被错过。

句法和语义关系还可以用来制定更精确的查询。例如，在回答“Which companies did Sun Microsystems acquire?”时可以限定句法约束，“Sun Microsystems”是“acquire”的主语而不是它的直接宾语，以避免检索到讨论 Oracle 收购 Sun 内容的段落。在语义层面，可以用 Sun 是 acquisition 的施事而不是受事，对搜索进行约束。Prager 等人[25]和 Moldovan 等人[26]将命名实体类型的信息合并到搜索索引中，以增加搜索结果的相关性。Tiedemann[27]在荷兰语的段落检索中利用了多层标注信息。用句法依存分析器处

理句子并扩充了同类、句法关系、命名实体标签和复合词（compound term）。利用这些额外的信息层的查询在很多 CLEF 问题上的性能要优于基于关键词的方法。Bilotti 等人[28]在新闻语料库上进行了分句、语义分析、命名实体识别的预处理。这些标注信息用于表述查询项的语义角色和命名实体类型约束的结构化查询。在 TREC 数据上的实验表明，结构化的查询相比基于关键词的查询，能够检索到更多排名很高的相关文档。13.5.1 节我们在搜索组件的讨论中，将给出结构化查询的例子。

435 除了对搜索性能的潜在影响，源语料库的预标注可以在运行时减少标当大的计算成本。在 13.5.2 节我们讨论的候选答案提取的结构化匹配方法中，这些结构化信息依赖于对问题和语料库中的句子进行句法或者语义分析。虽然预处理语料的手段为这些技术带来了很大的效果提升，但只有在具备大规模并行硬件或者响应时间不是重点的前提下，预处理才是可行的。例如，Cui 等人[29]为源预标注命名实体类型和句法依存树以在运行时加速他们的候选提取算法。另一方面，大型文件集标注的计算代价很高，并且新源的集成是费时和繁琐的。此外，每当标注方案或算法改变时，标注必须更新，搜索索引必须重建。

经过预处理的文档可以被 IR 系统索引，如 Indri[⊖] 和 Lucene[⊜] 都是可用的开源软件。这些系统主要是为了处理英文而开发，但是也适用于其他的语言，因为它们支持任意通过空格分隔的词元流，这些词元可以是词语、词素或者单独的字符。预处理阶段提取出的结构化信息，如实体之间的关系，通常存储在知识库中，以支持答案的快速和精确查找（参见 13.5.3 节）。

13.4　问题分析

问题分析阶段使用多种核心技术来抽取问题的信息，这些信息用于提供给下游组件，通常，需要识别查询串中的关键术语和短语，以便搜索组件在原文中检索出相关的文档和文本段落。拥有很少或没有语义信息的功能词（例如，冠词、代词、连词、助动词）通常被丢弃。复合词如“pass away”、“computer science”、“leave of absence”，通过查找字典和本体库，如 WordNet [30] 和 FrameNet [31, 32] 可以识别。大多数系统也利用命名实体识别工具库，识别常见类型的实例，如人名、地名、数字。

另外，往往利用句法和浅层语义分析器分析问题，并转换成结构化的表示形式。从问题中提取的结构信息可以在搜索阶段表述更精确的查询，并在候选提取时确保候选答案同问题中提到的实体的关系一致。在 13.5.2 节候选抽取的结构化匹配讨论中，我们对句法和语义表示的性质进行更详细的阐述。

大多数事实型 QA 系统的问题分析阶段所涉及的关键功能是对答案类型进行分类，即预测用户期望得到的答案类型。答案类型集合通常在事前已被定义。例如，问题“Who invented the light bulb?”对应的答案是一个人，然而问题“How many people live in Bangkok?”寻求的回答是一个数字。搜索组件利用答案类型作为约束，只对包含该预测
440 类型实例的文本段落进行搜索。候选结果抽取组件也只识别类型一致的实例作为候选回答，或者在回答评分阶段提升相同类型的候选回答的排名。

有些问题期望得到更加具体的答案类型（比较“In which city is the Colosseum?”与“Where is the Colosseum?”），为了支持不同的分类粒度，这些类型通常被组织成本体库。

⊖ https://www.lemurproject.org/indri/。
⊜ http://lucene.apache.org/。

一个类型本体库的例子如图 13-2 所示。典型地，QA 系统的开发者会手工构建本体库，它也反映了构建者希望系统所处理领域的深度和广度。在开放领域的 TREC QA 任务中，参与者采用几十个或者最多几百个类型的本体库，以便覆盖这些评测中的大部分问题[33, 34]。

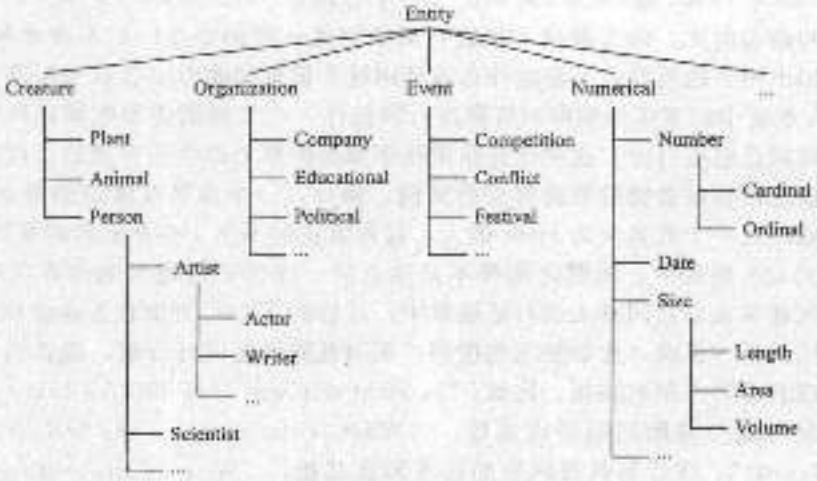


图 13-2 答案类型本体库实例

答案类型分类器可以为简单的正则表达式集合，其中每个表达式都同本体库中的一个类型关联，并匹配寻求此答案类型的问题。可以在给定开发集上通过去除问题中的不相关词元，并匹配剩下的词汇和句法变化，进而构造出匹配模式串。表 13-1 给出了例子。这个方法很容易实现，而且处理模板规定的问题类别时效果很好。为了达到更好的泛化能力，需要根据问题的词汇和句法特性设计更通用的分类规则，或者在人工标注的问题/答案类型对上，构造统计模型拟合这些特征，并且基于这些特征对问题类型做预测。

表 13-1 问题答案类型表达式实例

答案类型	正则表达式
公司	(what | which) (company | corporation)
日期	(when | on which date)
长度	how (long | tall | deep)
地点	(where | which (place | spot | site))
数字	(how many | what (is | was) the number)
作者	(who | which) (author | writer | novelist)

在这两种方法中，问题的焦点词（focus word）是决定答案类型的重要指示（例如，问题“Which country is the largest in population?”中的“country”或者“Who invented the light bulb”中的“Who”）。通过句法分析可以相当容易地得到问题的焦点词，进而可以把问题类型通过人工映射规则、用训练数据中学到的对应关系、利用外部资源（例如 WordNet [35, 36]）所提供的同义词和上位词信息，映射到静态的本体库中。为了保证答案类型分类器涵盖更广泛的类型，在上述例子中，需要把“company”、“corporation”和“enterprise”都映射到本体库的答案类型“company”中。另外一些特征，如主动词及其和焦点词的语法关系，也可以预测答案类型。例如，如果主动词是“eat”，直接宾语是焦点词，那么这个问题很可能是有关“food”类型的。

441

尽管基于静态类型本体库的 QA 策略相对容易实现、速度较快，并且在过去的 QA 评

测中证实非常有效，但是它们也有很大的局限性。通常问题类型的分类是一个艰难的决策，当给定了一个错误的问题类型后，QA 系统通常很难再恢复。此外，静态类型系统的覆盖范围有限而且可能不够具体。例如，对于问题“What redhead made Bobbie Gentry's 1970 song 'Fancy' a hit again in the 90s?”的中心词是“redhead”，本体库中可能不存在相应的条目和命名实体。作为替代，系统必须采用更一般的类型，如人或者歌手。最后，即使正确识别出回答的类型，下游组件有效利用这个信息的能力也会对 QA 系统的性能产生影响。QA 系统中的实体识别器对资源进行预标注，产生预期类型的候选回答，或者根据类型匹配对候选进行打分，这些任务在精确率和召回率上都是不完美的，可能会抽取出类型错误的候选回答或者错过预测类型的实例。例如，一个命名实体识别器会把“actor”类型错误地标注成一个其名字为 actor 的人，这样就识别不出一些不出名的演员。

在交互式 QA 场景中，问题之间都不是独立的，连续的问题可能存在关联和相互提及。这个情况通常发生在同真人的对话场景中。目前的 TREC 评测任务也尝试通过一些形式，对这个场景进行建模，比如把问题按照一系列共同话题进行分组，或者引入对前面的问题、回答或者系列主题的共指。比如，“In what city was the 1989 All-Star Game held?”的下一个问题可能与前面问题形成共指：“What is the name of the ballpark where the game was played?”，或者与前面问题的回答形成共指：“What is the seating capacity of the ballpark?”（TREC 15，Target 161）。为了有效处理这些依赖关系，QA 系统必须在抽取问题的关键术语和结构化信息前，解决共指问题。一系列问题下的共指（或者回指）问
442 题是比较难的，因为给定的上下文很少，而且 TREC 的大部分系统都借助于启发式方法处理任务中常见的共指问题（如 Hickl 等[20]）。

因为本组件是为下游的处理过程提供分析结果，所以应该根据分析结果的准确度以及下游模块如何使用这些分析结果来确定分析的类型。为了构建事实型 QA 系统，一般采用句法分析器识别问题的中心词，然后确定回答的类型，也会使用命名实体识别工具确定符合预测类型的候选回答。然而大多数目前效果最好的 QA 系统，也包含额外的分析过程，如共指消解、关系识别、语义分析。这些组件是现成的，而且在英语和其他一些语言（大部分是欧洲语言）的应用上可以得到相当准确的结果，但是对于通常研究较少的语言是有所差别的。是否采用这些工具，其他组件依赖这些工具的结果的程度，应该由经验决定。

13.5　搜索及候选抽取

现代 QA 系统通常检索非结构化数据源得到相关文档或者段落集合，同时也使用命名实体识别工具得到期望回答的类型，然后通过对问题和检索得到的文本做结构对齐，或用模式中匹配出相关的子串，进而抽取出候选回答集合。另外，经常出现的问题类型的回答，可以从结构化或者已经存在的半结构化资源中抽取，或者利用离线预处理（offline preprocessing）方式事先产生。这些内容已在 13.3 节中介绍。本节介绍这两个主要的候选回答抽取方法。因为现在大多数研究集中于从文本源中提取出子串来得到候选回答，而不是利用从源文本得到的信息合成候选回答，所以我们把候选回答抽取技术看成是候选回答生成技术的特殊情况并加以讨论。

13.5.1　非结构化资源搜索

在给出问题分析的结果后，很多 QA 系统构造一个或者多个查询，并在非结构化文本的索引上检索出相关文档集合。这些被检索出来的文档集合一般用于为接下来的系统组件

识别并评价候选回答。这些查询的构造复杂度不同，可以为简单的关键词，也可以为带有权重和位置限定符（proximity operator）的复杂查询，还有使用了源文的句法和语义标注信息的结构化查询（参见13.3节）。表13-2展示了问题“When did Apple buy Coral Software?”（answer，1989）产生的查询样例，这些查询申用于互联网搜索google和局部搜索Indri。

关键词查询是最通常的做法，也经常带来最高的召回率，然而，结构化查询会包含更多的约束，因此降低了误报率。但是，结构化查询的效果不仅取决于能否通过问题产生正确的查询，也取决于参考语料库标注的精确率和召回率。例如，表13-2表明，如果语料库上的关系识别运算，只能从类似“X bought Y”的语句中识别出“buy”关系，那么表 443
13-2中最后一个结构化查询的作用就很有限。相反，如果使用覆盖率更广的关系识别工具，可以从“X paid $20M in stock options for Y”中识别出相同的关系，那么，结构化查询很可能检索出一些不太复杂的查询所漏掉的相关文档。因此，语义丰富的查询效果，很大程度上取决于识别问题和语料库语义特征的组件的准确率。

表13-2 对问题“When did Apple buy Coral Software?”的查询

查 询	搜 索	描 述
Apple buy Coral Software	Google	简单关键词查询
Apple buy “Coral Software”	Google	要求短语 *Coral Software* 出现
Apple buy OR purchase OR acquire “Coral Software”	Google	相关项的析取
Apple # weight (1 buy 0.5 purchase 0.3 acquire) #1 (Coral Software)	Indri	相关项较少权重。#1 (...) 在Indri中等价于Google中的引号
#combine [org] (Apple) # weight (1 buy 0.5 purchase 0.3 acquire) # combine [org] (#1 (Coral Software))	Indri	*Apple* 和 *Coral Software* 必须在源文本中被标注为机构 (org)
#combine [sentence] (#any: date Apple buy Coral Software)	Indri	仅当包含属标注的日期才检索出句子
#combine [sentence] (# max (#combine [target] (buy # max (#combine [./arg0] (Apple)) # max (#combine [./arg1] (Coral))))))	Indri	检索包含 *buy* 事件且谓事为 *Apple*，受事为 *Coral* 的句子

虽然在荷兰语和英语[27，28]的QA系统中，成功地应用了结构化查询，但效果的提升很小，Chu-Carrol和Prager[37]提到，在英文文档上应用效果最好的命名实体和关系识别工具，能提高搜索性能。但是，因为搜索性能对于这些分析工具的准确率很敏感，所以当决定是否在其他语言上采用此方法时，应该慎重来进行实验评估。此外，结构化查询需要进行大量的源文本预处理，并且运行时成本很高。因此，关键词查询（有时结合权重、位置限定符）是现在最普遍使用的方法。

大多数QA系统检索文档集合或段落集合，其中段落包含一个或者多个句子。在相同的命中列表（hit list）长度[38]下，文档集合检索通常比段落检索得到的召回率更高，因为答案出现的位置通常不会离查询关键词很近。一个常见问题就是回指，即回指到前面句子中的关键词或者文档的标题。在日文和中文中回指现象特别多，因为倾向于采用短 444
句，如可从上文中推出（零回指），常省略主语和宾语。另一方面，如果分析搜索结果的时间成本很高，例如对结果做语义分析时，对段落进行处理则更加高效。查询关键词周围更短的段落也会产生少量不相关的候选回答，同样影响回答评分的效率和效果。

一些系统采用一种两步策略，首先检索出文档集合，然后分割成段落并对其排序。现在已经有了多种不同的相关度评价算法，这些算法通常类似于 IR 系统中众所周知的检索模型，对段落和问题进行相似度比较。例如，利用对查询术语的逆文档频率分数的累加进行段落排序、查询和段落术语向量的余弦相似度比较，以及 Okapi BM25 权重。Tellex 等[33]对多种段落排序算法做了定量对比实验。

一些 QA 系统也使用自动查询扩展技术来检索更多的相关结果。通常，利用多种信息扩充查询术语，如形态变体[12]、关联概念如 WordNet 中的同义词或者上位词和其他的本体资源[40, 41]，或者是从半结构化资源（维基百科的锚文本[22]和重定向[23]）中抽取得到的相关术语。然而，由于大多数词语都会有一词多义性质，所以虽然大多数问题的上下文都提供了充足的术语消歧信息，但识别出正确的词义并且映射到本体库中的正确解释也不是一件简单的任务。例如，“What movie star played the Joker in The Dark Knight?”中的术语“star”可以被解释成“celebrity”和“actor”，“What star on Orion's belt is most visible to the naked eye?”就可用“celestial body”来扩展。

可以根据参考语料库的冗余程度来决定是否需要查询扩展。在对冗余程度很高的互联网语料进行检索时，可以选择依赖语料中语言的自然变化代替查询扩展。另一方面，如果资源冗余很少，例如内部网的 QA 系统，就有必要运行查询扩展来获得合理的召回率。一般来说，查询扩展可以提高 QA 系统的平均性能，而且在评测任务中也有很广泛的使用。然而，一些系统只在召回率较低的时候才进行查询扩展，以减小添加不相关术语从而污染查询的风险[12, 41]。

另一种自动查询扩展方法是伪相关反馈（Pseudo Relevance Feedback，PRF），利用初始查询由从源文本中检索出相关文本段落，然后从检索结果中抽取术语扩充查询。一些系统进行网页搜索并从搜索引擎产生的摘要片段中抽取术语[42, 43]。然而，PRF 是否对 QA 任务有帮助，这点目前还没有定论，一些研究已报告表明它实际上会降低系统性能[43]。

13.5.2 非结构化源文本的候选抽取

根据查询类型的不同，搜索结果可以是文档、段落甚至单独的命名实体。目前研究者已经提出了从高层的搜索结果中抽取出事实回答的不同技术，通常，系统利用多种算法的组合处理不同的问题类型，并弥补单个算法的不足。 [445]

1. 基于类型的候选抽取

迄今最常用、最有效的候选生成策略之一，利用了问题分析阶段抽取出的答案类型信息（参见 13.4 节）。答案类型根据事先定义的、静态的类型本体库、命名实体识别定义。在候选回答抽取阶段，利用命名实体识别从源文本中抽取出同预测答案类型一致的实例（Moldovan[26]，Prager[25]）。如果可以详尽罗列出一个类型的所有实例，例如美国总统和工作日名称（weekday name），识别候选回答最优效的方法是把字典放到内存中。这个方法通常有很高的召回率，但是因为查找过程中没有消除歧义，所以精确率低。正则表达式方法适用于数字类型，例如数字或者日期，更加复杂的类型，例如人名、机构名和地名（Washington 可以是人名、城市名和州名）需要更复杂的启发式或者统计模型，这种模型可利用出现位置的上下文信息。

2. 结构化匹配的候选抽取

虽然基于答案类型信息的候选抽取在处理绝大多数事实问题上有效，但这个方法事先

假设在问题中表明了答案类型，而且用户认为命名实体识别器可以合理识别该类型的实例。考虑如下问题例子，“What is Indianapolis known for?”和“What word was coined by Karel Capek for a mechanical man in his play R. U. R?”。这些问题的焦点词是“What”和“Word”。第一个问题没有问题类型，第二个问题的问题类型很常规，所以几乎没用。

再者，问题类型分类是一种面向召回率的方法，只简单地根据类型信息抽取候选回答，而没有判断候选回答在语义关系上是否满足问题。因此，如果错误的候选答案同问题中的术语共现次数较多，也可能被选择。例如，问题“Who killed Lee Harvey Oswald?”(TREC 8，Question 110) 构造出的查询关键词，可能会检索出短语“Lee Harvey Oswald Killed Jon F. Kennedy”，因此人名“John F. Kennedy”很可能作为候选回答被抽取出来。另外，候选句子中经常包含预期回答类型的多个实例，而单纯的基于类型的抽取策略不能区分这些实例。Light 等 [44] 分析了 TREC 问题集中的样例并估计出，如果系统不采用额外的技术区分相同句子中这些期望类型的多个实例，而只采用基于类型的候选抽取技术，那么性能上界只能达到 70% 的精度。而且，只有在假设完美的问题分类、搜索和命名实体识别的基础上才能达到这个上界。本节探讨的候选抽取策略可以弥补基于类型抽取策略的不足，并在其他方面对其进行补充。

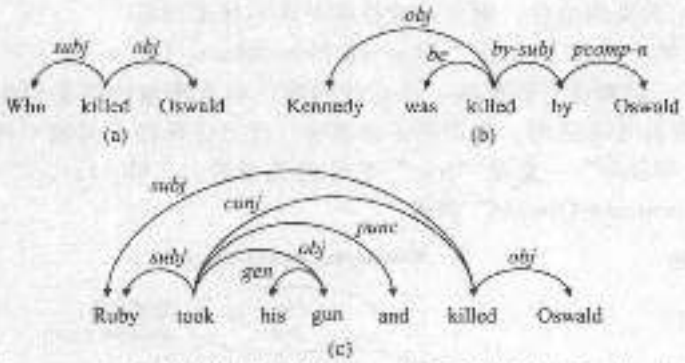


图 13-3 一个问题的依存分析树 (a)，以及两个候选句子 (b)、(c)

结构化匹配策略对问题和源文本中的句子进行分析，并试图在句法或者语义结构上对两者对齐。这些技术确保候选回答和问题的实体间有正确的关联关系，因此可以提高精确率。QA 系统的开发者通常利用开源的分析器抽取出需要的句法或者语义信息，但是对于不同的语言这些工具的可用性和效果不同。结构化匹配的一个可能的选择是，利用依存分析工具 [41，45，29] 抽取出句法依存关系。图 13-3 给出了问题“Who killed Oswald?”，候选句子“Kennedy was killed by Oswald”和“Ruby took his gun and killed Oswald”的依存分析树结构。分析树利用工具 Minipar [46] 得到，这是众多英语开源依存分析器的一种。需要注意的是，基于类型的方法可以从这两个句子中抽取出人名并且作为候选回答。如果候选回答按照同问题关键词的接近程度进行排序，那么“Kennedy”很可能被选为最可能的回答。利用结构化信息，我们可以从依存分析树中抽取出节点上的一组依存关系，得到依存路径。在此例中，我们可以从问题 (a) 中导出路径“Who SUBJ OBJ Oswald”，从句子 (b) 和 (c) 中分别导出路径“Kennedy OBJ by-SUBJ PCOMP_N Oswald”和“Ruby SUBJ OBJ Oswald”。问题的依存路径同句子 (c) 而不是 (b) 抽取出的路径匹配，所以“Ruby”而非“Kennedy”可以被确定为候选回答。

这种方法的一个通常弊病是，由问题和候选句子的句法差异导致的不匹配。Attardi

446

等[41]在原有句法结构上应用简单的启发式方法，推导出额外的依存路径来处理此问题，进而增加匹配的机会。例如，可以去除停用词，并连接经过这个词的两个节点来简化依存路径。在句子（b）中，停用词“by”可以去除并且把路径“killed by SUBJ PCOMP N Oswald”简化为“Killed SUBJ Oswald”。Cui等[29]利用统计方法学习训练数据集中依存关系的相似性，并进行依存路径的相似匹配。

或者，结构化匹配可以基于浅层语义信息，例如谓词-论元结构[42，36]和语义框架[47]。这里我们简述谓词-论元结构的使用，此结构捕捉事件和参与这些事件的实体，事件是动词，参与者为主语、宾语和动词的间接格论元（oblique argument of the verb），每个参与者都被指定为一个参与该事件的语义角色，例如“agent”（通常标注成“ARG 0”）、“patient”（通常标注成“ARG-1”）、“location”（ARGM-LOC），或者“time”（ARGM-TMP）。PropBank语料库[48]用谓词-论元结构进行了人工标注，并可以用来训练出一个可以自动进行标注工作的语义角色标注（SRL）系统。一个常用的SRL系统是开源的ASSERT（表示 Automatic Statistical SEmantic Role Tagger）[49]分析器，对
467 问题和候选句子进行分析，然后做语义结构的匹配。利用13.5.1节介绍的技术，问题术语可以用相关的概念进行补充，以促进与搜索结果术语的对齐。如果候选句子分析树中的论元包含了问题中丢失的信息，则可以被抽取并作为候选回答。

图13-4中的例子展示了问题“What did Peter Minuit buy in 1626?”的回答是如何应用基于为谓词-论元结构进行匹配的。注意此问题不包含明显的答案类型，因此基于类型的候选抽取策略在此处不适用。这个例子也表明，此方法保存了问题中指明的实体之间的关系（即“Peter Minuit”一定是“buy”事件的施事者，“Manhattan”是它的受事者），因此也适用于“Kennedy-Oswald”例子。



图13-4　语义角色标注和匹配例子

与基于类型的抽取策略相比，结构化匹配具有计算密集的特点，并会带来较低的召回率，因为它施加了额外的约束并且严重依赖于元信息（meta-information）[37]抽取组件的正确性，尤其是，语义分析比较慢且容易出错。一般，结构化方法会结以答案类型分析来提高效率和精确率。例如，Attardi等[41]在进行代价较高的分析前会把不包含期望类型实体的候选句子过滤掉，同样Schlaefer等[36]要求语义分析中抽取出的论元，如果作为候选回答，则必须和预测出的问题类型一致。

3. 基于表层模式的候选抽取

候选回答也可以利用表层模式进行抽取。表层模式指仅利用了搜索结构的词汇表述，并不需要句法或者语义分析。模式可以是问题术语实例化后的正则表达式模板，并同包含回答的文本段落进行匹配。例如，模板 [448]

<ORG> was (*founded* | *established*) *in* (*the year*)? <ANSWER>

可能用于抽取出关于一个给定机构的成立日期问题的候选回答。分类组件对问题进行分类，然后基于指定的类别选择一个表层模式。类别没有必要同问题类型完全相同，可以是更粗或更细粒度的。例如，段落 "Johann Sebastian Bach (31 March 1685 – 28 July 1750) was a German composer" 需要不同的表层模式来抽取 "date of birth" 和 "date of death"，但是基于答案类型抽取方式可以利用单纯的 "date" 识别器抽取出两个日期，并作为候选答案。表层串可以手工编写也可以利用训练数据的问题-回答对自动学习得到[50, 51]。更泛化的模式可自动从具体的模式中构建出来，以便匹配范围更广的相似表述方式，例如，常用词语可以用规范性、词类甚至通配符替换，命名实体类型可以被具体的类型实例所代替[52]。

表层模式同结构化匹配类似，也不依赖答案类型信息，因此适用于没有明显有用类型问题的处理，例如 "What is Enrico Fermi most known for?"（TREC 14, Question 87.5）。表层模式也可以保证问题的语义关系被保留。另一方面，这种方法仅适用于只有有限个预定义范畴的问题。例如，对于可能是问题主语的实体的适用属性，设计出它的类别和表层模式是可行的，如一个人的国籍或者职业，或一个机构的领导、规模。然而，问题 "What is the legal blood alcohol limit for the state of California?"（TREC 8, Question 41）需要有一个类别 "legal blood alcohol limit" 来保证对应的模式集的特殊性。另外，依赖于模式中的笼统性，这种方法通常会有低召回率或者低精确率的问题。具体化的模式可能会错失那些正确回答的实例，如果是在先前未见过的表层形式出现的回答，然而过于泛化的模式，会抽取出不正确的候选集合并引入噪声。

在本节讨论的候选抽取方法中，最常用的就是基于类型的候选产生方式，它依赖适用于语料的语言的应用领域的好的命名实体识别工具，或者是应用领域的本体相关库。结构化候选抽取方式依赖于更复杂的 NLP 技术，如果某个语言不存在效果好的命名实体工具，那么相应的复杂 NLP 技术也不可能存在。如果给定了目标语言内选择的问题类别充足的问题-回答对，那么表层模式方法可以保留，而且这个方法通常在英语 QA 系统中，用作对基于类型的候选抽取方式的补充，而且如果某种语言上基于类型方式的效果不好，那么此方法可能会充当更重要的作用。

13.5.3 结构化源文本的候选抽取

不同于在非结构化语料库上的搜索，QA 系统可以在结构化和半结构化源文件上进行问答查找。结构化数据通常存储在关系型数据库（relational database）或者是资源描述框架（Resource Description Framework, RDF）仓库中，实体和对应的属性存储其中。例如，数据库表可能包含著名的演员，以及他们的生日、民族、演过的电影和获得的奖项。结构化资源常常利用离线方式填入，或者通过开源资源例如 DBpedia[6] 和 Freebase[7]，利用自动关系抽 [449]

[6] http://dbpedia.org/.
[7] http://www.freebase.com/.

取技术处理非结构化源文本 [53, 16]，或者通过手工加工应用领域的相关数据。

半结构化资源的例子是结构化元素同半结构化文本混合的网站。例如，地名词典可能提供了不同国家的常见统计数据，例如国土大小、人口和官方语言，以及政策体制和经济的叙述。常用的统计数据可以存储成所有国家一样的结构化格式，叙述可以以非结构化的纯文本形式组织。类似地，维基百科的页面通常把非结构化的文本与表格结合在一起，这些表格使相同类别的实体，例如董事长或者公司，在所有页面内保持格式一致。半结构化源文本可以离线获取并转换成结构化的数据，或者在运行时透过封装存取，以便只抽取需要的或者最新的信息。

与非结构化源文本上的候选抽取方法相比，在结构化和半结构化源文本中查找回答，常常有较高的精确率但是较低的召回率，因为系统的效果受限于自动识别问题中支持的关系类型，以及正确映射此类型到源文本中表示的类型的能力。此外，构建和维护这些源的人力成本相当大。在实际应用中，往往整合结构化和非结构化源，以结合两个方法的优缺点。

13.6　回答评分

本节对常用、有趣的回答评分和验证方法做一个概述（参见 13.6.1 节）。我们进一步探讨如何整合多种证据源，合并或者强化和相似候选回答（参见 13.6.2 节）。我们也勾画出如何扩展这些技术和算法以处理事实型回答的列表（参见 13.6.3 节）。

13.6.1　方法概述

如果知识源语义冗余，即包含很多正确回答的实例，那么简单的基于频率的回答方法可以有效地从多个符合期望答案类型的实体中识别出回答。例如，Clarke、Cormack 和 Lynam [55] 先抽取出符合答案类型（例如长度）的所有实例，并利用类似信息检索中频率-逆文档频率（TF-IDF）的加权方式对它们进行排名。此算法对在搜索结果中出现频率
[450] 高的候选回答进行提升，对在知识源中总体上表现的回答进行处罚。其最根本的假设为，正确的回答在检索到的文本中最常见。这种情况常见于大规模的冗余源（例如互联网）中对问题关键词周围的段落作检索。这种方法很容易实现，并且也可以作为开发一个适用于任何语言的事实性 QA 系统的出发点。然而，如果源文件包含较少的相关段落，或者某个错误的回答经常和问题术语共现时，这种方法会失效，例如 13.5.2 节的 "kennedy-Oswal" 例子。

基于类型的候选回答抽取方法，可以与计算问题和候选段落间的词语级相似度方法结合，获得更加精确的置信分值。例如，段落中出现的问题关键词的数量（可以用 IDF 分数进行加权）和段落中这些关键词的接近程度都可以作为相关性的预判 [56, 57]。13.5.2 节中讨论的基于结构化匹配和表层模式的回答抽取算法也可以得出候选回答的置信分值。当利用句法依存路径的近似匹配时，问题和候选句子路径之间的相似度可以用于置信度估计 [29]。同样，基于浅层语义信息的结构化匹配，也可以得到反映匹配接近度的置信分值 [42]。当利用表层模式作候选抽取时，每个模式的精确率可以离线地在测试数据上估计，而且在利用此模式抽取候选回答时也可以指定为置信估计 [50]。

为了论证更深层的推理方法对问题回答的影响，Moldovan、Rus [58] 和 Moldovan 等 [59] 采用如下方法：根据问题和潜在包含回答的段落的句法分析结果，把它们转换成逻辑表达，并在回答评分阶段利用逻辑验证工具 COGEX 来合一这两个逻辑表达式。如果合一成功，则同问题 wh-slot 合一的段落中的实体，就被认为是此问题的回答。当包含回

答的段落的词汇和结构与问题相差较大时，这种方法对浅层方法的优势很明显。例如，考虑问题“Which company created the internet browser Mosaic?”和段落“A program called Mosaic, developed by the National Center for Supercomputing Applications, has been gaining popularity lately.[⊖]”。为了合一这两个文本的逻辑表达式，COGEX必须使“create”同“develop”合一，使“Internet browser Mosaic”同“Mosaic”合一，并把“National Center for Supercomputing Applications”识别成一个机构名并且使机构名的实例同“company”合一。COGEX借助从eXtended WordNet（XWN）注释[80]中自动导出的世界知识公理，实施合一。这些资源可以使“create”等同于“develop”，也利用了人工编码的NLP公理集，例如公理“复合名词的中心名词可等同于该复合名词”可以让“Internet browser Mosaic”同“Mosaic”联系起来。

新近在PASCAL RTE挑战下[1]，文本蕴涵技术已经被开发出来，要求系统判断一句话能否推导出另外一句话，而不是判断它们是否等价。例如，句子“Judge Drew served as Justice until Kennon returned to claim his seat in 1945”推导出假设“Kennon served as Justice”，但反之不然（RTE 3, Pair 12）。Harabagiu和Hickl[61]在QA系统的回答评分中结合了文本蕴涵工具，并已证实可以显著提高系统效果。 451

Magnini等[62]介绍了一种利用互联网冗余来估计候选回答置信分值的算法，即Web强化算法。对于每一个候选回答，该算法利用问题的关键词和候选回答构造出查询，并提交给互联网搜索引擎得到摘要片段。然后通过摘要中候选回答和问题关键词的接近程度，为此候选回答指定分值。此方法的基本原理为：与问题关键字密切相关的候选回答，在互联网上出现的位置也很可能同关键词非常接近。一些系统也利用外部语义资源验证回答。WordNet和从维基百科得到的结构化信息，可以验证一个候选是否是正确的类型[59, 63]。例如，WordNet中上位词关系和维基百科的文章分类都证实“Richard Feynman is a physicist”，所以问题“Which physicist developed the theory of quantum electrodynamics?”的一个合理回答是“Richard Feynman”。地名辞典类供的信息可以验证地理问题[24]。例如，CIA World Factbook[⊜]证实“Brazil is a country in South America”，因此可能是问题“Which country in South America has the largest area?”的回答。

Prager、Duboue和Chu-Carrol[64]提出了一个有趣的回答验证方法，即利用逆问题。例如，给定问题“What was the capital of Germany in 1985?”和候选回答“Bonn”，他们的方法表述了逆问题“Of what country was Bonn the capital in 1985?”然后重新执行QA查询流水线，如果逆问题得到的候选回答中包含“Germany”，那么QA系统会提高“Bonn”作为原始问题候选回答的置信度。逆问题方式可以在TREC问题上提升效果，但是它的计算量很大。

13.6.2 证据结合

通常利用统计技术结合多个证据源来对回答进行评分[56, 57, 24]。每个证据以数字或者类别特征的形式表示，统计模型利用这些特征估计候选回答正确的概率。概率的估计可以用来对回答排名，并决定最好的回答是否应该提供给用户。除了前面描述的方法，它融合的特征通常侦测性较弱，但是对正确答案的指示性较强，例如IR引擎指定搜索结果

⊖ 改编自Moldovan等人[39]。

⊜ https://www.cia.gov/library/publications/the-world-factbook/。

的排名和分值，并已其个候选回答是否同期望的答案类型匹配。统计模型也可以用于整合多种置信分值彼此不可相互比较的回答生成算法得到的候选集合。通常使用的统计方法包括逻辑回归（logistic regression）[65] 和最大熵模型 [66]。

因为搜索结果会冗余，并且由于源和上下文不同，相同的概念可以用不同的方式表达，所以 QA 系统最终产生的候选回答列表通常包含相似甚至等价的实体。等价的回答可
452 能是词汇和语义上的变化，例如，缩写（VW 与 Volkswagen）、不同拼写（Al-Qaeda 与 Al Qaida）、同义词（China 与 Middle Kingdom）、测量单位不同（100℃ 与 212 下）。另外，候选回答可以在不同粒度上相似。例如，一个候选回答可以比其他的更详细（Rome，Italy 与 Italy，George W. Bush 与 Bush），或者候选回答在数字表示上的不同（13.049m 与 12.053m）。相关的候选回答也为给定候选回答的正确性提供了证据，因此应该在回答评分阶段被考虑在内。

Prager、Luger 和 Chu-Carroll [67] 提出了基于规则的相关候选回答识别方法，这种方法也基于答案类型信息。例如，可以为多种地点类型、机构类型和数字实体类型构造不同的规则集合，并保留大多数早期例子，在相关候选回答分值的基础上，可以利用启发式方法提高此候选回答的分值，同时应考虑它们的相似程度。如果事先可以确定出相对小规模的高频答案类型集合，这个方法就会很明智，而且对于 TREC 评测中的问题很有效果。与 Prager 等人在一个独立的后处理步骤中提高相关候选集合的分值不同，Ko、Si 和 Nyberg [24] 为了在回答评分步骤中强化相似候选回答，提出了一个集成的方法。一个统一的概率框架融合了估计候选回答正确性的特征和衡量候选之间相似度的特征。字符串距离测度，如编辑距离（Levenshtein distance）和余弦相似度（cosine similarity），被用于衡量候选回答间的词汇相似度，而从 WordNet、维基百科编制而来的同义词数据库和手写规则，被用来识别语义相似的回答。

已经提出的一个挑战是跨语言的回答合并。这个场景在跨语言 QA（参见 13.7 节）中很可能出现，此时需要把从多个语种的源文件中抽取的回答进行汇总，它们支持的附属证据也需要被结合。未来的 NII Test Collection IR（NTCIR）会考虑跨语言回答合并的评测。

在回答一个事实型问题时，如果最高分低于先前规定的阈值也会返回空结果，QA 系统常常返回分值最高的回答或不回答。如果问题提得不好（如 "Who is the prime minister of the United States"）或者原文中不包含回答，则返回空结果是有效的选项。对于一个完美的 QA 系统来说，虽然可以提供一个答案，通知用户系统失败比返回一个错误和可能是误导的回答更好。

13.6.3　扩展到列表型问题

本节通过事实型问题的例子对一些技术进行说明，但对于希望得到事实性回答列表的问题也是适用的（例如，"What books did George Orwel write?"）。回答列表型问题时，通常返回最好的 n 个回答，n 可以在问题中给定（例如，"Who were the last ten presidents of the United States?"），或者用估计得到的置信分值动态确定。例如，一个系统可以选择从最优回答直到给定的置信度阈值的回答列表，或者在发现置信度大幅降低时停止选择。有效的相似候选问题合并对于列表型问题尤其重要，这样可以避免返回相同回答的
453 多个实例（例如，"Bill Clinton 和 William Jefferson Clinton"）。

13.7　跨语言问答

在跨语言问答中，提问的语言与知识源的语言不同，当将一个单语系统扩展到跨语言

系统的任务时，开发人员可以把源文档翻译成提问的语言，或者将问题和关键词翻译成文档的语言。哪种方法更有效的争论源于IR学界，但很难有确凿的证据来证明。因为每个翻译方向需要不同的机器翻译系统，所以两种方法性能上的不同可能源于方法的不同或者是机器翻译系统的不同。通过对TREC评测的跨语言的IR数据进行查询翻译和文档翻译的比较，McCarley[88]曾试图解决这个问题。这两种翻译方法将英法和法英数据集进行比较，而机器翻译模型也用相同数据集进行训练。不存在一种方法始终优于另一种方法，但是一个能够执行查询和文档翻译的混合系统与任何一种单一功能系统相比是更为有效的办法。令人惊讶的是，它甚至比人工的查询翻译更为有效，这表明在计算成本可控的条件下，混合方法是最理想的方法。

在QA系统中，这两种翻译方向都已成功实现。支持源翻译方法的普遍证据是对机器翻译错误的鲁棒性。假设重要的问题关键词没有准确翻译出来，则不可能选取正确的答案。另一方面源文本经常包括多个相关段落，若正确翻译其中一个段落，就已经足够。此外，源文本翻译可以在离线预处理状态下进行，在运行时不需要额外的费用，也不需要对QA流水线做任何的修改。另外，源文本若比较大，或者需要支持多种语言的问题，离线翻译也许较昂贵。因此，研究员可能需要采取更有效的但不大精确的机器翻译算法。进一步说，仅当源文本可以本地储存和索引，源翻译才是行得通的，因此这并不适用于Web搜索。此外，与13.3节中讨论的其他源文本预处理步骤类似，在机器翻译系统改进以后，源翻译也需要更新。

Bowden等人[89]在2007年的CLEF英法、英葡QA任务中进行了源翻译。这两项任务包含事实性问题、列表性问题和定义性问题。这些源文本被离线翻译成英语。单语QA系统从这些翻译中抽取答案。最后的答案被映射回法语或葡萄牙语源的相应文本片段。因此QA系统可以在没有对新的语言进行适应的情况下被使用，尽管需要放宽回答中一些语义和句法的限制来弥补不准确的翻译。

对于一些QA任务来说，翻译整个源文本是可行的。在问题分析时翻译问题，或者翻译从问题中摘取的关键词和短语并对源文本进行随后的流水线步骤是更为普遍的做法。在翻译整个问题时，可根据上下文消除词语间的歧义。此外，问题的句法结构映射到源语言中，可在源文本语料库中选取句法结构相似的句子。另一方面，如果问题是复杂的，就很难找到一个准确的、符合句法的翻译，在这种情况下，单独的关键词翻译可能更有效。在源语言中解析问题，仅当更可靠的NLP工具对该语言可用时才翻译关键词，可能是更好的做法。可结合多种翻译系统，通过投票方式来提高翻译的准确性[17]。有用的在线资源包括Google Translate[⊖]和BabelFish[⊜]，可用于问题翻译，维基百科中对其他语言写的文章的链接和Wiktionary[⊝]可用于关键词翻译。

另一个挑战来自于欧洲和亚洲语言间的跨语言QA系统中的专有名词的翻译，不管采用何种翻译方法。例如，一个美国人的名字可有完全不同的日语翻译，但是也可以用片假名书写系统转写，甚至可以在日语文本中用罗马字母来书写。进一步说，如果名字用片假名转写，经常有不止一种拼写方式，不同的作者可能会用不同的字符。这种歧义可以通过检索和匹配相关文本时考虑多种翻译来解决，或在预处理阶段用规范的形式来代替源中指向相同的实体的专名（参见13.3节）。后一种方法避免了运行时的计算开销，但在

454

⊖ http://translate.google.com/。
⊜ http://babelfish.yahoo.com/。
⊝ http://www.wiktionary.org/。

运行时需要识别指向同一实体的表达式并将其映射为唯一的表示。

13.8　案例研究

在这一节中我们提出一个案例研究，进一步说明本章前面几节介绍的概念和技术。我们把问题“The 2008 Summer Olympics took place in which city?”（答案：Beijing）作为一个运行的例子，我们展示如何在图 13-1 中介绍的典型的 QA 流水线的每一阶段来处理这个问题，包括问题分析、查询生成、搜索、候选生成和回答评分。我们运用一些最常见的和有效的 QA 算法，但请注意，这绝不是在 QA 中已经实现或可行的功能的完整概述。

一个典型的 QA 系统的问题分析组件显示于算法 13-1。给定问题字符串，我们的示例系统抽取两个命名实体（NE）“2008”和“Summer Olympics”。这可以通过使用正则表达式匹配年份的实例和一个常见事件或体育比赛的列表来完成。在字典中查找后来被用作查询术语的另外的关键术语。例如，本体 WordNet[a] 把“take place”识别为一个复合动词。一个功能词的列表被用来排除不能作为查询术语的词，只留下关键术语“2008”、
455 “Summer Olympics”、“took place”和“city”。

算法 13-1　典型 QA 流水线的问题分析组件

```
AnalyzeQuestion(String question)
    aq.question ← question
    // 抽取关键术语用于查询生成
    aq.nes ← extractNamedEntities(question)
    aq.keyTerms ← extractKeyTerms(question, aq.nes)
    // 抽取句法依存关系
    aq.depParse ← parseSyntacticDependencies(question)
    // 对某个类型的候选抽取，预测答案类型
    aq.focus ← extractQuestionFocus(aq.depParse)
    aq.answerType ← predictAnswerType(aq.depParse, aq.focus)
    return aq
```

图 13-5 显示了问题实例的一棵依存分析树。依存分析树可借助公开可用的工具，如 Minipar[b] 或 Stanford Parser[c] 生成。关键术语“city”把疑问词“which”作为其限定词，因此可以很容易地确定为问题的焦点。我们的示例系统没有一个“city”（城市）名命名实体识别程序，使“city”不适合作为一个答案类型。然而，在 WordNet 里，“city”的上位词是“location”（地理位置），而“location”可用各种开源工具包如 OpenNLP[d] 获取。因此我们把问题焦点词“city”映射为更加普遍的答案类型“location”。

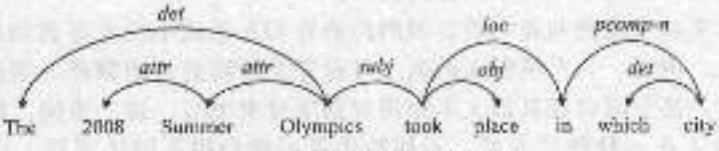


图 13-5　问题实例的依存分析树

[a] http://wordnet.princeton.edu/。
[b] http://webdocs.cs.ualberta.ca/~lindek/minipar.htm。
[c] http://nlp.stanford.edu/software/lex-parser.shtml。
[d] http://opennlp.sourceforge.net/。

查询生成组件，如算法 13-2 所示，构建了在前一步中提取的关键术语的查询。我们假设 Indri 信息检索系统[⊖]将用于搜索阶段，因此查询必须符合 Indri 的查询语言。我们的 QA 系统还用在外部结构资源中找到的相关术语来扩展问题关键术语。对于示例问题，从维基百科侧边栏提取的结构化信息可以用来识别出“Olympic Games”是“Summer Olympics”的一个上位词（指的是冬季奥运会和夏季奥运会）。另外，WordNet 为关键术语“take place”提供了另一同义词“happen”。给定了关键术语和相关术语，下面的查询可以构造出来：

```
1) #combine[p](2008 #1(Summer Olympics) #1(took place) city)

2) #combine[p](2008
        #weight(1 #1(Summer Olympics) 0.3 #1(Olympic Games))
        #weight(1 #1(took place) 0.5 happened)
        city)
```

算法 13-2 典型 QA 流水线的查询生成组件

```
GenerateQueries(AnalyzedQuestion aq)
    queries ← ∅
    //在维基百科、WordNet等中查找相关术语
    relatedTerms ← getRelatedTerms(aq.keyTerms)
    // 从关键术语和相关术语产生查询
    queries ← queries ∪ keyTermQuery(aq.keyTerms)
    queries ← queries ∪ expandedQuery(aq.keyTerms, relatedTerms)
    return queries
```

Indri 的查询操作符 #combine[p](…) 用于检索已经在源中预标注的段落。段落可以根据标点符号和现有的标记自动标注。在 #1(…) 中括起的术语必须作为连续词元出现在源中，类似于在网页搜索时用的引号。第二个查询包括相关术语但把较低的权重给同义词（0.3）和上位词（0.3），这里权重的具体数值只是用于说明，但应该用一组开发集问题来调整以优化搜索性能。

搜索组件，如算法 13-3 所示，使用查询从一个 Indri 索引集合来检索文本。这些索引是从本地源，如新闻语料库和一个维基百科的拷贝建立的。每个查询是针对每个源单独运行，搜索结果需合并。典型的 QA 系统检索 10～100 个段落，但为了简单起见，我们假设对问题实例，只检索到以下三个段落：

1) The 2008 Summer Olympics took place in Beijing, China, from August 8 to August 24, 2008.

（2008 年夏季奥运会在中国北京举行，从 2008 年 8 月 8 日到 24 日。）

2) The Summer Olympics were held in Peking in 2008, in Athens in 2004, and in Sydney in 2000.

（夏季奥运会 2008 在北京举行，2004 在雅典举行，2000 年在悉尼举行。）

3) When I visited Beijing during my trip to China in 2008, the airport was crowded because of visitors who came to watch the Olympics.

（当我 2008 年访问北京时，机场很拥挤，因为大量游客前来观看奥运会。）

⊖ http://www.lemurproject.org/indri/。

算法 13-3　典型 QA 流水线的搜索组件

```
Search(String[] queries, String[] indexPaths)
    passages ← ∅
    foreach query (queries)
        foreach indexPath (indexPaths)
            passages ← passages ∪ retrievePassages(query, indexPath)
    return passages
```

在候选生成阶段（参见算法 13-4），两个互补的答案抽取策略应用于检索的段落。面向召回率的候选生成器从所有段落中抽取出类型为“location”的实例，返回候选“Beijing，China”、“Peking”、“Athens”、“Sydney”、“Beijing”和“China”。第二个更精确的候选抽取器对段落进行依存句法分析，把问题分析树（参见图 13-5）中的依存路径“Olympics subj loc pcomp n city”与第一段中的相似路径“Olympics subj loc pcomp-n Beijing”进行匹配，并提取“Beijing，China”作为唯一的候选。

算法 13-4　典型 QA 流水线的候选生成组件

```
GenerateCandidates(String[] passages, AnalyzedQuestion aq)
    answers ← ∅
    foreach passage (passages)
        // 答案类型匹配
        answers ← answers ∪ extractAnswerType(passage)
        // 句法依存路径匹配
        answers ← answers ∪ extractDepPath(aq.depParse, passage)
    return answers
```

回答评分组件（参见算法 13-5）从候选答案中计算出能预测其正确性的特征，并根据特征值估算置信分值。当前最好的系统通常合并几十个甚至数以百计的特征，但在这里，我们集中于三个特征，可利用不同形式的证据：

1）在所检索的段落中候选答案的频率。

2）问题和检索出候选答案的段落的文本间的相似性。我们简单地把相似性估计为占段落的关键词占所有问题关键词（即 2008、Summer、Olympics、took、place 和 city）的比例。

3）一个二元特征，指示候选答案是否由一个段落蕴涵。这里，我们假设识别文本蕴涵（RTE）算法是可用的，能预测出候选 Beijing、China、Peking、Beijing and China（且仅有这些候选）是被蕴涵的正确答案。

算法 13-5　典型 QA 流水线的回答评分组件

```
ScoreAnswers(String[] answers, String[] passages, AnalyzedQuestion aq)
    scoredAnswers ← ∅
    foreach answer (answers)
        // 特征1：在检索的段落里的答案的频率
        freq ← countAnswerFrequency(answer, passages)
        // 特征2：段落和问题间的文本相似度
        textSim ← calculateTextSimilarity(answer, passages, aq.question)
        // 特征3：真答案是否一个段落蕴涵问题
        entailed ← recognizeEntailment(answer, passages, aq.question)
        // 从特征值估算置信分值
        score ← estimateScore(freq, textSim, entailed)
        scoredAnswers ← scoredAnswers ∪ (answer, score)
```

```
// 通过增强最佳分值来强化相似的答案
scoredAnswers = reinforceSimilarAnswers(scoredAnswers)
return scoredAnswers
```

第三个段落再次说明了 RTE 的困难。对一个人来说，通过阅读该段落，很明显得出 C58
北京是一个正确的答案。然而，系统必须推理出 Olympics 指 Summer Olympics 而不是指 Winter Olympics，因为任意一年只可能举办这两种奥运会之一。此外，它还得出结论，因为游客前来观看比赛，比赛可能会在北京举行，幸运的是，由于源文本中的语义冗余，所以这些复杂的处理过程往往在实践中是不必要的。在这个例子中，正确答案可以更容易地从第一或第二个段落中提取。

每个候选通过前面的特征表示的证据被合并为一个置信分值。特征值和总的置信分值见表 13-3。这里，我们简单地为每个特征赋予相同的权重，使用平均特征值作为评分。然而，在实践中，利用逻辑回归或其他的统计技术，建立一个模型并与人工判断的候选回答 C59
数据集相拟合，是更有效的。该模型可以通过机器学习工具，如 Weka[⊖] 或 MinorThird[⊜] 来估计。

在这个例子中，“Beijing”和“China”是并列排名第一。然而，一个问答强化算法通过字符串匹配识别出“Beijing, China”比这些候选更具体，并且在 WordNet 里“Peking”被列为“Beijing”的同义词。QA 系统增加这些类似候选的置信分值，并返回“Beijing, China”为最佳答案。因为它是最具体的。如果最高的候选得分低于预定义的阈值（例如，0.8），QA 系统反而会表明它不能找到答案。

表 13-3 对问题“The 2008 Summer Olympics took place in which city?”的候选回答的特征和置信分值

候选	频率	文本相似度	类型	分值
Beijing, China	1	0.53	1	0.84
Peking	1	0.5	1	0.83
Atlanta	1	0.5	1	0.80
Sydney	1	0.5	1	0.80
Beijing	2	0.33	1	1.11
China	2	0.33	1	1.11

13.9 评测

在过去的十年里，问答系统的研究是由有组织的评测工作所驱动，也创造了大量的社区资源可用于进一步开发：对英语 QA 有文本检索会议（TREC）以及后来的文本分析会议（TAC），对英语和其他欧洲语言的跨语言 QA 有跨语言评测论坛（CLEF），对亚洲语言 QA 有 NTCIR。下面，我们描述这些评测任务，并讨论评测方法和常见的性能指标。

13.9.1 评测任务

1999～2007 年 TREC 英语问答系统每年进行评测（TREC 8～16）[8]。这个评测论坛已成为英语 QA 研究的主要驱动力，评测中产生的问题集和答案关键字已成为标准测试集。最初，TREC 主要集中于事实型问题，但在后来的几年中，列表型、定义型和关系型

⊖ http://www.cs.waikato.ac.nz/ml/weka/。
⊜ http://minorthird.sourceforge.net/。

问题也增加了。虽然早期的测试集由独立的、自包含问题组成，在最近的评估中，问题被组织为具有一个共同主题的系列，包含对主题、前面的问题和答案的引用。2008 年，QA 任务被移入到刚设立的 TAC [9]，集中于询问观点、观点的对象和观点持有者的列表型问题。表 13-4 说明了 TREC 和 TAC 问题的常见类型。

表 13-4　TREC 和 TAC 的常见问题类型实例

问题类型	问题例子
事实型	Who was the first American in space? (*TREC 8*, *Question* 811) Where is the Valley of the Kings? (*TREC 9*, *Question* 249)
列表型	Name 20 countries that produce coffee. (*TREC 10 list task*, *Question* 1)
定义型	Who is Aaron Copland? (*RTREC 12 main task*, *Question* 1901) What is a golden parachute? (*TREC 12 main task*, *Question* 1905)
关系型	Are Israel's military ties to China increasing? (*TREC 14 relationship task*, *Question* 17)
观点型	Who likes Mythbusters? (*TAC*, *Question* 1018.1) Why do people like Trader Joe's? (*TAC*, *Question* 1047.2)

TREC 评测中，系统需要从文本集合中检索出答案，这些集合最初是报纸上的文章，后来增加了从网络上爬下来的博客网站。在 TAC 中，这个博客集合是观点型问题候选答案的唯一来源。博客语料库提出了新的挑战，因为它的规模不允许全面的预处理，而很差的文本质量则需要更鲁棒的自然语言处理工具。在 TREC 和 TAC 两个评测中，系统必须对每个答案从源文本中提出一个文档，该文档包含答案并提供证据。因此，即使系统允许利用额外
[460] 的源，如用 Web 来产生候选答案并进行评分，最后答案的理由必须源于评测的源。

NTCIR 研讨会关注的是亚洲语言的单语和跨语言 QA [11]。QA 系统中提出了事实型问题，最近，也包括更复杂的事件、传记、定义、关系问题，并且必须在新闻语料库中识别出答案。目前，源文本是日语或中文（简体和繁体），问题以相同的语言提出（单语 QA）或以英语提出（跨语言 QA）。在跨语言 QA 中，不需要把答案翻译回英语。虽然 TREC 集中于 QA 系统的端到端的评测，NTCIR 还提供了在团队之间交换问题分析和文档检索结果的任务。通过这种方式，参与者可以评估不同的算法组合，并对各个组件的有效性得出结论。

CLEF 评测各种欧洲语言的 QA 系统，包括保加利亚语、荷兰语、英语、法语、德语、意大利语、挪威语、葡萄牙语、罗马尼亚语和西班牙语 [10]。和 NTCIR 一样，评测也包括单语和跨语言的子任务，但是 CLEF 的特征是有更多的问题和语料库语言对。过去的评价包括事实和定义型问题，答案必须从文本质量差别很大的源文本中提取，如新闻文章、维基百科的文档、即时的讲话转录稿等。

13.9.2　判断答案正确性

[461] 虽然许多问题有多个可接受的答案，事实型答案的正确判断是最简单的，因为大多数可接受的答案是语义等价的，如 Volkswagen 和 VW。在有些情形下判断会更复杂，包括需要

返回数字答案的问题，正确答案可能是一个范围；或者可能会随时间而改变答案的问题。前者的一个例子是莱特兄弟第一次飞行的长度是多少？（TREC 11，1414 题）。对该问题 120 英尺或 120 英尺 4 英寸都被认为是正确答案。后者的例子是问一个人的年龄或一个公司的首席执行官。在过去的 TREC、TAC、NTCIR 和 CLEF 评测中，参与者提交的答案由评审员进行人工判断，评审员确定的正确答案随后可以被汇编成标准答案用于自动评测[⊖]。

评价答案的正确性，对于复杂的回答是有难度的，如对定义型问题的回答。例如，往往很难决定是否一个事实足够重要可以纳入回答中，甚至对回答的完整性是否是必不可少的。在 TREC 中，评审员在必须包括在答案中或可以不加惩罚答案的重要和可接受信息块的列表的基础上来评价回答。例如，对亚马逊河的定义型问题的完美答案（TREC 15，问题 187.7）必须提到亚马逊是世界上最长的河流，回答中包括亚马逊网站是以亚马逊河命名的信息是可接受的。评估工具，如 Nuggeteer [70] 和 Pourpre [71] 已经开发出来，可对复杂回答进行自动评估。

13.9.3 性能度量

评估 QA 系统性能的关键取决于它是否提供了问题的正确答案。然而，多年来，已经提出并采用了许多评价方法，试图用一个单一的性能指标来表示系统对回答排名的能力、对回答正确性的置信度等。

对于事实型 QA 最简单和直观的性能度量是**准确率**。设 n 是测试集的问题的数量，c 是系统返回的第一候选正确答案的问题数，那么准确率定义为

$$\text{准确率} = \frac{c}{n}$$

准确率仅基于可信度最高的答案，平均排名倒数（Mean Reciprocal Rank，MRR）度量也考虑了排名较后的正确答案。对于测试集的每一个问题 $q_i (i=1,\cdots,n)$，设 r_i 为问题产生的命中列表中第一个正确答案的排名（如果有正确答案的话）。MRR 计算如下：

$$\text{MRR} = \frac{1}{n}\sum_{i=1}^{n}\begin{cases}\dfrac{1}{r_i} & \text{如发现正确答案}\\ 0 & \text{否则}\end{cases}$$

MRR 通常是基于一个固定排名下的前几个回答。例如，MRR@5 只考虑每个问题排名最高的前 5 个答案。这个度量对描述系统的召回率是有用的，奖励能把正确答案排在答案列表较前位置的系统。MRR 通常只用于评价前 5 或 10 个候选的排名，因为它只对排在前面的候选敏感，排在后面的正确答案对这个度量几乎没有影响。 402

考虑系统对答案赋予置信度的一个度量是置信加权评分（Confidence Weighted Score，CWS），按照首选答案的置信度得分对问题进行降序排列。然后，定义 CWS 为：

$$\text{CWS} = \frac{1}{n}\sum_{i=1}^{n}\frac{\text{到问题 } i \text{ 为止正确的答案个数}}{i}$$

这个度量奖励正确答案以及在问题之间可比的可靠置信估计。

列表型问题的性能经常用 F 值来度量。设 t_i 为测试集中到第 i 个列表问题的正确答案的个数，r_i 为 QA 系统为该问题返回的答案的个数，c_i 为这些答案中正确的个数。进一步，设第 i 个问题的召回率和精确率定义如下：

⊖ 注意这些标准答案经常是不完备的且取决于参考语料库，因此需要不断更新以便在后续的实验中更精确地反映系统性能。

$$Recall_i = \frac{r_i}{t_i}, \text{并且 } Precision_i = \frac{r_i}{a_i}$$

那么 F 值是精确率和召回率的加权调和平均值：

$$F_i(\beta) = \frac{(\beta^2+1) \times Precision_i \times Recall_i}{\beta^2 \times Precision_i + Recall_i}$$

加权参数 β 确定精确率和召回率的相对重要性。β 越大，越多的权重将给予召回率，即找到所有的正确答案更重要，避免不正确的答案相对不重要。如果 $\beta=1$，精确率和召回率是同等重要的。QA 系统对一个列表问题集合的整体性能可以被定义为对 F 值的算术平均值：

$$F(\beta) = \frac{1}{n}\sum_{i=1}^{n} F_i(\beta)$$

F 值也用于评价定义型问题和其他具有复杂答案的问题。在 TREC 和 TAC 中，评审员汇集了他们认为是答案中重要的或可接受部分的信息块列表。召回率和精确率以 QA 系统产生的答案对这些信息块的覆盖率和答案的长度为基础定义［8，9］。

在 TREC 2007 评测中顶尖的 QA 系统对事实性问题的准确率为 71%，对列表型问题的 $F(1)$ 值是 0.48。在 CLEF 2008 多语种 QA 任务中［10］，最好的系统在单语任务中达到的准确率为 64%，而所有参与者的平均准确率是 24%。然而，在跨语言任务中最好的
[163] 系统只有 19%的准确率，而平均为 13%。在 NTCIR 2007 跨语言 QA 任务［72］中，对事实型问题报道的最好性能对于日语单语任务是 34%的准确率，而对于汉语单语 QA 任务是 52%。最好的跨语言系统的准确率要低得多：在英日任务中是 18%、英汉任务中是 25%。这些结果说明，在跨语言 QA 中查询或源文本的翻译成了巨大的额外挑战。TAC 评测和最近的 NTCIR 评测集中于具有复杂答案的问题，评测结果不在这里公布，因为它们并不直观，主要用于系统比较。

13.10　当前和未来的挑战

我们已经看到，问答系统通常使用简单的统计模型和启发式方法来抽取候选答案并对其排序。这种技术适用于源文本是语义冗余的且包含许多答案实例的情形，当源文本是庞大的或者问题是关于当前热门话题时这是常见的。然而，如果源文本不是冗余的，则可能需要更复杂的查询扩展技术来检索包含答案的文档和段落，并且更深层次的 NLP 和推理技术对于识别答案并找出理据是非常必要的。在最极端的语义匹配和文本蕴涵的例子中，整个源语料库中只有一个包含答案的段落，问答系统必须确定它是否蕴涵答案。通常，问题和文本段落间的语义关系并非明显，只能通过本体库和相关性计算来进行术语匹配、基于精确的句法和语义分析树来进行结构匹配、利用世界知识来进行逻辑推理，才能揭示出来。在概述（参见 13.1 节）中的 Volkswagen Bentley 例子和在（参见 13.6 节）中讨论的将 RTE 用于问答评分的例子都说明了文本蕴涵中的困难。

本章所描述的技术是从知识源中提取答案的技术而不是从文本中包含的信息中综合出答案。虽然这种技术对于大多数事实型问题都是适用的，但是当答案不是显式存在于资源中而必须从其他的陈述中推导出来的时候，这种技术就不可行了。例如，相对于文章的出版日期解析时间表达式（例如，昨日，政府宣布……），或者进行单位转换、数值相加等运算（例如，十大富豪以美元计算的联合净资产是多少？）是非常必要的。浅层的候选提取技术对于具有复杂回答的问题是不够有效的，因为组成一个自然、连贯、没有冗余的段落是非常重要的，并且答案需要从多个文档的事实中推演出来。

对于一个问答系统来说，将候选答案进行生成和评分是远远不够的，还需要对排名靠前的答案进行可靠的置信估计。如果现有的最好答案不大可能是正确的，则告知用户这个问题无法回答可能是更好的做法。不正确的回答会减少用户对于问答系统可靠性的信任，因此会影响系统对用户的有用性。置信估计对于列表型问题也是非常重要的，因为正确答案的数量是不可预估的，而取决于系统返回多少实例。在目前的系统中，置信估计在问题间经常是不一致的，而取决于各种因素如答案类型、资源的冗余或者是问题的长度。 [484]

跨语言的问答系统，例如在 NTCIR 和 CLEF 中的评测的系统，已经向着成熟的多语系统迈出了第一步。当前的系统能够把问题翻译为信息源的语言，并产生这种语言的答案，但是，答案并没有译回提问的语言。那些并不精通源文本语言的用户就会要求系统能够接受用户用自己的语言提问并返回同一种语言的答案，但是可以搜索各种语言的知识源。

TREC、CLEF、NTCIR 这些评测论坛，无疑推动了 QA 的技术进步，但是也导致了对具体相关任务的专门解决方案通常不能容易地适应新的领域和真实世界的应用。为了促进 QA 技术的实际应用，将来的研究应该集中在通用的问答算法和技术，从而可以更快地适应新的任务，并在不同领域的实现高性能[73]。

虽然迄今为止大多数的研究都集中在事实型问题和列表型问题，但是有着复杂答案的问题，例如定义型、关系型和观点型问题最近得到了更多的关注。然而，问答系统在提供复杂答案的情况下并不那么有效。问答算法和一致性自动评价方法的改进对于提高复杂问答系统的性能以达到实际应用的要求都是非常必要的。尤其难回答的问题包括：如何（how）和为什么（why）类问题，需要找出解释或理由；是非题，需要系统确定可用的信息源的联合知识是否蕴涵某个假设。处理这类问题的有效算法还需要开发。

13.11 总结和进一步阅读

问答系统可以看作目前盛行的信息检索系统的下一步发展。它们支持自然语言提问并且返回精确回答，提供了直观、高效的信息获取方式。问题回答是信息检索的强化这个观点，反映在了 QA 系统通用的架构上。大多数最先进的系统，原则上遵循如下组成流水线的组件：1）把问题转换成搜索引擎的查询；2）利用现存的 IR 系统检索出相关文本；3）抽取出候选回答并进行评分。然而，在基本的设计方式上也存在例外和变化。一些 QA 系统进行了多种预处理，以支持结构化查询，或者构造出用于查找回答的结构化信息知识库。如果先前搜索结果的召回率不能令人满意，那么执行额外的迭代搜索也是普遍做法，有些系统甚至重新运行部分或者整个流程来验证置信度高的候选回答。

我们讨论了已经应用于 QA 流水线中的问题分析、搜索和候选生成、评分阶段的多种算法，包括简单的启发式、模式匹配到统计模型，再到语义分析和推理。回答大多数事实型问题的一个简单有效的方法是利用答案类型信息。利用期望的问题类型对问题进行分类，并利用命名实体识别工具从检索得到的文本段落中抽取出这些类型的候选回答。这种方法可以作为实现一个 QA 系统的合理出发点，但是此方法受到预先定义的类型集合的限制，并在很大程度上依赖于冗余度来从匹配的期望类型候选中选择最终回答。 [485]

我们介绍了一些回答抽取和评分的算法，这些算法利用问题和段落的深层次分析克服这些限制，对不满足句法、语义或者逻辑约束的候选回答进行丢弃或者打折扣。这些算法通常可以提升 QA 性能，但是需要相当大的实现代价，并且相比单纯的基于类型的方法更为脆弱。QA 系统通常在查询生成、回答抽取和评分阶段融合不同的算法，每种算法都有自己的优劣。从非结构化的文本中抽取回答可以与结构化资源中的查找方式互补，对于常

[465]

五类型问题的回答，这种方法有很高的精确率和效率。统计类型可以结合多种证据源，并利用候选回答间关联相似度进行回答评分。

相似的架构和算法已被应用到不同语言的 QA 系统中，但在实现过程中需要解决具体语言带来的挑战。此外，深层 NLP 技术，例如语义回答抽取和文本推导，在一些语言上是不可行的，因为相关的 NLP 工具不可用或者不足够精确。跨语言 QA 系统通过翻译源文本，或翻译问题或者抽取出的关键词，来支持不同语言的问题和源文本。这两种方法在实际中都有使用，因为不能确定那种方法的效果一直地好于另一种方法。

问答学科同信息检索相比仍然处于早期阶段，但是在迅速地发展。过去的十年里，一些标准评测任务例如 TREC，CLEF 和 NTCIR 推动了研究的发展，这些评测任务为参评团体提供数据集和测量效果提升的评价标准，并组织了讨论班来分享研究思路。但是，这些计划也使研究重心导向了具体的 QA 任务，并通常得出高度专门化的解决方法，但这些方法并不足用于新的知识领域、资源和问题类型。一个主要的公开挑战是更通用的算法开发，这种算法可适用于更广泛的 QA 任务并且可以很容易适应新的任务。为了有效地解决具有复杂回答的问题，我们也有必要从回答抽取策略转向更灵活的回答产生算法。如果源文本缺少语义上的冗余，则需要更深层的 NLP 技术找到回答。最后，为了利用多语资源，例如互联网，我们需要支持用户交互，并支持多种不同语言的信息资源的 QA 系统。

对于进一步阅读，我们推荐《问答系统导论》[74]，它涵盖了基本原理和有趣的系统综述。目前也有两本较新的讨论 QA 系统的重大进展和创新方法的书籍[75, 76]。对于近期的英文 QA 出版物、测试集合以及以往的评测结果，我们推荐读者访问 NIST 网站，查阅 TREC[㊀] 和 TAC[㊁]，NTCIR[㊂] 和 CLEF[㊃] 评测论坛对于单语和跨语言 QA 系统（亚洲和欧洲语言）来说是极好的资源。

虽然我们的讨论主要集中于事实型和列表型问题的回答所涉及的算法和资源，但近些年已经做出了相当多的努力来解决其他更复杂回答的问题，例如定义型问题、观点型问题和关系型问题。Blair-Goldensohn、McKeown 和 Schlaikjer [77] 提出了回答定义型问题的一个混合方法，这种方法结合了基于知识和统计学的策略。Weischedel、Xu 和 Licuanan [78] 从人名相关的句子中自动抽取出语言学结构，例如同位语（appositive）和主题句（proposition），以回答 "Who is X?" 形式的传记型问题。最近，TREC 2004～2007 提出了多种定义型的 QA 解决策略。Kaisser、Scheible 和 Webber [79] 提出了一个简单有效的 Web 提升策略，即利用候选关键词在 Web 搜索结果中的出现频率对候选回答进行评分。Qiu 等 [80] 从源文本中抽取出句子集合，并利用结合了句法特征、信息检索分数和语言模型的统计模型对其排序。非英语语言的定义型 QA 也在 CLEF [10] 和 NTCIR [11] 中有涉及。

NRRC 对多视角问答（Multi Perspective Question Answering，MPQA）的研讨会 [81] 探讨了如何在文本语料库中识别和组织观点。Stoyanov、Cardie 和 Wiebe [82] 介绍了一个包含观点问题和已标注文档集合的数据集，在回答观点问题时，利用基于统计学和规则的过滤器剔除掉事实型信息，处理观点问题、观点持有者和观点对象等的 QA 系统在 TAC 2008 中被评测过 [9]。效果最好的系统 [83] 采用情感词典，在候选段落中识别出代表正面和反面观点的术语。

㊀ http://trec.nist.gov/。
㊁ http://www.nist.gov/tac/。
㊂ http://research.nii.ac.jp/ntcir/。
㊃ http://www.clef-campaign.org/。

TREC 2005 QA任务[84]包含了如下任务：系统必须解决实体间多种关系类型的问题，如金融的依赖关系、传播途径和组织关系。TREC 2006和2007[8]评测的复杂交互式QA任务的主题也是关系型问题。此任务是交互的：系统向评审员给出初始结果，在得到相关反馈（relevance feedback）后产生最终的回答。最近的NTCIR评测中也包含关系型问题[11]。

致谢

作者要感谢Eric Nyberg和John Prager在QA研究的多个领域分享他们的知识和经验，以及Teruko Mitamura和Hideki Shima在跨语言QA上富有见地的评论。

参考文献

[1] D. Giampiccolo, H. Dang, B. Magnini, and I. Dagan, "The fourth PASCAL recognizing textual entailment challenge," in *Proceedings of the 1st Text Analysis Conference*, 2008.

[2] R. Simmons, "Natural language question-answering systems: 1969," *Communications of the ACM*, vol. 13, no. 1, pp. 15–30, 1970.

[3] B. Green, A. Wolf, C. Chomsky, and K. Laughery, "Baseball: An automatic question-answerer," in *Proceedings of the Western Joint IRE-AIEE-ACM Computer Conference*, 1961.

[4] W. Woods, "Progress in natural language understanding: An application to lunar geology," in *Proceedings of the AFIPS National Computer Conference*, 1973.

[5] T. Winograd, *Understanding Natural Language*. New York: Academic Press, 1972.

[6] W. Lehnert, "A conceptual theory of question answering," in *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, pp. 158–164, 1977.

[7] J. Kupiec, "MURAX: A robust linguistic approach for question answering using an on-line encyclopedia," in *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 181–190, 1993.

[8] H. Dang, D. Kelly, and J. Lin, "Overview of the TREC 2007 question answering track," in *Proceedings of the 16th Text REtrieval Conference*, 2007.

[9] H. Dang, "Overview of the TAC 2008 opinion question answering and summarization tasks," in *Proceedings of the 1st Text Analysis Conference*, 2008.

[10] P. Forner, A. Peñas, E. Agirre, I. Alegria, C. Forăscu, N. Moreau, P. Osenova, P. Prokopidis, P. Rocha, B. Sacaleanu, R. Sutcliffe, and E. T. K. Sang, "Overview of the CLEF 2008 multilingual question answering track," in *Lecture Notes in Computer Science, Vol. 5706*, Springer, 2009.

[11] T. Mitamura, E. Nyberg, H. Shima, T. Kato, T. Mori, C.-Y. Lin, R. Song, C.-J. Lin, T. Sakai, D. Ji, and N. Kando, "Overview of the NTCIR-7 ACLIA tasks: Advanced cross-lingual information access," in *Proceedings of the NTCIR-7 Workshop Meeting*, 2008.

[12] S. Harabagiu, D. Moldovan, M. Paşca, R. Mihalcea, M. Surdeanu, R. Bunescu, R. Gîrju, V. Rus, and P. Morărescu, "The role of lexico-semantic feedback in open-domain textual question-answering," in *Proceedings of the 39th Association for Computational Linguistics Conference*, 2001.

[13] B. Katz, G. Borchardt, and S. Felshin, "Syntactic and semantic decomposition strategies for question answering from multiple resources," in *Proceedings of the AAAI 2005 Workshop on Inference for Textual Question Answering*, pp. 35–41, 2005.

[14] J. Chu-Carroll, J. Prager, C. Welty, K. Czuba, and D. Ferrucci, "A multi-strategy and multi-source approach to question answering," in *Proceedings of the 11th Text REtrieval Conference*, 2002.

[15] E. Nyberg, T. Mitamura, J. Callan, J. Carbonell, R. Frederking, K. Collins-Thompson, L. Hiyakumoto, Y. Huang, C. Huttenhower, S. Judy, J. Ko, A. Kupsć, L. Lita, V. Pedro, D. Svoboda, and B. V. Durme, "The JAVELIN question-answering system at TREC 2003: A multi-strategy approach with dynamic planning," in *Proceedings of the 12th Text REtrieval Conference*, 2003.

[16] T. Clifton and W. Teahan, "Bangor at TREC 2004: Question answering track," in *Proceedings of the 13th Text REtrieval Conference*, 2004.

[17] N. Lao, H. Shima, T. Mitamura, and E. Nyberg, "Query expansion and machine translation for robust cross-lingual information retrieval," in *Proceedings of the NTCIR-5 Workshop Meeting*, 2005.

[18] T. Kudo, K. Yamamoto, and Y. Matsumoto, "Applying conditional random fields to Japanese morphological analysis," in *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2004.

[19] J. Gao, M. Li, A. Wu, and C.-N. Huang, "Chinese word segmentation and named entity recognition: A pragmatic approach," *Computational Linguistics*, vol. 31, no. 4, 2005.

[20] A. Hickl, J. Williams, J. Bensley, K. Roberts, Y. Shi, and B. Rink, "Question answering with LCC's CHAUCER at TREC 2006," in *Proceedings of the 15th Text REtrieval Conference*, 2006.

[21] D. Moldovan, C. Clark, and M. Bowden, "Lymba's PowerAnswer 4 in TREC 2007," in *Proceedings of the 16th Text REtrieval Conference*, 2007.

[22] I. MacKinnon and O. Vechtomova, "Improving complex interactive question answering with Wikipedia anchor text," in *Advances in Information Retrieval*, Lecture Notes in Computer Science, pp. 438–445, Springer, 2008.

[23] B. Katz, G. Marton, S. Felshin, D. Loreto, B. Lu, F. Mora, O. Uzuner, M. McGraw-Herdeg, N. Cheung, Y. Luo, A. Radul, Y. Shen, and G. Zaccak, "Question answering experiments and resources," in *Proceedings of the 15th Text REtrieval Conference*, 2006.

[24] J. Ko, L. Si, and E. Nyberg, "Combining evidence with a probabilistic framework for answer ranking and answer merging in question answering," *Information Processing & Management*, vol. 46, no. 5, pp. 541–554, 2010.

[25] J. Prager, E. Brown, A. Coden, and D. Radev, "Question answering by predictive annotation," in *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2000.

[26] D. Moldovan, S. Harabagiu, M. Pasca, R. Mihalcea, R. Goodrum, R. Girju, and V. Rus, "LASSO: A tool for surfing the answer net," in *Proceedings of the 8th Text REtrieval Conference*, 1999.

[27] J. Tiedemann, "Integrating linguistic knowledge in passage retrieval for question answering," in *Proceedings of the Human Language Technology Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, 2005.

[28] M. Bilotti, P. Ogilvie, J. Callan, and E. Nyberg, "Structured retrieval for question answering," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007.

[29] H. Cui, K. Li, R. Sun, T.-S. Chua, and M.-Y. Kan, "National University of Singapore at the TREC-13 question answering main task," in *Proceedings of the 13th Text REtrieval Conference*, 2004.

[30] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, 1998.

[31] C. Fillmore, C. Johnson, and M. Petruck, "Background to FrameNet," *International Journal of Lexicography*, vol. 16, no. 3, pp. 235–250, 2003.

[32] J. Ruppenhofer, M. Ellsworth, M. Petruck, and C. Johnson, "FrameNet II: Extended theory and practice," *ICSI Technical Report*, 2005.

[33] X. Li and D. Roth, "Learning question classifiers," in *Proceedings of the International Conference on Computational Linguistics (COLING)*, 2002.

[34] A. Hickl, K. Roberts, B. Rink, J. Bensley, T. Jungen, Y. Shi, and J. Williams, "Question answering with LCC's CHAUCER-2 at TREC 2007," in *Proceedings of the 16th Text REtrieval Conference*, 2007.

[35] J. Prager, J. Chu-Carroll, and K. Czuba, "Statistical answer-type identification in open-domain question-answering," in *Proceedings of the Human Language Technology Conference*, 2002.

[36] N. Schlaefer, J. Ko, J. Betteridge, G. Sautter, M. Pathak, and E. Nyberg, "Semantic extensions of the Ephyra QA system in TREC 2007," in *Proceedings of the 16th Text REtrieval Conference*, 2007.

[37] J. Chu-Carroll and J. Prager, "An experimental study of the impact of information extraction accuracy on semantic search performance," in *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, 2007.

[38] C. Clarke and E. Terra, "Passage retrieval vs. document retrieval for factoid question answering," in *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2003.

[39] S. Tellex, B. Katz, J. Lin, A. Fernandes, and G. Marton, "Quantitative evaluation of passage retrieval algorithms for question answering," in *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2003.

[40] E. Hovy, L. Gerber, U. Hermjakob, M. Junk, and C.-Y. Lin, "Question answering in Webclopedia," in *Proceedings of the 9th Text REtrieval Conference*, 2000.

[41] G. Attardi, A. Cisternino, F. Formica, M. Simi, and A. Tommasi, "PiQASso: Pisa question answering system," in *Proceedings of the 10th Text REtrieval Conference*, 2001.

[42] R. Sun, J. Jiang, Y. Tan, H. Cui, T.-S. Chua, and M.-Y. Kan, "Using syntactic and semantic relation analysis in question answering," in *Proceedings of the 14th Text REtrieval Conference*, 2005.

[43] L. Pizzato, D. Mollá, and C. Paris, "Pseudo relevance feedback using named entities for question answering," in *Proceedings of the Australasian Language Technology Workshop (ALTW)*, 2006.

[44] M. Light, G. Mann, E. Riloff, and E. Breck, "Analyses for elucidating current question answering technology," *Journal of Natural Language Engineering*, vol. 7, no. 4, pp. 325–342, 2001.

[45] B. Katz and J. Lin, "Selectively using relations to improve precision in question answering," in *Proceedings of the EACL-2003 Workshop on Natural Language Processing for Question Answering*, 2003.

[46] D. Lin, "Dependency-based evaluation of MINIPAR," in *Proceedings of the Workshop on the Evaluation of Parsing Systems*, 1998.

[47] S. Narayanan and S. Harabagiu, "Question answering based on semantic structures," in *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, 2004.

[48] P. Kingsbury and M. Palmer, "PropBank: The next level of TreeBank," in *Proceedings of Treebanks and Lexical Theories*, 2003.

[49] S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky, "Shallow semantic parsing using support vector machines," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2004.

[50] D. Ravichandran and E. Hovy, "Learning surface text patterns for a question answering system," in *Proceedings of the 40th Association for Computational Linguistics Conference*, 2002.

[51] D. Zhang and W. Lee, "Web based pattern mining and matching approach to question answering," in *Proceedings of the 11th Text REtrieval Conference*, 2002.

[52] N. Schlaefer, P. Gieselmann, and G. Sautter, "The Ephyra QA system at TREC 2006," in *Proceedings of the 15th Text REtrieval Conference*, 2006.

[53] M. Fleischman, E. Hovy, and A. Echihabi, "Offline strategies for online question answering: Answering questions before they are asked," in *Proceedings of the 41st Association for Computational Linguistics Conference*, 2003.

[54] J. Lin and B. Katz, "Question answering from the World Wide Web using knowledge annotation and knowledge mining techniques," in *Proceedings of the 12th International Conference on Information and Knowledge Management (CIKM)*, 2003.

[55] C. Clarke, G. Cormack, and T. Lynam, "Exploiting redundancy in question answering," in *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001.

[56] A. Ittycheriah, M. Franz, W.-J. Zhu, A. Ratnaparkhi, and R. Mammone, "IBM's statistical question answering system," in *Proceedings of the 9th Text REtrieval Conference*, 2000.

[57] J. Xu, A. Licuanan, J. May, S. Miller, and R. Weischedel, "TREC2002 QA at BBN: Answer selection and confidence estimation," in *Proceedings of the 11th Text REtrieval Conference*, 2002.

[58] D. Moldovan and V. Rus, "Logic form transformation of WordNet and its applicability to question answering," in *Proceedings of the Association for Computational Linguistics*, 2001.

[59] D. Moldovan, C. Clark, S. Harabagiu, and S. Maiorano, "COGEX: A logic prover for question answering," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/-NAACL)*, pp. 87–93, 2003.

[60] S. Harabagiu, G. Miller, and D. Moldovan, "WordNet 2 - a morphologically and semantically enhanced resource," in *Proceedings of SIGLEX-99*, 1999.

[61] S. Harabagiu and A. Hickl, "Methods for using textual entailment in open-domain question answering," in *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, 2006.

[62] B. Magnini, M. Negri, R. Prevete, and H. Tanev, "Comparing statistical and content-based techniques for answer validation on the Web," in *Proceedings of the 8th Convegno AI*IA*, 2002.

[63] D. Buscaldi and P. Rosso, "Mining knowledge from Wikipedia for the question answering task," in *Proceedings of the International Conference on Language Resources and Evaluation*, 2006.

[64] J. Prager, P. Duboue, and J. Chu-Carroll, "Improving QA accuracy by question inversion," in *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, 2006.

[65] A. Agresti, *Categorical Data Analysis*. New York: Wiley, 2002.

[66] A. Berger, S. Della Pietra, and V. Della Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, 1996.

[67] J. Prager, S. Luger, and J. Chu-Carroll, "Type nanotheories: A framework for term comparison," in *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, 2007.

[68] J. McCarley, "Should we translate the documents or the queries in cross-language information retrieval?," in *Proceedings of the 37th Association for Computational Linguistics Conference*, 1999.

[69] M. Bowden, M. Olteanu, P. Suriyentrakorn, T. d'Silva, and D. Moldovan, "Multilingual question answering through intermediate translation: LCC's PowerAnswer at QA@CLEF 2007," in *Lecture Notes in Computer Science, Vol. 5152*, Springer, 2008.

[70] G. Marton and A. Radul, "Nuggeteer: Automatic nugget-based evaluation using descriptions and judgements," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2006.

[71] J. Lin and D. Demner-Fushman, "Methods for automatically evaluating answers to complex questions," *Information Retrieval*, vol. 9, no. 5, pp. 565–587, 2006.

[72] Y. Sasaki, C.-J. Lin, K.-H. Chen, and H.-H. Chen, "Overview of the NTCIR-6 cross-lingual question answering (CLQA) task," in *Proceedings of the NTCIR-6 Workshop Meeting*, 2007.

[73] D. Ferrucci, E. Nyberg, J. Allan, K. Barker, E. Brown, J. Chu-Carroll, A. Ciccolo, P. Duboue, J. Fan, D. Gondek, E. Hovy, B. Katz, A. Lally, M. McCord, P. Morarescu, B. Murdock, B. Porter, J. Prager, T. Strzalkowski, C. Welty, and W. Zadrozny, "Towards the open advancement of question answering systems," Tech. Rep., IBM Technical Report RC24789, 2009.

[74] J. Prager, "Open-domain question answering," *Foundations and Trends in Information Retrieval*, vol. 1, no. 2, pp. 91–231, 2006.

[75] M. Maybury, ed., *New Directions in Question Answering*. Menlo Park, CA: AAAI Press, 2004.

[76] T. Strzalkowski and S. Harabagiu, eds., *Advances in Open Domain Question Answering*. Dordrecht: Springer, 2006.

[77] S. Blair-Goldensohn, K. McKeown, and A. Schlaikjer, "Answering definitional questions: A hybrid approach," *New Directions in Question Answering* (T. Strzalkowski and S. Harabagiu, eds.), Dordrecht: Springer, 2006.

[78] R. Weischedel, J. Xu, and A. Licuanan, "A hybrid approach to answering biographical questions," *New Directions In Question Answering* (T. Strzalkowski and S. Harabagiu, eds.), Dordrecht: Springer, 2006.

[79] M. Kaisser, S. Scheible, and B. Webber, "Experiments at the University of Edinburgh for the TREC 2006 QA track," in *Proceedings of the 15th Text REtrieval Conference*, 2006.

[80] X. Qiu, B. Li, C. Shen, L. Wu, X. Huang, and Y. Zhou, "FDUQA on TREC 2007 QA track," in *Proceedings of the 16th Text REtrieval Conference*, 2007.

[81] J. Wiebe, E. Breck, C. Buckley, C. Cardie, P. Davis, B. Fraser, D. Litman, D. Pierce, E. Riloff, and T. Wilson, "NRRC summer workshop on multiple-perspective question answering, final report," Technical Report, Northeast Regional Research Center, Bedford, MA, 2002.

[82] V. Stoyanov, C. Cardie, and J. Wiebe, "Multi-perspective question answering using the OpQA corpus," in *Proceedings of the Human Language Technology Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, 2005.

[83] F. Li, Z. Zheng, Y. Tang, F. Bu, R. Ge, X. Zhu, X. Zhang, and M. Huang, "THU QUANTA at TAC 2008 QA and RTE track," in *Proceedings of the 1st Text Analysis Conference*, 2008.

[84] E. Voorhees and H. Dang, "Overview of the TREC 2005 question answering track," in *Proceedings of the 14th Text REtrieval Conference*, 2005.

第 14 章

Multilingual Natural Language Processing Applications: From Theory to Practice

提　炼

Vittorio Castelli, Radu Florian

14.1　概述

提炼是自然语言处理（NLP）中一个新兴的非传统分支，介于经典领域信息检索（IR）和问答系统（QA）之间。

与 IR 不同，提炼的目的是以检索结果集中的一个或者多个段落为依据，为查询提供答案，而不是根据用户查询来抽取相关的文档或者段落。提炼答案可能来自于段落中抽取的片段，也可能是合成的。提炼查询可能非常复杂，并且需要复杂的答案。例如，考虑如下示例：

Describe the reactions of <COUNTRY> to <EVENT>

这里 <EVENT>可以通过一个或多个自然语言句子指定。对此问题的答案可能是相当复杂的。因此，提炼的目标不仅要返回事实，而且还要根据复杂的查询确定复杂的答案。

正如我们不久将看到的，提炼不是会话语篇理解的代名词。提炼领域最新进展的主要推动力来自于 DARPA 全球自主语言开发计划（Global Autonomous Language Exploitation，GALE），它的任务就是提炼，根据直接或隐式的查询，以易于理解的形式，提供相关的、综合的信息……分析的数据应该是海量的多语言的语音和文本。本章详细介绍 GALE 对提炼任务的解释，并且讨论提炼系统的多模式和多语言的问题。在 GALE 中，提炼任务可以依靠适量的世界知识（比如，众所周知的白宫通常表示美国政府行政部门，而不是与同定义一致的特定建筑；然而，现任南卡罗来纳州长的政治派别不是可以用作提炼的世界知识）。另外，只有少量的推理是允许的，因此提炼系统不需要通过图灵（Turing）测试。

在 GALE 中，用户查询通过**模板**（template）指定。模板由以下几个部分组成：只有一个或多个**参数**（argument）的**问题**（question），外加可选的一个或多个参数的**等价项**（equivalent term）；能复述全部或部分查询的**相关术语**（related term）；以及**限制条件**（restriction），包括感兴趣的日期、文档日期、源端语言、源端的形式（音频或者文本）以及源端的特征（有结构的还是无结构的）。

本章所讨论的提炼方法，在很大程度上是语言独立的，因为提炼依赖于基本的 NLP 技术，比如提及检测，句法分析和语义角色标注，并建立在统计方法基础之上。这种整合的方法摒弃特定于语言的多数细节，如语料库的标记和用于统计方法的底层特征的开发，与此同时，保持高层的方法和架构不变。

本章从一个比较有目的性的例子开始来说明提炼的范围和为 NLP 带来的主要挑战，同时还讨论了两个概念：答案与查询的相关性（relevance）和不同答案的冗余度（redundancy），然后详细描述 Rosetta Consortium 针对 GALE 计划而开发的提炼系统。本章结尾则概述了用于提炼的各个方面和多语言提炼的几种方法。

14.2 示例

考虑以下查询：

WHAT CONNECTIONS ARE THERE BETWEEN [the Israeli pull-out of Gaza] (between 2005-05-01 and 2005-09-30) AND [the Gaza security situation]?

此示例的模板是：WHAT CONNECTIONS ARE THERE BETWEEN [event] AND [topic]。模板中有两个参数：参数一是<EVENT>，即 "*the Israeli pull-out of Gaza*"；参数二是<TOPIC>，即 "*the Gaza security situation*"。参数一限定在时间域内：与查询相关的是发生在 2005-09-01 至 2005-09-30 之间的事件 "*the Israeli pull-out of Gaza*"。而对参数二 "*the Gaza security situation*" 并没有限定时间域。

相关答案的示例如下：

1) Since then, Palestinians have continued to fire rocket from Gaza, and Israel has carried out periodic air strikes.

2) Israeli settlers and troops withdrew from Gaza last September, though the security situation for Gazans has deteriorated since then, with the Palestinian Authority unable or unwilling to confront the gunmen.

第一个答案中，"*then*" 解析为 "Israeli pull-out" 的结束，第二个答案，"*September*" 解析为 "September 2005"。

第一个答案是相关的，因为答案建立了事件与有关主题的两个事实之间的时间关系：1) 巴勒斯坦继续从加沙发射火箭（*Palestinian have continued to fire rocket from Gaza*）；2) 以色列开展定期的空袭（*Israel has carried out periodic air strikes*）。而第二个答案同样在事件和两个事实之间建立了时间关系，也是相关的：1) 安全局势已经恶化（*the security situation has deteriorated*）；2) 巴勒斯坦当局不能或者不愿意面对持枪者（*the Palestinian Authority is unable or unwilling to confront the gunmen*）。 476

不相关答案的示例如下：

1) Israeli Prime Minister Ariel Sharon previously pointed out that Egypt's role in the disengagement plan will be solely a security role.

2) Israel on Friday threatened to restrict travel and trade across Gaza's border if the Palestinians did not respond to Israeli security concerns within 48 hours at the recently reopened Rafah border crossing on Gaza's southern frontier with Egypt.

第一个答案描述了以色列撤出加沙（*Israeli pull-out from Gaza*）这一方面，但是没有反映加沙的安全局势（*security situation in Gaza*），以及两事件之间的关系信息。同样，第二个答案涉及加沙的安全局势（*security situation in Gaza*），但是与以色列撤出无关（*Israeli pull-out from Gaza*）。

现在考虑下面的答案：

Since the Israeli pullout, Palestinians have continued to fire rockets from Gaza.

这个答案显然是相关的，但是此答案没有为第一个相关答案增加信息。因此，当第一个相关答案已知时，此答案是冗余的。

14.3 相关性和冗余性

基于GALE计划对响应的定义来讨论相关性和冗余性。响应由一个主要片段、多个

支持片段和多个引用组成。

- （主）片段（snippet）由以下部分组成：1）来自文档的一段文本；2）这段文本的译文；3）总结；4）复述。为了讨论简便，定义一个由一个句子组成的文本片段。
- **支持片段**（supporting snippet）是一个至少给主片段中的某些相关信息提供额外支持的片断，并且不包含与主片段无关的信息。
- **引用**（citation）是生成主片段或者支持片段的文字摘录。

总的来说，为了回答查询，提炼系统必须识别包含与查询相关的文本的句子，将这些句子分为不同的集合，集合之间没有冗余，并且适当地报告具有代表性的句子的内容，这些句子能够刻画出集合中包含的信息。提炼系统还应该进一步报告那些能验证代表性句子的其他句子的内容。代表性句子的相关内容可以是摘录、总结或者复述，最终结果是主片段。与主片段相关联的集合有以下特点：集合中的每个句子至少包含与查询相关的某些信息，并且这些信息同样包含在该集合的代表性句子（该集合的主片段）中。同样，给定任
477 何一对答案集合，每一个对应的代表性句子至少包含一些与查询相关、但不包含在其他代表性句子中的信息。因此，给定集合的主片段，那么每个支持片段都是冗余的，但是主片段之间是非冗余的。在 GALE 计划中，只对主片段两两之间的冗余性做出要求，而不考虑找出一个主片段的集合的需求，这样集合中的每个元素都与其他元素是非冗余的。本章采用这个简化的假设。

支持片段为主片段描述的部分或者全部相关信息提供额外的证据。因为片段来自句子的某个部分，所以相同的句子可以为不同的主片段生成（不同的）支持片段，虽然不是对多个主片段进行提及。

根据对相关性和冗余性的一般理解，我们现在探索这些概念中一种可能的精确定义。考虑以下查询：

PROVIDE INFORMATION ON [Former Lebanese Prime Minister Rafik Hariri]

以及与它相关的句子：

A U. N. investigation into the truck bombing that killed Hariri and 20 others on Feb. 14 concluded in a preliminary report that the attack was the work of high-ranking Syrian and Lebanese intelligence officers.

此答案至少包含四个原子信息片段：Hariri 已死亡，是被谋杀的；死于汽车炸弹袭击；于 2 月 14 死亡（日期应该被系统解释为 2005/02/14）；还有叙利亚和黎巴嫩的高级情报人员与此次袭击有关的一些线索。GALE 中的原子信息片段称作“块”（nugget）。下面是相关的句子：

But Khaddam did not specifically accuse Assad of making or participating in the decision to assassinate Hariri.

此句与之前的句子至少有一个共同的块：Hariri 被暗杀。如果前一个句子整体作为一个主片段，那么后者可以作为其支持片段。接着考虑查询的其他答案：

Months before his assassination, the late Hariri, a self-made billionaire and once ally of the Syrian regime, had voiced strong opinions that Syria should stop interfering in Lebanese affairs.

上面句子至少包含四个块：1）Hariri 已故，被谋杀；2）Hariri 是白手起家的亿万富翁；3）Hariri 在某时间与叙利亚结盟；4）Hariri 被谋杀之前的数月，曾直言不讳地反对叙利亚在黎巴嫩的政治主张。这个句子可以作另一个主片段，因为此句并不与我们构造的第一个主片段共享所有信息块，因此只要其中每个主片段至少存在一个块不包含在另一个

主片段中，两个共享信息块的主片段（Harri 已亡，被谋杀的）就是可以接受的。

一个待解决的问题，如何将一个句子分割成块。GALE 的提炼标注指南建议依据句子 478
中的动词和其依附词 [1]，从概念上把句子分解为简单的子句。与查询相关的子句包含一个信息块，块可以根据捕获的信息分为不同的类别。GALE 中块的类别有：人物、地理政治实体（Geopolitical Entity，GPE）、组织机构、头衔、数值短语、命题块、时间短语、方位短语、修饰块和陈述。

除了命题块和修饰块，大多数块是不言而喻的，下面详细介绍这两种块。命题块主要围绕着片段的主要相关谓词构造，由谓词及其主要论元构成。修饰块是动词的一个表示因果、方式和其他修饰语的涵盖性范畴，但不包括时间和方位范畴。例如，一个包裹的收件人、目标、原因、目的和工具性短语。它们常表现为一个从句，描述原因、响应、解释，或者对另一个相关从句的影响。例如句子：

Making good on his main campaign pledge, Bolivia's President Evo Morales ordered troops to occupy the country's oil and natural gas fields on Monday and issued a decree giving the government majority control over the energy industry.

上面的句子是下面问题的相关答案：

LIST FACTS ABOUT EVENT: Bolivian President Evo Morales' takeover of gas fields.

信息块 "*Making good on his main campaign pledge*"（兑现他的主要竞选承诺）描述的是玻利维亚总统埃沃·莫拉莱斯为什么占领国家的石油和天然气地区（*Bolivia's President Evo Morales ordered troops to occupy the country's oil and natural gas fields*）的原因或理由。

现在可以根据块来定义冗余性。首先，如果块 n_1 和 n_2 传达相同的信息，则块 n_1 和 n_2 是等价的；比如 *Pierre Cartier* 买了希望之星 "*Pierre Cartier bought the Hope Diamond*" 与 *Cartier* 购买了希望之星 "*Cartier purchased the Hope Diamond*"（这里把 Cartier 解释为 Pierre Cartier）是等价的。

如果 S_1 和 S_2 为两片段，N_1 和 N_2 是与它们对应的块集合，如果 $N_2 \setminus N_1 = \emptyset$，那么对于给定的 S_1，S_2 是冗余的。

因此如果 S_1 和 S_2 是对应于相同查询的两个主片段，那么有 $N_1 \neq \emptyset$，$N_2 \neq \emptyset$，$N_2 \setminus N_1 \neq \emptyset$，$N_1 \setminus N_2 \neq \emptyset$，也就是说，两个主片段都至少包含一个块，且至少有一个块是另一片段没有的。如果 S_1 是一主片段，$\tilde{S}_1$ 是其中一个支持片段，对应的块集合是 $\tilde{N}_1$，那么 $\tilde{N}_1 \neq \emptyset$ 和 $\tilde{N}_1 \subseteq N_1$，也就是说，支持片段对应的块集合是主片段对应块集合的子集。

14.4 Rosetta Consortium 提炼系统

本节主要介绍一个实际的提炼系统，该系统是 GALE 计划的一部分，由 IBM 领导的 Rosetta 团队开发，该团队是 GALE 参与成员之一。系统是为了从庞大的、多语混合文本和语音记录组成的语料库中提炼问答而设计的，语料库语言涉及英语、阿拉伯语和汉语。 479
文本可以认为有两类：有结构（即新闻）和无结构（即网络博客）。类似地，音频也包括有结构（即演播室里的新闻记录）和无结构（即现场记者的新闻报道）两种。

提炼系统包括三个明确的步骤：准备文档，建立索引，回答查询。

14.4.1 文档和语料库准备

在文档准备阶段，对音频记录进行转录，并且把其他语言的文本和文字记录翻译成英语。英语和其他语言的文档通过 IBM 信息抽取系统 [2] 进行分析。信息抽取过程中对文

指执行词元化、过滤 HTML 标签、大小写还原（特别对于自动转换得到的文本）、进行句法分析、提及检测和指代消解。利用抽取到的信息，检测出提及对之间的关系，并且将语义角色标记与句法树节点关联起来。

词元化、词性标注、句法分析、提及检测和语义角色标注都使用最大熵模型 [3]；序列化解码由 Viterbi 算法 [4] 完成。提及检测引擎识别命名实体提及和 17 种事件类型的锚点。识别 36 种实体类型中的命名提及，名词性提及和代词提及，并标记它们的跨度和中心词的范围。事件的锚点为动词、名词性动词或者名词。

提及关系基于 Bell 树 [5] 的算法进行消解，从左到右对提及进行分析。通过树结构对提及与实体之间的连接过程建模。第一个提及对应第一个实体，相应地，树的根节点就产生了。当第二个提及出现时，可以连接到第一个实体，也可以连到一个新的实体。通过在树结构中生成一条边和一个节点来表示每种可能的操作。每个新提及都通过重复的过程完成消解，最终树的叶子节点表示文档中检测到的提及的所有可能类别；叶子节点的数目称作 Bell 数 [6]。当碰到新实体时需要扩展节点，其指算法利用二值最大熵分类器计算此提及连接到一个实体的概率；为节点中现有的每个实体创建一个分支，增加新的分支就意味着一个新的提及。每个分支都被赋予概率值，表明新的提及与相关实体之间的连接概率，由 MaxEnt 分类器计算得出，并保证它们的和为 1。每个实体都有一个规范提及，典型的是文档中的最长名称提及。

关系检测是建立在描述的其他阶段之上的预处理过程。关系引擎确定 36 种关系提及，这些关系提及是实体提及之间的关系或者实体提及与事件锚点之间的关系，由各句子中的文本显式支持。这些关系提及本质上与自动内容抽取评测 [7]（Automatic Content Extraction，ACE）中的概念类似，可认为是它的一个适当的超集。关系提及有多种属性，包括关系类型（包含 ACE 关系中的类型与子类型）、关系中提及的顺序（非对称关系中区分提及的角色）、时态（现在时、过去时、将来时，或者表明无时间限制的不定时）、特异性（决定此关系是否存在于特定或者普通实体之间）。关系通过级联的最大熵模型抽取，是 Kambhatla [8] 描述的模型的扩展。级联的第一步建立关系提及的存在，下一步抽取前面描述的属性。

与实体提及的情况一样，关系提及自动连接到文档级别的关系上；特别地，相同实体对的提及之间的同类型（且同顺序，如果关系是对称的）的关系连接到相同的文档级关系。没有统计模型能够用来完成这项任务，相反，依据共指链来确定性地执行该任务。

ACE 式的关系不足以描述文本建立的实体之间的所有连接。预处理过程中，实体之间额外的连接被抽取出，而这些连接并不是文档显式表明的，也不在一个句子的内部描述。这些扩展关系由一个模块识别，这个模块最初是为 2009 文本分析会议的知识库填充任务（TAC KBP）[9] 的插槽填充任务而开发的。参与插槽填充任务的人员被要求分析一个大型文本语料库，抽取出人物、组织机构、地理政治实体的特定属性（在评估时指定）。所需的属性类似于维基百科的信息框插槽 [10]，它们的值不仅仅局限于文本中单个句子的显式支持。用一个简单例子可以说明家庭关系 "*Bob and his mother Mary went to the mall. His brother John remained at home*"，这段信息确定关系 "*Mary is John's mother*"（假设这里的兄弟是指亲兄弟），但是此关系并没有在一个单独的句子中显式地体现，因此 ACE 式的关系不能捕捉这类事实。TAC KBP 任务解决这类问题的方法是：建立在关系检测和指代模型上的基于规则的系统。

这种槽系统依赖于三类广义规则：共指规则、关系规则、导出关系规则（表示推导出来的关系）。共指规则依据共指链，考虑下面规则：

488

IF (X IS-A Person Entity) AND (Y IS-AN Occupation) AND
(Y ISCOREFERENT WITH X) THEN (X PER, TITLE=Y)

在下面摘录中：

Barack Obama concluded his visit to China on Wednesday. The President expressed hope for further Sino-US cooperation.

上述规则发现两个人物提及：名字（*Barack Obama*）和名词性职务（*President*）；共指系统将它们连接到文档中的同一实体。因此建立一条规则，即 TAC-KPB 插槽中 "Barack Obama" 的 "PER, TITLE" 值为 "President"。

关系规则在一个或多个关系和共指链的基础上进行推导。比如，在先前的例子中得到的关系规则是 *Mary* 是 *John* 的妈妈 "*Mary is John's mother*"。

导出关系规则是关系规则的扩展，可基于抽取其他关系和导出关系规则的槽，以及 ACE 式的关系和共指链进行推导。例如如下的导出关系规则：

IF (X IS-A Person) AND (X ISPARTOFMANY G) AND (G HASTITLE T) THEN
(X HASTITLE T)

当上述规则应用于句子 "*Fifteen Senators supported the bill, including John McCain*"，则抽取出信息 *John McCain* 具有 "*Senators*" 头衔。这些规则可应用到单个文档，抽取的槽和值可以看作问答系统中 ACE 式的关系。

为了建立索引和回答查询准备语料库的最后一步要处理跨文档共指（Cross-Document Coreference，XDC），为语料库的每个实体赋予唯一的 ID。因此，实体 "*Barack Obama*" 与其出现的每一个文档中的 "*44rd U.S. President*" 有相同的 ID，而实体 "*George Bush, NANSCAR driver*" 与 "*George W. Bush, 43rd U.S. President*" 有不同的 ID。跨文档共指依赖于为 2009 TAC-KPB 实体链接任务而建立的系统，该任务为接受任务的参与者提供数据库（称为知识库，每个实体都有对应的文档描述）。查询的形式是一个实体名称和一篇提供消歧信息的文档。如果实体在知识库中，则答案是基于知识的实体 ID，否则为空。

实体链接系统可以扩展成一个 XDC 算法。首先通过加强知识库进行扩展；Rosetta 的提交系统中，把 dbpedia[⊖] 数据库与 TAC-KBP 知识库合并。第二种扩展是选择用来替代消歧文档的文本；选择的文本由包含文档级实体提及的句子集构成。有了这两项改变，XDC 可以被转换成为每个文档级实体分配唯一知识库 ID 的问题。

这导致了第二种扩展：为不在知识库中的实体分配 ID。为了发现扩展知识库中的实体需要以下两个阶段的处理：第一阶段包括基于字符三元组的快速名字匹配。这种简单方法对拼写变体和印刷错误具有鲁棒性，因为绝大多数情况下正确的实体在前 50 个命中中。这种方法也是高效的，因为它很容易在标准的搜索引擎上实现，如开源的 Lucene。前 50 个实体由第二阶段深入分析，其中结合了复杂的名称相似度分值。这种 SoftTFIDF 相似度来自 SecondString 包 [11]，还结合了上下文匹配分值。上下文匹配分值基于余弦相似度计算，衡量文档级别的实体与知识库中候选者上下文中的非停用词之间的重叠度。后者从候选者的维基百科信息框抽取。如果快速匹配和精细匹配的分值超过在保留数据集上学习到的阈值，那么匹配成功。如果匹配失败，则意味着实体不在知识库中，且没有关联的知识库 ID，那么采取回退策略。回退策略的做法是为快速匹配或精匹配成功的文档级实体赋予 XDC ID，并为剩余的实体赋予唯一的 ID。XDC ID 是最相似的知识库实体（即匹配

481

482

⊖ The dbpedia is available at http://dbpedia.org。

成功且得分最高的实体）的标识符，但加上一个可区分的前缀。这种方法的基本原理是：两个表示现实世界中同一实体的文档级实体很有可能指向同一个“最相似的”知识库实体。

14.4.2　索引

文档使用开源搜索引擎 Lucene 建立索引。索引允许对文档文本的词袋式搜索和命名实体的查询。为了支持后者，除了名词性提及和代词性提及，对其余实体提及都建立两种索引：用于准确匹配的完整词形索引和用于 n 元匹配的词元索引。

14.4.3　查询回答

查询回答提炼系统以 GALE 格式查询作为输入，如引言所述，返回主片段列表和相关的支持片段、引用，并按与查询的相关度降序排序。系统架构由五部分组成：查询预处理、文档检索、片段过滤、片段处理、规划。

1. 查询预处理

查询预处理阶段完成对查询组件的信息抽取：参数、相关术语和等价术语。对它们进行词元化、句法分析和语义角色标注，检测提及并且使其指向相关实体。XDC 系统的情为检测到的实体赋予跨文档 ID。对于 PERSON、ORGANIZATION、LOCATION、GPE 和 COUNTRY 等类型的参数，查询预处理阶段识别主实体，检测辅助指代，并确定它们与主实体的关系。对于 EVENT、TOPIC 或 CRIME 等参数类型，预处理阶段则另外计算参数、相关术语和等价术语的依存树。

一个查询实例：

DESCRIBE INVOLVEMENT OF [Russia] IN [attempts to freeze Iran's nuclear program]

有两个参数，第一个是“COUNTRY”类型，第二个是“EVENT”类型；它们都没有相关术语和等价术语。XDC 系统为俄国“*Russia*”赋予合适的跨文档 ID，对 EVENT 的定义进行信息抽取：文本词元化、句法分析、提及检测（*Iran*），且 XDC 系统为 *Iran* 赋予一个 XDC ID。

2. 文档检索

183 从提炼查询组件抽取的信息用来搜索 Lucene 索引。Lucene 查询由非终结符和所有实体组成，其中实体从提炼查询的组件中抽取。另外，把预处理阶段为命名实体赋予的所有 XDC ID 当作参数，用于搜索引擎查询。搜索引擎查询的结果是文档的集合，每个文档都赋予分值。搜索引擎返回最大指定数目的文档。因为所有文档都作为后续阶段的输入，所以可通过权衡整个系统的召回率和期望的响应时间来确定文档的最大数目。可以采用不同的标准方法选择此参数，比如利用保留集合报合数值。另外，根据文档得分自适应地选择文档数目的方法同样是有效的。在 Rosetta 提炼系统中，采用的是第一种方案，检索到的文档数目固定为 500。

在预处理的例子中，查询要求检索包含“Russia”和“Iran”两个提及的文档，也包括含有非停用词“*attempts*”、“*freeze*”、“*unclear*”和“*program*”的较大子集的文档。

3. 片段过滤

提交到搜索引擎的查询返回大量文档的集合，通常都与提炼查询相关。下一步要确定文档的相关部分并生成非冗余片段。一些类型的文档，特别是长的新闻组日志和转录新闻，通常文档中的大部分与查询无关；同时它们的大小使得对其进行详细的整体分析是不

现实的。Rosetta 提炼系统的解决方案是构建高召回率的过滤阶段，筛选出相关性高的句子，同时丢弃大量无关句子。此阶段基于启发式方法或者少量特征集的统计模型。例如对于查询模板 "DESCRIBE RELATION BETWEEN <PERSON1> AND <PERSON2>"。当两者的提及出现在同一句子中或者出现在相邻的句子中时，一条简单的启发式规则就可以起作用，达到在丢弃大量无关句子的同时保证较高的召回率的目标。对于其他模板，可能用到更复杂的统计方法。这些方法类似于片段处理阶段用到的方法，但是更简单些，接下来几节详细讨论这些方法。在任何适当的时候，系统记录那些通过片段过滤句子的分值，这些分值可以用于主片段预处理的回退策略。

对于上述实例，片段过滤器会检索出所有包含 Russia 或者代表 Russia 提及的片段，因为此阶段目的是高召回率，所以很多不相关的句子也能通过，例如：

Russia frustration at Iran's refusal to send uranium to Russia and France for processing into fuel hints at the possibility that Moscow may be open to a new UN Security Council resolution.

Russian President Dmitry Medvedev stated that there is agreement over sanctions for Iran but that this is still not the desired path.

Moscow and Tehran announced that Russia will build a nuclear reactor in Bushehr. [486]

4. 片段处理

片段处理阶段有两个目的，一是为通过片段过滤阶段的句子赋予相关度，另一个是确定信息，这些信息在规划阶段用来构成主片段、支持片段和引用。根据模板，如果句子描述了参数的属性（例如模板形式："PROVIDE INFORMATION ON <ORGANIZATION>"），涉及参数的事件或动作（例如模板 PRODUCE A BIOGRAPHY OF <PERSON>），或者参数之间潜在的复杂的相互作用（例如 DESCRIBE THE INVOLVEMENT OF <COUNTRY> IN <EVENT>），那么句子与查询是相关的。如果参数相当复杂，那么判定一个句子是否与某个参数相关是一项具有挑战性的任务（例如对于参数<EVENT>和<TOPIC>）。尽管可以手工创建模板依赖、基于规则的系统，即系统将为句子分配相关性分值，但是这种方式既不便宜，又不具备可伸缩性。

Rosetta 提炼系统的片段处理阶段依赖于基于模板的层次统计打分模型，模型利用手工标注的数据训练。由于训练层次模型并重新利用标注数据训练其他模型，这种方法具备可伸缩性。实体参数的参数模型是模型层次结构的基础，目的是检测句子是否包含或者描述人物、组织机构、地点、国家、地理政治实体等查询参数的主要实体。用来学习参数模型的学习算法是投票感知机[12]，在小规模训练集上，稍微优于最大熵模型。训练数据格式为三元组（QUERY，SENTENCE，LABEL），其中 QUERY 是简单问题，形式如 DOES THE SENTENCE CONTAIN <ARGUMENT>，这里 SENTENCE 一个句子，并附加指向句子来源文档的指针；LABEL 是标注人员依据参数标注指南，手工指派给句子的二值数据。标注指南指出了各种不同类型的参数在句子中被提及的诸多方式。

三元组（QUERY，SENTENCE，LABEL）通过特征抽取转换成特征向量集。可以依据输入空间对特征抽取器进行归类：QUERY-ARGUMENT，SENTENCE，QUERY-ARGUMENT×SENTENCE，QUERY-ARGUMENT×DOCUMENT，QUERY-ARGUMENT×SENTENCE×DOCUMENT。

例如，如果一个<PERSON>类型的参数包含的标题具有输入空间 QUERY-ARGUMENT，那么特征抽取器开始起作用，并检测一个句子是否包含输入空间为 SENTENCE 的片段。更为复杂的特征抽取器比较句子中感兴趣的提及与主参数提及的 XDC ID，这是

输入空间为 QUERY-ARGUMENT × SENTENCE 的一个实例。

众多特征抽取器，比如匹配 XDC ID 的特征抽取器、近似匹配句子和参数中提及的文本的特征抽取器，已经广泛用于参数模型中。其他特征抽取器根据特定参数类型构建：例如，因为国家经常用他们的行政官员提及（Dmitry Medvedev 是俄罗斯的有效代理人），或者用隐喻提及（白宫常常指美国，伦敦常常指英国），当国家模型（country model）检测到句子中国家的指代或者表示国家的隐喻时，即启用其特征函数。

[485] 检测复杂参数 EVENT 和 TOPIC 的层次结构类型的模型：一些特征来自其他参数模型的输出，通过含有人物、组织机构、国家、GPE，或者地点的提及的一个或多个句子或者短语，可以指定 EVENT 和 TOPIC。例如，“*The collapse of Lehman Brothers*”包含一个组织提及，而“*AIG sells Alico unit to MetLife*”包含二个组织机构提及，它们之间具有复杂的关系。参数中的提及在查询预处理过程中识别，如果合适的参数模型在句子中发现参数匹配，则可用于特征抽取器。例如句子“*On 9/15 the firm filed for Chapter 11 bankruptcy protection*”，这里公司 *Lehman Brothers* 是和第一个查询相关的，事件模型包含特征抽取器，用于从查询参数中选择“*Lehman Brothers*”，并利用 ORGANIZATION 模型将它匹配到名词性提及“*firm*”，并作为结果。

为了在复杂查询参数的提及中捕获内部依赖以及相关的非停用词，模型需要根据高级特征，如匹配句子中和参数中的 ACE 式关系的特征，以及当前句子和查询参数中的提及与非停用词之间匹配依赖结构的特征，参见[13]。

Rosetta 提炼系统使用为每个模板训练的统计模型为句子打分。这些模型根据双层方式进行特征抽取。第一层，找出参数模型，这些模型检测到句子中有参数存在时，则生成适当的特征并捕捉句子词汇、句法、语义方面的特征。这些特征抽取器会留下“痕迹”，也就是说能够识别触发特征抽取的句子的特定部分。第二层由抽取器组成。当识别出“痕迹”（由其他抽取器分析标识）上的句法的、语义的属性时，这些抽取器启用。第二层抽取器自己也能留下“痕迹”，因此能有层次地结合。

当查询参数是复杂类型时，比如 EVENT 和 TOPIC，决定句子与查询的相关度是相当有挑战的。在基于实体的查询中，句子中没有实体提及则意味着这个句子是不相关的，与之不同的是，关于事件和主题的信息通常体现在不包含参数描述的句子里。例如，关于特定主题的新闻文章通常以涉及主题的内容开始，然后提供未显式地提及主体的信息。为了克服这个困难，Rosetta 提炼系统的模板模型从已评分的句子周围句子的一定窗口的上下文句子中提取特征，选择的特征抽取器应用于这些窗口的句子中，统计哪些起作用，并分析产生特征，我们称为上下文特征（context feature）。

对模板模型分析的句子打分，并记录触发特征的“痕迹”，将具有能表明相关性的已评分的句子用作规划阶段的输入，剩余的丢弃。如果相关句子的个数低于用户指定的阈值，则回退策略启动，再次执行片段过滤，并把在过滤阶段产生的得分最高的那些句子添
[486] 加到结果集中。

参考所举的例子，片段处理是为了能选择以下相关句子而设计的：

Russia frustration at Iran's refusal to send uranium to Russia and France for processing into fuel hints at the possibility that Moscow may be open to a new UN Security Council resolution.

Russian President Dmitry Medvedev stated that there is agreement over sanctions for Iran but that this is still not the desired path.

并且丢弃以下句子：

Moscow and Tehran announced that Russia will build a nuclear reactor in Bushehr.

5. 规划

片段处理过程会生成相关句子集合，集合中的句子是潜在冗余的。规划阶段分析这些句子，确定非冗余片段，支持片段并支持引用。规划器依赖句子的分值和片段处理器产生的特征“痕迹”。句子按分数降序分析。分数最高的句子自动成为主片段的引用。选择包含所有特征“痕迹”的句子跨度，并在句法分析树中找出离树根最远的、能覆盖这个跨度的构成部分。然后找出句子中与此构成部分对应的文本。这些文本就构成了主片段的引文。至少包含一个未出现在任何其他主片段中的特征“痕迹”的句子构成一个新的主片段。支持片段的构建方法与主片段相同。对上述构建策略的加强包括：设置一个小于 1 的阈值，得分比阈值与最好句子的得分的乘积高的句子才能作为主片段和支持片段的候选，而其他的句子只能作为支持片段的候选。系统强制将片段处理器认为具有高可信度的相关句子选为主片段。

考虑之前例句，规划阶段对

Moscow might be open to a new UN Security Council Resolution

和

Russian President Dmitry Medvedev stated that there is agreement over sanctions for Iran but that this is still not the desired path

进行标记，以作为包含相关信息的句子成分。因为激发而检索片段的特征具有不同的“痕迹”，因此规划器认为这两个句子可以分别作为主片段。 487

14.5 其他提炼方法

详细介绍特定提炼系统之后，本节简单回顾文献中描述的一些提炼方法。特别地，讨论一些系统的架构和用于相关性检测与降低冗余的方法。结尾部分简要介绍基于语音数据和转录数据的多模态方法和依赖源语言文本及其翻译文本的多语言方法。并不关心其他的文档检索方法，因为这对提炼系统来说是唯一的，所以本节并未涉及文档检索。

14.5.1 系统架构

典型地，提炼系统遵循如下通用架构：首先利用信息检索技术生成抽取文档的一般框架，然后分析文档中的句子 [14, 15, 16]。Lin [17] 提出另外一种方法，文章中讨论了信息抽取在复杂查询问答中的作用。该文章讨论了两个主要问题：IR 技术是否能单独用来识别句子，这种句子与比事实性问题更复杂的问题相关；IR 技术是否可以用于降低或者消除抽取出的句子之间的冗余。作者得出结论：不能仅使用 IR 技术抽取与复杂问题相关的句子；相反，结果也指出 IR 技术具备处理冗余问题的能力。

14.5.2 相关度

Lin [17] 的研究是提炼系统的前身，文章讨论了回答关系询问（relationship question）的问题。在 2005 年，作为 TREC 问题回答跟踪 [18] 的一部分，关系询问被正式提出。关系询问提出如下问题：X 与 Y 如何相互作用？或者 X 如何影响 Y？这里 X 和 Y 可以是实体，事件或主题。例如用户可能问“凡尔赛条约对国际联盟的建立有什么影响”。Lin 探索传统信息检索技术，特别是句子检索，对回答关系询问到底有多大影响。作者构造了一些简单的特征：段落匹配分值，基于查询和候选句子出现的特有术语的逆向文档频率值（IDF 值）计算；术语 IDF 和召回率；句子长度。通过线性回归模型组合这些特征，得到相关度度量。上述简单的模型优于仅依赖 IR 分值的基准系统，特别是当时长度限定

为 1000 个字符或者更少时。

Levit 等人 [15] 描述了 IXIR 提炼系统中基于统计的相关性检测方法。IXIR 的高层结构与 Rosetta 系统类似，显著的不同点在于将片段过滤与片段处理融合。与 Rosetta 系统一样，查询相关性由特定模板统计系统建立，该系统依赖于广泛的特征。这种方法的核心如下：
488 IXIR 系统为每个句子构建包含这些特征的图示，称为图表（chart）。图表是基于句子的词构建的图，图中的边用与其相连的节点的特征标注。最简单的词汇特征是：B 直接跟在 A 之后，其中 A 与 B 为句子中的单词。类似地，句法关系、依存关系、语义角色标注关系等在图中也通过边表示。当表示命名实体参数时，IXIR 试图通过各种已知策略检测所说实体是否在句子中，比如近似匹配检测、拼写变形检测、同义词词典、地名表、修饰符字符测器和 WordNet [19]。如果匹配成功，系统就为这个句子的图表增加一层结构，描述匹配在哪以及如何发生。一旦图表构建完成，就可用于为检测参数是否出现在句子中的统计分类器计算特征。这些特征是传统的词 n 元组的扩展，词 n 元组可以通过简单图表中有 $n-1$ 条边的路径生成。这个图表仅包含形如 A 后面跟随 B 之类的词汇特征。IXIR 的特征是 n 元组，由图中具有 $n-1$ 条边的路径生成，而不管边是由哪些特征产生的。复杂的参数（例如事件）的处理有点不同，IXIR 计算参数描述的图表，并从参数图表和句子图表中抽取特征。

Kamangar [20] 和 Kamangar 等人 [21] 主张使用无监督学习方法进行句子抽取。首先，确定少量可能相关的句子集合以及可能不相关的句子集合，然后提出二种基于查询参数的词干化的非停用词的方法。第一种方法是通过词频（TF）选择，计算候选句子中非停用词的词频，同时计算平均值。保留词频高于平均值的词，标记包含所有保留词的句子为正例句子，不包含任何保留词的句子为反例句子；第二种方法基于 TF-IDF 选择，包括保留查询参数中 TF-IDF 值足够大的单词，同样的方法用来标记正例句子和反例句子，通过独立的训练数据学习 TF-IDF 阈值；第三种方法认为所有查询参数中的非停用词同等重要，将至少包括部分上述非停用词的句子标记为正例，将不包括任何一个的句子标记为反例句子。最初的句子集用于迭代的自训练算法：从自动产生的正例句子和负例句子训练出的分类器用于分类不在训练集中的候选句子。分类器必须给出句子的分值或者后验概率估计，得出的结果用来选择其他的正例句子（那些得分大于从训练集学习到的阈值的句子）和反例句子（得分小于从训练集中学习到的另一阈值的句子）。当没有新的正例或反例句子产生或达到最大迭代次数时，迭代过程终止。

14.5.3　冗余

降低冗余并不只存在于提炼问题中，TREC 的新奇跟踪（novelty track）[22] 很早就涉及这个问题。Lin [17] 为将包含在增量式构造的答案集 A 中的新候选 c 定义了效用函数：

489
$$\text{Utility}(c)=\text{Relevance}(c)-\lambda \max_{s\in A} sim(s,c)$$

其中 Relevance 的计算如先前描述，sim 为相似度函数，比如余弦相似度，λ 为需要调节的参数。Lin 比较了三种度量相关度的方法：第一种利用求最大数运算的回归分数计算相关度，另两种是使用 *max* 和 *average* 作为综合函数，计算基准系统的相关度分数。利用 POURPER [23] 分值作为评测度量，对 λ 进行调参。实验发现以检测冗余为目的，*max* 操作优于 *average* 操作。

14.5.4　多模态提炼

在 GALE 中，评测是在包含几十万文档的多模态、多语言大语料库上进行的。语料库的一部分是录音材料，处理基于录音材料提问的基本方法是：首先自动把这些录音材料

转录成文本，将中文和阿拉伯语的转录文本翻译成英文文本，然后在这些英文文档上运行标准的提炼系统。与纯文本（如新闻）提炼相比，转录和翻译的错误会大大降低系统的性能。Yaman 等人 [24] 分析了转录错误为提炼带来的影响，报告称系统性能降低了 35%。他们提出在片段处理阶段使用锚点语音识别（Anchored Speech Recognition，ASR）的方法来解决这个难点。他们假设，存在片段过滤阶段，该阶段可以确定候选片段并过滤明显不相关的片段。包含查询答案的片段可以分成两种：一种是片段中答案和问题的措辞类似；另一种是两者措辞不同。文章描述了分析前一种情况的方法，并声称可以扩展该方法来处理后一种情况。给定查询集，通过现有的语言模型从查询问题中识别词短语。对每一个问题，建立一个匹配问题中词序列的词网络，同时允许周围有其他词，但不能与问题中的词交叉。结合来自偏置语言模型的偏置词格和为特定问题建立的词网络，强制解码器仅接受包含查询的词网络的路径。然后为结果路径重新打分，如果无返回结果，则采用适当的纠正措施，包括降低识别的约束。当答案包含问题的准确措辞且无介入词时，降低约束后上述方法的效果会更好。作者实验表明，该方法能帮助提炼系统修复 30%转录带来错误，并且正确答案相关词错误率降低了 37%。

14.5.5 跨语言提炼

跨语言信息检索是最近探索的领域 [25]。检索其他语言形式书写的文档语料库，给出英语形式的答案，这一任务向传统的信息检索系统提出了新的挑战。解决此问题最简单的两个方案是：1）搜索其他语言的语料库，然后翻译检索结果；2）首先自动翻译语料库，然后在翻译的结果上进行信息检索。前者局限于有限的语言资源，这些资源用于训练和构建 IR 或提炼系统的基本部件。后者局限于当前机器翻译的水平：自动翻译的文档包 490
含许多错误、不符合语法的句子、精炼的音译专有名词和其他错误，它们都会影响信息的抽取。混合使用这两种方法可能会更有效。McCarley [26] 证明了这一点，他对比了跨语言信息检索的三种方法：翻译查询、翻译语料库以及赋予文档前两种方法所得结果的数学均值的混合方法。在各种数据集上，混合方法的结果都比其他两种方法好。令人惊讶的是，甚至当把查询人工翻译成语料库语言时，混合方法也是最好的。

Parton 等人 [27] 主要讨论如何为提炼设计跨语言信息检索。作者的方法称作**跨语言信息检索**（Translingual Information Retrieval，TIR），在检索索引如何建立、查询如何执行方面，与 McCarley 的方法不同。McCarley 为原始文档及其翻译文档分别建立索引，而 Parton 等人提出为每个文档及其翻译文档建立单一索引。以英文表达的提炼查询被翻译出来，并基于查询的原始文本和翻译文本构建 IR 查询，并在跨语言的索引上检索。最后，作者提出基于原始查询与翻译查询纠正潜在翻译错误的方法。

Singla 和 Hakkani-Tür [15] 讨论了处理汉语语料库和阿拉伯语料库时跨语言片段的问题。作者在英语语料库上建立一个统计片段处理模型，正如 Hakkani-Tür 和 Tür [28] 所描述，同时在源语言端建立类似模型。提出两种方法融合它们：后验概率插值和层叠。在后验概率插值方法中，通过源语言模型为候选句子评分，并使用英语语言模型为候选句子的翻译评分；最后对这两个分数进行凸插值。层叠两个模型的方法利用由模型得出的概率估计作为另一个模型的输入特征。作为模型联合的一种替换方法，作者从源语言句子及其翻译中抽取特征，并将其作为单个联合片段处理模型的输入。文章显示了对于英语和阿拉伯语的查询，凸模型插值似乎优于其他方法，也优于源语言片段处理方法。

跨语言提炼是很有前途的领域，但仍处于初级阶段。当前的成果是单语言技术的逻辑

扩展，或者说仅构成了初步的探索：例如 Singla 和 Hakkani-Tür [14] 只根据简单的词汇特征（n 元组）生成结果。然而，随着可用的语言学资源日益增多，这些资源可以用来构建或者提高基本的信息抽取组件，这些都清楚地表明不久的将来可能会取得实质性的进展。

14.6　评测和指标

提炼系统的评测比问答系统、信息检索系统的评测更复杂。传统的评测指标：精确度、召回率和 F 值，它们都是信息检索系统性能的评估指标。大量的文献讨论了如何评价
491 信息检索系统（参见 Vorhees [29]）。

增加评测提炼结果难度的因素主要有三个：查询的复杂度，答案的格式和需要在单个评测指标中融合系统输出的多个方面。即使系统是基于模板的，但提炼查询寻找语义复杂的查询的答案，比如涉及事件。决定一段信息是否与查询有关，可能取决于如何解释这个查询，因此结果往往是很主观的。这种原因显然在冗余检测中也存在：给定相同查询的两段回答信息，检测冗余即意味着识别出只出现在其中某个回答中的信息片（块），并决定这些块是否与查询相关。正如之前讲到的，提炼查询的答案是复杂的：包含一个主片段（从文档或者其复述中提取的文本片段）、额外的包含部分或全部主片段信息的支持片段，一个或者多个引用（搜索语料库文档中的真实摘录）。这时会出现一个问题：如何解释答案中不同组件的错误。例如，返回不相关主片段的错误比返回相关片段的不相关引用的错误更严重。此外，还会出现一些其他情况：例如考虑产生相同主片段和相同的引用，但不同的支持片段的两个系统：是否有简单的方法评价哪个答案更好？最后的难题是，如何构建一个能够捕捉答案的相关性，不同主片段的冗余性和返回结果完备性的评价指标。

GALE 计划的评价指标

GALE 计划研究这个问题已经多年了，下面描述为 Year 4 Go/No Go 评测提出的指标[30]。GALE 指标的目的是比较提炼系统的性能与利用最先进的搜索技术的人类分析家的性能。特别是，对每个查询，给分析家 30～60 分钟的时间，要求给出与提炼系统格式相同的答案（主片段、支持片段和引用）。对来自分析家和系统的结果进行人工分析，如下所示。

1. 相关度分析

首先，最少通过两名人工评判员将每一个主片段标记为相关，部分相关或者不相关，将包含答案的相关和部分相关的片段的实际部分标记出来，标记出来的部分自动标记为块，其余部分作为上下文。人工评判员对自动划块的结果复审并纠正。不相关的片段同样划块以生成错误的块，这对与精确度相关的数量计算有影响。每个评判员赋予块一个相关分数：1 表示完全相关块；0.8 表示部分相关块；0 表示非相关块。对不同评判员给出的分数求平均，得到块最终的相关分数。

2. 冗余检测

492 人工评判员分析片段的冗余性，对每一对主片段，人工评判员比较对应块，并识别包含相同语义信息的块对。两个主片段 A 和 B，如果 A 中至少存在一个在语义上与 B 中的块不等价的块，并且 B 中也至少存在一个在语义上与 A 中的块不等价的块，那么认为 A 和 B 是非冗余的。如果 A 中的每一个块都在 B 中存在语义相等的块，反之亦然，那么 A 与 B 等价。最后，如 A 中的每一个块都在 B 中存在语义相等的块，那么 A 相对于 B 是冗余的，但反之不成立。

3. 引用检查

人工评判员评定是否每一个引用全部支持与它相对应（相关且非冗余）的主片段。如

果引用不能完全支持相应的片段，那么为该引用给予惩罚。人工评判员给与支持片段相关的引用打分时方法稍有不同：如果引用对应的支持片段与主片段有至少一个共同块，并且如果引用完全支持片段，那么认为引用是正确的。

4. 主要任务指标

基于上述判断产生两个主要任务指标，信息内容指标（information-content metric）和文档支持指标（document-support metric），结合起来形成一个性能评价分数。我们简单介绍一下这些指标，有兴趣的读者可以参考引用中的官方评价文档[30]。

信息内容指标衡量块级别的性能。块 i 的相关分数为 $R(i)$，冗余分数为 $D(i)$；相关分数由先前描述计算得出，而冗余分数为 0 或者 1。N_j 是提炼器根据查询生成的块的个数，精确度的定义如下：

$$P_j = \frac{\sum_{i=1}^{N_j} R(i)D(i)}{N_j}$$

召回率的定义如下：

$$R_j = \frac{\sum_{i=1}^{N_j} R(i)D(i)}{M_j}$$

假设 M_j（语料库中相关片段的总数）是已知的。因为人工为每个查询分析成千上万或者百万个文档是不可行的，所以召回率计算如下：

$$R_j = \frac{\sum_{i=1}^{N_j} R(i)D(i)}{\widetilde{M}_j}$$

$\widetilde{M}_j$ 为 M_j 的最大似然估计。对应的 F 值表示为 F^I。

文档支持指标衡量与非冗余片段相对应的有效引用的个数。如果一个引用不与其他引用共享片段，那么该引用是有效的。如果引用完全支持相应的片段，且片段是主片段或者与主片段至少存在一个共享块的支持片段，那么这个有效的引用是正确的。如果 R_j 是由提炼器返回的正确引用的个数，V_j 是同一个提炼器返回的有效引用的个数，W_j 是语料库中查询的有效引用的全部个数，那么文档支持的精确度和召回率的定义如下： 493

$$P_j^D = \frac{R_j}{V_j}$$

和

$$R_j^D = \frac{R_j}{\widetilde{W}_j}$$

其中 $\widetilde{W}_j$ 为 W_j 的估计（实际中为所有的提炼系统和分析人员返回正确引用的总数）。对应的 F 值表示为 F^D。

正式指标定义为文档 F 值的平方根调整信息的召回率：

$$R^{I'} = \sqrt{F^D} R^I$$

然后计算调整后的信息召回率和初始信息精确率的调和平均值：

$$F^{I'} = \frac{2P^I R^{I'}}{P^I + R^{I'}}$$

5. 额外指标

GALE year-4 评测也基于其他指标，这些指标捕获提炼系统特定方面的性能。另外，官方的评测文档描述了多种计算信息召回率的其他方法。这些方法的讨论超出了本章的范

时，有兴趣的读者可以参考正式评测计划文档[30]。

6. 备注

多年来，GALE计划提出的指标的缺点是需要大量的人工劳力，人工评判员必须分析提炼系统输出的每一个答案，还要比较不同的答案来检测冗余。因此，评测一个完整的提炼系统是耗时而昂贵的，对提炼系统的发展是一个潜在的限制。现在存在的一个悬而未决的问题是，如何构建一个耗费较少的指标来衡量提炼结果的质量。特别地，基于提炼系统
494 各个组件的评价指标，为整个提炼系统建立有意义的性能上界是有可能的。

14.7 总结

提炼是自然语言处理中相对较新的领域，弥补了信息检索与问答系统之间的缺口。DARPA GALE计划有力地推动了该领域的进步，并为开发和评测多语言和跨语言提炼系统提供了框架。

当前的提炼方法融合了信息抽取、QA技术和新的统计方法，这些统计方法允许系统处理复杂的查询，比如那些涉及特定事件的查询。在现存系统中，查询利用有一个或者更多的参数的模板指定。结果是从语料库中检索的摘录段落或复述段落，并将它们分成无冗余的组。

提炼面临的两个主要难题：缺乏可用于衡量领域进步的公共语料库；评测提炼系统输出结果的难度和代价，原因在于缺乏自动评测指标。

参考文献

[1] "Phase 4 GALE distillation annotation guidelines," Tech. Rep., BAE Systems, July 2009.

[2] R. Florian, H. Hassan, A. Ittycheriah, H. Jing, N. Kambhatla, X. Luo, N. Nicolov, and S. Roukos, "A statistical model for multilingual entity detection and tracking," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, pp. 1–8, 2004.

[3] A. Berger, S. Della Pietra, and V. Della Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[4] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. IT-13, pp. 260–267, 1967.

[5] X. Luo, A. Ittycheriah, H. Jing, N. Kambhatla, and S. Roukos, "A mention-synchronous coreference resolution algorithm based on the bell tree," in *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, p. 135, 2004.

[6] E. T. Bell, "Exponential numbers," *American Mathematics Monthly*, vol. 41, pp. 411–419, 1934.

[7] NIST, "ACE (automatic content extraction) English annotation guidelines for relations," 2008. http://projects.ldc.upenn.edu/ace/docs/English-Relations-Guidelines_v6.2.pdf.

[8] N. Kambhatla, "Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations," in *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, p. 22, 2004.

495

[9] "TAC 2009 knowledge base population track," 2009. http://apl.jhu.edu/paulmac/kbp.html.

[10] "Help:infobox," http://en.wikipedia.org/wiki/Help:Infobox.

[11] W. W. Cohen, P. Ravikumar, and S. Fienberg, "A comparison of string metrics for matching names and records," in *KDD Workshop on Data Cleaning and Object Consolidation*, 2003.

[12] Y. Freund and R. E. Schapire, "Large margin classification using the perceptron algorithm," in *Machine Learning*, vol. 37, no. 3, pp. 277–296, 1999.

[13] D. M. Bikel and V. Castelli, "Event matching using the transitive closure of dependency relations," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies*, pp. 145–148, 2008.

[14] A. K. Singla and D. Hakkani-Tür, "Cross-lingual sentence extraction for information distillation," in *Proceedings of the 9th International Conference of the International Speech Communication Association (Interspeech 2008)*, 2008.

[15] M. Levit, D. Hakkani-Tür, G. Tür, and D. Gillick, "IXIR: A statistical information distillation system," *Computer Speech & Language*, vol. 23, no. 4, pp. 527–542, 2009.

[16] M. Levit, B. E., and M. Freedman, "Selecting on-topic sentences from natural language corpora," in *Proceedings of the 8th Conference of the International Speech Communication Association (Interspeech 2007)*, 2007.

[17] J. Lin, "The role of information retrieval in answering complex questions," in *Proceedings of the COLING/ACL on Main Conference Poster Sessions*, pp. 523–530, 2006.

[18] E. V. Hoa, E. M. Voorhees, and H. T. Dang, "Overview of the TREC 2005 question answering track," in *Proceedings of the Text Retrieval Conference (TREC)*, 2005.

[19] G. A. Miller, "WordNet: A lexical database," *Communications of the ACM*, vol. 38, no. 11, 1995.

[20] K. Kamangar, "Unsupervised learning for information distillation," Tech. Rep., Idiap Research Institute, 2007.

[21] K. Kamangar, D. Hakkani-Tür, G. Tür, and M. Levit, "An iterative unsupervised learning method for information distillation," in *Proceedings of the IEEE International Conference on Acoustic Speech and Signal Processing*, pp. 4949–4952, 2008.

[22] I. Soboroff and D. Harman, "Overview of the TREC 2003 novelty track," in *Proceedings of the 12th Text Retrieval Conference (TREC 2003)*, 2003.

[23] J. Lin and D. Demner-Fushman, "Automatically evaluating answers to definition questions," in *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 931–938, 2005.

[24] S. Yaman, G. Tür, D. Vergyri, D. Hakkani-Tür, M. Harper, and W. Wang, "Anchored speech recognition for question answering," in *Proceedings of the North American Chapter of the Association for Computational Linguistics Human Language Technologies Conference*, pp. 265–268, 2009.

[25] F. Gey and D. Oard, "The TREC-2001 cross-language information retrieval track: Searching Arabic using English, French or Arabic queries," in *Proceedings of the 10th Text REtrieval Conference*, pp. 16–25, 2001.

[26] J. S. McCarley, "Should we translate the documents or the queries in cross-language information retrieval?," in *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*, pp. 208–214, 1999.

[27] K. Parton, K. R. McKeown, J. Allan, and E. Henestroza, "Simultaneous multilingual search for translingual information retrieval," in *Proceeding of the 17th ACM Conference on Information and Knowledge Management*, pp. 719–728, 2008.

[28] D. Hakkani-Tür and G. Tür, "Statistical sentence extraction for information distillation," in *Proceedings of the IEEE International Conference on Acoustic Speech and Signal Processing*, vol. 4, pp. IV–1–IV–4, 2007.

[29] E. M. Voorhees, "The evaluation of question answering systems: Lessons learned from the trec qa track," in *Proceedings of Question Answering: Strategy and Resources Workshop at LREC-2002*, 2002.

[30] "Phase 4 formal evaluation plan for GALE distillation," Tech. Rep. TR-2458, BAE Systems, Oct. 2009.

第 15 章

Multilingual Natural Language Processing Applications: From Theory to Practice

口语对话系统

Roberto Pieraccini, David Suendermann

15.1 概述

本章我们将讨论商用口语对话系统开发的问题。一个口语对话系统是一个复杂的机器，它可以管理面向目标的用户交互。口语对话的功能结构一般可以划分成三个部分：语音识别和理解模块、语音生成模块、对话管理器。现有大量的关于对话系统和它们各种各样的发展范例的研究文献；这里我们讨论关于设计、开发、部署并且维护一个商用口语对话应用的问题和方法。特别地，我们展示了部署大规模对话系统的商业组织是如何使用收集到的丰富数据来持续地调整系统并改善它的性能。本章讨论的问题与将应用移植到一个不同语言的开销有关。尽管提示本地化是很直观的，并且通常由专业的人工翻译来实施，但语音识别和理解的本地化造成了资源和开销的问题。然而，因为在当代口语对话系统中文法经常是由从大量被标注的口语语料中学习到的统计语言模型和分类器实现的，我们可以将文法本地化问题定义为语料翻译问题。我们将展示商业上可用的机器翻译如何用来翻译包含几百万口语句子的大规模语料并且允许创建与特定上下文有关的文法，在很少人为干涉的情况下，可以与人工调试的系统性能相比。

15.2 口语对话系统

口语对话系统可能是最为广泛接受的语音识别应用。口语对话系统是指在一系列连续的交互变化中机器可以使用语音与人类交谈的应用。最简单的情况下，一个对话系统可以由图 15-1 的功能图描述。

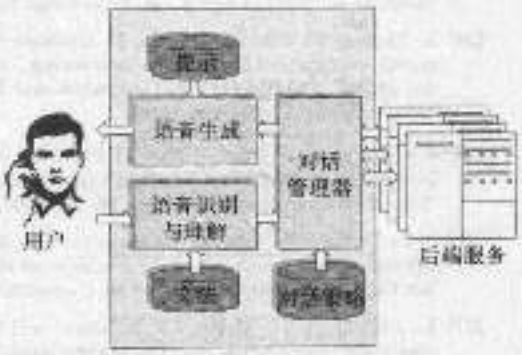


图 15-1 口语对话系统的高层功能性视图

基于一个称作**对话策略**（dialog strategy）的规则集的**对话管理器**（dialog manager）控制着一个**语音生成**（speech generation）模块，控制其在接收到由一个**语音识别和理解**（speech recognition and understanding）模块产生的用户语音的解释后，将生成怎样的信息或者请求。对话管理器也和外部的后端服务（backend service）通信，比如数据库、顾客关系管理系统（Customer Relationship Management，CRM）或者网络，从而抽取出完成交互必要的额外信息。我们应该说明的是，这里所说的方法是独立于具体语言的。然而，一些语言可能需要一些进行额外处理的应用，主要是在语音识别和理解模块部分，来处理词元的不同概念定义。

15.2.1 语音识别和理解

语音识别引擎是一个可以将输入语音解码成它的成分词的系统。理解模块则负责将语音识别模块返回的词串加上语义标签。例如，用户用"*You bet.*"回应一个提示："*Have you recently reboot your PC?*"一个良好的理解模块应该返回语义标签"YES"。研究型口语对话系统[1]经常在交互中的每一个阶段进行大规模的词汇识别，它假设使用者可能并且将会在对话的任何时候说任何话，这样系统就可以相应地给出响应。然而，众所周知使用者说话内容的分布将受到特定对话状态的影响，语言模型给予特定上下文更适合内容的性质，这一点已被证明对口语对话系统的性能提高是有益处的[2]。尽管现在可以建立一个从万级甚至十万级的大规模通用词汇表中进行解码的语音识别器，但是理解模块却做不到。无论理解模块是如何实现的，语言理解针对不同领域都会不一样，有时甚至对一个需要被解释表述的不同上下文也会不一样。因为这些原因（对特定对话上下文的语言模型的依赖性，甚至是对理解模块更加特殊的依赖性），还有建立有效的统计语言模型固有的复杂度，以及缺少特定的资源，早期口语对话产业主要是使用特定上下文正规有限状态文法——或者基于规则的文法（rule-based grammar）。基于规则的文法一般是根据SRGS标准[①]（Speech Recognition Grammar Specification）编写，尽管不同的识别引擎支持其他专有的文法定义语言，SRGS允许使用嵌入式的ECMAScript[②]代码写出任意上下文无关的文法规则。这些规则会定义识别引擎接受和识别的表达式的语法。ECMAScript依据语义槽标签定义了对返回串的解释。例如，这是一个SRGS文法规则片段，它被使用在一个百货公司的指路应用样例中：

```
  <rule id="selection" scope="public">
 <item repeat='0-1'><ruleref uri='prefixes.xml'/></item>
 <one-of>
   <item><ruleref uri='#rule_Footwear'/>
     <tag>out.answer='Footwear';</tag>
   </item>
   <item><ruleref uri='#rule_Jewelry'/>
     <tag>out.answer='Jewelry';</tag>
   </item>
   <item><ruleref uri='#rule_MensWear'/>
     <tag>out.answer='MensWear';</tag>
   </item>
   <item><ruleref uri='#rule_Movers'/>
     <tag>out.answer='Movers';</tag>
   </item>
 </one-of>
</rule>
```

<one-of>元素定义了一个可选集，它们由<item>元素指定。<ruleref>是另一个规则的引用，该规则被一个相关的URL标识。<tag>元素包含ECMAScript片段。例如，如果口语输入句子由规则#rule_Footwear解析，则执行ECMAScript表达式out.answer='Footwear'。ECMAScript的对象out在调用应用命名空间中可作为语音识

① http://www.w3.org/TR/speech-grammar/。

② ECMAScript(http://www.ecmascript.org/)是由国际组织ECMA标准化的脚本语言，JavaScript是ECMAScript的一个变种。

别返回的语义。prefixes.xml 的引用包含一个可选的前缀文法。

规则引用 #rule_Footwear 能够被下面的 SRGS 片段扩展：

```
<rule id="rule_Footwear" scope="public">
  <one-of>
    <item>footwear</item>
    <item>foot wear</item>
    <item>shoes</item>
    <item>boots</item>
  </one-of>
</rule>
```

这一规则声明：任何一个由<item>元素定义的任何表达式都是可说的，如果被识别，"Footwear" 将被赋值给输出槽（out.answer）。从例子中可以很明确地看出，基于规则的文法会变得非常复杂，并且可能包含对其他规则和脚本的引用。它们的维护等价于复杂代码的维护，基本上不可以自动生成，并且很多变化或者改进需要由语法专家手工处理（工业上认为是语音系统专家（speech scientist））。

从功能上来看，一个文法可以认为有两个组成部分，如图 15-2 所描述的。语言模型定义了所有可能由语音识别器处理的词串空间，并且语义分类器将任意词串映射到语义标签的一个有限集。如本章前面所示，语言模型可以被一个基于规则的文法中的规则描述，并且语义分类器能够由 ECMAScript代码实现。如果我们转到统计文法领域，语言模型则由一个 n 元组集合定义，并且语义分类一般由一个训练好的统计分类器完成，该分类器对由语音识别组件返回的词串进行处理。尽管一个基于规则的文法一般由手工完成，但统计语法则建立自一个大规模的样例句子集上，这些样本被转录成对应的词串，并且用相应的语义标签进行标注。表 15-1 展示了一个被转录和标注的口语句子的集合样例，这些口语句子主要来自人们对美国某大型光缆公司技术支持的电话咨询。被转录的口语句子是对 *"Please tell me the reason for your call"* 提示的响应。

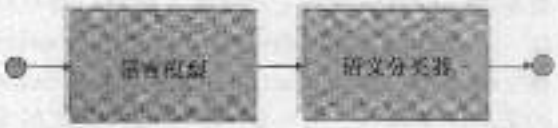


图 15-2　语音对话文法等价的功能性视图

表 15-1　口语表达的转录和标注集示例

转　录	标　注
want to cancel the account	SERVICE_CANCEL
cancel service	SERVICE_CANCEL
cancellation of the service	SERVICE_CANCEL
I want to discontinue the service	SERVICE_CANCEL
I can't send a particular message to a certain group of people	CANT_SEND_RECEIVE_EMAIL
I can't get messages on my email and Outlook Express	CANT_SEND_RECEIVE_EMAIL
I can't receive all my email	CANT_SEND_RECEIVE_EMAIL
I'm trying to send an email and it says it's not going through	CANT_SEND_RECEIVE_EMAIL
my emails are not being received at the address I send them to	CANT_SEND_RECEIVE_EMAIL
can't send	CANT_SEND_RECEIVE_EMAIL
can't send large files	CANT_SEND_RECEIVE_EMAIL
bounce message notfication	CANT_SEND_RECEIVE_EMAIL
message won't be sent won't send	CANT_SEND_RECEIVE_EMAIL
it concerns mac mail I can't open it	SETUP_EMAIL
when I set up the internet you didn't give the email account	SETUP_EMAIL
I can't set up my email account	SETUP_EMAIL

（续）

转 录	标 注
setting up email account	SETUP_EMAIL
cannot configure the email	SETUP_EMAIL
ihay registered my modem from my Internet and I need to get my email address	SETUP_EMAIL
all I need is to find out how to set up my sent email box to save my sent email	SETUP_EMAIL
I'd like to set up an additional email account	SETUP_EMAIL

一旦获得大规模的转录和对应的语义标注，一个统计语言模型便能够建立并用于约束语音识别器以及一个统计语义分类器。该统计语言模型是 n 元组形式的，一般 $n=3$，或者叫三元组。一个三元组是口语表达中任意词以其前面任意可能的词对作为历史的概率集合。所以，如果 t 是某口语表述中一个词的下标索引，那么组成三元组的概率集合将有如下形式：

$$p(w_t \mid w_{t-1} w_{t-2}) \tag{15.1}$$

有几种方式估算三元组，主要的问题是如何处理那些没出现在训练集中的三元组。为了达到那个目的，许多文献中探讨了不同的回退技术；详情参阅第 5 章的语言模型。

至于我们关注的统计语义分类器，可以使用多种技术。对应用于大规模口语语料库的不同分类器性能的探讨可参见 Evanini、Suendermann 以及 Pieraccini [3]。

15.2.2 语音生成

商用口语对话系统中的语音生成模块非常有限或者根本不存在。研究者实验了在自然语言生成（Natural Language Generation，NLG）模块之后使用文本到语音模块（Text-To-Speech，TTS）。然而，NLG 和 TTS 相结合的解决方案特性并不足以支持一个大范围使用的商业口语对话系统。甚至 TTS 应用于预先定义好的文本都会受限于一些情况，主要是信息的多变性使得高质量提示的事先录音变得不实际也不可能。实际上，大多数的商用系统使用有经验的解说员和配音员为应用事先录好所有需要预先定义的提示。对于复杂的应用，例如技术支持客户服务，录下 5000～10000 这样大量的提示再正常不过了。当需要时，语音合成的一种简单形式是用来播放可变的提示，例如任意数字。

15.2.3 对话管理器

口语对话系统的一个典型商业实现中，对话策略被描述成 call-flow（呼叫流程）[5]。一个 call-flow 对应于一个有限状态机的规格说明，通常组织成层次结构，其中节点代表了对话活动而对应于状态的弧。一个典型的活动能够指导语音生成模块，从而播放一个已被记录的提示，并且同时使用特定的文法来激活语音识别模块。其他活动可以查询外部的后端服务器；设置并且计算内部的变量，执行任意类型的计算，或者唤醒另一个 call-flow 作为一个子对话。

历史上，口语对话管理器是首先由软件工程师使用传统的编程语言（C、C++、Java）实现的，每一个新的应用都是被硬编码为一个特殊的有限状态机（或者 call-flow）。随着 VoiceXML 标准——最初由 VoiceXML 论坛[㊀]于 20 世纪 90 年代末起草，然后被万维网联盟（World Wide Web Consortium，W3C[㊁]）推荐采用，口语对话应用开始被实现为一个 Web 应用程序，类似于一个可视化的 Web 浏览器，例如 IE 和 Firefox；在每一轮的

㊀ http://www.voicexml.org/。

㊁ http://www.w3.org/TR/2007/REC-voicexml21-20070619/。

交互中，一个语音浏览器解释一个标记语言（如 VoiceXML），从而控制它的资源（如语音识别、TTS）。作为一个可视化的 Web 浏览器，语音浏览器使用 HTTP 协议与应用服务器通信，并且通过相应 HTTP 请求来获得 VoiceXML 文档。VoiceXML 文档指示浏览器播放特定的提示，并且借助特定的语音识别器和文法来识别输入的语音。播放提示资源、TTS，以及语音识别引擎由浏览器通过特定的协议控制，例如媒体资源控制协议（Media Resource Control Protocol，MRCP）[⊖]。

VoiceXML 标记语言也包含用于指示浏览器通过有条件方式取得另一个文件的指令，这样把一个静态的 call-flow 实现为一个已链接文档的集合。然而，随着应用复杂度的不断增加，正如在传统可视化 Web 应用中发生的那样，开发者从静态的模型——静态文档集合——转向动态的模型（基于需求生成的标签）。在这种情况下，应用服务器运行一个程序，该程序执行 call-flow 有限状态机，并且在交互的每一轮中动态产生 VoiceXML，以指导浏览器播放提示，并且识别输入的语音。而且，开发者能够构建一个通用目的 call-flow 引擎——对话管理器——并且连同它的属性使用一个指定的、专有
504 的标记语言指定 call-flow 的拓扑。对于对话管理演化的详细描述，请参看 Pieraccini 和 Huerta [6]。

图 15-3 展示了一个现代的商用口语对话系统。交互语音应答（Interactive Voice Reponse，IVR）平台包含一个 VoiceXML 浏览器，由它来解释 VoiceXML 文档，还有一个电话接口，从而可以连接公共电话网（或者等价地，一个 IP 电话网关）。VoiceXML 浏览器通过 MRCP 协议层控制着标准的语音识别和 TTS 引擎。

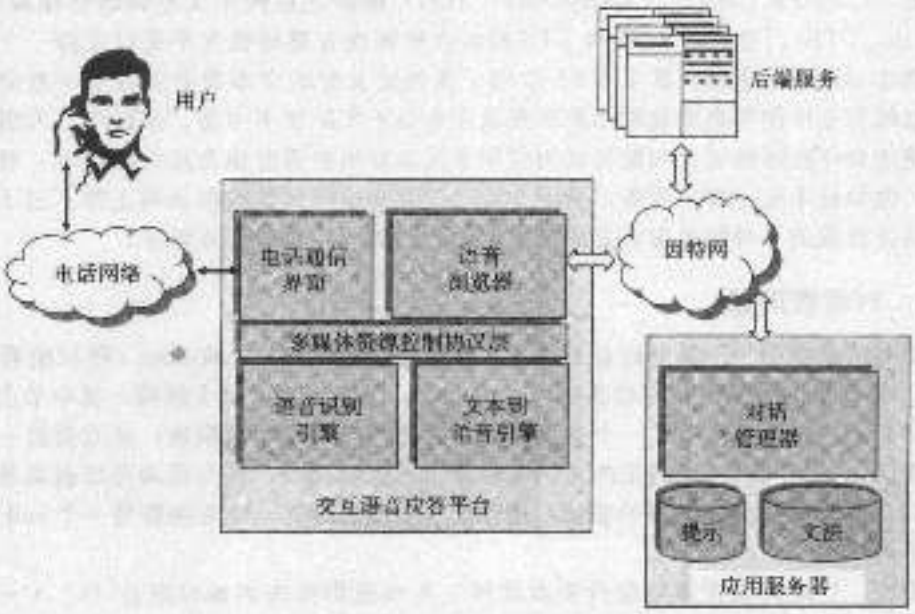


图 15-3　现代商用口语对话系统的体系结构

交互由应用服务器管理（通过一个常规的 Web 服务器实现），它通过由 IVR 平台发出的 HTTP 请求来提供 VoiceXML 文档。提示和文法一般与 URL 相关联，并且可能寄宿在与应用服务器相同的一台 Web 服务器上，或者网络的其他地方。对话管理可能偶尔访问

⊖　http://tools.ietf.org/html/rfc4463。

后端服务器，经常使用 SOAP（Simple Object Access Protocol）指令。IVR 平台和应用不必在同一本地网络而可能是地理上分布的，就如同通常情况一般。

15.2.4 语音用户接口

对于一个给定的应用，语音用户接口（Voice User Interface，VUI）是用来描述系统的提示是什么，在交互的每个步骤中被语音识别器接受的表达式的范围是什么，同时描述应用的一般逻辑。一个 call-flow 是一个用来描述应用 VUI 的有限状态机。VUI 常使用所见即所得（WYSIWYG）拖放工具来开发，该开发工具允许将所有的交互细节看作是一个层次化有限状态机。call-flow 创作工具编译用 call-flow 标记语言（典型地一个专有语言）编写的图形表示，然后被一个 call-flow 应用引擎使用，通过动态产生 VoiceXML 实施交互。图 15-4 是 call-flow 模块（流程）的一个图解描述[⊖]。 505

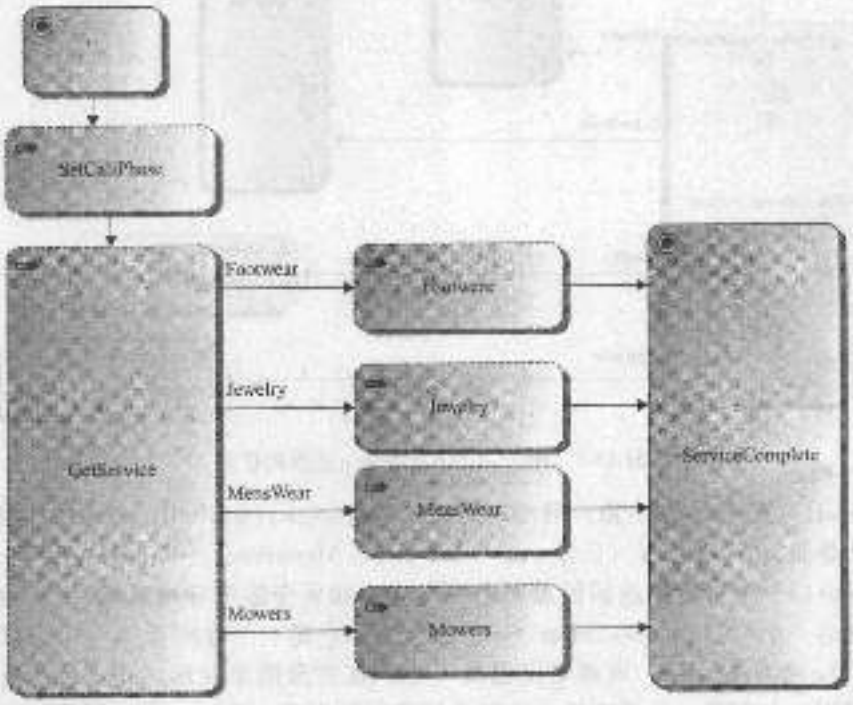


图 15-4 使用 WYISWYG 创作工具构建的 call-flow 说明示例

这些方框表示活动等价于传统的过程编程语言语句，弧代表了条件转移到其他活动的状态；也就是说它们等于传统面向过程语言中的 if-then-else 从句。图 15-4 左上角的第一个状态是处理的入口点，并且图表右边的活动，由～ServiceComplete 表明，表示返回到调用过程（在这里是主过程）。图 15-4 中其他所有的活动都是过程的引用：它们和定义在类似图 15-4 中的其他图上的子程序调用相对应。过程引用和传统程序语言的函数调用等价。例如，GetService 过程由图 15-5 表示。 506

⊖ 这里以及本章剩余部分，我们用 SpeechCycle's RPA Compose 当作开发高级语音对话系统的工具的例子进行讲解（可见 http://www.speechcycle.com），其他研究工具也是公开可用的，比如卡内基梅隆大学开发的开源工具 Olympus（http://apexart.speech.cs.cmu.edu/）还有麻省理工学院提供的 Galaxy（http://groups.csail.mit.edu/sls/technologies/galaxy.shtml）。

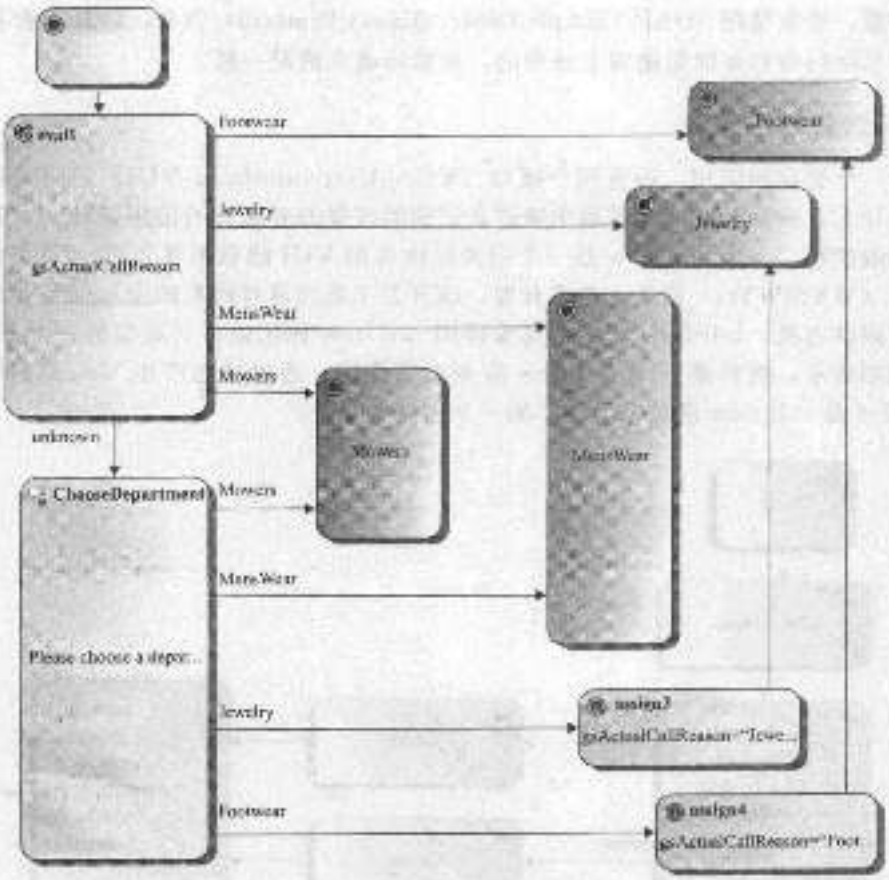


图 15-5　图 15-4 中 GetSevice 进程的扩展

称作 eval1（图 15-5 左上角）的活动对应于变量 getActualCallReason 的计算。如果变量的计算结果是 4 个可能值（Footwear、Jewelry、MensWear、Mowers）中的一个，那么当前过程会以一个合适的返回值返回给调用者。如果变量没有被赋值（值不确定），那么过程会到达一个叫做 ChooseDepartment 的活动，它是一个提问活动（用方框中左上角的图标表示），也叫做 DM，或者对话模块[7]。在它最简单的形式中，DM 播放一个提示，并且使用一个或者一些指定的文法激活语音识别引擎。然而，DM 需要处理一些口语问题，例如超时、重复提示以及确认。所以，DM 合适的配置需要设计者设定一系列的功能参数。

507

描述 DM 所有属性的意义如图 15-6 所示，这个描述超出了这一章的范围，但是了解 DM 使用的各种不同的提示和文法是最重要的性质这个考虑就足够了。例如，声明是提示的第一个部分，其中语音打断是不可行的，例如，“Please choose a department”。问题是提示的内容部分，例如“Footwear”、“Jewelry”、“Men's Wear”或者“Lawn Mowers”。一般只有热点词汇，如“help”、“operator”，以及那些对应语法，在 DM 的声明部分是活跃的。所有其他的内容文法在问题过程中是活跃的。活跃的语法由识别文本属性来指示。多个文法可以同时使用。例如，对这里所描述的 DM，三个文法在问题中是活跃的，如图 15-7 所示。

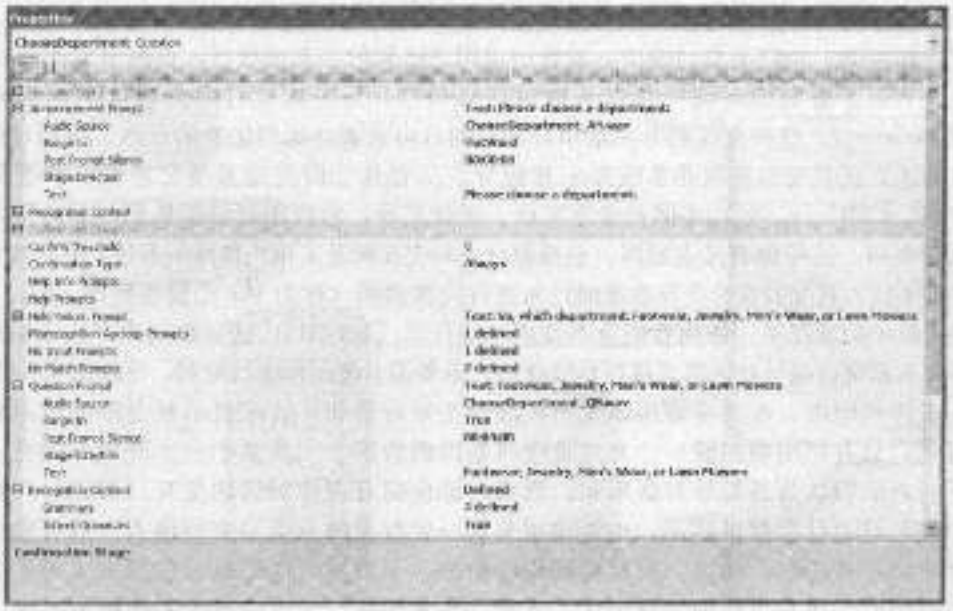


图 15-6 一个对话模块的属性



图 15-7 在图 15-6 中的对话模块的某阶段的活跃文法

图 15-7 中三个活跃的并行文法是一个语音文法，包括对每一个可能部分描述的表述（见前面的文法例子）。DTMF 文法描述电话键盘上哪个按键响应哪个选择，以及捕获多种应答操作请求方式的操作文法。

为了总结这一节，我们注意到，构建一个复杂的应用需要设计 call-flow，该设计在包含提示和语法的活动基础上连同许多其他参数，充分描述了交互的演变过程。构建一个口语对话应用需要一个可用的创作工具辅助完成编程工作，例如前面描述的应用，它允许开发一个 call-flow，可将它看作是一个层次有限状态机。当前的复杂应用的商业对话系统，例如技术支持，可能包含上百页，如图 15-4 和图 15-5 中所示的那些，以及上千个活动。

15.3 对话形式

当今采用的大多数商用对话系统遵循指导性对话范例，意味着系统一般通过提问题和解

样用户回答来指导对话的进程。在交互的每一个步骤，对话系统会问一个特定的问题，如通过提供选项列表、建议可能的响应，或者引用用户已知的一系列项目，如城市名或者日期。

另一方面，研究型系统一直以开放对话为目标，它可以允许一定程度上的混合主导（mixed initiative）。这种交互的方式给用户充分的自由去表达他们想要的东西，特别是使用自然语言输入，而只受到有限的系统提示和指导。尽管几个研究型系统已经在一些受限领域（如 ATIS、CMU [8, 9]）不同程度地实现了混合主导，但商用对话系统仍然尽可能地保留指导对话范例。这样做有几点原因，包括混合主导交互缺乏实用、鲁棒的表述，并且难以分析对所有可能输入状况的混合主导系统的行为进行完整的测试（称为 VUI 完整性原则 [6]）。而且，在开放式提示的情况下，使用者通常不知道该说什么 [10, 11]，这导致用户产生不明确的请求，即要求系统以指导对话形式进行后续处理，或者请求超出系统的限制，导致交互失败。

509 介于这些原因，在当今商用系统中，混合主导对话和开放式提示是受限的或者根本没有。通常，只有初始的问题——尤其是呼叫原因的辨识——是基于一个开放式的提示，然后剩下的对话被认为是指导对话形式。然而，即使拥有指导对话的交互，即提示严格地指示用户或者对说什么提供暗示，也能够观察到一定数量的不受约束的输入，或者输入与提示所要求的不相匹配。例如，付账应用程序中的一个提示“你想通过信用卡支付还是支付中心支付呢?”，用户可能回答借记卡、支票，或者在线，与之前提示的选择全不同。

15.4　自然语言呼叫路由选择

尽管通过精巧设计的提示，大多数用户趋向于使用关键词和短语来回答，但仍然存在着几种应用，这些应用指导对话的方式不实用。这种类型的应用以一个领域类型为特征，这种类型很复杂，并且为大多数用户所不知。例如，呼叫路由选择应用就完全基于这种策略。在这种应用中可能存在着大量的不同类型的呼叫原因（有时多到数百个 [12]），这不可能被单一的或者储存的指导对话所处理。辨别所有的呼叫原因需要多个问题，虽然呼叫者能够有效并且清楚地通过一句话表达他们需要的。

一个可能的解决办法就是提供一个清单，其中包含了所有用户可能使用的有语义区别的原因。然而，所有可能原因的列表可能会过于庞大，并且构建一个捕获所有可能被用来描述原因的表达文法可能是不现实的。另一方面，一个详尽的无歧义的、使用堆叠的或者层次的、用于识别呼叫原因的指导对话菜单可能需要解决多个问题，导致过长的交互，糟糕的用户体验，甚至可能丢了客户。这种情况下，一种解决方法就是让呼叫者自由地表达，并且系统后方有一个可以自动将用户划分到预定类别的分类器，正如前文所述。这种技术叫做 How May I Help You [13]，统计呼叫路由选择 [14, 15]，或者统计自然语言理解 [16]，是一种简化了的语言理解方式，结合了一个包含自然语言灵活性（一个开放性的、会引发大量可能的用户表述的提示）结构方法（有限类别或路由）的鲁棒性。事实上，对话还可以以指导对话方式来构建，因为交互的输出是预定义类别中的一个。统计文法基于学习自大量数据的 n 元组文法，一般可以被用来解决这个问题。消息的语义解释由统计分类器得到，如前所述。

15.5　三代对话应用

随着 20 世纪 90 年代中期电话口讲对话产业的开始，我们见证了至少三代这种系统的
510 进化。每一代的困难不仅是复杂度的增加，而且还包括所使用的不同结构。

表 15-2 展示了各代系统特点的概括。第一代系统信息量最大的在于它们从用户那里

请求一些信息，然后提供一些信息作为返回。第一代系统的例子，大多数开发于20世纪90年代中期或者晚期，主要是包裹追踪、简单的金融应用，以及获取状态信息。那时，对于对话系统的开发没有统一的标准，因此第一代对话应用在私有平台上完成，典型的就是已有的按键式IVR结构的发展。

表15-2 几代对话系统（VXML语言扩展标记语言，SLU统计语言理解）

	第一代	第二代	第三代
时间段	1994~2001年	2000~2005年	2004年至今
应用类型	信息型	交易型	问题解决型
例子	包裹追踪，航班状态	银行，股票交易，车票预订	客户服务，技术支持，咨询台
体系结构	专有的	静态VXML	动态VXML
复杂度（DM的数量）	10	100	1000
交互次数	较少	10	100
对话形式	有指导的	有指导的+自然语言（SLU）	有指导的+自然语言（SLU）+混合驱动的

一个call-flow中DM的数量一般预示着一个应用的复杂度。第一代应用显示的复杂度一般是几个到十几个DM，跨越几轮交互。早些的应用支持严格的指导对话交互，在每一轮中它也会导致有限的文法或者词汇。

第二代应用是典型的交易型，也就是说它们会代表用户进行实际的交易，例如在银行间移动基金、股票交易，或者买卖股票。这一类应用大多数都是遵循新标准开发的，如VoiceXML文档集合。复杂度受到了数十个DM，有若干轮交互，每一轮大概交互几十次。同时，一些应用开始使用统计文法来将受限的用户口语表达映射到一个有限的预先定义好的语义类型（即SLU统计语言理解）。自然语言模态——与指导对话截然相反，开始被用于呼叫路由选择。

511

虽然第一代和第二代对话应用的模型可以被描述为填表范例，而且交互遵循预先定义好的简单脚本，第三代系统在复杂度上已经有了一个质的提高。问答应用，如客服、咨询台，以及技术支持，都已经达到了数千个DM的复杂度，以及若干轮动态交互，每轮可交互100或者更多次。随着应用复杂度的进化，系统结构也改变了，逻辑部分从客户端（VoiceXML浏览器或语音浏览器）转移到了服务器端[6]。如前所述，现在越来越多的系统基于通用对话应用服务器，它解释对话规格说明，并且向语音浏览器提供动态产生的VoiceXML文档。最终，第三代系统的交互形式从严格的指导对话应用转向了使用更加自然的语言，以及一定程度的混合主导。

15.6 持续的改进循环

通常，第三代对话系统具有集成功能，该功能与后台数据库或者远程设备进行通信，支持多输入和输出形式，有时可以与使用者保持超过20分钟的交互。为了使呼叫者接受这样的环境，一些先进的VUI技术的使用是很关键的，比如结合自然语言理解、受限的混合主导，以及动态应答生成。如前所述，自然语言理解是最先被应用到自动口语对话系统的，在第二代中，它作为呼叫分类器，或者呼叫路由。呼叫者在呼叫开始时被询问一些问题，例如“Briefly tell me what you're calling about today”。呼叫者的回答随即被识别，并且基于语义分类器的结果将呼叫转到合适的代理。然后人工代理与呼叫者交互，提供服务，例如包括技术问题解答、账单支持、预订处理等。相比较而言，第三代对话系统被设

计用来在更大的程度上模仿人类的角色。

随着对话系统不断完善，呼叫者的体验越来越好。现代对话系统的几个设计的特色鼓励呼叫者像他们与真人交互那样对话。这样的特点包括在对话开始时的开放式问题，以及如“help”和“repeat”这样的对话中随时的全局命令。这一设计鼓励呼叫者说明那些明显地不是由对话系统提示的东西。此外，贯彻的指导性的对话提示，在这些提示中呼叫者被要求从列表中选择一项，经常无意地引出不合语法的、不完整的、含糊的，或太细节的话语。那会导致人工的基于规则的文法无法应对不少的用户输入。即使听几百次呼叫也很难为每月接听几百万次的对话系统提供一个对于任何时候都可能出现的口语表述的全面覆盖，使用当前手工的基于语法的方法几乎不可能达到这一期望。Suendermann 等 [17] 提出了一个方法，通过使用呼叫者的口语表达来调整 SLU 分类器，并且在每一个对话
512 识别上下文中使用，它可以持续地改善对话性能，即使当指导对话提示请求一个简单的回答，例如是或者否。为了能够将该过程变得更加自动化，收集、转录、标注、语言模型、分类器训练、基线测试，以及文法公布均是以程序方式执行，是一个几乎不需要专家监督的连续循环规程。其目的是确保系统性能的不断完善，并且达到最高可能性的识别效果，该识别效果以统计方式反映了呼叫者的实际行为。这个过程已在超过 200 万的口语句子中被验证，这些句子来自一个复杂呼叫路由选择和解答难题的对话系统超过 50 万的完整呼叫，从本质上提高了系统的性能。

15.7　口语句子的转录和标注

对第三代大规模对话系统的调整通常需要成百上千的句子被转录和进行语义上的标注。尽管对如此大规模的数据进行转录和标注是部分自动的，但这仍将让一些人忙上几个月。虽然转录是相对简单的工作，但是语义标注（例如将一个词汇内容映射到大量语义符号中的一个）需要有相应的应用知识。不仅是标注者需要了解呼叫者在系统提示的上下文中表达的意思，对于语义标注还存在几个方面使得它不易理解的方面，例如：

口语表达可能在给定的语义分类集中没有代表，表明它们不在文法的表示范围之内；

当不符合文法表达的口语比例上升并且容易区分它们自己的模式时，标注者会建议引入新的语义分类，该分类必须符合系统的逻辑；

口语表达可能是存在歧义的、模糊化的、过于特定的，或者是其内容属于多个语义分类，这使得标注者很难做出抉择；

标注必须遵循一些确定的标准，从而产生准确有力的结果，包括完整性（completeness）、一致性（consistency）、相容性（congruence）、关联性（correlation）、混乱度（confusion）、覆盖度（coverage）以及语料库（corpus）大小等标准，也称为 C⁷ [18]。

这些问题强调，口语对话系统中语音识别全面调整需要细心的计划和协调。

15.8　口语对话系统的本地化

公司使用的大规模第三代口语对话系统大多数用于优化他们客户服务的电话服务。很多这样的公司都是国际化运作的，从而需要本地化他们的电话服务，包括口语对话系统。
513 而且，一些国家会有大量的多语言用户，例如英国的英语和西班牙语用户，以及加拿大的英语和法语用户。

如前所述，对话系统中的用户交互主要由三个内容决定：call-flow，提示，以及文法。另外，我们必须考虑语音识别的本地化，如果需要，还要考虑 TTS 引擎。然而，倘若使

用商用识别以及TTS引擎，而且大多数关商用引擎制造商提供语言包（例如，所有主要语言关声学和发音的模型扩展关集合），本地化语音识别和TTS引擎就像获得产品中语音引擎必需关语言扩展那样直接。因此，本节其余内容我们分析和口语对话系统本地化有关的call-flow、提示以及文法问题。

15.8.1 呼叫流程本地化

call-flow决定了交互的逻辑，粗略地说，就是在交互中什么时候什么问题应被问及，以及什么信息应被呈现给用户。然而，在被呈现给用户的问题和叙述中的特殊语言形式，事实上也就是提示，以及指定呼叫者可以说什么以及如何被转化为语义标签关文法，不应该是call-flow的一部分。它们应该被特定的占位符代表（动态变量、表），从而在逻辑和语言内容间实现一个清晰的分割。call-flow是一个大图说明，图的节点代表了系统活动，并且更与条件关联。例如由一个语义分类器生成关用户交互后返回的语义标签。而且，符号语义标签可以是语言独立的，一个逻辑和语言内容之间分割的例子是前面提到过的call-flow布局和DM属性。

我们一般假设，call-flow的逻辑部分在于不同语言间不会改变。尽管这一假设在一些语言和文化比较相近的语言间是成立关，例如美国英语和西班牙语，但是在一些语言和文化非常不同关情况下它可能就是不成立的，比如英语和日语。这里，将应用从一种语言移植到另一种语言可能需要改变问题关提问顺序，而且由于文化的原因可能需要修改call-flow。然而，这一章剩下的部分，我们假设call-flow的逻辑部分不需要本地化。

15.8.2 提示本地化

提示代表了在交互的每一步中系统所说的内容。典型地，为了从用户收集一条信息，需要几条提示，因为通过语音识别收集信息可能需要在不同的对话活动中进行，而且口语语言交互中存在很多典型问题，例如语音识别的拒识、超时以及低可信度。所有关这些活动都是一个对话模块的逻辑部分。下面列出了所有需要被设计用来收集信息的典型提示，假设我们要收集一个电话号码：

- 主收集提示：该提示在信息第一次被请求时提出，例如，*"Say or enter your ten-digit telephone number"*。
- 重试提示：如果语音识别拒绝了第一次输入，一个重试提示将建议用户将同样的信息再说一遍，例如，*"I didn't get that. Please say or enter your ten-digit telephone number again"*。
- 确认提示：如果识别器返回一个中等的可信度，候选识别将被确认。例如，*"That was three one zero nine two six seven one two three，right?"*
- 修正提示：如果用户否认了被确认的候选识别，系统会再一次提示。例如，*"I am sorry，please say or enter your ten-digit telephone number again"*。
- 超时提示：这个提示会出现在当分配给用户说话的时限结束而用户还没有发话并且语音识别将超时。例如，*"I didn't hear anything. Please say or enter your ten-digit telephone number."*
- 帮助提示：当用户要求帮助时，该提示会出现，例如 *"Sure，here's some more information. I'm looking for the phone number you are calling from. Please say or enter your telephone number one digit at a time，starting with the area code"*。

514

- 返回提示：当系统再次面对相同问题时该提示会出现，在一个转移之后，例如在一个帮助提示之后，如 *"So, just say or enter your ten digit telephone number"*。
- 操作提示：该提示出现在用户明确地要求一个人工操作员或者按零时，例如 *"I understand you would like to speak to an agent, but I need to get your phone number first in order to route you to the right agent. Please enter or say..."*

这份清单并没有穷尽，并且每一个对话模块的可能提示集可能比这个更大。例如，在第一次尝试之后可能存在不同的重试以及超时提示，并且当这些提示达到最大数量，系统声明无法完成信息收集之后，你也应该为此设计提示。或者提示会根据系统已知关于用户的信息而个性化，例如对话系统对不同专业层次的用户会有不同的提示，对不同年龄的用户不同的提示等。因此，对于每一份收集来的信息，你不能只提供一个提示而应该是很多个以处理所有可能的谈话情况。所以，即使是一个简单的系统，有几百个提示也是很正常的，而对于复杂的系统，当该系统需要本地化时，需要设计、管理，以及翻译成不同语言的成千上万的提示。

因为一个口语对话系统的性能非常容易受到提示质量的影响，所以从语言学和听觉角度来看，当播放提示时获得其考虑到上下文的高质量翻译是非常重要的。显然，完成一个高质量的翻译的唯一途径就是通过雇用一个专业的翻译人员在一个 VUI 设计者的帮助下将提示逐个翻译。尽管一个口语对话系统可能包含成千上万的提示，但是翻译的代价也不至于太高，并且翻译工作一般不用任何自动化处理。

提示翻译中的一个主要技术就是不同语言中 call-flow 和提示集之间关系的维护。call-flow 维护环境需要包含本地化和提示管理工具，这些工具允许保持不同语言中面向应用修改的提示的不同版本。工具需要标记一种语言中被修改、添加或删除的提示，并且对于其他语言中的相应提示需要相似的操作。没有这个工具，多语口语对话系统的维护可能变得过于不实用并且昂贵。

15.8.3　文法的本地化

将一种口语对话系统移植到另一种不同语言时，文法的本地化是问题最多的。一方面，基于规则的文法很难翻译，因为我们不能将它们以它们原有的形式展示给专业的翻译人员，例如一个之前描述的基于规则文法的例子，又希望它们生成一个精确的翻译。这是因为短语和句子经常被划分为不同层次结构的语法成分。搞清楚它们的意思与搞清楚其他人写的软件一样难。所以，翻译一个基于规则文法的努力应该比得上重写这些规则。但是一个专业的翻译人员不会写文法，所以我们需要雇用一个说母语的语言科学家，或者让语言科学家与目标语言的一个专业翻译专家合作。

另一方面，第三代应用中普遍使用的统计文法不能够直接翻译，因为它们由 n 元组和统计分类器组成。如果专业的翻译人员不能够编写基于规则的文法，那更谈不上写 n 元组和统计分类器了！而从目标语言的转录和标注的语料库中更容易重新训练出 n 元组和分类器。但是获得目标语言的新语料库可能很困难或不切实际，而且完整地将有成千上万口语句子的语料库翻译成目标语言经济上是不允许的，并且人工翻译也很难有合理的时间表。然而，机器翻译可以用来做这件事。

下面，我们报告一下使用商业机器翻译引擎自动完成本地化统计文法的研究工作，尤其是当目标语言没有充足数据资源时，这非常有用。

15.8.4　源端数据

作为使用机器翻译来进行文法本地化研究的一个例子，我们使用互联网上搜集的属于

英语对话系统的大规模数据，如 Acomb 等 [4] 描述的。时间跨度超过 3 年，该系统处理了几百万的呼叫。从这些呼叫中的一个相当大的子集获得口语句子，进行转录，并且基于一个完整的语义类别列表进行语义标注。表 15-3 列出了源语言可获得的数据规模概况：它列出了转录口语呼叫的数量，转录（但是不同的）和标注的口语句子的数量，以及一个系统复杂度的标示——它考虑了 DM 和文法的数量。

原始的英语对话系统在考虑将它本地化之前已经经历了持续的循环优化（如前所述）。因为上述涉及转录和标注生成的人工工作量随着收集的数据越来越多而逐渐减少，效率不断增加，越来越多的口语句子可以在给定的时间里被处理 [19]。图 15-8 展示了口语句子在收集时间段上的分布，这表明收集的量从项目开始时就不断增加。

表 15-3 源端英语数据概况

呼叫	1 159 540
转录的口语句子	4 293 898
不同的口语句子	278 917
标注的口语句子	3 845 650 (85.5%)
DM	3332
文法	253

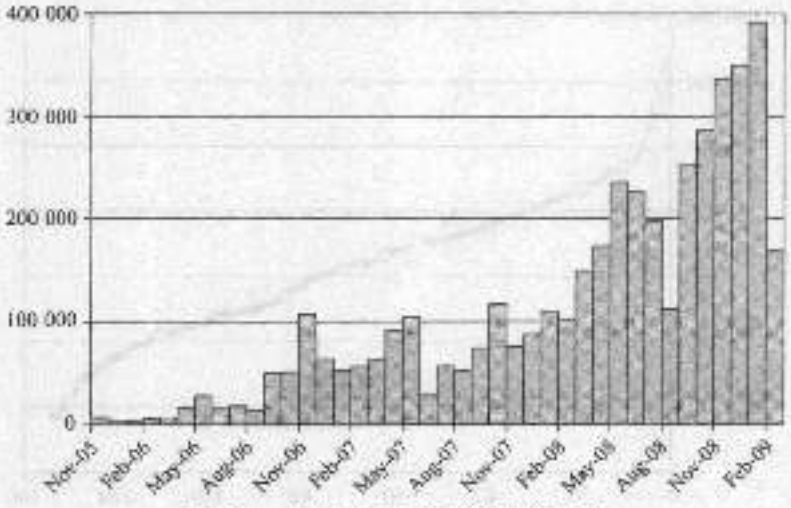


图 15-8 每个月收集到口语句子的数量

15.8.5 训练

表 15-3 中所有的 4 293 898 个转录的口语句子，这些句子由高川统计机器翻译软件从英语翻译到西班牙语。事实上，这只是通过翻译 278 917 个不同的句子完成的，并将翻译关联到源端口语句子的上下文。图 15-9 展示了语料库中不同句子的类似 Zipf 的分布。翻译完全是以一种无监督的方式进行。没有对输出进行修正或对机器翻译进行调整。对于所有不同的文法，被翻译为西班牙语的口语句子和它们的原始语义标注被分别用来训练统计语言模型和统计分类器，其中参数使用标准的设置，因为没有开发集数据可用。

图 15-10 以降序展示了英语语料库中口语句子对于每个文法数量的分布，这表明，有的文法有超过一百万的口语句子（典型的 yes/no 上下文），但有许多文法面临数据稀疏问题（22 个文法特征只有少于 100 个训练句子）。

15.8.6 测试

因为在文法生成的时候，西班牙语目标端的系统还没被部署，所以我们只好在自动翻译文法的一个子集上做测试。为了达到那个目的，我们收集、转录，并且标注来自一个类

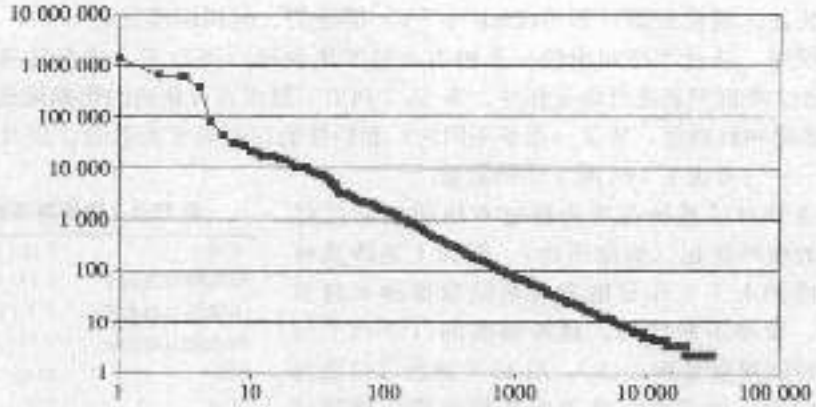


图 15-9　英语语料库中不同句子总频率分布

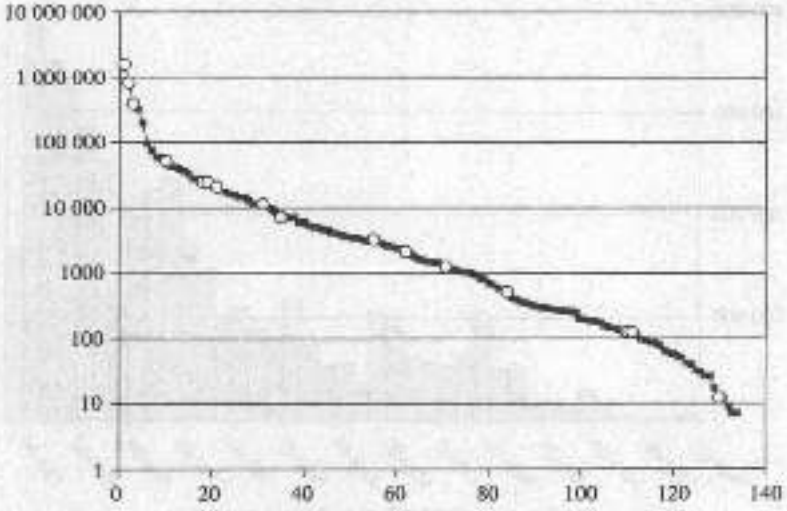


图 15-10　英语语料库中每个文法的口语句子数量的分布

[516]

似互联网疑难解答对话系统的已有的西班牙语版本中的有限数量的口语句子。这些数据的特性在表 15-4 中展示。

表 15-4　西班牙语测试数据的特点

呼叫数	953
转录的口语句子	11 476
标注的口语句子	11 476 (100%)
DM	144
文法	17

在测试集中发现的文法在图 15-10 中以白色的圈表示，表明它们在可用训练数据上分布于不同数量级上。收集到的口语句子集在各自上下文下使用自动翻译的文法来进行语音识别和分类的批量实验。对于 11 470 个口语句子中的每一个，分类结果现在被用来和相同口语句子的语义标注进行比较，评估实验结果的准确率是与标注匹配的语义分类结果的听觉事件的数量再除以听觉事件的总数。这些事件包含范围内和范围外的口语句子，以及噪声、背景语音等。整个测试集的总体精度是 85.0%，这相比于基于手工制作文法的对话系统的性能确实有很大提高。事实上，我们的经验是基于规则文法的系统的平均准确率往往低于 80%。为了得到一个更可靠的比较标准，我们观察了英语源端对话系统的性能，该系统被不断调整优化使用了许多年，并且发现最新的系统版本的性能达到了

90.7%（用930个呼叫，11 274个完整标注的口语句子来测试）。

使用相同的系统，但是性能却低于在源语言端的原因可以解释如下：

- 目标端声学模型的劣势。在我们的实验中，我们使用了一个过时的西班牙语语音识别器，它的声学模型性能明显达不到英语的部分。例如，在“yes/no (si/no)”上下文中，相比相同的独立于任何语言学因素的英语上下文，我们观察到系统拥有更高比例的错误接受及拒绝。
- 翻译模型的劣势。统计翻译不仅产生大量众所周知的错误，而且人工翻译也可能会有出错的情况：文法一般是基于呼叫者的口语句子设计的，该口语句子是用来应答限制呼叫者使用语言的系统提示。例如，一个西班牙语提示可能会说“Cuando esté desconectado, diga continúe”，这翻译自英文提示“When it's unplugged, say continue”。因此大多数英语回答者将是“continue”，一个像人一样的机器可能翻译成西班牙语的“continuar”，而不依赖提示的正确的“continúe”。所以，为了达到更准确的翻译候选结果，应考虑使用各自系统的提示和其他一些应用相关的信息来对它们重新打分。 519
- 这个实验没有可用的开发集数据，因为这需要收集一（小）部分来自目标语言的口语句子，和它们的转录与标注。一旦目标端系统的第一个版本投入生产，这些数据就可以得到，并且能够用来调整语言模型和分类器。

无论如何，我们已经展示了即使一个最初性能较低的对话系统也能够用很少人力就应用于不同语言中，而且，一旦系统部署好并且收集到可观数据量的数据，就可以引入前面章节所述的持续调整过程改进语音识别性能以达到一个可接受的水平。

15.9 总结

在这一章中，我们详述了当前用来构建商用对话系统的架构、技术以及方法。一个商用对话系统的架构主要由三个模块组成：语音识别和理解，语音生成，以及对话管理器。语音识别和理解模块的目标是为每个语音输入分配一个或者多个语义标签。尽管工业上仍然使用基于规则的文法，该文法编码了可能口语句子的语法和语义，但采用完全统计的方法仍然有几个好处。尤其是使自动调整所有的文法成为可能，并提供了一个获得大量用户口语句子的转录和语义标注的机会。商用对话系统中的语音生成大多数是基于高质量的提示录音集来完成的。最后，对话管理器使用有限状态机方法，明确地把整个交互编码为所谓的 call flow。非常有效的 GUI 工具使得 VUI 设计者能够设计和开发非常复杂的、通常包含数千个模块的交互过程。然后我们描述了已经开发出来的并且商用的不同类型的对话。对话系统产业包括从非常简单的信息传递应用到交易型的问题解决系统。然后我们讨论了对话系统对于不同语言本地化的问题。当大量转录和标注的源端语言可用，并且统计文法而不是传统的基于规则文法被使用在整个应用中时，我们展示了使用机器翻译的语音识别的本地化可以是很直观的并且代价不高。对建议方法的一个样本实现的测试表明这个 520
方法胜过手工操作，即使它不能达到原始对话系统在源端语言中相同的准确度。当然，一旦系统用在新的目标语言中，持续的调整将带来性能提高甚至达到源语言的水平。

参考文献

[1] D. Bohus, A. Raux, T. Harris, M. Eskenazi, and A. Rudnicky, “Olympus: An open source framework for conversational spoken language interface research,” in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2007.

[2] A. Gruenstein, C. Wang, and S. Seneff, "Context-sensitive statistical language modeling," in *Proceedings of the Annual Conference of the International Speech Communication Association*, 2005.

[3] K. Evanini, D. Suendermann, and R. Pieraccini, "Call classification for automated troubleshooting on large corpora," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, 2007.

[4] K. Acomb, J. Bloom, K. Dayanidhi, P. Hunter, P. Krogh, E. Levin, and R. Pieraccini, "Technical support dialog systems: Issues, problems, and solutions," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2007.

[5] W. Minker and S. Bennacef, *Speech and Human-Machine Dialog*. New York: Springer, 2004.

[6] R. Pieraccini and J. Huerta, "Where do we go from here? Research and commercial spoken dialog systems," in *Proceedings of the SIGdial Workshop on Discourse and Dialogue*, 2005.

[7] E. Barnard, A. Halberstadt, C. Kotelly, and M. Phillips, "A consistent approach to designing spoken-dialog systems," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, 1999.

[8] C. Hemphill, J. Godfrey, and G. Doddington, "The ATIS spoken language systems pilot corpus," in *Proceedings of the Workshop on Speech and Natural Language*, 1990.

[9] A. Rudnicky and W. Xu, "An agenda-based dialog management architecture for spoken language systems," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, 1999.

[10] S. Oviatt, "Predicting spoken disfluencies during human-computer interaction," *Computer Speech and Language*, vol. 9, no. 1, 1995.

[11] J. Williams and S. Witt, "A comparison of dialog strategies for call routing," *Speech Technology*, vol. 7, no. 1, 2004.

[12] D. Suendermann, P. Hunter, and R. Pieraccini, "Call classification with hundreds of classes and hundred thousands of training utterances . . . and no target domain data," in *Proceedings of the 4th IEEE Tutorial and Research Workshop on Perception and Interactive Technologies for Speech-Based Systems: Perception in Multimodal Dialogue Systems*, 2008.

[13] A. Gorin, G. Riccardi, and J. Wright, "How may I help you?," *Speech Communication*, vol. 23, no. 1/2, 1997.

[14] J. Chu-Carroll and B. Carpenter, "Vector-based natural language call routing," *Computational Linguistics*, vol. 25, no. 3, 1999.

[15] I. Zitouni, "Constrained minimization and discriminative training for natural language call routing," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 16, no. 1, 2008.

[16] V. Goel, H. Kuo, S. Deligne, and C. Wu, "Language model estimation for optimizing end-to-end performance of a natural language call routing system," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2005.

[17] D. Suendermann, J. Liscombe, K. Evanini, K. Dayanidhi, and R. Pieraccini, "From rule-based to statistical grammars: Continuous improvement of large-scale spoken dialog systems," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2009.

[18] D. Suendermann, J. Liscombe, K. Evanini, K. Dayanidhi, and R. Pieraccini, "C5," in *Proceedings of IEEE Workshop on Spoken Language Technologies*, 2008.

[19] D. Suendermann, J. Liscombe, and R. Pieraccini, "How to Drink from a fire hose: One person can annoscribe 693 thousand utterances in one month," in *Proceedings of the SIGdial Workshop on Discourse and Dialogue*, 2010.

第16章

Multilingual Natural Language Processing Applications: From Theory to Practice

聚合自然语言处理引擎

John F. Pitrelli, Burn L. Lewis

16.1 概述

许多早期的语音和自然语言处理应用程序都是基于单处理引擎，例如，实现听写的语音到文本转换（STT，也即语音识别）引擎或者一个实现文本翻译的翻译引擎。然而，现在许多引擎已经达到足够的精确度，能够把它们结合起来完成以前不可能完成的更复杂的任务，尽管结合中会带来固有的误差积累。例如文本领域中的应用，包括语义搜索、企业报告和其他商业智能系统、问答系统、医案挖掘、跨语言搜索等。音频处理的例子包括音频/视频搜索和编目、语音到语音翻译、外语广播新闻分析等。

这类应用程序共享许多通用的引擎，如说话人识别、语音到文本转换、文本切分、语法分析、命名实体检测、共指分析、词性标注、翻译等。共享的优势加上应用程序的绝对复杂性，促使了把这些应用程序实现为一系列步骤，而这些步骤则由独立的引擎组件执行。这样做使得组件的开发和测试可以分开进行，同时降低了大型应用程序调试的难度。

制作这些应用程序的原型通常需要把一个引擎的输出按照下一个引擎的要求重新格式化，并将其输入到该引擎等。但应用程序开发者可以通过创建一个聚合处理器，将数据自动从一个引擎移动到下一个引擎，当需要时即重新格式化。这样会带来很大的好处，如下所示：

- 一组引擎的单点调用；
- 引擎之间的高效数据传输——无须手动传输转换；
- 容错，失效备援——如果引擎出现故障，自动切换到一个备份；
- 通过系统组合技术可提高系统的准确性[7，12]。

然而，聚合同样带来了一些挑战：

- 异构计算环境：不同的引擎往往是由不同的群体，使用不同的操作系统，不同的编程语言，不同的字节顺序等来开发的；
- 远程操作：引擎常常在不同的地点被开发和维护，这样使得远程处理更有利，同时可避免把引擎从一个站点移植到另一个站点，避免软件更新的传播等。即使共用一个站点的引擎，由于处理的要求也可能需要在单独的机器上运行；
- 数据格式：不同的引擎往往需要相互矛盾的数据格式。例如，语音到语音翻译可以分解为语音到文本、文本翻译，然后再从文本到语音合成。然而，语音到文本自然输入是一个音频信号，并且其自然输出是把文本与信号的时间域相关联（往往一次一个词），而文本翻译引擎与音频无关，只是期望文本字符串作为输入，典型形式是大于一个词的块。因此，聚合需要在各个引擎的格式之间进行适当的数据转换和重新组织；
- 异常处理：当引擎遇到问题时，它经常以一种无法为聚合中独立研发的引擎所能正确识别的方式报告，这样可能导致错误被忽略，甚至导致聚合失败。异常处理

设施和约定是必要的。比如，处理可以容忍的错误、一个导致可接受的数据损失的错误，以及那些必须终止应用程序的错误等。

本章我们将研究创建的软件框架以应对这些问题，并讨论了几个集合系统实例来执行复杂的任务。

16.2　聚合语音和 NLP 引擎架构的期望属性

软件框架需具备几个基本属性，以便充分实现聚合引擎的优势，在下面的几节中，我们把这些属性归为 4 个方面：模块化的组件、计算效率、数据管理和伸缩性。

16.2.1　灵活的分布式组件化

处理复杂应用程序的基础是需要支持模块化设计。引擎的功能，例如语音到文本转换、翻译引擎和信息抽取，早于那些组合这些功能的复杂应用系统，而且集成执行每个功能的引擎通常是不切实际的。因此，首先要求简单的组件化设计。框架应该被设计成很容易适应在其上运行的现有引擎，这些引擎可由任何常用的编程语言编写并且运行在任何常用的操作系统。这种适应性可能只采用简单的封装形式以使引擎符合一个简单的应用程序编程接口。一旦引擎是可适应的，那么配置一个聚合系统也应该是简单的，只要描述引擎间的处理流程，以及将需要的数据重组织。

一般情况下，组件可以独立开发，因此这个框架必须能够处理异构的操作系统和编程语言，引擎应该不需要转换成一个通用的操作系统和编程语言；相反，该框架应使每个引擎在其各自的本地计算环境里运行。

此外，该框架应允许应用程序定义组件之间的数据接口。一个可定制的、可扩展的数据模型将允许组件更换或升级，并允许新数据格式的新组件很容易地添加。

各个引擎不应该移植到一个共同的地点。随着技术的进步，允许引擎从各自站点中进行操作，以使引擎维护简单化和升级容易。因此框架应该支持远程操作，将每个组件通过网络以可访问的服务方式进行部署，而聚合应用程序则可视作这些分布式引擎的远程客户端。

另外，这些引擎组件对很多应用程序有用，因此聚合框架应该为引擎提供多应用程序客户端服务，例如，可对它们的请求进行排队。应该启用负载平衡，便于长时间运行的多个实例或者有大量请求的引擎可以被预分配以服务这样的队列。

16.2.2　计算效率

为了实现其功能，不同的引擎需要不同数量的上下文。例如，当处理一句话或者一个段落的上下文时，翻译引擎也许运行得最好，而一个新闻主题聚类引擎通常会给整个新闻故事指定一个标签。鉴于这个原因，框架必须能够处理内容的片段，片段的大小必须是应用程序可控制的，其大小可以是多少分钟的音频、多少个字符的文本等。

有效处理这些片段需要一些基本的能力。首先是流水线，在聚合的每个不同阶段具有同时处理多个内容片段的能力。当聚合程序完成第一个片段时，第一个引擎才开始处理第二个片段，而不是当第二个引擎接收到第一个片段后就立即去执行第二个片段，则一个 N 引擎的顺序聚合程序的吞吐量将是次优的，性能降低 N 倍。

通常，一个聚合程序包括一组引擎，其处理是相互独立的。在系统合成组件前使用多重语音到文本的转换和翻译引擎就是一个例子，比如 Rover [7] 或者多引擎机器翻译 [12]。如利用这个机会减少延迟，则需要总体框架并行调用独立的引擎，处理相同的数据

片段，并对并行产生的结果进行重组。

另外一个问题经常会一两个特别的引擎功能，比如语音到文本转换，在一个处理聚合中由于处理时间长而变成明显的瓶颈。对于框架来说，支持部署一个引擎的多个实例服务其处理队列往往是有利的，这样可以一次处理多个数据段，对客户端应用程序透明，从而提高吞吐量。

最后，这个框架应该提供将组件以服务的方式运行的功能，这些服务为客户端应用程序共享。结合前述能力，本质性为不同计算需求的分布式引擎集提供了灵活的动态负载平衡，有效地为分布式的客户端应用程序集服务。

16.2.3 数据操作功能

正如前面所说，不同的数据类型是不同引擎功能的基础。语音到文本和文本到语音的引擎把文本字符串和一段音频信号关联起来，而翻译引擎关联两个不同语言的文本字符串。因此，配置了三个引擎的语音到语言翻译应用程序必须协调两种语言的音频和文本，同时控制翻译引擎不处理音频，并且这两个语音处理引擎每次只处理一种语言。因此，框架必须具有协调不同类型数据的能力，而数据以不同的概念表示单一的一段内容。必须维持数据段中各种录像、音频和文本数据表示的对齐，将数据适当组织以使每个引擎能集中于其适当的表示而忽略其他部分。当添加新的引擎，这种能力必须容易被扩充。

在某些情况下，引擎将会有冲突的分割要求，或者一个引擎的输出会对随后引擎的数据输入确定正确的分割起一定的作用。例如，语音到文本转换或许已经生成文本，而文本作为故事边界检测引擎的输入，以确定各个片段的适当边界，然后再将这些边界传送到一个主题聚类引擎，所有的文本、音频和其他内容表示，将组合起来并重新分割，同时保持它们之间正确的对齐联系。所以，对框架的另一个要求是它能够处理动态的内容再分割。

16.2.4 鲁棒性处理

在不同的引擎里不可避免地会发生异常，而且有时引擎或应用程序客户端与引擎服务器之间的网络连接会失败。为了能够处理这些异常，聚合框架必须具有方便的、灵活的错误处理和流量控制机制。源于引擎处理的异常应该被捕获到，以免引擎的意外输出变成另一个引擎有问题的输入，可能使故障更复杂化。聚合程序的配置必须具有规定标准和处理引擎或连接故障后果的能力。这些状况可能包括超时、放弃重试之前的失败次数以及放弃一个引擎后的操作，例如跳过、调用备用的组件或完全终止处理。

此外，随着远程服务的出现，越来越需要远程监控和管理。一个生命周期管理系统应该提供一种通告机制，它可以对任何远程服务问题进行通告，而且能够从管理控制台启用监视、开始及停止服务。

16.3 聚合的架构

一些现有的架构支持上面介绍的许多属性。下面几节我们将介绍几个流行的、有前景的架构。

注意，一些基于文本的 NLP 工具库仅支持简单的顺序处理，本质上依赖于应用程序开发者实现聚合，例如，OpenNLP、NLTK、Ellogon、OpenCalais、Weka、Kea、OpenCalais、LingPipe、FreeLing。因此这些工具本身对复杂的聚合没有用处，但以下框架已经开发出使这些工具融入更复杂应用程序的封装器。

16.3.1　UIMA

非结构化信息管理体系结构（Unstructured Information Management Architecture，UIMA）是用于创建、发现、创作及部署广泛的多模式分析能力和与搜索技术整合的架构和软件框架。这种架构已经被结构化信息标准促进组织（Organization for the Advancement of Structured Information Standards，OASIS）[15] 作为一个开放标准所接受。

UIMA 允许多个分析引擎以聚合方式结合，并且提供了一个可定制的类型系统，可使不同的引擎在共同的数据结构中共享它们的结果。每个引擎都实现了 UIMA 注释界面，并且传递了以共同分析结构（Common Analysis Structure，CAS）表示的分析数据，这里包含了所有早期注释器产生的数据。UIMA 分析引擎（Analysis Engine，AE）可能是一个单一的注释器，或者多个分析引擎的聚合体，分析引擎间的流量由一个可定制的流量控制器管理。每一个 CAS 包含了一个或多个被分析的数据表示（如一个文本文件、图片、一段音频或视频）以及引擎所添加的元数据（注释）的表示。CAS 还包含一个类型系统的表示和一个可以高效访问类型实例（在文档里可以由位置索引）的索引库。

Apache UIMA 是在 Apache 软件基金会的网站 [4] 上提供的一个开源实现，还提供了以下特色：

- 一个共同的分析结构来组织并保持一段数据（如文本、音频）和其上的所有分析结果；
- 一个用于使输入和输出数据格式规范化的、可扩展的类型系统机制；
- 一个可扩展的基于组件的框架，简化了 UIMA 兼容的分析器的集成和部署；
- 支持 JAVA、C++、Perl、Python 和 Tcl 编写的分析器；
- 支持 Linux、Windows 和 MacOS X；
- 分别用于开发和测试组件的工具；
- 具有把组件作为互联网上的共享服务运行的能力；
- 具有复杂的错误处理选项创建自定义分析流程的能力；
- 具有通过在聚合的不同阶段同时处理多个数据段的能力，以增加吞吐量；
- 具有通过多个引擎并行处理同一片段以减少延迟的能力；
- 重新分割数据的能力。

327

1. 灵活的分布式组件化

Apache UIMA 框架用 Java 实现，但 AE 可以用 Java、C++ 或者脚本语言来编写，例如，Perl、Python、Tcl。AE 可以用一个简单的线性流或用户定义的流来组合成聚合体。Apache ActiveMQ [3]，Java 消息服务（Java Message Service，JMS）的开源实现，提供了与远程服务的通信。

UIMA 的基本数据元素是被分析的数据区域的一个注释。对于文本文档，该区域通常是一个串字符，但其他形式可以是一个音频样本或视频帧序列。UIMA 标注以 TIPSTER 架构 [20] 为基础，并且包含被分析的数据的不可改区域的开始和结束的偏移量。每个组件指定了处理的 CAS 中的数据类型，框架通过合并所有需求为应用程序构成一个完整的类型系统。组件只需要访问在 CAS 的数据子集中匹配它们的定义类型，而且不受 CAS 中其他数据变化的影响。对于远程服务，每个 CAS 中的数据都以一种与平台无关的格式（XML 或二进制格式）进行传输，而且仅返回对 CAS 的修改。

2. 计算效率

AE 的聚合体中，每个 AE 都可以在不同的数据片段上独立（远程或本地）运行，架

得以最小的延迟，处理聚合体中的每个CAS。为进一步提速，一些AE可以并行运行处理相同的CAS，当速度最慢的一个AE运行结束时，再合并它们的结果。由于最慢的AE变成了瓶颈，因此可以部署多个实例以增加吞吐量。对于远程服务，各个实例可以被分配在多处理器上，所有处理器都服务同一个JMS队列，提供负载平衡以及个别故障的鲁棒性。

3. 数据操作性能

一个CAS可能包括多个分析数据或文档的视图。例如，一个语音到语音的应用程序 528
可能会从一个包含一段音频的视图开始，随后加入一个包含由语音到文本的引擎所生成的文本记录的视图，然后视图保存由翻译引擎生成的翻译。每个视图包含它的数据表示，连同注释和索引在内，这种表示为引擎提供一个一致的、自然的接口，并且独立于数据的原始形式。网络爬虫可能开始于一个HTML视图，然后创建一个去掉标签的文本视图，进而创建去掉标签的文本翻译视图。这使得翻译引擎服务能够处理来自音频和Web应用程序的CAS，仅分析包含文字转录或去除了标记的文本。类型系统中的某些类型可以在视图之间提供交叉引用，因此可以保持对齐。

AE通常用一个单一的CAS作为输入并添加其结果，但还可以创建从其输入CAS导出的新CAS。以这种方式，AE可以把分析的数据划分成较小的片段，或者进行复制以便用聚合的不同部分来处理，或者根据数据中检测到的特征把初始CAS序列再分割成一序列。一个应用程序可能首先把一个长音频流分割成较短的固定长度的片段，然后根据在转录文本中检测的边界，再分割成可变长度的片段。

4. 鲁棒性处理

Apache UIMA为每个AE提供了大量的可配置的错误处理选项。错误可能是由于基础设施的问题引起的，如远程服务、连接失败，或诸如无效数据引起的应用程序的问题。在这两种情况下，流量控制器可以决定是否重试，以决定未经AE处理的CAS是否继续，或终止应用程序。如果连接失败是短暂的或者如果服务已经部署了多个AE，则重试是适当的。如果一个远程服务产生许多不可接受的错误，则可以将流量控制器配置为避免流量流向该AE，或者改变为流向另外一个AE。

远程服务使用统计信息可以实现监控，以帮助识别瓶颈或未被充分利用的资源。Apache UIMA也没有一个完整的生命周期管理系统，但一些应用程序已使用外部资源来实现此功能，如IBM WebSpere应用服务器社区版或JCraft（Java下的SSH）。

16.3.2 GATE

GATE（General Architecture for Text Engeering，文本工程的通用架构）[10]是谢菲尔德大学所开发的、用于自然语言处理实验的工具。它包括具有图形界面的开发环境，以及一套由语言和处理资源组成的、可重复使用的组件。它支持一个简单的以JavaBeans实现的标注流水线，这些JavaBeans的文档和标注可用Java对象的特征映射（feature map）来扩充。目前，GATE局限于处理预定义的文本文档语料库，没有远程执行，并且组件顺序执行，可根据数据跳过一些组件。由于UIMA和GATE共享有序、重叠、类型化的标注的类似概念，已经开发出一些封装允许GATE应用程序作为UIMA分析引擎，反之亦然，通过XML映射文件介绍怎样将特定标注进行转换。通过这种方法，GATE应用程序可以得益于UIMA的灵活部署特性，而UIMA应用程序也得益于 529
GATE提供的许多插件。

1. 灵活的分布式组件化

组件接口只支持 Java，因此用其他语言比如 C++、Tcl 编写的组件必须封装。数据类型很容易定制，因为标注是保持感兴趣区域的开始和结束偏移量的 Java 对象，以及另外的涉及其他的标注或 Java 对象的引用。

2. 计算效率

远程执行、流水操作、并行处理和横向扩展都不支持。

3. 数据操作能力

处理仅局限于文本文件。

4. 鲁棒性处理

因为所有的执行过程都是本地的，因此很少有机会能从错误中恢复。

16.3.3 InfoSphere Streams

InfoSphere Streams[11] 是由 IBM 公司提供的商业产品，设计用于多个实时资源信息流的快速分析，提高了不同领域的决策速度和准确性，如保健、天文、制造和金融交易。应用程序被开发为数据流处理图，其中每个处理单元消耗并产生多个事件流，用可用计算资源对图的元素进行自动赋值。处理单元声明操作的每个数据流的名称和类型，并且框架通过把消费者的输入要求与适当的流生产者进行匹配以编制流图。

1. 灵活的分布式组件化

处理单元可能是用 Java 或 C++ 编写的，但是唯一的支持平台是 Linux。组件之间的数据流没有限制：数据流的每种类型都要命名，并且与处理单元的输入和输出流联系起来。任何数据封装类型的缺少都会复杂化组件的共享和重用。

2. 计算效率

由于每个组件是数据驱动的，所以流水操作支持是固有的。当组件消耗相同的流时，就实现了并行处理，分离的输出流输送到“接合”组件，从而合并为一个单一的数据流。流可以被过滤成多个较慢速度的流，在相对缓慢的组件的多个实例中分配工作量，然后合并成单一的数据流。

3. 数据操作功能

由于没有固定的数据模型，因此应用程序负责所有的数据管理。

4. 鲁棒性处理

框架监控每个组件的状态，出现故障时可以重新启动组件或将其移到另一台机器上，重新连接所有的数据流。除非已被声明为“高可用性”组件，否则一些数据可能会丢失。有大量处理资源时，采用可视化工具则有助于优化组件的布局，但是这些必须是相同操作环境下的专用资源。

架构设计的焦点一直是实时数据的高带宽、低延迟处理，比如股市交易、新闻提要、天气数据和 RFID 事件。在应用程序过载的情况下，某个流的一些数据包的丢失是可以接受的。它应该能够处理与音频和文档的 NLP 流相关联的、更大的上下文关键数据包，并可能适合于一些具有较强实时性要求的应用程序。

16.4 案例研究

在以下三个案例研究中，我们描述了不同需求的应用程序（例如，远程与本地处理、

实时响应与批处理、专用与共享的引擎服务)。因为重点是描述与各种聚合场景相关的问题，而不是比较聚合软件框架相关的问题，而且因为三个应用程序都需要 Apache UIMA 支持的最好功能，所以三个案例都讨论 UIMA 聚合。

16.4.1 GALE 互操作性演示系统

大型的、分布式语音和文本处理引擎的聚合的一个例子是互操作性演示 (Interoperability Demo，IOD) [16] 系统，由美国国防高级研究项目局 (Defense Advanced Research Project Agency，DARPA) 所赞助的全球自主语言开发 (Global Autonomous Language Exploitation，GALE) 研究计划所开发的系统。GALE 包含了推进许多语音和文本处理技术的研究，而 IOD 的目标是为了证明在许多 GALE 网站上运行的引擎间的互操作性。UIMA 被选为 IOD 的聚合框架，是因为它非常适合处理语音和文本，以及处理各种计算环境下运行的原有引擎。本节中 IOD 的描述举例证明并解释了如何使用 UIMA 从一组引擎中创建聚合系统。

IOD 由两个应用程序组成，来自美国和欧洲的大学和公司中运行的 15 个引擎聚合而得。其中一个应用程序——IOD-video，采用全部 15 个引擎使得阿拉伯语广播新闻可以浏览英语文本，并通过英语语音合成转换成语音。另一个应用程序，IOD-web，使用相同引擎的一个子集使得阿拉伯语网页文本类似地可用英语浏览和收听。为了实现这一点，IOD 运行各种各样的引擎功能：方言识别 (DID)、性别或说话者检测 (GSD)、语音到文本转换 (STT)、命名实体检测 (ED)、转换为英文的机器翻译 (MT)、能执行系统组合功能的多引擎机器翻译 (MEMT)、故事边界检测 (SBD)、故事的主题聚类 (TC)、产生主题摘要的多文档文摘、故事和话题的标题生成、文本到语音的合成 (TTS)。这些引擎运行在原来的网站上它们本地的操作系统里——Linux 或者 Microsoft Windows，用它们的本地编程语言——C++、Java、Tcl、Perl 或它们的结合物。目前，这些网站包括 IBM [1，8，9，18]、纽约的哥伦比亚大学 [19]、宾夕法尼亚州的卡内基梅隆大学 (CMU) [12，14]、马萨诸塞州的 Raytheun BBN 科技 [5]、德国的亚琛工业大学 [6]、法国的 Systran 公司、在阿默斯特市的马萨诸塞大学 [2]。这些应用程序参见图 16-1。

IOD 每天处理两个阿拉伯新闻网阿拉伯电视台和半岛电视台的大约 4 小时的新闻节目，持续超过 3 年的时间。IOD 的输入由分割成 2 分钟的片段节目组成，选择该持续时间是为音频处理提供足够的上下文，同时避免过度延迟。在处理期间，根据检测到的故事边界，聚合器重新分割该内容。结束时，它输出到浏览器界面，其内容有主题标题的菜单列表和从翻译合成的任意英语音频。点击主题标题可以挖掘到主题摘要、故事提要、实体提及、对齐到视频关键帧的故事翻译。所有这些都是由聚合器生成的。

IOD web 把初始内容作为文本处理，因此跳过了音频处理引擎。这个聚合程序以一个从内容中去除 HTML 标签和相关材料的组件开始。由于大部分处理的网页已经是故事，因此 IOD-web 应用程序也跳过故事边界检测。

1. 功能描述

IOD 引擎在图 16-1 中被描绘成黑体文本框。IOD-video 从确定不同的阿拉伯语方言和说话者性别的时间跨度开始。在已知演讲者的情况下，比如经常出现的主持人和世界领袖，它也确定说话人身份的时间跨度。然后片段被同时传递到若干 STT 引擎，并行产生阿拉伯语讲音的转录。当所有 STT 引擎已经处理完毕，再对结果文本进行阿拉伯语实体检测，然后片段被同时发送到若干 MT 引擎。采用多个 STT 和 MT 引擎有两个原因。一

551

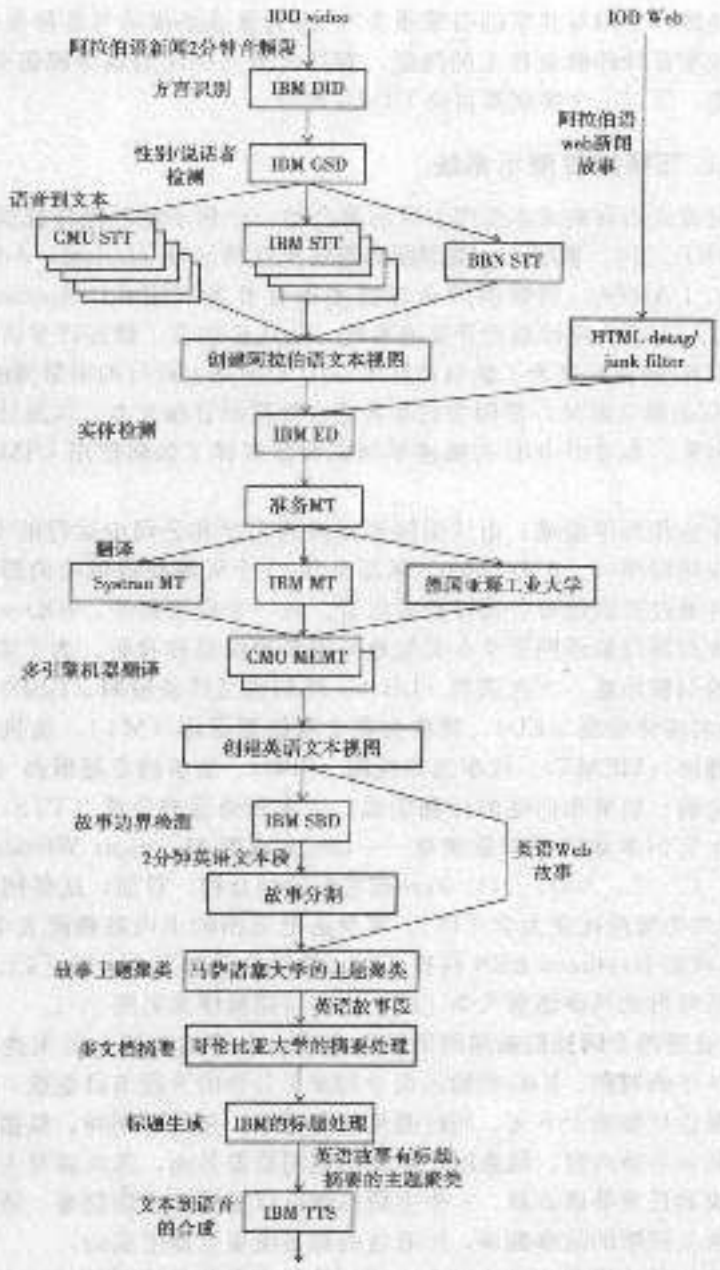


图 16-1　IOD 应用程序的引擎集合系统框图。粗实线箭头表示 IOD video 应用程序的数据流；虚线箭头表示 IOD Web，粗实线箭头表示两者

个是采用 MEMT，即系统组合引擎，从每个 STT-MT 所提供的组合中挖掘出更好的翻译。另一原因是在一个或两个任意类型的引擎失败或网络连接失败的情况下，提供容错机制。故事边界检测利用阿拉伯语转录文本和定时信息，比如 STT 检测的停顿。主题聚类把主题标示符赋给每一个故事，使得被分类为相同主题的故事共享相同的标示符。多文档摘要在每一个故事段上用标记相同主题标示符的所有故事的一个汇总摘要来标注。标题生成引擎添加了来源于故事翻译的标题和另一个来源于摘要的标题，并且文本到语音的合成引擎通过翻译创建了一个音频流。

532
|
533

2. 实现

视图 IOD使用三种视图为每个引擎提供需要的输入数据：

- 一个音频视图，把段内容表示为一系列波形样本；
- 对每个STT引擎的一个阿拉伯语文本视图，基于SST引擎文字转录输出，把段内容表示为阿拉伯语字符序列；
- 一个英文文本视图，典型地基于MEMT引擎的翻译输出，把段内容表示为英文字符序列。

这种安排能够使音频处理引擎只看音频视图，而文本处理引擎只看文本视图。单语引擎只看没有被其他语言内容混淆的相关视图。

类型系统 GALE类型系统（GALE Type System，GTS）[17]的创建是为了NLP引擎的UIMA聚合，比如IOD。为保持各种各样的语音和文本处理引擎的输入和输出，GTS定义适合于音频和文本视图的数据类型。类型系统作为一个通用的数据格式，充当各引擎之间互操作性的基础。

GTS描述代表每个引擎类型输入和输出的固有属性的数据类型。例如，STT引擎自然而然地对内容的音频视图进行操作，也就是大量的语音样本代表了一段时间序列，并且它的自然输出是口语词的转录。GTS提供了一个名为AudioToken[台]的数据类型代表这个输出；此类型提供了一个字符串属性“spelling”，用来保存单词，浮点值属性begin、end和confidence保存音频视图中单词的开始和结束时间，以及识别结果的置信度。一些STT引擎输出使用的另一个GTS类型是SU（sentence unit，句子单元）；它同样有begin和end属性，并且它的存在标志着引擎相信这些时间点括起了一个句子。

同样，对于一个文本处理引擎的自然输入，比如MT引擎，被名为WordToken和Sentence的GTS类型所代表。两者都具有整数属性begin和end，代表组成文本视图的字符数组索引，对于文本处理引擎来说是内容的自然视图。

应当注意，在聚合包含语音到文本引擎紧接着一个文本处理引擎的情况下，AudioTokens和SU很可能分别与WordToken和Sentence对象有一一对应关系。然而，这些相应的数据类型是不一样的。具体而言，语音数据类型面向基于时间的视图，如音频视图，这对于语音处理引擎是合适的，但WordToken和Sentence反映了一个字符串的位置而没有 534
参考时间，同样适合对音频和时间一无所知的文本处理引擎。因此，当聚合这类引擎时，正如16.4.1节所描述的，数据重组组件，在语音到文本之后、文本处理之前调用，以利用由语音到文本引擎已放置在音频视图里的数据创建适当的文本视图和在这些视图里的面向文本的对象类型。此外，GST提供了交叉引用类型，如AudioXref，此引用类型映射到视图中以使相同内容的各种表示保持互相对齐。此类型能够与音频时间同步翻译输出，便于自动的音频字幕。

GST描述了更多数据类型，这些数据类型适合许多其他类型的引擎，比如实体检测、故事边界检测、说话者识别。

注意，类型系统的规范在构成聚合的引擎之间不构成完备的数据“契约”。除了像GTS的共享类型系统，聚合的设计需要描述怎样使用类型系统。这包括一些问题，比如哪个引擎负责创建哪种类型，那种类型和那种属性是必要的还是可选的，例如在IOD中，STT引擎需要生成AudioTokens，而confidence属性是可选的，而且SU类型也是可选的。

台 打字字体用于指示GTS的类型和属性。

使引擎适应 UIMA　由于 IOD 采用的引擎先于聚合，因此它们需要适应 UIMA 框架和 GTS 数据模型。UIMA 有一个简单的 API，该 API 只有一个必需的方法来处理数据段。如果引擎需要特殊的初始化或终止操作，则其他方法可能也要被实现。GTS 为每个引擎功能提供自然的数据类型。因此，实际上只需提供引擎的一个小封装以符合 API 和数据格式。具体地，当已经存在的引擎被 UIMA 封装时，封装器的处理功能通常把表示输入的 GTS 类型转换成引擎规定的格式，运行该引擎，然后把它的输出转换成表示其引擎功能的适当的 GTS 输出类型。这些转换一般是很直接的，因为 GTS 类型就是为反映每个引擎功能固有的输入和输出而设计的。封装一个模拟 STT 引擎以成为一个 AE 的代码如本章最后的 16.7 节所示。

数据重组　如前所述，组装一个包括 STT 和文本处理引擎的应用程序需要在 STT 之后使用数据重组组件，以负责：

1）通过拼接 AudioTokens 中的字符串创建文本视图。

2）创建一组 Sentence 标注，把音频视图中时间跨度的 SU 转换成文本视图的字符跨度。

3）创建 AudioXref，把文本视图中的单词显示对齐到产生它们的 AudioTokens，以便维持时间对齐。

4）在文本视图上创建 WordToken 标注，便于为后续的文本处理引擎所用。

535 用这种方法，聚合程序跨越了 STT 与文本处理之间的不兼容性以及 STT 固有输出与文本处理引擎输入之间的差异。这个组件在图 16-1 中被描述为“创建阿拉伯语文本视图”。注意，包含多个 STT 引擎的聚合将产生多个并行转录，每个转录导致内容的另一个阿拉伯语文本视图。

数据重组组件在概念上与 STT、MT 等“引擎”很不相同，后者往往代表正在进行研究的主题——实验性 NLP 技术，然而数据重组组件会执行更多“机械性的”数据操作任务。然而，就 UIMA 框架而言，这两种组件类型看起来是相同的，都作为 AE 实现。一个用来实现数据重组组件最后一步的 AE 代码附加于本章最后部分的 16.7 节。

类似地，在图 16-1 中非粗体框表示的是其他数据重组组件也是需要的，用于在 IOD 的引擎之间的接口起到类似作用。一个类似于前文所述组件从 MT 和 MEMT 在阿拉伯语文本跨度上标注的英语翻译字符串上创建英语文本视图，并且在英语和阿拉伯语之间映射标注，如命名实体，这是有必要的，因为翻译引擎内在地处理多种语言，并由此生成 TranslationResult 的 GTS 对象，TranslationResult 把一种语言文本视图中的一串字符用另一种语言的字符串来标注。许多其他文本处理引擎一次处理一种语言，因此处理 MT 输出的引擎需要目标语言的视图，因此“创建英语文本视图”数据重组组件建立一个英语文本视图，为之后的引擎在不知道阿拉伯语的情况下处理英语提供服务。

在 MT 之前，另一个数据重组组件“准备 MT”创建另外的 GTS 数据类型 Translatable 指定需一一翻译的跨度。目前，这些仅是 Sentence，但也可以由一些其他算法定义，如把文本字符串合成更长的单元以使 MT 引擎可受益于更多的上下文，或分解成更短的单元以便专门的翻译引擎处理，如名字翻译引擎。另外一个组件“故事分割”根据在其之前的引擎检测的故事边界把内容重新分割成故事片段。“故事分割”创建新的 CAS，其数据元素对应于原来的 2 分钟片段，但是根据新的边界重新索引。这个重组组件服务于之后的引擎，这些引擎的自然输入是故事，比如主题聚类、摘要、标题生成。

最后，两个组件没有画出来，但前后穿插了整个过程，一个是在聚合开始前读取输入数据的集合并创建 CAS，另一个是在聚合的最后从 CAS 中提取数据转换成应用程序所需

的格式。在 UIMA 中它们分别被称为数据集读者组件和 CAS 消费者组件。

上下文相关处理 IOD 的片段以两种不同的方式处理需要该片段之外的上下文。一种方式涉及紧紧围绕的片段。故事边界检测模型为 6 分钟窗口的中间 2 分钟提供输出。给定 IOD 的 2 分钟片段，这个引擎因此必须缓冲一个片段，仅在收到 $N+1$ 段后在 N 段的一个节目上生成它的输出，这样该输出才能基于由片段 $N-1$、N、$N+1$ 组成的 6 分钟窗口。

其他上下文依赖性包括主题聚类和摘要。必须维持与每个主题聚类相关的、以往内容 536
的历史，这样才可以为当前的片段赋以适当的聚类，为主题内容生成一个累积的摘要信息。根据客户端应用程序实例，这两个引擎维护的历史应该被隔离，从而保证不同用户的历史不会混在一起。

计算效率 如前所述，IOD 调用多个引擎，在同一片段上并行处理相同的功能，因为这些引擎对同样的输入类型产生同样的输出类型，因此相互没有依赖。IOD 也可以一次性将多个片段通过聚合传输，一旦第一个片段经第一个引擎处理完，第二个片段马上进入第一个引擎，而不是等待第一个片段退出整个聚合之后。

除了并行调用引擎和多个片段的流水线处理技术，IOD 还利用了 UIMA 对于配置控制流的其他特征。可配置聚合器为每种引擎服务处理超时，以及超时出现时决定如何采取下一步动作。例如，在某个 STT 或实体识别引擎因本身失效或网络连接发生故障时，因为有其他 STT 引擎，聚合系统会继续运行。作为另一个例子，假如自动文摘引擎无法正常工作，IOD 会启动一个简单的备份组件，把所有同一主题的文本翻译拼接起来，这虽然不能取代文本摘要，但是可以为后续的引擎提供可信的输入，以便为该主题生成标题。当多引擎翻译服务失效时，另一种流标准适用。在这种情况下，一个 STT-MT 组合产生的翻译结果可以作为首选的翻译使处理继续进行。然而，当故事边界检测失败时，由于系统没有后备提供后续处理所需要的功能，应用终止。

实际上，一个引擎功能——STT，比其他引擎需要更多的计算。为了减轻对吞吐量的影响，部署了两个 STT 引擎的多个实例，以服务于这些引擎的客户任务队列，如图 16-1 中层叠框所示。对于 IOD-web，多引擎机器翻译（MEMT）是瓶颈问题，因此相应地部署了两个 MEMT 引擎实例。

IOD 利用了 UIMA 提供的引擎服务能力使其为多种应用程序所共享。IOD-video 和 IOD-web 同时对 IOD 的许多引擎服务的请求进行排队。这种能力和多实例部署能力是为使用一系列引擎服务的大量应用程序的可伸缩部署进行动态负载均衡的关键。

3. 灵活的应用程序构建

除了 IOD-video 和 IOD-web 外，一个基于 GUI 的应用程序配置工具——UIMA 组件容器（UIMA Component Container，UCC），已经部署于卡内基梅隆大学。UCC 允许用户 537
上传数据并配置 IOD 引擎的聚合来处理上传的数据。UCC 通过添加必需的数据重组组件来自动完成聚合。

16.4.2 跨语言自动语言开发系统

TALES（Translingual Automated Language Exploitation System）是 IBM 公司开发的类似于 IOD 的聚合系统也用 UIMA 实现，集成了语音识别、信息、翻译等功能。但 TALES 的聚合场景有所不同，它操作在同一地理位置的机器集群，处理多语言，它接近于一个生产系统，包括部分操作的实时需求。

TALES 是一个集成了视频处理和 Web 处理的聚合器，参见图 16-2。TALES 聚合的

引擎包括实体识别、其他语种到英语的翻译，以及用于各种类型的数据搜索、浏览以及监控等的设备。除此以外，TALES 的视频处理聚合器包括 STT、性别和说话人识别、语言/方言检测以及英语文本到语音合成等。已经部署的 TALES 的几个实例，机器的数目不同，处理的视频频道数目也不同，处理的语言数目也不同，如阿拉伯语、汉语、西班牙语和英语等。



图 16-2　TALES 聚合系统图

TALES 的最高优先级保持与输入视频流同步，通过调节 STT 模型使得每个 2 分钟的视频片段所需的处理时间实现低于 2 分钟，并且对于每个视频频道都有专门的聚合器和处理硬件。机器翻译（MT）是个特例，由于语言模型的庞大，机器翻译的若干实例在多个

聚合系统间共享，Web处理聚合器的请求处于次要位置，因为视频处理要求实时性。

TALES运行在无须外界网络连接的环境中，仅需要提供视频流与一个网络爬虫。TALES聚合器可终止于若干个组件，使得对输出进行多种使用。如：

- 分段浏览当前视频节目，可使用户播放视频时选择英文字幕、方言/说话者标注、实体高亮显示、英语语音合成配音代替原始音频；
- 处理内容的英文关键词搜索包括根据日期、原始语言、来源是视频还是Web限制搜索；
- 警报，用户发出一个关键词查询，系统在处理匹配查询的新内容后，以用户指定的方式向用户发出警报，比如通过电子邮件。

16.4.3 实时翻译服务

IBM的实时翻译服务（Real Time Translation Services，RTTS）实现双向的、自由形式语音翻译，帮助那些不使用相同语言的人们进行沟通。对于会话的一个参与方的每一句话，采用三个引擎：STT、MT、TTS，向另外一方提供翻译的语音。因此，虽然IOD和TALES的优先考虑分别是分布式处理和吞吐量，以与到来的视频流保持一致，RTTS的优先权是处理多个同时的低延迟任务。RTTS通过部署作为UIMA服务的每个引擎类型的集群来完成该任务。每一个呼叫由一个UIMA客户端处理，该客户端发送请求到可以提供所需服务的JMS队列。因为应用程序直接地实现了流水线式的服务，所以它可以很容易监控翻译的进度，处理任何因过载服务引起的延时。这同样也可以通过使用一个UIMA定制的流量控制器来完成。

16.5 经验教训

使用UIMA配置的NLP应用程序的经验已经揭示出了一些这类聚合引擎固有的问题。解决接下来讨论的问题是有益的。

16.5.1 分割涉及延迟和精度之间的权衡

处理音频或视频时，我们已逐步向处理2分钟片段的做法靠拢。每个分割段限制了可被引擎利用以提高精度的上下文数量。然而，当引擎完成一段后，下一个引擎才能开始工作时，长片段会导致高延迟。

在框架上增加一个广义的“流”操作模式，操作可能改善。不作明确的分割，数据可以以任意小的增量从一个组件传递到下一个组件，而组件本身要为足够的上下文工作负责，因此其输出总是滞后于输入。这样的变化将为框架增加很大的复杂性，特别是并行调用多个引擎后管理数据收集，以及在单引擎服务上部署多个实例时调和这种能力。

16.5.2 联合优化与互操作性

引擎功能的联合优化可以产生更高的精度。比如，整合一个STT引擎和一个翻译引擎的语言模型可以减少不能翻译词语的出现。引擎更紧密的结合，一般来说可以增加准确性，降低引擎必须做出“艰难”决策的程度以限定接下来的引擎所要考虑的假设空间。

然而，大量引擎的聚合的互操作性需要一个共享数据模型，对于每个引擎功能，要有一种约定规定数据输入和输出的标准特性。这样的公分母模型将倾向于导致引擎间相对薄的数据流。比如，选择文本字符串格式作为语音到文本引擎的标准输出和翻译引擎的输入，这比选择假设的转录文本这种格式更容易。联合优化的一些好处必须丢弃以使互操作

538
7
539

性可行。一个站点的 STT 引擎和 MT 引擎可能需要协调，然而互操作性使得每个功能可以用多个引擎，以便得到系统组合策略的好处。

16.5.3　数据模型需要使用约定

对于数据传送，聚合基本上需要两层模型。为了使聚合程序运行，一组公认的数据类型是必要的，比如 GTS 那样的一个 UIMA 类型系统。如前所述，类型系统实际上是在聚合中协调引擎的 API 数据组件，以便引擎可以通过一个共同的数据格式进行交流。

540

然而，进一步描述各种格式数据类型的应用约定还是很有必要的。一个共识是，哪个数据类型必须或者可选地被每个组件创造以及如何用数据格式表示各种异常的情况必须达成一致。独立开发的引擎常在标准输出流上用局部约定来编码异常情况，比如用某个文本码对一个未知的单词编码，如果处理不小心，它在聚合中可能会被随后的引擎误解为不同的单词。

16.5.4　性能评估的挑战

对这类聚合的准确率进行定量评估因为多个因素而变得复杂。其中一个因素是简单地对引擎进行各种组合。由 10 个引擎功能组成的集合中，得出的潜在聚合数范围是 45～1 013，这取决于引擎间的相互依赖[⊖]。因此，用 STT 引擎转录的翻译故事的主题聚类的准确率估计区别于文本源的翻译故事的主题聚类等，需要很多的评价标准。

一个相关的挑战是获取真实数据用来和大的聚合结果做比较的复杂性。一般来说，每个不同的聚合需要自己的用于评价的参考语料库，这是一项繁重的任务。

最后，一些引擎功能，即使孤立，也缺乏明确定义的评价准则。例如，摘要和主题聚类算法的评估必定有一定程度的主观性。

对于用来聚合的引擎各方面准确性的评估，研究者们已经开始努力定义一个形式化方法，目标是形成量化评估准则，以评价各种引擎对聚合中整体错误率的贡献。

16.5.5　引擎的前向波训练

理想地，每个统计引擎在运行时间接收到的典型数据上进行训练。在聚合的情况下，典型引擎的输入是聚合中前面的若干引擎的输出。由于这个原因，当引擎更新时，无论是由于新的算法、新的训练数据，或者改变的模型格式，任何导致准确率的改进在聚合中都有失去的风险，因为引入了其输出数据和后续引擎过时的训练条件间的失配。因此，聚合的正确率取决于引擎的前向波训练。理想的是，第一个引擎更新，然后为后续的引擎产生新的输入数据，这些引擎用那些数据重新训练，通过那些引擎的聚合运行，又为后续的引擎产生新的训练数据，依此类推。

541

16.6　总结

语音和文本处理算法已经进展到这样一种程度，即尽管引擎间有混合错误，但迥然不同的引擎功能，如语音到文本、翻译、命名实体检测、文本到语音，以及其他专用的信息

⊖ 如每个引擎 $N=1, \cdots, 10$ 被限制为它前面必须有 $N-1$ 个引擎成成为聚合的开始，并且后面必须有 $N+1$ 个引擎成结束聚合，那么可能的聚合的数目是开始和结束引擎的选择个数 $=10(10-1)/2=45$。另一极端的计算方式是，如果引擎之间没有任何互相依赖关系，并且每个引擎可选择是否包括在聚合中，那么一共可以有 $2^{10}-10-1=1013$ 中聚合，即每个引擎是否包括在聚合中的情形，减去只有一个引擎（因此不成为聚合）的 10 种情形，减去 0 个引擎的 1 种情形。

抽取处理器，可形成大规模的聚合，提供有用的输出。引擎的聚合开启了跨语言 NLP 应用程序的大门。为了使存在的引擎能灵活聚合，需要像 UIMA 那样的软件架构来提供异构的计算环境、互联网的远程操作，以及多种应用程序客户端请求的管理，请求由引擎服务的多个实例相从运行。它也需要数据表示的共享约定以及重组数据的组件，以使聚合中的引擎可以处理前面引擎输出的数据。这样一个平台使得复杂的、分布式的任务可以通过分布式引擎的单点调用实现，分布式引擎在其原来的环境中运行，便于由其作者维护和改进。偶准确率需要引擎升级时进行协调，以使每个引擎保持和所处理数据的类型一致，然而，聚合准确率的正式评估处于初级阶段，因为建立合适的评估准则和测试语料库是一大挑战。虽说如此，但越来越多的应用已经在运行，从多研究室原型到实时部署系统都有涉及。

16.7 UIMA 样本代码

几个标注器作为 Apache UIMA 的一部分被打包在 http://uima.apache.org 的 UIMA 沙盒中；其他的可能在 CMU 管理的 UIMA 组件库中找到：http://uima.lti.cs.cmu.edu/UCR。

下面是一个简单的执行部分数据重组的分析引擎的实现，该引擎在 16.4.1 节中进行了描述。

```
import java.util.Iterator;
import java.util.regex.Matcher; import
java.util.regex.Pattern;

import org.apache.uima.analysis_component.JCasAnnotator_ImplBase;
import org.apache.uima.analysis_engine.AnalysisEngineProcessException;
import org.apache.uima.cas.CASException;
import org.apache.uima.jcas.JCas;
import org.gale.WordToken;

/**
 * Tokenizes all Transcription views creating                                    542
 * whitespace-delimited WordToken annotations
*/

public class TokenizeMI extends JCasAnnotator_ImplBase {

  public void process(JCas aJcas) throws
      AnalysisEngineProcessException {

    Pattern p = Pattern.compile("\\S+");
    try {
      Iterator<JCas> viewItr = aJcas.getViewIterator("SourceText");
      while (viewItr.hasNext()) {
        JCas view = viewItr.next();
        Matcher m = p.matcher(view.getDocumentText());
        while (m.find()) {
          (new WordToken(view, m.start(), m.end())).addToIndexes();
        }
      }
    } catch (CASException e) {
      throw new AnalysisEngineProcessException(e);
    }
  }
}
```

所有的 UIMA 标注器必须有一个 XML 组件描述器，描述接口、名称、数据类型和任何所需的参数。上述标注器的简单 XML 组件描述器如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<analysisEngineDescription
  xmlns="http://uima.apache.org/resourceSpecifier">
 <frameworkImplementation>org.apache.uima.java
 </frameworkImplementation>
 <primitive>true</primitive>
 <annotatorImplementationName>org.gale.pipe.TokenizeMT
 </annotatorImplementationName>
 <analysisEngineMetaData>
   <typeSystemDescription>
     <imports>
       <import name="GaleTokenTypes"/>
     </imports>
   </typeSystemDescription>
  </analysisEngineMetaData>
</analysisEngineDescription>
```

下面的代码表示现有的 STT 引擎如何被封装，以创建分析引擎，处理由 URL 定义的音频，并注释音频视图（其中每个解码的单词都有 AudioToken），参见 16.4.1 节：

```
package org.gale.gus;

import java.io.BufferedInputStream;
import java.util.ArrayList;
import org.apache.uima.UimaContext;
import org.apache.uima.analysis_component.JCasAnnotator_ImplBase;
import org.apache.uima.analysis_engine.AnalysisEngineProcessException;
import org.apache.uima.cas.CASException;
import org.apache.uima.jcas.JCas;
import org.apache.uima.resource.ResourceInitializationException;
import org.apache.uima.util.Level;
import org.apache.uima.util.Logger;

import org.gale.AudioToken;
import org.gale.SU;

/**
 * Demo STT annotator
 */

public class DemoSTT extends JCasAnnotator_ImplBase {

  private Logger logger;
  private String compId;

  public void initialize(UimaContext aContext) throws
     ResourceInitializationException {

    super.initialize(aContext);
    logger = aContext.getLogger();
    compId = (String) aContext.getConfigParameterValue("ComponentId");
  }

  public void process(JCas jcas) throws AnalysisEngineProcessException {

    try {
      jcas = jcas.getView("Audio");
    } catch (CASException e) {
      throw new AnalysisEngineProcessException(e);
    }
```

```
    logger.log(Level.INFO, compId + ": Processing audio URL '"
                           + jcas.getSofaDataURI() + "'");

    String audioMimeType = jcas.getSofaMimeType();
    BufferedInputStream audioStream = new
      BufferedInputStream(jcas.getSofaDataStream());

    // Run a pretend STT that puts its results in two arrays
    ArrayList<String> words = new ArrayList<String>(100);
    ArrayList<Float> endTimes = new ArrayList<Float>(100);
    runSTT(audioStream, audioMimeType, words, endTimes);

    // Get the STT results and add AudioTokens to CAS
    float time = 0;
    for (int i = 0; i < words.size(); ++i) {
      AudioToken atok = new AudioToken(jcas);
      atok.setSpelling(words.get(i));
      atok.setBegin(time);
      time = endTimes.get(i);
      atok.setEnd(time);
      atok.setComponentId(compId);
      atok.addToIndexes();
    }

    // Add one SU spanning all of the audio
    SU su = new SU(jcas);
    su.setBegin(0);
    su.setEnd(time);
    su.setComponentId(compId);
    su.addToIndexes();
  }


  // Demo code pretending to perform STT

  private void runSTT(BufferedInputStream in, String mimeType,
          ArrayList<String> words, ArrayList<Float> endTimes) {
    logger.log(Level.INFO,
               "runSTT: pretending to process audio ... creating 2 fake words");
    words.add("hello");
    endTimes.add(0.65f);
    words.add("world");
    endTimes.add(1.38f);
  }

}
```

544

其描述文件包含用于确定 CAS 中条目创建者的参数。

```
<?xml version="1.0" encoding="UTF-8"?>
<analysisEngineDescription
   xmlns="http://uima.apache.org/resourceSpecifier">
  <frameworkImplementation>org.apache.uima.java
  </frameworkImplementation>
  <primitive>true</primitive>
  <annotatorImplementationName>org.gale.gus.DemoSTT
  </annotatorImplementationName>
  <analysisEngineMetaData>
    <configurationParameters>
      <configurationParameter>
        <name>ComponentId</name>
```

```
    <description>Name of STT engine</description>
    <type>String</type>
    <mandatory>true</mandatory>
   </configurationParameter>
  </configurationParameters>
  <configurationParameterSettings>
   <nameValuePair>
    <name>ComponentId</name>
    <value>
        <string>STTx</string>
    </value>
   </nameValuePair>
  </configurationParameterSettings>
  <typeSystemDescription>
   <imports>
    <import name="GaleSpeechTypes"/>
   </imports>
  </typeSystemDescription>
 </analysisEngineMetaData>
</analysisEngineDescription>
```

下面的 XML 描述代码例子中用到的 GTS 类型，Apache UIMA 的 SDK 包含 Eclipse 插件，可方便创建并开发分析引擎以及类型系统描述器。

```
<typeDescription>
 <name>org.gale.WordToken</name>
 <description>A basic unanalyzed word
 </description>
 <supertypeName>org.gale.NonWhiteSpaceToken</supertypeName>
</typeDescription>

<typeDescription>
 <name>org.gale.NonWhiteSpaceToken</name>
 <description>A span of characters that meet the Unicode
  definition of non-whitespace.
 </description>
 <supertypeName>org.gale.Token</supertypeName>
</typeDescription>

<typeDescription>
 <name>org.gale.Token</name>
 <description>Tokenizer output - these should be
  non-overlapping.  Frequently the set of Tokens
  will cover the entire document, but this is not
  required.  The type hierarchy derived from Token
  is used purely for constructing specific iterators,
  not for data inheritance.
  </description>
  <supertypeName>uima.tcas.Annotation</supertypeName>
 </typeDescription>

 <typeDescription>
  <name>org.gale.AudioToken</name>
  <description>Word-like units</description>
  <supertypeName>org.gale.AudioSpan</supertypeName>
  <features>
   <featureDescription>
    <name>spelling</name>
    <description>Spelling of the word; typically does not
```

```
      include capitalization, optional diacritics, or
      punctuation</description>
    <rangeTypeName>uima.cas.String</rangeTypeName>
   </featureDescription>
   <featureDescription>
    <name>confidence</name>
    <description>Value representing the "score" of this AudioToken, such
 as the probability that the span actually contains the annotated
     word spoken within.
    </description>
    <rangeTypeName>uima.cas.Float</rangeTypeName>
   </featureDescription>
  </features>
</typeDescription>

<typeDescription>
  <name>org.gale.SU</name>
  <description>Sentence-like units.  An SU spans one or more AudioTokens.</description>
  <supertypeName>org.gale.AudioSpan</supertypeName>
</typeDescription>

<typeDescription>
  <name>org.gale.AudioSpan</name>
  <description>The basic unit of a time duration (similar to an Annotation). This is
   a base class that should not be instantiated.</description>
  <supertypeName>uima.cas.TOP</supertypeName>
  <features>
    <featureDescription>
      <name>begin</name>
      <description>Begin time in seconds from the beginning of the segment</description>
      <rangeTypeName>uima.cas.Float</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>end</name>
      <description>End time in seconds from the beginning of the segment</description>
      <rangeTypeName>uima.cas.Float</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>componentId</name>
      <description>ID of the STT component that created this annotation</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
```

547

参考文献

[1] Y. Al-Onaizan and K. Papineni, "Distortion models for statistical machine," in *Proceedings of the Association for Computational Linguistics*, pp. 529–536, 2006.

[2] J. Allan, S. Harding, D. Fisher, A. Bolivar, S. Guzman-Lara, and P. Amstutz, "Taking topic detection from evaluation to practice," in *Proceedings of the Hawaii International Conference on System Sciences*, 2005.

[3] Apache ActiveMQ, http://activemq.apache.org

[4] Apache UIMA, http://uima.apache.org

[5] http://bbn.com/products_and_services/bbn_broadcast_monitoring_system/; BMS includes BBN's AMC STT engine.

[6] O. Bender, E. Matusov, S. Hahn, S. Hasan, S. Khadivi, and H. Ney, "The RWTH Arabic-to-English spoken language translation system," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, pp. 396–401, 2007.

[7] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, pp. 347–354, 1997.

[8] R. Florian, H. Hassan, A. Ittycheriah, H. Jing, N. Kambhatla, X. Luo, N. Nicolov, and S. Roukos, "A statistical model for multilingual entity detection and tracking," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2004.

[9] M. Franz and J.-M. Xu, "Story segmentation of broadcast news in Arabic, Chinese and English using multi-window features," in *Proceedings of the International ACM Special Interest Group on Information Retrieval (SIGIR) Conference*, 2007.

[10] GATE, http://gate.ac.uk/

[11] InfoSphere Streams, http://www.ibm.com/software/data/infosphere/streams/

[12] S. Jayaraman and A. Lavie, "Multi-engine machine translation guided by explicit word matching," in *Proceedings of the 10th Annual Conference of the European Association for Machine Translation*, pp. 143–152, 2005.

[13] U. Murthy, J. F. Pitrelli, G. Ramaswamy, M. Franz, and B. L. Lewis, "A methodology and tool suite for evaluation of accuracy of interoperating statistical natural language processing engines," in *Proceedings of the Annual Conference of the International Speech Communication Association*, 2008.

[14] M. Noamany, T. Schaaf, and T. Schultz, "Advances in the CMU/InterACT Arabic GALE transcription system," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2007.

548 [15] UIMA as OASIS Standard, http://www.oasis-open.org/committees/uima

[16] J. F. Pitrelli, B. L. Lewis, E. A. Epstein, M. Franz, D. Kiecza, J. L. Quinn, G. Ramaswamy, A. Srivastava, and P. Virga, "Aggregating distributed STT, MT, and information extraction engines: The GALE Interoperability-Demo System," in *Proceedings of the Annual Conference of the International Speech Communication Association*, 2008.

[17] J. F. Pitrelli, B. L. Lewis, E. A. Epstein, J. L. Quinn, and G. Ramaswamy, "A data format enabling interoperation of speech recognition, translation and information extraction techniques: The GALE type system," in *Proceedings of the Annual Conference of the International Speech Communication Association*, 2008.

[18] G. Saon, D. Povey, and G. Zweig, "Anatomy of an extremely fast LVCSR decoder," in *Proceeding of the 9th European Conference on Speech Communication and Technology*, 2005.

[19] B. Schiffman, A. Nenkova, and K. McKeown, "Experiments in Multidocument Summarization," in *Proceedings of the Human Language Technologies Conference*, 2002.

549 / 550 [20] TIPSTER Text Program, http://www.itl.nist.gov/iaui/894.02/related_projects/tipster/overv.htm, including R. Grishman, *TIPSTER Architecture Design Document Version 2.3*, Technical report, DARPA, 1997; see http://www.itl.nist.gov/iaui/894.02/related_projects/tipster/

索　引

索引中的页码为英文原书页码，与书中边栏标注的页码一致。

. (period), sentence segmentation markers (句号，句子分割标记)，30
"" (Quotation marks), sentence segmentation markers (引号，句子分割标记)，30
! (Exclamation point), as sentence segmentation markers (感叹号，句子分割标记)，30
? (Question mark), sentence segmentation markers (问号，句子分割标记)，30
80/20 rule (vital few) (80/20 法则，能者多劳)，14

a priori models, in document retrieval (先验模型，文档检索)，377
Abbreviations, punctuation marks in (简写中的标点符号)，30
Absity parser, rule-based semantic parsing (Absity 分析器，基于规则的语义分析)，122
Abstracts (摘要)
  in automatic summarization (自动文摘)，397
  defined (定义)，403
Accumulative vector space model, for document retrieval (累计型向量空间模型，用于文档检索)，374-375
Accuracy, in QA (准确率，QA)，462
ACE (自动内容抽取)，参见 Automatic content extraction (ACE)
Acquis corpus (Acquis 语料库)
  for evaluating IR systems (用于信息检索系统评价)，390
  for machine translation (用于机器翻译)，358
Adequacy, of translation (忠实度，翻译)，334
Adjunctive arguments, PropBank verb predicates (辅查元，ProBank 动词谓词)，119-120
AER (Alignment-error rate) (对齐错误率)，343
AEs (Analysis engines), UIMA (分析引擎，UIMA)，527
Agglutinative languages (黏着语)
  finite-state technology applied to (应用有限状态技术)，18
  linear decomposition of words (词的线性分解)，192
  morphological typology and (词法类型学)，7
  parsing issues related to morphology (词法学相关的分析问题)，90-91
Aggregate processor, combining NLP engines (聚合处理器，融合 NLP 引擎)，523
Aggregation architectures, for NLP (聚合架构，用于 NLP)，参见 Natural language processing (NLP), combining engines for
  GATE，529-530
  InfoSphere Streams (InfoSphere 流)，530-531
  overview of (概述)，527
  UIMA，527-529
Aggregation models, for MLIR (聚合模型，用于 MLIR)，385
Agreement features, of coreference models (一致特征，共指模型)，391
Air Travel Information System (ATIS) (空旅信息系统)
  as resource for meaning representation (作为意义表示资源)，168
  rule-based systems for semantic parsing (基于规则的语义分析系统)，150
  supervised systems for semantic parsing (有监督的语义分析系统)，150-151
Algorithms (算法)，参见各种类型
Alignment-error rate (AER) (对齐错误率)，343
Alignment, in RTE (对齐，RTE)
  implementing (实现)，233-246
  latent alignment inference (潜在对齐推理)，247-248
  learning alignment independently of entailment (独立于蕴涵学习对齐)，244-245

leveraging multiple alignments (利用多对齐), 245
modeling (建模), 226
Allomorphs (变体词素), 6
"almost-parsing" language model ("近似句法分析"语言模型), 181
Ambiguity (歧义)
disambiguation problem in morphology (形态消歧问题), 91
in interpretation of expressions (表达式解释), 10-13
issues with morphology induction (形态归纳问题), 21
PCFGs and, 83-85
resolution in parsing (句法分析中的歧义消解), 80
sentence segmentation markers and (句子分割标记), 30
structural (结构), 89
in syntactic analysis (语法分析中), 61
types of (类型), 8
word sense and (词义), 参见 Disambiguation systems, word sense
Analysis engines (AEs), UIMA (分析引擎, UIMA), 527
Analysis, in RTE framework (分析, RTE框架)
annotators (标注器), 219
improving (改进), 248-249
multiview representation of (多视图表示), 220-222
overview of (概述), 220
Analysis stage, of summarization system (分析阶段, 文摘系统)
building a summarization system and (建造文摘系统), 421
overview of (概述), 400
Anaphora resolution (指代消解), 参见 Coreference resolution
automatic summarization and (自动文摘), 398
cohesion of (衔接), 401
multilingual automatic summarization and (多语自动文摘), 410
QA architectures and (QA架构), 438-439
zero anaphora resolution (零指代消解), 248, 444
Anchored speech recognition (锚点语音识别), 490
Anchors, in SSTK (锚点, SSTK), 246
Annotation/annotation guidelines (标注/标注指南)
entity detection and (实体检测), 293
in GALE, 478
Penn Treebank and (宾州树库), 87-88
phrase structure trees and (短语结构树), 68-69
QA architectures and (QA架构), 439-440
in RTE, 219, 222-224
snippet processing and (片段处理), 485
for treebanks (用于树库), 62
of utterances based in rule-based grammars (基于规则文法的语句), 502-503
of utterances in spoken dialog systems (口语对话系统的语句), 513
Answers, in QA (答案, QA)
candidate answer extraction (候选答案抽取), 参见 Candidate answer extraction, in QA
candidate answer generation (候选答案生成), 参见 Candidate answer generation, in QA
evaluating correctness of (评估正确性), 461-462
scores for (分值), 450-453, 458-459
scoring component for (评分组件), 436
type classification of (类型分类), 440-442
Arabic (阿拉伯语)
ambiguity in (歧义), 11-12
corpora for relation extraction (关系抽取语料库), 317
distillation (提炼), 479, 490-491
EDT and, 286
ElixirFM lexicon (ElixirFM 词典), 20
encoding and script (编码和字体), 368
English-to-Arabic machine translation (英阿机器翻译), 116
as fusional language (作为屈折语), 8
GALE IOD and, 532, 534-536
IR and (信息检索), 371
irregularity in (不规则性), 8-9
language modeling (语言建模), 189-191, 193
mention detection experiments (提及检测实验), 294-296
morphemes in (词素), 6
morphological analysis of (形态分析), 191
multilingual issues in predicate-argument structures (谓词-论元结构的多语问题), 146-147
polarity analysis of words and phrases (词和短语的极性分析), 269
productivity/creativity in (生产性/创造性), 15
regional dialects not in written form (非书写形式的方言), 195
RTE in, 218
stem-matching features for capturing morphological

similarities (增强形态相似度的词干匹配特征), 301
TALES case study (TALES 案例研究), 438
tokens in (词元), 4
translingual summarization (跨语际文摘), 398-399, 424-426
unification-based models (基于合一的模型), 19
Architectures (架构)
aggregation architectures for NLP (NLP 聚合架构), 527-529
for question answering (QA) (用于问答系统), 435-437
of spoken dialog systems (口语对话系统), 505
system architectures for distillation (用于提炼的系统架构), 488
system architectures for semantic parsing (用于语义分析的系统架构), 101-102
types of EDT architectures (EDT 架构类型), 286-287
Arguments (论元)
consistency of argument identification (论元识别一致性), 323
event extraction and (事件抽取), 321-322
in GALE distillation initiative (GALE 提炼计划), 476
in RTE systems (RTE 系统), 220
Arguments, predicate argument recognition (论元, 谓词-论元识别)
argument sequence information (论元序列信息), 137-138
classification and identification (分类和识别), 139-140
core and adjunctive (核心论元和附加论元), 119
disallowing overlaps (不允许重叠), 137
discontiguous (非连续), 121
identification and classification (识别和分类), 123
noun arguments (名词论元), 144-146
ART (artifact) relation class (人造物关系类), 312
ASCII
as encoding scheme (作为编码方式), 368
parsing issues related (相关句法分析问题), 89
Asian Federation of Natural Language Processing (亚洲自然语言处理联盟), 218
Asian languages (亚洲语言), 参见各个亚洲语言
multilingual IR and (多语信息检索), 385, 390
QA and (问答系统), 434, 437, 455, 460-461, 466
Ask.com, 435
ASR (automatic speech recognition) (自动语音识别)
sentence boundary annotation (句子边界标注), 29
sentence segmentation markers (句子分割标记), 31
ASSERT (Automatic Statistical SEmantic Role Tagger) (自动统计语义角色标注器), 147, 447
ATIS, 参见 Air Travel Information System (ATIS)
Atomic events, summarization and (原子事件, 文摘), 418
Attribute features, in coreference models (属性特征, 共指模型), 301
Automatic content extraction (ACE) (自动内容抽取)
coreference resolution experiments (共指消解实验), 302-303
event extraction and (事件抽取), 320-321
mention detection and (提及检测), 287, 294
relation extraction and (关系抽取), 311-312
in Rosetta Consortium distillation system (Rosetta 协会提炼系统), 480-481
Automatic speech recognition (ASR) (自动语音识别)
sentence boundary annotation (句子边界标注), 29
sentence segmentation markers (句子分割标记), 31
Automatic Statistical SEmantic Role Tagger (ASSERT) (自动统计语义角色标注器), 147, 447
Automatic summarization (自动文摘)
bibliography (参考文献), 427-432
coherence and cohesion in (连贯和衔接), 401-404
extraction and modification processes in (抽取和修改过程), 399-400
graph-based approaches (基于图的方法), 48
history of (历史), 398-399
introduction to (介绍), 397-398
learning how to summarize (学习如何做摘要), 406-409
LexPageRank, 406
multilingual (多语) 参见 Multilingual automatic summarization
stages of (阶段), 400
summary (摘要), 426-427
surface-based features used in (使用的表层特征), 400-401
TextRank, 404-406
Automatic Summary Evaluation based on n-gram graphs (AutoSummENG) (基于 n 元组图的自

动文摘评估)，419-420

Babel Fish (巴别鱼)
crosslingual question answering and (跨语言问答)，455
Systran，331

Backend services，of spoken dialog system (后端服务，口语对话系统)，500

Backoff smoothing techniques (回退平滑技术)
generalized backoff strategy (广义回退策略)，183-184
in language model estimation (语言模型估计)，178
nonnormalized form (非规范形式)，175
parallel backoff (平行回退)，184

Backus-Naur form，of context-free grammar (Backus-Naur 范式，上下文无关文法)，58

BananaSplit，IR preprocessing and (BananaSplit，信息检索预处理)，392

Base phrase chunks (基本短语块)，132-133

BASEBALL system，in history of QA systems (BASEBALL 系统，问答系统历史)，434

Basic Elements (BE) (基本单元)
automatic evaluation of summarization (文摘自动评价)，417-419
metrics in (指标)，420

Bayes rule，for sentence or topic segmentation (贝叶斯法则，用于句子或主题分割)，39-40

Bayesian theorem，maximum likelihood estimation and (贝叶斯定理，最大似然估计)，476

Bayesian parameter estimation (贝叶斯参数估计)，173-174

Bayesian topic-based language models (基于主题的贝叶斯语言模型)，186-187

BBN，event extraction and (BBN，事件抽取)，322

BE (Basic Elements) (基本单元)
automatic evaluation of summarization (文摘自动评价)，417-419
metrics in (指标)，420

BE with Transformation for Evaluation (BEwTE) (BE 评价变换)，419-420

Beam search (柱搜索)
machine translation and (机器翻译)，346
reducing search space using (用于减少搜索空间)，290-291

Bell tree，for coreference resolution (Bell 树，用于共指消解)，297-298

Bengali (孟加拉语)，参见 Indian languages

Berkeley word aligner，in machine translation (Berkeley 词对齐工具，机器翻译)，357

Bibliographic summaries，in automatic summarization (参考文献摘要，自动文摘)，397

Bilingual latent semantic analysis (bLSA) (双语潜在语义分析)，197-198

Binary classifier，in event matching (二元分类器，事件匹配)，323-324

Binary conditional model，for probability of mention links (二元条件模型，用于提及链接概率)，297-300

BLEU
machine translation metrics (机器翻译度量指标)，334，338
mention detection experiments and (提及检测实验)，295
ROUGE compared with (与 ROUGE 比较)，415-416

Block comparison method，for topic segmentation (块比较方法，用于主题分割)，38

bLSA (bilingual latent semantic analysis) (双语潜在语义分析)，197-198

BLUE (Boeing Language Understanding Engine) (波音语言理解引擎)，242-244

BM25 model，in document retrieval (BM25 模型，文档检索)，375

BNC (British National Corpus) (英国国家语料库)，118

Boeing Language Understanding Engine (BLUE) (波音语言理解引擎)，242-244

Boolean models (布尔模型)
for document representation in monolingual IR (用于单语 IR 文档表示)，372
for document retrieval (用于文档检索)，374

Boolean named entity flags，in PSG (布尔命名实体标志，PSG)，126

Bootstrapping (拿靴，滚雪球)
building subjective lexicon (构建主观性词典)，266-267
corpus-based approach to subjectivity and sentiment analysis (基于语料库的主观性和情感分析方法)，269
dictionary-based approach to subjectivity and sentiment analysis (基于字典的主观性和情感分析方法)，273
ranking approaches to subjectivity and sentiment

analysis (对主观性和情感分析方法进行排名), 275-276
semisupervised approach to relation extraction (半监督关系抽取方法), 318
Boundary classification problems (边界分类问题)
overview of (概述), 33
sentence boundaries (句子边界), 参见 Sentence boundary detection
topic boundaries (主题边界), 参见 Topic segmentation
British National Corpus (BNC) (英国国家语料库), 158
Brown Corpus, as resource for semantic parsing (Brown 语料库, 作为语义分析资源), 104
Buckwalter Morphological Analyzer (Buckwalter 形态分析器), 191

C-ASSERT, software program for semantic role labeling (C-ASSERT, 语义角色标注软件), 147
Call-flow (呼叫流程)
localization of (本地化), 514
strategy of dialog manager (对话管理器策略), 504
voice user interface (VUI) and (语音用户界面), 505-506
Call routing, natural language and (呼叫路由选择, 自然语言), 510
Canadian Hansards (加拿大议会语料库)
corpora for IR (信息检索语料库), 391
corpora for machine translation (机器翻译语料库), 358
Candidate answer extraction, in QA (候选答案抽取, QA)
answer scores (回答评分), 450-453
combining evidence (合并证据), 453-454
structural matching (结构匹配), 446-448
from structured sources (源自结构源), 449-450
surface patterns (表层模式), 448-449
type based (基于类型的), 446
from unstructured sources (源自非结构源), 445
Candidate answer generation, in QA (候选答案生成, QA)
components in QA architectures (QA 体系结构组件), 435
overview of (概述), 443
Candidate boundaries, processing stages of segmentation tasks (候选边界, 划分任务处理阶段), 48
Cannonization, deferred in RTE multiview representation (规范化, 在 RTE 多视点表示中延迟), 222
Capitalization (Uppercase), sentence segmentation markers (大写化 (大写), 句子分割标记), 39
CAS (Common analysis structure), UIMA (通用分析结构, UIMA), 527, 536
Cascading systems, type of EDT architectures (级联系统, EDT 系统结构类型), 286-287
Case (大小写)
parsing issues related to (句法分析相关问题), 88
sentence segmentation markers (句子分割标记), 38
Catalan (加泰罗尼亚语), 189
Categorical ambiguity, word sense and (类别歧义, 词义), 104
Cause-and-effect relations, causal reasoning and (因果关系, 因果推理), 259
CCG (Combinatory Categorical Grammar) (组合范畴语法), 129-130
CFGs, 参见 Context free grammar
Character n-gram models (字符 n 元模型), 370
Chart decoding, tree based models for machine translation (线图解码, 基于树的机器翻译模型), 351-352
Chart parsing, worst-case parsing algorithm for CFGs (线图分析, CFG 最坏情形分析算法), 74-79
Charts, IXIR distillation system (图表, IXIR 提炼系统), 488-489
CHILL (Constructive Heuristics Induction for Language Learning) (语言学习的建构性启发式归纳), 151
Chinese (汉语)
anaphora frequency in (回指频率), 444
challenges of sentence and topic segmentation (句子和主题分割挑战), 30
corpora for relation extraction (关系抽取语料库), 317
corpus-based approach to subjectivity and sentiment analysis (基于语料库的主观性和情感分析方法), 274-275
crosslingual language modeling (跨语言语言建模), 197-198
data sets related to summarization (文摘相关数据集), 424-425

dictionary-based approach to subjectivity and sentiment analysis (基于字典的主观性和情感分析方法), 272-273
distillation (提炼), 479, 490-491
EDT and, 286
Hownet lexicon for (知网词典), 195
human assessment of word meaning (词义人类评估), 333
IR and, 366, 393
isolating (analytic) languages (孤立 (分析) 语), 7
as isolating or analytic language (作为孤立语或分析语), 7
language modeling in without word segmentation (不分词的语言建模), 193-194
lingPipe for word segmentation (lingPipe 分词), 423
machine translation and (机器翻译), 322, 354, 358
mention detion experiments (提及检测实验), 294-296
multilingual issues in predicate-argument structures (谓词-论元结构的多语问题), 146-147
phrase structure treebank (短语结构树库), 70
polarity analysis of words and phrases (词和短语的极性分析), 259
preprocessing best practices in IR (IR 中的预处理最佳实践), 372
QA and, 461, 464
QA architectures and (QA 体系结构), 437-438
resources for semantic parsing (语义分析资源), 122
RTE in, 218
scripts not using whitespace (不用空格的书写方式), 369
subjectivity and sentiment analysis (主观性和情感分析), 259-260
TALES case study (TALES 案例研究), 538
translingual summarization (跨语际文摘), 399, 418
word segmentation and parsing (分词和句法分析), 89-90
word segmentation in (分词), 4-5
word sense annotation in (词义标注), 104
Chomsky, Noam (乔姆斯基, 诺姆), 13, 98-99
Chunk-based systems (基于块的系统), 132-133
Chunks (块)
defined (定义), 292
meaning chunks in semantic parsing (语义分析的意义块), 97
CIDR algorithm, for multilingual summarization (CIDR 算法, 用于多语文摘), 411
Citations (引用)
evaluation in distillation (提炼评价), 493
in GALE distillation initiative (GALE 提炼计划), 477
CKY algorithm, worst-case parsing for CFGs (CKY 算法, CFG 的最坏情形分析), 76-78
Class-based language models (基于类的语言模型), 178-179
Classes (类)
language modeling using morphological categories (用形态类别的语言建模), 193
of relations (关系), 311
Classification (分类)
of arguments (论元), 125, 139-140
data-driven (数据驱动), 287-289
dynamic class context in PSG (PSG 中的动态类上下文), 128
event extraction and (事件抽取), 321-322
overcoming independence assumption (克服独立性假设), 137-138
paradigms (范式), 133-137
problems related to sentence boundaries (句子边界相关问题), 参见 Sentence boundary detection
problems related to topic boundaries (主题边界相关问题) 参见 Topic segmentation
relation extraction and (关系抽取), 312-316
Classification tag lattice (trellis), searching for mentions (分类标签格 (栅), 搜索提及), 289
Classifiers (分类器)
in event matching (事件匹配), 323-324
localization of grammars (文法本地化), 516
maximum entropy classifiers (最大熵分类器), 37, 39-40
in mention detection (提及检测), 292-293
pipeline of (流水线), 321
in relation extraction (关系抽取), 313, 316-317
in subjectivity and sentiment analysis (主观性和情感分析), 270-272, 274
Type classifier in QA systems (QA 系统的类型分类器), 440-442
in word disambiguation (词义消歧), 110

CLASSIFY functions (CLASSIFY 函数), 313
ClearTK tool, for building summarization system (ClearTK 工具，用于建立文摘系统), 423
CLIR, 参见 Crosslingual information retrieval
Clitics (附着词)
  Czech example (捷克语例子), 5
  defined (定义), 4
Co occurence, of words between languages (共现，语言间词), 337 338
Coarse to fine parsing (先粗后细分析), 77-78
Code switchers (编码切换)
  Impact on sentence segmentation (对句子分割的影响), 31
  multilingual language modeling and (多语言建模), 195-196
COGEX, for answers in QA (COGEX，用于 QA 中回答评分), 451
Coherence, sentence-sentence connections and (连贯，句间联系), 402
Cohesion, anaphora resolution and (衔接，指代消解), 401-402
Collection language, in CLIR (文档集语言，CLIR), 355
Combination hypothesis, combining classifiers to boost performance (合并假设，合并分类器以增强性能), 293
Combinatory Categorical Grammar (CCG) (组合范畴语法), 129-130
Common analysis structure (CAS), UIMA (通用分析结构，UIMA), 527, 536
Communicator program, for meaning representation (Communicator 程序，用于意义表示), 148-150
Comparators, RTE (比较器，RTE), 218, 222-223
Competence vs. performance, Chomsky on (乔姆斯基论语言能力和运用), 13
Compile/replace transducer (Beesley and Karttunen) (编译/替换转录机，Beesley 和 Karttunen), 17
Componentization of design, for NLP aggregation (设计组件化，NLP 聚合), 524-525
Components of words (词的部件)
  lexemes (语素), 5
  morphemes (词素，形素), 5-7
  morphological typology and (形态类型学), 7-8
Compound slitting (复合词分割)
  BananaSplit tool (BananaSplit 工具), 392
  normalization for fusional languages (屈折语规范化), 371
Computational efficiency (计算效率)
  desired attributes of NLP Aggregation (NLP 聚合的期望性质), 525-526
  in GALE IOD, 537
  in GATE, 530
  in InfoStream Streams (InfoStream 流), 530-531
  in UIMA, 528
Computational Natural Language Learning (CoNLL) (计算自然语言学习), 132
Concatenative languages (连接型语言), 8
Concept space, interlingual document representations (概念空间，中间语言文档表示), 361
Conceptual density, as measure of semantic similarity (概念密度，语义相似度度量), 112
Conditional probability, MaxEnt formula for (条件概率，MaxEnt 公式), 316
Conditional random fields (CRF) (条件随机场)
  in discriminative parsing model (区分性分析模型), 84
  machine learning and (机器学习), 342
  measuring token frequency (计算词元频率), 359
  mention detection and (提及检测), 287
  relation extraction and (关系抽取), 316
  sentence or topic segmentation and (句子或主题分割), 38-40
Confidence weighted score (CWS), in QA (置信度权值，QA), 463
CoNLL (Computational Natural Language Learning) (计算自然语言学习), 132
Constituents (成分)
  atomic events and (原子事件), 418
  in PSG 127
Constituents, in RTE (RTE 中的成分)
  comparing annotation constituents (比较标记成分), 222-224
  multi-view representation of analysis and (分析的多视图表示), 220
  numerical quantities (NUM) (数量), 221, 233
Constraint-based language models (基于约束的语言模型), 177
Contructive Heuristics Induction for Language Learning (CHILL) (语言学习的建构性启发式归纳), 151
Content Analysis Toolkit (Tika), for preprocessing IR documents (内容分析工具包 Tika，用于 IR 文档预处理), 392

Content word, in PSG (实词, PSG), 125-126
Context, as measure of semantic similarity (上下文, 语义相似性度量), 112
Context-dependent process, in GALE IOD (GALE IOD 上下文相关过程), 536-537
Context features, of Rosetta Consortium distillation system (上下文特征, Rosetta 协会提炼系统), 486
Context-free grammar (CFG) (上下文无关文法)
for analysis of natural language syntax (用于自然语言句法分析), 60-61
dependency graphs in syntax analysis (句法分析依存图), 85-87
rules of syntax (句法规则), 59
shift-reduce parsing (移进归约分析), 72-73
worst-case parsing algorithm (最坏情形分析算法), 77-78
Contextual subjectivity analysis (上下文主观性分析), 261
Contradiction, in textual entailment (矛盾, 文本蕴涵), 211
Conversational speech, sentence segmentation in (对话语音, 句子分割), 31
Core arguments, PropBank verb predicates (核心论元, PropBank 动词谓词), 119
Coreference resolution (共指消解); 参见 Anaphora resolution
automatic summarization and (自动文摘), 398
Bell tree for (Bell 树), 297-298
experiments in (实验), 302-303
information extraction and (信息抽取), 103, 285-286
MaxEnt model applied to (应用 MaxEnt 模型), 300-301
models for (模型), 298-300
overviews of (概述), 296-298
as relation extraction system (作为关系抽取系统), 311
in RTE (RTE 中), 212, 227
Corpora (语料库)
for distillation (用于提炼), 480-483
for document-level annotations (用于文档级标注), 274
Europarl (European Parliament) (欧洲议会), 295, 345
for IR systems (用于 IR 系统), 390-391
for machine translation (MT) (用于机器翻译), 358
for relation extraction (用于信息抽取), 317
for semantic parsing (用于语义分析), 104-105
for sentence-level annotation (用于句子级标注), 271-272
for subjectivity and sentiment analysis (用于主观性和情感分析), 262-263, 274-275
for summarization (用于文摘), 405, 425
for word/phrase level annotations (用于词/短语级标注), 267-269
Coverage rate criteria, in language model evaluation (覆盖率标准, 语言模型评价), 170
Cranfield paradigm (Cranfield 范式), 387
Creativity/productivity, and the unknown word problem (创造性/能产性, 未登录词问题), 13-15
CRFs, 参见 Conditional random fields (CRFs)
Cross-Language Evaluation Forum (CLEF) (跨语言评测论坛)
applied to RTE to non-English language (应用于非英语语言 RTE), 218
IR and, 377, 390
QA and, 434, 454, 460-464
Cross-language mention propagation (跨语言提及传播), 293, 295
Cross-lingual projections (跨语言投射), 275
Crossdocument coreference (XDC), in Rosetta Consortium distillation system (跨文档共指, Rosetta 协会提炼系统), 482-483
Crossdocument structure theory Bank (CSTBank) (跨文档结构理论库), 425
Crossdocument structure theory (CST) (跨文档结构理论), 425
Crosslingual distillation (跨语言提炼), 490-491
Crosslingual information retrieval (CLIR) (跨语言信息检索)
best practices (最佳实践), 382
interlingual document representations (中间语言文档表示), 381-382
machine translation (机器翻译), 380-381
overview of (概述), 365, 378
translation-based approaches (基于翻译的方法), 378-380
Crosslingual language modeling (跨语言建模), 196-198
Crosslingual question answering (跨语言问答), 454-455

Crosslingual summarization (跨语言文摘), 398
CST (Crossdocument structure theory) (跨文档结构理论), 426
CSTBank (Crossdocument Structure Theory Bank) (跨文档结构理论库), 425
Cube pruning, decoding phrase-based models (立方剪枝, 基于短语的模型的解码), 347-348
CWS (Confidence weighted score), in QA (QA 中的置信度权值), 463
Cyrillic alphabet (西里尔字母表), 371
Czech (捷克语)
  ambiguity in (歧义), 11-13
  dependency graphs in syntax analysis (句法分析依存图), 62-65
  dependency parsing in (依存分析), 79
  finite-state models (有限状态模型), 18
  as fusional language (作为屈折语), 8
  language modeling (语言建模), 193
  morphological richness of (形态丰富), 355
  negation indicated by inflection (曲折变化表示否定), 5
  parsing issues related to morphology (与形态学相关的分析问题), 91
  productivity/creativity in (能产性/创造性), 14-15
  syntactic features used in sentence and topic segmentation (句子和主题分割的句法特征), 43
  unification based models (基于合一的模型), 19

DASML (Dialog Act Markup in Several Layers) (多层对话行为标注), 31
Data-driven (数据驱动)
  machine translation (机器翻译), 331
  question detection (提及检测), 287-289
Data formats, challenges in NLP aggregation (数据格式, NLP 聚合挑战), 524
Data-manipulation capabilities (数据处理能力)
  desired attributes of NLP aggregation (NLP 聚合的期望性质), 525
  in GATE, 530
  in InfoSphere Streams (InfoSphere 流), 531
  in UIMA, 528-529
Data reorganization, speech-to-text (STT) and (数据重组织, 语音到文本), 535-536
Data sets (数据集)
  for evaluating IR systems (用于评价 IR 系统), 389-391
  for multilingual automatic summarization (用于多语自动文摘), 425-426
Data types (数据类型)
  GALE Type System (GTS) (GALE 类型系统), 534-535
  usage conventions for NLP aggregation (NLP 聚合使用惯例), 540-541
Databases (数据库)
  of entity relations and events (实体关系和事件), 309-310
  relational (关系型), 449
DATR, unification-based morphology and (DATR, 基于合一的形态学), 18-19
DBpedia, 449
de Saussure, Ferdinand (索绪尔, 费迪南德), 13
Decision trees, for sentence or topic segmentation (决策树, 用于句子或主题分割), 39-40
Decoding phrase-based models (基于短语的模型的解码)
  cube pruning approach (立方剪枝方法), 347-348
  overview of (概述), 345-347
Deep representation, in semantic interpretation (深度表示, 语义解释), 101
Deep semantic parsing (深度语义分析)
  coverage in (覆盖率), 102
  overview of (概述), 98
Defense Advanced Research Projects Agency (DARPA) (国防高级研究计划局)
  GALE distillation initiative (GALE 提炼计划), 475-476
  GALE IOD case study (GALE IOD 案例研究), 参见 Interoperability Demo (IOD), GALE case study
  Topic Detection and Tracking (TDT) program (主题检测与跟踪计划), 32-33
Definitional questions, QA and (QA 与定义型问题), 433
Deletions metrics, machine translation (删除数指标, 机器翻译), 335
Dependencies (依存)
  global similarity in RTE and (RTE 全局相似度), 247
  high-level features in event matching (事件匹配高级特征), 324-325
Dependency graphs (依存图)
  phrase structure trees compared with (与短语结构树相比较), 69-70

in syntactic analysis (句法分析), 53-57
in treebank construction (树库构建), 62
Dependency parsing (依存分析)
implementing RTE and (实现 RTE), 227
Minipar and Stanford Parser (Minipar 和 Stanford 分析器), 455
MST algorithm for (MST 算法), 79-80
shift-reduce parsing algorithm for (移进归约算法), 73
structural matching and (结构匹配), 447
tree edit distance based on (基于~的树编辑距离), 240-241
worst case parsing algorithm for CFGs (CFG 最坏情形分析算法), 78
Dependency trees (依存树)
non projective (非投射性), 65-67
overview of (概述), 130-132
patterns used in relation extraction (用于关系抽取的模式), 318
projective (投射性), 64-65
Derivation, parsing and (推导, 分析), 71-72
Devanagari, preprocessing best practices in IR (梵文, IR 预处理最佳实践), 87
Dialog Act Markup in Several Layers (DAMSL) (多层对话行为标注), 31
Dialog manager (对话管理器)
directing speech generation (指导语音生成), 499-500
overview of (概述), 504-505
Dialog module (DM) (对话模块)
call-flow localization and (呼叫流程本地化), 516
voice user interface and (语音用户界面), 507-508
Dialogs (对话)
forms of (形式), 509-510
rules of (规则), 499-500
Dictionary-based approach, in subjectivity and sentiment analysis (基于字典的方法, 主观性和情感分析)
document-level annotations (文档级标注), 272-273
sentence-level annotations (句子级标注), 270-271
word/phrase-level annotations (词/短语级标注), 264-267
Dictionary-based morphology (基于字典的形态学), 15-16
Dictionary-based translations (基于字典的翻译)
applying to CLIR (用于 CLIR), 386
crosslingual modeling and (跨语言建模), 197
Directed dialogs (指导性对话), 509
Directed graphs (有向图), 79-80
Dirichlet distribution (狄利克雷分布)
hierarchical Dirichlet process (HDP) (层次狄利克雷过程), 187
language models and (语言模型), 174
latent Dirichlet allocation (LDA) model (潜在狄利克雷分配模型), 186
DIRT (Discovery of inference rules from text) (文本推理规则发现), 242
Disambiguation systems, word sense (消歧系统, 词义)
overview of (概述), 103
rule-based (基于规则), 105-108
semantic parsing and (语义分析), 152-153
semi-supervised (半监督), 114-116
software programs for (软件), 116-117
supervised (有监督), 108-112
unsupervised (无监督), 112-114
Discontiguous arguments, PropBank verb predicates (非连续论元, PropBank 动词谓词), 121
Discourse commitments (beliefs), RTE system based on (语篇约束 (信念), 基于~的 RTE 系统), 239-240
Discourse connectives, relating sentences by (语篇连接词, 关联句子), 29
Discourse features (语篇特征)
relating sentences by discourse connectives (用语篇连接词来关联句子), 29
in sentence and topic segmentation (句子和主题分割), 44
Discourse segmentation (语篇切分), 参见 Topic segmentation
Discourse structure (语篇结构)
automatic summarization and (自动文摘), 398, 410
RTE applications and (RTE 应用), 249
Discovery of inference rules from text (DIRT) (文本推理规则发现), 242
Discriminative language models (判别性语言模型)
modeling using morphological categories (用形态类别建模), 192-193
modeling without word segmentation (无分词建模), 194
overview of (概述), 179-180
Discriminative local classification methods, for sentence/topic boundary detection (判别性局部分类方法, 用于句子/主题边界检测), 36-38
Discriminative parsing models (判别性句法分析模型)

morphological information in (形态信息), 91-92
overview of (概述), 84-87
Discriminative sequence classification methods (判别性序列分类方法)
complexity of (复杂性), 40-41
overview of (概述), 34
performance of (性能), 41
for sentence/topic boundary detection (用于句子/边界检测), 38-39
Distance-based reordering model, in machine translation (基于距离的调序模型, 机器翻译), 344
Distance- features of coreference models (距离, 共指模型的特征), 301
Distillation (提炼)
bibliography (文献), 495-497
crosslingual (跨语言), 490-491
document and corpus preparation (文档和语料库准备), 480-483
evaluation and metrics (评价和指标), 493-494
example (例子), 476-477
indexing and (索引), 483
introduction to (介绍), 475-476
multimodal (多模态), 490
query answers and (查询答案), 483-487
redundancy reduction (冗余消除), 489-490
relevance and redundancy and (相关性和冗余性), 477-479
relevance detection (相关性检测), 488-489
Rosetta Consortium system (Rosetta 协会系统), 479-480
summary (总结), 495
system architectures for (系统结构), 488
DM (Dialog module) (对话模块)
call-flow localization and (呼叫流程本地化), 514
voice user interface and (语音用户界面), 507-508
Document-level annotations, for subjectivity and sentiment analysis (文档级标注, 用于主观性和性感分析)
corpus-based (基于语料库), 274
dictionary-based (基于字典), 272-273
overview of (概述), 272
Document retrieval system, INDRI (文档检索系统 INDRI), 323
Document structure (文档结构)
bibliography (文献), 49-56
comparing segmentation methods (比较分割方法), 40-41
discourse features of segmentation methods (切分方法的语篇特征), 44
discriminative local classification method for segmentation (切分的判别性局部分类方法), 36-38
discriminative sequence classification methods for segmentation (切分的判别性序列分类方法), 38-39
discussion (讨论), 48-49
extensions for global modeling sentence segmentation (句子分割全局建模扩展), 40
features of segmentation methods (切分方法特征), 41-42
generative sequence classification method for segmentation (切分的生成性序列分类方法), 34-36
hybrid methods for segmentation (切分的混合方法), 39-40
introduction to (介绍), 29-30
lexical features of segmentation methods (切分方法的词法特征), 42-43
methods for detecting probable sentence or topic bounaries (检测可能的句子或主题边界的方法), 33-34
performance of segmentation methods (切分方法性能), 41
processing stages of segmentation tasks (切分任务的处理阶段), 48
prosodic features for segmentation (切分的韵律特征), 45-48
sentence boundary detection (segmentation) (句子边界检测/切分), 30-32
speech-related features for segmentation (切分的语音相关特征), 45
summary (总结), 49
syntactic features of segmentation methods (切分方法的句法特征), 43-44
topic boundary detection (segmentation) (主题边界检测/分割), 32-33
typographical and structural features for segmentation (切分的排版和结构特征), 44-45
Document Understanding Conference (DUC) (文档理解会议), 404, 424
Documents, in distillation systems (文档, 提炼系统)
indexing (索引), 483
preparing (准备), 480-482
retrieving (检索), 483-484
Documents, in IR (IR 的文档)

interlingual representation (中间语言表示), 361-362
monolingual representation (单语表示), 372-373
preprocessing (预处理), 366-367
a priori models (先验模型), 377
reducing MLIR to CLIR (把 MLIR 化简为 CLIR), 383-384
syntax and encoding (句法和编码), 367-378
translating entire collection (翻译全部文档集), 379
Documents, QA searches (文档, QA 搜索), 444
Domain dependent scope, for semantic parsing (领域相关范围, 语义分析), 102
Domain independent scope, for semantic parsing (领域无关范围, 语义分析), 102
Dominance relations (支配关系), 325
DSO Corpus, of Sense-Tagged English (DSO 语料库, 英语, 语义标注), 134
DUC (Document Understanding Conference) (文档理解会议), 404, 424
Dutch (荷兰语)
IR and, 390-391
normalization and (规范化), 371
QA and, 435, 444, 461
RTE in, 218

Edit distance, features of coreference models (编辑距离, 共指模型特征), 301
Edit Distance Textual Entailment Suite (EDITS) (编辑距离文本蕴涵套件), 240-241
EDT, 参见 Entity detection and tracking (EDT)
Elaborative summaries, in automatic summarization (详细摘要, 自动文摘), 397
ElixirFM lexicon (ElixirFM 词典), 20
Ellipsis, linguistic supports for cohesion (省略, 衔接的语言学支持), 401
EM algorithm (EM 算法), 参见 Expectation-maximization (EM) algorithm
Encoding (编码)
of documents in information retrieval (信息检索中的文档), 356
parsing issues related to (相关的句法分析问题), 89
English (英语)
call-flow localization and (呼叫流程本地化), 514
co-occurrence of words between languages (语言间词的同现), 337-339
corpora for relation extraction (关系抽取语料库), 317
corpus based approach to subjectivity and sentiment analysis (基于语料库的主观性和情感分析方法), 271-272
crosslingual language modeling (跨语言建模), 197-198
dependency graphs in syntax analysis (句法分析依存图), 55
discourse parsers for (语篇分析器), 403
distillation (提炼), 478, 490-491
finite-state transducer applied to English example (有限状态转录机应用于英语例子), 17
GALE IOD and, 532, 534-536
IR and, 390
as isolating or analytic language (作为孤立或分析型语言), 7
machine translation and (机器翻译), 322, 354, 358
manually annotated corpora for (手工标注语料库), 274
mention detection (提及检测), 287
mention detection experiments (提及检测实验), 294-296
multilingual issues in predicate-argument structures (谓词-论元结构的多语问题), 146-147
normalization and (规范化), 370
phrase structure trees in syntax analysis (句法分析中的短语结构树), 62
polarity analysis of words and phrases (词和短语的极性分析), 269
productivity/creativity in (能产性/创造性), 14-15
QA and, 444, 461
QA architectures and (QA 架构), 457
RTE in, 218
sentence segmentation markers (句子分割标记), 30
subjectivity and sentiment analysis (主观性和情感分析), 259-260, 263
as SVO language (作为主动宾语言), 356
TALES case study (TALES 案例研究), 538
tokenization and (词元切分), 410
translingual summarization (跨语际摘要), 398-399, 424-426
word order and (词序), 356
WordNet and, 109
Enrichment, in RTE (RTE 中的富化)
implementing (实现), 228-231

modeling (建模), 225
Ensemble clustering methods, in relation extraction (关系抽取中的集成聚类方法), 317-318
Entities (实体)
classifiers (分类器), 292-293
entity based relation extraction (基于实体的关系抽取), 314-315
events (事件), 参见 Events
relations (关系), 参见 Relations
resolution in semantic interpretation (语义解释中的消解), 100
Entity detection and tracking (EDT) (实体检测和跟踪)
Bell tree for (Bell 树), 297-298
bibliography (文献), 303-307
combining entity and relation detection (合并实体和关系检测), 320
coreference models (共指模型), 298-300
coreference resolution (共指消解), 295-296
data-driven classification (数据驱动分类), 287-289
experiments in coreference resolution (共指消解中的实验), 302-303
experiments in mention detection (提及检测实验), 294-295
features for mention detection (提及检测特征), 291-294
introduction to (介绍), 285-287
MaxEnt model applied to (把 MaxEnt 模型应用于—), 300-301
mention detection task (提及检测任务), 287
searching for mentions (搜索提及), 289-291
summary (总结), 303
Equivalent terms, in GALE distillation initiative (等价术语, GALE 提炼计划), 475
Errors (错误)
machine translation (机器翻译), 335-337, 343, 349
parsing (句法分析), 141-144
sentence and topic segmentation (句子和主题分割), 41
ESA (Explicit semantic analysis), for interlingual document representation (中间语言文档表示的显式语义分析), 382
Europarl (European Parliament) corpus (欧洲议会语料库)
evaluating co-occurrence of word between languages (评测语言间词的同现), 357
for IR systems (用于 IR 系统), 391
for machine translation (用于机器翻译), 358
phrase translation tables (短语翻译表), 345
European Language Resources Association (欧洲语言资源协会), 218
European languages (欧洲语言), 参见各种语言
crosslingual question answering and (跨语言问答), 455
QA architectures and (QA 架构), 437
whitespace use in (空格的使用), 369
European Parliament Plenary Speech corpus (欧洲议会全会语音语料库), 295
EVALITA, applying to RTE to non-English languages (EVALITA 用于非英语的 RTE), 218
Evaluation in automatic summarization (自动文摘的评测)
automated evaluation methodologies (自动评测方法学), 416-418
manual evaluation methodologies (手工评测方法学), 413-415
overview of (概述), 412-413
recent developments in (最新进展), 418-419
Evaluation, in distillation (评测, 提炼)
citation checking (引用检查), 493
GALE and, 492
metrics (度量指标), 493-494
overview of (概述), 491-492
relevance and redundancy and (相关性和冗余性), 492-493
Evaluation, in IR (评测, IR)
best practices (最佳实践), 391
data sets for (数据集), 389-390
experimental set up for (实验设定), 387
measures in (度量), 388-389
overview of (概述), 386-387
relevance assessment (相关性评估), 387-388
trec_eval tool for (trec_eval 工具), 395
Evaluation, in MT (评测, 机器翻译)
automatic evaluation (自动评测), 334-335
human assessment (人工评测), 332-334
meaning and (意义), 332
metrics for (指标), 335-337
Evaluation, in QA (评测, QA)
answer correctness (回答正确性), 461-462
performance metrics (性能指标), 462-464
tasks (任务), 460-461

Evaluation, in RTE (评测, RTE)
general method and (一般方法), 224
improving (改进), 251-252
performance evaluation (性能评测), 213-214
Evaluation, of aggregated NLP (评测, 聚合 NLP), 541
Evaluative summaries, in automatic summarization (评价摘要, 自动文摘), 397
Events (事件), 参见 Entities
extraction (抽取), 320-322
future directions in extraction (抽取的未来方向), 325
matching (匹配), 323-325
moving beyond sentence processing (超出句子处理), 323
overview of (概述), 320
resolution in semantic interpretation (语义解释消解), 160
Exceptions (例外)
challenges in NLP aggregation (NLP 聚合挑战), 524
functional morphology models and (函数式形态学模型), 19
Exclamation point (!), as sentence segmentation marker (感叹号, 句子分割标记), 30
Existence classifier, in relation extraction (Existence 分类器, 关系抽取), 313
Expansion documents, query expansion and (扩展文档, 查询扩展), 377
Expansion rules, features of predicate-argument structures (扩展规则, 谓词-论元结构特征), 145
Expection-maximization (EM) algorithm (期望最大化算法)
split-merge over trees using (用一对树分裂合并), 83
symmetrization and (对称性), 340-341
word alignment between languages and (语言间词对齐), 339-340
Experiments (实验)
in coreference resolution (共指消解), 302-303
in mention detection (提及检测), 294-295
setting up for IR evaluation (IR 评测设定), 387
Explicit semantic analysis (ESA), for interlingual document representation (显式语义分析, 用于中间语言文档表示), 382
eXtended WordNet (XWN) (扩充 WordNet), 451
Extraction (抽取)
in automatic summarization (自动文摘), 399-400
as classification problem (作为分类问题), 312-313
of events (事件), 320-322, 326
of relations (关系), 310-311
Extraction, in QA (抽取, QA)
candidate extraction from structured sources (结构源候选抽取), 449-450
candidate extraction from unstructured sources (非结构源候选抽取), 445-449
candidate extraction techniques in QA (QA 候选抽取技术), 443
Extracts (摘录)
in automatic summarization (自动文摘), 397
defined (定义), 400
Extrinsic evaluation, of summarization (外围评测, 摘录), 412

F-measure, in mention detection (提及检测, F 值), 294
Factoid QA systems (事实型 QA 系统)
answer correctness (回答正确性), 461
answer scores and (回答评分), 450-453
baseline (基准系统), 443
candidate extraction or generation and (候选抽取或生成), 435
challenges in (挑战), 464-465
crosslingual question answering and (跨语言问答), 454
evaluation tasks (评测任务), 460-461
extracting using high-level searches (采用高层搜索抽取), 445
extracting using structured matching (采用结构匹配抽取), 446
MURAX and, 434
performance metrics (性能指标), 462-463
questions (问题), 433
type classification of (类型分类), 440
Factoids, in manual evaluation of summarization (事实型问题, 文摘的人工评测), 413
Factored (cascaded) model (分解/级联模型), 413
Factored language models (FLM) (因子化语言模型)
machine translation and (机器翻译), 365
morphological categories in (形态范畴), 193
overview of (概述), 183-184
Feature extractors (特征抽取器)
building summarization systems (建立文摘系统), 423

distillation and (提炼), 485-486
summarization and (文摘), 486
Features (特征)
in mention detection system (提及检测系统), 291-294
typed feature structures and unification (有类型的特征结构与合一), 18-19
in word disambiguation system (词义消歧系统), 110-112
Features, in sentence or topic segmentation (特征, 句子或主题分割)
defined (定义), 33
discourse features (语篇特征), 42
lexical features (词法特征), 42-43
overview of (概述), 41-42
predictions based on (基于…的预测), 28
prosodic features (韵律特征), 45-48
speech-related features (语音相关的特征), 45
syntactic features (句法特征), 43-44
typographical and structural features (排版和结构特征), 44-45
Fertility, word alignment and (繁衍率, 词对齐), 348
File types, document syntax and (文件类型, 文档句法), 367-368
Finite-state morphology (有限状态形态学), 16-18
Finite-state transducers (有限状态转录机), 16-17, 20
Finnish (芬兰语)
as agglutinative language (作为黏着型语言), 7
IR and, 390-391
irregular verbs (不规则动词), 10
language modeling (语言建模), 189-191
parsing issues related to morphology (与形态相关的分析问题), 91
summarization and (文摘), 399
FIRE (Forum for Information Retrieval Evaluation) (信息检索评测论坛), 390
Flexible, distributed componentization (灵活的分布式组件化)
desired attributes of NLP aggregation (NLP 聚合的期望属性), 524-525
in GATE, 550
in InfoSphere Streams (InfoSphere 流), 530
in UIMA, 528
FLM, 参见 Factored language models (FLM)
Fluency, of translation (翻译的流利度), 334
Forum for Information Retrieval Evaluation (FIRE) (信息检索评测论坛), 390
FraCaS corpus, applying natural logic to RTE (FraCaS 语料库, 把自然逻辑应用于 RTE), 245
Frame elements (框架元素)
in PSG (PSG 中), 126
semantic frames in FrameNet (FrameNet 中的语义框架), 118
FrameNet
limitation of (限制), 122-123
resources (资源), 122
resources for predicate-argument recognition (用于谓词-论元识别的资源), 118-122
Freebase, 449
French (法语)
automatic speech recognition (ASR) (自动语音识别), 379
dictionary-based approach to subjectivity and sentiment analysis (基于字典的主观性和情感分析方法), 267
human assessment of translation English to (把英语翻译成法语的人工评估), 332-333
IR and, 378, 390-391
language modeling (语言建模), 188
localization of spoken dialog systems (口语对话系统的本地化), 513
machine translation and (机器翻译), 350, 353-354, 358
phrase structure trees in syntax analysis (句法分析中的短语结构树), 68
polarity analysis of words and phrases (词和短语的极性分析), 268
QA and, 454, 461
RTE in, 217-218
translingual summarization (跨语际文摘), 398
word segmentation and (分词), 90
WordNet and, 139
Functional morphology (函数式形态学), 19-21
Functions, viewing language relations as (把语言关系视作函数), 17
Fusional languages (屈折型语言)
functional morphology models and (函数式形态学模型), 19
morphological typology and (形态类型学), 8
normalization and (规范化), 371
preprocessing best practices in IR (IR 中的预处理最佳实践), 371

GALE，参见 Global Autonomous Language Exploitation (GALE)
GALE Type System (GTS)（GALE 类型系统），534-535
GATE，参见 General Architecture for Text Engineering (GATE)
Gazetteer，features of mention detection systems（地名词典，提及检测系统的特征），293
GEN-AFF（general-affiliation），relation class（GEN-AFF，一般关系，关系类别），312
Gender（性）
  ambiguity resolution（消歧），13
  multilingual approaches to grammatical gender（语法性的多语方法），398
General Architecture for Text Engineering (GATE)（文本工程通用架构）
  attributes of（属性），530
  history of summarization systems（文摘系统历史），399
  overview of（概述），529-530
  summarization frameworks（文摘框架），422
General Inquirer，subjectivity and sentiment analysis lexicon（通用查询器，主观性和情感分析词典），262
Generalized backoff strategy，in FLM（广义回退策略），183-184
Generative parsing models（生成式分析模型），83-84
Generative sequence classification methods（生成式序列分类方法）
  complexity of（复杂性），40
  overview of（概述），34
  performance of（性能），41
  for sentence/topic boundary detection（用于句子/主题边界检测），34-36
Geometric vector space model，for document retrieval（几何向量空间模型，用于文档检索），375
GeoQuery
  resources for meaning representation（用于意义表示的资源），149
  supervised systems for semantic parsing（有监督的语义分析系统），151
German（德语）
  co-occurrence of words between languages（语言间词的同现），337-339
  dictionary-based approach to subjectivity and sentiment analysis（基于字典的主观性和情感分析方法），265-266，273
  discourse parsers for（语篇分析器），403
  as fusional language（作为屈折型语言），8
  IR and，393-395
  language modeling（语言建模），189
  mention detection（提及检测），287
  morphological richness of（形态丰富性），354-355
  normalization（规范化），370-371
  OOV rate in（未登录词率），191
  phrase-based model for decoding（用于解码的基于短语的模型），345
  polarity analysis of words and phrases（词和短语的极性分析），269
  QA and，481
  RTE in，218
  subjectivity and sentiment analysis（主观性和情感分析），259，278
  summarization and（文摘），358，403-404，420
  WordNet and，109
Germanic languages，language modeling for（日耳曼语系，语言建模），189
GetService process，of voice user interface (VUI)（GetService 过程，语音用户界面），506-507
Giza，machine translation program（Giza，机器翻译程序），423
GIZA toolkit，for machine translation（GIZA 工具包，用于机器翻译），357
Global Autonomous Language Exploitation (GALE)（全球自主语言开发）
  distillation initiative of DARPA（DARPA 的提炼计划），475-478
  evaluation in distillation（提炼评测），482
  Interoperability Demo case study（互操作性演示案例研究），参见 Interoperability Demo (IOD)，GALE case study
  metrics for evaluating distillation（提炼评测指标），494
  relevance and redundancy in（相关性和冗余性），477-479
Global linear model，discriminative approach to learning（全局线性模型，区分性学习方法），84
Good Turing（古德-图灵）
  machine translation and（机器翻译），345
  smoothing techniques in language model estimation（语言模型估算的平滑方法），172
Google，435
Google Translate（Google 翻译），331，455
Grammars（语法，文法）

Combinatory Categorial Grammar (CCG) (组合范畴语法), 129-130
context-free (上下文无关), 参见 Context-free grammars (CFGs)
head-driven phrase structure grammar (HPSG) (中心词驱动的短语结构文法), 18
localization of (本地化), 514, 516-517
morphological resource grammars (形态资源文法), 19, 21
phrase structure (短语结构), 参见 Phrase Structure Grammar (PSG)
probabilistic context-free (概率上下文无关), 参见 Probabilistic context-free grammars (PCFGs)
rule-based grammars in speech recognition (基于规则的语音识别文法), 501-503
Tree Adjoining Grammar (TAG) (树邻接语法), 130
voice user interface (VUI) (语音用户界面), 508-509
Grammatical Framework (文法框架), 19, 21
Graph-based approaches, to automatic summarization (基于图的方法, 自动文摘)
applying RST to summarization (把 RST 用于文摘), 402-404
coherence and cohesion and (连贯与衔接), 401-402
LexPageRank, 406
overview of (概述), 401
TextRank, 404-406
Graph generation, in RTE (RTE 中的图生成)
implementing (实现), 231-232
modeling (建模), 226
Graphemes (形素), 4
Greedy best-fit decoding, in mention detection (贪心最佳优先解码, 提及检测), 322
Groups, aligning views in RTE (组, RTE 中对齐视图), 233
Grow-diag-final method, for word alignment (Grow-diag-final 方法, 用于词对齐), 341
GTS (GALE Type System) (GALE 类型系统), 334-335
Gujarati (古吉拉特语), 参见 India languages

HDP (Hierarchical Dirichlet process) (层次狄利克雷过程), 187
Head-driven phrase structure grammar (HPSG) (中心词驱动的短语结构文法), 18
Head word (中心词)
dependency trees and (依存树), 131
in Phrase Structure Grammar (PSG) (短语结构文法), 134
Headlines, typographical and structural features for sentence and topic segmentation (标题, 句子和主题分割的排版和结构特征), 44-45
Hebrew (希伯来语)
encoding and script (编码和书写方式), 368
preprocessing best practices in IR (IR 中的预处理最佳实践), 371
tokens in (词元), 4
unification-based models (基于合一的方法), 18
HELM (hidden event language model) (隐事件语言模型)
applied to sentence segmentation (应用于句子分割), 36
methods for sentence or topic segmentation (句子或主题分割方法), 40
Hidden event language model (HELM) (隐事件语言模型)
applied to sentence segmentation (应用于句子分割), 35
methods for sentence or topic segmentation (句子或主题分割方法), 40
Hidden Markov model (HMM) (隐马尔可夫模型)
applied to topic and sentence segmentation (用于主题和句子分割), 34-36
measuring token frequency (测量词元频率), 359
mention detection and (提及检测), 287
methods for sentence or topic segmentation (句子或主题分割方法), 39
word alignment between languages and (语言间的词对齐), 340
Hierarchical Dirichlet process (HDP) (层次狄利克雷过程), 187
Hierarchical phrase-based models, in machine translation (基于层次短语的模型, 机器翻译), 350-351
Hierarchical phrase pairs, in machine translation (层次短语对, 机器翻译), 351
High level features, in event matching (高级特征, 事件匹配), 324
Hindi (印地语), 参见 Indian languages
IR and, 390
resources for semantic parsing (语义分析资源), 122
translingual summarization (跨语种文摘), 399

History, conditional context of probability (历史，概率条件上下文), 83
HMM. 参见 Hidden Markov model (HMM)
Homonymy (同音异义，同形异义)
in Korean (朝鲜语), 19
word sense ambiguities and (词义歧义), 104
HowNet
dictionary-based approach to subjectivity and sentiment analysis (基于字典的主观性和情感分析方法), 272-273
semantic parsing resources (语义分析资源), 105
HTML Parser, preprocessing IR documents (HTML 分析器，预处理 IR 文档), 392
Hunalign tool, for machine translation (Hunalign 工具，用于机器翻译), 357
Hungarian (匈牙利语)
dependency graphs in syntax analysis (句法分析依存图), 65
IR and, 390
morphological richness of (形态丰富), 355
Hybrid methods, for segmentation (分割的组合方法), 39-40
Hypergraphs, worst-case parsing algorithm for CFGs (超图，CFG 最坏情形分析算法), 74-79
Hypernyms (上位词), 442
Hyponymy (上下义关系), 310
Hypotheses, machine translation and (假设，机器翻译), 346

IBM Models, for machine translation (IBM 模型，用于机器翻译), 338-341
Identification, of arguments (识别，论元), 133, 139-140
IDF, 参见 Inverse document frequency (IDF)
IE, 参见 Information Extraction (IE)
ILP (Integer linear programming) (整数线性规划), 247
Implementation process, in RTE (实现过程，RTE)
alignment (对齐), 233-236
enrichment (富化), 228-231
graph generation (图生成), 231-232
inference (推理), 235-238
overview of (概述), 227
preprocessing (预处理), 227-228
training (训练), 238
IMS (It Makes Sense), program for word sense disambiguation (IMS，词义消歧程序), 117
Independence assumption (独立性假设)
document retrieval and (文档检索), 372
overcoming in predicate argument structure (在谓词-论元结构中克服), 137-138
Indexes (索引)
of documents in distillation system (提炼系统的文档), 483
for IR generally (一般用于信息检索), 366
latent semantic indexing (LSI) (潜在语义索引), 381
for monolingual IR (用于单语信息检索), 373-374
for multilingual IR (用于多语信息检索), 383-384
phrases indices (短语索引), 366, 369-370
positional indices (位置索引), 366
translating MLIR queries (翻译 MLIR 查询), 384
Indian languages, IR (印度语言，IR), 390. 参见 Hindi
INDRI document retrieval system (INDRI 文档检索系统), 323
Inexact retrieval models, for monolingual information retrieval (不精确检索模型，用于单语信息检索), 374
InfAP metrics, for IR performance (InfAP 指标，用于衡量 IR 性能), 389
Inference, textual, (推理，文本), 参见 Textual inference
Inflectional paradigms (屈折变化范式)
in Czech (捷克语), 11-12
in morphologically rich languages (形态丰富的语言), 189
Information context, as measure of semantic similarity (信息上下文，作为语义相似度的度量), 112
Information extraction (IE) (信息抽取), 参见 Entity detection and tracking (EDT)
defined (定义), 285
entity and event resolution and (实体和事件消解), 100
Information retrieval (IR) (信息检索)
bibliography (文献), 394-396
crosslingual (跨语言), 参见 Crosslingual information retrieval (CLIR)
data sets used in evaluation of (评测中使用的数据集), 389-391
distillation compared with (与提炼做比较), 475
document preprocessing for (文档预处理),

365-367
document syntax and encoding (文档句法和编码), 367-368
evaluation in (评测), 386-387, 391
introduction to (介绍), 366
key word searches in (关键词搜索), 433
measures in (指标), 388-389
monolingual (单语), 参见 Monolingual information retrieval
multilingual (多语), 参见 Multilingual information retrieval (MLIR)
normalization and (规范化), 370-371
preprocessing best practices (预处理最佳实践), 371-372
redundancy problem and (冗余性问题), 488
relevance assessment (相关性评估), 387-388
summary (总结), 393
tokenization and (词元化), 369—370
tools, software, and resources (工具, 软件与资源), 391-393
translingual (跨语际), 491
Informative summaries, in automatic summarization (信息型摘要, 自动文摘), 401-404
InfoSphere Streams (InfoSphere 流), 530-531
Insertion metric, in machine translation (插入指标, 机器翻译), 335
Integer linear programming (ILP) (整数线性规划), 247
Interactive voice response (IVR) (交互语音应答), 365, 511
Interoperability Demo (IOD), GALE case study (互操作性演示, GALE 案例研究)
computational efficiency (计算效率), 537
flexible application building with (用来构建灵活应用), 537
functional description (功能性描述), 532-534
implementing (实现), 534-537
overview of (概述), 531-532
Interoperability, in aggregated NLP (互操作性, 聚合 NLP), 540
Interpolation, language model adaptation and (插值, 语言模型适应), 178
Intrinsic evaluation, of summarization (内部评测, 文摘), 412
Inverse document frequency (IDF) (逆文档频率)
answer scores in QA and (QA 中的回答评分), 456-461
document representation in monolingual IR (单语 IR 中的文档表示), 373
relationship questions and (关系问题), 488
searching over unstructured sources (搜索非结构源), 445
Inverted indexes, for monolingual information retrieval (倒排索引, 用于单语信息检索), 373-374
IOD case study (IOD 案例研究), 参见 Interoperability Demo (IOD), GALE case study
IR, 参见 Information retrieval (IR)
Irregularity (不规则性)
defined (定义), 8
issues with morphology induction (形态归纳问题), 21
in linguistic models (语言模型), 8-10
IRSTLM toolkit, for machine translation (IRSTLM 工具包, 用于机器翻译), 357
Isolating (analytic) languages (孤立语/分析型语言)
finite-state technology applied to (应用有限状态技术), 18
morphological typology and (形态类型学), 7
It Makes Sense (IMS), program for word sense disambiguation (IMS, 词义消歧程序), 117
Italian (意大利语)
dependency graphs in syntax analysis (句法分析中的依存图), 63
IR and, 390-391
normalization and (规范化), 371
polarity analysis of words and phrases (词和短语的极性分析), 259
QA and, 461
RTE in, 218
summarization and (文摘), 389
WordNet and, 109
IVR (interactive voice response) (交互语音应答), 365, 511
IXIR distillation system (IXIR 提炼系统), 488-489
Japanese (日语)
as agglutinative language (作为黏着型语言), 7
anaphora frequency in (回指频率), 446
call-flow localization and (呼叫流程本地化), 514
crosslingual QA (跨语言问答), 455
discourse parsers for (语篇分析器), 403
EDT and, 285
GeoQuery corpus translated into (GeoQuery 语料

库翻译为日语), 349
IR and, 390
irregular verbs (不规则动词), 10
language modeling (语言建模), 193-194
polarity analysis of words and phrases (词和短语的极性分析), 269
preprocessing best practices in IR (IR 中的预处理最佳实践), 371-372
QA architectures and (QA 架构), 437-438, 461, 464
semantic parsing (语义分析), 122, 151
subjectivity and sentiment analysis (主观性和情感分析), 259, 267-271
word order and (词序), 356
word segmentation in (分词), 4-5
JAVELIN system, for QA (JAVELIN 系统, 用于问答), 437
Joint inference, NLP and (联合推理, NLP), 320
Joint systems (联合系统)
optimization vs. interoperability in aggregated NLP (聚合 NLP 中的优化和互操作性), 540
types of EDT architectures (EDT 架构类型), 285
Joshua machine translation program (Joshua 机器翻译程序), 357, 423
JRC-Acquis corpus (JRC-Acquis 语料库)
for evaluating IR systems (用于评测 IR 系统), 390
for machine translation (用于机器翻译), 358

KBP (Knowledge Base population), of Text Analysis Conferences (TAC) (知识库填充, 文本分析会议), 481-482
Kernel functions, SVM mapping and (核函数, SVM 映射), 317
Kernel methods, for relation extraction (核方法, 用于关系抽取), 319
Keyword searches (关键词搜索)
in IR, 433
searching over unstructured sources (搜索非结构源), 443-445
KL-ONE system, for predicate-argument recognition (KL-ONE 系统, 用于谓词-论元识别), 123
Kneser-Ney smoothing technique, in language model estimation (Kneser-Ney 平滑方法, 语言模型估计), 172
Knowledge Base population (KBP), of Text Analysis Conferences (TAC) (知识库填充, 文本分析会议), 481-482
Korean (朝鲜语)
as agglutinative language (作为黏着型语言), 7
ambiguity in (歧义), 10-11
dictionary-based approach in (基于字典的方法), 16
EDT and, 286
encoding and script (编码和书写方式), 368
finite-state models (有限状态模型), 18
gender (性), 13
generative parsing model (生成式句法分析模型), 92
IR and, 390
irregular verbs (不规则动词), 10
language modeling (语言建模), 196
language modeling using subword units (用子词单元进行语言建模), 192
morphemes in (词素), 6-7
polarity analysis of words and phrases (词和短语的极性分析), 269
preprocessing best practices in IR (IR 预处理最佳实践), 371-372
resources for semantic parsing (语义分析资源), 122
word segmentation in (分词), 4-5
KRISPER program, for rule based semantic parsing (KRISPER 程序, 用于基于规则的语义分析), 151

Language identification, in MLIR (语言识别, MLIR), 383
Language models (语言模型)
adaptation (适应), 176-178
Bayesian parameter estimation (贝叶斯参数估计), 175-176
Bayesian topic-based (基于主题的贝叶斯), 186-187
bibliography (文献), 199-208
class-based (基于类), 178-179
crosslingual (跨语言), 196-198
discriminative (判别性), 179-180
for document retrieval (用于文档检索), 375-376
evaluation of (评测), 170-171
factored (因子化), 183-184
introduction to (介绍), 169
language-specific problems (具体语言相关问

型), 188-189
large scale models (大规模模型), 174-176
MaxEnt, 181-183
maximum-likelihood estimation and smoothing (最大似然估计与平滑), 171-173
morphological categories in (形态类别), 192-193
for morphologically rich languages (形态丰富语言), 189-191
multilingual (多语), 195-196
n-gram approximation (n元近似), 170
neural network (神经网), 187-188
spoken vs. written languages and (口语与书面语), 194-195
subword unit selection (子词单元选择), 191-192
summary (总结), 198
syntax-based (基于句法的), 180-181
tree-based (基于树的), 185-186
types of (类型), 178
variable-length (变长), 179
word segmentation and (分词), 193-194
The Language Understanding Annotated Corpus (语言理解标注语料库), 425
Langue and parole (de Saussure) (索绪尔的语言和言语), 13
Latent Dirichlet allocation (LDA) model (潜在狄利克雷分配模型), 186
Latent semantic analysis (LSA) (潜在语义分析)
bilingual (bLSA) (双语~), 197-198
language model adaptation and (语言模型适应), 176-177
probabilistic (PLSA) (概率~), 176-177
Latent semantic indexing (LSI) (潜在语义索引), 381
Latin (拉丁文)
as fusional language (作为屈折语), 8
morphologies of (形态学), 20
preprocessing best practices in IR (IR中的预处理最佳实践), 371
transliteration of scripts to (拉丁转写), 368
Latvian (拉脱维亚语)
IR and, 390
summarization and (文摘), 398
LDA (Latent Dirichlet allocation) model (潜在狄利克雷分配模型), 186
LDC, 参见 Linguistic Data Consortium (LDC)
LDOCE (Longman Dictionary of Contemporary English) (朗曼当代英语词典), 104
LEA, 参见 Lexical entailment algorithm (LEA)
Learning, discriminative approach to (区分性学习方法), 84
Lemmas (原形)
defined (定义), 5
machine translation metrics and (机器翻译度量指标), 335
mapping terms to (把术语转换为原形), 370
Lemmatizers (词形还原工具)
mapping terms to lemmas (把术语转换为原形), 370
preprocessing best practices in IR (IR中的预处理最佳实践), 371
Lemur IR framework (Lemur信息检索框架), 392
Lesk algorithm (Lesk算法), 105-106
Lexemes (语素)
functional morphology models and (函数式形态模型), 19
overview of (概述), 5
Lexical chains, in topic segmentation (词汇链, 主题分割), 38, 43
Lexical choice, in machine translation (选词, 机器翻译), 354-355
Lexical collocation (词汇搭配), 401
Lexical entailment algorithm (LEA) (词法蕴涵算法)
alignment stage of RTE model (RTE模型的对齐阶段), 236
enrichment stage of RTE model (RTE模型的富化阶段), 228-231
inference stage of RTE model (RTE模型的推理阶段), 237
preprocessing stage in RTE model (RTE模型的预处理阶段), 227-228
training stage of RTE model (RTE模型的训练阶段), 238
Lexical features (词汇特征)
context as (上下文作为~), 110
in coreference models (共指模型), 301
in event matching (事件匹配), 324
in mention detection (提及检测), 292
of relation extraction systems (关系抽取系统), 314
in sentence and topic segmentation (句子和主题分割), 42-43
Lexical matching (词汇匹配), 212-213
Lexical ontologies, relation extraction and (词汇本体, 关系抽取), 316
Lexical strings (词汇串), 17, 18

Lexicon, of languages (词典，语言)
building (构建), 265-266
dictionary-based approach to subjectivity and sentiment analysis (基于字典的主观性和情感分析), 270, 273
ElixirFM lexicon of Arabic (阿拉伯语 ElixirFM 词典), 20
sets of lexemes constituting (组成一当语素集合), 5
subjectivity and sentiment analysis with (用一进行主观性和情感分析), 262, 270-270
LexPageRank, approach to automatic summarization (LexPageRank, 自动文摘方法), 408, 411
LexTools, for finite-state morphology (LexTools, 用于有限状态形态学), 15
Linear model interpolation, for smoothing language model estimates (线性模型插值，用于平滑语言模型估计), 173
LinearRank algorithm, learning summarization (LinearRank 算法，学习文摘), 408
lingPipe tool, for summarization (lingPipe 工具，用于文摘), 423
Linguistic challenges, in MT (机器翻译中的语言学挑战)
lexical choice (选词), 354-355
morphology and (形态学), 355
word order and (词序), 356
Linguistic Data Consortium (LDC) (语言数据联盟)
corpora for machine translation (机器翻译语料库), 358
evaluating co-occurrence of word between languages (评测语言间词的同现), 337
history of summarization systems (文摘系统历史), 399
OntoNotes corpus (OntoNotes 语料库), 104
on sentence segmentation markers in conversational speech (对话语音中的句子分割标记), 31
summarization frameworks (文摘框架), 422
List questions (列表型问题)
extension to (扩展), 453
QA and, 433
Local collocations, features of supervised systems (局部搭配，有监督系统的特征), 110-111
Localization, of spoken dialog systems (本地化，口语对话系统)
call flow localization (呼叫流程本地化), 514
localization of grammars (文法本地化), 516-517
overview of (概述), 513-514
prompt localization (提示本地化), 516-516
testing (测试), 519-520
training (训练), 517-519
Log-linear models, phrase-based models for MT (对数线型模型，机器翻译短语模型), 348-349
Logic-based representation, applying to RTE (基于逻辑的表示，应用于 RTE), 242-244
Logographic scripts, preprocessing best practices in IR (象形文字书写方式，IR 预处理最佳实践), 371
Long-distance dependencies, syntax-based language models for (长距离依赖，基于句法的语言模型), 180-181
Longman Dictionary of Contemporary English (LDOCE) (朗曼当代英语词典), 104
Lookup operations, dictionaries and (查找操作，词典), 16
Loudness, prosodic cues (音量，韵律提示), 46-47
Low-level features, in event matching (低级特征，事件匹配), 324
Lucene
document indexing with (用来进行文档索引), 483
document retrieval with (用于文档检索), 483-484
IR frameworks (IR 框架), 392
LUNAR QA system (LUNAR 问答系统), 434
Machine learning (机器学习), 参见 Conditional random fields (CRFs)
event extraction and (事件抽取), 322
measuring token frequency (计算词元频率), 369
summarization and (文摘), 405-409
word alignment as learning problem (词对齐作为学习问题), 341-343
Machine translation (MT) (机器翻译)
alignment models (对齐模型), 340
automatic evaluation (自动评测), 334-335
bibliography (文献), 360-363
chart decoding (线图解码), 351-352
CLIR applied to (CLIR 用于~), 380-381
co-occurrence of words and (词的同现), 337-338
coping with model size (控制模型的大小), 348-350
corpora for (语料库), 358
crosslingual QA and (跨语言问答), 454
cube pruning approach to decoding (立方剪枝解

两方法), 347-348
data reorganization and (数据重组), 536
data resources for (数据资源), 356-357
decoding phrase-based models (基于短语模型的解码), 345-347
expectation maximization (EM) algorithm (期望最大化算法), 339-340
future directions (未来方向), 358-359
in GALE IOD, 532-533
hierarchical phrase based models (基于层次短语对模型), 350-351
history and current state of (历史和现状), 331-332
human assessment and (人工评估), 332-334
IBM Model 1 (IBM 模型 1), 338-339
lexical choice (选词), 354-355
linguistic choices (语言学选择), 354
log-linear models and parameter tuning (对数线性模型和调参), 348-349
meaning evaluation (意义评测), 332
metrics (度量指标), 335-337
morphology and (形态学), 355
multilingual automatic summarization and (多语自动文摘), 420
overview of (概述), 331
paraphrasing and (复述), 59
phrase-based models (基于短语的模型), 343-344
programs for (程序), 423
RTE applied to (RTE 用于~), 217-218
in RTTS, 538
sentences as processing unit in (句子作为处理单元), 29
statistical (统计), 参见 Statistical machine translation (SMT)
summary (总结), 358
symmetrization (对称化), 340-341
syntactic models (句法模型), 352-354
systems for (系统), 357-358
in TALES, 538
tools for (工具), 356-357, 392
training issues (训练问题), 197
training phrase-based models (训练基于短语的模型), 344-345
translation-based approach to CLIR (基于翻译的 CLIR 方法), 378-380
tree-based models (基于树的模型), 350
word alignment and (词对齐), 337, 341-343
word order and (词序), 358
MAP (maximum a posteriori) (最大后验)
Bayesian parameter estimation and (贝叶斯参数估计), 173-174
language model adaptation and (语言模型适应), 177-178
MAP (Mean average precision), metrics for IR systems (平均精确率均值, IR 系统度量指标), 389
Marathi (马拉地语), 390
Margin infused relaxed algorithm (MIRA) (MIRA 算法)
methods for sentence or topic segmentation (句子或主题分割方法), 38
unsupervised approaches to machine learning (无监督的机器学习方法), 342
Markov model (马尔可夫模型), 参见 Hidden Markov model (HMM), 34-35
Matches, machine translation metrics (匹配, 机器翻译度量指标), 335
Matching events (匹配事件), 323-325
Mate retrieval setup, relevance assessment and (配对检索设置, 相关评估), 386
MaxEnt model (最大熵模型)
applied to distillation (用于提炼), 480
classifiers for relation extraction (关系抽取分类器), 316-317
classifiers for sentence or topic segmentation (句子或主题分割分类器), 37, 39-40
coreference resolution with (用来进行共指消解), 300-301
language model adaptation and (语言模型适应), 177
memory-based learning compared with (与基于内存的学习比较), 322
mention detection (提及检测), 287-289
modeling using morphological categories (用形态类别建模), 193
modeling without word segmentation (无分词建模), 194
overview of (概述), 181-183
subjectivity and sentiment analysis with (用来进行主观性和情感分析), 274
unsupervised approaches to machine learning (机器学习的无监督方法), 342
Maximal marginal relevance (MMR), in automatic summarization (最大边缘相关, 自动文摘), 395
Maximum a posteriori (MAP) (最大后验)

Bayesian parameter estimation and (贝叶斯参数估计), 173-174
language model adaptation and (语言模型适应), 177-178
Maximum-likelihood estimation (最大似然估计)
Bayesian parameter estimation and (贝叶斯参数估计), 173-174
as parameter estimation language model (作为语言模型参数估计方法), 171-173
used with document models in information retrieval (用于信息检索文档模型), 375-376
MEAD system, for automatic summarization (MEAD 系统, 用于自动文摘), 410-411, 423
Mean average precision (MAP), metrics for IR systems (平均精确率均值, IR 系统度量指标), 389
Mean reciprocal rank (MRR), metrics for QA systems (平均排名倒数, QA 系统度量指标) 462-463
Meaning chunks, semantic parsing and (意义块, 语义分析), 97
Meaning of words (词的意义), 参见 Word meaning
Meaning representation (意义表示)
Air Travel Information System (ATIS) (空旅信息系统), 148
Communicator program (Communicator 程序), 148-149
GeoQuery, 149
overview of (概述), 147-148
RoboCup, 149
rule-based systems for (基于规则的系统), 150
semantic interpretation and (语义解释), 101
software programs for (软件程序), 151
summary (总结), 153-154
supervised systems for (有指导系统), 150-151
Measures (度量), 参见 Metrics
Media Resource Control Protocol (MRCP) (媒体资源控制协议), 504
Meeting Recorder Dialog Act (MRDA) (会议录音对话行为), 31
Memory-based learning (基于内存的学习), 328
MEMT (multi-engine machine translation), in GALE IOD (多引擎机器翻译, GALE IOD), 532-533
Mention detection (提及检测)
Bell tree and (Bell 树), 297
computing probability of mention links (计算提及链的概率), 297-300
data-driven classification (数据驱动分类), 287-289
experiments in (实验), 294-295
features for (特征), 291-294
greedy best-fit decoding (贪心最适解码), 332
MaxEnt model applied to entity-mention relationships (最大熵模型用于实体提及关系), 301
mention-matching features in event matching (事件匹配中的提及匹配特征), 324
overview of (概述), 287
problems in information extraction (信息抽取中的问题), 285-286
in Rosetta Consortium distillation system (Rosetta 协会提炼系统), 480-481
searching for mentions (搜索提及), 289-291
Mention-synchronous process (提及同步过程), 297
Mentions (提及)
entity relations and (实体关系), 310-311
named, nominal, pronominal (命名的, 名词性的, 代词性的), 287
Meronymy (整体部分关系), 310
MERT (minimum error rate training) (最小错误率训练), 349
METEOR, metrics for machine translation (METEOR, 机器翻译度量指标), 336
METONYMY class, ACE (ACE 的 METONYMY 类), 312
Metrics (度量指标)
distillation (提炼), 481-494
graph generation and (图生成), 231
IR, 388
machine translation (机器翻译), 335-337
magnitude of RTE metrics (RTE 指标的绝对值), 233
for multilingual automatic summarization (用于多语自动文摘), 419-420
QA, 462-464
RTE annotation constituents (RTE 标注成分), 223-224
Microsoft, history of QA systems and (微软, 问答系统历史), 435
Minimum error rate training (MERT) (最小错误率训练), 349
Minimum spanning trees (MSTs) (最小生成树), 79-80
Minipar

dependency parsing with (依存分析), 456
rule-based dependency parser (基于规则的依存分析器), 131-132
MIRA (margin infused relaxed algorithm) (MIRA 算法)
methods for sentence or topic segmentation (句子或主题分割方法), 39
unsupervised approaches to machine learning (机器学习的无监督方法), 342
Mixed initiative dialogs, in spoken dialog systems (混合主导对话, 口语对话系统), 509
MLIR, 参见 Multilingual information retrieval (MLIR)
MLIS MUSI summarization system (MLIS MUSI 文摘系统), 389
MMR (maximal marginal relevance), in automatic summarization (最大边缘相关, 自动文摘), 389
Models, information retrieval (信息检索模型)
monolingual (单语), 374-376
selection best practices (选择最佳实践), 377-378
Models, word alignment (词对齐模型)
EM algorithm (EM 算法), 339-340
IBM Model 1 (IBM 模型 1), 338-339
improvements on IBM Model 1 (IBM 模型 1 的改进), 340
Modern Standard Arabic (MSA) (现代标准阿拉伯语), 186-191
Modification processes, in automatic summarization (修改过程, 自动文摘), 389-403
Modifier word, dependency trees and (修饰词, 依存树), 131
Monolingual information retrieval (单语信息检索), 参见 Information retrieval (IR)
document a priori models (文档先验模型), 377
document representation (文档表示), 372-373
index structures (索引结构), 373-374
model selection best practices (模型选择最佳实践), 377-378
models for (模型), 374-376
overview of (概述), 372
query expansion technique (查询扩展技术), 376-377
Monotonicity (单调性)
applying natural logic to RTE (把自然逻辑应用于 RTE), 246
defined (定义), 224
Morfessor package, for identifying morphemes (Morfessor 包, 识别词素), 191-192
Morphemes (词素)
abstract in morphology induction (形态归纳的抽象), 21
automatic algorithms for identifying (识别一的自动算法), 191-192
defined (定义), 4
examples of (例子), 6-7
functional morphology models and (函数式形态模型), 19
Japanese text segmented into (日语文本切分成一), 438
language modeling for morphologically rich languages (形态丰富语言的语言建模), 189
overview of (概述), 5-6
parsing issues related to (相关句法分析问题), 90-91
typology and (类型学), 7-8
Morphological models (形态模型)
automating (morphology induction) (自动形态归纳), 21
dictionary based (基于字典的), 15-16
finite state (有限状态), 16-18
functional (函数式), 19-21
overview of (概述), 13
unification based (基于合一的), 18-19
Morphological parsing (形态分析)
ambiguity and (歧义), 10-13
dictionary lookup and (查字典), 15
discovery of word structure by (自一发现词结构), 3
irregularity and (不规则性), 8-10
issues and challenges (问题和挑战), 8
Morphology (形态学)
categories in language models (语言模型中的形态类别), 192-193
compared with syntax and phonology and orthography (与句法、语音学、正字法比较), 3
induction (归纳), 21
language models for morphologically rich languages (形态丰富语言的语言模型), 189-191
linguistic challenges in machine translation (机器翻译的语言学挑战), 365
parsing issues related to (相关的句法分析问题),

50-52
typology (类型学), 7-8
Morphs (segments) (形态/节)
data sparseness problem and (数据稀疏性问题), 285
defined (定义), 5
functional morphology models and (函数式形态模型), 19
not all morphs can be assumed to be morphemes (并非所有形元都是词素), 7
typology and (类型学), 8
Moses system (Moses 系统)
grow-diag-final method (grow-diag-final 方法), 341
machine translation (机器翻译), 337, 423
MPQA corpus (MPQA 语料库)
manually annotated corpora for English (手工标注英语语料库), 274
subjectivity and sentiment analysis (主观性和情感分析), 263, 272
MRCP (Media Resource Control Protocol) (媒体资源控制协议), 434
MRDA (Meeting Recorder Dialog Act) (会议录音对话行为), 31
MRR (Mean reciprocal rank), metrics for QA systems (平均排名倒数, QA 系统指标), 462-463
MSA (Modern Standard Arabic) (现代标准阿拉伯语), 189-191
MSE (Multilingual Summarization Evaluation) (多语文摘评测), 398, 425
MSTs (minimum spanning trees) (最小生成树), 79-80
Multext Dataset, corpora for evaluating IR systems (Multext 数据集, 用于评测 IR 系统的语料库), 390
Multi-engine machine translation (MEMT), in GALE IOU (多引擎机器翻译, GALE IOU), 532-533
Multilingual automatic summarization (多语自动文摘)
automated evaluation methodologies (自动评测方法学), 415-418
building a summarization system (构建文摘系统), 420-421, 423-424
challenges in (挑战), 409-410
competitions related to (相关竞赛), 424-425
data sets for (数据集), 425-426
devices/tools for (工具), 423
evaluating quality of summaries (评测文摘质量), 412-413
frameworks summarization system can be implemented in (可以实现的框架文档系统), 422-423
manual evaluation methodologies (手工评测方法学), 413-415
metrics for (度量指标), 419-420
recent developments (最新进展), 418-419
systems for (系统), 410-412
Multilingual information retrieval (MLIR) (多语信息检索)
aggregation models (聚合模型), 385
best practices (最佳实践), 385-386
defined (定义), 382
index construction (索引构建), 383-384
language identification (语言识别), 383
overview of (概述), 382
query translation (查询翻译), 384
Multilingual language modeling (多语语言建模), 195-196
Multilingual Summarization Evaluation (MSE) (多语文摘评测), 399, 425
Multimodal distillation (多模态提炼), 493
Multiple reference translations (多个参考翻译), 336
Multiple views, overcoming parsing errors (多视图, 克服分析错误), 102-114
MURAX, 434

n-gram (n 元组)
localization of grammars and (文法的本地化), 516
trigrams (3 元组), 502-503
n-gram approximation (n 元组近似)
language model evaluation and (语言模型评价), 170-171
language-specific modeling problems (特定语言建模问题), 188-189
maximum-likelihood estimation (最大似然估计), 171-172
smoothing techniques in language model estimation (语言模型评价的平滑技术), 173
statistical language models using (使用一的统计语言模型), 170
subword units used with (一用的亚词单元), 192

n-gram models (n元模型)，参见 Phrase indices
AutoSummENG graph (AutoSummENG 图)，419
character models (字符模型)，370
defined (定义)，369-370
document representation in monolingual IR (单语 IR 中的文档表示)，372-373
Naive Bayes (朴素贝叶斯)
classifiers for relation extraction (关系抽取分类器)，315
subjectivity and sentiment analysis (主观性和情感分析)，274
Named entity recognition (NER) (命名实体识别)
aligning views in RTE (RTE 中对齐视图)，233
automatic summarization and (自动文摘)，398
candidate answer generation and (候选答案生成)，448
challenges in RTE (RTE 中的挑战)，212
enrichment stage of RTE model (RTE 模型的富化阶段)，229-230
features of supervised systems (有监督系统的特征)，112
graph generation stage of RTE model (RTE 模型的图生成阶段)，231
impact on searches (对搜索的影响)，464
implementing RTE and (实现 RTE)，227
information extraction and (信息抽取)，100
mention detection related to (与…相关的提及检测)，287
in PSG，125-126
QA architectures and (QA 架构)，439
in Rosetta Consortium distillation system (Rosetta 协会提炼系统)，480
in RTE，221
National Institute of Standards and Technology (NIST) (美国国家标准技术研究院)
BLEU score (BLEU 分数)，295
relation extraction and (关系抽取)，311
summarization frameworks (文摘框架)，422
textual entailment and (文本蕴涵)，211，213
Natural language (自然语言)
call routing (呼叫路由选择)，510
parsing (句法分析)，57-59
Natural language generation (NLG) (自然语言生成)，503-504
Natural language processing (NLP) (自然语言处理)
applications of syntactic parsers (句法分析器的应用)，59
applying to non English languages (用于非英语语言)，218
distillation and (提炼)，参见 Distillation
extraction of document structure as aid in (抽取文档结构，便于…)，89
joint inference (联合推理)，320
machine translation and (机器翻译)，333
minimum spanning trees (MST) and (最小生成树)，79
multiview representation of analysis (分析的多视图表示)，220-222
packages for (包)，253
problems in information extraction (信息抽取问题)，286
relation extraction and (关系抽取)，310
RTE applied to NLP problems (RTE 用于 NLP 问题)，214
RTE as subfield of (RTE 作为一子领域)，参见 Recognizing textual entailment (RTE)
syntactic analysis of natural language (自然语言句法分析)，57
textual inference (文本推理)，209
Natural language processing (NLP), combining engines for aggregation architectures (自然语言处理，为聚合架构组合引擎)，527
bibliography (文献)，548-549
computational efficiency (计算效率)，525-526
data manipulation capacity (数据处理能力)，526
flexible, distributed componentization (灵活的分布式组件化)，524-525
GALE Interoperability Demo case study (GALE 互操作性演示案例研究)，531-537
General Architecture for Text Engineering (GATE) (文本工程通用架构)，528-530
InfoSphere Streams (InfoSphere 流)，530-531
introduction to (介绍)，523-524
lessons learned (得到的教训)，543-542
robust processing (鲁棒处理)，526-527
RTTS case study (RTTS 案例研究)，538-540
summary (总结)，542
TALES case study (TALES 案例研究)，538
Unstructured Information Management Architecture (UIMA) (非结构化信息管理架构)，527-529，542-547
Natural Language Toolkit (NLTK) (自然语言处理工具包)，428

Natural language understanding (NLU) (自然语言理解), 209
Natural logic-based representation, applying to RTE (基于自然逻辑的表示, 应用于 RTE), 245-246
NDCG (Normalized discounting cumulative gain) (归一化折扣累计增益), 389
NER, 参见 Named entity recognition (NER)
Neural network language models (NNLMs) (神经网络语言模型)
  language modeling using morphological categories (采用形态类别进行语言建模), 193
  overview of (概述), 187-188
Neural networks, approach to machine learning (神经网络, 机器学习方法), 342
Neutralization, homonyms and (中性化, 同音异义词), 12
The New York Times Annotated Corpus (纽约时报标注语料库), 425
NewsBlaster, for automatic summarization (NewsBlaster, 用于自动文摘), 411-412
NII Test Collection for IR Systems (NTCIR) (NII 信息检索系统测试集)
  answer scores in QA and (QA 中的回答评分), 453
  data sets for evaluating IR systems (评测 IR 系统的数据集), 390
  evaluation of QA (QA 系统评测), 460-464
  history of QA systems and (QA 系统历史), 434
NIST, 参见 National Institute of Standards and Technology (NIST)
NLG (natural language generation) (自然语言生成), 503-504
NLP, 参见 Natural language processing (NLP)
NLTK (Natural Language Toolkit) (自然语言工具包), 422
NNLMs (neural network language models) (神经网络语言模型)
  language modeling using morphological categories (采用形态类别进行语言建模), 193
  overview of (概述), 187-188
NOMinalization LEXicon (NOMLEX) (名词化词典), 121
Non projective dependency trees (非投射性依存树), 65-66
Nonlinear languages, morphological typology and (非线性语言, 形态类型学), 8
Normalization (规范化)
  Arabic (阿拉伯语), 17
  overview of (概述), 370-371
  tokens and (词元), 4
  Z-score normalization (Z-score 归一化), 385
Normalized discounting cumulative gain (NDCG) (归一化折扣累计增益), 389
Norwegian (挪威语), 461
Noun arguments (名词论元), 144-146
Noun head, of prepositional phrases in PSB (名词中心词, PSB 的介词短语), 127
NTCIR, 参见 NII Test Collection for IR Systems (NTCIR)
Numerical quantities (NUM) constituents, in RTE (数量成分, RTE), 221, 233
Objective word senses (客观性词义), 261
OCR (Optical character recognition) (光学字符识别), 31
One vs. All (OVA) approach (一对多的方法), 136-137
OntoNotes corpus (OntoNotes 语料库), 194
OOV (out of vocabulary) (词汇表以外的)
  coverage rates in language models (语言模型的覆盖率), 170
  morphologically rich languages and (形态丰富的语言), 189-190
OOV rate (未登录词率)
  in Germanic languages (日耳曼语系), 191
  inventorying morphemes and (编制词素表), 192
  language modeling without word segmentation (无分词语言建模), 194
Open-domain QA systems (开放域问答系统), 434
Open Standard by the Organization for the Advancement of Structured Information Standards (OASIS) (结构化信息标准促进组织的开放标准), 527
OpenCCG project (OpenCCG 项目), 21
openNLP, 423
Opinion questions, QA and (意见问题, QA), 433
OpinionFinder
  as rule-based system (作为基于规则的系统), 263
  subjectivity and sentiment analysis (主观性和情感分析), 271-272, 275-276
  subjectivity and sentiment analysis lexicon (主观性和情感分析词典), 262
Optical character recognition (OCR) (光学字符识

别), 31
OPUS project, corpora for machine translation (OPUS 项目, 机器翻译语料库), 358
Ordinal constituent position, in PSG (成分位置顺序), 127
ORG-AFF (organization-affiliation) class (组织机构类), 311-312
Orthography (正字法)
  Arabic (阿拉伯语), 11
  issues with morphology induction (形态归纳问题), 21
Out of vocabulary (OOV) (词汇表以外)
  coverage rates in language models (语言模型覆盖率), 170
  morphologically rich languages and (形态丰富语言), 189-190

PageRank
  automatic summarization (自动文摘), 401
  LexPageRank compared with (与 LexPageRank 比较), 406
  TextRank compared with (与 TextRank 比较), 404
Paradigms (范式)
  classification (分类), 133-137
  functional morphology models and (函数式形态模型), 19
  inflectional paradigms in Czech (捷克语屈折变化范式), 11-12
  inflectional paradigms in morphologically rich languages (形态丰富语言的屈折变化范式), 189
ParaEval
  automatic evaluation of summarization (文摘自动评测), 418
  metrics in (指标), 420

Paragraphs, sentences forming (段落, 句子构成), 29
Parallel backoff (并行回退), 184
Parameter estimation language models (语言模型参数估计)
  Bayesian parameter estimation (贝叶斯参数估计), 173-174
  large-scale models (大规模模型), 174-176
  maximum-likelihood estimation and smoothing (最大似然估计与平滑), 171-173
Parameter tuning (调参), 348-349
Parameters, functional morphology models and (参数, 函数式形态模型), 19

Paraphrasing, parsing natural language and (复述, 分析自然语言), 58-59
Parasitic gap recovery, in RTE (RTE 寄生恢复), 249
parole and langue (de Saussure) (言语和语言, 索绪尔), 13
Parsing (句法分析)
  algorithms for (算法), 70-72
  ambiguity resolution in (歧义消解), 80
  defined (定义), 97
  dependency parsing (依存分析), 79-80
  discriminative models (判别性模型), 84-87
  generative models (生成性模型), 83-84
  hypergraphs and chart parsing (超图和线图分析), 74-79
  natural language (自然语言), 57-59
  semantic parsing (语义分析), 参见 Semantic parsing
  sentences as processing unit in (句子作为处理单位), 29
  shift-reduce parsing (移进归约分析), 72-73
Part of speech (POS) (词性)
  class-based language models and (基于类的语言模型), 178
  features of supervised systems (有监督系统的特征), 113
  implementing RTE and (实现 RTE), 227
  natural language grammars and (自然语言文法), 60
  in PSG, 125-127
  QA architectures and (QA 架构), 438
  in Rosetta Consortium distillation system (Rosetta 协会提炼系统), 481
  for sentence segmentation (用于句子分割), 43
  syntactic analysis of natural language (自然语言句法分析), 57-58
PART-WHOLE relation class (部分整体关系类), 311
Partial order method, for ranking sentences (偏序方法, 为句子排序), 407
Particle language model, subword units in (小品词语言模型, 亚词单元), 192
Partition function, in MaxEnt formula (配分函数, 最大熵公式), 316
PASCAL, 参见 Pattern Analysis, Statistical Modelling and Computational Learning (PASCAL)
Path (路径)
  in CCG, 130
  in PSG, 124, 128-129

in TAG, 130
for verb sense disambiguation (用于动词同义消歧), 112
Pattern Analysis, Statistical Modelling and Computational Learning (PASCAL) (模式分析, 统计建模与计算学习)
evaluating textual entailment (评测文本蕴涵), 313
RTE challenge (RTE挑战), 451-452
textual entailment and (文本蕴量), 311
Pauses, prosodic cues (停顿, 韵律提示), 45-47
Peer surveys, in evaluation of summarization (同行调查, 文摘评测), 412
Penn Treebank (宾州树库)
dependency trees and (依存树), 130-132
parsing issues and (分析问题), 87-89
performance degradation and (性能下降), 147
phrase structure trees in (短语结构树), 68, 70
PropBank and, 125
PER (Position-independent error rate) (位置无关错误率), 335
PER-SOC (personal-social) relation class (个人社会关系类), 321
Performance (性能)
of aggregated NLP (聚合 NLP), 541
combining classifiers to boost (Combination by bothesis) (合并分类器以提升~, 合并假设), 293
competence vs. performance (Chomsky) (语言能力和运用, 乔姆斯基), 13
of document segmentation methods (文档分割方法), 41
evaluating IR (评测 IR), 389
evaluating QA (评测 QA), 462-464
evaluating RTE (评测 RTE), 313-314
feature performance in predicate-argument structure (谓词-论元结构的特征性能), 138-140
Penn Treebank (宾州树库), 147
Period (.), sentence segmentation markers (句号, 句子分割标记), 30
Perplexity (困惑度)
criteria in language model evaluation (语言模型评价标准), 178-179
inventorying morphemes and (语素词素表), 192
language modeling using morphological categories (用形态类别进行语言建模), 193
language modeling without word segmentation (无分词语言建模), 194
Persian (波斯语)
IR and, 390
unification based models (基于合一的模型), 19
Phoenix, 150
Phonemes (音素), 4
Phonology (语音学)
compared with morphology and syntax and orthography (与形态学, 句法, 正字法比较), 3
issues with morphology induction (形态归纳问题), 21
Phrasal verb collocations, in PSG (PSG 中的动词短语搭配), 126
Phrase-based models, for MT (用于机器翻译的基于短语的模型)
coping with model size (处理模型大小), 349-350
cube pruning approach to decoding (立方剪枝解码方法), 347-348
decoding (解码), 346-347
hierarchical phrase-based models (基于层次短语的方法), 350-351
log-linear models and parameter tuning (对数线性模型和调参), 348-349
overview of (概述), 343-344
training (训练), 344-345
Phrase feature, in PSG (PSG 中的短语特征), 124
Phrase indices, tokenization and (短语索引, 词元化), 356, 358-370
Phrase level annotations, for subjectivity and sentiment analysis (短语级标注, 用于主观性和情感分析)
corpus-based (基于语料库的), 267-269
dictionary-based (基于字典的), 264-267
overview of (概述), 264
Phrase Structure Grammar (PSG) (短语结构语法), 124-129
Phrase structure trees (短语结构树)
examples of (例子), 68-70
morphological information in (形态信息), 91
in syntactic analysis (句法分析), 67
treebank construction and (树库构造), 62
Phrases (短语)
early approaches to summarization and (早期的文摘方法), 400
types in CCG (CCG 中的类型), 128-130
PHYS (physical) relation class (PHYS 物理关系类), 321

Pipeline approach, to event extraction (流水线方法，事件抽取), 320-321
Pitch, prosodic cues (音高，韵律提示), 45-47
Pivot language, translation-based approach to CLIR (枢轴语言，基于翻译的 CLIR 方法), 379-380
Polarity (极性)
  corpus-based approach to subjectivity and sentiment analysis (基于语料库的主观性和情感分析方法), 269
  relationship to monotonicity (与单调性的关系), 246
  word sense classified by (由一~分类的词义), 251
Polysemy (多义), 104
Portuguese (葡萄牙语)
  IR and, 390-391
  QA and, 451
  RTE in, 218
POS, 参见 Part of speech (POS)
Position-independent error rate (PER) (与位置无关的错误率), 335
Positional features, approaches to summarization and (位置特征，文摘方法), 401
Positional indices, tokens and (位置索引，词元), 368
Posting lists, term relationships in document retrieval (倒排表，文档检索中的术语关系), 373-374
Pre-reordering, word order in machine translation (预调序，机器翻译中的词序), 356
Preboundary lengthening, in sentence segmentation (边界前延长，句子分割), 47
Precision, IR evaluation measure (精确率，IR 评测指标), 388
Predicate-argument structure (谓词-论元结构)
  base phrase chunks (基本短语块), 132-133
  classification paradigms (分类范式), 133-137
  Combinatory Categorial Grammar (组合范畴语法), 125-130
  dependency trees (依存树), 130-132
  feature performance, salience and selection (特征性能，显著性和选择), 138-140
  FrameNet resources (FrameNet 资源), 118-119
  multilingual issues (多语问题), 146-147
  noun arguments (名词论元), 144-146
  other resources (其他资源), 121-122
  overcoming parsing errors (克服分析错误), 141-144
  overcoming the independence assumption (克服独立性假设), 137-138
  Phrase Structure Grammar (PSG) (短语结构语法), 124-129
  PropBank resources (PropBank 资源), 119-121
  robustness across genres (对各种类型的鲁棒性), 147
  semantic interpretation and (语义解释), 130
  semantic parsing (语义分析), 参见 Predicate-argument structure
  semantic role labeling (语义角色标注), 118
  sizing training data (训练数据大小调整), 140-141
  software programs for (软件程序), 147
  structural matching and (结构匹配), 447-448
  summary (总结), 153
  syntactic representation (语法表示), 123-124
  systems (系统), 122-123
  Tree-Adjoining Grammar (树邻接语法), 130
Predicate context, in PSG (PSG 中的谓词上下文), 129
Predicate feature, in Phrase Structure Grammar (PSG) (短语结构语法中的谓词特征), 124
Prepositional phrase adjunct, features of supervised systems (附属介词短语，有监督系统的特征), 111
Preprocessing, in IR (IR 中的预处理)
  best practices (最佳实践), 371-372
  documents for information retrieval (用于信息检索的文档), 356-367
  tools for (工具), 392
Preprocessing, in RTE (RTE 中的预处理)
  implementing (实现), 227-228
  modeling (建模), 224-225
Preprocessing queries (预处理查询), 483
Preterminals (前终结符), 参见 Part of speech (POS)
Previous role, in PSG (PSG 中的前一角色), 126
PRF (Pseudo relevance feedback) (伪相关反馈)
  as alternative to query expansion (作为查询扩展的替代法), 445
  overview of (概述), 377
Private states (私人状态), 260, 参见 Subjectivity and sentiment analysis
Probabilistic context-free grammars (PCFGs) (概率上下文无关文法)
  for ambiguity resolution (用于消歧), 86-88
  dependency graphs in syntax analysis (句法分析

中的依存图), 56-57
generative parsing models (生成式分析模型), 83-84
parsing techniques (句法分析技术), 78
Probabilistic latent semantic analysis (PLSA) (概率潜在语义分析), 176-177
Probabilistic models (概率模型)
document a priori models (文档先验模型), 377
for document retrieval (用于文档检索), 375
Probability (概率)
history of (历史), 83
MaxEnt formula for conditional probability (条件概率的最大熵公式), 316
Productivity/creativity- and the unknown word problem (能产性/创造性, 未登录词问题), 13-15
Projective dependency trees (投射性依存树)
overview of (概述), 64-65
worst-case parsing algorithm for CFGs (CFG 的最坏情形分析算法), 78
Projectivity (投射性)
in dependency analysis (依存分析中), 64
non projective dependency trees (非投射性依存树), 65-67
projective dependency trees (投射性依存树), 64-65
Prompt localization, spoken dialog systems (提示本地化, 口语对话系统), 514-518
PropBank
annotation of (标注), 447
dependency trees and (依存树), 130-132
limitation of (局限), 122
Penn Treebank and (宾州树库), 123
as resource for predicate-argument recognition (作为谓词-论元识别资源), 119-122
tagging text with arguments (用论元标注文本), 124
Prosody (韵律)
defined (定义), 45
sentence and topic segmentation (句子和主题分割), 45-48
Pseudo relevance feedback (PRF) (伪相关反馈)
as alternative to query expansion (作为查询扩展的替代法), 445
overview of (概述), 377
PSG (Phrase Structure Grammar) (短语结构语法), 124-129
Publications, resources for RTE (出版物, RTE 资源), 252
Punctuation (标点)
in PSG, 129
typographical and structural features for sentence and topic segmentation (用于句子和主题分割的排版和结构特征), 44-45
PUNDIT, 122
Pushdown automaton, in CFGs (CFG 中的下推自动机), 72
Pyramid, for manual evaluation of summarization (金字塔, 用于文摘的手工评测), 413-415

QA, 参见 Question answering (QA)
QUALM QA system (QUALM 问答系统), 434
Queries (查询)
evaluation in distillation (提炼中的评测), 482
preprocessing (预处理), 483
QA architectures and (QA 架构), 439
searching unstructured sources (搜索非结构源), 443-445
translating CLIR queries (翻译 CLIR 查询), 379
translating MLIR queries (翻译 MLIR 查询), 384
Query answering distillation system (查询回答提炼系统)
document retrieval (文档检索), 483-484
overview of (概述), 483
planning stage (规划阶段), 487
preprocessing queries (预处理查询), 483
snippet filtering (片段过滤), 484
snippet processing (片段处理), 485-487
Query expansion (查询扩展)
applying to CLIR queries (应用于 CLIR 查询), 380
for improving information retrieval (用于改进信息检索), 376-377
searching over unstructured sources (搜索非结构源), 445
Query generation, in QA architectures (查询生成, QA 系统结构), 435
Query language, in CLIR (查询语言, CLIR), 386
Question analysis, in QA (QA 中的问题分析), 435, 440-443
Question answering (QA) (问答)
answer scores (回答分值), 450-453
architectures (架构), 435-437
bibliography (文献), 467-473

candidate extraction from structured sources（结构源候选抽取），449-450
candidate extraction from unstructured sources（非结构源的候选抽取），445-449
case study（案例研究），455-460
challenges in（挑战），464-465
crosslingual（跨语言），454-455
evaluating answer correctness（评估回答正确性），461-462
evaluation tasks（评测任务），460-461
introduction to and history of（介绍和历史），433-435
IR compared with（与信息检索比较），365
performance metrics（性能指标），462-464
question analysis（查询分析），440-443
RTE applied to（RTE用于一），215
searching over unstructured sources（搜索非结构源），443-445
source acquisition and preprocessing（获取源与预处理），437-440
summary（总结），465-467
Question mark（?），sentence segmentation markers（问号，句子分割标记），30
Questions, in GALE distillation initiative（问题，GALE提炼计划），475
Quotation marks（" "），sentence segmentation markers（引号，句子分割标记），30

R summarization framework（R文摘框架），422
RandLM toolkit, for machine translation（RandLM工具包，用于机器翻译），357
Random forest language models（RFLMs）（随机森林语言模型）
modeling using morphological categories（用形态类别建模），193
tree-based modeling（基于树的建模），185-186
Ranks methods, for sentences（等级方法，用于句子），407
RDF（Resource Description Framework）（资源描述框架），450
Real-Time Translation Services（RTTS）（实时翻译服务），538-540
Realization stage, of summarization systems（实现阶段，文摘系统）
building a summarization system and（建造文摘系统），421
overview of（概述），400
Recall, IR evaluation measures（召回率，IR评测指标），388
Recall-Oriented Understudy for Gisting Evaluation（ROUGE）（面向召回率的基本意思翻译评测的粗浅研究）
automatic evaluation of summarization（文摘自动评测），415-418
metrics in（指标），420
Recognizing textual entailment（RTE）（识别文本蕴涵）
alignment（对齐），233-236
analysis（分析），220
answer scoring and（回答评分），464
applications of（应用），214
bibliography（文献），254-258
case studies（案例研究），238-239
challenge of（挑战），212-213
comparing constituents in（比较成分），222-224
developing knowledge resources for（开发知识源），249-251
discourse commitments extraction case study（语篇约束抽取案例研究），239-240
enrichment（富化），228-231
evaluating performance of（性能评价），213-214
framework for（框架），219
general model for（一般方法），224-227
graph generation（图生成），231-232
implementation of（实现），227
improving analytics（改进分析），248-249
improving evaluation（改进评测），251-252
inference（推理），236-238
introduction to（介绍），209-210
investing/applying to new problem（用于新问题），249
latent alignment inference（潜在对齐推理），247-248
learning alignment independently of entailment（独立于蕴涵学习对齐），244-245
leveraging multiple alignments（利用多对齐），245
limited dependency context for global similarity（全局相似性的有限依存上下文），247
logical representation and inference（逻辑表示与推理），242-244
machine translation（机器翻译），217-218
multiview representation（多视图表示），220-222
natural logic and（自然逻辑），245-246

in non-English languages (非英语语言), 218-219
PASCAL challenge (PASCAL 挑战), 451
preprocessing (预处理), 227-228
problem definition (问题定义), 210-212
QA and, 215, 433-434
requirements for RTE framework (RTE 框架要求), 215-220
resources for (资源), 252-253
searching for relations (搜索关系), 215-217
summary (总结), 253-254
Syntactic Semantic Tree Kernels (SSTKs) (句法语义树核), 246-247
training (训练), 238
transformation based approaches to (基于转换的方法), 241-242
tree edit distance case study (树编辑距离案例研究), 240-241
Recombination, machine translation and (重合并, 机器翻译), 366
Recursive transition networks (RTNs) (递归转移网络), 150
Redundancy, in distillation (提炼中的冗余性)
detecting (检测), 482-483
overview of (概述), 477-479
reducing (减少), 488-490
Redundancy, in IR (IR 中的冗余性), 468
Reduplication of words, limits of finite-state models (词的重复, 有限状态模型的限制), 17
Reference summaries (参考文摘), 412, 419
Regular expressions (正则表达式)
surface patterns for extracting candidate answers (抽取候选回答的表层模式), 449
in type-based candidate extraction (基于类型的候选抽取), 446
Regular relations, finite-state transducers capturing and computing (正规关系, 有限状态转录机捕捉和计算), 17
Related terms, in GALE distillation initiative (相关术语, GALE 提炼计划), 475
Relation extraction systems (关系抽取系统)
classification approach (分类方法), 312-313
coreference resolution as (共指消解作为一), 311
features of classification based systems (基于分类的系统的特征), 313-315
kernel methods for (核方法), 319
overview of (概述), 310
supervised and unsupervised (有监督和无监督), 317-319
Relational databases (关系数据库), 449
Relations (关系)
bibliography (文献), 327-330
classifiers for (分类器), 318
combining entity and relation detection (合并实体和关系检测), 320
between constituents in RTE (RTE 成分间的一), 220
detection in Rosetta Consortium distillation system (Rosetta 协会提炼系统的一检测), 480-482
extracting (抽取), 310-315
features of classification based extractors (基于分类的抽取器的特征), 313-315
introduction to (介绍), 309-310
kernel methods for extracting (用于抽取的核方法), 319
recognition inspecting searches (识别影响搜索), 444
summary (总结), 325-327
supervised and unsupervised approaches to extracting (用于抽取的有监督和无监督方法), 317-319
transitive closure of (传递闭包), 324-326
types of (类型), 311-312
Relationship questions, QA and (关系型问题, QA), 433, 488
Relevance, feedback and query expansion (相关性, 反馈与查询扩展), 376-377
Relevance, in distillation (相关性, 提炼中的)
analysis of (分析), 492-493
detecting (检测), 488-489
examples of irrelevant answers (不相关的例子), 477
overview of (概述), 477-479
redundancy reduction and (冗余性消除), 488-490
Relevance, in IR (IR 中的相关性)
assessment (评估), 387-388
evaluation (评测), 388
Remote operation, challenges in NLP aggregation (远程操作, NLP 聚合挑战), 524
Resource Description Framework (RDF) (资源描述框架), 458
Resources, for RTE (RTE 的资源)
developing knowledge resources (开发知识源),

249-251
overview of (概述), 253-254
Restricted domains, history of QA systems (受限领域, QA 系统历史), 434
Result pooling, relevance assessment and (结果汇集, 相关性评估), 387
Rewrite rules (in phonology and morphology) (重写规则, 在音学和形态学), 17
RFLMs (Random forest language models) (随机森林语言模型)
modeling using morphological categories (用形态类别建模), 193
tree-based modeling (基于树的建模), 186-189
Rhetorical structure theory (RST), applying to summarization (修辞结构理论, 应用于文摘), 401-404
RoboCup, for meaning representation (RoboCup, 用于意义表示), 149
Robust processing (鲁棒处理)
desired attributes of NLP aggregation (NLP 聚合的期望属性), 526-527
in GATE, 529
in InfoSphere Streams (InfoSphere 流), 531
in UIMA, 529
Robust risk minimization (RRM), mention detection and (鲁棒的风险最小化模型, 提及检测), 287
Roget's Thesaurus (罗氏文类词典)
semantic parsing (语义分析), 104
word sense disambiguation (词义消歧), 105-107
Role extractors, classifiers for relation extraction (角色抽取器, 关系抽取分类器), 316
Romanian (罗马尼亚语)
approaches to subjectivity and sentiment analysis (主观性和情感分析方法), 276-277
corpus-based approach to subjectivity and sentiment analysis (基于语料库的主观性和情感分析方法), 271-272
cross-lingual projections (跨语言投射), 275
dictionary-based approach to subjectivity and sentiment analysis (基于字典的主观性和情感分析方法), 264-266, 276
IR and, 389
QA and, 461
subjectivity and sentiment analysis (主观性和情感分析), 259
summarization and (文摘), 399
Romanization, transliteration of scripts to Latin (Roman) alphabet (罗马化, 到拉丁或罗马字母表的音译转写), 568
Rosetta Consortium system (Rosetta 协会系统)
document and corpus preparation (文档和语料库准备), 480-483
indexing and (索引), 483
overview of (概述), 479-480
query answers and (查询回答), 483-487
ROUGE (Recall Oriented Understudy for Gisting Evaluation) (面向召回率的基本意义翻译评测的组浅研究)
automatic evaluation of summarization (文摘的自动评测), 415-418
metrics in (指标), 420
RRM (robust risk minimization), mention detection and (鲁棒的风险最小化模型, 提及检测), 287
RST (rhetorical structure theory), applying to summarization (修辞结构理论, 应用于文摘), 401-404
RTNs (recursive transition networks) (递归转移网络), 159
RTTS (Real-Time Translation Services) (实时翻译服务), 538-540
Rule-based grammars, in speech recognition (基于规则的文法, 语音识别), 501-502
Rule-based sentence segmentation (基于规则的句子分割), 31-32
Rule-based systems (基于规则的系统)
dictionary-based approach to subjectivity and sentiment analysis (基于字典的主观性和情感分析方法), 270
for meaning representation (用于意义表示), 150
statistical models compared with (与统计模型相比), 292
subjectivity and sentiment analysis (主观性和情感分析), 267
word and phrase-level annotations in subjectivity and sentiment analysis (主观性和情感分析中的词和短语级标注), 263
for word sense disambiguation (用于词义消歧), 105-108
Rules, functional morphology models and (规则, 函数式形态模型), 19
Russian (俄语)
language modeling using subword units (用亚词单元进行语言建模), 198

parsing issues related to morphology (与形态相关的分析问题), 91
unification-based models (基于合一的模型), 19
SALAAM algorithms (SALAAM 算法), 114-115
SALSA project, for predicate argument recognition (SALSA 项目, 用于谓词-论元识别), 122
Sanskrit (梵语)
ambiguity in (歧义), 11
as fusional language (作为屈折语), 8
Zen toolkit for morphology of (Zen 工具包用于其形态分析), 20
SAPT (semantically augmented parse tree) (语义增强分析树), 151
Scalable entailment relation recognition (SERR) (可扩展蕴涵关系识别), 215-217
SCGIS (Sequential conditional generalized iterative scaling) (顺序条件广义迭代演算), 289
Scores (分值)
ranking answers in QA (QA 中的回答排序), 435, 450-453, 458-459
ranking sentences (给句子排序), 407
sentence relevance in distillation systems (提炼系统中的句子相关性), 485-486
Scripts (书写方式)
preprocessing best practices in IR (IR 预处理最佳实践), 371-372
transliteration and direction of (音译和书写方向), 368
SCUs (summarization content units), in Pyramid method (文摘内容单元, 金字塔方法), 414-415
Search component, in QA architectures (搜索组件, QA 架构), 438
Searches (搜索)
broadening to overcome parsing errors (放宽以克服分析错误), 144
in mention detection (提及检测), 289-291
over unstructured sources in QA (QA 中的非结构源), 443-445
QA architectures and (QA 架构), 439
QA vs. IR (问答与信息检索对比), 433
reducing search space using beam search (用柱搜索减少搜索空间), 293-294
for relations (关系), 215-217
SEE (Summary Evaluation Environment) (摘要评测环境), 413
Seeds, unsupervised systems and (种子, 无监督系统), 112
Segmentation (切分)
in aggregated NLP (聚合 NLP), 540
sentence boundaries (句子边界), 参见 Sentence boundary detection
topic boundaries (主题边界), 参见 Topic segmentation
Semantic concordance (SEMCOR) corpus, WordNet (语义索引语料库), 104
Semantic interpretation (语义解释)
entity and event resolution (实体和事件消解), 100
meaning representation (意义表示), 101
overview of (概述), 98-99
predicate-argument structure and (谓词-论元结构), 100
structural ambiguity and (结构歧义), 99
word sense and (词义), 99-100
Semantic parsing (语义分析)
Air Travel Information System (ATIS) (空旅信息系统), 148
bibliography (文献), 154-157
Communicator program (Communicator 程序), 148-149
corpora for (语料库), 104-105
entity and event resolution (实体和事件消解), 100
GeoQuery, 149
introduction to (介绍), 97-98
meaning representation (意义表示), 101, 147-148
as part of semantic interpretation (作为语义解释一部分), 98-99
predicate argument structure (谓词-论元结构), 参见 Predicate argument structure
resource availability for disambiguation of word sense (用于词义消歧的资源可获得性), 104-105
RoboCup, 149
rule-based systems (基于规则的系统), 105-108, 150
semi-supervised systems (半监督系统), 114-116
software programs for (软件程序), 116-117, 151
structural ambiguity and (结构歧义), 99
summary (总结), 151
supervised systems (有监督系统), 109-112,

158-159
system paradigms (系统范式), 101-102
unsupervised systems (无监督系统), 112-114
word sense and (词义), 99-100, 102-105
Semantic role labeling (SRL) (语义角色标注), 参见 Predicate-argument structure
challenges in RTE and (RTE 中的挑战), 212
combining dependency parsing with (与依存分析合并), 132
implementing RTE and (实现 RTE), 227
overcoming independence assumption (克服独立性假设), 137-138
predicate-argument structure training (谓词-论元结构训练), 467
in Rosetta Consortium distillation system (Rosetta 协会提炼系统), 480
in RTE, 221
sentences as processing unit in (句子作为处理单位), 29
for shallow semantic parsing (用于浅层语义分析), 118
Semantically augmented parse tree (SAPT) (语义增强分析树), 151
Semantics (语义学)
defined (定义), 97
explicit semantic analysis (ESA) (显式语义分析), 382
features of classification-based relation extraction systems (基于分类与关系抽取系统的特征), 315-316
finding entity relations (找出实体关系), 310
latent semantic indexing (LSI) (潜在语义索引), 381
QA and, 439-440
structural matching and (结构匹配), 446-447
topic detection and (主题检测), 33
SEMCOR (semantic concordance) corpus, WordNet (语义索引语料库), 104
SEMEVAL, 263
Semi-supervised systems, for word sense disambiguation (半监督系统, 用于词义消歧), 114-116
Semistructured data, candidate extraction from (半结构化数据, 从中抽取候选), 449-450
SemKer system, applying syntactic tree kernels to RTE (SemKer 系统, 把句法树核应用于 RTE), 246
Sense induction, unsupervised systems and (词义归纳, 无监督系统), 112
SENSEVAL, for word sense disambiguation (SENSEVAL, 用于词义消歧), 105-107
Sentence boundary detection (句子边界检测)
comparing segmentation methods (比较分割方法), 40-41
detecting probable sentence or topic boundaries (检测可能的句子或主题边界), 33-34
discourse features (语篇特征), 44
discriminative local classification method for (判别性局部分类方法), 36-38
discriminative sequence classification method for (判别性序列分类方法), 38-39
extensions for global modeling (全局建模扩展), 40
features of segmentation methods (分割方法的特征), 41-42
generative sequence classification method (生成式序列分类方法), 34-36
hybrid methods (混合方法), 39-40
implementing RTE and (实现 RTE), 227
introduction to (介绍), 29
lexical features (词汇特征), 42-43
overview of (概述), 30-32
performance of (性能), 41
processing stages of (预处理阶段), 48
prosodic features (韵律特征), 45-48
speech-related features (语音相关特征), 45
syntactic features (句法特征), 43-44
typographical and structural features (排版和结构特征), 44-45
Sentence-level annotations, for subjectivity and sentiment analysis (句子级标注, 用于主观性和情感分析)
corpus-based approach (基于语料库的方法), 271-272
dictionary-based approach (基于字典的方法), 270-271
overview of (概述), 269
Sentence splitters, tools for building summarization systems (句子拆分器, 建立文摘系统的工具), 423
Sentences (句子)
coherence of sentence-sentence connections (句间联系的连贯性), 402
extracting within-sentence relations (抽取句内关系), 310

methods for learning rank of (学习排序的方法), 407
parasitic gap recovery (寄生缺省), 249
processing for event extraction (处理—以抽取事件), 323
relevance in distillation systems (提炼系统的相关性), 485-486
units in sentence segmentation (句子分割单位), 35
unsupervised approaches to selection (无监督选择方法), 489
Sentential complement, features of supervised systems (句子补足语, 有监督系统的特征), 111
Sentential forms, parsing and (句型, 句法分析), 71-73
Sentiment analysis (情感分析), 参见 Subjectivity and sentiment analysis
SentiWordNet, 282
Sequential conditional generalized iterative scaling (SCGIS) (顺序条件广义迭代演算), 289
SERR (scalable entailment relation recognition) (可扩展蕴涵关系识别), 216-217
Shallow semantic parsing (浅层语义分析)
coverage in semantic parsing (语义分析覆盖率), 102
overview of (概述), 98
semantic role labeling for (语义角色标注), 118
structural matching and (结构匹配), 447
Shalmaneser program, for semantic role labeling (Shalmaneser 程序, 用于语义角色标注), 147
Shift-reduce parsing (移进归约分析), 72-73
SHRDLU QA system (SHRDLU QA 系统), 434
SIGHAN, Chinese word segmentation (SIGHAN, 汉语分词), 194
SIGLEX (Special Group on LEXicon) (词典特别兴趣组), 103
Similarity enablement, relation extraction and (相似能量, 关系抽取), 310
Slovene unification based model (斯洛文尼亚语基于合一的模型), 19
SLU (statistical language understanding) (统计语言理解)
continuous improvement cycle in dialog systems (对话系统中的连续改进循环), 513-513
generations of dialog systems (对话系统四代), 511-512
Smoothing techniques (平滑技术)
Laplace smoothing (Laplace 平滑), 174
machine translation and (机器翻译), 345
n-gram approximation (n 元近似), 172-173
SMT, 参见 Statistical machine translation (SMT)
Snippets, in distillation (片段, 提炼)
crosslingual distillation and (跨语言提炼), 491
evaluation (评测), 492-493
filtering (过滤), 484
main and supporting (主片段和支持片段), 477-478
multimodal distillation and (多模态提炼), 490
planning and (规划), 487
processing (处理), 486-487
Snowball Stemmer (雪球词干分析器), 392
Software programs (软件程序)
for meaning representation (用于意义表示), 151
for predicate-argument structure (用于谓词-论元结构), 147
for semantic parsing (用于语义分析), 116-117
Sort expansion, machine translation phrase decoding (有序扩展, 机器翻译短语解码), 347-348
Sources, in QA (源, QA)
acquiring (获取), 437-440
candidate extraction from structured (结构源候选抽取), 445-450
candidate extraction from unstructured (非结构源候选抽取), 445-449
searching over unstructured (搜索非结构源), 443-445
Spanish (西班牙语)
code switching example (编码切换例子), 31, 195-196
corpus-based approach to subjectivity and sentiment analysis (基于语料库的主观性和情感分析方法), 272
discriminative approach to parsing (句法分析的判别性方法), 91-92
GeoQuery corpus translated into (GeoQuery 语料库翻译为—), 148
IR and, 390-391
localization of spoken dialog systems (口语对话系统的本地化), 513-514, 517-520
mention detection experiments (提及检测实验), 294-296
morphologies of (形态学), 20
polarity analysis of words and phrases (词和短语的极性分析), 269

QA and, 461
resources for semantic parsing (语义分析资源), 122
RTE in, 318
semantic parser for (语义分析器), 151
summarization and (文摘), 398
TAC and, 424
TALES case study (TALES 案例研究), 538
WordNet and, 109
Special Group on LEXicon (SIGLEX) (词典特别兴趣组), 103
Speech (语音)
discourse features in topic or sentence segmentation (主题或句子分割的话语特征), 44
lexical features in sentence segmentation (句子分割的词汇特征), 42
prosodic features for sentence or topic segmentation (句子或主题分割的韵律特征), 45-48
sentence segmentation accuracy (句子分割精确率) 41
Speech generation (语音生成)
dialog manager directing (指导～的对话管理器), 499-500
spoken dialog systems and (口语对话系统), 503-538
Speech recognition (语音识别)
anchored speech recognition (锚点语音识别), 490
automatic speech recognition (ASR) (自动语音识别), 29, 31
language modeling using subword units (用亚词单元进行语言建模), 192
MaxEnt model applied to (最大熵模型用于～), 181-183
Morfessor package applied to (Morfessor 包用于～), 191-192
neural network language models applied to (神经网络语言模型用于～), 188
rule-based grammars in (～中的基于规则文法), 501-502
spoken dialog systems and (口语对话系统), 500-503
Speech Recognition Grammar Specification (SRGS) (语音识别文法说明), 501-502
Speech-to-text (STT) (语音到文本)
data reorganization and (数据重组织), 535-536
in GALE IOD, 532-533
NLP and, 525-524
in RTTS, 538
Split-head concept, in parsing (分解中心词概念, 句法分析), 78
Spoken dialog systems (口语对话系统)
architecture of (体系结构), 503
bibliography (文献), 521-522
call-flow localization (呼叫流程本地化), 514
continuous improvement cycle in (连续改进循环), 512-513
dialog manager (对话管理器), 504-505
forms of dialogs (对话形式), 509-510
functional diagram of (功能框图), 499-500
generations of (代), 510-512
introduction to (介绍), 498
localization of (本地化), 513-514
localization of grammars (文法本地化), 516-517
natural language call routing (自然语言呼叫路由选择), 510
prompt localization (提示本地化), 514-516
speech generation (语音合成), 503-504
speech recognition and understanding (语音识别和理解), 500-503
summary (总结), 520-521
testing (测试), 519-520
training (训练), 517-519
transcription and annotation of utterances (话语的转录和标注), 513
voice user interface (VUI) (语音用户界面), 505-509
Spoken languages, vs. written languages and language models (口语与书面语和语言模型), 194-195
SRGS (Speech Recognition Grammar Specification) (语音识别文法说明), 501-502
SRILM (Stanford Research Institute Language Modeling) (斯坦福研究院语言模型工具)
overview of (概述), 184
SRILM toolkit for machine translation (SRILM 机器翻译工具包), 357
SRL, 参见 Semantic role labeling (SRL)
SSI (Structural semantic interconnections) algorithm (结构语义互连算法), 107-109
SSTKs (Syntactic Semantic Tree Kernels) (句法语义树核), 246-247
Stacks, of hypotheses in machine translation (机器翻译中的假设栈), 345
Stanford Parser, dependency parsing with (斯坦福分析器, 用于依存分析), 468
Stanford Research Institute Language Modeling

(SRILM)(萨里梅研究院语言模型工具)
overview of(概述),184
SRILM toolkit for machine translation(SRILM 机器翻译工具包),357
START QA system(START 问答系统),435-436
Static knowledge, in textual entailment(静态知识,文本蕴涵),210
Statistical language models(统计语言模型)
n-gram approximation(n 元近似),170-171
overview of(概述),159
rule-based systems compared with(与基于规则系统比较),202
spoken vs. written languages and(口语与书面语),194-195
translation with(用于翻译),331
Statistical language understanding(SLU)(统计语言理解)
continuous improvement cycle in dialog systems(对话系统中的连续改进循环),512-513
generations of dialog systems(对话系统的代),511-512
Statistical machine translation(SMT) applying to CLIR(统计机器翻译用于 CLIR),331
cross-language mention propagation(跨语言提及传播),293-294
evaluating co-occurrence of words(评价词的同现),337-338
mention detection experiments(提及检测实验),293-294
Stemmers(词干分析器)
mapping terms to stems(术语映射为词干),370
preprocessing best practices in IR(IR 预处理最佳实践),371
Snowball Stemmer(雪球词干分析器),392
Stems, mapping terms to(词干,把术语映射为~),370
Stop-words, removing in normalization(停止词,规范化时去除),371
Structural ambiguity(结构歧义),89
Structural features(结构特征)
of classification-based relation extraction systems(基于分类的关系抽取系统),314
sentence and topic segmentation(句子和主题分割),44-45
Structural matching, for candidate extraction in QA(结构匹配,用于 QA 中候选抽取),446-448
Structural semantic interconnections(SSI) algorithm(结构语义互连算法),107-109
Structure(结构)
of documents(文档~),参见 Document structure
of words(词~),参见 Word structure
Structured data(结构数据)
candidate extraction from structured sources(结构源候选抽取),449-450
candidate extraction from unstructured sources(非结构源候选抽取),445-449
Structured knowledge(结构知识),434
Structured language model(结构语言模型),181
Structured queries(结构查询),444
STT(Speech-to-text),参见 Speech-to-text(STT)
Subcategorization(次范畴化)
in PSG,125
in TAG,130
for verb sense disambiguation(用于动词词义消歧),112
Subclasses, of relations(关系子类),311
Subject/object presence, features of supervised systems(主/宾语存在性,有监督系统的特征),111
Subject, object, verb(SOV) word order(主语,宾语,动词语序),356
Subjectivity(主观性),260
Subjectivity analysis(主观性分析),260
Subjectivity and sentiment analysis(主观性和情感分析)
applied to English(应用于英语),262
bibliography(文献),278-281
comparing approaches to(方法比较),276-277
corpora for(语料库),262-263
definitions(定义),260-261
document-level annotations(文档级标注),272-274
introduction to(介绍),259-260
lexicons and(词典),262
ranking approaches to(方法排名),274-276
sentence-level annotations(句子级标注),268,270-272
summary(总结),277
tools for(工具),263-264
word and phrase level annotations(词和短语级标注),264-269
Substitution, linguistic supports for cohesion(替代,衔接的语言学支持),401
Subword units, selecting for language models(亚词单元,选择语言模型),191-192

SUMMA
history of summarization systems (文摘系统历史), 399
for multilingual automatic summarization (用于多语自动文摘), 411
summarization frameworks (文摘框架), 423
SUMMARIST, 398
Summarization, automatic (文摘, 自动), 参见 Automatic summarization
Summarization content units (SCUs), in Pyramid method (文摘内容单元, 金字塔方法), 414-415
Summary Evaluation Environment (SEE) (文摘评测环境), 413
SunumBank
history of summarization systems (文摘系统历史), 399
summarization data set (文摘数据集), 425
Supertags, in TAG (TAG中的超级标签), 130
Supervised systems (有监督系统)
for meaning representation (用于意义表示), 150-151
for relation extraction (用于关系抽取), 317-319
for sentence segmentation (用于句子分割), 37
for word sense disambiguation (用于词义消歧), 109-112
Support vector machines (SVMs) (支持向量机)
classifiers for relation extraction (关系抽取分类器), 316-317
corpus-based approach to subjectivity and sentiment analysis (基于语料库的主观性和情感分析方法), 272, 274
mention detection and (提及检测), 287
methods for sentence or topic segmentation (句子或主题分割方法), 37-39
training and test software (训练和测试软件), 135-137
unsupervised approaches to machine learning (无监督的机器学习方法), 342
Surface-based features, in automatic summarization (基于表层的特征, 自动文摘), 400-401
Surface patterns, for candidate extraction in QA (表层模式, 用于QA的候选抽取), 448-449
Surface strings (表层字符串)
input words in input/output language relations (输入/输出语言关系中的输入词), 17
unification based morphology and (基于合一的形态学), 18
SVMs, 参见 Support vector machines
SVO (subject, verb, object) word order (主语、动词、宾语语序), 356
Swedish (瑞典语)
IR and, 390-391
morphologies of (形态学), 29
semantic parsing and (语义分析), 122
summarization and (文摘), 399
SwiRL program, for semantic role labeling (SwiRL程序, 用于语义角色标注), 147
Syllabic scripts (音节文字), 371
Symmetrization, word alignment and (对称化, 词对齐), 340-341
Syncretism (同态), 8
Synonyms (同义词)
answers in QA systems and (QA系统的回答), 442
machine translation metrics and (机器翻译指标), 336
Syntactic features (句法特征)
of classification-based relation extraction systems (基于分类的关系抽取系统), 315
of coreference models (共指模型), 301
of mention detection system (提及检测系统), 292
in sentence and topic segmentation (句子和主题分割), 43-44
Syntactic models, for machine translation (句法模型, 用于机器翻译), 352-354
Syntactic pattern, in PSG (PSG中的句法模式), 126
Syntactic relations, features of supervised systems (句法关系, 有监督系统的特征), 111
Syntactic representation, in predicate argument structure (句法表示, 谓词-论元结构), 123-124
Syntactic roles, in TAG (句法角色, TAG), 130
Syntactic Semantic Tree Kernels (SSTKs) (句法语义树核), 246-247
Syntactic Structures (Chomsky) (句法结构, 乔姆斯基), 88-89
Syntax (句法)
ambiguity resolution (消歧), 80
bibliography (文献), 92-95
compared with morphology and phonology and orthography (与形态学、语音学和正字法比较), 3

context-free grammar (CFGs) and (上下文无关文法), 58-61
dependency graphs for analysis of (用于句法分析的依存图), 53-57
discriminative parsing models (判别性分析模型), 86-87
of documents in IR (IR中的文档), 357-368
generative parsing models (生成式分析模型), 83-84
introduction to (介绍), 57
minimum spanning trees and dependency parsing (最小生成树和依存分析), 75-80
morphology and (形态学), 90-92
parsing algorithms for (分析算法), 70-72
parsing natural language (分析自然语言), 57-59
phrase structure trees for analysis of (短语结构树用于句法分析), 67-70
probabilistic context-free grammars (概率上下文无关文法), 80-83
QA and, 438-440
shift reduce parsing (移进归约分析), 72-73
structural matching and (结构匹配), 445-447
summary (总结), 92
tokenization, case, and encoding and (词元切分、大小写、编码), 87-88
treebanks data driven approach to (树库, 数据驱动方法), 61-63
word segmentation and (分词), 88-90
worst-case parsing algorithm for CFGs (CFG的最坏情形分析算法), 74-79
Syntax-based language models (基于句法的语言模型), 180-181
Synthetic languages, morphological typology and (综合语言, 形态类型学), 7
System architectures (系统结构)
for distillation (提炼), 488
for semantic parsing (语义分析), 101-102
System paradigms, for semantic parsing (系统范式, 用于语义分析), 102-103
Systran's Babelfish program (Systran, Babelfish程序), 332

TAC, 参见 Text Analysis Conferences (TAC)
TAG (Tree Adjoining Grammar) (树邻接语法), 130
TALES (Translingual Automated Language Exploitation System) (跨语际自动语言开发系统), 538
Tamil (泰米尔语)
as agglutinative language (作为黏着型语言), 7
IR and, 390
Task based evaluation, of translation (基于任务的评测, 翻译), 334
TBL (transformation-based learning), for sentence segmentation (基于转换的学习, 用于句子分割), 37
TDT (Topic Detection and Tracking) program (主题检测与跟踪), 32-33, 42, 425-426
Telugu (泰卢固语), 390
Templates, in GALE distillation initiative (模板, GALE提炼计划), 475
Temporal cue words, in PSG (时间提示词, PSG), 127-128
TER (Translation-error rate) (翻译错误率), 337
Term document matrix, document representation in monolingual IR (术语文档矩阵, 单语IR的文档表示), 375
Term frequency inverse document frequency (TF-IDF) (术语频率-倒文档频率)
multilingual automatic summarization and (多语自动文摘), 41
QA scoring and (QA评分), 456-461
unsupervised approaches to sentence selection (无监督的句子选择方法), 489
Term frequency (TF) (术语频率)
TF document model (术语频率文档模型), 373
unsupervised approaches to sentence selection (无监督的句子选择方法), 489
Terms (术语)
applying RTE to unknown (把RTE应用于未知), 217
early approaches to summarization and (早期文摘方法), 490
in GALE distillation initiative (GALE提炼计划), 475
mapping term vectors to topic vectors (术语向量映射为话题向量), 381
mapping to lemmas (把一词映射为原形), 370
posting lists (倒排表), 373-374
Terrier IR framework (Terrier IR框架), 393
Text Analysis Conferences (TAC) (文本分析会议)
competitions related to summarization (与文摘有关的竞赛), 424
data sets related to summarization (与文摘相关的数据集), 426

evaluation of QA systems (QA 系统评测), 460-464
history of QA systems (QA 系统历史), 434
Knowledge Base Population (KBP) (知识库填充), 481-482
learning summarization (学习文摘), 408
Text REtrieval Conference (TREC) (文本检索会议)
data sets for evaluating IR systems (IR 系统评测数据集), 389-390
evaluation of QA systems (QA 系统评测), 460-461
history of QA systems (QA 系统历史), 434
redundancy reduction (冗余消除), 489
Text Tiling method (Hearst) (文本排列方法)
sentence segmentation (句子分割), 42
topic segmentation (主题分割), 37-38
Text to speech (TTS) (文本到语音转换)
architecture of spoken dialog systems (口语对话系统架构), 505
history of dialog managers (对话管理器历史), 534
localization of grammars and (文法本地化), 514
in RTTS, 535
speech generation (语音生成), 503-504
TextRank, graphical approaches to automatic summarization (TextRank, 自动文摘的图方法), 404-406
Textual entailment (文本蕴涵), 参见 Recognizing textual entailment (RTE)
contradiction in (矛盾), 211
defined (定义), 210
entailment pairs (蕴涵对), 210
Textual inference (文本推理)
implementing (实现), 236-238
latent alignment inference (潜在对齐推理), 247-248
modeling (建模), 225-227
NLP and, 209
RTE and, 242-244
TF-IDF (term frequency-inverse document frequency) (术语频率-反文档频率)
multilingual automatic summarization and (多语自动文摘), 411
QA scoring and (QA 评分), 450-451
unsupervised approaches to sentence selection (无监督的句子选择方法), 489
TF (term frequency) (术语频率)
TF document model (术语频率文档模型), 373
unsupervised approaches to sentence selection (无监督的句子选择方法), 489
Thai (泰语)
as isolating or analytic language (作为孤立语或分析型语言), 7
word segmentation in (分词), 4-5
Thot programs, for machine translation (Thot 程序, 用于机器翻译), 423
Tika (Content Analysis Toolkit), for preprocessing IR documents (Tika, 文本分析工具包, 用于预处理 IR 文档), 392
TinySVM software, for SVM training and testing (TinySVM 软件, 用于 SVM 训练和测试), 135-136
Token streams (词元流), 372-373
Tokenization (词元化)
Arabic (阿拉伯语), 12
character *n*-gram models and (字符 *n* 元模型), 379
multilingual automatic summarization and (多语自动文摘), 410
normalization and (规范化), 370-371
parsing issues related to (与~相关的句法分析问题), 87-88
phrase indices and (短语索引), 368-370
in Rosetta Consortium distillation system (Rosetta 协会提炼系统), 480
word segmentation and (分词), 369
Tokenizers, tools for building summarization systems (词元化工具, 用于构建文摘系统), 423
Tokens (词元)
lexical features in sentence segmentation (句子分割的词汇特征), 42-43
mapping between scripts (normalization) (书写系统间的映射, 规范化), 370-371
MLIR indexes and (MLIR 索引), 384
output from information retrieval (信息检索输出), 356
processing stages of segmentation tasks (分割任务的处理阶段), 48
in sentence segmentation (句子分割), 39
translating MLIR queries (翻译 MLIR 查询), 384
in word structure (词结构), 4-5
Top-k models, for monolingual information retrieval (Top-k 模型, 用于单语信息检索), 375
Topic-dependent language model adaptation (主题相关的语言模型适应), 176

Topic Detection and Tracking (TDT) program (主题检测与跟踪计划), 32-33, 43, 425-426
Topic or domain, features of supervised systems (主题或领域，有监督系统的特征), 111
Topic segmentation (主题分割)
  comparing segmentation methods (比较分割方法), 40-41
  discourse features (语篇特征), 44
  discriminative local classification method (判别性局部分类方法), 36-38
  discriminative sequence classification method (判别性序列分类方法), 38-39
  extensions for global modeling (全局建模扩展), 40
  features of (特征), 41-42
  generative sequence classification method (生成式序列分类方法), 34-35
  hybrid methods (混合方法), 39-40
  introduction to (介绍), 29
  lexical features (词汇特征), 42-43
  methods for detecting probable topic boundaries (检测可能的主题边界的方法), 33-34
  overview of (概述), 32-33
  performance of (性能), 41
  processing stages of segmentation tasks (分割任务的处理阶段), 48
  prosodic features (韵律特征), 45-48
  speech related features (语音相关特征), 45
  syntactic features (句法特征), 43-44
  typographical and structural features (排版和结构特征), 44-45
Topics, mapping term vectors to topic vectors (主题，把术语向量映射为主题向量), 381
Traces onto, Treebanks (迹节点，树库), 120-121
Training (训练)
  issues related to machine translation (MT) (机器翻译相关问题), 197
  minimum error rate training (MERT) (最小错误率训练), 348
  phrase-based models (基于短语的模型), 344-345
  predicate-argument structure (谓词-论元模型), 140-141, 447
  recognizing textual entailment (RTE) (识别文本蕴涵), 238
  in RTE, 238
  spoken dialog systems (口语对话系统), 517-519
  stage of RTE model (RTE 模型阶段), 238
  support vector machines (SVMs) (支持向量机), 135-137
Transcription (转写)
  of utterances based on rule-based grammars (基于规则的文法的话语~), 502-503
  of utterances in spoken dialog systems (口语对话系统的话语~), 513
Transducers, finite-state (转录机，有限状态), 16-17
Transformation-based approaches, applying to RTE (基于转换的方法，应用于 RTE), 241-242
Transformation-based learning (TBL), for sentence segmentation (基于转换的学习，用于句子分割), 37
Transformation stage, of summarization systems (文摘系统的转换阶段), 400, 421
Transitive closure, of relations (传递闭包，关系), 324-326
Translation (翻译)
  human assessment of word meaning (词义的人工评估), 333-334
  by machines (通过机器), 参见 Machine translation (MT)
  translation-based approach to CLIR (基于翻译的 CLIR 方法), 378-380
Translation-error rate (TER) (翻译错误率), 337
Translingual Automated Language Exploitation System (TALES) (跨语际自动语言开发系统), 538
Translingual information retrieval (跨语际信息检索), 491
Translingual summarization (跨语际文摘), 398, 参见 Automatic summarization
Transliteration, mapping text between scripts (音译，书写系统间的文本映射), 368
TREC, 参见 Text REtrieval Conference (TREC)
trec-eval, evaluation of IR systems (trec eval, IR 系统评测), 393
Tree-Adjoining Grammar (TAG) (树邻接语法), 130
Tree-based language models (基于树的语言模型), 185-186
Tree-based models, for MT (基于树的模型，用于机器翻译)
  chart decoding (线图解码), 351-352
  hierarchical phrase-based models (基于层次短语

的模型), 350-351
linguistic choices and (语言学选择), 354
overview of (概述), 350
syntactic models (句法模型), 352-354
Tree edit distance, applying to RTE (树编辑距离, 应用于 RTE), 240-241
Treebanks (树库)
data-driven approach to syntactic analysis (句法分析的数据驱动方法), 61-63
dependency graphs in syntax analysis (句法分析依存图), 63-67
phrase structure trees in syntax analysis (句法分析短语结构树), 67-70
traces nodes marked as arguments in PropBank (PropBank 中的跟踪节点作为论元), 120-121
worst-case parsing algorithm for CFGs (CFG 最坏情形分析算法), 77
Trigger models, dynamic self-adapting language models (触发器模型, 动态自适应语言模型), 176-177
Triggers (触发器)
consistency of (一致性), 323
finding event triggers (找出事件触发器), 321-322
Trigrams (三元组), 502-503
Troponymy (方式关系), 310
Tuning sets (调参集), 348
Turkish (土耳其语)
dependency graphs in syntax analysis (句法分析依存图), 62, 65
GeoQuery corpus translated into (GeoQuery 语料库翻译为~), 149
language modeling for morphologically rich languages (形态丰富语言的语言建模), 189-191
language modeling using morphological categories (用形态类别进行语言建模), 192-193
machine translation and (机器翻译), 354
morphological richness of (形态丰富性), 355
parsing issues related to morphology (与形态相关的分析问题), 90-91
semantic parser for (语义分析器), 151
syntactic features used in sentence and topic segmentation (句子和主题分割中使用的句法特征), 43
Type based candidate extraction, in QA (基于类型的候选抽取, QA), 445, 451
Type classifier (类型分类器)
answers in QA systems (QA 系统的回答), 440-442
in relation extraction (关系抽取), 313
Type system, GALE Type System (GTS) (类型系统, GALE 类型系统), 534-535
Typed feature structures, unification-based morphology and (有类型的特征结构, 基于合一的形态学), 18-19
Typographical features, sentence and topic segmentation (排版特征, 句子和主题分割), 44-45
Typology, morphological (类型学, 形态), 7-8

UCC (UIMA Component Container) (UIMA 组件容器), 537
UIMA, 参见 Unstructured Information Management Architecture (UIMA)
Understanding, spoken dialog systems and (理解, 口语对话系统), 500-503
Unicode (UTF-8/UTF-16)
encoding and script (编码和书写方式), 368
parsing issues related to encoding systems (与编码系统相关的分析问题), 89
Unification based morphology (基于合一的形态学), 18-19
Unigram models (Yamron) (一元模型, Yamron), 35-36
Uninflectedness, homonyms and (零屈折变化, 同音异义词), 12
Units of thought, interlingual document representations (思维单位, 中间语言文档表示), 381
Unknown terms, applying RTE to (未知术语, 应用于 RTE), 217
Unknown word problem (未登录词问题), 8, 13-15
Unstructured data, candidate extraction from (非结构数据, 从中抽取候选), 445-449
Unstructured Information Management Architecture (UIMA) (非结构信息管理架构)
attributes of (属性), 528-529
GALE IOD and, 535, 537
overview of (概述), 527-528
RTTS and, 538-542
sample code (代码样例), 542-547
summarization frameworks (文摘框架), 422
UIMA Component Container (UCC) (UIMA 组件容器), 537
Unstructured text, history of QA systems and (非结构文本, QA 系统历史), 434

Unsupervised adaptation, language model adaptation and (无监督适应，语言模型适应), 177
Unsupervised systems (无监督系统)
machine learning (机器学习), 342
relation extraction (关系抽取), 317-319
sentence selection (句子选择), 489
subjectivity and sentiment analysis (主观性和情感分析), 254
word sense disambiguation (词义消歧), 112-114
Update summarization, in automatic summarization (更新文摘，自动文摘), 397
Uppercase (capitalization), sentence segmentation markers (大写，大写化，句子分割标记), 38
UTF-8/UTF-16 (Unicode)
encoding and script (编码和书写方式), 368
parsing issues related to encoding systems (与编码系统有关的分析问题), 69
Utterances, in spoken dialog systems (话语，口语对话系统)
rule-based approach to transcription and annotation (基于规则的转写和标注方法), 502-503
transcription and annotation of (转写和标注), 513

Variable length language models (变长语言模型), 179
Vector space model (向量空间模型)
document representation in monolingual IR (单语IR的文档表示), 372-373
for document retrieval (用于文档检索), 374-375
Verb clustering, in PSG (动词聚类，PSG), 126
Verb sense, in PSG (动词词义，PSG), 126-127
Verb, subject, object (VSO) word order (动词，主语，宾语语序), 368
VerbNet, resources for predicate-argument recognition (VerbNet，谓词-论元识别资源), 121
Verbs (动词)
features of predicate-argument structures (谓词-论元结构特征), 146
relation extraction and (关系抽取), 319
Vietnamese (越南语)
as isolating or analytic language (作为孤立型或分析型语言), 7
NER task in (命名实体识别任务), 287
Views (视图)
in GALE IOD, 534
RTE systems (RTE系统), 220
Vital few (80/20 rule) (重要多数，80/20法则), 14
Viterbi algorithm (Viterbi算法)
applied to Rosetta Consortium distillation system (用于Rosetta联合提炼系统), 480
methods for sentence or topic segmentation (句子或主题分割方法), 39-40
searching for mentions (搜索提及), 291
Vocabulary (词汇表)
indexing IR output (索引IR输出), 356
language models and (语言模型), 169
in morphologically rich languages (形态丰富语言), 180
productivity/creativity and (能产性/创造性), 14
topic segmentation methods (主题分割方法), 38
Voice Extensible Markup Language (语音可扩展标注语言), 参见 VoiceXML (Voice Extensible Markup Language)
Voice feature, in PSG (语态特征，PSG), 124
Voice of sentence, features of supervised systems (句子的语态，有监督系统的特征), 111
Voice quality, prosodic modeling and (语音质量，韵律建模), 47
Voice user interface (VUI) (语音用户界面)
call flow (呼叫流程), 505-505
dialog module (DM) of (对话模块), 507-508
GetService process of (GetService过程), 506-507
grammars of (文法), 508-509
VUI completeness principle (VUI完整性原则), 508-510
VoiceXML (Voice Extensible Markup Language) (语音可扩展标注语言)
architecture of spoken dialog systems (口语对话系统架构), 505
generations of dialog systems (对话系统的代), 511-512
history of dialog managers (对话管理器历史), 504
VUI, 参见 Voice user interface (VUI)

W3C (World Wide Web Consortium) (万维网协会), 504
WASP program, for rule-based semantic parsing systems (WASP程序，用于基于规则的语义分析系统), 151
Web 2.0, accelerating need for crosslingual retrieval (Web 2.0，跨语言检索的巨大需求), 365

WER (word error rate), machine translation metrics and (词错误率，机器翻译指标), 336-337
Whitespace (空格)
  preprocessing best practices in IR (IR 预处理最佳实践), 371
  in word separation (词语分), 368
Wikipedia (维基百科)
  answer scores in QA and (QA 回答评分), 452
  for automatic word sense disambiguation (用于自动词义消歧), 115-116
  crosslingual question answering and (跨语言问答), 455
  as example of explicit semantic analysis (作为显式语义分析例子), 382
  predominance of English in (英语是一主导语言), 438
WikiRelate! program, for word sense disambiguation (WikiRelate! 程序，用于词义消歧), 117
Wiktionary (维基词典)
  crosslingual question answering and (跨语言问答), 455
  as example of explicit semantic analysis (作为显式语义分析例子), 382
Witten-Bell smoothing technique, in language model estimation (Witten-Bell 平滑技术，语言模型估计), 172
Wolfram Alpha QA system (Wolfram Alpha QA 系统), 435
Word alignment, cross-language mention propagation (词对齐，跨语言提及传播), 293
Word alignment, in MT (词对齐，机器翻译)
  alignment models (对齐模型), 340
  Berkeley word aligner (伯克利词对齐程序), 357
  co-occurrence of words between languages (语言间词的同现), 337-338
  EM algorithm (EM 算法), 339-340
  IBM Model 1 (IBM 模型 1), 338-339
  as machine learning problem (作为机器学习问题), 341-343
  overview of (概述), 337
  symmetrization (对称化), 340-341
Word boundary detection (词边界检测), 227
Word-error rate (WER), machine translation metrics and (词错误率，机器翻译指标), 336-337
Word lists (词表), *参见* Dictionary based morphology
Word meaning (词义)
  automatic evaluation (自动评测), 334-335
  evaluation of (评测), 332
  human assessment of (人工评估), 332-334
Word order (词序), 366
Word/phrase-level annotations, for subjectivity and sentiment analysis (词/短语级标注，用于主观性和情感分析)
  corpus based approach (基于语料库的方法), 267-269
  dictionary-based approach (基于字典的方法), 264-267
  overview of (概述), 264
Word segmentation (分词)
  in Chinese, Japanese, Thai, and Korean writing systems (汉语、日语、泰语、朝鲜语书写系统), 4-5
  languages lacking (没有分词的语言), 193-194
  phrase indices based on (基于…的短语索引), 369-370
  preprocessing best practices in IR (IR 预处理最佳实践), 371
  syntax and (句法), 89-90
  tokenization and (词元化), 369
Word sense (词义，义项)
  classifying according to subjectivity and polarity (根据主观性和极性进行区分), 261
  disambiguation (消歧), 105, 152-153
  overview of (概述), 102-104
  resources (资源), 104-105
  rule-based systems (基于规则的系统), 105-109
  semantic interpretation and (语义解释), 99-100
  semi-supervised systems (半监督系统), 114-116
  software programs for (软件程序), 116-117
  supervised systems (有监督系统), 109-112
  unsupervised systems (无监督系统), 112-114
Word sequence (词序列), 159
Word structure (词结构)
  ambiguity in interpretation of expressions (表达式解释的歧义), 16-18
  automated morphology (morphology induction) (自动形态学/形态学归纳), 21
  bibliography (文献), 22-28
  dictionary based morphology (基于字典的形态学), 15-16
  finite-state morphology (有限状态形态学), 16-18
  functional morphology (函数式形态学), 19-21
  introduction to (介绍), 3-4

irregularity in linguistic models (语言模型中的不规则性), 8-10
issues and challenges (问题和挑战), 8
lexemes (语素), 5
morphemes (词素), 5-7
morphological models (形态模型), 15
morphological typology (形态类型学), 7-8
productivity/creativity and the unknown word problem (能产性/创造性和未登录词问题), 13-15
summary (总结), 22
tokens and (词元), 4-5
unification-based morphology (基于合一的形态学), 18-19
units in sentence segmentation (句子分割单位), 33
WordNet
classifying word sense according to subjectivity and polarity (根据主观性和极性区分词义), 261
eXtended WordNet (XWN) (扩展 WordNet), 451
features of supervised systems (有监督系统的特征), 112
hierarchical concept information in (层次概念信息), 109
QA answer scores and (QA 回答评分), 452
as resource for domain-specific information (作为具体领域信息的资源), 122
RTE applied to machine translation (RTE 用于机器翻译), 218
SEMCOR (semantic concordance) corpus (语义一致性语料库), 104-105
subjectivity and sentiment analysis lexicons (主观性和情感分析词典), 262
synonyms (同义词), 335
word sense disambiguation and (词义消歧), 117
World Wide Web Consortium (W3C) (万维网协会), 504
Written languages, vs. spoken languages in language modeling (书面语与语音建模中的口语), 194-195
WSJ, 147

XDC (Crossdocument coreference), in Rosetta Consortium distillation system (跨文档共指, Rosetta 协会提炼系统), 482-483
Xerox Finite-State Tool (XFST) (施乐有限状态工具), 16
XWN (eXtended WordNet) (扩展 WordNet), 451
YamCha software, for SVM training and testing (YamCha 软件, 用于 SVM 训练和测试), 135-136
Yarowsky algorithm, for word sense disambiguation (Yarowsky 算法, 用于词义消歧), 114-115

Z-score normalization, for MLIR aggregation (Z score 归一化, MLIR 聚合), 385
Zen toolkit for morphology, applying to Sanskrit (Zen 形态工具包, 用于梵语), 20
Zero anaphora resolution (零回指消解), 248, 444