《牛津计算语言学手册》[1]导读

冯志伟

一、 计算语言学的发展历史与现状

计算语言学（Computational Linguistics）是当代语言学中的一个新兴学科，在这门学科的发展过程中，曾经在计算机科学、电子工程、语言学、心理学、认知科学等不同的领域分别进行过研究。之所以出现这种情况，是由于计算语言学包括了一系列性质不同而又彼此交叉的学科。这里，我们简要介绍计算语言学的萌芽期、发展期、繁荣期，并分析计算语言学当前的一些特点。

计算语言学的萌芽期

从 20 世纪 40 年代到 50 年代末这个时期是计算语言学的萌芽期。

在“计算语言学’这个术语出现之前，关于语言与计算的研究早就开始了。有四项基础性的研究特别值得注意：

-- 一项是关于马尔可夫模型的研究，

-- 一项是关于可计算性理论和图灵机模型的研究，

-- 一项是关于概率和信息论模型的研究，

-- 一项是关于形式语言理论的研究。

早在 1913 年，俄罗斯著名数学家 A. Markov（马尔可夫）就注意到俄罗斯诗人普希金的叙事长诗《欧根·奥涅金》（Ougene Onegin）中语言符号出现概率之间的相互影响，他试图以语言符号的出现概率为实例，来研究随机过程的数学理论，提出了马尔可夫链（Markov Chain）的思想，他的这个开创性的成果用法文发表在俄罗斯皇家科学院的通报上[2]。后来 A. Markov 的这一思想发展成为在计算语言学中广为使用的马尔可夫模型（Markov model），是当代计算语言学最重要的理论支柱之一。

在计算机出现以前，英国数学家 A. M. Turing（图灵）就预见到未来的计算机将会对自然语言研究提出新的问题。

1936 年，Turing 向伦敦权威的数学杂志投了一篇论文，题为《论可计算数及其在判定问题中的应用》。在这篇开创性的论文中，Turing 给“可计算性”下了一个严格的数学定义，并提出著名的“图灵机”(Turing Machine)的数学模型。“图灵机”不是一种具体的机器，而是一种抽象的数学模型，使用这样的数学模型可以制造一种十分简单但运算能力极强的计算装置，用来计算所有能想象得到的可计算函数。1950 年 10 月，Turing 在《机器能思维吗》一文中指出：“我们可以期待，总有一天机器会同人在一切的智能领域里竞争起来。但是，以哪一点作为竞争的出发点呢？这是一个很难决定的问题。许多人以为可以把下棋之类的极为抽象的活动作为最好的出发点，不过，我更倾向于支持另一种主张，这种主张认为，最好的出发点是制造出一种具有智能的、可用钱买到的机器，然后，教这种机器理解英语并且说英语。这个过程可以仿效小孩子说话的那种办法来进行。”Turing 提出，检验计算机智能高低的最好办法是让计算机来讲英语和理解英语，他天才地预见到计算机和自然语言将会结下不解之缘。

---

[1] The Oxford Handbook of Computational Linguistics，《牛津计算语言学手册》， 外语教学与研究出版社、牛津大学出版社合作出版，2009 年 9 月，北京，ISBN: 978-7-5600-6913-3。

[2] A. A. Markov, Essai d’une recherche statistique sur le texte du roman “Ougene Onegin” illustrant la liaison des epreuve en chain, Bulletin de l’Academie Impériale des Sciences de St-Pétersbourg, 7, 153-162.

20 世纪 50 年代提出的自动机理论来源于 Turing 在 1936 年提出的可计算性理论和图灵机模型，Turing 的划时代的研究工作被认为是现代计算机科学的基础。Turing 的工作首先导致了 McCulloch-Pitts 的神经元（neuron）理论。一个简单的神经元模型就是一个计算的单元，它可以用命题逻辑来描述。接着，Turing 的工作还导致了 Kleene 关于有限自动机和正则表达式的研究。

1948 年，美国学者 Shannon（香农）使用离散马尔可夫过程的概率模型来描述语言的自动机。

Shannon 的另一个贡献是创立了“信息论”（Information Theory）。他把通过诸如通信信道或声学语音这样的媒介传输语言的行为比喻为“噪声信道”（noisy channel）或者“解码”（decoding）。Shannon 还借用热力学的术语“熵”（entropy）来作为测量信道的信息能力或者语言的信息量的一种方法，并且他用概率技术首次测定了英语的熵[3]。

1956 年，美国语言学家 N. Chomsky（乔姆斯基）从 Shannon 的工作中吸取了有限状态马尔可夫过程的思想，首先把有限状态自动机作为一种工具来刻画语言的语法，并且把有限状态语言定义为由有限状态语法生成的语言。这些早期的研究工作产生了“形式语言理论”（formal language theory）这样的研究领域，采用代数和集合论把形式语言定义为符号的序列。Chomsky 在研究自然语言的时候首先提出了“上下文无关语法”（Context-free Grammar），后来，Backus 和 Naur 等在描述 ALGOL 程序语言的工作中，分别于 1959 年和 1960 年也独立地发现了这种上下文无关语法。这些研究都把数学、计算机科学与语言学巧妙地结合起来。

Chomsky 在计算机出现的初期把计算机程序设计语言与自然语言置于相同的平面上，用统一的观点进行研究和界说。他在《自然语言形式分析导论》一文中，从数学的角度给语言提出了新的定义，指出：“ 这个定义既适用于自然语言，又适用于逻辑和计算机程序设计理论中的人造语言”。在《语法的形式特性》[4]一文中，他专门用了一节的篇幅来论述程序设计语言，讨论了有关程序设计语言的编译程序问题，这些问题，是作为“组成成分结构的语法的形式研究”，从数学的角度提出来，并从计算机科学理论的角度来探讨的。他在《上下文无关语言的代数理论》一文中提出：“我们这里要考虑的是各种生成句子的装置，它们又以各种各样的方式，同自然语言的语法和各种人造语言的语法二者都有着密切的联系。我们将把语言直接地看成在符号的某一有限集合 V 中的符号串的集合，而 V 就叫做该语言的词汇.....，我们把语法看成是对程序设计语言的详细说明，而把符号串看成是程序。”在这里乔姆斯基把自然语言和程序设计语言放在同一平面上，从数学和计算机科学的角度，用统一的观点来加以考察，对“语言”、“词汇”等语言学中的基本概念，获得了高度抽象化的认识。

Markov, Turing, Shannon 和 Chomsky 这四位著名学者对于语言和计算关系的探讨，是计算语言学萌芽期最重要的研究成果。

在应用研究中，计算语言学首先在语音的计算方面取得了令人兴奋的成绩。1946 年，könig 等研究了声谱，为尔后语音识别奠定了基础。20 世纪 50 年代，第一个机器语音识别器研制成功。1952 年，Bell 实验室的研究人员研制的语音识别系统，可以识别由一个单独的说话人说出的 10 个任意的数目字。该系统存储了 10 个依赖于说话人的模型，它们粗略地代表了数目字的头两个元音的共振峰。Bell 实验室的研究人员采用选择与输入具有最高相关系数模式的方法来进行语音识别，达到了 97-99％的准确率。

在 20 世纪 50 年代末期到 60 年代中期，处于萌芽期的计算语言学明显地分成两个阵营：

---

[3] C. E. Shannon, A mathematical theory of communication [J], *Bell System Technical Journal*，27: pp 379-423, 1948.

[4] N. Chomsky, Formal properties of grammars, In *Handbook of mathematical Psychology*, Vol. 2, Wiley, New York, 1963.

一个是符号派（symbolic），一个是随机派（stochastic）。

符号派的工作可分为两个方面。

一方面是 50 年代后期以及 60 年代初期和中期 Chomsky 等的形式语言理论和生成句法研究，很多语言学家和计算机科学家热衷于研究剖析算法，1960 年，John Cocke 提出使用二分的上下文无关规则来分析自然语言的 Cocke 算法，接着，Younger 和 Kasami 等分别进行这种算法的研究，形成了 Cocke-Younger-Kasami 算法（简称 CYK 算法），同时提出的分析算法还有自顶向下分析算法、自底向上分析算法、动态规划算法。这样以来，形式语法理论便成为了一种可以计算的理论，被直接应用到自然语言的计算机处理中，成为了自然语言自动剖析的有力工具。美国语言学家 Zelig Harris 研制了最早的完整的英语自动剖析系统“转换与话语分析课题“（Transformation and Discourse Analysis Project，简称 TDAP），这个剖析系统于 1958 年 6 月至 1959 年 7 月在宾夕法尼亚大学研制成功。

符号派另一方面的工作是人工智能的研究。在 1956 年夏天，John McCarthy，Marvin Minsky，Claude Shannon 和 Nathaniel Rochester 等学者汇聚到一起，组成了一个为期两个月的研究组，讨论关于他们称之为“人工智能”（Artificial Intelligence，简称 AI）的问题。尽管有少数的 AI 研究者着重于研究随机算法和统计算法（包括概率模型和神经网络），但是大多数的 AI 研究者着重研究推理和逻辑问题。Newell 和 Simon 研制了“逻辑理论家”（Logic Theorist）和“通用问题解答器”（General Problem Solver）等可以自动进行逻辑推理的系统。早期的自然语言理解系统几乎都是按照他们的观点建立起来的。这些简单的系统把模式匹配和关键词搜索与简单试探的方法结合起来进行推理和自动问答，它们都只能在某一个领域内使用。在 60 年代末期，学者们又研制了更多的形式逻辑系统。

随机派主要是一些来自统计学专业和电子学专业的研究人员。在 20 世纪 50 年代后期，他们使用“贝叶斯方法”（Bayesian method）来解决最优字符识别的问题。1959 年，Bledsoe 和 Browning 建立了用于文本识别的贝叶斯系统，该系统使用了一部大词典，首先计算出词典的单词中所观察的字母系列的似然度，然后把单词中每一个字母的似然度相乘，就可以求出整个字母系列的似然度来。1964 年，Mosteller 和 Wallace 用贝叶斯方法解决了在《联邦主义者》（The Federalist）文章中的原作者的分布问题。

20 世纪 50 年代还出现了基于转换语法的第一个人类语言计算机处理的可严格测定的心理模型；并且还出现了第一个联机语料库：布朗美国英语语料库（Brown corpus），该语料库包含 100 万单词的语料，样本来自不同文体的 500 多篇书面文本，涉及的文体有新闻、中篇小说、写实小说、科技文章等。这些语料是布朗大学（Brown University）在 1963－64 年收集的。

计算语言学萌芽期的这些出色的基础性研究和应用性研究，为计算语言学的理论和技术奠定了坚实的基础。计算语言学从萌芽期一开始，就把不同的学科紧密地结合起来，带有明显的边缘性交叉学科的特点，可以说，计算语言学是在各个相关学科的交融和协作中萌芽成长起来的。

机器翻译是计算语言学最重要的应用领域。在计算语言学的萌芽期，机器翻译研究得到长足的进展。

1946 年，美国宾夕法尼亚大学的 J. P. Eckert（埃克特）和 J.W.Mauchly（莫希莱）设计并制造出了世界上第一台电子计算机 ENIAC，电子计算机惊人的运算速度，启示着人们考虑翻译技术的革新问题。因此，在电子计算机问世的同一年，英国工程师 A.D. Booth（布斯）和美国洛克菲勒基金会副总裁 W. Weaver（韦弗）在讨论电子计算机的应用范围时，就提出了利用计算机进行语言自动翻译的想法。1947 年 3 月 6 日，Booth 与 Weaver 在纽约的洛克菲勒中心会面，Weaver 提出，“如果将计算机用在非数值计算方面，是比较有希望的”。在 Weaver 与 Booth 会面之前，Weaver 在 1947 年 3 月 4 日给控制论学者 N. Wiener（维纳）写

信，讨论了机器翻译的问题，Weaver 说：“我怀疑是否真的建造不出一部能够作翻译的计算机？即使只能翻译科学性的文章（在语义上问题较少），或是翻译出来的结果不怎么优雅（但能够理解），对我而言都值得一试。”可是，Wiener 给 Weaver 泼了一瓢冷水，他在 4 月 30 日给 Weaver 的回信中写道：“老实说，恐怕每一种语言的词汇，范围都相当模糊；而其中表示的感情和言外之意，要以类似机器翻译的方法来处理，恐怕不是很乐观的。”不过 Weaver 仍然坚持自己的意见。1949 年，Weaver 发表了一份以《翻译》为题的备忘录，正式提出了机器翻译问题。在这份备忘录中，他除了提出各种语言都有许多共同的特征这一论点之外，还有两点值得我们注意：

第一，他认为翻译类似于解读密码的过程。他说：“当我阅读一篇用俄语写的文章的时候，我可以说，这篇文章实际上是用英语写的，只不过它是用另外一种奇怪的符号编了码而已，当我在阅读时，我是在进行解码。”

这段话中。Weaver 首先提出了用解读密码的方法进行机器翻译的想法，这种想法成为后来“噪声信道理论”的滥觞，是统计机器翻译的重要的理论依据。

备忘录中还记载了一个有趣的故事，布朗大学数学系的 R. E. Gilmam （吉尔曼）曾经解读了一篇长约一百个词的土耳其文的密码，而他既不懂土耳其文，也不知道这篇密码是用土耳其文写的。

Weaver 认为，Gilman 的成功足以证明解读密码的技巧和能力不受语言的影响，因而可以用解读密码的办法来进行机器翻译。

第二，他认为原文与译文“说的是同样的事情”，因此，当把语言 A 翻译为语言 B 时，就意味着，从语言 A 出发，经过某一“通用语言”（ Universal Language）或“中间语言”（Interlingua），然后转换为语言 B，这种“通用语言”或“中间语言”，可以假定是全人类共同的。

可以看出，Weaver 把机器翻译仅仅看成一种机械的解读密码的过程，他远远没有看到机器翻译在词法分析、句法分析以及语义分析等方面的复杂性。

早期机器翻译系统的研制受到 Weaver 的上述思想的很大影响，许多机器翻译研究者都把机器翻译的过程与解读密码的过程相类比，试图通过查询词典的方法来实现词对词的机器翻译，因而译文的可读性很差，难于付诸实用。

由于学者的热心倡导，实业界的大力支持，美国的机器翻译研究一时兴盛起来。1954 年，美国乔治敦大学在国际商用机器公司（IBM 公司）的协同下，用 IBM-701 计算机，进行了世界上第一次机器翻译试验，把几个简单的俄语句子翻译成英语，接着，苏联、英国、日本也进行了机器翻译试验，机器翻译出现热潮。

1952 年，在美国的 MIT 召开了第一次机器翻译会议，在 1954 年，出版了第一本机器翻译的杂志，这个杂志的名称就叫做 Machine Translation（《机器翻译》）。尽管人们自然语言的计算方面进行了很多的研究工作，但是，直到 20 世纪 60 年代中期，才出现了 computational linguistics（计算语言学）这个术语，而且，在刚开始的时候，这是术语是偷偷摸摸地、羞羞涩涩地出现的。

1965 年 Machine Translation 杂志改名为 Machine Translation and Computational Linguistics（《机器翻译和计算语言学》）杂志，在杂志的封面上，首次出现了“Computational Linguistics”这样的字眼，但是，“and Computational Linguistics”这三个单词是用特别小号的字母排印的。这说明，人们对于“计算语言学”是否能够算为一门真正的独立的学科还没有把握。计算语言学刚刚登上学术这个庄严的殿堂的时候，还带有“千呼万唤始出来，犹抱琵琶半遮面”那样的羞涩，以致于人们不敢用 Machine Translation 同样大小的字母来排印它。当时 Machine Translation 杂志之所以改名，是因为在 1962 年美国成立了“机器翻译和计算语言学学会”（Association for machine Translation and Computational Linguistics），通过改名可以使杂志的

名称与学会的名称保持一致。

根据这些史料，我们认为，远在 1962 年，就出现了“计算语言学”这个学科了，尽管它在刚出现的时候还是偷偷摸摸的，显示出少女般的羞涩。但是，无论如何，计算语言学这个新兴的学科终于萌芽了，她破土而出，悄悄地登上了学术的殿堂。

1964 年，美国科学院成立了语言自动处理谘询委员会（Automatic Language Processing Advisory Committee，简称 ALPAC 委员会），调查机器翻译的研究情况，并于 1966 年 11 月公布了一个题为《语言与机器》的报告，简称 ALPAC 报告[5]，这个报告对机器翻译采取了否定的态度，报告宣称：“在目前给机器翻译以大力支持还没有多少理由”；这个报告还指出，机器翻译研究遇到了难以克服的“语义障碍”（semantic barrier）。在 ALPAC 报告的影响下，许多国家的机器翻译研究低潮，许多已经建立起来的机器翻译研究单位遇到了行政上和经费上的困难，在世界范围内，机器翻译的热潮突然消失了，出现了空前萧条的局面。

美国语言学家 David Hays 是 ALPAC 委员会的成员之一，他参与起草了 ALPAC 报告，在 ALPAC 报告中，他建议，在放弃机器翻译这个短期的工程项目的时候，应当加强语言和自然语言计算机处理的基础研究，可以把原来用于机器翻译研制的经费使用到自然语言处理的基础研究方面，David Hays 把这样的基础研究正式命名为 Computational Linguistics（计算语言学）。所以，我们可以说，“计算语言学”这个学科名称最早出现于 1962 年，而在 1966 年才在美国科学院的 ALPAC 报告中正式得到学术界的承认。

计算语言学的发展期

20 世纪 60 年代中期到 80 年代末期年是计算语言学的发展期。

在计算语言学的发展期，各个相关学科的彼此协作，联合攻关，取得了一些令人振奋的成绩。

统计方法在语音识别算法的研制中取得成功。其中特别重要的是“隐马尔可夫模型”（Hidden Markov Model）和“噪声信道与解码模型”（Noisy channel model and decoding model）。这些模型是分别独立地由两支队伍研制的。一支是 Jelinek，Bahl，Mercer 和 IBM 的华生研究中心的研究人员，另一支是卡内基梅隆大学（Carnegie Mellon University）的 Baker 等，Baker 受到普林斯顿防护分析研究所的 Baum 和他的同事们的工作的影响。AT&T 的贝尔实验室（Bell laboratories）也是语音识别和语音合成的中心之一。

逻辑方法在计算语言学中取得了很好的成绩。1970 年，Colmerauer 和他的同事们使用逻辑方法研制了 Q 系统（Q-system）和“变形语法”（metamorphosis grammar）并在机器翻译中得到应用，Colmerauer 还是 Prolog 语言的先驱者，他使用逻辑程序设计的思想设计了 Prolog 语言。1980 年 Pereira 和 Warren 提出的“定子句语法”（Definite Clause Grammar）也是在计算语言学中使用逻辑方法的成功范例之一。1979 年 Kay 对于“功能语法”（functional grammar）的研究，1982 年 Bresnan 和 Kaplan 在“词汇功能语法”（Lexical Function Grammar，简称 LFG）方面的工作，都是特征结构合一（feature structure unification）研究方面的重要成果，他们的研究引入了“复杂特征”（complex feature）的概念，与此同时，我国学者冯志伟提出了“多叉多标记树形图模型”（Multiple-branched Multiple-labeled Tree Model，简称 MMT 模型），在他设计的多语言机器翻译 FAJRA 中采用了“多标记”（Multiple label）的概念。“多标记”的概念与“复杂特征”的概念实质上是一致的，这些关于自然语言特征结构研究成果，都有效地克服了 Chomsky 短语结构语法的生成能力过强的缺陷。

在这个时期，自然语言理解（natural language understanding）也取得明显的成绩。自然

[5] ALPAC, Language and machines: computer in translation and linguistics, A report by the Automatic Language Processing Advisory Committee, Division of Behavioral Sciences, National Academy of Sciences, National Research Council, Publication 1416, Washington.

语言理解肇始于 Terry Winograd 在 1972 年研制的 SHRDLU 系统，这个系统能够模拟一个嵌入玩具积木世界的机器人的行为。该系统的程序能够接受自然语言的书面指令（例如，“Move the red block on top of the smaller green one”[请把绿色的小积木块移动到红色积木块的上端]），从而指挥机器人摆弄玩具积木块。这是一个非常复杂而精妙的系统。这个系统还首次尝试建立基于 Halliday（韩礼德）系统语法（systemic grammar）的全面的英语语法。Winograd 的模型还清楚地说明，句法剖析也应该重视语义和话语的模型。1977 年，Roger Schank 和他在耶鲁大学的同事和学生们建立了一些语言理解程序，这些程序构成一个系列，他们重点研究诸如脚本、计划和目的这样的人类的概念知识以及人类的记忆机制。他们的工作经常使用基于网络的语义学理论，并且在他们的表达方式中开始引进 Fillmore（费尔摩）在 1968 年提出的关于“深层格”（deep case）的概念。

在自然语言理解研究中也使用过逻辑学的方法，例如 1967 年 Woods 在他研制的 LUNAR 问答系统中，就使用谓词逻辑来进行语义解释。

计算语言学在话语分析（discourse analysis）方面也取得了很大的成绩。基于计算的话语分析集中探讨了话语研究中的四个关键领域：话语子结构的研究、话语焦点的研究、自动参照消解的研究、基于逻辑的言语行为的研究。1977 年，Crosz 和她的同事们研究了话语中的“子结构”（substructure）和话语焦点；1972 年，Hobbs 开始研究“自动参照消解”（automatic reference resolution）。在基于逻辑的言语行为研究中，Perrault 和 Allen 在 1980 年建立了“信念－愿望－意图”（Belief-Desire-Intention，简称 BDI）的框架。

在 1983－1993 年的十年中，计算语言学研究者对于过去的研究历史进行了反思，发现过去被否定的有限状态模型和经验主义方法仍然有其合理的内核。在这十年中，计算语言学的研究又回到了 50 年代末期到 60 年代初期几乎被否定的有限状态模型和经验主义方法上去，之所以出现这样的复苏，其部分原因在于 1959 年 Chomsky 对于 Skinner 的“言语行为”（Verbal Behavior）的很有影响的评论在 80 年代和 90 年代之交遭到了理论上的反对。

这种反思的第一个倾向是重新评价有限状态模型，由于 Kaplan 和 Kay 在有限状态音系学和形态学方面的工作，以及 Church 在句法的有限状态模型方面的工作，显示了有限状态模型仍然有着强大的功能，因此，这种模型又重新得到计算语言学界的注意。

这种反思的第二个倾向是所谓的“重新回到经验主义”；这里值得特别注意的是语音和语言处理的概率模型的提出，这样的模型受到 IBM 公司华生研究中心的语音识别概率模型的强烈影响。这些概率模型和其他数据驱动的方法还传播到了词类标注、句法剖析、名词短语附着歧义的判定以及从语音识别到语义学的联接主义方法的研究中去。

此外，在这个时期，自然语言的生成研究也取得了引人瞩目的成绩。

计算语言学的繁荣期

从 20 世纪 90 年代开始，计算语言学进入了繁荣期。1993 年 7 月在日本神户召开的第四届机器翻译高层会议（MT Summit IV）上，英国著名学者哈钦斯（J. Hutchins）在他的特约报告中指出，自 1989 年以来，机器翻译的发展进入了一个新纪元。这个新纪元的重要标志是，在基于规则的技术中引入了语料库方法，其中包括统计方法，基于实例的方法，通过语料加工手段使语料库转化为语言知识库的方法，等等。这种建立在大规模真实文本处理基础上的机器翻译，是机器翻译研究史上的一场革命，它将会把计算语言学推向一个崭新的阶段。随着机器翻译新纪元的开始，计算语言学进入了它的繁荣期。

在 20 世纪 90 年代的最后五年（1994-1999），计算语言学的研究发生了很大的变化，出现了空前繁荣的局面。这主要表现在如下三个方面。

第一，概率和数据驱动的方法几乎成为了计算语言学的标准方法。句法剖析、词类标注、参照消解、话语处理、机器翻译的算法全都开始引入概率，并且采用从语音识别和信息检索

中借过来的基于概率和数据驱动的评测方法。

第二，计算语言学的应用研究日新月异。由于计算机的速度和存储量的增加，使得在计算语言学的一些应用领域，特别是在语音合成、语音识别、文字识别、拼写检查、语法检查这些应用领域，有可能进行商品化的开发。自然语言处理的算法开始被应用于“增强交替通信”（Augmentative and Alternative Communication，简称 AAC）中，语音合成、语音识别和文字识别的技术被应用于“移动通信”（mobile communication）中。除了传统的机器翻译和信息检索等应用研究进一步得到发展之外，信息抽取（information extraction）、问答系统（question answering system）、自动文摘（text summarization）、术语的自动抽取和标引（term extraction and automatic indexing）、文本数据挖掘（text data mining）、自然语言接口（natural language interaction），计算机辅助语言教学（computer-assisted language learning）等新兴的应用研究都有了长足的进展，此外，自然语言处理技术在多媒体系统（multimedia system）和多模态系统（multimodal system）中也得到了应用。计算语言学的应用研究出现了日新月异的局面。

第三，多语言在线自然语言处理技术迅猛发展。随着网络技术的发展，因特网（Internet）逐渐变成一个多语言的网络世界，因特网上的机器翻译、信息检索和信息抽取的需要变得更加紧迫。目前，在因特网上除了使用英语之外，越来越多地使用汉语、西班牙语、葡萄牙语、德语、法语、俄语、日语、韩国语等英语之外的语言。从 2000 年到 2005 年，因特网上使用英语的人数仅仅增加了 126.9%，而在此期间，因特网上使用俄语的人数增加了 664.5%，使用葡萄牙语的人数增加了 327.3%，使用中文的人数增加了 309.6%，使用法语的人数增加了 235.9%。因特网上使用英语之外的其他语言的人数增加得越来越多，英语在因特网上独霸天下的局面已经打破，因特网确实已经变成了多语言的网络世界，因此，网络上的不同自然语言之间的计算机自动处理也就变得越来越迫切了。网络上多语言的机器翻译、信息检索、信息抽取正在迅猛地发展。语言辨别（language identification）、跨语言信息检索（cross-language information retrieval）、双语言术语对齐（bilingual terminology alignment）和语言理解助手（comprehension aids）等计算语言学的多语言在线处理技术（multilingual on-line processing）已经成为了互联网技术的重要支柱。

在信息时代，科学技术的发展日新月异，新的信息、新的知识如雨后春笋地不断增加，出现了“信息爆炸”（information explosion）的局面。现在，世界上出版的科技刊物达 165000 种，平均每天有大约 2 万篇科技论文发表。专家估计，我们目前每天在因特网上传输的数据量之大，已经超过了整个 19 世纪的全部数据的总和；我们在新的 21 世纪所要处理的知识总量将要大大地超过我们在过去 2500 年历史长河中所积累起来的全部知识总量。而所有的这些信息主要都是以语言文字作为载体的，也就是说，网络世界主要是由语言文字构成的。

为了说明计算语言学的重要性，我们可以把它与物理学做如下的类比：我们说物理学之所以重要，是因为物质世界是由物质构成的，而物理学恰恰是研究物质运动的学科；我们说计算语言学之所以重要，是因为网络世界主要是由语言文字构成的，而计算语言学恰恰是研究语言文字自动处理的学科。

可以预见，知识日新月异的增长和网络技术突飞猛进的进步，一定会把计算语言学的研究推向一个崭新的阶段。计算语言学有可能成为当代语言学中最有发展潜力的学科，计算语言学已经给有着悠久传统的古老的语言学注入了新的生命力，在计算语言学的推动下，语言学有可能真正成为当代科学百花园中的一门名副其实的领先学科。

当前计算语言学发展的四个特点

21 世纪以来，由于互联网的普及，自然语言的计算机处理成为了从互联网上获取知识

的重要手段，生活在信息网络时代的现代人，几乎都要与互联网打交道，都要或多或少地使用计算语言学的研究成果来帮助他们获取或挖掘在广阔无边的互联网上的各种知识和信息，因此，世界各国都非常重视计算语言学的研究，投入了大量的人力、物力和财力。

当前国外计算语言学研究有四个显著的特点：

第一，随着语料库建设和语料库语言学的崛起，大规模真实文本的处理成为计算语言学的主要战略目标：在过去的四十多年中，从事计算语言学系统开发的绝大多数学者，都把自己的目的局限于某个十分狭窄的专业领域之中，他们采用的主流技术是基于规则的句法-语义分析，尽管这些应用系统在某些受限的“子语言”（sub-language）中也曾经获得一定程度的成功，但是，要想进一步扩大这些系统的覆盖面，用它们来处理大规模的真实文本，仍然有很大的困难。因为从自然语言系统所需要装备的语言知识来看，其数量之浩大和颗粒度之精细，都是以往的任何系统所远远不及的。而且，随着系统拥有的知识在数量上和程度上发生的巨大变化，系统在如何获取、表示和管理知识等基本问题上，不得不另辟蹊径。这样，就提出了大规模真实文本的自动处理问题。1990年8月在芬兰赫尔辛基举行的第13届国际计算语言学会议（即COLING'90）为会前讲座确定的主题是：“处理大规模真实文本的理论、方法和工具”，这说明，实现大规模真实文本的处理将是计算语言学在今后一个相当长的时期内的战略目标。为了实现战略目标的转移，需要在理论、方法和工具等方面实行重大的革新。1992年6月在加拿大蒙特利尔举行的第四届机器翻译的理论与方法国际会议（TMI-92）上，宣布会议的主题是“机器翻译中的经验主义和理性主义的方法”。所谓“理性主义”，就是指以生成语言学为基础的方法，所谓“经验主义”，就是指以大规模语料库的分析为基础的方法。从中可以看出当前计算语言学关注的焦点。当前语料库的建设和语料库语言学的崛起，正是计算语言学战略目标转移的一个重要标志。随着人们对大规模真实文本处理的日益关注，越来越多的学者认识到，基于语料库的分析方法（即经验主义的方法）至少是对基于规则的分析方法（即理性主义的方法）的一个重要补充。因为从“大规模”和“真实”这两个因素来考察，语料库才是最理想的语言知识资源。但是，要想使语料库名符其实地成为自然语言的知识库，就有必要首先对语料库中的语料进行自动标注，使之由“生语料”变成“熟语料”，以便于人们从中提取丰富的语言知识。

第二，计算语言学中越来越多地使用机器自动学习的方法来获取语言知识。传统语言学基本上是通过语言学家归纳总结语言现象的手工方法来获取语言知识的，由于人的记忆能力有限，任何语言学家，哪怕是语言学界的权威泰斗，都不可能记忆和处理浩如烟海的全部的语言数据，因此，使用传统的手工方法来获取语言知识，犹如以管窥豹，以蠡测海，这种获取语言知识的方法带有很大的主观性。传统语言学中啧啧地称道的所谓“例不过十不立，反例不过十不破”的朴学精神，貌似严格，实际上，在浩如烟海的语言数据中，以十个正例或十个反例就轻而易举地来决定语言规则的取舍，难道就能够万无一失地保证这些规则是可靠的吗？这是大大地值得怀疑的。当前的计算语言学研究提倡建立语料库，使用机器学习的方法，让计算机自动地从浩如烟海的语料库中获取准确的语言知识。机器词典和大规模语料库的建设，成为了当前计算语言学的热点。这是语言学获取语言知识方式的巨大变化，作为21 世纪的语言学工作者，应该注意到这样的变化，逐渐改变传统的获取语言知识的手段。

第三，计算语言学中越来越多地使用统计数学方法来分析语言数据。使用人工观察和内省的方法，显然不可能从浩如烟海的语料库中获取精确可靠的语言知识，必须使用统计数学的方法。目前，计算语言学中的统计数学方法已经相当成熟，如果我们认真地学会了统计数学，努力地掌握了统计数学，就会使我们在获取语言知识的过程中如虎添翼。目前，在机器翻译中使用统计方法获得了很好的成绩，统计机器翻译（statistical machine translation，简称SMT）成为了机器翻译的主流技术。

2003 年 7 月，在美国马里兰州巴尔的摩（Baltimore, Maryland）由美国商业部国家标

准与技术研究所 NIST/TIDES (National Institute of Standards and Technology) 主持的评比中，来自德国亚琛大学（Achen University）的年青的博士研究生奥赫（F. J. Och）获最好成绩。他使用统计方法，在很短的时间之内就构造了阿拉伯语和汉语到英语的若干个机器翻译系统。伟大的希腊科学家 Archimedes（阿基米德）说过：“只要给我一个支点，我就可以移动地球。”(“Give me a place to stand on, and I will move the world.”)，而这次评比中，Och 也模仿着 Archimedes 说：“只要给我充分的并行语言数据，那么，对于任何的两种语言，我就可以在几小时之内给你构造出一个机器翻译系统。”(“Give me enough parallel data, and you can have translation system for any two languages in a matter of hours.”) 这反映了新一代的机器翻译研究者朝气蓬勃的探索精神和继往开来的豪情壮志。看来，Och 似乎已经找到了机器翻译的有效方法，至少按照他的路子走下去，也许有可能开创出机器翻译研究的一片新天地，使我们在探索真理的曲折道路上看到了耀眼的曙光。过去我们研制一个机器翻译系统往往需要几年的时间，而现在采用 Och 的方法构造机器翻译系统只要几个小时就可以了，研制机器翻译系统的速度已经大大地提高了。这是当前计算语言学中令人兴奋的新进展。

第四，计算语言学中越来越重视词汇的作用，出现了“词汇主义”（lexicalism）的倾向。词汇信息在自然语言的计算机处理中起着举足轻重的作用，单词之间的相似度（similarity）的计算、词汇的搭配关系（lexical collocation）和词汇联想关系（lexical association）的自动获取、动词的次范畴框架（subcategorization frame）的自动获取、词汇语义学（lexical semantics）等都是当前计算语言学研究的热点。在统计方法中引入了词汇信息，可以大大地提高统计分析的精确度，在句法分析中引入词汇信息，可以减少结构上歧义，提高句法分析的效率。机器可读词典和词汇知识库成为了自然语言处理最关键、最重要的语言资源。

我国计算语言学已经取得不少成绩，但是，与国际水平相比，差距还很大。计算语言学是国际性的学科，我们应该参与到国际计算语言学的研究中去，用国际的水平和国际的学术规范来要求我们的研究。这样，学习和了解国外计算语言学的研究成果和最新动态，就显得非常重要了。

《牛津计算语言学手册》由 Ruslan Mitkov（米特科夫）教授主编，收录了包括语言学家、计算机专家和语言工程人员在内的 49 位学者撰写的 38 章针对计算语言学主要领域的综述性文章，各章的写作风格力求一致，使得全书前后关联，浑然一体，可读性强。《牛津计算语言学手册》内容丰富，全面地反映了国外计算语言学的最新成果，是我们了解国外计算语言学发展动向的一个窗口，正好满足了我们学习和了解国外计算语言学的研究成果和最新动态的要求。

本书主编 Ruslan Mitkov 是计算语言学家与语言工程专家，他毕业于德国德累斯顿大学（Dresden University），现为由英国伍尔弗汉普顿大学（University of Wolverhampton）教授，他的研究兴趣是回指消解、机器翻译和自动索引，曾于 2002 年出版过《回指消解》(Anaphora Resolution）的专著。著名计算语言学家 Martin Kay（马丁·凯伊）为本书作序，Martin Kay 是美国斯坦福大学语言学教授，曾任计算语言学会主席、国际计算语言学委员会主席，是国际计算语言学界的领军人物。

二、 本书主要内容

本书内容分三大部分：1. 与计算语言学有关的语言学基础理论（1-9 章），2. 计算语言学中自然语言的处理、方法与资源（10-26 章），3. 计算语言学的应用（27-38 章）等三大部分，几乎涵盖了计算语言学的所有领域。书末有按照字母顺序编排的计算语言学术语表，每个术语均有简要的定义和解释，便于读者查询。下面我们分别介绍各章的内容。

第一章“音系学”（phonology）介绍了描写音系学和计算音系学的基本知识，着重介绍了非线性音系学中的有限状态模型，音位的特征-值矩阵描述方法以及音系学研究中的计算工具。

第二章“形态学”（morphology）介绍了诸如语素、词、屈折、派生等形态学的基本知识，分析了形态学对于音系学的影响，着重介绍计算形态学中的有限状态分析方法，并介绍了双层形态学和双层规则的形式化描述方法，最后介绍了结构段形态学。

第三章“词典学”（lexicography）首先简要地回顾了词典学的发展历史，接着讨论了人编词典在计算机应用中的不足，说明了计算词典学对于传统的词典编纂技术提出了挑战。本章着重讨论了词汇在计算语言学中的功能以及计算技术在词典编纂中的作用。说明了计算技术改变了词典编纂工作的面貌，为新型词典的编纂提供了有力的技术手段。本章强调地指出，计算机辅助的词典编纂应该成为今后词典编纂工作的发展方向。

第四章“句法学”（syntax）首先列举了一些有趣的句法现象，分析了这些现象在计算上的意义，接着介绍正则语法和有限状态语法、上下文无关的短语结构语法、转换语法、扩充转移网络、各种基于约束的特征结构语法（功能语法、词汇功能语法、中心语驱动的短语结构语法、PATR 语法），最后，介绍了两种在语言学上和计算上有意义的句法框架（广义短语结构语法、树邻接语法）。

第五章“语义学”（semantics）集中介绍计算语义学的基本内容。首先讨论语义的表示问题，介绍了语义的高阶逻辑（higher-order-logic）表示法和语义的特征值矩阵（Attribute-Value Matrix）表示法；接着讨论句法语义接口，介绍了“并行对应模型”（Parallel Correspondence Model，简称 PCM）；针对 Frege 的“组成性原则”（principle of compositionality），介绍了“非组成性的语义学”；最后，介绍了语义解释的动态模型。

第六章“话语”（discourse）首先列举了一些话语平面的现象，阐明了“话语”研究的对象是句子之间的关联问题，计算语言学中的话语研究要揭示句子之间关联的机制。接着讨论参照表示（referring expressions）和话语结构（discourse structure），说明参照表示的工作原理和参照表示的选择方法，并讨论主题（theme）与述题（rheme），话题（topic）与焦点（focus），以及预设（presupposition）、蕴含（implicature）等问题。最后讨论“话语树”（discourse tree），介绍了“修辞结构理论”（rhetorical structure theory）和“中心理论”（centering theory）。

第七章“语用学和对话”（pragmatics and dialogue）讨论语用学及其在计算机对话模型中的应用。首先介绍言语行为（speech act）、言外力（illocutionary force）、合作原则（cooperative principle，简称 CP）、关联（relevance）等语用学的基本概念，并且介绍了意图（intention）、信念（belief）、知识（knowledge）和推论（inference）等与概念表达有关的问题。着重讨论了计算语用学中的对话模型（dialogue model），说明了从话语行为到对话行为的计算机制，并介绍了话语的管理模型（dialogue management models）。

第八章“形式语法与形式语言”（formal grammars and languages）介绍形式语言理论的基本知识，分别论述了形式语法和自动机，把形式语法看成是语言的生成装置，把自动机看成语言的识别装置。为了便于文科读者理解本章的内容，对于一些基本概念都给出了定义和实例，但是对于一些基本的结论则不在数学上加以证明。首先介绍了 Chomsky 的形式语法，给出了形式语法的 Chomsky 分类，分别讨论了上下文无关语言（context-free languages）、线性和正则语言（linear and regular languages）、半线性语言（semilinear languages）、上下文有关语言（context-sensitive languages）、柔性上下文有关语言（mildly context-sensitive languages）。接着介绍自动机理论，分别讨论了有限自动机（finite automata）、下推自动机（pushdown automata）、线性有界自动机（linear bounded automata）、图灵机（Turing machine）。

第九章“计算复杂性”（complexity）介绍自然语言处理中的计算复杂性问题。首先介绍复杂性的度量方法和复杂性的类别，分别讨论了多项式算法（Polynomial algorithm，简称 P）

和非确定多项式算法（Nondeterministic Polynomial algorithm，简称 NP），并介绍了自然语言处理中关于“NP 完全问题”（NP-complete problem）的一些研究。接着，讨论正则语言问题的计算复杂性，介绍了确定性（determinism）和非确定性（non-determinism）的概念、线性（linearity）和有限状态特性（finite-stateness）的概念，说明了有限状态方法的可应用性。然后，讨论上下文无关语言的计算复杂性，介绍了基于搜索的上下文无关识别（search-based context-free recognition）、自顶向下识别（top-down recognition）、线性时间与空间中的确定性语法识别（deterministic grammar recognition in linear time and space）。最后，讨论了概率语法和启发式搜索、并行处理和实际效用等问题，说明计算复杂性分析在理解自然语言的复杂性以及在建立实际的自然语言处理系统中的用途。

第十章“文本切分”（text segmentation）介绍两方面的内容：一方面是“词例还原”（tokenization），一方面是“句子分离”（sentence splitting）。词例还原的目标是把文本中的单词、标点符号、数字、字母数字字符切分出来，以便进行进一步的处理。本章分别介绍了单词自动切分、缩写切分（例如，“Mr., Dr., kg.”中的黑点）、连字符处理（例如，“self-asessment, forty-two, F-16”中的连字符）的技术，并且讨论了汉语和日语等东方语言中“词例还原”（也就是“切词”）的特殊问题。句子分离的目标是把文本中的句子分离出来，在很多自然语言处理系统中，都需要进行句子分离。本章介绍了基于规则的句子分离、基于统计的句子分离、非规范输入文本中的句子分离等技术。

第十一章“词类标注”（part-of-speech tagging）介绍了词类标注器（POS tagger）的设计技术以及兼类词的消歧（disambiguation）方法。简要回顾了词类标注发展的历史，介绍了基于局部性手写规则的词类标注器、基于 n-元语法的词类标注器、基于隐马尔科夫模型（Hidden Markov Models）的词类标注器、基于机器学习的词类标注器、基于全局性手写规则的词类标注器、基于混合方法的词类标注器，重点介绍了手工消歧语法（handwritten disambiguation grammars）。

第十二章“句法剖析”（parsing）介绍了自动句法剖析的基本概念和关键技术。句法剖析的深度因自然语言处理的具体要求的不同而不同，有浅层的句法剖析（shallow parsing），也有深层的句法剖析（deep parsing）。本章首先介绍了浅层句法剖析，这种剖析只要把句子剖析为语块（chunks）就可以了。接着，介绍了依存剖析（dependency parsing）。在介绍上下文无关剖析（context-free parsing）时，比较详细地讨论了 CYK 算法、自底向上剖析、左角分析法、自底向上的活性线图分析法（bottom-up active chart）。在介绍基于合一的剖析（unification-based parsing）时，讨论了特征-值矩阵。剖析时可能得到若干个结果，因此，还讨论了剖析结果的歧义消解问题。最后，讨论了剖析算法准确性的评测、剖析程序的效率以及剖析语法覆盖面的度量方法等问题。

第十三章“词义消歧”（word-sense disambiguation，简称 WSD）讨论如何在上下文中确定多义词的准确意义。首先介绍了在计算语言学研究的早期提出的 WSD 的优选语义学方法、词专家剖析方法，这些方法由于缺乏可供使用的词汇资源，出现了“知识获取的瓶颈问题”（knowledge acquisition bottleneck），这些问题由于大规模词汇库和知识库的出现而得到缓解，又由于统计方法和机器学习方法的应用而可以从语料库中获取精确的数据来加以避免。近年来，在 WSD 中普遍使用基于词典的方法、联结主义方法（connectionist）、统计方法、机器学习方法，取得了很大的进步。最后讨论 WSD 的评测，介绍了 SENSEVAL 的评测活动，并介绍 WSD 的一些实际应用。

第十四章“回指消解”（anaphora resolution）首先列举了一些回指现象，说明了回指现象的各种变体。接着讨论回指消解所需要的知识源、回指消解的过程、回指消解在自然语言处理中的应用。最后回顾了回指消解研究的发展历史和现状，讨论了今后回指消解研究中应当注意的问题。

第十五章“自然语言生成”（natural language generation，简称 NLG）介绍了自然语言生成研究的理论和实践问题，力图说明在人们的心智上以及在计算机中，语言究竟是怎样产生出来的。自然语言生成是一个知识密集的问题，可以从语言学、认知科学和社会学的角度来探讨。可以把自然语言生成看成一个映射问题，也可以把它看成一个选择问题，还可以把它看成一个规划问题。自然语言生成可以分为四个问题：宏观规划（macroplanning）、微观规划（microplanning）、表层实现（surface realization）、物理表达（physical presentation）。对于宏观规划，介绍了说话内容的规划、文本的规划、以及使用修辞结构理论的规划方法，对于微观规划，着重介绍了词汇生成的问题，最后介绍了表层生成的技术。

第十六章“语音识别”（speech recognition）研究如何把作为声学信号的声波转换为单词的序列。现在，最有效的语音识别方法是语音信号统计建模的方法。本章简要地介绍了语音识别中的主要方法和技术：声学语音信号的建模，语音识别中的词汇表示，语音识别中的语言模型，解码。重点介绍独立于说话人的大词汇量连续语音识别（large-vocabulary continuous speech recognition，简称 LVCSR）的最新的技术。目前，语音识别主要应用于自动听写机的设计、口语对话系统、语音文献的自动转写、语音信息检索等领域中。最后讨论了语音识别技术将来的研究方向。

第十七章“文本-语音合成”（text-to-speech synthesis,，简称 TTS）介绍文本-语音合成的最新成果。TTS 既涉及到自然语言处理技术，也涉及到数字信号处理的技术。本章主要从自然语言处理的角度来介绍 TTS。首先介绍 TTS 系统的概貌以及它的商业应用价值，然后描述 TTS 系统的功能结构以及 TTS 系统的组成部分，TTS 系统中的自动形态-句法分析、自动语音分析、自动韵律生成，说明了如何从文本中近似地计算语音的声调和时长。最后，介绍声波生成的两种技术：规则合成技术（synthesis by rules）与毗连合成技术（concatenative synthesis）。

第十八章“有限状态技术”（finite-state technology）首先举例介绍有限状态语言、词汇转录机、重写规则等基本概念，然后介绍基本正则表达式的运算方法和复杂的正则表达式，最后讨论有限状态网络的形式特性。

第十九章“统计方法”（statistical methods）介绍计算语言学中的统计方法。目前，统计方法成为自然语言处理的主流方法。本章首先介绍数理统计的基本概念（如，样本空间、概率测度、随机变量、条件概率、熵、随机过程）以及如何把它们在应用于自然语言的模拟问题，分别介绍了隐马尔科夫模型（hidden Markov models）和最大熵模型（maximum-entropy models），最后介绍了这些模型的一些技术细节，如，韦特比搜索（Viterbi search）、最大熵方程（maximum-entropy equation）等。

第二十章“机器学习”（machine learning）介绍了如何通过有指导的训练实例（supervised training examples）来自动地获取语言资源中蕴含的决策树（decision-tree）和规则（rules），描述了怎样从经过标注的训练实例中进行推理的各种算法和知识表达技术，并介绍了如何使用已经获得的知识来进行分类的基于实例的分类方法（instance-based categorization），较详细地介绍了 k-邻近分类算法（k nearest-neighbour categorization algorithm）。这些机器学习的技术可以应用来解决计算语言学中的形态分析、词类标注、句法剖析、词义自动歧义消解、信息抽取、前指消解等各种各样的问题。

第二十一章“词汇知识的获取”（lexical knowledge acquisition）首先介绍了词汇知识自动获取的一些背景，包括词汇知识的形式、词汇知识获取的资源和工具，单词的共现和相似度，然后介绍了从语料库中自动获取词汇的搭配关系（lexical collocation）和联想关系（lexical association）的方法，词汇相似度（similarity）计算与叙词表（thesaurus）构建的方法，动词的次范畴框架（subcategorization frame）的获取方法，分析了词汇语义学（lexical semantics）和词汇知识获取的关系，最后介绍了从机器可读的词典中获取词汇知识的方法。由于在自然

语言处理中越来越重视词汇知识的作用，自然语言处理的形式模型中越来越多地采用“词汇化”（lexicalized）的方法，词汇知识的自动获取是当前计算语言学研究的亮点之一。。

第二十二章“评测”（evaluation）专门讨论自然语言处理系统的评测问题。评测是推动自然语言处理研究发展的一个重要手段，评测的结果对于自然语言处理系统的投资者、开发者和使用者都是很有价值的。在自然语言处理技术发展的早期主要使用基于技术的评测（technology-based evaluation），在自然语言处理技术比较成熟的时候，就可以使用以用户为中心的评测（user-centred evaluation）。根据评测时的输入与输出，评测技术又可以分为分析成分的评测（evaluation of analysis components）、输出技术的评测（evaluation of output technologies）和交互系统的评测（evaluation of interactive systems）。分析成分的评测把语言映射为它的内部表达作为输出（例如，有标记的片段、树形图、抽象的意义表达式等）。输出技术的评测要把处理的结果用具体的语言表示出来（例如，文摘、生成的文本、翻译的译文等），这种评测可以分别使用内部评测指标（intrinsic measures）和外部评测指标（extrinsic measures）来进行。交互系统的评测容许用户与系统进行交互。本章总结了评测的各种技术，并指出它们的优点和缺点。

第二十三章“子语言和可控语言”（sublanguage and controlled language）首先讨论了在限定语义领域中的计算语言学，指出了在当前的水平之下，在某些限定领域中应用自然语言处理技术的必要性。然后举例说明了某些自发地形成的子语言，分析了子语言的特性，讨论了子语言在机器翻译、文本数据抽取、自然语言生成、自动文摘中应用的问题。接着讨论可控语言，分析了使用可控语言的必要性和局限性，介绍了可控语言的一个实例--简化英语AECMA。最后讨论子语言与可控语言的关系，分析了把子语言转变为可控语言的途径。

第二十四章“语料库语言学”（corpus linguistics）主要讨论了语料库在自然语言处理中的应用问题。首先从语料的抽样框架、语料的代表性、语料的平衡性等方面说明了建立语料库的基本要求，简要地回顾了语料库的发展历史，然后着重地讨论了语料库的标注（annotation）问题。标注过的语料库的优点是：开发和研究上的方便性，使用上的可重用性，功能上的多样性，分析上的清晰性。学术界对于语料库标注的批评主要来自两方面：一方面认为，语料库经过标注之后失去了客观性，所得到的语料库是不纯粹的；另一方面认为，手工标注的语料库准确性高而一致性差，自动或半自动的标注一致性高而准确性差，语料库的标注难以做到两全其美，而目前大多数的语料库标注都需要人工参与，因而很难保证语料库标注的一致性。在分析了语料库在自然语言处理中的应用问题之后，作者指出，不论标注过的语料库还是没有标注过的语料库，在自然语言处理中都是有用的，语料库语言学有助于计算语言学的发展。

第二十五章“知识本体”（ontology）讨论了知识本体及其在自然语言处理中的应用。首先分别介绍了哲学传统的知识本体、认知和人工智能传统的知识本体、语言学传统的知识本体，并讨论了语言学中的知识本体与词汇语义学的关系。然后说明，在自然语言处理中，知识本体可以用来帮助系统进行语言的结构分析（例如，英语中的 PP 附着问题、错拼更正、句法检错、语音识别），也可以用来进行局部的自然语言理解（例如，信息检索中的问题搜索、文本分类），并具体说明了知识本体在信息检索、信息抽取、自动文摘、语义相似度计算、词义消歧中的应用。

第二十六章“树邻接语法”（tree-adjoining grammar，简称 TAG）介绍一种局部化的语法形式模型：树邻接语法（TAG）和词汇化的树邻接语法（lexicalized tree-adjoining grammar，简称 LTAG）。首先讨论上下文无关语法 CFG 的局部化问题，说明 TAG 与 CFG 的不同：TAG 以句法结构树作为核心操作对象，在树的基础上来组织语言知识，它的产生式规则也对应着树结构，它以线性的一维形式来表达二维的树结构，而 CFG 以符号串作为操作对象，CFG 是一个基于符号串的形式语法，而 TAG 是基于树的形式语法。然后讨论上下文无关语法 CFG

的词汇化问题，介绍了 LTAG。LTAG 对于 TAG 的扩充主要在于把每一个初始树（initial tree）和辅助树（auxiliary tree）都与某一个或某一些叫做“抛锚点”（anchor）的具体单词关联起来。最后讨论 LTAG 的一些重要特性及其与别的形式系统的关系。

第二十七章“机器翻译：总体回顾”（machine translation: general overview）介绍了从 20 世纪 50 年代到 90 年代的基于规则的机器翻译系统（rule-based machine translation，简称 rule-based MT）的主要概念和方法：直接翻译方法、中间语言方法、转换方法、基于知识的方法，并介绍了主要的机器翻译工具，简要回顾了机器翻译的历史。

第二十八章“机器翻译：新近的发展”（machine translation: latest developments）介绍了当前机器翻译系统的研究、开发和应用的情况，讨论了经验主义的机器翻译系统：基于实例的机器翻译（example-based MT）和统计机器翻译（statistical MT），并把它们与传统的基于规则的机器翻译系统进行了对比，同时还介绍了把各种方法融为一炉的混合机器翻译系统（hybrid MT）。在当前基于规则的机器翻译的开发中，回指消解的研究以及基于中间语言和基于知识的机器翻译的研究取得较大的进展，本章也做了介绍，此外，还介绍了口语的机器翻译，讨论了少数民族语言和不发达语言的机器翻译前景，讨论了因特网上的机器翻译（特别是网页翻译）的问题。最后，介绍了译者的电子翻译工具，特别讨论了双语语料库、翻译记忆、双语上下文索引等问题，并介绍了一些面向译者的词处理工具。

第二十九章“信息检索”（information retrieval）主要介绍文本的信息检索。信息检索系统的任务在于，对于用户提出的提问或者命题，给出与之有关文献的集合，作为检索的结果。首先分析了信息检索系统的软件组成成分，包括文献处理、提问处理、检索匹配技术，然后讨论自然语言处理技术对于信息检索的推动和促进作用，讲述了如何使用自然语言处理所得到的形态信息、短语信息、句法信息来改进信息检索中的索引技术，并且指出，当前的趋向是使用语义信息来进行信息检索。最后展望信息检索的发展前景。

第三十章“信息抽取”（information extraction，简称 IE）讨论如何从自由文本中自动地识别特定的实体（entities）、关系（relation）和事件（events）的方法和技术。本章主要讨论两种类型的信息抽取：一种是名称的自动抽取（extraction of names）。一种是事件的自动抽取（extraction of events），并介绍书写抽取规则的方法。对于名称的自动抽取，介绍了名称标注器（name tagger），对于事件抽取，介绍了事件识别器（event recognizer）。同时，还介绍了如何从已经标注了有关名称或事件信息的文本语料库中自动地学习抽取规则的方法，这种方法也就是信息抽取的统计模型。最后介绍了信息抽取的评测和应用。

第三十一章“问答系统”（question answering，简称 QA）讨论如何从大规模真实的联机文本中对于指定的提问找出正确回答的方法和技术，这是文本信息处理的一个新的发展趋向。由于 QA 要对于指定的提问给出一套数量不多的准确回答，在技术上，它更接近于信息检索（information retrieval），而与传统的文献检索（document retrieval）有较大的区别，QA 要生成一个相关文献的表作为对于用户提问的回答。与信息抽取相比，QA 要回答的提问可以是任何的提问，而信息抽取只需要抽取事先定义的事件和实体。在开放领域的 QA 系统中，使用有限状态技术和领域知识，把基于知识的提问处理、新的文本标引形式以及依赖于经验方法的回答抽取技术结合起来，这样，就把信息抽取技术大大地向前推进了一步。本章首先介绍了 QA 系统的类别和 QA 系统的体系结构，接着介绍了开放领域 QA 系统中的提问处理、开放领域 QA 系统中提问类型以及关健词抽取技术，并讨论了开放领域 QA 系统中的文献处理方法和提问抽取方法，最后展示了 QA 系统的发展前景。

第三十二章“自动文摘”（text summarization）介绍对单篇或多篇文本进行自动文摘的方法。首先讨论自动文摘的性质和自动文摘的过程。接着介绍自动文摘的三个阶段：第一阶段是主题辨认（topic identification），第二阶段是主题融合（topic fusion），第三阶段是文摘生成（summary generation）；并介绍了多文本的自动文摘。最后介绍自动文摘的评测方法，

讨论了自动文摘评测的两个指标：压缩比（compression ratio，简称 CR）和内容保留率（retention ratio，简称 RR）。

第三十三章“术语抽取和自动索引”（term extraction and automatic indexing）介绍术语自动处理的技术。术语广泛地出现在科技文献中，术语的自动识别对于科技文献的分析、理解、生成、翻译具有关键性的作用。随着网络的普及和数字技术的发展，出现在互联网、政府、工业部门和数字图书馆中的专业文献日益增多，术语的自动处理对于这些文献的信息检索、跨语言问答、多媒体文本自动索引、计算机辅助翻译、自动文摘等都具有重要作用。本章把面向术语的语言自动处理分为术语发现（term discovery）和术语识别（term recognition）两个部门，分别介绍了主要的技术和系统，最后介绍了双语言术语的自动抽取技术。

第三十四章“文本数据挖掘”（text data mining，简称 TDM）介绍本文数据挖掘技术。文本数据挖掘的目的在于从大规模真实文本数据中发现或推出新的信息，找出文本数据集合的模型，发现文本数据中所隐含的趋势，从文本数据的噪声中分离出有用的信号。本章首先讨论文本数据挖掘与信息检索的区别，分析了文本数据挖掘与计算语言学和范畴元数据（category metadata）的关系。本章举出实例，具体地说明了怎样使用生物医学文献中的文本数据来推测偏头痛（migraine headaches）的病因，怎样使用专利文献中的文本数据来揭示专利文本与已经发表的研究文献之间的关系，并介绍了 LINDI（Linking Information for Novel Discovery and Insight）系统，这个系统的软件能够根据大规模的文本集合来发现文本中蕴含的重要的新信息。

第三十五章“自然语言接口”（natural language interaction 简称 NLI）介绍计算机自然语言接口系统。这样的 NLI 系统可以把用户使用口头的自然语言或书面的自然语言提出的问题转化为计算机可以处理的形式。首先介绍了 NLI 系统的基本组成部分、意义表达语言（meaning representation language，简称 MRL）、同义互训软件（paraphraser）、问题生成软件（response generator）以及可移植工具（portability tools）。然后介绍口语对话系统（spoken dialogue systems，简称 SDS），分别介绍了 SDS 的单词识别软件、任务模型、用户模型、话语模型、对话管理软件、消息生成软件、语音合成软件。最后讨论 SDS 系统的灵活性、现状以及将来的应用前景。

第三十六章“多模态和多媒体系统中的自然语言”（natural language in multimodal and multimedia systems）讨论自然语言在多模态系统和多媒体系统应用中的重要作用，说明了怎样把自然的口语或书面语与多媒体输入协同地融合为一体，怎样把自然语言与其他的媒体结合起来以生成更加有效的输出，怎样使用自然语言处理技术来改善多媒体文献的存取。首先介绍包含自然语言的多模态和多媒体输入的分析问题，讨论了怎样把自然语言处理技术作为多模态分析的基础，怎样把不同的模态结合起来的技术。接着介绍包含自然语言的多媒体输出的生成问题，讨论了怎样把自然语言处理技术作为多媒体生成的基础，并讨论了不同模态的调和问题（包括不同模态的配置、不同模态输出的裁剪、模态输出中空间和时间的配合）。还讨论了用于多媒体数据存取的自然语言处理技术（包括基于自然语言处理的图形和图像检索、图形和图像数据库的自然语言接口、多媒体信息的自然语言摘要）。最后讨论在多媒体环境中使用语言的问题。

第三十七章“计算机辅助语言教学中的自然语言处理”（natural language processing in computer-assisted language learning）介绍在计算机辅助语言教学（computer-assisted language learning，简称 CALL）中使用自然语言处理技术的问题。首先介绍 CALL 的发展历史，接着介绍在自然语言处理背景下的 CALL，语料库与 CALL，双语语料库，讨论自然语言处理技术在形态学教学、语法教学、偏误的识别和诊断中的应用。最后讨论自然语言处理技术在 CALL 中应用的评估问题。

第三十八章“多语言的在线自然语言处理”（multilingual on-line natural language

processing）讨论在因特网上的多语言处理问题。因特网现在已经发展成多语言的网络，英语独霸互联网天下的局面已经成为历史，非英语的网站越来越多，语言的障碍日益严重，为了克服语言障碍，机器翻译当然是一个最重要的手段，除了机器翻译之外的各种使用自然语言处理技术的多语言处理工具也雨后春笋般地开发出来。本章介绍了语言辨别（language identification）、跨语言信息检索（cross-language information retrieval，简称 CLIR）、双语言术语对齐（bilingual terminology alignment）和语言理解助手（comprehension aids）4 个方面的研究情况。语言辨别的目的在于让计算机自动地判断书面文本是用什么语言写的，这显然是多语言自动处理必须经过的第一步。跨语言信息检索 CLIR 的目的在于使用一种语言提问来检索其他语言文本的信息。本章介绍了在 CLIR 中的译文发现技术（finding translation）、翻译变体的修剪技术（pruning translation alternatives）、翻译变体的加权技术（weighting translation alternatives）。在这些应用中，双语言词典或多语言词典是最重要的资源，而这些词典的覆盖面可以使用双语言术语对齐的技术来提升。语言理解助手的目的在于给用户提供软件工具来理解外语写的文本，而不必使用全自动机器翻译的技术。本章介绍了施乐公司欧洲研究中心（Xerox Research Centre Europe，简称 XRCE）的语言理解助手 LocoLex 和语义模型，并介绍了施乐公司使用语言助手来改善数字图书馆 Callimaque 的技术。

本章最后附有各章作者简介、计算语言学术语表，作者索引、主题索引。

三、 本书的简要评价

本书是手册性的专著，有如下三个明显特点：

- 专家执笔：本书的 38 章是分别由各个领域内的 46 位知名专家执笔的，由于这些专家对于自己的领域都是精研通达的内行，有力地保证了本书的学术质量和专业水平。
- 涵盖全面：本书几乎涵盖了计算语言学的所有领域，反映了当前计算语言学的最新成就，使我们对于计算语言学获得全面而系统的认识。
- 深入浅出；本书各章写作风格一致，内容协调，浑然一体，特别适合对计算语言学感兴趣和刚入门的读者阅读。本书使用流畅的文笔和有趣的实例来介绍艰深的技术问题，尤其适合于文科背景的读者阅读。

我国曾经翻译出版过有关计算语言学和自然语言处理的大部头专著，如《自然语言处理综论》（电子工业出版社出版，2005 年）被称为自然语言处理教材的“黄金标准”。但是，这部专著主要是针对理工科背景的读者写的，数学公式较多，文科背景的读者阅读和理解起来常常会感到困难。与《自然语言处理综论》比较，本书尽量不使用繁难的数学公式，文笔浅显而流畅，内容新颖而有趣，更加适合于文科背景的读者阅读。目前计算语言学这个新兴的学科不仅吸引了大量理工科背景的研究人员，同时，也有不少文科背景的研究人员投身到计算语言学的研究行列中来，本书的出版正好满足文科背景的研究人员需要。当然，由于本书内容涵盖面广，专业性强，对于理工科背景的研究人员也有很大的参考价值。

四、 有关计算语言学的其他参考文献

[1] 冯志伟，自然语言的计算机处理[M]，上海，上海外语教育出版社，1996 年。
[2] 冯志伟，应用语言学综论[M]，广州，广东教育出版社，200？年。
[3] 冯志伟，机器翻译研究[M]，北京，中国对外翻译出版公司，2004 年 12 月。

[4] 冯志伟，计算语言学概论[M]，商务印书馆，200？年。
[5] 冯志伟，自然语言处理的形式模型[M]，纪念中国科学技术大 50 周年校庆校友文库，中国科学技术大学出版社，，2009 年。
[6] Bill Manaris, Natural language processing: A human-computer interaction perspective [A], Advances in Computers, Volume 47，1999.
[7] Carstensen Kai-Uwe et al, *Computerlinguistik und Sprachtechnologie, Eine Einführung* [M], Heidelberg/Berlin, Spektrum Akademischer Verlag, 2004.
[8] Daniel Jurafsky, James H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition* [M], Upper Saddle River, New Jersey, Prentice Hall, 2000.。中文译本，冯志伟、孙乐 译，《自然语言处理综论》，电子工业出版社，2005 年。