

载《咸宁学院学报》，2005.4，第 60-66 页。

## 从比较中看计算语言学

刘海涛<sup>1</sup>

(中国传媒大学应用语言学系, 100024)

**摘要:** 本文通过对两本计算语言学著作进行比较, 讨论了计算语言学的定义、方法、理论、资源和应用等基本问题。作者希望本文不但有助于理清这些基本问题, 也有益于大学计算语言学专业课程的设置和各类计算语言学人才培养计划的制定。

**关键词:** 计算语言学 语言技术 自然语言处理 理论体系 方法

虽然早在 1962 年, 机器翻译的研究者们在美国普林斯顿成立的 AMTCL (机器翻译和计算语言学学会) 组织名称中就用了“计算语言学”一词, 但一般认为术语“计算语言学”正式登台是在著名的 ALPAC 报告之中 (1966: 29-31, 冯 1996: VI)。自然语言计算机处理的第一个领域是始于 1947 年的机器翻译研究。事实上, 正是早期机器翻译的研究者们奠定了计算语言学作为一个研究领域得以存在的基础。著名机器翻译历史的研究者 John Hutchins 认为: 机器翻译的先锋们所做的工作不仅仅有益于机器翻译这一领域本身的研究和发展, 而且对于计算语言学、自然语言处理以及从人工智能的角度研究语言问题, 都有极大的意义 (Hutchins 2000: 1)。如果仔细研读收录在 Hutchins (2000) 一书中有关机器翻译先驱的 30 篇文章, 我们不难发现, 许多在 50 年前困惑研究者的问题, 直到今天还是没有答案的难题。虽然这些年来, 世界各国的学者创立了不少面向语言形式化的语言学理论, 计算机技术本身也有了日新月异的发展, 但遗憾的是, 自然语言计算机处理的实质进展并没有人们所预料的那样大。为什么会出现这种尴尬的局面呢? 几十年来, 这一领域的科学家们到底在做什么呢? 构成计算语言学基础的有哪些知识? 为了进入计算语言学的殿堂, 人们需要学哪些东西? 本文通过比较两本较著名的计算语言学著作, 从计算语言学的定义、方法、理论和发展的角度, 对这些问题进行了讨论。希望本文有助于大学计算语言学专业课程的设置和各类计算语言学人才培养计划的制定。

为了均衡反映世界计算语言学的研究成果, 我们选择了 Carstensen, et al. (2001): *Computerlinguistik und Sprachtechnologie*. 和 Jurafsky/Martin (2000): *Speech and Language Processing*. 作为比较研究的主要资料。为了方便讨论, 以下我们称 Carstensen, et al. (2001) 为 CuS, Jurafsky/Martin (2000) 为 SLP。我们知道, 有关计算机语言学的著作数以百计, 就是综合性的教材也有许多, 为什么单单选择这两本书呢? 就目前世界的计算语言学研究而言, 我们可以大致将其分为两个大的学派, 一是以德国学者为代表的欧洲大陆学派, 在他们的著作中我们常常可以看到更多的人文因素, 具体而言就是更多地注意语言本身的问题; 而以美国学者为代表的学派,

---

<sup>1</sup> (作者简介) 刘海涛, 男, 中国传媒大学应用语言学系教授。主要研究方向: 计算语言学, 国际语语言学, 语言规划及生态语言学, 依存语法和配价理论。

更多的贡献在于技术方面，特别是语言的形式化理论。可以说，如果没有不久前刚刚出现的CuS，也就没有现在的比较，按照Manfred Pinkal 的说法，此书是填补德语区空白的计算语言学著作。而SLP，在问世不久已是好评如潮。许多著名的学者认为SLP是有史以来内容最丰富、结构最得当的计算语言学著作，已被多所大学用作教材<sup>2</sup>。如 Teller (2000) 在“Computational Linguistics”上对此书的评论开篇的文字为“盼望已久的 Jurafsky 和 Martin 之大作建立了一个难以超越的黄金标准。他们的书将基于知识和统计的方法，将语言学基础和计算模型成功地结合在一起，各部分内容不但做到了均衡、完整，而且非常便于阅读。”

从篇幅方面看，SLP 达 934 页之巨，CuS 也有 600 页之多。CuS 是由来自德国、瑞士、奥地利的 41 位学者合作写就，从书末所附的作者名单来看，涉及德语区各主要从事计算语言学的大学和机构。SLP 主要由美国科罗拉多大学的 Daniel Jurafsky 和 James H. Martin 操刀，只有少数章节是由他人代劳的。为了方便与读者沟通，二者均设立了专门的互联网站：<http://www.cs.colorado.edu/~martin/slp.html> (SLP)，<http://www.ifi.unizh.ch/groups/CL/CLBuch/> (CuS)。

计算语言学不仅仅是一个跨越多个领域的学科，也是一个迅速发展的学科。这就意味着，像 SLP 和 CuS 的著作难免会有这样那样的印刷错误和不足。因此，一本好的教科书应该具有与时俱进的精神，不断修正自己的错误，从而使自己更加完善。这一方面，CuS 虽然问世较晚，但已于 2004 年出版了修订和扩充的第二版。而 SLP 也不甘落后，从 2004 年 11 月在自己的网站开设的“更新<sup>3</sup>”栏目，开始在网上发布某些章节的修订版。此前，SLP 也在网上开设了“勘误表<sup>4</sup>”栏目。

下面我们将从基本概念(定义)、理论基础、方法、资源、应用等几方面，比较、研习这两本著作。此比较框架基本上是按照 CuS 的结构而来的。

## 1、计算语言学的基本定义和发展简史

“**计算语言学**是研究用机器来处理自然语言的学科。它是由信息技术和语言学交叉而成的”(CuS: 1)。有趣的是我们没有在 SLP 中发现计算语言学的确切定义。SLP 的作者在开篇借用了 Stanley Kubrick 科幻片中的人物 HAL，HAL 是一个通晓英语的机器人。作者引入 HAL 的目的是想说明，为了构建这样一个可与人通过自然语言进行交流的机器人，需要哪些知识和技术：语言理解方面有**语音识别和自然语言理解**（包括唇读技术），表达方面需要**自然语言生成和语音合成**，另外 HAL 也需要有**信息检索、信息提取和推理**方面的能力。而解决这些问题一般会涉及以下学科：自然语言处理，计算语言学，语音识别和合成。SLP 的作者将这三者合起来称为**语音及语言处理**，除了以上 HAL 所用的这些技能外，SLP 也囊括了其他重要的语言处理领域，如：**拼写校正、语法检查和机器翻译** (SLP: 1-2)。由此可以看出，在 SLP 作者的眼里计算语言学有别于自然语言处理，但二者的差别何在，作者没有进一步的说明。根据以上说法，我们不难推出 SLP 里的计算语言学的研究对象也是

---

<sup>2</sup> 本书的中译本《自然语言处理综论》已由电子工业出版社于 2005 年出版，译者为冯志伟、孙乐。冯志伟教授在中国传媒大学开设的“语言信息处理专题”研究生课程，已采用此书作为教材。

<sup>3</sup> <http://www.cs.colorado.edu/~martin/slp/updated.html>

<sup>4</sup> <http://www.cs.colorado.edu/~martin/slp/slp-errata.html>

自然语言的计算机处理。

为了让读者对于计算语言学有更全面、完整的了解，CuS 特设一章“计算语言学面面观”来讨论计算语言学的学科定位、与相邻学科的关系、学科细分等问题，这对于读者把握计算语言学的学科意义是非常有用的。而 SLP 没有类似的内容。这可能与我们开始所说的德、美两国的研究传统有关。

计算语言学作为学科有以下四种研究路向 (CuS: 10-11):

- 作为语言学的分支 (如同社会语言学, 心理语言学等), 它只研究语言及语言处理与计算相关的方面, 而不管其在计算机上的具体实现。语法形式化模型的研究便是其重要的研究领域之一。这类似于英语术语 **computational linguistics** (计算语言学) 的涵义。
- 作为开发语言学相关程序以及语言学数据处理的学科。这一方向近年来随着大规模语料库的出现和应用, 愈加重要。
- 作为在计算机上实现自然语言能力的学科 (德语区一般称为“机器的语言处理”, 对应于英语的“自然语言处理”)。这一方向的研究与语言哲学、人工智能以及认知科学关系尤为密切, 它的主要研究目标是建立语言处理的形式化模型, 进而通过计算机来验证这种模型。
- 作为面向实践的、工程化的语言软件开发。(即: 语言技术)

以上划分较好地解决了我们在阅读计算语言学相关文献时, 关于学科定义的困惑。至少在现在, 我们知道以上这四种研究方向都属于计算语言学, 只不过侧重点不同而已。我们也可以理解为什么前苏联计算语言学的先驱之一, **Meaning-Text Model (MTM, Mel'čuk: 1988)** 语言理论的创立者, 现任加拿大蒙特利尔大学教授的 **I. A. Mel'čuk** 对于机器翻译 (计算语言学) 的极端说法, 机器翻译应该是“没有翻译、没有机器、没有算法的翻译”, “没有任何事物比一个叫得响的理论更实用” (**Hutchins 2000:249**)。关于计算语言学的研究路向问题, 袁毓林也曾撰文研究, 可参考 (袁, 2001)。

计算语言学是一门交叉性的学科。内容涉及语言学、信息技术、哲学 (特别是语言哲学和逻辑学)、人工智能、认知科学、数学 (数理逻辑、自动机和形式语言、图论、统计学等) (CUS: 11-14)。关于计算语言学学科的交叉性, 我国学者也多有论述, 冯志伟先生在自己的多部著作中都认为语言的计算机处理涉及语言学、计算机技术和数学等领域, 从而强调培养跨学科人才的重要性 (冯, 1996: III)。值得注意的是, 我国的研究者对于计算语言学的人文性认识不足。许多人认为计算语言学就是计算, 从而忽视了计算语言学作为一个语言学分支学科所具有的人文本质。例如, 我们很难看到我国学者从语言哲学方面对于计算语言学的探索 (刘, 1993)。从心理学和认知科学的角度, 探索计算语言学的东西也不是很多 (袁, 1998)。我国的计算语言学研究也有近五十年的历史了, 但是真正能够拿到国际上的东西不多, 这与我们长期忽视有关基础理论的研究有一定的关系。陈力为院士在其主编的全国第三届计算语言学联合学术会议论文集《计算语言学进展与应用》的前言中说: “国内基础理论的研究仍显薄弱, 同国外同行相比我们还有较大的差距, 这种状况应当引起我们的注意”。我们一直认为, 没有理论的实践是盲目的, 对于计算语言学这样的前沿领域更是如此。结合上述四种计算语言学的研究路向来看, 其中的第一、三项是基础, 没有这两项研究的支撑, 第四项的研究不会有什么真正意义上的突破。

任何急于求成，只看到短期效益的语言技术应用，只会是昙花一现。当然，我们也应该看到在语言学的所有分支学科中，计算语言学是最具实践性的。不论是 CuS 还是 SLP，都对于计算语言学的实践部分（语言技术）给予了足够的重视。如何处理好计算语言学理论与实践的关系，看来是任何意欲涉足这一学科的结构与研究者首先应该考虑的问题。

如果说 CuS 在开篇重点介绍了有关计算语言学的方方面面的话，SLP 的做法却是直指目标：语言知识和歧义问题。假如只用一句话来概括计算语言学几十年来的历史，那就是一部与“歧义”作斗争的历史。斗争的结果，是我们充分认识到，语言的处理（理解）过程，必须有知识的支撑。为此，SLP 的作者也在那位人造的 HAL 先生头脑里，加上了语音、构词、句法、词汇语义、合成语义、语用、语篇等知识。为了说明语言、思维和理解的关系，SLP 不但介绍了著名的图灵测试，也给我们引见了大名鼎鼎、会与人交流、但没有脑子的伊丽莎（Eliza）。CuS 虽然没有过早地将令人头疼的“歧义”摆在读者面前，但是对于语言知识问题倒是没有隐瞒。另外，CuS 在介绍计算语言学应用的例子时，拿出来的是由多国学者合作、但主力为德国学者研究开发的 Verbmobil 口语自动翻译系统，虽然此系统只能把别人说过的话，用另一种语言说出来，但其智商则远远高于伊丽莎。

二者都在开篇简要介绍了计算语言学的历史以及各个阶段的代表性理论与系统。这说明了解过去的研究成果，对于任何学科的发展都是必不可少的。根据书中的介绍，我们列出以下简表：

年代	理论方面	实用系统方面
1950	有关计算语言学的最早设想 自动机 概率及信息理论模型 形式语言理论 人工智能	语音识别系统  机器翻译的实验性系统
1960	语言处理的字符串运算理论	基于转换语法的语言心理模型 布朗美国英语语料库
1970	利用上下文无关文法的句法分析 词法分析 Colmerauer 的 Q-系统 ATN 文法	Winograd 的 SHRDLU 自动音节划分
1980	线图分析 语义结构	自然语言的数据库接口 拼写校正 机器翻译进入日常运用
1990	语篇表示理论 合一语法，双级词法 基于约束的语法 基于统计的标注	基于统计的机器翻译

2000

## 2、理论基础

值得注意的是 CuS 和 SLP 的结构完全不同。CuS 分为以下几部分：什么是计算语言学？理论基础、方法、资源、应用以及系统测评。而 SLP 采用了人们习惯的分层次语言描述和处理方式，即：导论、词、句法、语义、语用等。笔者认为 CuS 的结构比较符合我们文章的目的，故掠来用之。

我们曾经按照软件工程的思路把计算语言学系统的构建过程分为三个模块：理论语言学、计算语言学和计算机科学（刘海涛 1995）。其中计算语言学家的主要任务就是语言的形式化。显然，形式化的基础离不开数学和其他面向计算的理论。为此这一部分更确切的标题应该是自然语言的形式化理论基础。严格说来，一本关于计算语言学的著作完全可以不含这方面的内容，但是，任何计算语言学的方法均是建立在这些理论基础之上的。实践证明，如果没有足够的理论准备，人们很难真正进入计算语言学的殿堂。为此，我们认为 CuS 专门设立这一章节来简要介绍现代计算语言学中形式化的基本理论是必要的。对于打算从事计算语言学研究的个人而言，可以从中了解需要掌握什么样的数学工具。对于培养计算语言学的机构而言，可以据此设立相关的课程。事实上，CuS 在这一部分介绍的许多东西，在 SLP 中都得到了一定的应用。换言之，如果没有这些预备知识，是无法读懂 SLP 中的相关内容的，更不要说其它有关计算语言学的时文了。

在题为“集合与逻辑”的第一节里，CuS 为我们介绍了有关集合论、命题逻辑、谓词逻辑、类型（高阶）逻辑、Lambda 演算、模态逻辑等。就目前计算语言学中的趋势而言，采用逻辑的领域已经不再局限于过去语义和语用方面了，也已出现不少基于逻辑的句法描述体系。

“自动机和形式语言”是 CuS 有关形式化理论部分第二节的内容。除了我们大家熟悉的形式语言、有限状态理论、上下文自由及相关文法等内容外，作者也讲到了新近流行的树邻接文法（TAG）以及算法的可计算性和确定性。

在“图论和特征结构”这一节里，有关图和树的介绍，使我们对这两方面极易忽略的基础知识，有了更深的了解。特征结构和合一是一对孪生兄弟，有关他们的内容当然是少不了的。类型化特征结构（Typed feature structure, Typisierte Merkmalsstrukturen）是一种能够有效解决一般特征结构不足的方法，所以 CuS（p.101-105）和 SLP（p.437-441）对此都进行了专门的介绍。

基于语料库的计算语言学方法毫无疑问正引起越来越多学者的注意，以至于产生了诸如数据密集方法、基于经验的方法、基于实例的方法、基于类比的方法、面向数据的方法等形形色色的说法和做法。从表面来看，这些技术和方法略有区别。但他们的理论基础基本上都是基于概率统计的。CuS 用专门的一节来介绍基本的概率理论和用的最多的隐马尔柯夫模型（HMM）。

CuS(第二版)在这一部分增加了题为“文本技术基础”的章节，简要介绍了有关

置标语言和基于 XML 的文本标注等技术。考虑到文本技术是语言学家参与到语言信息处理领域的一种重要途径，CuS 适时添加有关内容的做法是值得肯定的。当然，限于篇幅我们无法完全领略到文本技术对于计算语言学的巨大作用，但两本新近出版的德文著作(Lobin/Lemnitzer 2004, Mehler/Lobin 2004)可以弥补这种缺憾。

### 3、方法

“方法”是 CuS 第三部分的标题。所谓计算语言学方法，实际上就是采用形式化的手段，对自然语言各个层次进行描述的方法。CuS 用了大概 230 页的篇幅 (p.135-360)，对语音、词法、句法、语义、语用及文本生成等方面做了介绍。对于每一部分内容，CuS 均先介绍相关的语言学概念和理论，然后再介绍怎么将这些东西形式化，即相应的计算语言学方法。由于篇幅所限，CuS 对这一部分内容的介绍，比较简洁，但还算全面。当然，如果你打算通过 CuS 的学习，就能精通这些计算语言学方法，进而在计算机上实现这些算法，那还是有些困难的。好在对于懂德语的人来说，有一本 Naumann/Langer (1994) 可供参考。除了偶尔提醒我们有这么一回事外，CuS 几乎没有给概率与统计方法留下什么空间，虽然它在第一部分的最后一节引入了相关的概念。CuS 第二版在这一部分增加了“浅层句子处理”一节，具体内容为“分词”(Tokenisierung)、“词性标注”和“组块分析”(Chunk-Parsing)。在第一版里，CuS 只从目前语言学的各个研究的层面来讨论计算语言学的方法，但这一节的加入使得人们意识到语言的计算机处理还是有一些特殊之处的，如，词的切分和词性标注就是计算机处理语言怎么也绕不开的问题。“组块分析”作为一种句法剖析的过渡性手段，近些年来在计算语言学中得到了一定的应用，因此也是值得一提的。在“语用学”一章里也新增了一些内容：文本、语篇和会话，指代消解和焦点。这一部分新增内容说明了近两年来越来越多的计算语言学家们已经认识到计算语言学的根本任务是研究一个能与用人用自然语言交流的计算机系统(Hausser 2001<sup>5</sup>)。

不过读者大可不必为 CuS 这一方面的缺陷担心，因为 SLP 里几乎所有的篇幅都是用来讲述计算语言学方法的。下面我们就将注意力转向 SLP。

SLP 的在题为“词”(Word)的第一部分 (p.19-284)，分六章介绍了有关词的拼写、发音与构成的计算模型，通过这些模型可以构建以下三种与词汇知识密切相关的应用：自动语音识别 (ASR)，文语转换 (TTS) 和拼写错误校正。虽然，SLP 没有单辟一个专栏讲述基础理论，但是相关的内容可是一点也不少。如有关自动机的理论，SLP 仅在此就一口气拉进来四个：有限态自动机 (FSA) 与规则表达式，有限态转录机 (FST)，加权转录机和隐马尔柯夫模型 (HMM)。另外，也为我们引见了大名鼎鼎的 N 元语法 (N-grams)。是否你仍然记得 SLP 中的 S 是 Speech，如果你是为此而来的，那么我要提醒你 SLP 中有关口语的内容几乎都在这一部分了。假如你想构建某个口语信息处理系统，或者想更深入地了解有关有声语言的计算机处理，只有这些东西显然是不够的。请参考另一本近 1000 页的专著 (Huang/Acero/Hon 2001)。如果你没有构造有声语言系统的打算，那么 SLP 在这一

---

<sup>5</sup> 笔者有关此书的评论见《咸宁学院学报》2003 年第一期。

部分提供的有关词语处理的计算模型知识对于你而言足够了。在介绍传统的自动机的同时，SLP 也没有忘了引入有关概率的计算模型。

句法是 SLP 第二部分的主题 (p.285-498)。句法的主要任务是研究词与词之间的关系。SLP 在“句法”这一部分的六章里，介绍了如何将词分为不同的类 (POS)，词是如何与其邻居组成短语的，以及一个句子中词与词之间的依存关系。但是在了一本有关计算语言学的著作中，对于句法仅作如此理解是不够的。事实上，作者在这一部分也为读者准备了几乎所有的用来处理句法的计算模型，如：上下文自由文法、词汇语法、特征结构和合一。除此之外，作者也比较详细地介绍了用于句法剖析的 Earley 和 CYK 算法以及用于特征结构的合一算法。其详细程度，基本上达到了依据书里的描述可以直接用程序设计语言实现有关的算法。在基于概率的句子分析方面，作者为我们提供了 HMM 句法标记器和 PCFG (概率上下文自由文法)。鉴于心理语言学近年来和计算语言学的关系越来越密切，以至于有了计算心理语言学的说法。作者也用一定的篇幅讨论了人类句法处理的心理模型。由于依存语法及其形式化是我自己的爱物，我特别注意了一下两本著作有关这一方面的讨论。SLP 单列标题为“依存语法”的一节内容，虽然篇幅不大，倒是基本涵盖了用英语发表的基于依存语法分析的几篇主要文献。如果你了解依存语法诞生于法国、成长于德国的历史，难免会对 CuS 寄予厚望，但是我们又错了，虽然 CuS 也提到了依存语法，但用料比 SLP 更少，意外的收获是其所举例子的依存结构树竟然来自 1883 年的一本书里，而其对手 (短语结构) 的树则晚了好几十年！

SLP 从 499 页到 666 页是语义的领地。语篇 (文本) 的意义之重要性怎么说都不过分，因为我们使用语言形式的目的就是为表达意义、传递意义。比如说，Mel'cuk 将自己的语言理论就称为意义—语篇理论 (模型，MTT) (Mel'cuk 1988)，语言理解和生成的过程，说穿了，就是意义和语篇之间的转换与映射。事实上，正如开篇所说的那样，计算语言学几十年进展不大的主要障碍是歧义，即自然语言结构各个层面上的一种形式对应多个映射的问题。实事求是地讲，虽然我们现在离歧义问题的最终解决仍然很远，但是几十年来众多学者的探索还是积累了不少有意义的成果。SLP 在这一部分基本上介绍了目前处理语义的主要手段。首先遇到的问题是意义的表示，虽然对于人而言，意义的表示原本不是什么问题，因为语言就是用来表示意义的。但是，无处不在的歧义迫使人们为自己 (哲学家) 和为机器 (计算语言学家) 寻求更简单明了、不会产生歧义的表达手段，就现在来看，逻辑是我们所能发现的最好的东西了。正如许多具有哲学素养的计算语言学家所言，逻辑不但能够无歧义地表达意义，更神奇的是通过各种逻辑运算，我们能够操作意义。逻辑有如此功效，难怪各种逻辑方法层出不穷，甚至于也可见到完全基于逻辑表达的自然语言理解全过程的逻辑系统 (Kempson 2001)。但是逻辑对于意义的解决是有限的，其根源我们曾经在 10 年前简单讨论过 (刘海涛 1993)，即使这世界变化实在是快，十年后的今天，我们对此看法依然没有改变。如何利用现阶段的技术来构造能够懂得一些意思的系统？SLP 的作者在“语义分析”一节进行了介绍，如：在 Early 剖析程序中集成语义分析技术，或者怎么想办法弄一个经得住折腾的语义分析系统出来等。为了帮助读者搞清楚词汇本身的意义问题，作者专门设立“词汇语义学”一章，对于词汇语义学的一些基本问题进行了介绍。在这里，大家不但能够读到有关题元、选择性限制、义素分解、语义场等常见的东西，也看到了人们希望通过词

间关系来发现词义的努力 WordNet。词义消歧 (WSD) 是一个值得大书特书的领域，但是不知为什么它只能和信息检索搅在一章里，搞的没名没份。在减少了传统的基于选择限制的消极策略之后，作者引入了两种更经得住折腾的 WSD 技术：基于词典的方法和基于机器学习的方法，后一种显然从名字就可以看出是基于概率的方法。把信息检索和 WSD 放在一起，虽然难以理解作者的意图，但是信息检索作为一个越来越重要的计算语言学应用领域，值得引起注意。就它们对于计算语言学的重要性而言，WSD 和信息检索都值得在 SLP 中有一套自己的房子。

为了更好地处理语言信息，单有词法、句法、语义的知识是不够的。虽然人们目前对这三个领域仍然有大量未知的东西，但是计算语言学家的触角已经开始伸到了语用的领域。语用学有狭义和广义之区别，狭义的语用学实际上是研究句子如何组成语篇的理论。计算语用学研究如何将语篇结构通过形式化的手段来描述或如何建立语篇的计算模型。当前，语用理论的主要应用领域为：指代问题的处理和语篇的生成。语用较之句法和语义而言，人们对它的了解就更少了。为此，SLP 中有关“语用”的第四部分，在简单介绍了语篇的一般理论和指代算法之后，就将自己的目光转向三个实际的计算语言学应用领域：对话智能体、自然语言生成和机器翻译。通过这些应用的构造过程和原理，我们可以更好地把握语篇及语用在计算语言学中的作用。最后 SLP 就像是捎带着一般介绍了一下机器翻译，给人的感觉是简单了一些，可能是作者考虑到已经有许多机器翻译教科书的原因了吧。(Hutchins/Somers 1992, Trujillo 1999) 至此，有关 SLP 的故事结束。

#### 4、资源

提到资源，首先想到的一个说法就是“资源共享”。计算语言学发展到现在已经有不少研究的成品或半成品了。如何有效地利用这些资源，是不少计算语言学家及机构经常考虑的一个问题。互联网的出现，使得这种资源的共享，变得越来越简单，越来越容易。应该注意的是，我们这里所说的资源是计算语言学研究的原材料，即：各种形式的语言材料，而不仅仅是一些单纯具有工具性质的软件。冯志伟先生在一次学术讲座中提到了这个问题，他认为：机器翻译系统开发有两种思路，其一可称为“劳动密集型”，在这种开发方式下，语言学家的工作是为某个具体的机器翻译系统开发词典、规则库等语言知识库，语言学家的工作依附于某个具体的机器翻译系统，不具有独立性。其二为“资源密集型”，采用这种开发方式的语言学家的研究成果以语言资源的形式呈现出来，语言专家不是仅仅为某一个机器翻译系统服务，其工作应具有一定的独立性。机器翻译系统尽可能利用已有的公开的语言资源，资源密集型的机器翻译研究将成为今后的方向。资源密集型的开发方式是计算机专家和语言专家的一种更合理的分工形式，并将成为主流。学术研究（包括学生选题）应采用资源密集型方式。语言学家应该尽量做公共的资源，少做专用的资源。(冯 2003) 他的这种说法，虽然谈的是机器翻译，但其中所含的基本思想也适合计算语言学、特别是语言技术的其它领域。

为此，CuS 将“资源”单放一处，是一个很好的做法。下面我们来看一下，计算语言学的资源到底是些什么东西。

首先登场的是 WWW，这三个 W 说穿了就是通过网页的形式提供给人们信息

的一种方式。构成 WWW 网页的基础是置标语言，目前一般用的为 HTML，但是对于计算语言学而言，XML 和 SGML 更为有用。观察近年来出现的标注文本库，XML 几乎可以称为一种显式标注文本的 Lingua Franca。本节介绍了采用 XML、SGML 等作为标注语言的几个项目：TEI, MULTEXT, GDA 等。我们认为 XML 在计算语言学界的流行，不单单是来自互联网方面的推动，也不单单是它的结构易于标准化、便于共享。自然语言的理解和生成需要各种知识，为了让计算机处理语言，人必须把许多的有关语言结构和有关世界的知识，用显式的方法赋予计算机。XML 具备了担当如此重任的基本素质，所以值得推广和研究。可惜 SLP 没有提及这方面的事情。

语料库毫无疑问是目前最广为人知的（计算）语言学资源，CuS 简要介绍了语料库的分类、建立、标注等一般应该了解的知识。对于常用的一些利用语料库进行语言研究的软件也有涉及，但只是点到为止。树库是一种经过深加工的语料库，所谓的深加工一般是对于语料库中的语料进行句法方面的标注。CuS 引入了目前最有名的 Penn Treebank（宾夕法尼亚大学树库）和德语的老虎库（TIGER Baumbank）。虽然着墨不多，但也可以对树库有一个基本的认识。如果有兴趣深入研究，CuS 和 SLP 均提供了大量的文献索引。

普林斯顿大学的 WordNet 基本已成为研究词汇语义的楷模了。各种语言近年来纷纷建立自己语言的（语义）网络。CuS 介绍了德语的 GermaNet 和 Euro WordNet，后者通过一种媒介语索引（Interlingual Index）将英语、西班牙语、荷兰语、意大利语、法语、德语、捷克语、爱沙尼亚语等偶合在一起，这种多语言的词汇语义知识库对于各语言之间词义的比较、翻译的研究都是很有好处的。

树库为计算语言学使用语料库来作为句法分析的知识库提供了可能。事实上，观察近年来研究树库者的学术背景，发现大多为计算语言学家。而 WordNet 为计算机语义处理开辟了一条航道，诸如 WSD（词汇歧义消解）、信息检索、语料的语义标注等均可从 WordNet 所含的词汇语义知识受益。

CuS 第二版在这一部分增加了“非语言知识”一章。虽然篇幅很短，只有 7 页，但及时提醒人们要想构造任何有实用意义的智能系统，非语言知识是不可缺少的。知识和知识表示是人工智能研究的一个重要领域，有兴趣的读者可在那里找到更多的参考资料。

除此之外，CuS 也对词典、百科全书、有声语言数据库等语言学资源做了介绍。为了建立“资源密集”式的语言技术开发体系，我们需要知道我们有哪些资源，这些资源存放在何处。CuS 在这一方面开了一个好头，我们希望有更多的资源使用路线图问世，只有这样才能形成真正意义上的“资源共享”开发模式。

## 5、应用

与其他语言学分支有所不同的是，计算语言学中含有构建成分，可以将它称为“应用驱动”的语言学研究。换言之，我们前面提到的所有基础理论、方法与资源，最终都是为了构建出能够处理自然语言的人工智能体而服务的。计算语言学是一种应该能够解决问题的语言学分支，正如国际语语言学（interlinguistics）的目标是解决人际跨语交际一般，计算语言学要解决的则是人机自然语言的交流问题。计算语

言学的这种应用特质，使得一些人也把它称为“语言技术”，CuS 中的 S 就是德文“语言技术”的第一个字母。这种用法在英语中也并非罕见，如由 Cole 等人编辑的反映计算机语言处理现状及成果一书的标题中就有“语言技术”的字样（1997）。既然 CuS 在标题中都有“语言技术”，所以它能够单设一章列举计算语言学的种种用处，也就是顺理成章之事了。应该提及的是，虽然 SLP 没有专开场地讲述计算语言学的诸多用处，但是我们在它近千页的篇幅里，随处可见应用的影子。以下就是 CuS 列举的计算语言学的主要应用：

写作辅助工具。这一类软件中最常用就是拼写校正程序。虽然目前我们使用的拼写校正软件的智商不高，但是仍然能为我们解决一些实际问题。现在的拼写校正系统大多基于一个固定的词表，对于未登录词的处理很不理想。已经有人开始研究基于 N 元语法的拼写校正系统了，希望这种基于上下文的校正程序能够解决目前系统的一些不足。另外，语法、文体等方面的计算机辅助修改软件也是非常有用的，当前市面上的程序很是幼稚。较为理想的解决方案，可能还需要计算语言学其他方面的进展。无论如何，有关这一方面的需求是旺盛的。我们期待智商更高的系统的出现。

计算机辅助词典学。词汇以及词典几乎是任何计算语言学应用中必不可少的成分。如果考虑到目前语言学中的泛词汇主义倾向（Hudson 1990, Starosta 1988），有关词汇的研究和词典的构建是非常重要的。词汇研究一般包括以下几个步骤：词汇信息的获得（从语料库和传统的词典），词汇信息的表示（内容方面，技术方面和标准化方面）。这其中如何自动提取词汇，并把它们表示为便于人或计算机使用的形式，是研究的主要课题。计算机辅助术语学是另一个非常有前景的领域。有关词汇和术语的工作，不单局限在单语的范畴，也可以通过双（多）语的信息源来（半）自动构造相应的词典和术语集。

全文搜索和文本挖掘。互联网的流行，使得智能全文搜索和文本挖掘成为计算语言学中重要的研究领域，其实用价值非常大。如果计算机没有办法解决由于自己的长大而带来的信息泛滥问题，信息对于人类就是灾难，而非福音。CuS 用 17 页的篇幅介绍了有关全文搜索和文本挖掘的一般原理和常用方法。虽然信息检索是人类很早就开始用较大的精力研究的一个领域，但是在查全和查准之间的斗争似乎仍在延续。大量的工作还在等着人们去做。

文本分类、信息提取、文本摘要解决“信息泛滥”问题的另外三个研究领域。如果我们翻开近年来有关计算语言学的书刊报章，就会发现这三个领域着实很“火”。CuS 能把这三个领域都单列一章介绍，一方面说明了计算语言学确实是一个解决问题的学科，另一方面也体现了国外计算语言学界确实想培养一些实用性的人才。如果互联网的热潮不断继续，如果“信息唾手可得”不再只是比尔·盖茨一个人梦想，那么人们需要计算语言学，需要通过计算语言学家的努力来解决他们遇到的问题。目前，互联网上已有一些文本分类、信息提取、文本摘要的系统在日夜运行。如果我们从学术的角度分析，不难发现它们还有这样那样的问题，但是由于它们适时推出，在一定的程度上满足了市场的需要，所以取得了成功。这说明在计算语言学里，“完美”的实现是没有的，成功只会属于那些“抓住机遇”的系统，只要你能“与时俱进”地解决实际问题，哪怕只是部分问题，你就离目标又近了一步。

语音识别和语音合成是构建一个完整的人机自然语言交互系统不可缺少的组成

部分。虽然已有不少商业系统可供用户选用，但是其性能仍然存在一定的问题。对于合成而言，如何让机器说的话更加自然，一直是研究者们奋斗的目标。对于识别来说，非特定人、非受限领域以及连续语音的问题在短时间内恐怕仍然是一个不好解决的问题。作为一本教科书，CuS能做的也就是介绍一般的方法和未来的努力方向。关于合成，CuS也提到了更具挑战性的从概念到语音的合成问题。这事实上就是SLP中所说的文本（语篇）生成系统，只不过加了一张电子嘴而已。

自然语言人机接口是一个便于人类的使用者和计算机数据库打交道的系统。通过自然语言到数据库查询语言的转换，使用者可以用自然语言在数据库里寻找感兴趣的东西。目前已有一些基于受限语言的接口可供使用，但如果你想用吩咐一个秘书那样的语言来让数据库干点什么，还尚需时日。

著名的图灵测试要求我们构造出一个能说会道的机器来，而且这机器的智商应该能迷惑人类，进而会将这机器视为同类。于是乎，构建一个能听会说的机器人便成了许多研究者努力的目标。人类对此事如此热衷，就连近乎白痴的伊丽莎白都迷倒一片追随者。由于制造一个智能机器人几乎要用到所有我们上面提到过的知识，也许它永远都是人类一个美好梦想。在这个问题上，Arthur C. Clarke可能又对了，“The only difference between humans and machines is that humans can be created by unskilled labor.”用我自己的话说就是：世上所有最聪明的人试图用非传统的方式来制造人类的努力，无论如何也比不上世上最普通的人用传统的方式在不经意间所创造的具有相同功能的产品。这绝对不是玩笑，而是每一个打算从事计算语言学事业的人，应该考虑的问题。只有这样我们才能切实制订自己的工作目标和努力方向，也就是：理论求高，应用求实。

计算机辅助教学系统是语言技术可以展示其才能的又一领域。无独有偶，CuS的最后一章也是“机器翻译”，作为语言信息处理中最早出现的领域，在两本有关计算语言学的书里，都最后一个出场。这也许是偶然，但是我们难道打算用“偶然”来应付所有我们无法解释的东西吗？也许，Bar-Hillel的“FAHQMT永远只是人类的一个梦想”的预言在可预见的未来都是真的。一个真正意义上的机器翻译系统几乎集成了所有自然语言处理的技术和理论，这可能正是它为什么最早上场、最晚谢幕的原因。也许是FAHQMT太难了，所以CuS(第二版)在“应用”这一部分新增的唯一章节就是“计算机辅助翻译”。

计算语言学与其他语言学分支还有一个不同的地方是，它的成果可以通过评测来进行客观的评价。通过评价人们可以一目了然地了解各种系统的优劣。近年来，各种测试样本和方法层出不穷。评测虽然一般分为面向用户和面向开发者，但与任何有关应用的行当一样，小平同志的“猫鼠”理论永远都是有道理的。CuS最后一部分的内容专门用来讨论语言信息处理系统的评价，绝非偶然。

## 6、小结

严格说来，这篇东西写到这里，似乎已经脱离了开篇提出的目标。它既不是书评，也不是论文，只能看做是一种读书随笔。希望仍然对于各位同行有点用处。也算是响应冯志伟先生号召，为“资源共享”做点力所能及之事。从这篇不成样子的东西里，我们可以看出计算语言学是一门非常有用的学科，是一门能够解决问题的

学科，是一门发展迅速的学科，是一门新方法、新理论层出不穷的学科，是一门搞好了就能来钱的学科。遗憾的是，就其终极目标而言，它也是一门进展不大的学科。这是正常的，如果我们环顾周围那些探索人类自身奥妙的学科，就会发现它们也似乎处在一种永恒的探索状态。计算语言学所具有的双重性，容易导致我们在制订目标时，总是对困难估计不足。就面向应用的那一部分计算语言学研究而言，如何切实根据现有的理论和技术，制订一个能够达到的目标可能是至关重要的。计算语言学不是简单地将计算机和语言学加在一起，它所涉及的学科之广，超乎我们的想像。它对于人类的意义，绝对不只是构建一个能说会道的机器人。

应该承认德国人严谨的学术传统在 CuS 还是得到很好的体现。如果我的学生是文科背景，我会推荐他们把 CuS 作为主要参考著作，把 SLP 作为辅助读物；如果他们理工科背景，就是以 SLP 为主，而 CuS 为辅了。这里介绍的 SLP 和 CuS 可以互为补充，都是优秀的有关计算语言学的著作。如果从文本联贯的角度看，Allen (1995) 仍然是一本值得参考的著作。如果你了解构造一个完整的人机自然语言交流系统需要什么知识，如果你想了解如何用一套理论有机地将词法、句法、语义、语用结合在一个系统里，那么 Hausser (2001) 可能是更好的选择。如果你对基于概率统计的方法有浓厚的兴趣，如果你有足够的时间，那么读读 Manning/Schütze(1999)，否则看看 Charniak (1993) 也能应付了。

## 参考文献

Allen, James (1995) *Natural Language Understanding*. Second Edition. Redwood/CA: Benjamin/Cumming. 中译本《自然语言理解》，刘群等译，电子工业出版社，2004。

ALPAC (1966) *Languages and machines: computers in translation and linguistics*. A report by the Automatic Language Processing Advisory Committee, Division of Behavioral Sciences, National Academy of Sciences, National Research Council. Washington, D.C.: National Academy of Sciences, National Research Council, 1966.

Carstensen, Kai-Uwe; Ebert, Christian; Endriss, Cornelia; Jekat, Susanne; Klabunde, Ralf; Langer, Hagen (2004, ed.) *Computerlinguistik und Sprachtechnologie*. Eine Einführung. 2. Auflage. Heidelberg/Berlin: Spektrum Akademischer Verlag.

Charniak, Eugene (1993). *Statistical Language Learning*. Cambridge: MIT Press.

Cole, R. A. (1997) *Survey of the State of the Art in Human Language Technology*. Cambridge: Cambridge University Press.

冯志伟 (1996) *自然语言的计算机处理*，上海外语教育出版社。

冯志伟 (2003) *机器翻译与语言学研究*，学术讲稿，北京广播学院。

Hausser, Roland (2001) *Foundations of Computational Linguistics: Man-Computer Communication in Natural Language*. The second edition. Berlin-New York: Springer-Verlag.

Hausser, Roland (2000) *Grundlagen der Computerlinguistik: Mensch-Maschine – Kommunikation in natürlicher Sprache*. Berlin-New York: Springer-Verlag.

Huang, Xuedong, Alex Acero, Hsiao-Wuen Hon (2001) *Spoken Language Processing. A Guide to Theory, Algorithm, and System Development*. Prentice Hall.

Hudson, R. A. (1990) *English Word Grammar*. Oxford: Blackwell.

Hutchins, W. John (2000, ed.) *Early Years in Machine Translation*. Amsterdam/Philadelphia: John Benjamins Publishing Company.

Hutchins, W. J. & Somers, H.L. (1992) *An Introduction to Machine Translation*. London: Academic Press.

Jurafsky, Daniel; James H. Martin (2000) *Speech and Language Processing. An introduction to natural language processing, computational linguistics, and speech recognition*. New Jersey: Prentice Hall.

Kempson, Ruth, Meyer-Viol, Wilfried, and Gabbay, Dov (2001) *Dynamic Syntax: The Flow of Language Understanding*. Blackwell Publishers.

Lenders, Winfried & Willee Gerd (1986) *Linguistische Datenverarbeitung. Ein Lehrbuch*. Opladen: Westdeutscher Verlag.

Lobin, Henning & Lothar Lemnitzer (2004, eds.) *Texttechnologie: Perspektiven und Anwendungen*. Tuebingen: Stauffenburg.

刘海涛 (1993) 维特根斯坦语言哲学对计算语义学的影响, 载《计算语言学研究与应用》, 北京语言学院出版社.

刘海涛 (1995) 计算语言学应用中的模块化概念, 《语言文字应用》, 1995.4.

Manning, Christopher D.; Hinrich Schütze (1999) *Foundations of Statistical Natural Language Processing*. MA, Cambridge: MIT Press. 中译本《统计自然语言处理基础》, 电子工业出版社, 苑春法等译, 2004.

Mehler, Alexander & Henning Lobin(2004, eds.) *Automatische Textanalyse: Systeme und*

Methoden zur Annotation und Analyse natuerlichsprachlicher Texte. Wiesbaden: VS verlag fuer Sozialwissenschaften.

Naumann, S./Langer H. *Parsing. Eine Einfuehrung in die maschinelle Analyse natuerlicher Sprache*. Teubner, Stuttgart. 1994.

Mel'čuk, Igor A. (1988) *Dependency Syntax: Theory and Practice*. Albany: State University of New York Press.

Schmitz, Ulrich (1992) *Computerlinguistik: Eine Einfuehrung*. Opladen: Westdeutscher Verlag.

Starosta, S. (1988) *The case for lexicase*. London: Pinter.

Teller, Virginia (2000) Book Reviews: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. *In* "Computational Linguistics" 26(4): 638-641.

Trujillo, Arturo (1999) *Translation Engines: Techniques for Machine Translation*. London: Springer Verlag.

袁毓林 (1998) *语言的认知研究和计算分析*. 北京: 北京大学出版社.

袁毓林 (2001) 计算语言学的理论方法和研究取向. 《中国社会科学》, 4: 157—168.

**后记:** 1999 年我通过比较两本德文的国际语学著作, 写出了题为 "Kiel evoluas interlingvistiko?" (国际语学的发展) 的文章, 此文刊登在 "Language Problems and Language Planning" (23/1: 65-77), 反映不错。于是也想在计算语言学领域如法炮制一篇类似的东西, 目的仍在于理清学科的一些基本问题。天随人愿, 在随后的两年里, 连续出了两本符合我们既定标准的计算语言学著作。本文开始于 2001 年 11 月末, 由于发生了许多不可预见的事情, 直到 2002 年 12 月 31 日晚上 11 时 30 分, 我才完成了初稿。2004 年底, 又根据 CuS 第二版, 对文章做了修改和补充。感谢 Roland Hausser 教授寄赠的 CuS 和他的其它著作, 感谢邓振波先生赠与的 SLP 和其他外文书刊, 感谢 Hutchins 先生的赠书, 感谢 Carstensen 博士寄赠的 CuS 第二版。冯志伟教授、侯敏教授阅读了本文初稿, 并提出了一些修改意见, 在此表示感谢。