工学博士学位论文

基于统计方法的汉语依存句法分析研究

Research on Chinese Dependency Parsing Based on Statistical Methods

马金山

哈尔滨工业大学

2007 年 12 月

国内图书分类号：**TP391.2**
国际图书分类号：**681.37**

工学博士学位论文

基于统计方法的汉语依存句法分析研究

博士研究生：马金山
导　　师：李　生　教授
副 导 师：刘　挺　教授
申请学位级别：工学博士
学科、专业：计算机应用技术
所在单位：计算机科学与技术学院
答辩日期：2007 年 12 月
授予学位单位：哈尔滨工业大学

Classified Index：TP391.2
U. D. C. ： 681.37

A Dissertation for the Degree of D. Eng

Research on Chinese Dependency Parsing Based on Statistical Methods

Candidate：	Ma Jinshan
Supervisor：	Prof. Li Sheng
Associate Supervisor：	Prof. Liu Ting
Academic Degree Applied for：	Doctor of Engineering
Specialty：	Computer Application Technology
Affiliation	School of Computer Science and Technology
Date of Oral Examination：	December, 2007
University：	Harbin Institute of Technology

摘　要

句法分析的任务是根据给定的语法，自动推导出句子的语法结构。句法分析性能的提高将对信息检索、信息抽取以及机器翻译等应用产生重要的推动作用。

在句法分析的研究中，依存语法以其形式简洁、易于标注、便于应用等优点，逐渐受到研究人员的重视。目前，已经被自然语言处理领域的许多专家和学者所采用，应用于多种语言之中。但由于语料资源以及技术等原因，汉语在依存句法分析方面的相关研究并不多。为了弥补这方面的不足，本文使用基于语料库的统计学习方法，对汉语的依存句法分析技术进行了探索。本文的工作分五个部分，具体内容如下：

1. 实现了一个包含分词和词性标注的词法分析系统，并增加了动词子类标注的功能。区分动词的语法属性是为了减少由动词引起的语法歧义，降低句法结构的复杂度。本文首先制定了一个动词细分类体系，将动词分为 8 个子类，然后使用最大熵的方法对动词进行子类标注，用以改善句法分析的性能。

2. 名词复合短语是各种语言中的普遍存在的一种语法结构，对信息抽取、机器翻译等应用有很大的影响。由于传统的句法分析对此类结构的处理不够理想，本文对名词复合短语进行专门处理，以降低句法分析的难度。针对汉语名词复合短语的特点，本文提出一种基于隐马尔科夫树模型的名词复合短语分析方法，较好地解决了此类短语对句法分析的影响。

3. 句法分析对句子的长度非常敏感，随着长度的增加，句法分析的效率以及准确率均会受到严重的影响。为了减少句子长度的影响，本文对句子片段进行识别。先将句子划分为多个片断，并使用基于支持向量机的方法对每个片断类型进行识别；然后对片段进行依存分析，再识别出各片断之间的依存关系，最后将各个片断组合为一个完整的分析树。

4. 根据汉语的特点，探索了一个高效的汉语依存句法分析算法。针对汉语语法结构灵活、树库资源不是非常充分的情况，本文使用分治策略对句子中的一些特定语法结构进行预处理。在搜索算法上，使用动态局部优化的确定性分析算法对句子进行解码，提高了搜索的效率。

5. 为了检验汉语依存句法分析方法的扩展性，并探索单语依存分析同多

语依存分析的不同之处，本文实现了一个基于分步策略的多语依存分析系统，并在 CoNLL 2006 的数据集上进行了实验。通过对实验结果的分析以及同评测结果的比较，验证了本文方法的有效性。

关键词 汉语句法分析；依存语法；名词复合短语；动态局部优化；多语依存分析

Abstract

The goal of parsing is to derive the syntactic structures of sentence according to a certain grammar. The improvement of parsing will give an enormous impetus to natural language processing applications such as information retrieval, information extraction and machine translation.

The statistical parsing community has begun to reach out for dependency grammar that is easy to understand, annotate and use. In recent years, dependency grammar has been adopted by many researchers in the field of natural language processing, and applied to many languages. However, dependency grammar has not been researched fully for Chinese because of the shortage of treebank resource and the problems of technology. To solve this problem the techniques of Chinese dependency parsing are investigated based on statistical learning methods in this paper. The work in this paper falls into five parts that includes:

1. A lexical analysis system that includes segmentation and POS tagging is implemented, and particularly the function of verb subdividing is added to the system. Distinguishing the different properties of verbs aims to reduce the syntactic ambiguities resulting from verbs, and decrease the complexity of syntactic structures. This paper makes a verb subclasses scheme that divides verbs into eight subclasses. Then the maximum entropy method is used to distinguish the verb subclasses to improve the performance of dependency parsing.

2. Noun compounds are popular grammatical structures in many kinds of languages. They have a great influence on some applications such as information extraction and machine translation. Because traditional parsing methods are not good at processing noun compounds, this paper solves the problem specially to reduce the difficulties of syntactic analysis. As to the characteristics of Chinese noun compounds a method based on hidden markov tree model is presented to mitigate the effect of such phrases on the parsing.

3. Syntactic analysis is very sensitive to the sentence length. The efficiency

of searching algorithm and parsing accuracy will suffer with the increase of the sentence length. This paper presents a segment-based method to solve the problem of length in the parsing. Firstly, a sentence is divided into different segments, whose types are identified by SVM classifier. Then the sentence is parsed based on the segments. Finally, all the segments are linked through the dependency relations to form a complete dependency tree.

4. According to the characteristics of language an efficient Chinese dependency parsing algorithm is proposed. For the flexical syntactic structures and the lack of Chinese treebank a divide-and-conquer strategy is used to deal with the specific grammatical structures. In the process of decoding a deterministic parsing algorithm based on dynamic local optimization is proposed which improves the efficiency of searching.

5. To validate the extension of the methods of Chinese parsing and explore the law of monolingual and multilingual dependency parsing this paper implements a multilingual dependency parsing system based on the two-stage strategy. Finally several experiments are carried out on the data set of CoNLL-X shared task 2006. The comparison to the results of shared task is made, and the analysis of experimental results shows the availability of the method in the paper.

Keywords Chinese parsing; Dependency grammar; Noun compounds; Dynamic local optimization; Multilingual dependency parsing

目录

Contents

第1章 绪论

1.1 课题的背景和意义

自然语言处理的分析技术，可以大致分为两个层面，一个是浅层分析，如分词，词性标注。这些技术一般只需对句子的局部范围进行分析处理，目前已经基本成熟，其标志就是它们已经被成功地用于文本检索、文本分类、信息抽取等应用之中，并对这些应用产生了实质性的帮助。另一个层面是对语言进行深层的处理，如句法分析、语义分析。这些技术需要对句子进行全局分析，目前，深层的语言分析技术还没有达到完全实用的程度。

对语言的深层处理过程中，句法分析处于一个十分重要的位置。句法分析工作包括两方面的内容，一是确定语言的语法体系，即对语言中合法句子的语法结构给与形式化的定义；另一方面是句法分析技术，即根据给定的语法体系，自动推导出句子的语法结构，分析句子所包含的句法单位和这些句法单位之间的关系[1]。

在语法体系中，依存语法以其形式简洁、易于标注、便于应用等优点，逐渐受到研究人员的重视。目前，已经被自然语言处理领域的许多专家和学者所采用，应用于许多国家的语言中，并对其不断地发展和完善。在国际会议 CoNLL（Computational Natural Language Learning）的 shared task 中，2006、2007 连续两年举行了多语依存句法分析（Multi-lingual Dependency Parsing）的评测[2, 3]，对包括汉语在内的十几种语言进行依存分析。许多国家的研究人员参加了该评测，说明基于依存语法的句法分析已经被各国学者广泛地接受。

作为的底层核心技术之一，句法分析可在自然语言处理的各项应用中提供直接和间接的帮助。

（1）直接应用

句法分析的结果可直接用于机器翻译、自动问答、信息抽取等应用。目前的机器翻译主要依据短语对齐的结果，而准确高效的句法分析可以提高短语对齐的准确率，改善机器翻译的效果。

在基于自然语言的自动问答中，查询扩展以及答案匹配均需要对句子进

行深入的理解和分析。已有的一些工作将依存分析用于自动问答的问题分类中，取得了较好的效果[4]，也证明了句法分析对自动问答所起的重要作用。

句法分析的另一个直接应用是信息抽取。为了从非结构化的文本中自动抽取特定的结构化信息，句法分析的作用至关重要[5]，Surdeanu 等人在句法分析的基础之上进行信息抽取，提高了信息抽取，特别是事件抽取系统的性能[6]。

（2）间接应用

句法分析同样可以对自然语言处理的基础技术提供帮助。目前的汉语分词主要采用词汇一级分析技术[7]，并已经达到实用水平，但是要将该问题真正解决，只有借助于更深层的分析技术，借助语法语义等信息帮助分词。分词、词性标注以及句法分析的一体化分析技术是一个很有前途的解决方案，Luo 和 Pung 尝试了基于字的句法分析方法，实现了分词和句法分析的一体化，取得了较好的效果[8, 9]。

自然语言处理的目标是对语言进行语义一级的理解，词义消歧和语义分析是达到这一目标的必经之路。而真正要达到语义一级的分析，句法分析同样必不可少。词义消歧面临的困难之一就是上下文特征的选择[10]，依存句法分析能够识别句子中词与词之间的关系，可以用来优化词义消歧的特征选择，依存句法分析对汉语词义消歧的积极作用已经在一些工作中得到了验证[11]。语义分析是另外一个很重要的基础研究工作。在 20 世纪 70 年代就受到学者的重视[12]，最近几年刚刚开始的面向开放域语义的分析，是在句法分析的基础之上进行的，并强烈地依赖于句法分析技术的发展[13]。

综上所述，句法分析位于自然语言处理中的一个核心位置，其性能的好坏，对其他技术有着重要的影响。如能将其有效地加以解决，一方面是对相应的语法理论的验证，能够证明某一语法理论的正确性和有效性，促进语法理论的研究和发展，为人类掌握语言的规律提供实践性的检验。另一方面可以作为自然语言处理技术的一个基础，为语言的深层理解提供一个新的平台，有效支撑各种语义、语用等分析技术。也可以直接对各种上层应用，比如机器翻译、信息获取、自动文摘等提供帮助。

1.2 句法分析的研究方法概述

句法分析的研究有着较长的历史，从 20 世纪 50 年代初机器翻译课题被提出算起，自然语言处理经历了 50 余年的发展，在这期间，句法分析一直是

自然语言处理的研究重点，也是制约自然语言处理研究进展的主要障碍。

在早期的句法分析研究中，基于规则的方法曾一度占据主流。该方法由语言学家描述语言的语法，语言学家认为所有人类语言的构造都是有层次的，层次结构可以用规则的形式表示出来，而规则的集合就是语法。对于一个输入的文字串，根据语法可以推导出该文字串的语法结构。

对基于规则的方法来说，规则的获取是一个十分繁琐的过程，它完全依赖于知识工程师的语言知识和经验。除了开发规则的成本巨大之外，这种方法不能保证系统的性能随着调试句子的增多而提高，有时增加的规则反而会对系统的性能产生负面影响，这是由于人的知识表示能力还存在不足。另外，对基于规则的方法，很难找到一种有效的途径，提高规则开发的效率。

自上世纪 90 年代以来，随着语料资源的获取变得容易，基于统计的方法开始在自然语言处理领域成为主流。这种方法采用统计学的处理技术从大规模语料库中获取语言分析所需要的知识，放弃人工干预，减少对语言学家的依赖。它的基本思想是：（1）使用语料库作为唯一的信息源，所有的知识（除了统计模型的构造方法）都是从语料库中获得。（2）语言知识在统计意义上被解释，所有参量都是通过统计处理从语料库中自动习得的[14]。

基于统计的方法具有效率高、鲁棒性好的优点，大量的实验已经证明了该方法的优越性。目前，统计方法已经被句法分析的研究者普遍采用。

基于统计方法的句法分析，需要解决两个问题：第一个问题是语法歧义的消解。自然语言区别于人工语言的一个显著特点就是它存在大量的歧义现象。人类可以依靠丰富的先验知识有效地消除各种歧义现象，而目前对人类认知的机理还没有完全掌握，在知识表示以及知识获取方面的研究工作也还有诸多不足，如何利用语料库资源建立一个统计模型，用以消解自然语言的语法歧义，是统计方法面临的主要问题。

第二个问题是句法树的空间搜索。同序列标注以及分类的问题相比，句法分析显得更为复杂。在进行句法分析时，一个句子会产生大量的符合语法规范的候选分析树。给定一个长度为 n 个词的句子，其候选句法分析树的数量高达 n 的指数级。因此在设计句法分析模型时不仅仅要加强模型消除歧义的能力，还必须要控制好模型的复杂度，从而保证解码器能够在可接受的时间内搜索到最优的句法分析树。

下面分别针对这两个问题，结合统计方法所需要的树库资源对句法分析的研究方法做一个简单的综述。

1.2.1 树库资源建设

在基于统计的方法中，根据所使用的语料不同，可以分为有指导的方法和无指导的方法。有指导的方法需要事先按照一定的语法规范，人工标注好一些句子作为训练数据，然后通过各种概率统计方法或机器学习方法，从训练数据中获取句法分析所需要的知识。无指导的方法则使用没有经过标注的数据进行训练，按照一定的机制，从中自动学习语法规律。有指导的句法分析是现在的主流方法，目前在英语等语言中已经达到了较高的准确率。

在有指导的句法分析中，事先标注的用于训练的句子集叫做树库。目前绝大多数的统计句法分析模型都是利用标注好的树库以有指导学习方式来训练模型的参数。因此，树库建设是一个非常重要的工作，其质量和规模直接关系到句法分析的训练效果。目前在这方面开展了大量的研究和开发工作。

书写描述各种语言现象的句法规则是十分困难的，但人工对一个具体的句子进行句法分析相对要容易一些。当前最著名的树库是 Penn Treebank，是目前使用最为广泛的英文树库。Penn treebank 前身为 ATIS 和华尔街日报(WSJ)树库，它的第一版出现于 1991 年中期， 第二版出现于 1994[15]。从第一版出现到现在 Penn treebank 已发展了十几年，整个过程一直都在维护、修正。Penn Treebank 具有较高的一致性和标注准确性，已经成为当前研究英语句法分析所公认的训练集和测试集。

随着 Penn Treebank 的成功，其他语言的树库标注工作也纷纷展开，德语、希伯来语、日语、西班牙语等语言都建立了自己的树库，开展句法分析的研究工作[16-19]。汉语方面，相应的树库建设工作也已经开始展开，较早的是台湾中央研究院标注的 Sinica 树库[20]、美国宾夕法尼亚大学 Penn Chinese treebank 树库[21]，以及清华大学的汉语树库[22]。这些树库多采用短语结构的标注形式。

除了短语结构之外，基于依存的树库建设工作，在国外的一些语言中也已经开始展开。比较著名的依存树库有捷克语的布拉格树库[23]，英语的 PARC 树库[24]，以及俄语、意大利语等语言的树库[25, 26]。在汉语的依存树库建设方面，Lai 等人开展了依存语法的分析和标注工作，但其标注语料的规模较小，有 5000 词左右[27]。

能够看出，树库资源建设方面已经取得了很大的成绩，但是汉语依存树库的建设还存在很多不足，这为汉语依存分析的研究工作带来了困难，也成为本文工作所要解决的一个主要问题。

1.2.2 句法分析的统计模型

句法分析的统计模型是指利用统计的方法，从文本数据中自动学习语言的语法构成规律，对其参数化之后所构成的概率模型。随着数据资源的不断丰富和机器学习方法的快速发展，统计模型的构建方法也在不断变换，下面对当前使用较多的几种建模方法进行介绍。

1.2.2.1 基于生成模型的统计方法

生成模型将概率函数 $Score(x,y)$定义为联合概率 $Score(x,y|\theta)$，使目标函数 $\prod_{i=1}^{n} Score(x_i,y_i;\theta)$ 最大的 θ 值作为模型的参数。其中，x 是已知序列，y 是目标序列。对句法分析来说，已知序列是输入的句子 $S = w_1, w_2, \ldots, w_n$，$w_i$ 是句子中的第 i 个词；输出是一棵结构化的语法树 T。生成模型的目标是从训练语料中获取参数 θ，使联合概率 $P(T,S)$最大，则最优的句法分析树 t^*为

$$t^* = \arg\max_{t \in T} P(T,S)$$

为了简化问题，也为了减少数据稀疏，通常要对句法分析问题作不同程度的独立假设，以便于计算联合概率 $P(T,S)$。同时，需要研究者根据语言的规律，找出最具有判别力的信息计算模型的参数。

选择哪些信息用以消解语法歧义，是建立生成模型的难点。虽然统计句法分析不需要像基于规则的方法那样，人工将有利于歧义化解的知识用规则的形式的逐条罗列出来，但是统计模型同样需要这些语言知识。这需要模型设计者具有较强的语言直觉，对语言的规律有深入的理解，才能在设计模型时选择区分能力强的语言信息，使模型能够自动获取这些知识。句法分析中，生成模型的构建主要使用三类信息：词性信息、词汇信息和结构信息。

（1）词性信息

对句法分析来说，词性是最简单，也是最重要的信息，几乎在任意一个句法分析模型中，都使用了词性信息计算模型的参数。在早期的基于规则的系统中，主要以产生式形式推导句法树，而产生式由非终结符和词性标记构成，词性是计算条件概率的最主要信息[28-30]。在近期的句法分析系统中，Klein 使用词性信息实现了一个非词汇化的句法分析模型性，在 Penn Treebank 上的分析结果接近了当前的最高水平[31]。汉语方面，周强从标注语料中自动获取结构优先关系，最大限度地消除了词性规则所产生的歧义，并实现了一个汉语句法分析系统[32]。Ma 基于一个小规模的标注数据，利用词性信息估计依存关系的概率，实现了一个汉语的依存分析系统[33]。

但由于词性标记的数量少，信息粒度偏粗，不可避免地会产生大量的句法歧义。目前，随着数据规模的增大，词汇信息已被广泛使用，但作为词汇参数的平滑手段，词性仍然是不可或缺的重要信息。

（2）词汇信息

词汇本身是最有区别力的信息，语言在词汇层面上，几乎是没有歧义的（除了极少数语用上的歧义）。人们很早就注意到词汇信息的优点，Hindle 早在 1993 年就利用词汇信息解决介词短语的附着歧义问题[34]。但词汇化方法在句法分析中的应用是最近几年随着树库资源的丰富才逐渐开始，尤其是英语树库 Penn Treebank 的出现。

Jelinek、Magerman 等人的 SPATTER 句法分析器，是最早应用词汇信息句法分析系统之一，在英语的 Penn Treebank 上获得了超过 84%的准确率，是当时在相同测试集所达到的最好结果[35, 36]。

词汇以及词汇之间的依存，包含着丰富的表征信息，这使得越来越多的句法分析器趋向于词汇化。Collins 在 1996 年利用二元词汇依存关系，实现了一个新的统计句法分析器[37]。该方法先将句法分析分为两部分，一部分是 BaseNPs，一部分是依存结构，再根据这两部分进行句法分析。在识别依存结构时，利用了词汇的信息。Collins 于 1997 年对该模型进行改进，实现三个词汇化的生成模型[38]。其基本思想是将规则词汇化，按照 Magerman 和 Jelinek 的方法获取句法成分（constituents）的核心词，然后计算每条规则的概率。其规则的表示形式不再只包括非终结符和词性标记，而是增加了核心节点的词汇信息。

Collins 的模型取得了很大的成功，其准确率接近了 88%，超过了以前的所有句法分析系统。该方法也极大地促进了词汇信息的应用，之后，大多数的句法分析系统都使用了词汇信息[39-41]。其中 Charniak 指出，句法分析器的词汇化，是近几年来统计句法分析研究发生的最大变化，词汇化使英语句法分析器性能从 75%增加到 87%-88%。

词汇化建模中存在的难点是词汇信息的数据稀疏问题。在生成模型中，主要采用词性信息进行平滑，如将词单元对〈词，词〉回退为〈词，词性〉、〈词性，词〉、〈词性，词性〉[37, 38]，随着树库资源的进一步丰富，以及各种知识库的建立，该问题有望得到缓解。

（3）结构信息

这里指的结构信息，是指能够反映语言规律的句法特征。词性和词汇是结构信息的构成单元，结构信息通过词性和词汇体现出来。如果将最终的句

法树比作躯体，词性和词汇就是躯体的血肉，而结构就是躯体的骨骼。

同前两种信息相比，结构信息最为复杂。在生成模型中，它需要研究者根据语言学知识或语言上的直觉对结构信息进行提取，然后融入到模型之中。

在短语结构树中，最简单、最基本的结构信息是句法标记，也就是短语结构语法中的非终结符，如名词短语 NP、动词短语 VP 等。句法标记的缺点是过于粗糙，包含信息量少。如动词短语 VP 有单复数之分，及物非及物之分，但是树库中只用一个 VP 来表示；做主语的名词短语 NP 与做宾语名词短语 NP 是不同的，无主语的 S 与有主语的 S 同样也是不同的。

针对这个问题，Johnson 通过扩展句法标记的方法来重新表示语法，他利用非终结符节点所在的上下文结构进行辅助标注[42]。该标注基于这样的假设：分析树中的节点标记可看作是一个“信道”，它负责该节点管辖的子树和其他子树之间的信息传递。如果将上下文信息附加在该节点标记上，可以使这个信道传递更多的信息，从而削弱 PCFG 模型的独立性假设。采用什么样的标注方法更有效，与具体的语言、树库有关。Levy 指出，利用祖父节点、父节点辅助标注，ETB（English Penn Treebank）的效果要强于 CTB（Chinese Penn Treebank）的效果[43]。

另一种重要的结构信息是动词次范畴。Collins97 的模型 2 中时，就利用了次范畴知识[38]。通过核心词增加次范畴概率，识别句子的补足/附属结构（complement/adjunct），提高了句法识别效果。

除了上述两种较为通用的结构信息之外，还有很多和语言相关的结构特征。如（Collins, 1997）在设计模型3时，对英语中的 wh-movement 结构进行处理；孙宏林、吴云芳等对汉语的并列结构进行了识别[44, 45]。Levy分析了汉语句法分析中的几种常见错误：NP-NP修饰错误，IP/VP附着错误等，然后对对每种歧义现象分别标注额外的信息加以解决[43]。

针对结构信息的自动获取以及它的语言相关问题，很多研究者尝试利用已标注的树库自动获取语法规则[46]，或者直接从未标注的原始语料中自动抽取动词的次范畴[47]，以提高结构信息的获取效率。

1.2.2.2 基于判别模型的统计方法

判别模型将概率函数 $Score(x,y)$定义为条件概率 $Score(x|y;\theta)$，使目标函数 $\prod_{i=1}^{n} Score(y_i \mid x_i;\theta)$ 最大的 θ 值作为模型的参数。其中，x 是已知序列，y 是目标序列。

McCallum，Laffery 和 Johnson 等对生成模型和判别模型在句法分析和序

列标记上的优劣作了对比[48-50]。一般而言，判别模型的参数估计采用的是对数线性模型，比生成模型能容纳更多的特征。判别模型常以分类器的形式用于句法分析，首先把句法分析分解为一系列操作，然后用分类器选择当前应该执行的操作。在句法分析中应用较多的有如下几种判别模型。

（1）最大熵（Maximum Entropy, ME）

Ratnaparkhi 最早将最大熵方法应用于英语句法分析[51]，他将句法分析的过程分为三个步骤，每个步骤由三个操作组成：Start, Join, Other，然后根据上下文特征，利用最大熵的方法预测在当前环境下应该执行哪一个操作。其上下文特征主要包括：成分的核心词，核心词的组合，非特定组合（less-specific）信息，以及部分已完成的子树（limited lookahead）信息。

中文方面，Luo 和 Pung 分别应用最大熵方法进行汉语的句法分析[8, 9]，分析的过程同 Ratnaparkhi 的方法类似，不同之处在于，他们结合汉语需要分词的特点，实现了基于字的中文句法分析。

（2）支持向量机（Supported Vector Machines, SVM）

支持向量机是基于 Vapnik 提出的统计学习原理构建的一种线性分类器[52]，其基本思想是使构成的超平面分割训练数据能够获得最大的边缘（Large Margin）。由于支持向量机理论的完备及其良好的应用效果，经常被用作分类器处理文本分类等各种自然语言处理问题[53]。

Kudo 和 Yamada 等人将 SVM 方法应用于句法分析，在日语和英语上分别进行了试验[54-56]。Cheng 和 Jin 分别使用 SVM 进行汉语的依存分析，其过程和（Yamada，2003）大致相同，区别在于对依存分析的操作以及操作的特征选择进行了改进[57-59]。

支持向量机的主要缺点是其训练效率偏低，（Yamada，2003）为了解决 SVM 训练成本过高问题，使用了 pairwise 的方法缓解[56]。支持向量机的另一个缺点就是其对于输出结果不能从概率上进行解释，也就是不能准确地给出各个输出结果的概率分布，这就给一些利用概率结果的后处理应用带来了麻烦，也限制了它在句法分析这种对概率需求较强的任务中的应用。

（3）决策树（Decision Tree）

决策树是另外一种比较典型的判别学习方法[60]，它利用一系列的查询问答来判断和分类某一模式，后一问题的提法依赖于前一问题的回答，将全部问题集用一棵有向树表示，称为决策树。这种“问卷表”方式的做法对非度量数据特别有效。分类过程首先从根节点开始，对模式的某一属性的取值提问，根据不同的回答，转向相应的后续子节点，直至到达叶节点，表明没有问题

可问了，也就得到了问题的答案。

决策树的形式是呈树状的规则，因此学习结果容易观察，而且便于修改。Jelinek 和 Magerman 首先将决策树的方法应用于句法分析，在英语的 Penn Treebank 上取得了 84%以上的准确率[35, 36]。决策树学习方法的问题在于处理高维问题时效果不够理想。

（4）其他方法

传统的句法分析方法可归结为 History-based (HB) models，此类方法将句法分析树表示为一个派生(derivation)，即一个决策序列，句法树的概率为所有决策的概率乘积。此类方法的一个缺点是很难将各种特征融合到统一的框架中。

Collins 提出一种重排序的思想用以解决句法分析的特征融合问题。重排序的基本思想是首先用一个句法分析器输出多个候选结果，并带有一个初始的排序，然后建立一个排序模型，对候选结果进行重新排序，该排序模型可以很容易的融合各种特征，这样就可以充分利用判别模型的优点。Collins 尝试了使用一种对数线性模型—马尔科夫随机域（Markov Random Fields, MRFs)和增强模型（boosting mode）[61]，以及用感知器算法进行重排序[62]。Charniak 使用最大熵方法进行重排序，均取得了很好的结果[63]。

1.2.2.3 基于无指导的统计方法

树库标注是一个非常繁重的工作，需要花费大量的人力。而未经过加工的生语料却很容易获取，并且语料的规模几乎不受限制。于是，自动地从原始语料中学习句法知识，用无指导的方法进行句法分析，受到了人们越来越多的重视。

Yuret 早在 1998 年就曾用无指导的方法识别依存关系。他首先计算文本中词的互信息，包括相邻词和远距离词，该过程是一个带反馈的过程，然后使用搜索算法进行依存分析。这个工作中只对实词的依存关系进行了识别，未识别虚词[64]。

周强也在 1998 试验了汉语概率型上下文无关语法的自动推导，首先用句法规则自动构造工具生成初始规则集，通过对训练语料的自动短语界定预处理以及集成不同的知识源，保证了初始规则集的合理和有效，然后使用 EM 算法进行匹配分析，算法能够快速地收敛于符合语言事实的规则概率分布状态。通过和树库获取的规则集进行比较，该方法获得的 800 多条规则表现了较高的可靠性[65]。

Sarkar 在 2001 年使用 Co-training 的方法进行句法分析，Co-training 的方法是一种半无指导的学习方法，首先用一个少量的标注数据作为种子集，然后，在较大的未标注数据上进行无指导的迭代训练，提高了只利用种子集进行句法分析的效果[66]。

Gao 在 2003 年用无指导的方法建立一个汉语的依存句法分析器，其无指导依存分析的参数训练分为两步：（1）将词分为核心词和修饰词，利用词性确定句子中的核心词；（2）在句子中设定一个窗口，包含 3 核心词，组合成三对依存关系，然后使用最大似然估计，用 EM 的方法进行迭代。Gao 将无指导方法获取的依存关系同 N-gram 结合，改进了语言模型[67]。

Klein 在无指导句法分析上进行了较为深入的研究，分别在英语 Penn Treebank，德语的 NEGRA 树库，汉语的 CTB 上进行无指导依存分析，并提出了一个 DMV(dependency model with valence)模型。模型基于词类信息，并使用了配价(valence)信息，即考虑核心节点已有的论元(argument)对后来的论元产生的影响。模型用 EM 算法进行参数估计，然后，又使用了一个组合模型，结合 DMV，达到较高的分析准确率：英语为 47.5%，汉语为 55.2%[68]。

尽管无指导的方法省略了手工标注语料的繁重劳动，但用于学习的数据本身没有标注正确的结构，其效果不如有指导的方法。所以无指导的方法通常用来辅助手工标注语料，或当训练数据较少时作为一种平滑方法[69, 70]。总体来说，无指导的方法现在还不是很成熟，还没有形成理论化的研究方法，目前尚处于探索阶段。但是根据自然语言处理的发展方向以及无指导方法所具有的优点，无指导句法分析将是一个很有前景的研究方向。

1.2.3 句法分析的搜索算法

早期句法分析的搜索算法主要来自于形式语言理论，其中最为著名的就是 LR 算法。它的前提是形式语言的语法必须是没有二义性的，即句法分析结果必须是确定的。自然语言的语法充满了歧义，一个句子可能有多个符合语法的分析结果，所以上述的方法不适合自然语言的句法分析。为此，人们对其进行了改进，其中比较著名的是 Tomita 在 80 年代提出的 GLR 算法[71]。

目前的自然语言句法分析系统中，搜索算法通常和概率模型统一起来，即在解码的同时，根据模型的概率选择最优的分析结果。一般来说，当前的句法分析算法可以分为两类：基于全局寻优的搜索策略和基于局部寻优的搜索策略。

1.2.3.1 基于全局寻优的搜索算法

这种方法是基于动态规划的思想，也叫做基于线图（chart）的分析技术，分自底向上和自顶向下两种。其关键在于采用动态规划的思想保留已获得的中间分析结果，从而避免回退，节约时间。CKY 算法是典型的自底向上搜索算法[72]，标准 CKY 处理的规则必须表示为乔姆斯基范式的形式。Earley 提出的算法是自顶向下算法中比较有代表性的一种[73]。之后的许多句法分析算法都是在这两种算法的基础上进行改进，主要的改进方式是取消规则形式的限制，引入路径搜索策略的启发式规则，以及对规则进行概率化。

在搜索算法的时间复杂度方面，一般忽略规则的查找和匹配，只计算生成语法树过程中的时间复杂度。以 GLR 为代表的基于下推自动机的算法，随着歧义数量的增加，时间复杂度可能呈指数级增长[74]，以 CKY 为代表的基于动态规划算法，一般认为时间复杂度为 $O(n^3)$，但 Eisner 指出，对自然语言的分析，这种算法的时间复杂度实际上达到了 $O(n^5)$，并提出一种在立方时间之内，根据依存文法进行句法分析的搜索算法[75-77]。Lai 等借鉴了这种方法进行了汉语的依存分析[78]。

Eisner 算法的核心思想是，针对依存文法的特点，以 span 结构代替以前的 constituent 结构，在常数时间内实现两个 span 组合成新的 span（两个 constituent 组合成新的 constituent 的时间复杂度约为 n^2）。Eisner 将 span 定义为这样的子串：子串内部的词不同外部的词发生联系，而只是通过边界词（end word）联系。而 constituent 通过它的核心词（head word）同外部联系，其中核心词可以位于 constituent 的任何位置。Span 可以是一段连续的依存弧，也可以是两段独立的依存弧，或者一个单独的节点和一段依存弧。算法在进行组合操作时，要求左边的 span 要么是一段封闭的依存弧，要么是只有两个节点，然后对两个 span 进行合并或者连接，生成一组新的 span，这个操作在常数时间内完成。除了这部分处理不同之外，该算法的其余步骤也是基于动态规划的思想，和其他的搜索算法大致相同。

1.2.3.2 基于局部最优的搜索算法

局部寻优的算法分为两种，一种对全局寻优的算法进行简化，将立方时间的算法改为平方或线性时间；另一种是使用确定性分析技术，算法的时间复杂度为线性。

（1）简化分析算法

Gao 实现了一个简化的搜索算法[67]，该算法从左向右依次扫描句子中的

每个词，每扫描一个词，都将它同前面的所有词建立依存关系，并将这个关系压入堆栈中，然后检查堆栈中的关系是否存在交叉（crossing）或环路（cycle），如果存在，则将产生冲突的依存弧中概率最小的弧删除。算法根据概率大小决定依存关系的强度，当一个强依存关系同几个弱依存关系交叉时，即使几个弱依存关系的概率之和大于强依存关系，也删除这几个弱依存关系，而保留强依存关系。这样的处理使算法的时间复杂度为 $O(n^2)$。

Ratnaparkhi 实现了一个线性观测时间的搜索算法[51, 79]。算法分三遍对句子进行扫描，第一遍进行词性标注，第二遍进行组块识别，第三遍进行句法成分的建立（Build）和检验（Check）。算法使用广度优先的策略进行搜索，每次只扩展前 K 个分值最高的中间分析结果，最终获取 M 个分析结果。句法成分的建立过程被分为若干操作，在预测操作时，只取概率值之和为 Q 的前 m 个操作。整个过程实际是一种贪心算法，具有线性属性。

（2）确定性分析算法

确定性分析算法被较多地用于依存句法分析，Harri 实现了一个芬兰语的句法分析器，在线性时间内进行二元依存结构的确定性分析[80]。该方法将依存结构对应一个图（government map），即一个二维矩阵，分析过程即是填充该矩阵的过程。算法在分析句子时，进行分段处理，分段的边界是连词和分隔符。另外，其依存关系多为确定性的，即一个关系类型基本上能确定地对应两个词性，这也是算法能够进行线性时间分析的原因。

Kudo、Yamada 等人使用确定性算法对日语、英语、汉语进行了分析[55, 56, 58, 59]，分析过程用 SVM 来消除句法分析中的歧义，把句法分析的不确定性过程变成一个确定性的过程。英语分析中，首先将 Penn Treebank 转换成依存结构，然后用确定性算法进行分析，算法包含三个操作：Shift—当前两个词没有关系；Left—左边的词是核心词；Right—右边的词是核心词。用 SVM 根据当前状态判断下一个操作。分析过程类似于移进—归约操作，能够在线性时间内将句子归约完毕[56]。

Nivre 使用类似的过程完成了对瑞典语、英语的确定性分析，不同之处是 Nivre 使用 Memory-based 的方法消除句法歧义，同时句法操作包含 Left-Arc，Right-Arc，Shift，Reduce 四种[81-83]。

确定性分析过程只分析当前的操作，不保存以前的分析结果，所以能够实现线性分析，但是不能够回溯，这是该算法的一个不足之处。

1.3 依存句法分析的研究现状

句法分析首先要遵循某一语法体系，根据该体系的语法确定语法树的表示形式。目前，在句法分析中使用比较广泛的有短语结构语法和依存语法。如下面的句子：

西门子将努力参与中国的三峡工程建设。

其基于两种语法的分析结果分别如图 1-1和图 1-2所示。

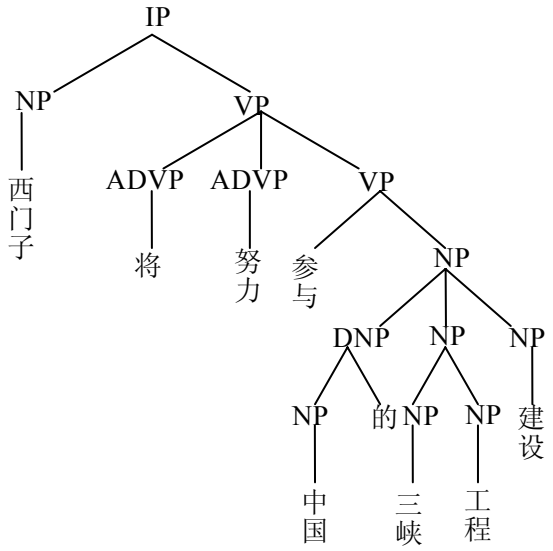


图1-1 基于短语结构语法的分析树

Figure 1-1 The parsing tree based on phrase structures

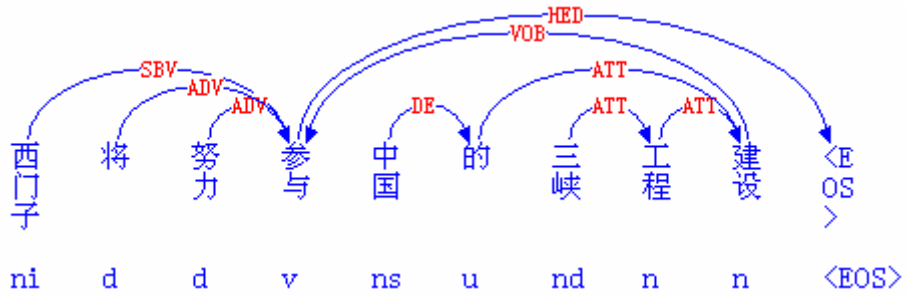


图1-2 基于依存语法的分析树

Figure 1-2 The parsing tree based on dependency grammar

短语结构树由终结点、非终结点以及短语标记三部分组成。根据语法规则，若干终结点构成一个短语，作为非终结点参与下一次归约，直至将整个句子归约为根节点。

从图 1-2的依存树中能够看到，依存分析的结构没有非终结点，词与词之间直接发生依存关系，构成一个依存对，其中一个是核心词，也叫支配词，另一个叫修饰词，也叫从属词。依存关系用一个有向弧表示，叫做依存弧。在本文中，规定依存弧的方向为由从属词指向支配词。图 1-2中，每个依存弧上有一个标记，叫做关系类型，表示该依存对中的两个词之间存在什么样的依存关系。本文将依存关系分为 24 种类型，具体类别将在第四章介绍。

在本文中，采用了依存语法作为句法分析的语法体系，主要基于以下几点考虑：

（1） 依存语法的表示形式简洁，易于理解。这一点可以从图 1-1和图 1-2的对比中看出。依存语法直接表示词语之间的关系，不额外增加语法符号。即使没有语言学背景的人也很容易掌握该语法形式，这对于树库建设工作非常有利。

（2） 依存语法的表示侧重于反映语义关系，这种表示更倾向于人的语言直觉。这有利于一些上层应用，如语义角色标注、信息抽取等。

（3） 基于依存语法的表示有利于实现线性时间的搜索算法，除了效率提高，这些搜索算法的准确率也达到了当前的最高水平。

（4） 依存语法在表示交叉关系时更有优势。欧洲语言中，很多语言的语序灵活，交叉关系的比例较大，如荷兰语、捷克语等。同其他语法相比，依存语法无论在表示形式还是句法分析方面，均能较好地处理此类问题。

（5） 应用依存语法进行句法分析的语言范围较广。由于短语结构语法的提出和发展均是以英文为基础，目前的研究更多的集中在英语之上，而依存语法的研究在欧洲的许多种语言中均以开展。

1.3.1 英语依存分析

英语的句法分析中，短语结构语法一直占据主流，依存语法的理论与实践研究开展得则要晚一些。Melchuk 对英语的依存语法理论做了全面系统的研究[84]，Eisner 最先将 Penn Treebank 转化为依存表示形式，然后进行依存句法分析的实验[75, 76]。在数据转换时，Eisner 排除了包含连词的句子，对其余的句子使用规则进行自动转换。实验中，Eisner 使用面向说话者的生成模

型，得到了 90.0%的依存准确率。

近几年，越来越多的研究者投入到依存句法分析的工作中来，并对英语的依存体系进行完善[85-89]。在实践方面，Yamada 等人将 Penn Treebank 中的句子完全转换为依存结构，然后使用确定性的分析算法，获得了 90.3%的准确率[56]，为英文依存分析工作奠定了坚实的基础。在此基础上，Nivre 和 McDonald 将英语的依存分析工作进一步深入，在以下几方面对英语依存分析进行了推动。

（1） Penn Treebank 转换为依存树库。在 Magerman、Collins 以及 Yamada 等人的工作基础上，Nivre 对 Penn Treebank 到依存结构的转换工作做了进一步的完善，开发了一套实用的转换工具 Penn2Malt[1]，为英语的依存分析工作提供了统一的数据平台[90]。

（2） 探索了一套高效的确定性分析算法。除了效率之外，确定性搜索过程在依存分析中还表现了较好的准确率，其效果在近期一些工作中得到了证实[56, 83, 91]。

（3） 提高了依存分析的准确率。英语的依存分析虽然起步较晚，但目前在 Penn Treebank 上的分析结果已经接近了基于短语结构的分析[83, 92, 93]。

1.3.2 汉语依存分析

在汉语方面，依存句法分析的工作在最近几年开始受到重视。Zhou 是最早从事这方面的研究者之一，他采用分块的思想，应用一些制定的语法规则，先对句子进行分块处理，找出关系固定的语块，然后再对整个句子进行依存分析[94]。Lai 等人使用基于 span 的思想、Gao 等利用无指导的方法在汉语依存分析方面做了有价值的研究工作[67, 78]。

随着汉语应用的日益广泛，国外也开始了汉语依存分析的研究工作。Cheng 等人分别在 CKIP 树库和 Chinese Penn Treebank (CTB)上进行了依存分析的实验。在基于 CKIP 树库的实验中，首先将树库自动转化依存结构，然后利用确定性搜索算法进行依存分析[58]。实验使用了 CKIP 树库中的部分数据，平均句长 5.7 词。根据不同的篇章类型分别进行测试，其中，文学类效果最好，准确率分别为：依存关系 87%，句子核心词 94%，整句 71%；新闻类效果最差，依存关系 74%，核心词 86.9%，整句 50%。在基于 CTB 的实验中，作者对 Nivre 的方法进行了改进，一个是增加了全局特征，另一个是在

[1] http://w3.msi.vxu.se/~nivre/research/Penn2Malt.html

寻找句子根节点时，对根节点两侧的句子分别分析，降低复杂度。训练数据是将 CTB 5.0 进行转换，转换后以逗号作为句子的末尾。实验获得了 86.18% 的依存关系准确率[59]。

Jin 等人针对 Nivre 和 Yamada 方法的缺点，使用两阶段移进-归约方法分析汉语的依存关系，第一阶段归约左边的依存弧和右边的体词性节点，第二阶段归约右边的动词性依存节点，对 Nivre 的算法进行了改进。训练时，按照周明的依存体系对 CTB 4.0 进行转换，抽取其中的部分句子进行实验，准确率为 84.42%[57]。

中文依存分析的研究工作虽然取得了一定的进展，但在以下几方面还存在着一些问题需要进一步解决：

（1）树库资源

英语句法分析的发展得益于 Penn Treebank 的建立，Penn Treebank 的规模大，标注质量高。更为重要的是，它已经成为英语句法分析事实上的标准，几乎所有的研究工作都基于该树库进行，这就使得大家的研究工作可以比较，可以继承。同时，将 Penn Treebank 转换为依存结构的工作也已经成熟。反观汉语方面，树库建设工作还有差距，既缺少统一的依存标注体系，也缺少大规模的依存树库。

（2）词法分析

汉语不同于英语，在句法分析之前，需要先进行分词，这无疑增加了句法分析研究的工作量。目前的分词技术虽然已经较为成熟，但是机器分词的准确率最高约为 96%左右[7]，这对汉语的句法分析性能造成了一定的影响。另外，汉语词汇使用灵活，并且没有词语变形信息，一个词经常能够充当多种语法成分，尤其是动词，导致句子中语法歧义的数量增大。

（3）语言相关性

目前汉语的句法分析主要是借鉴英语句法分析的研究成果，将在英语方法表现较好的模型移到汉语中。这种方式虽然节省了研究的成本，提高了汉语句法分析研究的速度，但是也忽略了语言本身的差异，以及方法的适应性。比如，在英语方面表现很多很好的 Collins 模型 3，其中用到了一些同英语相关的语言现象，如 wh-movement 结构的处理。而汉语中，存在着大量语言相关的语法结构，对这些特定的语法结构，目前还缺乏针对性的研究。

（4）搜索算法

由于目前的中文依存句法没有达到实用的程度，在解码的过程中，效率问题还没有引起研究者的重视，更多地将重点放在了依存分析准确率的提高

上。而一个好的搜索算法除了分析的准确率较高之外，在解码速度上也要达到应用的要求。

1.3.3 多语依存分析

多语依存分析是单语分析发展的一个自然延伸。由于依存语法在各种语言中被广泛接受，瑞典、捷克、丹麦、保加利亚、日本以及英语等不同语言的依存分析器纷纷出现[55, 82, 93, 95-97]，人们开始关注语言无关的依存句法分析技术。

2006 年的 CoNLL-X shared task 将多语依存分析作为评测的内容，极大地促进了多语依存分析技术的进展。该评测任务中，提供了 13 种语言的训练数据，数据的格式一致，评测者需要使用统一的模型来处理不同语言的数据[2]。评测结果让人看到了多语依存分析的发展前景：一个多语依存分析器可以使多种语言同时达到较高的准确率[98-100]。但多语依存分析也暴露出其内在的缺点，即对部分语言，尤其是语言规律同其他语言有较大差异的，其准确率仍然偏低。

目前，对特有的一些语言现象，研究者已经给与了充分的重视。比如对于依存分析中的交叉弧问题（non-projective），已经得到了很好的解决[93, 97, 101, 102]。

由于多语依存分析在开发成本、应用范围上所表现出的优势，使得研究者对该方向表现出越来越浓厚的兴趣，在 2007 年的 CoNLL-X shared task 中，又举行了多语依存分析的评测。

1.4 本文的主要研究内容

由于汉语依存分析的起步较晚，而且在树库资源建设方面也落后于英语等语言，使得汉语在依存分析技术方面显得有些落后。另一方面，依存句法分析表现出了良好的应用前景，如果能在一些关键技术上有所突破，可以满足多种信息处理系统的要求。结合国内外的技术发展趋势，以及中文信息处理的战略发展需求，本文将对汉语的依存句法分析进行深入的研究，主要内容包括以下几部分：

第一章首先介绍本文的研究目的和意义，概述了国际上句法分析方法及其发展趋势。然后分别对英语、汉语以及多语的依存句法分析做了简单的总结。

第二章的内容是汉语的词法分析，包括汉语的分词以及词性标注技术。本章提出了基于最大熵方法的动词子类标注，即将动词等词性进行细分类，通过动词的子类来增加动词在句法上区分力，细分类之后的动词减少了句法结构的歧义，提高了依存分析的准确率。

第三章提出了基于隐马尔科夫树模型的名词复合短语分析。名词复合短语是一种普遍存在的语言现象，对信息抽取、信息检索等应用均有帮助，本章用其改善依存分析的性能。在短语的分析上，结合语言现象的特点，本章引入了隐马尔科夫树模型，在识别名词复合短语边界的同时，也分析了短语的内部依存结构，提高了依存分析的准确率。

第四章的内容是识别汉语句子的片段。在分析长句子时，无论在复杂度还是在准确率方面均有很大的难度。本章首先根据标点的功能将句子划分为不同的片段，然后使用基于 SVM 的方法进行片段识别，最后对各个片段进行依存分析，简化了分析的复杂度。

第五章提出了一个基于动态局部优化的汉语依存分析方法。该方法首先将句子中一些特有的语言结构提取出来，然后应用确定性的分析算法对句子进行解码。同以往工作不同的是，本文在建立依存分析模型时，引入词汇的支配度，有效消除句子中的非法结构；在搜索过程中，本文提出了一种动态寻找局部最优点的技术，既保证了搜索的效率，又改善了准确率。

第六章实现了一个分步策略的多语依存分析器。本章将依存分析分为两个步骤，第一步对句子进行依存骨架分析，第二步根据骨架分析的结果，使用基于 SNoW 的方法标注每个依存弧的关系类型。最后给出了本方法在 CoNLL-X shared task 2006 上的分析结果。

第2章 基于最大熵方法的动词子类标注

2.1 引言

词法分析处于自然语言处理的基础地位，对句法分析等工作有着重要的影响。作为句法分析的重要组成部分，本文首先需要解决汉语的词法分析问题。汉语的词法分析主要包括两部分内容：分词和词性标注。

（1）分词

汉语的词与词之间没有自然的界限，在对汉语进行分析之前，首先需要进行自动分词的工作。其中，需要解决的难点问题是歧义词切分和未登录词识别。分词中的歧义有两类，组合型歧义和交集型歧义，一般性的定义可以表述为：

交集型歧义(Overlapped ambiguities)：A、X、B 分别为汉字串，如果其组成的汉字串 AXB 满足 AX 和 XB 同时为词，则汉字串 AXB 为交集型歧义字段。

例如：“研究生命的起源”可以切分为

研究 生命 的 起源

研究生 命 的 起源

其中，“研究生命”为交集歧义字段。

组合型歧义(Combinatorial ambiguities)：汉字串 AB 满足 A、B、AB 同时为词，则该汉字串为组合型歧义字段。

例如： “他从马上下来”可以切分为

他 从 马 上 下来

他 从 马上 下来

其中，“马上”为组合型歧义字段。

目前，歧义切分的研究已经比较成熟，通过统计和规则相结合的方法，歧义字段的正确切分已经达到了较高的水平[103, 104]。

未登录词是指没有在词表中出现的词，也称作 OOV(out of vocabulary)，主要包括：

中国人名，如：韦小宝，赵一曼

外国人名，如：哈迪库斯，卡里姆·哈杰姆

地名，如：李家庄，热那亚

机构名，如：新华社，联合国

其他专有名词，如：白虎团，道—琼斯

数词，如：５０％，３００万

时间词，如：１９９２年，２９日

词语的重叠形式，如：看看，看一看，打听打听，高高兴兴

专业术语，如：线性回归，韦特比算法

新词，如：非典，博客

上述的未登录词类别较广，而且有些未登录词的识别已经不属于词法分析范畴。比如，复杂机构名通常由命名实体识别来完成，新词通常由专门的新词发现工作负责，专业领域的术语通常由信息抽取工作负责。本文的词法分析主要处理其他几类未登录词，对数词、时间词、词语的重叠形式通过制定语法规则的方法进行处理，对人名、地名、机构名以及专有名词使用基于角色标注的方法进行识别。

未登录词的数量开放，在自然语言处理中，无论如何扩大词表，未登录词都无法避免。与分词的歧义切分相比，未登录词的识别难度更大，已经成为影响分词准确率的最主要、最直接的因素，是当前词法分析的一个研究热点。

（2）词性标注

词性标注就是在给定句子中判定每个词的语法范畴，确定其词性并加以标注的过程。词性标注中需要解决的是词性歧义问题，词性歧义也叫词性兼类，指自然语言中一个词语有多个词性的现象，例如下面的句子：

S1=“他是哈尔滨工业大学的教授。”

S2=“他在哈尔滨工业大学教授计算语言学。”

句子 S1 中，“教授”是一个表示职称的名词，而句子 S2 中“教授”是一个动词。对于人来说，这样的词性歧义现象比较容易排除，但是对于没有先验知识的机器来说是比较困难的。

目前的词性标注问题解决得比较好，在分词正确的情况下，准确率已经达到了 97%左右，能够满足大部分应用的需求。但由于词性标注提供的是一种通用的解决方案，并不面向专门的应用，使得标注结果无法满足某些应用的特殊需求。

对句法分析来说，最重要的词性是动词，动词对表达句子的语义有着重

要的作用，在句中处于核心地位。而且动词使用灵活，语法功能多样，在句子中能承担不同的角色，如以下句子：

对/p 违法/v 行为/n 不/d 敢/v 站/v 出来/v 反对/v

句子中有 5 个动词，分别起着不同的功能：“违法”名动词，修饰“行为”；“敢”属于助动词，修饰谓语动词；“出来”是趋向动词，对“站”进行补充；“站”和“反对”是一般动词，做句子的谓语。

从这个例子可以看出，动词的功能复杂，如果不加以区分，将对句法分析造成很大的影响。本文首先根据动词的语法属性，制订了一个动词细分类体系，将动词分为 8 个子类。然后在通用词性标注的基础上，使用最大熵方法对动词进行子类标注，之后再进行依存句法分析。实验表明，动词的子类标注提高了句法分析的性能。

2.2 基于词类的分词概率模型

2.2.1 模型的理论推导

设 S 是一个中文句子，也即一个汉字串。对于所有可能的分词序列 W，我们的目的是选取使得条件概率 $P(W|S)$ 最大的分词序列 W^*，因此，$W^* = \underset{W}{\operatorname{argmax}} P(W|S)$。根据贝叶斯公式，得出下面的等价公式：

$$W^* = \underset{W}{\operatorname{argmax}} P(W)P(S|W) \tag{2-1}$$

把中文的词按照一定的方法分成若干类，它们是多对一的关系，即一个中文词唯一的映射到一个确定的词类中去。因此，可以把分词序列 W 唯一的映射到一个词类的序列 C 上[105]。因此，公式(2-1)可以转换为：

$$C^* = \underset{C}{\operatorname{argmax}} P(C)P(S|C) \tag{2-2}$$

公式(2-2)是基于词类的分词概率模型的基本形式，它是信源信道模型在分词中的应用。这个模型假定一个中文句子 S 是这样被生成的：首先，以概率 $P(C)$ 选择一个概念的序列（即我们所说的词类的序列 C）；然后，用一个汉字序列把每一个概念表达出来，这个转换过程的概率是 $P(S|C)$。

这个模型可以用另外一种方式表述如下：$P(C)$ 是一个概率模型，它用来估计词类序列 C 的概率。例如，人名出现在头衔词（如：教授）前面的概率

更大。因此 $P(C)$ 通常被称作上下文模型（context model）。$P(S|C)$ 用来估计在给定一个词类 C 的情况下字串 S 出现的概率。例如，与字串“里小龙”相比，字串“李小龙”更有可能是一个人名，因为“李”是一个常见的中文姓氏，而“里”不是。因此 $P(S|C)$ 也被称作词类模型（word class model）。在分词概率模型中，它实际上是由一个上下文模型和一系列的词类模型组成的，每一个词类对应一个词类模型。

尽管公式(2-2)表明词类模型和上下文模型可以通过简单的相乘关系结合起来，但是在实际应用中，经常需要给它们加上一些权重。因为不同的词类模型是用不同的方法构建的，因此不同的词类模型的概率变化范围可能差别很大。一种解决办法是为不同的词类模型加上不同的权重，因此 $P(S|C)$ 就替换成 $P(S|C)^{\lambda}$，其中 λ 为词类模型系数，λ 可以通过在开发数据集上使分词性能达到最优而确定。

在基于词类的分词概率模型的框架下，分词的过程包括两个步骤：首先，在给定的输入字串中找出所有的分词候选词，并保存在一个切分词图中，每一个候选词都被标注了词类以及词类模型的概率 $P(S'|C)$，其中 S' 是 S 的任意字串。然后，根据公式(2-2)在切分词图中找出一条最优的分词路径（即求出 C^*）。

2.2.2 词类的定义

关于中文词的定义并没有统一的标准，语言学家从不同的角度对词进行定义，而这些定义往往彼此并不完全兼容。本文不去追究分词的语言学定义，而是更关注于切分出的词单元是否能更好的服务于上层的信息处理。然而，不同应用对于分词规范的要求是不尽相同的。例如，键盘输入系统，为了提高输入速度，一些共现频率比较高的字也经常作为输入单位，如“这是”、“每一”、“再不”等等。而检索系统则更倾向于把分词切分成粒度较小的单元，比如，把“并行计算机”切成“并行/计算机”而不是作为一个词，这样无论用“并行计算机”还是“计算机”都能够检索到。本文的分词主要面向句法分析，对词语的切分没有特殊的要求，主要是根据训练语料的标注情况，而不对分词规范做专门的规定。

本文把词分为以下几种类型：（1）词典词，（2）数词，（3）时间词，（4）中国人名，（5）外国人名，（6）地名。这里划分的几种类型并不是从语言学的角度，而完全是从分词的角度考虑。例如，“鲁迅”这个词在词典中出现，则它就不能归为中国人名那一类，其他类别同理。设词典词的个数

是 N，则词类的总数是 N+5，每一个词类对应一个词类模型，不同词类模型的词类概率计算方法也不尽相同，如表 2-1所示。

表2-1 词类及词类模型

Table 2-1 Word class and the model of word class

词类	词类模型
词典词（LEX）	$P(S \mid LEX)$
数词（NUM）	$P(S \mid NUM)$
时间词（TIME）	$P(S \mid TIME)$
中国人名（PER）	$P(S \mid PER)$
外国人名（TRANS）	$P(S \mid TRANS)$
地名（LOC）	$P(S \mid LOC)$

2.3 基于角色标注的未登录词识别

2.3.1 角色的定义

给定一个句子：

李岚清副总理在达沃斯经济讨论会上发表讲话。

在未登录词识别之前，这个句子被切分为

李 岚 清 副总理 在 达 沃 斯 经济 讨论会 上 发表 讲话 。

这里，“李岚清”和“达沃斯”都没有包含在词表中。可以看出，这个切分序列中不同的单元扮演着不同的角色，各个单元的角色如表 2-2所示。

如果能够正确标注切分序列的角色，就可以识别出其中的未登录词，这样就将未登录词识别问题转化为序列标注问题。这里首先需要解决的问题是如何定义角色，在某种程度上，角色与词性标记比较类似。在未登录词识别中，一个切分单元在不同的上下文环境中可以充当不同的角色。例如，“曾”这个切分单元在切分序列“曾/菲/小姐”中是人名的姓，在“记者/唐/师/曾”中是三字人名尾，在“胡/锦/涛/曾/视察/西柏坡”中是人名的下文。

如果角色标记集太大的话，可能会导致数据稀疏的问题。因此，本文并不定义一个包含所有未登录词类别的角色标记集，而是针对每种未登录词类别分别定义其各自的角色标记集。本文中，针对人名识别和地名识别分别定义了一个角色标记集，并且对不同的角色标记集分别进行参数训练。人名识

别和地名识别的角色标记集见附录 1。

表2-2 切分单元的角色列表

Table 2-2 The roles list of units

切分单元	角色
李	中国人名的姓
岚	三字中国人名中间的字
清	三字中国人名末尾的字
副总理	人名的下文
在	地名的上文
达	地名的首字
沃	地名的中间的字
斯	地名的尾字
经济	地名的下文
讨论会；上；发表；讲话	其他（不与未登录词相邻）

2.3.2 基于隐马尔科夫模型的角色标注

角色标注的训练数据来自于已有的词性标注语料，在词性标注语料中，对于每一类未登录词都有其特定的词性标记，因此可以根据与这些未登录词的位置关系来确定每一个切分单元的角色，自动生成角色语料。

基于隐马尔科夫模型标注需要训练初始概率、转移概率和发射概率。设 $p(w|r)$ 是发射概率，即角色 r 发射词 w 的概率，$p(r_j|r_i)$ 是角色 r_i 到角色 r_j 的转移概率，如果 r_i 为句首，则 $p(r_j|r_i)$ 即为初始概率。这些参数均利用极大似然估计法进行计算：

$$p(w|r) \approx \frac{C(w,r)}{C(r)} \tag{2-2-3}$$

$$p(r_j|r_i) \approx \frac{C(r_i,r_j)}{C(r_i)} \tag{2-2-4}$$

其中 $C(w,r)$ 表示词 w 标记为 r 的次数，$C(r)$ 表示角色 r 出现的次数；$C(r_i,r_j)$ 表示角色 r_i 与角色 r_j 共现的次数，即角色 r_i 在前角色 r_j 在后的次数。

根据转换后的角色语料，对角色的参数进行训练。训练好隐马尔科夫模

型的参数之后，利用 Viterbi 算法对一个切分单元序列进行角色序列的自动标注。设 W 是经过分词之后的切分序列，R 是 W 对应的角色序列，R*是最优的角色序列，即：

$$W = (w_1, w_2, ..., w_m)，\ m>0 \tag{2-5}$$

$$R = (r_1, r_2, ..., r_m)，\ m>0 \tag{2-6}$$

$$R^* = \arg\max_R P(R \mid W) \tag{2-7}$$

根据贝叶斯定理，可以得到

$$P(R \mid W) = P(R)P(W \mid R) / P(W) \tag{2-8}$$

对于一个特定的切分单元序列，$P(W)$ 是常数，因此根据公式(2-7)和(2-8)，可以得出

$$R^* = \arg\max_R P(R)P(W \mid R) \tag{2-9}$$

把 W 看作观察值序列，而把 R 看成状态序列。根据隐马尔科夫假设，可以得到

$$P(R)P(W \mid R) \approx \prod_{i=1}^{m} p(w_i \mid r_i) p(r_i \mid r_{i-1}) \tag{2-10}$$

其中 r_0 代表句子的开始，因此，

$$R^* \approx \arg\max_R \prod_{i=1}^{m} p(w_i \mid r_i) p(r_i \mid r_{i-1}) \tag{2-11}$$

为了计算的方便，通常用概率的负对数来代替原始的概率形式，上式可以改写为：

$$R^* \approx \arg\min_R \sum_{i=1}^{m} [-\log p(w_i \mid r_i) - \log p(r_i \mid r_{i-1})] \tag{2-12}$$

计算出隐马尔科夫模型的参数之后，可以利用 Viterbi 算法来求解上述方程，取得最优的角色序列。以“本报记者段心强报道” 这句话为例来说明 Viterbi 算法的运行过程。经过基本的分词之后，这句话会被切分为“本/报/记者/段/心/强/报道”。Viterbi 算法的运行过程见图 2-1。

为了简单起见，并没有画出所有的边，实际上相邻两列的每两个节点之间都存在一条边，如图中第二列和第三列之间所示。节点中的数字代表节点

中的角色发射其对应的切分单元的发射概率的负对数，以第一列第一个节点为例，代表-logP(本|PER_SURNAME)。边上的数字代表两个节点所代表状态之间的转移概率。

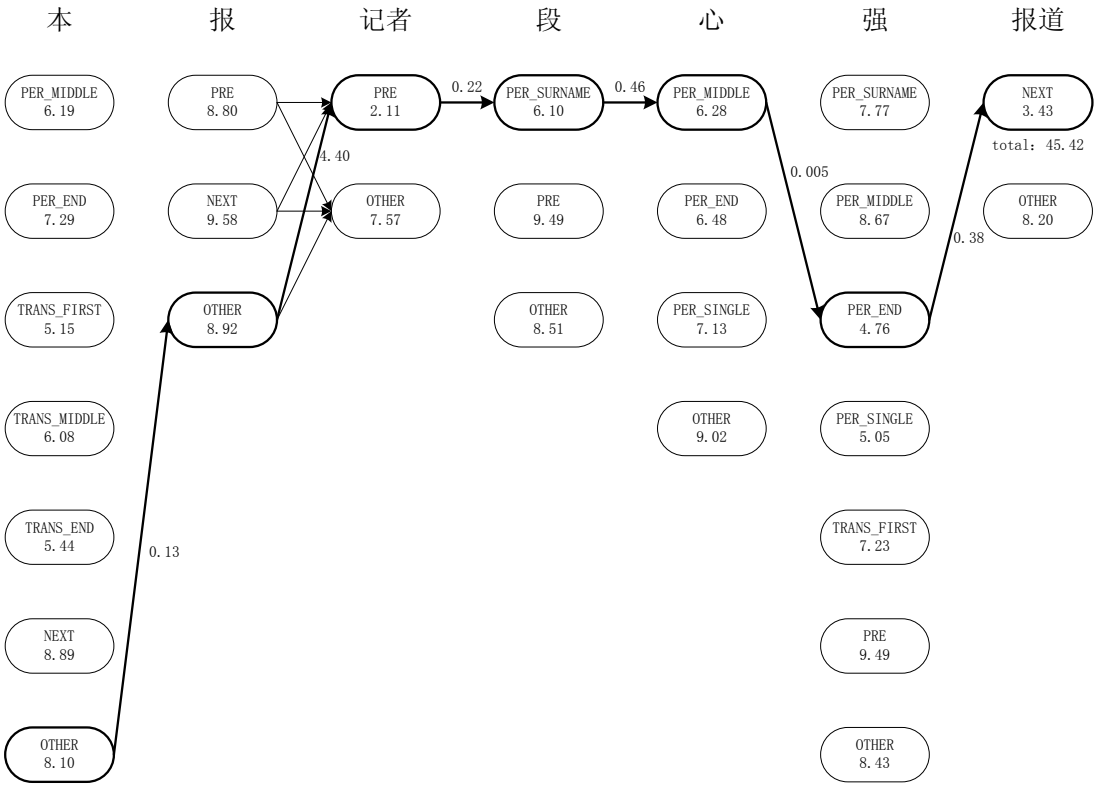


图2-1 利用 Viterbi 算法选择最优的角色序列

Figure 2-1 Searching the best path in the viterbi algorithm

Viterbi 算法运行的过程是这样的：首先把第一列每个节点的角色的初始概率与节点的发射概率（都是取过负对数的）相加并把结果保存在节点中；然后从第二列开始，每一列的每个节点都与其前一列的每个节点进行一次计算，把前一列的节点中所保存的值与两个节点之间的边上的转移概率值相加并保存到本节点中（称之为累计概率），同时保存前一个节点的下标，如果在计算过程中遇到更小的累计值，则替换以前的值。这个过程一直进行下去，到最后一列的每个节点都计算完毕。然后在最后一列的所有节点中找到一个具有最小累计概率值的节点，根据其保存的前一个节点的信息向前回溯，就可以找到最优的角色序列。如图中的粗体线所代表的就是通过从最后一个节点向前回溯得到的最优的角色序列。

至此，已经实现对一个切分单元序列标注出其最优的角色序列。根据这个角色序列，可以用简单的模版匹配的方法识别出其中的未登录词。如上的例中可以根据最优的角色序列“本/OTHER 报/OTHER 记者/PRE 段/PER_SURNAME 心/PER_MIDDLE 强/PER_END 报道/NEXT”得出“段心强”是一个人名。

为了能使未登录词加入到切分词图中并与其他的普通词进行统一竞争，必须要有一个统一的概率模型，基于词类的分词概率模型很好地解决了这个问题。在前面基于词类的分词概率模型中，对于中国人名、外国人名和地名的词类模型并没有给出具体的定义。下面给出一个经验公式。设 $U(w_i, w_{i+1}, ..., w_{i+k})$ 是识别出的未登录词，$R(r_i, r_{i+1}, ..., r_{i+k})$ 是其对应的角色序列，可以计算未登录词的词类模型概率如下：

$$P(U \mid R) = \prod_{j=0}^{k} p(w_{i+j} \mid r_{i+j}) \times \prod_{j=0}^{k-1} p(r_{i+j+1} \mid r_{i+j}) \tag{2-13}$$

如对于上面的人名“段心强”，可以计算 P(段心强|PER)为：
P(段心强|PER) = p(段|PER_SURNAME)×p(PER_MIDDLE|PER_SURNAME)×p(心|PER_MIDDLE)×p(PER_END |PER_MIDDLE)× p(强|PER_END)。

2.4 基于最大熵的动词细分类

对于词性标注来说，不同的标记集对标注的准确率影响很大，因此确定一个合理的词性标记集至关重要。如果词性标记集过大，则会直接影响人工标注语料库的难度和词性标注的准确率；如果词性标记集过小，则达不到词性标注的目的，对上层的信息处理不能提供足够的语法信息。

自然语言处理中，对于多少个词性最为合适的问题还存在争论。从早期 Brown 语料库的 87 个词性标记到 London-Lund 英语口语语料库的 197 个标记[106]，研究者趋向于使用数量较大的标注集。在英文树库 Penn Treebank 中，Marcus 等人通过删除一些词汇的冗余信息，把词性集减少到 36 个标记[15]，这个标记集已经基本被英文的句法分析所接受。

汉语的词性标记集目前还很不统一，数量的差别也比较大。我们倾向于标记集的大小保持一个较小的规模，然后，不同的任务可以根据自己的特点对某些特定的词性进行细分。这样既可以保证小规模标记集具有良好的兼容性，又可以满足不同任务对词性数量的不同需求。例如，除了一些专有名词，

命名实体识别并不需要过多的词性标记。而当训练数据较充分时，句法分析则希望获得更为详细的词性标记信息。

在 2003 年国家 863 分词与词性标注一体化评测中，颁布了一个含有 28 个词性的词性标记集。该标记集涵盖了词语的全部语法属性，能够满足自然语言处理中的大部分应用，并且能够从当前的绝大部分词性标记集转换获得，具有很好的兼容性。由于我们的词法分析系统需要面向很多应用，如信息检索、文本分类等。为了使词性标注模块更具有通用性，本文的依存句法分析采用了 863 词性标记集（见附录 2）。

但由于本文的训练数据使用的是人民日报语料，而该语料采用的是北大词性标记集[107]，共含有 40 多个词性标记。所以需要对人民日报中的词性标记按照 863 词性标记集的标准进行了转换，具体的转换关系见附录 3。

在通用词性标注上，本文采用的是基于隐马尔科夫模型方法，具体的过程与上一节中的角色标注过程相似，只不过是把角色标记替换成词性标记。词性标注的输入是分词后的结果，输出是标注了词性标记的分词序列，如对于输入的分词序列：

迈向 充满 希望 的 新 世纪

经过词性标注之后的结果为：

迈向/v 充满/v 希望/n 的/u 新/a 世纪/n

2.4.1 动词细分类体系

语言建模中的一个难点是如何恰当地选择语言知识的粒度。建立句法分析模型时通常使用两类信息：词性信息和词汇信息。随着 Penn Treebank 等大规模标注语料库的出现[15]，词汇化方法被广泛应用于句法分析中并取得了很好的效果[39, 40]。但在引入词汇信息的同时，数据稀疏的问题也随之而来，尤其是对一些还不具备充足的训练数据的句法分析系统，问题显得更为突出。考虑到词性和词的一些固有缺陷，人们提出了一些具有不同粒度的其它信息。其中包括是词类信息，该信息通常用来改进统计语言模型的性能[108]。再比如动词的子类框架结构[109]，该信息也被用于一些句法分析中。但这两类信息的获取本身就是一个比较困难的工作，本文尝试了一种简单实用的方法，即对词性进行细分类，将动词进一步分为多个子类，以提高句法分析的性能。

一般而言，词性根据语法功能划分为名词、动词等，这种分类可以满足大多数自然语言处理的应用需求。然而，在应用于句法分析时，多数的词性标记集显得粒度过粗，对语法结构的区分度不够。在目前词汇信息不能充分

获取的情况下，我们通过对词性进行细分类来解决这个问题。

句法分析中最为棘手的问题就是语法结构的歧义问题，而动词在语法属性上是差别最大的一类词，因此动词更易于引起歧义结构。在大多数词性标注集中，动词都被分成多个子类。Beale 在开发词典时，根据功能原则将 2500 个动词分为 11 类[110]。在 Penn Treebank 中，动词则根据词尾的变形信息被划分为 6 类[15]。

和英文不同，汉语词没有形态变化信息，动词无论是作为名词还是作为副词均不会有任何形态上的变化。所以汉语动词通常按照语法功能进行分类，标记集中动词的数量也往往较小。例如，Penn 中文树库中每个动词只包含 3 个子类[21]，人民日报语料所使用的北大词性标记集中，动词被分为 4 个子类[107]。

本文的句法分析采用中国国家 863 评测用的词性标记集，该标记集中的动词只有一类。为了更准确地描述动词的属性，本文对其中的动词进行细分，根据语法功能划分为 8 个子类，表 2-3中列出了各个类别的具体信息。

表2-3 动词子类列表

Table 2-3 The scheme of verb subclasses

verb	Description	Examples
vx	系动词	他 ***是*** 对 的
vz	助动词	你 ***应该*** 努力 工作
vf	形式动词	他 要求 ***予以*** 澄清
vq	趋向动词	他 认识 ***到*** 困难
vb	补语动词	他 看 ***完*** 了 电影
vg	一般动词	他 ***喜欢*** 踢 足球
vn	名动词	参加 我们 的 ***讨论***
vd	副动词	产量 ***持续*** 增长

经过细分之后，每个动词子类都保持单一的语法功能，例如，助动词不能充当句子的谓语，而名动词在句子中只能充当名词所具备的功能。

对词性的进一步划分显然将有助于减少句法分析中的歧义结构。词性粒度对句法分析的积极影响已经在很多工作中得到了证明，但这些工作大多数是标注词汇的父节点标记，或者对一些常用词语进行词性划分。Eisner 等人

在英文句法分析中利用了词性标记的子类信息，取得了一些较好的效果[31, 75]。对中文，Xue 等人根据谓词-论元结构将动词派生出 40 个子类用于语义角色标注[111]。但这些工作只需事先对词性进行确定性的分类，无须在后续工作中进行子类的判别。而本文对词性的划分采用的是软分类的原则，即一个词性标记在不同的上下文环境中可能属于不同的子类。

按照表 2-3的分类原则，本文建立了一个动词词表，该表共包含 12205 个动词，其中的 6703 个动词具有一个以上的动词子类。这些兼类动词多数为常用的动词，增加了对动词子类进行自动识别的难度。本文首先对依存树库中的动词进行了人工标注，建立了一个带有汉语动词子类标记的训练集，然后分别使用基于改进隐马尔科夫模型和基于最大熵模型的方法对动词进行子类自动标注。

2.4.2 基于改进隐马尔科夫模型的动词细分类

动词细分类的问题与词性标注很类似，隐马尔科夫模型是词性标注的经典方法，本文首先利用隐马尔科夫模型来进行动词细分类。但是动词细分类问题本身又有其特殊性，只是简单地照搬隐马尔科夫模型，效果并不理想。因此，本文提出了一种改进的隐马尔科夫模型。

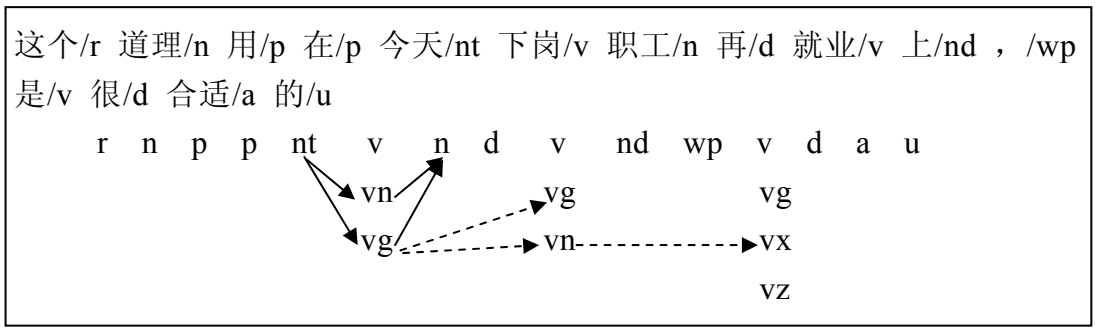


图2-2 使用改进隐马尔科夫模型标注动词子类（实线为发射概率，虚线为转移概率）

Figure 2-2 Verb subclasses tagging based on the improved HMM (Solid lines represent emit probabilities and dashed lines transfer probabilities)

在词性标注中，观察值是待标注的分词结果序列，状态是词性标记。而在动词细分类中，由于其输入是经过词性标注之后的结果序列，动词细分类只需关注其中的动词，而不需要改动其它词性，因此对于动词细分类来说，其观察值是句子中的动词，而状态是动词的子类标记。如果只用传统的隐马尔科夫模型，那么需要计算两个动词之间的转移概率，而两个动词可能并不

是在位置上相邻，这样并没有考虑动词附近的其它词性对动词细分类的影响，而事实上动词附近的其它词性对动词细分类是有影响的。如图 2-2中，分词词性标记序列中，“就业”这个词由于其左侧的词性是 d（副词），右侧的词性是 nd（方位名词），因此“就业”在这里更可能是 vn（名动词），而不太可能是其它的动词类别。

为了更好地利用动词附近的其它词性信息，本文提出了一种带有局部转移概率信息的隐马尔科夫模型，如图 2-2所示。即在 Viterbi 算法中利用动态规划算法计算序列的概率时，除了计算发射概率和动词类别之间的转移概率之外，同时还要计算动词类别与其左侧和右侧词性类别之间的转移概率。其形式化表述如下：

设 V 是句子中的动词序列，T 是这个动词序列对应的动词类别序列。

$$V=(v_1,v_2,...,v_m)，\ \mathrm{m}>0 \tag{2-14}$$

$$T=(t_1,t_2,...,t_m)，\ \mathrm{m}>0 \tag{2-15}$$

按照传统的隐马尔科夫模型，能够推导出

$$T^* \approx \arg\max_T \prod_{i=1}^{m} p(v_i \mid t_i)p(t_i \mid t_{i-1}) \tag{2-16}$$

再加上动词左右两侧的局部转移概率信息，能够得到

$$T^* \approx \arg\max_T \prod_{i=1}^{m} p(v_i \mid t_i)p(t_i \mid t_{i-1})p(t_i \mid t_{before(i)})p(t_{after(i)} \mid t_i) \tag{2-17}$$

其中，$t_{before(i)}$ 代表第 i 个动词之前的词性标记，$p(t_i \mid t_{before(i)})$ 为词性标记 $t_{before(i)}$ 与动词类别 t_i 之间的转移概率，如果第 i 个动词之前为动词或者是句首时，则定义 $p(t_i \mid t_{before(i)})$ 的值为 1；$t_{\mathrm{after}(i)}$ 代表第 i 个动词之后的词性标记，$p(t_{after(i)} \mid t_i)$ 为词性标记 t_i 与动词类别 $t_{\mathrm{after}(i)}$ 之间的转移概率，如果第 i 个动词之后为动词或者是句尾时，则定义 $p(t_{after(i)} \mid t_i)$ 的值为 1。公式(2-17)即为最后得出的带有局部转移概率信息的隐马尔科夫模型。

2.4.3 基于最大熵模型的动词细分类

最大熵模型（Maximum Entropy Model）是一种广泛用于自然语言处理领域的统计学习模型，在机器翻译、词性标注以及句法分析等应用中均取得了很好的效果。最大熵模型能够把各种形式的上下文信息按照统一的原则结合起来，并且不对训练语料强制施加独立假设，在一定的约束条件下可以得到与训练数据一致的概率分布。

最大熵模型通过求解一个有条件约束的最优化问题来得到概率分布的表达式。设已知有 n 个训练样本集$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$，其中 x_i 是由 k 个属性特征构成的样本向量 $x_i=\{ x_{i1}, x_{i2}, \ldots, x_{ik}\}$，$y_i$ 为类别标记。在本文的动词细分类中，x_i 为目标动词的上下文特征，y_i 为动词子类。所要求解的问题是，在给定一个新样本 x 的情况下，判断 y_i 的类别。

随机事件的不确定性可以用条件熵来衡量，其定义如下：

$$H(p) \equiv -\sum_{x,y} \tilde{p}(x) p(y \mid x) \log p(y \mid x) \qquad \text{（2-18a）}$$

其中 p 需要满足下面的条件：

$$P = \{p \mid E_p f_i = E_{\tilde{p}} f_i, 1 \le i \le k\} \qquad \text{(2-18b)}$$

$$\sum_y p(y \mid x) = 1 \qquad \text{(2-18c)}$$

式中，f_i 是定义在样本集上的特征函数，$E_p f_i$ 表示特征 f_i 在模型中的期望值，$E_{\tilde{p}} f_i$ 表示特征 f_i 在训练集上的经验期望值。两种期望的表示形式如下：

$$E_{\tilde{p}} f_i = \sum_{x,y} \tilde{p}(x,y) f_i(x,y) \qquad \text{(2-19)}$$

$$E_p f_i = \sum_{x,y} \tilde{p}(x) p(y \mid x) f_i(x,y) \qquad \text{(2-20)}$$

其中，$\tilde{p}(x, y)$ 是 (x,y) 出现的经验概率且 $\tilde{p}(x,y) = \frac{1}{N} \times Count(x,y)$，$Count(x,y)$ 指 (x,y) 在训练样本集中出现的次数。$p(y \mid x)$ 指 x 出现的情况下，y 的实际概率，$\tilde{p}(x)$ 是 x 出现的经验概率。

对公式(2-18a)、(2-18b)和(2-18c)，根据拉格朗日极值定理，可求出满足条件极值的概率如下：

$$p(y \mid x) = \frac{1}{Z(x)} \exp\left(\sum_{i=1}^{k} \lambda_i f_i(x,y)\right) \qquad \text{（2-21）}$$

$$Z(x) = \sum_y \exp\left(\sum_i \lambda_i f_i(x,y)\right) \qquad \text{（2-22）}$$

在本文的动词细分类中，x 表示目标动词的上下文，y 表示候选的动词子类，$Z(x)$是规一化因子。函数 $f_i(x,y)$代表目候选词的第 i 个特征，λ_i 是它的权

重，k 是模型中的特征数量。本文用表 2-4中的例子来说明特征提取的过程。

表2-4 经不同阶段处理的一个句子实例

Table 2-4 An example of a sentence at different stage

输入句子	武汉取消了 49 个收费项目
分词	武汉 取消 了 49 个 收费 项目
词性标注	ns v u m q v n
动词细分类	ns vg u m q vn n

首先，特征表示为一个二值函数，对表 2-4中的句子，其中的一个特征函数可以表示如下：

$$f_i(x,y)=\begin{cases}1 & if\ y=vg\ and\ its\ next\ tag=u\\ 0 & otherwise\end{cases} \tag{2-23}$$

从表 2-4中能看出动词子类识别问题和传统的词性标注问题有两点不同：首先是在一个句子中并不是所有的词都需要进行判别，其次当对一个词进行动词子类判别时，该词的下一个词的词性可能是已知的。根据第二个区别，将模型的上下文特征定义为：

$$Feature=\{t_{i-2},\ t_{i-1},\ word_i,\ t_{i+1},\ t_{i+2}\} \tag{2-24}$$

表 2-4的句子有两个动词需要被判别，它们的上下文特征分别为：

动词“取消”的上下文特征：

t_{i-2} = *begin*

t_{i-1} = ns

$word_i$ = 取消

t_{i+1} = u

t_{i+2} = m

动词“收费”的上下文特征：

t_{i-2} = m

t_{i-1} = q

$word_i$ = 收费

t_{i+1} = n

t_{i+2} = *end*

此处，“*begin*”表示句子的开始位置，“*end*”表示句子的结束位置。同时，目标动词的子类除了受上下文的影响以外，本文还设定了一个强制约束，即动词的子类必须是动词词表中该动词的一个候选子类。

最大熵模型中的每个特征 f_i 对应于一个权重 λ_i，该权重即为模型的参数，它反映了该特征的重要程度。本文使用 GIS 算法对该参数进行估计，该算法可参阅 Darroch 等人的文献[112]。

2.4.4 动词细分类对比实验及其对句法分析的影响

本实验室已经发布了一个包含 1 万个句子的汉语依存树库，该树库标注了分词、词性和句法信息[1]。为了获得动词细分类的训练数据，在依存树库的基础上，增加了动词的子类标注。全部数据被划分为三个部分进行动词细分类和句法分析的实验，第 1-8000 句作为训练集，第 8001-9000 句作为开发集，其余 1000 句作为测试集，测试集中句子的平均长度是 21.4 个词。

第一组实验测试基于改进隐马尔科夫模型和基于最大熵方法的动词细分类的性能的对比。本文以选择最大概率动词子类的方法作为 baseline，即不考虑动词的上下文信息，只是根据该动词的子类在训练数据中出现的先验概率进行选择，然后分别使用基于改进隐马尔科夫模型的方法和基于最大熵模型的方法进行动词的子类标注。实验一的结果参见表 2-5。

表2-5 动词细分类对比实验结果

Table 2-5 The experimental results on the verb subdivision

动词细分类模型	Baseline	改进隐马尔科夫模型	最大熵模型
准确率	0.753	0.877	0.883

从实验结果中能够看出，同 Baseline 相比，本文两种方法的动词细分类准确率均有较大提高。基于改进隐马尔科夫模型的方法利用了目标动词两侧的相邻词性信息，以及动词类别之间的转移概率信息，这两种信息对于动词子类的判别具有很好的区分作用，取得了很好分类效果，达到了 87.7%的准确率。

而最大熵方法具有判别模型的优点，能够方便的融入各种特征，充分地利用上下文的信息。基于最大熵模型的方法增加了动词子类识别的效果，获

[1] 本树库可按照信息检索研究室主页提供的程序免费获取: http://ir.hit.edu.cn.

得了比 Baseline 提高 13%的识别准确率。

表2-6 三组实验输入句子的形式

Table 2-6 The format of input sentences in three experiments

E1	武汉/ns 取消/v 了/u 49/m 个/q 收费/v 项目/n
E2	武汉/ns 取消/vg 了/u 49/m 个/q 收费/vg 项目/n
E3	武汉/ns 取消/vg 了/u 49/m 个/q 收费/vn 项目/n

表2-7 未标注动词子类的依存分析结果(实验 1)

Table 2-7 The parsing results no verb subclasses (E1)

句子长度	≤10	≤15	≤20	≤30	≤40	≤50
依存关系	0.687	0.649	0.623	0.604	0.592	0.587
依存搭配	0.790	0.740	0.708	0.695	0.686	0.682
核心词	0.836	0.767	0.708	0.688	0.675	0.673

表2-8 自动标注动词子类的依存分析结果（实验 2）

Table 2-8 The parsing results after verb subdividing automatically (E2)

句子长度	≤10	≤15	≤20	≤30	≤40	≤50
依存关系	0.7224	0.6806	0.6569	0.6382	0.6246	0.6207
依存搭配	0.8118	0.7637	0.7357	0.7174	0.7056	0.7021
核心词	0.8490	0.7981	0.7612	0.7345	0.7253	0.722

表2-9 手工标注动词子类的依存分析结果（实验 3）

Table 2-9 The parsing results after verb subdividing manually (E3)

句子长度	≤10	≤15	≤20	≤30	≤40	≤50
依存关系	0.7297	0.6824	0.6656	0.64349	0.6317	0.6284
依存搭配	0.8253	0.7701	0.7448	0.7248	0.7152	0.7127
核心词	0.8867	0.8012	0.7693	0.7295	0.7160	0.715

第二组实验是为了评价动词细分类在句法分析中的作用，句法分析器采用的是信息检索研究室研发的依存句法分析器。本文将实验分为三组：在实验 1（E1）中，句法分析的输入为经过原始词性标注集标注的句子；在实验 2（E2）中，输入的句子经过了动词的自动分类，即第一组实验中基于最大

熵方法的输出结果；在实验 3（E3）中，输入的是经过手工标注动词子类的句子，如表 2-6所示。三组实验结果如表 2-7至表 2-9所示。

图 2-3说明三个组输入对句法分析的影响。图中显示句法分析的准确率随句子最大长度变化的曲线，三条曲线分别表示三组实验结果。

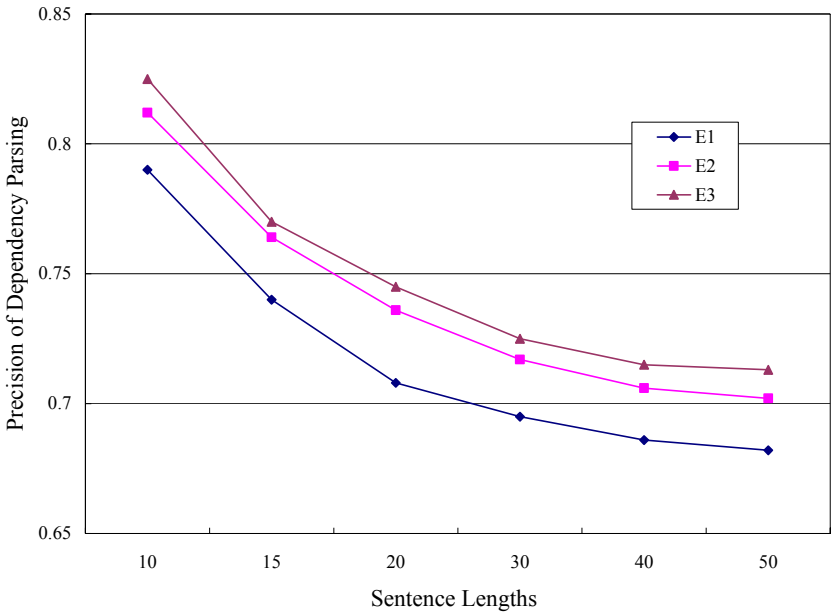


图2-3 三个句法分析实验结果的曲线图

Figure 2-3 The chart of parsing results on three experiments

从图 2-3中可以看出，由于动词子类的引入，E2 和 E3 中句法分析的准确率分别比 E1 提高了 2%和 3%。这说明句法结构对动词的子类信息比较敏感，正确识别的动词子类能够显著提高句法分析的性能。

2.5 本章小结

本章内容为汉语的词法分析，词法分析是自然语言处理的基础技术，同时，词法分析的输出也是句法分析的输入，对句法分析有着重要的影响。汉语的词法分析包括两部分：分词和词性标注。针对依存句法分析，本章对这两部分进行了针对性的处理。

在分词阶段，需要解决了两个问题：歧义词消解和未登录词识别。本章

使用基于词类的语言模型进行分词，同时应用基于角色标注的方法对未登录词进行识别。在词性标注阶段，本章使用了基于隐马尔科夫模型的词性标注方法。针对句法分析的问题需求，本章提出了一个对汉语动词进行细分类的体系框架，并采用最大熵模型对动词子类进行自动判别。最大熵模型能够有效地融合上下文特征，取得了较好的分类效果。

本章的动词细分类工作主要面向句法分析，并不是要适用于所有的自然语言处理任务，其目标是减少词汇化句法分析中数据稀疏问题，并增加对句子中歧义结构的识别。通过实验，动词细分类模块较好的改善了句法分析的性能。这表明本文的动词子类划分比较合理，也说明了动词子类对句法分析有积极的影响。另外，在评价动词子类对句法分析的作用时，并没有使用与语言的相关知识，因此如果把这种方法应用到其它语言上，相信也会取得类似的效果。

第3章 基于隐马尔科夫树模型的名词复合短语分析

3.1 引言

名词复合短语（Noun Compounds，NC）是一种特定类型的短语，广泛存在于各种语言中。目前，对这类短语的命名并不是很统一，大致表达这一类短语的名称有：

- noun compounds, compound nouns, compounds;
- nominal compounds, compound nominals, complex nominals;
- noun premodifiers;
- nominalisations;
- noun sequences, noun-noun compounds, noun + noun compounds.

英语中对名词复合短语的一个定义为：在功能上充当名词的连续两个或两个以上的词汇序列，其中不包含属格标志和名称[113]。以下是几个英语 noun compounds 的例子：

a. stone fish

b. party animal

c. horse riding

d. leading soprano

e. outback adventure tour

f. emergency bus fuel

g. killer whale attack

从语法角度来说，名词复合短语和词比较相似，很多短语和词的界限不易区分。这些短语对自然语言处理相关的一些应用有很大的影响。机器翻译中，对名词复合短语如果仅依靠逐词对译很难准确地表达源语言的语义，需要根据内部关系将短语作为整体来翻译[114]。在信息抽取中，专业术语通常以名词复合短语的形式出现，分析短语的语法关系有助于准确地获取这些信息[115]。

名词复合短语在英语中的研究较为深入，但随着语言分析技术的逐步深入以及对自然语言处理的实用化需求，名词复合短语已经受到了各国研究者

的普遍重视，其研究范围也更加广泛，目前的研究内容主要包括以下几个方面：

（1）边界识别。识别出名词复合短语在句子中的边界，早期的研究工作中多集中在这方面。

（2）语法分析。分析短语内部的语法关系，既是进一步分析语义的基础，也可直接用于信息抽取等工作之中[115, 116]。

（3)语义分析。确定名词复合短语的修饰语和核心词之间所存在的语义关系，语义关系的分析对机器翻译的质量至关重要[117, 118]。

（4）分类。确定名词复合短语所属的语义类别，可用于信息抽取以及自动问答等工作[119, 120]。

对名词复合短语还开展了一些其他方面的研究工作，如探索名词复合短语的表示和处理，建立大规模的多语词典[121]；从大规模语料库中自动抽取名词复合短语，用以改善信息检索的效果[122]；以及通过对名词复合短语的预测来提高文本输入的效率[123]。

名词复合短语有着自己特定的语法结构，而且此类短语的分析效果对汉语句法分析造成了较大的影响。本文的目的是通过分析名词复合短语来改善句法分析的质量，因此，只研究上述四类问题中的前两类：边界识别和语法分析。

这两方面在英文中已有一些相关的研究工作，如边界识别方面，Lauer 应用了邻接模型和依存模型识别短语的边界[113]，Nakov 等人利用搜索引擎进行无指导的短语边界识别[124]。在语法分析方面，Yoon 等人利用在大规模语料中抽取的词汇知识分析短语的语法结构[115]，Buckeridge 等人使用潜在语义索引的方法消除名词复合短语中的语法歧义[116]。汉语方面，目前只有短语分类以及短语自动获取的研究[119, 122]，在边界识别和语法分析方面，还未看到有相应工作。

在名词复合短语的边界识别和语法分析方面，已有工作的做法是将两个问题完全独立，分别进行处理，这导致了一些问题：边界识别时，不能充分利用语法结构信息；而语法分析时，默认边界识别是正确的，掩盖了边界错误所造成的影响。这种对紧密联系的两个问题的割裂，破坏了问题的整体性。对名词复合短语来说，语法的内部结构会随着边界的不同而变化，而边界的确定也会受到语法结构的影响，这两个问题互相影响，有着密切的联系。

根据问题的特点，本文引入了隐马尔可夫树模型，对短语的边界识别和语法分析采用一体化的处理策略。隐马尔可夫树能够把短语的边界信息和语

法结构表示在一棵树上，将两个问题融合为一体，较好地解决了以前方法中存在的问题。同分步处理策略相比，该方法无论在边界识别以及结构分析方面均有较大提高。在对名词复合短语分析效果的评价上，将分析模块应用于句法分析中，明显地改善了依存句法分析的效果。

3.2 名词复合短语

3.2.1 汉语名词复合短语的定义

由于语言的灵活多样，在不同的语言中，名词复合短语有着不同的表述形式。汉语虽有一些相关的研究工作，但这些工作并未给出名词复合短语的完整定义。本文试图从语言功能和表现形式两方面对汉语的名词复合短语进行阐述。

汉语属于意合语言，在名词短语的构成上异常灵活，名词性成分通过简单的合成即可构成较长的序列，不需要助词等连接成分，如图3-1中的(a)、(b)。而且汉语的动词没有变形信息，既可充当名词的功能作短语的核心词，如(c)，又可以动词的成分作为修饰语或修饰语的一部分，如(d)、(e)。

(a). 红外线体温检测仪
(b). 中国当代文学发展综史
(c). 经贸**合作**
(d). **管理**办法
(e). 中国**驻**埃及大使馆

图3-1 汉语名词复合短语的例子（黑体字为动词）

Figure 3-1 The examples of Chinese noun compounds (Bold words are verbs)

此类短语我们称之为名词复合短语，它是由体词及谓词成分按顺序构成的词语序列，在语义上代表某一特定的实体或概念，短语的内部结构稳定，不可拆分，在语法功能上等同于该短语的核心词。借鉴赵军给汉语基本名词短语作的定义[125]，这里对名词复合短语形式化地表示为：

名词复合短语 = 限定语 + 核心词

核心词 →名词$_1$|简称|符号串|名动词

限定语 →名词$_2$|简称|符号串|区别词|名动词|一般动词。

其中，名词 $_1$ 包括普通名词和专有名词，名词 $_2$ 在名词 $_1$ 的基础上，增加了时间词和处所词。名动词是在语法功能上相当于名词的动词，简称和符号串均为名词性。

通过上面对名词复合短语的定义和分析，可以总结出汉语的名词复合短语有以下几个特点：

（1）短语由限定词和核心词构成，限定词只由名词（包括专有名词、时间词、处所词等各种名词性成分）、动词、简称、区别词或特殊符号充当，核心词由名词（不包含时间词和处所词）、名动词或简称充当。

（2）短语中没有结构助词和时态助词，没有数词和形容词，使得短语不具有语法上的派生性；如“自然语言处理”，“企业承包合同”。而“第二次中东战争”，“复杂的特征”，“老师写的评语”等短语可通过语法派生为“第三次中东战争”、“简单的特征”、“学生写的评语”等等，均不属于名词复合短语。

（3）短语内部结构稳定，不可拆分，组合之后顺序固定，不能改变，很多词具有专有名词性质。事实上，由一个以上词构成专有名词以及命名实体，多数都属于名词复合短语。比如“线性判别函数”，“哈尔滨工业大学”。

（4）短语各成分的组合不遵照严格的语法，含两个以上的词时，其关系不能完全确定。如“中国当代文学发展综史”，在语法上“中国”修饰“当代”或修饰“文学”，“文学”修饰“发展”或修饰“综史”，对语义都没有影响。

（5）短语的末尾词为整个短语的核心词，在功能上代表该短语，短语通过核心词与外部发生联系，限定成分被屏蔽。如句子：

成功来自[董事总经理梁伯韬]等人对[大陆国企改革形势]的洞察。

其中，“董事总经理梁伯韬”和“大陆国企改革形势”分别属于名词复合短语，如果省略句子的限定成分，如：

成功来自[梁伯韬]等人对[形势]的洞察。

并不改变句子的语义。

3.2.2 名词复合短语与命名实体的关系

无论在语法还是语义的角度，名词复合短语和命名实体都有一定的相似性。命名实体(Named Entity，NE)，是指被命名的唯一确定的最小信息单位，主要包括人名、地名、机构名、专有名词等。名词复合短语是对具体和抽象

事务的表述，既可以是特指，也可以是泛指。

首先，可以将复合名词短语看作命名实体的扩充，即对实体对象的一种表述方式上的一种扩充。例如：“董事/n 总经理/n 梁伯韬/nh”，在命名实体识别中“梁伯韬/nh”被识别为人名这个实体。“唯一确定”是命名实体的一个根本特点，然而实际的重名问题为人名引入了歧义性，因为“最小”使NE缺省了一些具体信息。而名词复合短语的识别会将“董事/n 总经理/n 梁伯韬/nh”均识别为一个名词复合短语，而且此类短语中包含了这个被命名实体的一些更加具体的信息，如头衔“董事/n 总经理/n”。同时也使它与“主任/n 医师/n 梁伯韬/nh”简单的区分，真正体现其“唯一确定”性。

此外，像命名实体中机构名和较长的地名，完全符合名词复合短语的定义。例如：“哈尔滨工业大学”，既属于命名实体，又属于名词复合短语。

可以看出，命名实体和名词复合短语具有紧密的联系，如果能够识别出名词复合短语，并能够根据核心词判断出短语的类别，则可以从中获得命名实体的信息，将两个问题融合到一起。

3.2.3 名词复合短语与句法分析关系

名词复合短语对句法分析的影响很大，图3-2的依存分析结果中，“铁路地方建设附加费”属于一个名词复合短语，通过核心词“附加费”同外部发生句法关系。如果不能将短语正确地识别，句法分析时容易发生以下两类错误：

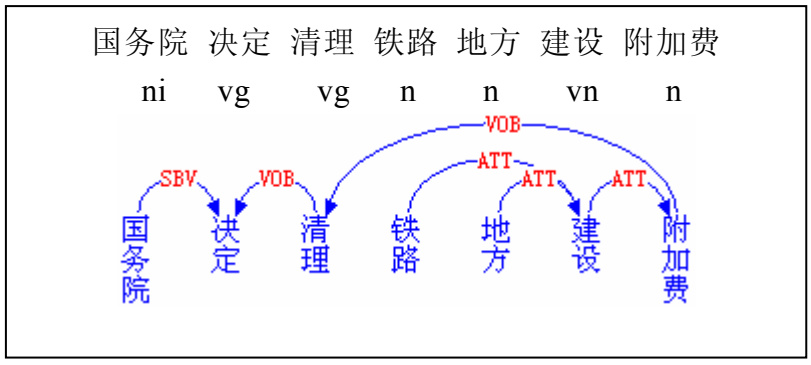


图3-2 依存分析结果的树形结构

Figure 3-2 A dependency tree of parsing result

（1）内部错误

名词复合短语内部的修饰关系同语言的其他部分有些差别，多数为定中修饰关系。如果和句子的其他部分统一分析，短语内部容易发生错误，比如

将动词“建设”和左面的词分析为主谓关系，和右边的词分析为动宾关系。

（2）外部错误

短语内部的一些词汇和外部词汇可能属于常用的搭配，如“清理”和“铁路”或“地方”的依存概率均大于“附加费”。这时，短语容易被分割，引起句法结构的交叉错误，这种错误对分析结果产生的影响比内部错误要大。

句法分析中，因名词复合短语导致的错误占有较大比例。在我们的依存句法分析结果中，名词复合短语的内部错误就占到依存弧总数的占3%左右。为此，本文将名词复合短语的处理从句法分析中独立出来，先对其进行边界识别和内部结构的分析，再以短语的核心词参与整句的句法分析。英语的句法分析中，Collins曾应用类似的过程先识别出句子中的基本名词短语，然后再进行句法分析，取得了较好的效果[37]。

将名词复合短语从句法分析中独立出来有两个优点，一是名词复合短语具有自己的语法特点，其构成规律同句子的其他部分不一致，如将这类短语和整个句子一起分析，容易产生噪声，干扰句子的分析。另外，核心词能够代表名词复合短语的语法功能，只用短语核心词参与句子的分析，可以避免短语的修饰成分对句法分析的影响，并降低句法分析的复杂度。

3.3 名词复合短语分析

3.3.1 隐马尔科夫树模型

隐马尔可夫树（Hidden Markov Tree, HMT）模型是信号处理领域提出的一种方法[126], 该方法曾被用于 Web 站点的分类工作[127], 取得了很好的效果。同隐马尔可夫链类似，HMT 也有观察值和隐含状态，不同之处是隐马尔可夫树的观察值是树状结构，而不是链状结构。用 $\boldsymbol{w} = (w_1, w_2, \ldots, w_n)$ 代表 HMT 的观察值，其中，w_1 是树的根节点。$\boldsymbol{s} = (s_1, s_2, \ldots, s_n)$代表对应于 $\boldsymbol{w}$ 的隐含状态，根节点为 s_1，隐含状态所对应的状态集合为$\{1, 2, \ldots, K\}$。图 3-3为一棵隐马尔可夫树，其中，第 u 个节点的父节点序号为 $\rho(u)$，结点 u'的子节点序号为 c_1，c_2。

HMT 的层间状态转移满足隐马尔可夫树特性，即当前节点状态只与其父节点和子节点状态有关：

$$p(s_u \mid \{s_{u'} \mid u' \neq u\}) = p(s_u \mid s_{\rho(u)}, s_{c_1}^u, \ldots\ldots, s_{c_{n_u}}^u)$$

隐马尔可夫树模型由以下三个参数描述：

（1）初始分布概率：$\pi=(\pi_k)_{k\in\{1.....K\}}$

（2）状态转移概率：$A=(a^{rm}_{\rho(u),u})$，$a^{rm}_{\rho(u),u}=p(s_u=m\,|\,s_{\rho(u)}=r)$

（3）状态发射概率：$B=(b_j(k))_{k\in\{1.....K\}}$，随问题的不同，$b_j(k)$ 有不同的计算形式。

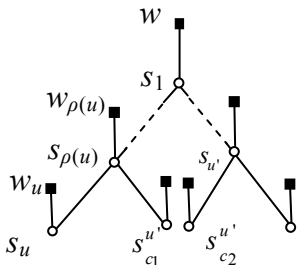


图3-3 隐马尔可夫树结构（白点是隐含状态，黑点是观察值）

Figure 3-3 The structure of a hidden markov tree (white dots are hidden states and black are observed values)

3.3.2 基于隐马尔可夫树的一体化分析

可以用两种策略完成本文的任务，第一种是分步策略，即先识别出短语的边界，然后在已知边界的基础上，对短语的语法结构进行分析；第二种是一体化分析，即在分析过程中，同时确定将短语的边界以及内部语法结构。第一种策略作为 Baseline 在本文的实验部分给出其结果，这里介绍第二种策略。

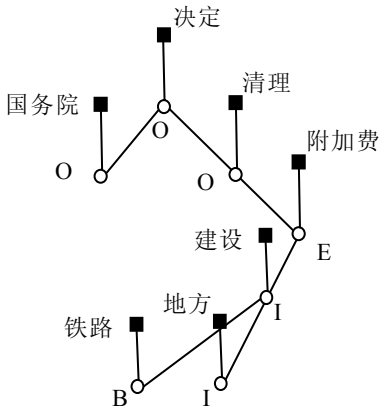


图3-4 依存分析结果所对应的隐马尔可夫树

Figure 3-4 The HMT of a dependency parsing results

图 3-2中，句子的语法结构为一棵依存树，名词复合短语属于依存树的一

部分，仍为树状结构。将依存树的词汇节点作为观察值，每个节点对应一个状态标记，表示观察值的隐含状态。状态集合 S 包括五个标记：$S=\{B，I，E，S，O\}$，分别表示短语的开始、中间、末尾、单个词的短语和短语的外部节点。这样，图 3-2的依存树即同隐马尔可夫树相对应，如图 3-4所示。

欲获得短语的边界和内部结构，只需要确定构成短语的字符串所对应的隐马尔可夫树。该过程分为以下几个步骤：

（1）选定候选字符串

按照定义中的形式化描述，选择符合要求的词性序列作为候选串。

（2）分析候选串中的 HMT

从句子中选定候选字符串(w_m, ……, w_n)之后，由后向前依次分析候选串的子串，如图 3-5所示，并从子串 $w_i=(w_{n-i+1}, \ldots, w_n)$ $(1\leqslant i\leqslant n-m+1)$中选择全部的合法依存树集合 $\boldsymbol{T_i}$，将集合中的每个依存树同隐马尔可夫树对应，获取子串的最佳隐马尔可夫树 t_i^*。

（3）获取短语的分析结果。

从所有子串的最佳隐马尔可夫树中，选出概率值最大的树，其对应的节点状态即为短语的边界，树内部的结构即为短语的语法结构。

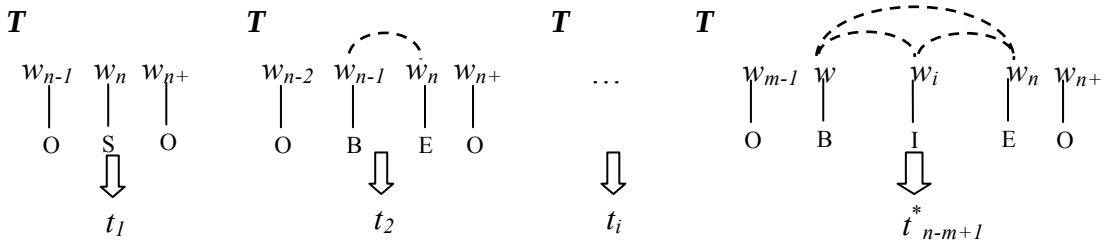


图3-5 从候选字符串中分析隐马尔可夫树（虚线部分为子串生成依存树的过程）

Figure 3-5 Parsing HMT from the candidate strings (building the dependency tree is represented by dashed lines)

3.3.3 隐马尔可夫树模型参数的计算

HMT 的概率可以通过隐马尔可夫树模型的上向－下向算法（upward-downward algorithm）获得：

$$p(w) = p(w_m, \ldots, w_n)$$

$$= \sum_{k=1}^{K} \beta_u(k)\ \alpha_u(k), \quad u \in \{m,\cdots,n\}$$

其中，$\beta_u(k)=P(T_u \mid S_u=k)$ 为上向变量，表示以节点 u 为父节点，其状态

为 k 时子树 T_u 的概率。$\alpha_u(k)=P(S_u=k,T_{1\backslash u})$ 为下向变量，$T_{1\backslash u}$ 表示子树 T_u 以外部分的概率值。两个变量可利用模型参数通过迭代的方式计算获得，具体过程可参考 Crouse 等人的文章[126]，这里介绍本文模型参数的计算。

（1）初始概率定义为短语的边界概率，左边界概率 $p(B)=p(w_m\mid w_{m-1})$，右边界概率 $p(E)=p(w_n\mid w_{n+1})$，初始概率 $\pi=p(B)\times p(E)$。

（2）转移概率包括状态的转移概率和观察值的转移概率，状态转移概率 $a_s=p(s_u\mid s_{\rho(u)})$，观察值转移概率 $a_w=p(w_u\mid w_{\rho(u)})$，转移概率 $a_{\rho(u),u}=a_s\times a_w$。

（3）发射概率定义为由隐含状态生成相应观察值的概率：$b_j(k)=p(w_j\mid s_j=k)$。

在计算 HMT 模型的参数时，由于训练集规模的限制以及短语中的名词多为专有名词，使得训练集中的名词非常稀疏。本文用两个方法解决这个问题，一个是使用词性信息对观察值进行平滑，另外是借助名词的词义信息。

由前一节的分析，短语内部的修饰关系主要是语义一级的，因此，利用词义信息将有助于分析短语内部的关系。本文使用的词义来自于《哈工大信息检索研究室同义词词林扩展版》[1]，该词典共收录 8 万余词，在词语分类上沿袭的《同义词词林》[128]的三层体系，并在此基础上对词语继续细分类，增加两层，得到最终的五层分类体系，如图 3-6所示。

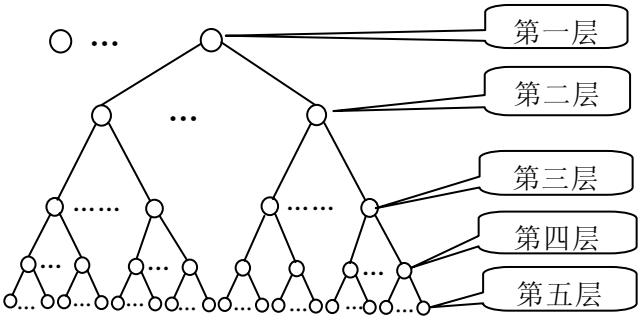


图3-6 词语组织结构

Figure 3-6 The organizational structure of words

本文整理了其中的名词语义类，共 41451 个名词。在使用词义时，选择了第三层语义类，将同类名词归在一起，赋予一个语义代码，例如：

[1] 该语义词典及本文实验用的树库均已免费发布，可通过网址http://ir.hit.edu.cn获得。

Bf03 露 露水 露珠 寒露 霜 严霜 冷霜 早霜 晚霜 柿霜 终霜 霜花 白霜

这里选择词林的第三层，是因为该层的名词语义类数量适中，共有 518 个类别，类别数量介于词性数量和词汇数量之间，既可以缓解词汇数据的稀疏问题，还具有较好的覆盖性，较为准确地描述了短语内部的语义关系。然后取每类的语义代码，作为隐马尔可夫树的观察值。

3.3.4 基于错误驱动的边界修正

错误驱动（Error-driven）的方法也叫做基于转换（Transformation-Based）的学习方法，由 Brill 提出并在自然语言处理中广泛使用[129]。其基本思想是从训练集中自动学习规则，然后利用规则对初始结果进行转换。该方法既避免了人工制定规则时效率低下的缺点，又具有灵活有效的特点。常结合其他方法作为后处理模块改善实验结果。

本文利用错误驱动的方法修正部分短语的边界错误。规则模版由左边界、短语本身以及右边界组成，用词性（POS）表示，形式如下：

(1) $POS = x \ \& \ compound \ \& \ POS = y \rightarrow 1$

(2) $POS = x \ \& \ compound \ \& \ POS = y \rightarrow 0$

其中，（1）为正规则，即短语的候选字串符合规则左部的条件时，判定短语的边界正确；（2）为反规则，即字符串同规则左部匹配时，判定短语的边界错误。

规则获取分为两步，首先从训练集中自动抽取频率较大的初始规则，然后利用开发集对初始规则进行评价，从中选出分值高的规则。对初始规则采用修正率进行评价，即每条规则在开发集中所修正的正确边界数量和引起的错误边界数量的差值，最终选出正规则 11 条，反规则 23 条。

3.3.5 实验结果及分析

实验数据采用《哈工大信息检索研究室汉语依存树库》，该树库共有 1 万个句子，21 万词，句子中标注了依存关系，主要用于训练汉语依存句法分析器。为了本文的研究工作，在句子中进一步对名词复合短语进行了标注。树库中共包含名词复合短语 14249 个，不同长度短语的数量及短语中依存结构所占的比例如表 3-1所示。其中，长度是指名词复合短语中词的数量，依存结构是指名词复合短语经过内部语法分析后依存弧的数量。

取树库中的 8000 句作为训练集，1000 句作为开发集，1000 句作为测试集，并用同样的集合对短语分析和依存分析进行试验。评价指标包括短语的

边界、短语的语法结构以及短语分析对依存分析的影响。

表3-1 名词复合短语的长度和依存结构分布

Table 3-1 The length and dependency structures of NCs in the treebank

Length (Words)	2	3	4	5~
Compounds	0.684	0.203	0.067	0.046
Dependencies	0.456	0.271	0.135	0.138

边界识别的实验中，以最长子串的方法作为 Baseline，即从候选串中选择长度最大的子串作为名词复合短语。以基于隐马尔可夫树模型的一体化分析方法作为第一个模型（Model 1），隐马尔可夫树结合错误驱动的方法作为第二个模型（Model 2）。以准确率（P）、召回率（R）和 F-Score（F）评测实验结果，如表 3-2和图 3-7所示。

表 3-2列出了不同长度短语的分析结果，短语的最小长度为 2 个词，≥2 的部分即为全部分析结果。从表中能够看到，Baseline 的边界识别率较高，说明在汉语中，两个连续的名词性成分多数情况下都构成名词复合短语，而二词短语的语法结构是确定的，隐马尔可夫树的方法对其影响并不大，所以 Model 1 对二词短语的边界识别提高不是很大。但随着长度的增加，连续的名词性成分构成名词复合短语的比例下降，而且短语内部结构变得复杂，隐马尔可夫树充分发挥了作用。尤其对长度为四词及以上的短语，边界识别的准确率提高了 7 个多百分点。

这里的候选字符串未包括一般动词，如果将一般动词也加入候选串一起分析，准确率将受到严重的影响，Baseline 方法中边界识别的 F 值只能达到 51.4%。这主要是因为，一般动词在名词复合短语中所占的比例很小，约为 3%，而将其同候选串一起分析时产生的干扰却十分严重。所以本文的工作暂时未处理一般动词，将其放在以后的工作中做进一步的探索。

在隐马尔可夫树模型的基础上，Model 2 增加了基于转换的错误驱动方法，该方法根据边界词汇的特征，从训练数据中自动学习一些修正规则，有效弥补了统计方法所引起的一些过学习的情况，对二词短语的边界修正发挥了较大的作用，提高了一个多的百分点。

实验中的第二个评测指标是短语内部语法结构的分析，在以最长子串方法识别出短语边界的基础上，将短语和句子一起进行依存分析的结果作为 Baseline。实验结果如表 3-3和图 3-8所示。

表3-2 名词复合短语的边界识别结果（Baseline：最长子串方法；Model 1：隐马尔科夫树模型；Model 2：隐马尔科夫树模型+错误驱动。）

Table 3-2 The results of boundary recognition of NCs (Baseline: the longest substring; Model 1: HMT; Model 2: HMT + error-driven)

Model		Baseline	Model 1	Model 2
2 Words	R	0.843	0.844	0.867
	P	0.922	0.928	0.925
	F	0.881	0.884	0.895
≥2 Words	R	0.814	0.829	0.846
	P	0.894	0.901	0.898
	F	0.852	0.863	0.871
≥3 Words	R	0.752	0.794	0.801
	P	0.835	0.842	0.839
	F	0.791	0.817	0.820
≥4 Words	R	0.651	0.741	0.753
	P	0.767	0.822	0.836
	F	0.704	0.779	0.792

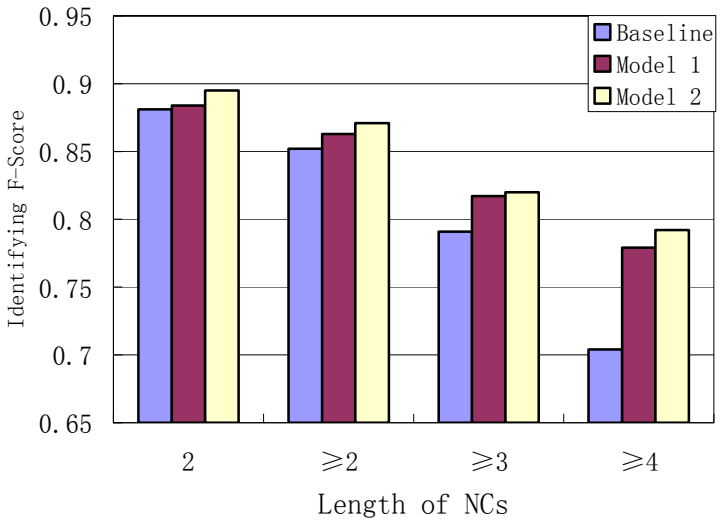


图3-7 短语边界识别结果

Figure 3-7 The results of boundary recognition of NCs

表3-3 名词复合短语的结构分析结果（Baseline：最长子串方法；Model 1：隐马尔科夫树模型；Model 2：隐马尔科夫树模型+错误驱动。）

Table 3-3 The results of structure analysis of NCs (Baseline: the longest substring; Model 1: HMT; Model 2: HMT + error-driven)

Model		Baseline	Model 1	Model 2
2 Words	R	0.846	0.862	0.888
	P	0.925	0.947	0.947
	F	0.884	0.902	0.917
≥2 Words	R	0.723	0.769	0.782
	P	0.809	0.827	0.824
	F	0.764	0.797	0.803
≥3 Words	R	0.623	0.688	0.690
	P	0.712	0.726	0.721
	F	0.665	0.707	0.705
≥4 Words	R	0.583	0.666	0.668
	P	0.708	0.717	0.717
	F	0.639	0.690	0.692

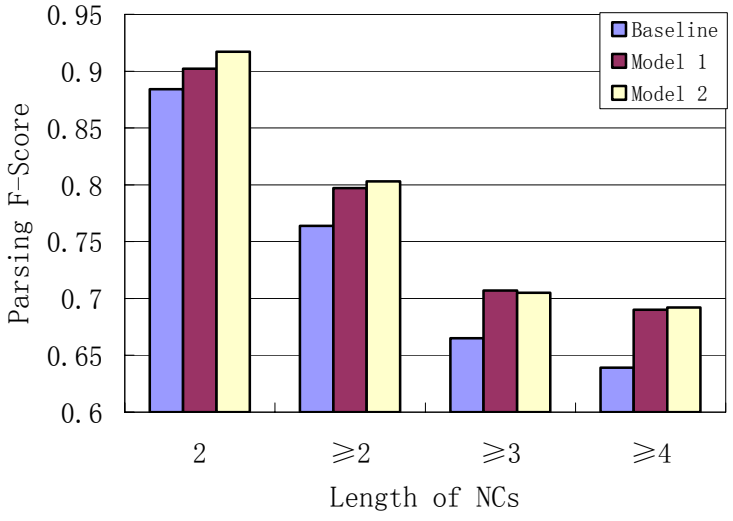


图3-8 短语结构分析结果

Figure 3-8 The results of structure analysis of NCs

从表 3-3能够看到，在总体的 F 值上，隐马尔可夫树模型对短语内部结构的分析，提高得幅度要比边界识别高很多，这是因为隐马尔可夫树主要改善了多词短语的分析效果，而测试集中多词短语的依存结构比例要比短语数量增加很多。由于模型 2 中的转换规则多用于修正二词短语，所以对结构分析的提高并不明显。

作为对名词复合短语分析的检验，本文将该模块嵌入依存分析中，即在依存句法分析之前，先分析短语的边界和结构，然后利用短语的核心节点同句子的其他部分进行依存分析，句法分析的结果如表 3-4所示。其中，第二列表示未使用名词复合短语分析模块依存分析结果，第三列表示使用了名词复合短语分析模块的依存分析结果。

表3-4 NCs 对依存句法分析的影响

Table 3-4 The effect of NCs on the dependency parsing

	无名词复合短语分析	有名词复合短语分析
依存关系	0.677	0.701
依存搭配	0.707	0.734

表 3-4中，第二列是数值没有使用名词复合短语分析模块的依存分析结果，依存搭配和依存关系的准确率分别为 67.7%和 70.7%。第三列是使用了短语分析之后的结果，从结果中可以看出，名词复合短语对句法分析有着较大的影响，分析模块有效地改善了句法分析的性能。

3.4 本章小结

本章对汉语名词复合短语的边界识别和结构分析进行了研究。该工作既可直接用于信息抽取等多个应用领域，又能够作为自然语言处理的基础模块，改善句法分析的结果。

针对名词复合短语的边界识别和语法分析的紧密联系，本章引入了隐马尔可夫树模型的方法，将这两个问题进行一体化处理。由于隐马尔可夫树模型能够把短语的边界信息和语法结构统一在一棵树上，避免了分步处理时所存在的问题，使名词复合短语的边界识别和结构分析均达到了较高的准确率。同时，将模块应用与依存分析中，改善了句法分析的性能。

第4章 面向句法分析的句子片段识别

4.1 引言

一般来说，句子的语法结构比较完整，在语义上也相对比较独立，目前的句法分析系统，多数都是以句子为单位进行分析。存在的问题是，文本中的一些句子长度较大，给句子的分析造成了很多困难。尤其类似《人民日报》这样的语料，其中的文本多数为政治类和经济类的题材，句子的平均长度达到了20多个词。随着长度的增加，句法分析的时空复杂度会急剧上升。更为严重的是，句子的长度越大，产生的语法歧义就越多，严重影响了句法分析的准确率。

句法分析中，缓解句子过于复杂的一个方法是采用分治策略，(Shiuan and Ann)对句子中的连接词进行消歧处理，然后根据这些连接词的功能将句子分为不同的部分，再对每个部分分别进行处理[130]。Braun等使用类似的方法对德语文本进行浅层句法分析，其分治策略识别了更多的成分，如命名实体、动词词组等[131]。(Lyon and Dickerson)提出了另外一种分治策略对句子进行分解，该工作将句子分为pre-subject, subject, predicate三部分，分析之后再将三个部分进行组合[132]。

这些工作有效地简化了复杂句子的分析，但存在的问题是，句子分解之后的各个部分差别较大，需要使用不同的方法对各个部分进行分析，使句法分析过程变得复杂。另外一个问题是，这些分治策略都是语言相关的，难以借鉴到其他语言之中。

针对句子的长度问题，英语中的一个重要工作是从句识别。从句识别不但能减少由于句子过长而造成的负面影响，而且对文语转换、机器翻译等应用也有着重要的作用[133]。CoNLL-01专门组织了评测任务来识别英语的从句，最好结果的准确率达到了78.63% [134]。

但是英语中的这些工作也不能完全借鉴到其他语言之中。比如，汉语是一种意合语言，句子的生成侧重于语义的表达，在语法规范上的限制不是很严格，没有明确的从句概念，这一点同英语有着很大的差异。

本文针对句法分析中长句处理所遇到的问题，提出了一种基于片段的句法分析方法。首先根据句子的语法结构，对句子中的标点进行角色标注。根据标注的结果，将句子切分为不同的片段，每个片段作为句法分析的基本单位。这样，就

减少了句法分析中的句子长度，提高了分析的效率。对每个片段分别进行句法分析之后，使用分类器识别出片段之间的依存关系，然后将各个片段进行合并，完成整个句子的依存分析。

实验中，对基于整句的句法分析和基于片段的句法分析进行了比较，后者在准确率上获得了较大的提高。同多语依存分析器 MSTParser 的比较中，基于片段句法分析器在中文树库上表现了更好的性能。

开展句法分析工作之前，需要有一个人工标注的树库作为训练和测试数据，而在这之前，还没有已标注的用于句法分析的大规模汉语依存树库。为了探索汉语依存句法分析的规律，同时为后续的研究工作奠定基础，本课题制订了一个汉语依存标注体系，建立了一个较大规模的汉语依存树库。本章首先介绍制订的语法体系和依存树库的建设过程，然后介绍面向句法分析的句子片段识别。

4.2 依存树库建设

语料库是统计自然语言处理的基础，基于语料库的方法对自然语言处理领域的各个方向产生了重大的影响。在各类语料资源中，影响最大的要算是标注了语法信息的数据，因为标注后的句子为一个树形结构，所以也称这样的语料为树库。由于标注树库时需要制订复杂的语法规范，需要投入大量的人力，同时树库又是开展句法分析工作所必需的数据资源，因此树库建设就成为句法分析中最为重要的内容之一。英文树库 Penn Treebank 的成功，更增加了研究人员对树库标注的重视，目前，已经建立了很多高质量的树库，用于句法分析以及知识获取等各种用途。表 4-1列出了当前一些比较著名的树库及其相关信息。

树库建设的第一个内容是制订标注规范。合理的标注规范应该尽量简洁、清晰，便于标注者理解，这对保证树库的一致性非常重要。另外，标注体系应该具有很好的语法覆盖性，这对于句法分析的效果以及基于句法分析的应用有着很大的影响。

树库的标注规范主要有以下两部分：标注方式和标记集。标注方式代表语法树的组织形式，即遵循什么原则组织句子中词与词之间的关系。一般来说，标注规范都遵循某一已有的语法理论。如表 4-1所示，目前的树库所遵循的语法体系主要有以下几类：

（1）短语结构语法，如宾夕法尼亚大学的英文树库 Penn English

Treebank(PET)[15]和中文树库 Penn Chinese Treebank(PCT)[21]，清华大学的汉语树库 TCT973[22]。

（2）依存语法，如布拉格树库 Prague Dependency Treebank(PDT)[23]，施乐公司的依存树库 PARC 700 Dependency Bank(DEPBANK)[24]，清华大学的语义依存树库 Semantic Dependency Net(SDN)[135]。

（3）自定义语法，如台湾中央研究院 Sinica Treebank(Sinica)[20]，德语树库 TIGER Treebank (TIGER)[16]，香港理工大学的汉语浅层树库 PolyU Treebank(PolyU)[136]。

表4-1 几个著名树库信息

Table 4-1 The information of several famous treebanks

树库	语言	语法	规模	时间	用途	单位
PET	英语	短语结构	300 万词	1994	多种用途	Upenn
PCT	汉语	短语结构	25 万词	2003	多种用途	Upenn
TIGER	德语	自定义	3.5 万句	2002	多种用途	Saarland University
Sinica	汉语	自定义	24 万词	2000	多种用途	台湾中央研究院
PolyU	汉语	浅层分析	100 万词	2004	浅层分析搭配抽取	香港理工大学
PDT	捷克	依存语法	50 万词	1998	多种用途	Charles University
DEPBANK	英语	依存语法	700 句	2003	谓词-论元结构评价	Palo Atto Research Center
SDN	汉语	依存语法	100 万词	2003	知识抽取	清华大学电子系
TCT973	汉语	短语结构	100 万字	2003	多种用途	清华大学计算机系

（注：表中列举的时间是文章发表的时间，树库的规模是文章撰写时所标注的数量，很多树库的标注工作在文章发表之后继续进行，所以表中的规模不代表其树库的最终规模。）

标记集代表加工的深度，包括词性标记集，短语类型标记集，语法功能标记集及语义角色标记集：

- 词性是指如名词（N），动词（V），形容词（ADJ）等标记；
- 短语类型是指如名词短语（NP），动词短语（VP），介词短语（PP）等标记；

- 语法功能是指如主语（SBJ），宾语（OBJ），状语（ADV）等标记；
- 语义角色是指如位置（LOC），方式（MNR），目的（PRP）等标记。

当前的树库多倾向于标注深层的语言学知识，以满足多种需求。比如 Sinica 树库将标注的重点放在语义结构上；PARC 树库在大量语法特征的基础上，标注了谓词-论元结构；PCT 也标注了大量的功能标记和空类型（null categories）；SDN 标注了多达 59 个标记的语义依存关系。

本文标注树库的目标是构建高性能的依存分析器，将重点放在语法属性的标注上。选择的语料是人民日报语料，在已有的人工分词和词性标注的基础上，进行了句子片段类型、动词子类、名词复合短语边界和依存关系的标注。标注过程采用自动标注与人工标注相结合的方式进行，自动标注时采用一种增量式的标注策略。下面依次介绍依存树库的标注体系和标注过程。

4.2.1 依存标注体系

除了语言外，标注规范是各种树库相互区别的重要特征。一个合理的规范不但能够很好的反映语言的规律，为句法分析的应用研究建立良好的基础，而且层次清晰，结构合理，易于被使用者理解。这点对树库的质量，人工校对的效率有着很大的影响。合理的标准规范也便于用户对训练和测试数据进行分析，进而发现其中的语言规律。

针对汉语的特点，我们制定了具体的依存关系规范，这个规范规定了两个问题，即在一个句子中，哪些词之间应该存在依存关系，以及这些关系属于什么样的类型。对第一个问题，遵循两个原则：

（1） 语义原则

一般来说，语法分析是为语义理解服务的，所以我们认为，在句子中，语义上存在联系的词语之间存在依存关系，也可以说，由于这些词语之间发生依存关系才产生新的语义，本文称之为“语义原则”。在标注依存关系的时候，首先遵守这个原则。例如：

海尔具有先进的经营管理经验。

在这个句子中，海尔具有的是“经验”，“具有”和“经验”这两个词义进行组合，才能产生符合这个句子的语义，所以，“具有”和“经验”之间存在依存关系。

（2） 主干原则

在一个句子中，有些词属于主要的词，对表达句子的意思起主要作用，在句子中不可缺少；有些词属于次要的词，在句子中起辅助作用，一些次要

词在句子仅起完善句子结构的作用，去掉也不影响语义的表达。在标注依存关系时，尽量保证主要的词作为依存关系的核心，其附属成分依存于该核心词。这样，对于后面的应用，只要根据依存关系，抽取句子的主要词语，就能得到句子的主干，本文称之为“主干原则”。例如：

加强和改善金融管理是必须解决好的重大课题。

该句的依存关系标注为如图 4-1所示。

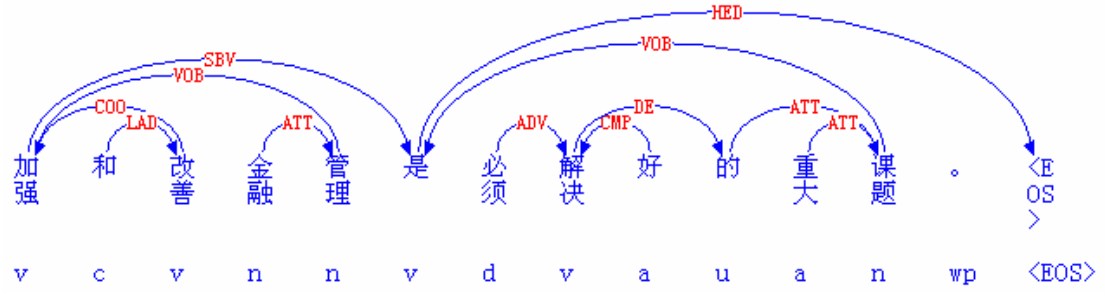


图4-1 依存标注的例子

Figure 4-1 An example of dependency annotation

通过句子的依存结构，得到句子的主干是：加强是课题。将主语进一步细化，可以说：加强管理是课题。

另外，一些定中结构或状中结构中，不同语法标注对语义的影响不大。修饰成分可以修饰邻接词，也可以修饰局部结构的核心词，而语义效果基本不变。此时遵守“先语义后向心”的原则，即先从语义上考虑，将语义上联系最为紧密的词标注在一起，如：

也未能阻止这个进球（也 → 未能）

“阻止”是核心词，但在语义上，副词“也”和“未能”的关系比和“阻止”的关系更为密切，所以这种情况下“也”依存于“未能”。

在语义联系没有区别的情况下，采用“向心”的原则，规定修饰词依存于整个局部结构的核心词。如：

正抓紧进行 （正 → 进行）

香港对外出口贸易 （香港 → 贸易）

将会全面展开 （将 → 展开）

这几个结构中，“正”修饰“抓紧”或“进行”，“香港”修饰“对外”或“贸易”，“将”修饰“会”或“展开”，对语义均没有影响。

第二个问题是制订依存关系的类型，即确定词之间的依存关系的标记集。

在确定依存关系类型上，本文树库的标注体系主要考虑了以下几个因素：

（1）语法关系的覆盖度；

（2）关系标记的数量；

（3）同其他标注体系的相互转换；

（4）易于标注者和使用者理解。

为了覆盖各种语法现象，一定数量的关系类型是必需的，但是太多、太细的关系类型无疑会给标注者增加困难，也会导致数据稀疏的问题。另外，本标注体系参考了多个树库的标注体系，充分考虑了同已有树库之间的转换，制定了一个包含 24 个语法标记的关系列表，如表 4-2所示。

表4-2 依存关系标记集

Table 4-2 The tagset dependency relations

关系	符号	关系	符号
定中关系	ATT(attribute)	“的”字结构	DE
数量关系	QUN(quantity)	“地”字结构	DI
并列关系	COO(coordinate)	“得”字结构	DEI
同位关系	APP(appositive)	“把”字结构	BA
前附加关系	LAD(left adjunct)	“被”字结构	BEI
后附加关系	RAD(right adjunct)	状中结构	ADV(adverbial)
比拟关系	SIM(similarity）	动宾关系	VOB(verb-object)
语态结构	MT(mood-tense)	主谓关系	SBV(subject-verb)
独立结构	IS(indep. structure)	连动结构	VV(verb-verb)
动补结构	CMP(complement）	关联结构	CNJ(conjunctive)
介宾关系	POB(prep-obj)	独立分句	IC(indep. clause)
核心	HED (head)	依存分句	DC(dep. clause)

汉语的依存关系主要分为三个层次：局部依存关系，单句依存关系，复句依存关系。其中，局部依存关系是指句子中除谓语之外的其他成分中词与词之间的关系；单句依存关系是指句子中谓语核心词与其他词之间的关系；复句依存关系是指单句与单句之间的关系。其中，有些依存关系既属于局部依存关系，也属于单句依存关系。

4.2.2 标注过程

为了节省标注者的劳动量，提高标注的效率，本文采用了增量式的标注策略，即逐渐增加标注文本的数量，每个标注文本采用自动分析和人工标注相结合的方式。整个标注过程共分为以下几步：

（1） 无指导依存分析

从大规模的生语料中自动抽取搭配，根据搭配的强度，对小规模文本进行依存句法分析，生成符合依存语法的结果。无指导的依存分析错误较多，但是能够提供给标注者一棵完整的依存树，标注者在分析结果的基础上进行修改，可以节省工作量。经过该步骤，标注了一个 7,300 个句子的小规模依存树库 CDT-1。CDT-1 选择的文本来自于新加坡小学课本，里面的句子长度较小，并且分析时只进行了依存骨架分析，并未标注依存弧的关系类型，这样就减少自动分析和人工标注的难度，较容易地获得了初始标注语料。

（2） 基于小规模训练集的依存分析

在 CDT-1 的基础上，使用有指导的学习方法训练了一个依存句法分析器[33]。由于数据规模小，该分析器主要利用词性信息计算依存弧的概率，概率计算的方法使用的是极大似然估计。使用该依存分析器，自动分析了 45,000 个句子，然后用人工对自动分析的结果进行了校正。本次标注的集合作为 CDT-2，同 CDT-1 一样，该树库也只进行了依存骨架分析，未标注依存弧的关系类型。

（3） 基于词汇化的依存分析

利用步骤 2 的标注结果 CDT-2，本文得到了一个大规模的训练集。利用 CDT-2 中丰富的词汇信息，本文构建了一个词汇化的依存骨架分析器[137]，对 10,000 个句子进行自动分析。之后，采用人工方式校正句法分析的结果，校正的同时，对依存弧的关系类型进行标注。由于两个词性节点之间的关系类型是不确定的，例如，两个动词之间至少存在以下四种依存关系：

V←V：VOB VV COO CMP

为此，本文设计了一个人工辅助的半自动过程对关系类型进行标注，如图 4-2 所示。

算法中，列表（list）用于存储所有可能的关系标记，队列（queue）用于存储某一个词性搭配已经标注过的关系标记，每个词性对对应一个队列。开始标注时，关系标记从队列或列表中自动选择，然后由人工校正。经过校正的关系标记存入相应的队列中，并按照出现频率进行排序，高频的标记位于

队列的首部。

标注的初始阶段，主要由人工选择标记，经过一段时间标注之后，算法自动记录了高频关系类型，并作为首选提供给标注者，大大提高了标注的效率。

```
Set all the queue empty initially
for each dependency arc and its queue q_i
if q_i is not empty
    Choose its top one R as relation tag
else
    Choose one tag R from list manually
    Append R to q_i
The count of R increases one
Sort q_i by frequency
```

图4-2 依存关系标注算法

Figure 4-2 The algorithm of dependency annotation

整个标注过程中，除了通过依存分析器提高树库标注的效率之外，开发的辅助工具也发挥了很大的作用。在不同的标注阶段，均开发了相应的工具，在以下几方面对标注人员给与帮助：

- 加速标注过程。通过可视化工作显示自动分析的结果，然后标注人员通过点击鼠标修改错误的依存弧和依存关系。
- 检查标注过程中出现的错误。人工标注时难免出现一些错误，导致出现非法的语法结构。标注工具会自动检验标注结果，向标注者提示句子中出现的不符合依存语法规范的地方。
- 标注后处理。树库中的一些标注错误往往重复出现，标注工具提供了一个交互式的界面，如果检查者输入一个错误的语法结构，工具返回全部此类错误，供检查者修改。

按照以上标注过程，本文构建了一个含有 10,000 个句子的高质量依存树库，用于汉语依存分析的训练和测试[138]，并已将该树库向学术界免费共享。

4.3 句子片段识别

4.3.1 片段分类

句子长度对汉语句法分析的准确率有着很大的影响，这一点可以通过比较 Sinica Treebank 和 Penn Chinese Treebank 的分析结果看出。Penn Chinese Treebank 的句子平均长度是 27 个词，目前的句法分析最好水平为 80%左右[41]；Sinica Treebank 的句子平均长度是 5.9 个词，对该树库的多语依存分析就已经达到了 90%的准确率[100]。因此，对句子进行片段切分，缩短句法分析单元的长度，就显得很有必要。

为了解决句子的切分问题，首先需要确定标点的作用以及句子的概念。在句法分析中，通常把以句号、问号或叹号结尾的一段文字作为一个完整的句子，而逗号通常只作为句子中间的暂停标志。但由于汉语对标点的使用没有严格的语法限制，尤其是对逗号的使用更为随意一些，有时表达完一个意思之后，会以逗号作为停顿，继续表达下一个意思，导致了句子的长度增加。另外一种情况是，很多句子在语法结构没有完整时，就用逗号作为停顿，导致句子中的很多片段缺少主语或宾语。

本文将以逗号、冒号、分号、句号、问号和叹号结尾的字符串称为一个片段。在本文所所使用的 863 词性标记集中，这些标点被统一标识为 wp。根据片段的语法结构，本文制订了一个分类标准，将片段分为不同的类别，类别值通过片段末尾的标点进行标识。片段的类别共分为以下五种：

（1）分句。分句是语法结构完整的片段，分句之间只有语义上的联系，在句法结构上没有联系，标识的方法是将片段末尾标点的词性标注为 wp1，如：

> 香港中华总商会今天上午举行会员元旦团拜酒会，新华社香港分社副社长秦文俊出席了酒会。
> 香港/ns 中华/nz 总商会/n 今天/nt 上午/nt 举行/v 会员/n 元旦/nt 团拜/n 酒会/n ，/wp1 新华社/ni 香港/ns 分社/n 副/b 社长/n 秦文俊/nh 出席/v 了/u 酒会/n 。/wp1

（2）无主语结构。片段的语法结构不完整，主语被省略或者位于前面的片段之中。将该结构末尾的标点标识为 wp2，如：

按保护价敞开收购，充分掌握粮源，才能保护农民的利益，才能做到顺价销售，才不会被粮贩子牵着鼻子走。
按/p 保护价/n 敞开/d 收购/v ，/wp2 充分/d 掌握/v 粮源/n ，/wp2 才/d 能/v 保护/v 农民/n 的/u 利益/n ，/wp2 才/d 能/v 做到/v 顺价/d 销售/v ，/wp2 才/d 不/d 会/v 被/p 粮贩子/n 牵/v 着/u 鼻子/n 走/v 。/wp2

（3）无宾语结构。片段的谓语是及物动词，但是谓语和宾语之间被标点间隔，将该结构末尾的标点标识为 wp3，如：

我们在调研过程中了解到，目前我国企业集团的发展总体上是比较顺利的，但存在一些值得注意的问题。
我们/r 在/p 调研/n 过程/n 中/nd 了解/v 到/v ，/wp3 目前/nt 我国/n 企业/n 集团/n 的/u 发展/n 总体/n 上/nd 是/v 比较/d 顺利/a 的/u ，/wp1 但/c 存在/v 一些/m 值得/v 注意/v 的/u 问题/n 。/wp2

（4）短语。片段由一个句法成分构成，是一个短语或者一个词，片段中无谓语动词。通常是名词短语、介词短语或者连词。将该结构末尾的标点标识为 wp4，如：

今日清晨，在嘹亮的国歌声中，拉萨隆重举行升国旗仪式。
今日/nt 清晨/nt ，/wp4 在/p 嘹亮/a 的/u 国歌声/n 中/nd ，/wp4 拉萨/ns 隆重/d 举行/v 升/v 国旗/n 仪式/n 。/wp1

（5）停顿语。片段由两个或以上的句法成分构成，并且在句法结构和语义上均不完整，标点只起停顿的作用。将该结构末尾的标点标识为 wp5，如：

双方应在此基础上，妥善处理两国间存在的问题。
双方/n 应/v 在/p 此/r 基础/n 上/nd ，/wp5妥善/a 处理/v 两/m 国/n 间/nd 存在/v 的/u 问题/n 。/wp3

汉语句子的平均长度虽然较长，但如果将句子分割为多个片段，则每个部分的长度减小。Sinica Treebank 中的句子就是经过分割之后的片段，所以句子的平均长度很小。但完全以片段作为句法分析的单位并不合理，如（5）中的句子经过依存句法分析之后，结果如图 4-3所示。句子中的第一个片段属于停顿语，由三个独立的句法成分组成，每个成分通过一个节点同外部发生关系。如果单独对这样的片段进行句法分析，将会发生错误。所以，停顿语后的标点只起暂停的作

用，并不能将这样的片段从句子中分割出来。

另外，对于可以分割的句子，片段之间也存在着句法关系，如图 4-4所示，三个片段分别通过核心词发生依存关系。获得片段的类别之后，将有助于识别片段之间的依存关系。

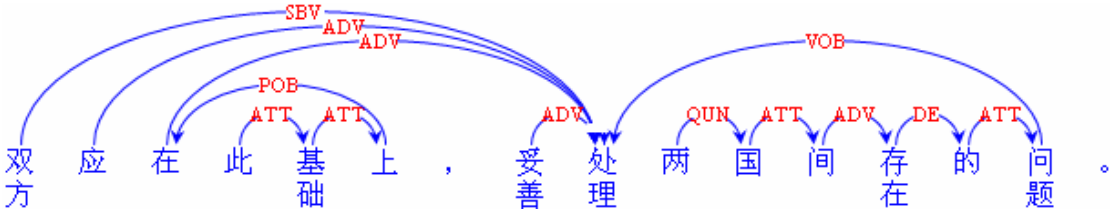


图4-3 依存句法分析结果（第一个片段有三个节点同外部发生联系）

Figure 4-3 The result of dependency parsing (There are three nodes in the first segment that link with the second one)

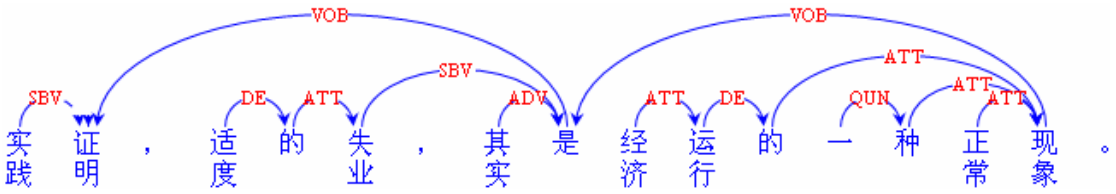


图4-4 依存句法分析结果（片段之间存在依存关系）

Figure 4-4 The result of dependency parsing (There are dependency relations between the segments)

针对句子中的这两种情况，本文首先对片段的类型进行标注，即根据片段末尾的标点识别片段的属性。然后，对片段进行依存分析之后，再识别片段之间的关系。

4.3.2 基于 SVM 的片段类型识别

支持向量机（SVM）方法建立在统计学习的 *VC* 维理论和结构风险最小化（Structure Risk Minimize , SRM）原理之上，根据有限的样本信息在模型的复杂性（即对特定训练样本的学习精度）和学习能力（即无错误的识别任意样本的能力）之间寻求最佳折衷，以期获得最好的推广能力。在 SVM 中，SRM 体现为寻找一个超平面，它将正例训练样本与反例以最大的间隙分开。SVM 实现的基本思想是：

给定n组训练样本(x_1, y_1)，…，(x_l, y_l)，…，(x_n, y_n)，$l = 1$，…，n，$x_l \in \mathrm{Rd}$，$y_l \in \{-1, +1\}$，S 通过某种事先选择的非线性映射将输入向量x

映射到一个高维特征空间Z上（映射函数为Φ：Rd $\rightarrow$Z），然后在这个高维特征空间中构造最优分类超平面。在线性情况下，$\Phi(x)=x$；在非线性情况下，映射函数$\Phi(\cdot)$以核函数K(·,·)的形式体现，核函数定义了Z空间的内积。SVM 给出的可分超平面如下：

$$w\times\Phi(x)+b=0 \tag{4-1}$$

如果离超平面最近的训练样本与超平面之间的距离最大，则该超平面称为最优超平面,该距离与权值向量w 的模成反比。计算这个超平面等价于求解以下的优化问题。

$$V(w,b,\xi)=\frac{1}{2}\|w\|+C\sum_{i=1}^{n}\xi_i \tag{4-2}$$

其约束条件为：

$$Y_i[w\times\Phi(x_i)+b]\geqslant 1-\xi_i \tag{4-3}$$

其中，$i=1,2,\cdots,n,\xi_i\geqslant 0$。

约束条件要求所有的训练样本被正确分类，线性不可分时训练误差用松弛项ξ_i 表示。

（4-2）式中的C（惩罚因子）是某个指定的常数，实现训练错误与算法复杂度之间的折中。这个优化问题是一个凸二次规划问题，它的对偶形式如下：

$$W(\alpha)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j K(x_i,x_j) \tag{4-4}$$

其中，$i=1,2,\cdots,n,C\geqslant\alpha_i\geqslant 0$，并且：

$$\sum_{i=1}^{n}y_i\alpha_i=0$$

最优解α^*决定了最优超平面的系数：

$$w^*=\sum_{i=1}^{n}y_i\alpha_i^*\Phi(x_i) \tag{4-5}$$

将w*代入式（4-1）得到SVM的决策超平面。其中阈值b由非上界的支撑向量(unbounded support vector) 解得：$b=y-w\times x$

$$y=\operatorname{sgn}\sum_{i=1}^{n}y_i\alpha_i^*K(x_i\times x)+b \tag{4-6}$$

由决策超平面可得样本的预测值，接下来的一个重要工作就是选择SVM的核函数和核参数，以获得最好的分类性能。Vapnik等人在研究中发现，不同的核函数对性能的影响不大，反而SVM核函数的参数和误差惩罚因子*C*是影响SVM性能的关键因素。对于不同的特征空间，数据的分布是不同的，经验风险随VC维的变化也不同，相应的最优的*C*值也不同。要想获得推广能力良好的分类器，首先要选择合适的SVM核函数参数，将数据映射到合适的特征空间，然后针对该确定的特征空间寻找合适的*C*值，以使学习器的置信范围和经验风险具有最佳比例。

目前，SVM分类算法的研究已经比较成熟，被广泛应用与人工智能的各个领域之中，并有许多分类效果很好的工具包免费发布，本文使用已有的工具包进行句子的片段类型和片段关系识别。

本文对经过标点分割之后的片段按照上节的分类标准进行识别，标注其末尾标点的角色，然后根据片段的类型识别结果，对片段做进一步的处理。片段共有五个类别，属于多元分类问题。SVM 分类器从每个片段的上下文中抽取特征向量，所使用的特征如表 4-3所示。

表4-3 片段类型识别的上下文特征

Table 4-3 The context features of type identification

特征	描述	特征	描述
len	片段长度是否大于 4	shi	是否含有“是”
vg	是否含有一般动词	you	是否含有“有”
vx	是否含有系动词	zhe	是否含有“着、了、过”
vt	是否含有及物动词	vg_n	是否含有动宾搭配
vi	是否含有不及物动词	p_nd	是否有介词和方位词的组合
nd	末尾是否为方位名词	tag1	第一个词的词性
de	是否含有“的”	tag_1	最后一个词的词性
dei	是否含有“得”	punct	片段末尾的标点

4.3.3 基于 SVM 的片段关系识别

根据片段识别的结果，将句子进行分割，分割的位置选择标识为 wp1、wp2、wp3 和 wp4 的标点。这些分割之后的部分在语法上是独立的，即只有一个核心词同外部发生联系。分割之后，以每个片段作为句法分析的基本单元，本文使用一个已有的依存句法分析器对片段进行分析[137]。为了得到完整的句法分析树，

还需要将片段的分析结果进行合并。由于每个片段只有一个核心节点，各个片段之间通过该核心节点发生句法关系。只要分析出各个片段的核心节点之间的依存关系，即可将句子合并。

片段之间的关系同片段内部的关系不同，片段内部的依存关系主要是词与词之间的关系，片段间的依存关系则主要是短语或句子之间的关系，主要的关系类型如表 4-4所示。

表4-4 片段之间的依存关系类型

Table 4-4 The types of dependency relations between the segments

关系类型	描述	关系类型	描述
IC	独立分句	VV	连动结构
SBV	主谓结构	ADV	状中结构
VOB	动宾结构	CNJ	关联结构

本文将片段间依存关系的识别作为多元分类的问题，仍然使用 SVM 分类器解决。同上节不同的是，片段已经经过了依存分析，其内部的依存关系已经获得，可以利用更多的信息作为特征向量。本文分别从两个片段之中分别抽取特征，选择的特征如表 4-5所示。

表4-5 片段间依存关系识别的上下文特征

Table 4-5 The context features of relation identification between the segments

特征	描述	特征	描述
Len_B	长度是否为 1	RChd_B	根节点同最右侧孩子关系
Root_B	片段根节点的词性	Subj_B	是否含有主语
Pred_B	根节点是否可以做谓语	Obj_B	是否含有宾语
Adv_B	根节点是否是副词	Semi_L	片段末尾标点是否为分号
Cnj_B	根节点是否为连词	Posi_L	根节点是否在片段尾部
LChd_B	根节点同最左侧孩子关系	Posi_R	根节点是否在片段首部

表 4-5中，后缀为“B”的特征从两个片段中分别抽取，后缀为“L”的特征只从左面的片段中抽取，后缀为“R”的特征只从右面的片段中抽取。识别片段之间的依存分析关系时，还有一种情况，就是两个片段之间不存在依存关系，如图 4-4中的前两个片段，将此种情况的关系类型表示为 NOT。获得片段之间的依

存关系之后，需要将各个片段进行合并，形成一棵完整的依存分析树。本文采用局部寻优的过程对片段进行合并，算法的描述如图 4-5所示。

```
size = the number of Segments
While size is more than 1
For i = 0 to size
    Relation = GetRelation (Segment_i, Segment_i+1)
    If Relation != NOT
        Join (Segment_i, Segment_i+1)
size = the number of Segments
```

图4-5 片段合并算法

Figure 4-5 The algorithm of segments union

算法中，函数 GetRel( )从分类器中获得两个片段之间的依存关系，Join( )利用获得的关系将两个片段的根节点连接到一起。

4.4 实验及分析

该实验使用的数据是《哈工大信息检索研究室汉语依存树库》，数据来自 1998 年上半年的人民日报，该树库共有 212527 个词，1 万个句子，每个句子的平均长度为 21.3 个词。全部的实验中，以前 8000 个句子作为训练数据，8000~9000 句作为开发集，最后 1000 个句子作为测试集。

4.4.1 片段类型识别

本文第一个实验是识别片段的类别，即标注片段末尾的标点。树库中的片段类型分布如表 4-6所示。实验中，SVM 分类器选择的是 LIBSVM[1]。按照表 4-3 所列的特征，5 类标点的标注结果如表 4-7所示。

表4-6 片段类型的分布

Table 4-6 The types of segments in the treebank

片段类型	wp1	wp 2	wp 3	wp 4	wp 5
片段数量	7838	8975	2367	3673	943

[1] http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html

表4-7 片段类型的识别结果

Table 4-7 The results of the type identification

	wp1	wp 2	wp 3	wp 4	wp 5
准确率	0.7623	0.8056	0.7887	0.8857	0.3918
召回率	0.7893	0.8788	0.5840	0.7815	0.4000
F-Score	0.7756	0.8406	0.6711	0.8304	0.3958

从表 4-7的结果中，wp5 的片段类型识别准确率较低，主要是因为该片段在句子中的比例较少，特征数据不足，导致分类器在识别的时候产生困难。

4.4.2 片段关系识别

第二个实验是识别片段之间的关系类型，仍然以 LIBSVM 作为分类器。按照表 4-5所列的特征，关系类型的识别结果如表 4-8所示。

表4-8 片段间依存关系的识别结果

Table 4-8 The results of relation identification between the segments

	ADV	CNJ	IC	SBV	VOB	VV	NOT
准确率	0.8772	0.6923	0.7282	0.7273	0.6915	0.6476	0.7364
召回率	0.8230	0.7826	0.5792	0.6316	0.5963	0.9168	0.5398
F-Score	0.8493	0.7347	0.6452	0.6761	0.6404	0.7591	0.6230

表 4-8中，标记“NOT”表示两个片段之间不存在依存关系。这里，将 NOT 也作为一个关系类型，同其他关系一同识别，简化了问题的分析。

4.4.3 整句依存分析

经过片段合并之后，得到完整的依存分析结果，本文使用关系准确率、搭配准确率和句子核心词三个指标对结果进行评价，结果如表 4-9所示。

表4-9 整句的依存分析结果

Table 4-9 The results of dependency parsing on the sentence

	依存关系	依存搭配	核心词
整句	0.6455	0.6957	0.701
片段	0.6757	0.7292	0.754

表 4-9中，第 2 行是未对句子进行片段划分，直接进行整句分析的依存分析结果；第 3 行是按照本文所描述基于片段的句法分析所得到的结果。（本实验使

用的依存分析数据为第 6 章多语依存分析器在中文树库 CDT 上的训练和分析结果）

在本实验的结果中，基于片段的分析和整句分析相比，依存句法分析的准确率有了较大的提高，表明基于片段的分析方法对句法分析的改善具有显著的效果。从分析结果中也能够看到，句子核心词的准确率提高幅度较大，这是因为片段的长度变小，减少了句子中的干扰，句法分析能够更准确的找到句子的根节点。

4.4.4 同其他工作的比较

作为比较，本文使用了 MSTParser 对本实验的树库 CDT 进行了分析。MSTParser 是一个语言无关的多语依存句法分析器，该分析器的多语依存分析达到了很高的性能，CoNLL2006 的评测中，取得了第一名的成绩[98]。MSTParser 的分析结果如表 4-10所示。

表4-10 MSTParser 的依存句法分析结果

Table 4-10 The parsing results of MSTParser

	MST-Sinica	MST-CDT
Average Length	5.9	21.3
Labeled Score[2]	86.60%[3]	63.12 %
Unlabeled Score	90.76%	66.09 %

表 4-10中，第二列是在 Sinica 树库上的分析结果，第三列是在 CDT 上的分析结果。从表中能够看出，MSTParser 在两个树库上的分析结果相差很大，进一步说明了句子长度对句法分析的准确率有着很大的影响。

Kim 等使用分治策略来简化英语句法分析的复杂度，应用最大熵的方法对句子进行切分[139]。该工作作为机器翻译的一部分，以减少句法分析的时空复杂度为目标，本文的重点在于减少句法分析的歧义结构，提高句法分析的准确率。

对汉语长句的分割，Jin 等人做了一些研究工作，将逗号分为分句之内和分句之间两种情况[140]。对逗号进行标注之后，将句子从分句之间的标点处断开。本文的不同之处在于对所有的片段末尾标点进行标注，包括冒号、句号等。同时，本文还探讨了分句之间的依存关系识别。

[2] Labeled Score 和 Unlabeled Score 是 CoNLL-X shared task 的评测标准，具体介绍见第六章。

[3] 该值是用 McDonald 主页上提高的版本重新进行分析得到的结果，同 MSTParser 在 CoNLL2006 评测上的值略有差别。

4.5 本章小结

针对自然语言句子长度大、语法结构复杂的问题，本章提出了一种面向句法分析的句子片段识别方法，先将句子划分为多个片断，通过片断分割来缩减句子的长度，从而简化语法结构。对每个片断进行依存分析之后，再通过机器学习的方法识别各片断之间的依存关系，最后将各个片断组合为一个完整的分析树。

基于片段的分析方法在汉语依存分析中表现出了积极的作用，并且这种思想同样可以用于其他类似的语言之中，只要该语言的句子能被分为一些在句法上独立的片段，就可以采用这样的分析策略。对降低句法分析的复杂度，提高句子的准确率方面将会有所帮助。

第5章 基于动态局部优化的汉语依存分析

5.1 引言

句法分析的建模方式可以分为两种，一是将所有句法结构统一对待，用同一个模型进行分析。这种方式能够简化建模过程，但对数据的规模要求较高，因为语言中的一些语法现象并不常见，或者其构成方式过于复杂，导致训练样本不足，统一建模的方法难以处理此类问题。

另一种方式是根据语法结构的特点，针对不同的语言现象，采用不同方法进行处理。语言中的句法结构多种多样，有些结构在语法构成上相差很大。在句法树库的规模不是很大的情况下，采用这种方法通常能达到较好的建模效果。

汉语中，主谓、动宾、定中以及状中这样的语法结构数量很多，在句子中占有较大的比例。但由于汉语的语法比较灵活，还存在着很多其他形式的语法结构，如并列结构、时间短语、名词复合短语等。这些结构在语法上具有自己的特点，而且每种结构的数量不是很多，如果利用统一建模的方法对这些结构进行处理，每种结构的训练数据不够充足，识别的效果不好。

本文采用分治策略，对一些特定的语言现象单独进行分析，对其他语法结构则建立统一的概率模型。该方法有效地解决了少数语言现象因训练数据不足导致的准确率不高的情况。

在解码方式上，本文提出了一种基于动态局部优化的搜索算法。句法分析搜索策略主要包括两种，一种是全局最优的搜索，另一种是局部寻优的搜索。在依存句法的骨架分析中，采用穷尽式搜索的全局寻优策略表现了很好的性能[77]。但是随着关系句法模型的参数增多，算法的搜索效率成为一个必须考虑的问题。近年来，基于局部寻优的确定性搜索策略受到了广泛的关注。该策略将句法分析树的生成过程分解为一系列的操作，然后用一个分类器选择每个步骤的最佳操作。Yamada 等定义三种操作构建依存树，并使用 SVM 分类器对操作进行选择，然后使用一个自底向上的过程构建依存树[56]。Nivre 使用一个自底向上和自顶向下的混合策略构建依存分析树，并使用基于记忆（memory-based）的学习方法训练分类器[82]。

在生成句法树的过程中，确定性分析算法不记录冗余信息，并且不进行回溯，使得算法能够在线性时间内完成解码。但是当搜索局部最优解的过程按照句子的顺序进行时，前面的一些错误分析结果会对后续的分析造成影响，造成错误蔓延，尤其对一些长距离的依存关结构影响更为严重。Jin 使用两步归约策略来减少这种错误，提高了长距离依存的准确率[57]。

本章尝试了一种新的搜索策略——基于动态局部优化的确定性搜索算法，根据依存弧关系的概率，算法动态寻找概率最高的局部依存结构，而不是完全按照从左到右的顺序分析句子。在归约的过程中，算法利用大量的结构信息检查依存关系的合理性，使得解码过程能够得到合理的依存结果。

本章首先介绍汉语中一些特定语言结构的处理，包括动词习语、简单并列结构、时间及数字短语。分析完这些结构之后，以结构的核心词参与整个句子的分析，对整个句子建立统一的依存概率模型，然后使用动态局部优化的方法对句子进行解码，获得最终的依存分析结果。最后给出本实验的结果以及实验分析。

5.2 特定语言结构分析

通过对数据的观察发现，树库中存在一些这样的语言结构：句法构成的方式较为复杂，简单的统计学习无法掌握其构成规律，如并列结构，动词习语；或者句法结构的构成规律较为明显，但是在训练数据中出现的次数并不是很多，如时间短语，数字短语。对这几类结构，本章采用统计结合规则的方法对每种类型分别进行处理。

5.2.1 动词习语

本文的习语是指经常使用但不在词表中出现的词，这类短语的语义固定，无法拆分。虽然都不在词表中，但习语不同于新词，新词是随着新的事件而出现的词，使用的频率不稳定。而动词习语是广泛使用的词，具有很强的语法规律。同时，动词习语也表现出很强的开放性，比如，“顺流而下”也可以转化成“顺江而下”、“顺河而下”。习语中，动词词性的短语占有较大的比重，本文主要研究动词习语的识别和分析。这些习语的识别也会对信息抽取、机器翻译等应用产生帮助。

这里将动词习语分为两种：二字习语和四字习语。二字习语的构成形式简单，多为动宾结构（V+N），宾语主要为名词或简称，如：

烧 菜
挑 刺
访 美

在句法分析中，这类词的识别要容易一些，不会造成太大的影响。 但四字习语在构成方式上要复杂得多，表现形式和成语接近，主要有：

（1）V+N+V+N结构：

望 女 成 凤
舔 犊 护 犊
报 喜 藏 忧

（2）V+N+C+N结构：

顺 流 而 下
围 泽 而 垦

（3）V+N+P+N结构：

还 路 于 民
爱 书 如 命

这类习语结构多样，数量开放，无法在词表中全部收录，而且在句法分析中难以正确识别其边界，错误率非常高。本文的识别方法是先在树库中标出所有的动词习语，然后按照其语法构成进行分类，通过统计获得习语的边界和内部语法构成规律。

5.2.2 并列短语

并列短语是由两个或两个以上的成分并列在一起而构成的短语。并列短语对于语言的深入分析非常重要，国内已有一些工作对此进行了专门的研究[44, 141]。并列短语的构成方式非常灵活，并列的部分既可以是相同的成分，也可以是不同的成分，既可以是一个词，也可以是多个词。根据短语的并列标志，可分为三种情况：

（1）连词并列。并列标志包括：和、及、与、或、而、并、并且等，构成形式为“A 和 B”，如：

老师和学生

（2）顿号并列。由顿号或连词作为并列标志，构成形式为“A、B、C 和 D”，如：

督促、检查、消化

（3）无标记并列。没有并列标志，构成形式为“AB”，如：

法中友好小组（“法中”为并列结构）

根据并列成分的复杂程度可以将并列结构分为两类：

（1）简单并列结构。并列的成分具有单一性和一致性，即每个成分为一个词或短语，且各个成分在语法结构上相同，如：

擦车、洗车

行政管理体制和机构改革

（2）复杂并列结构。并列结构的各个成分的语法属性不一致，甚至可以为句子。如：

加强司法机构的相互往来、军队之间的交往及民间交流

由于房地产开发过热和大规模的旧城改造

复杂并列结构的研究需要深层的语言分析，具有很大的难度。本文主要研究结构上较为简单的一些并列短语，对其边界和内部结构进行分析，以改善句法分析的效果。本文主要使用规则的方法分析并列短语。首先根据并列标志对并列结构进行定位，然后根据简单并列结构的单一性和一致性，识别并列短语的边界并分析短语的内部关系。

5.2.3 时间及数字短语

时间短语是指由时间词构成的、表达时间和日期的短语。时间短语的构成较为简单，即由连续两个或两个以上的时间词组成的序列，如：

1 9 4 9 年/nt 4 月/nt

1 9 9 5 年/nt 春节/nt

2 6 日/nt 凌晨/nt 时/nt

数字短语是指表达数字概念的短语，由数字和一些附属成分构成。数字短语也具有较明显的规律性，其构成主要包括三种情况：

（1）主干数字+附加数字，如：

5万 多

8% 左右

30 出头

十 来（岁）

（2）主干数字+附加数字+单位数字，如：

10 多 万

300 多 亿

（3）列举数字，如：

7 — 8 （人）

一个 一个 （地）

对于这两类短语，由于结构简单，本文使用规则的方法便达到了较好的识别效果，满足了句法分析的要求。

除了以上几种结构之外，本章还将名词复合短语也作为特定语法结构进行处理，名词复合短语的边界识别和内部分析较为复杂，本文已在第三章中对其进行了专门的分析。分析完这些特定语言结构之后，取每种结构的核心词作为独立的节点，同句子的其余部分进行整体分析。下面介绍句子整体分析的概率模型和搜索算法。

5.3 依存概率计算

除特定结构之外，句子中的其余部分需要建立统一的概率模型。对句子的依存分析，既要对所有的节点进行分解，以使该问题成为一个可计算的问题，又要考虑到依存弧之间所存在的联系，以减少独立假设所损失的信息。本文将依存概率的计算分为两种，一种是关系概率，即计算分解之后的两个节点之间存在某一依存关系概率；另一种是结构概率，即计算不同依存弧之间相互关联的概率。其中，关系概率是一种静态概率，在句子未分析时即可计算获得；结构概率是一种动态概率，要在依存树生成的过程中才能得到。

5.3.1 关系概率

一棵n节点依存树T由$n-1$个依存弧构成：$T = \{A_1, A_2, \cdots\cdots, A_{n-1}\}$。依存弧$A_{ij}$ $(1 \leqslant i, j \leqslant n-1)$表示为一四元组：$A_{ij} = <Node_i, Node_j, Direction, Relation>$，$Node_i$和$Node_j$是依存弧的两个节点，$Direction$表示依存弧的方向，分为向左和向右两种。$Relation$为依存弧上的关系标记，表示该依存弧的关系类型。

作为一元结构，依存弧A_{ij}的概率$P(A_{ij})$由两个节点$Node_i$和$Node_j$唯一确定。为了缓解数据稀疏，充分利用每个节点所蕴含的信息，本文使用了词性的粗类、细类以及词性和词汇的组合等多种方式计算依存弧的一元结构概率。同时利用了上下文信息，考虑邻接节点的词性对一元依存结构的影响。依存弧概率$P(A_{ij})$的计算如下：

$$P_1(A_{ij}) = P(R, D|Tag_i, Word_j)$$

$$P_2(A_{ij}) = P(R, D|Word_i, Tag_j)$$

$$P_3(A_{ij}) = P(R, D|Word_i, Word_j)$$

$P_4(A_{ij}) = P(R, D|Tag_i, Tag_j, Dist)$

$P_5(A_{ij}) = P(R, D|Tag_i, Tag_j)$

$P_6(A_{ij}) = P(R, D|Tag_{i-1}, Tag_i, Tag_j, Tag_{j+1})$

P_1至P_6是根据节点的不同表示形式分别计算出的依存关系概率。其中，R为依存关系的类型，D为依存关系的方向。Tag是词性标记，$Word$是词形本身，Tag_{i-1}和Tag_{i+1}分别是依存弧节点的前一个和后一个邻接词性。$Dist$表示节点$Node_i$和$Node_j$节点之间的距离，距离分为以下4部分：

$Dist = 1 \ if \quad j-i = 1$

$Dist = 2 \ if \quad j-i = 2$

$Dist = 3 \ if \quad 3 \leqslant j-i \leqslant 6$

$Dist = 4 \ if \quad j-i > 6$

通过对依存树库的训练，本文使用极大似然估计方法计算不同节点类型的依存关系概率。然后对这些概率值进行插值平滑，获得最终的依存弧概率P：

$$P = \lambda_1 P_1 + \lambda_2 P_2 + \lambda_3 P_3 + \lambda_4 P_4 + \lambda_5 P_5 + \lambda_6 P_6$$

根据经验，本文取插值系数$\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 0.15$，$\lambda_5 = \lambda_6 = 0.2$。

5.3.2 结构信息

完全独立的依存关系是不存在的，每个依存弧都将和其他一个或多个依存弧发生联系。本文用结构信息表示依存弧之间的语法关系，结构信息分为三种：支配度、二元结构以及序列依赖。

（1）支配度

在上节的概率模型中，假定了一个句子中各依存弧之间相互独立，即依存弧只与其构成节点有关，而与其他依存弧无关。这种假设简化了问题，却忽略依存树中的结构信息。例如，句法分析时常出现图 5-1(a)中的错误。

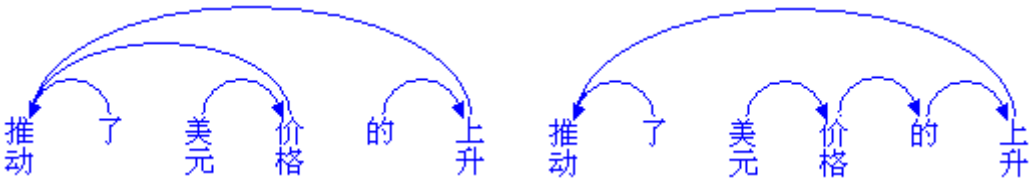


图5-1 (a)错误的分析结果　　(b)正确的分析结果

Figure 5-1 (a) The wrong result of parsing　　(b) The right result of parsing

图 5-1 (a)中的“推动”多支配了一个从属词“价格”，而“的”缺少从属词，

这种错误属于非法结构。产生这种非法结构的一个原因是搭配的过度拟合，因为“推动”和“价格”是一个很强的搭配，在这里错误的构成依存关系。显然，这种方法忽略了词汇的结构信息：“推动”所能支配的从属词上限及“的”所能支配的从属词下限。图 5-1 (b)为正确的分析结果，“推动”仅支配“上升”，“的”字支配“价格”。

为了解决此类错误，本文引入了词汇支配度。词汇支配度是用来描述一个词支配其从属词的能力，被支配度是指一个词依存于核心词的能力。如果把依存树看作有向图，则支配度相当于节点的入度，被支配度相当于节点的出度。根据依存语法，依存树中每个节点（根结点除外）有且只有一个父结点，所以每个结点的被支配度均为 1。但结点的支配度却不受此限制，一个词可同时支配多个从属词。

进一步细化，将支配度分为左支配度和右支配度，例如，图 5-2的句法树中，“取消”的支配度分别为：

左支配度：$Degree_{L}$(取消) = 1

右支配度：$Degree_{R}$ (取消) = 2

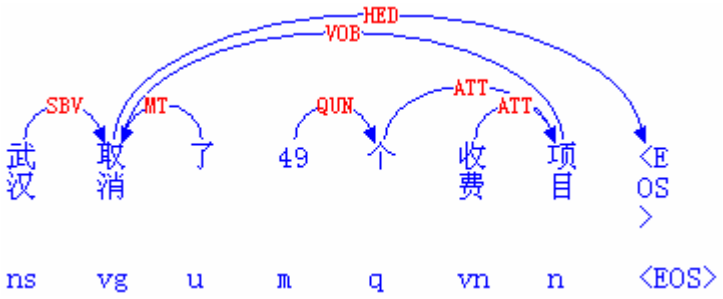


图5-2 句子的支配度

Figure 5-2 The governing degree of words

在配价语法理论中，以“价”来描述一个动词所能携带的论元个数，如果一个动词不能支配任何论元，那它就是零价动词，如“地震、刮风”；如果一个动词能支配一个论元，那它就是一价动词，如“休息、游泳”等，以此类推。本文的词汇支配度是对动词配价的扩展，这种扩展包括以下两方面：

（1）将动词扩展为所有的词；

（2）将体词性的论元扩展为所有的从属词。

做这种扩展是面向本文所要解决的问题，句法分析要求解析句子中的所有词汇关系，而除了动词外，其他词在支配从属词的时候也都表现出一定的

规律性，如介词总是支配其右侧的一个名词成分，支配度常为 1；助词“的”总是支配左侧的一个或多个词，支配度大于或等于 1。

另外，配价理论中动词的论元只为名词或代词等体词性词语，本文将词汇所支配的从属词扩充到所有的成分，支配度不但包括所支配的论元，而且对一些修饰成分、附加成分也计算在内。这两点较之单纯的动词配价更适合句法分析。

（2）二元结构

除了通过支配度控制节点的语法结构，本文还计算了二元结构概率，用来进一步约束语法结构的生成。二元结构概率是当前词在已有子节点的条件下，生成下一个依存关系的概率。如图 5-2中，当节点 “个”已经和“49”形成依存关系“QUN”之后，它的下一个关系为“ATT”的概率为：

$P(ATT|Node=个/q, R=QUN)$

对二元结构进一步细化，将其分为左右两侧的二元结构关系概率。如图 5-2中的节点“取消”已获得子节点“武汉”和“了”之后，和节点“项目”构成 VOB 关系的二元概率为：

$P_R(\text{VOB} \mid Node=取消/vg, R=MT)$

$P_L(\text{VOB} \mid Node=取消/vg, R=SBV)$

一元结构关系可以从词汇支配度的角度约束搭配的过度拟合，二元结构关系则从依存关系的邻接概率方面限制了搭配的过度拟合问题。支配度和二元结构关系的引入，为充分利用结构信息提供了有效的帮助。由于结构关系是在句法树生成的过程中动态获得，本文将支配度作为约束条件，同下节的动态局部优化算法结合，用来检验依存结构的合理性，防止非常结构的生成。二元依存概率同关系概率结合，用于计算新生成的依存弧概率。

支配度和二元结构概率均通过极大似然估计的方法从训练数据统计获得。在计算支配度时，首先获得每个节点的所有支配度概率，

左支配度概率：$P\ (\ Degree_{\mathrm{L}}(Node_i) = m\ )$

右支配度概率：$P\ (\ Degree_{\mathrm{R}}\ (Node_i) = n\ )$

然后保存其中概率较大的节点，这样的支配度比较稳定，对句法分析帮助较大。其他概率较小的节点支配度变化大，无法为句法分析提供有效的信息。本文将支配度的概率阈值取为 0.65，小于该阈值的词汇支配度不予考虑。

（3）序列依赖

句子的依存结构中，存在一种顺序依赖关系，即多个同向的依存弧依次连接，构成一个有向的依存序列。如图 5-3所示。

图5-3 (a) 子节点在左面的有向序列 (b) 子节点在右面的有向序列

Figure 5-3 (a) The directed sequence with child node being left (b) The directed sequence with child node being right

图 5-3的有向序列中，只有核心词节点的方向称为子节点方向，只有附属词节点的方向称为父节点方向。图 5-3(a)中，序列的方向为从左向右，图 5-3(b)的序列方向为从右向左。

有向序列中，边界位置的节点只充当子节点或父节点，而短语内部的节点则同时充当双重角色，既是父节点又是子节点。这样的依存弧之间存在着制约关系，即当前依存弧的归约要依赖于前一个依存弧的归约情况，这就使得归约操作需要按照一定的顺序，即从子节点的方向父节点的方向归约。图 5-3 (a)中，需要从左向右归约，才能得到正确的结果。如果先从右侧归约，比如先归约“取胜”和“后”，则“取胜”作为叶子节点从依存树中删除，节点“决赛”将失去父节点而发生错误。图 5-3 (b)中正好相反，需要从右向左归约，如先归约“批评”和“得”，则节点“好”将丢失父节点。

本文将这种依存弧的制约关系叫做序列依赖，并通过邻接依存弧的相对概率值来处理依存弧的序列依赖问题。如果有向序列的邻接依存弧概率值相差较小，说明序列依赖的程度较大，如果概率值相差很大，说明依赖的程度较弱。在归约时，算法对当前依存弧进行序列依赖检验：如果子节点方向的邻接依存概率和当前依存弧概率的比值大于阈值λ，则检验不通过，停止归约。本文的阈值取λ=0.85，如图 5-3(a)中，依存弧“取胜→后”的概率 P_1 虽然大于“决赛→取胜”的概率 P_2，但由于 $P_2 > \lambda P_1$，说明序列依赖较为严重，则停止对依存弧“取胜→后”进行归约。

另外，如果序列中子节点的依存概率较大，但因未通过检验而被阻塞，则父节点也将停止归约。下一节将对此进一步说明。

5.4 基于局部动态优化的确定性分析算法

基于历史(history-based)的模型在句法分析中一直发挥着重要的作用，该方法将句法树的生成看作一系列派生(derivation)，分析时根据以前的历史派

生规则选择下一个派生[142]。目前在 Penn Treebank 上获得较高性能的句法分析器，多数都采用基于历史的建模方法[38, 39, 79]。这种方法有很多优点，但也存在着一些问题，比如难以在模型之中使用约束条件。由于搜索过程是非确定性的，模型的概率表示不容易改变，致使模型的改进存在困难。

目前对这个问题的一个解决办法是通过重排序(Reranking)进行后处理，在重排序过程中处理新的句法知识以改进初始模型[61]。另外，非确定性算法由于进行全局搜索或近似全局搜索，所以对时间和空间的开销非常大，影响分析的效率。

另外一个解决方法是使用判别模型。该方法以分类器识别句法分析的操作，然后使用确定性搜索算法寻找合理的分析结果。由于判别式模型能够很好地融合多种特征，方便地使用各种约束条件，增加模型的灵活性，在当前的句法分析领域受到了广泛的关注，是目前较为常见的方法[56] [83]。

传统的确定性分析方法存在的一个问题是，分析时按照顺序对句子进行处理，容易导致错误蔓延的问题。图 5-4中，依存弧 a 和 b 属于高频的依存搭配，完全按照顺序分析时，a 先被归约，即使 b 的概率大于 a，也将因节点“北京”成为子节点而无法归约。

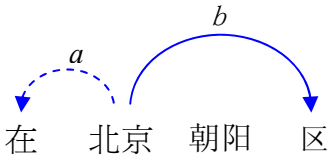


图5-4 确定性分析中的错误蔓延问题

Figure 5-4 The spread of errors in the deterministic parsing

针对这个问题，本文提出了一种动态局部优化的搜索算法，动态寻找局部最优解，而不是按照顺序分析句子。下面对算法过程进行描述。

5.4.1 算法描述

本文采用确定性分析算法对句子进行解码，但不完全按照句子的物理顺序进行处理，而是根据句子的局部依存概率值动态选择归约位置。算法的基本思想是：对句子进行多遍扫描，每次仅对依存概率较大的叶子节点进行归约，直至到根节点。

该算法的优点是灵活性高，能够方便地将多种知识融合到模型之中，向模型中增加新的信息也较为容易，便于模型的改进。还可将规则的方法应用于分析的过程中，对一些规律性强、统计数据不足的语言现象作用非常明显。

另外，该算法的时间和空间开销非常小，时间复杂度为线性。

算法的输入为一个词汇的符号序列，输出为一个依存分析树。同传统的移进-归约分析算法不同，该算法包括以下四个操作：检验（Check），归约（Reduce），插入（Insert），删除（Delete）。算法描述如图 5-5。

```
Input Sentence: S = (w1, w2, ..., wn)
Initialize:
for i = 1 to n
    Ri = Calculate(wi,wi+1);
    Put Ri into Sequence;
Sort(Sequence);
Start:
j = 1       // the top of Sequence;
While Sequence is not empty
    if   Check(Rj) is true
        Reduce(Rj);
          Adjust(Rj)
          j = 1;
    else if j is equal to n
        Reduce(R1);
          Adjust(R1)
          j = 1;
    else
          j++;
```

图5-5 动态局部优化的确定性搜索算法

Figure 5-5 The deterministic parsing algorithm based on dynamic local optimization

其中S表示输入的句子，w_i表示句子中的第i个词，R表示两个词构成的依存关系类型。函数Calculate计算相邻两个节点能够构成的依存关系类型及概率，Sort将队列中的关系类型按照概率值由大到小进行排序。然后按照概率值对队列首部的元素进行处理。Check操作检查当前关系R是否符合归约条件，如果符合，则进行归约（Reduce），然后对其余的依存弧进行调整；如

果未通过检验，则阻塞当前依存弧。调整(Adjust)操作包括删除（Delete）以归约节点作为父节点的依存弧，以及按概率值插入（Insert）调整之后新生成的依存弧。如图5-6所示，归约依存弧R之后，先删除R'，再插入新生成的R''。如果检查未能通过，则检查队列中的下一个元素。如果全部无法归约，则强制归约队列首部元素，直至队列为空，此时整个句子分析完毕。

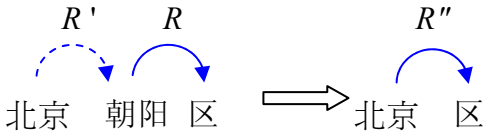


图5-6 调整操作

Figure 5-6 Adjustment

整个解码过程中，检验操作扮演着重要的角色。我们知道，使用动态规划进行解码的一个缺点是很难将各种特征融合到统一的框架中。Collins通过重排序的方法来融合多种特征[61]；Yamada和Nivre等人则借助分类器融合各类特征[56] [83]。本文在使用动态局部优化的方法搜索依存分析树时，是通过“检验”操作来融合多种信息的。“检验”操作既能充分利用各种结构信息，又使这一过程具有良好的扩充性，便于将后续的信息融入到其中。

在本算法中，检验操作使用了支配度和序列依赖作为归约的约束条件。第一，归约时节点的支配度需要在节点的正常范围之内，如图5-1 (a)中，如果归约节点“的”和“上升”，此时“的”的支配度为0，由于它的正常支配度为大于等于1，不满足要求，停止进行归约。第二，检查依存弧的序列依赖情况，图5-6中，待归约的依存弧为R，以下两种情况R将被停止归约：

（1）$P(R') > P(R)$，但R'因未通过检验而被阻塞；

（2）$P(R) > P(R') > \lambda P(R)$，$\lambda$=0.85。

上面是动态局部优化算法的全部过程，该算法具有如下几个优点：

（1） 能够保证最终的分析结果为一棵完整的依存树；

（2） 时间复杂度为$O(n)$；

（3） 对句子中当前概率最大的节点进行归约，而不是从句子的开始，最大限度地避免了错误蔓延问题。

（4） 可在检验过程中融合多种信息，如句法结构，支配度，已生成的子节点等，并可根据语言的特点进行有针对性地增加，使得该算法易于扩充。

5.4.2 算法示例

下面通过一个例子来说明算法的解码过程。设输入的经过分词及词性标注的句子为：

几/m 票/q 可疑/a 的/u 舱单/n 被/p 拎/v 出来/v 。/wp

初始化

首先进行依存树的初始化，计算出相邻节点的依存关系概率，然后根据概率值排序，存入队列中。队列中的依存关系为：

的→舱单	ATT	0.098389
被→拎	ADV	0.121646
几→票	QUN	0.401353
可疑→的	DE	0.406151
拎←出来	CMP	0.764586
票←可疑	RAD	3.66079
舱单→被	SBV	5.61232

初始化之后，依存树的初始状态如图5-7所示。其中，虚线表示未经过检验的依存弧。

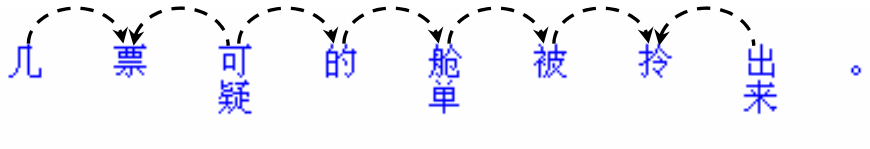


图5-7 依存树的初始状态

Figure 5-7 The initial state of dependency tree

然后，对队列中的元素依次进行处理。

第一遍扫描

Step1：从队列首部开始依次选择最大概率的依存弧，对其进行约束检验。如图5-8所示。

Step2：归约通过检验的依存弧，重新调整队列。包括

- 删除-生成的依存弧（几→票）
- 删除-同叶节点相关的依存弧（无）
- 插入-新的依存弧（无）

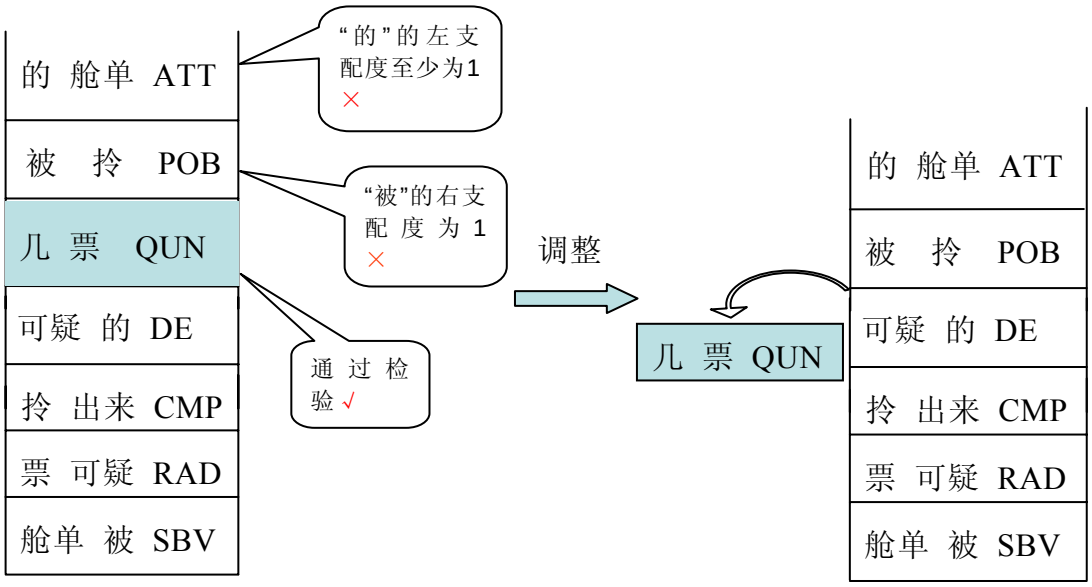


图5-8 队列中元素的检验和调整

Figure 5-8 The actions of check and adjustment to the elements of sequence

第 1 遍扫描后的依存树状态如图 5-9所示。其中，带关系的实线表示检验之后经过归约得到的依存弧。

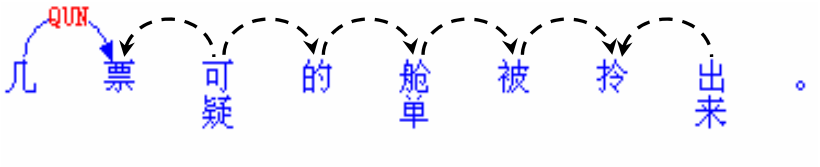


图5-9 第一遍扫描后的依存树状态

Figure 5-9 The state of dependency tree after the first scan

第2遍扫描

Step1：从队列首部开始依次选择最大概率的依存弧，对其进行约束检验。队列首部的依存弧“的→舱单”和“被→拎”仍被不能通过检验，第三个依存弧“可疑→的”通过。如图5-10所示。

Step2：归约通过检验的依存弧，重新调整队列。

- 删除-生成的依存弧（可疑→的）
- 删除-同叶节点相关的依存弧（票←可疑）
- 插入-新的依存弧（票→的）

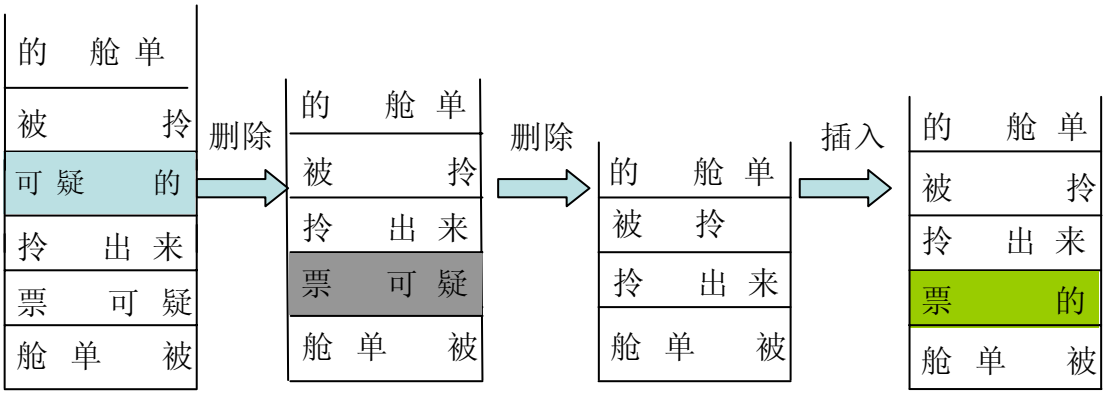


图5-10 队列中元素的检验和调整

Figure 5-10 The actions of check and adjustment to the elements of sequence

第 2 遍扫描后的依存树状态如图 5-11所示。

图5-12 第 2 遍扫描后的依存树状态

Figure 5-11 The state of dependency tree after the second scan

第3遍扫描

Step1：从队列首部开始依次选择最大概率的依存弧，对其进行约束检验。节点“的”的支配度已为1，队列首部的依存弧“的→舱单”通过检验。检验和调整的过程如图5-13所示。

Step2：归约通过检验的依存弧，重新调整队列。

- 删除-生成的依存弧（的→舱单）
- 删除-同叶节点相关的依存弧（票→的）
- 插入-新的依存弧（票→舱单）

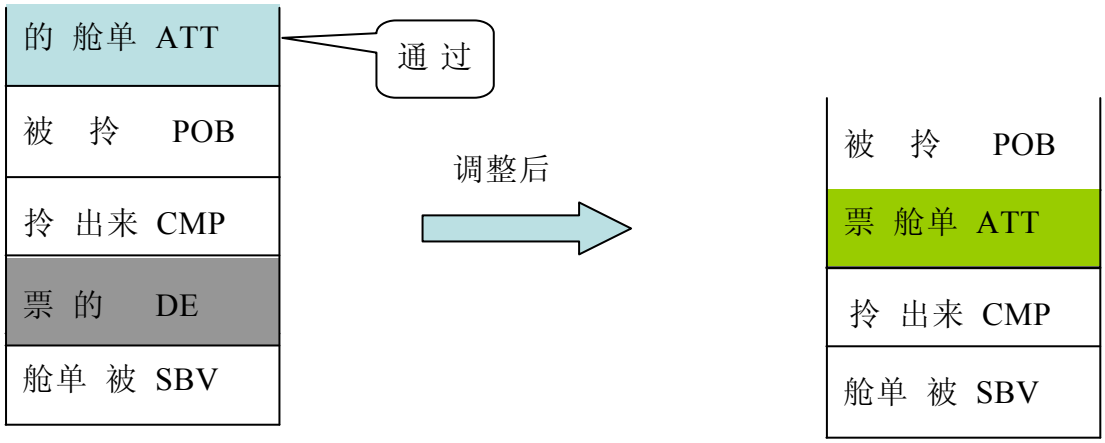


图5-13 队列中元素的检验和调整

Figure 5-12 The actions of check and adjustment to the elements of sequence

第 3 遍扫描后的依存树状态如图 5-14所示。

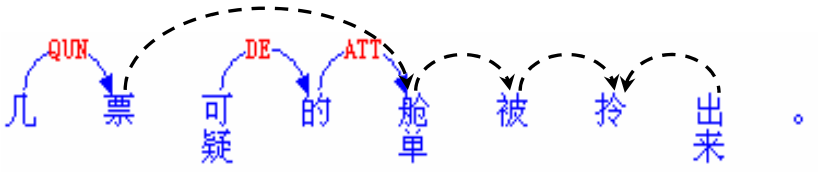


图5-15 第 3 遍扫描后的依存树状态

Figure 5-13 The state of dependency tree after the third scan

第4遍扫描

Step1：从队列首部开始依次选择最大概率的依存弧，对其进行约束检验。依存弧“被→拎”仍未通过检验，“票→舱单”通过检验。

Step2：归约通过检验的依存弧，重新调整队列。

- 删除-生成的依存弧（票→舱单）
- 删除-同叶节点相关的依存弧（无）
- 插入-新的依存弧（无）

第 4 遍扫描后的依存树状态如图 5-16所示。

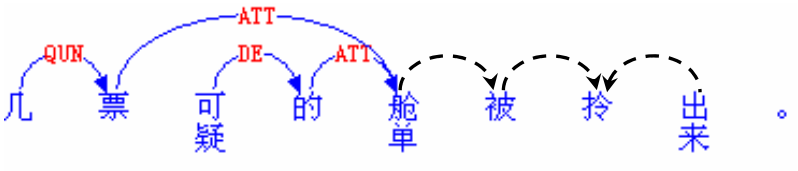


图5-16 第 4 遍扫描后的依存树状态

Figure 5-14 The state of dependency tree after the fourth scan

第5遍扫描

Step1：从队列首部开始依次选择最大概率的依存弧，对其进行约束检验。依存弧“被→拎”仍未通过检验，“拎←出来”通过检验。

Step2：归约通过检验的依存弧，重新调整队列。

- 删除-生成的依存弧（拎←出来）
- 删除-同叶节点相关的依存弧（无）
- 插入-新的依存弧（无）

第 5 遍扫描后的依存树状态如图 5-17所示。

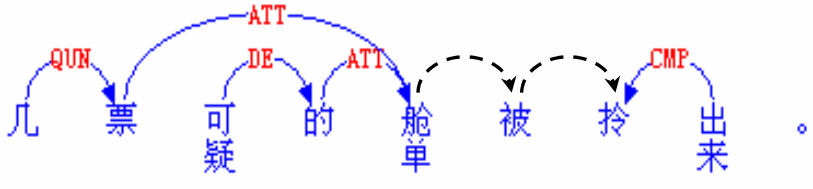


图5-17 第 5 遍扫描后的依存树状态

Figure 5-15 The state of dependency tree after the fifth scan

第6遍扫描

Step1：从队列首部开始依次选择最大概率的依存弧，对其进行约束检验。“被”后面已没有节点，依存弧“被→拎”通过检验。检验和调整的过程如图5-18所示。

Step2：归约通过检验的依存弧，重新调整队列。

- 删除-生成的依存弧（被→拎）
- 删除-同叶节点相关的依存弧（舱单→被）
- 插入-新的依存弧（舱单→拎）

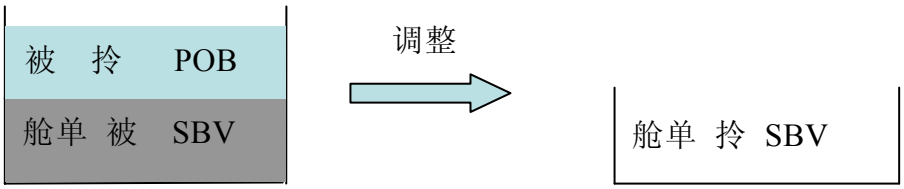


图5-18 队列中元素的检验和调整

Figure 5-16 The actions of check and adjustment to the elements of sequence

第 6 遍扫描后的依存树状态如图 5-19所示。

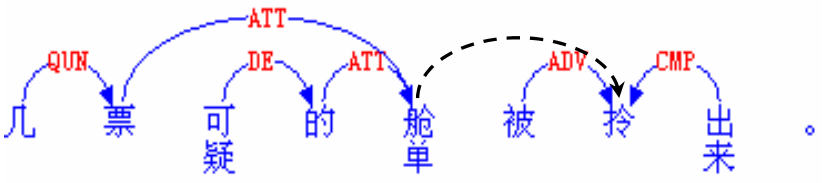


图5-19 第 6 遍扫描后的依存树状态

Figure 5-17 The state of dependency tree after the sixth scan

第7遍扫描

归约最后一个依存弧“舱单→拎”，生成完整的依存树，如图 5-20所示。

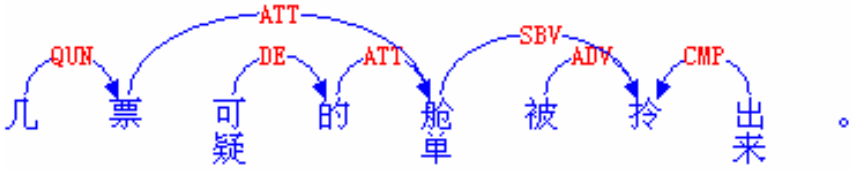


图5-20 第7遍扫描后的依存树状态

Figure 5-18 The state of dependency tree after the eighth scan

5.5 实验结果及分析

实验数据采用《哈工大信息检索研究室汉语依存树库》，该树库共有 212527 个词，1 万个句子，每个句子的平均长度为 21.3 个词。全部的实验中，以前 8000 个句子作为训练数据，8000~9000 句作为开发集，最后 1000 个句子作为测试集。

实验包括对特定语言结构进行分析，以及使用动态局部优化的方法对句子进行依存句法分析。前者的评价指标包括特定结构的准确率、召回率和 F

值，后者的评价指标包括依存关系准确率、依存搭配准确率以及句子核心词的准确率。

5.5.1 特定语言结构分析实验结果

特定语言结构在树库中的比例并不是很高，开发集中共含有依存弧 18145 个，四种特定结构内部所构成依存弧共 656 个，占 3.6%。表 5-1是开发集中各类结构的比例。

表5-1 特定语言结构的内部依存弧比例

Table 5-1 The proportion of dependency arcs of specific structures

	动词习语	简单并列短语	时间及数字短语
数量	92	452	112
比例	0.51%	2.49%	0.62%

虽然特定语言结构所占比例并不是很大，但是这些结构的错误常会引起外部的一系列错误，造成错误的扩散。所以这些结构对整个句法分析的结果有着较大的影响。表 5-2是特定语法结构的识别结果，表 5-3列出了这些特定结构对句法分析的影响。

表5-2 特定结构的识别结果

Table 5-2 The results of specific structures identification

	动词习语	简单并列短语	时间及数字短语
准确率	77.1%	84.2%	89.6%
召回率	77.1%	93.1%	92.8%
F 值	77.1%	88.4%	91.1%

表5-3 特定语法结构对依存句法分析的影响

Table 5-3 The effect of specific structures on the dependency parsing

	依存关系	依存搭配	句子核心词
无特定结构	68.01%	71.44%	67.1%
有特定结构	70.11%	73.39%	68.0%

从表 5-2中能够看到，由于进行了有针对性的识别，这些特定结构的依存分析准确率要高于句法分析的结果，而且这些识别出的结构减少了对后续分析的影响，也减少了后续工作的难度。通过表 5-3的结果对比中也验证了这一点。

5.5.2 句法分析实验结果

除了以上的特定语法结构之外，本章的句法分析中还使用了名词复合短语分析模块（第 3 章）。经过特定结构和名词复合短语分析两个模块的处理之后，将识别的短语核心节点同句子中的其余部分一起进行依存分析，得到完整的依存树。最终的依存分析结果如表 5-4所示。

为了对动态局部优化算法的优势进行对比，表 5-4列出了另一个句法分析器的分析结果，该分析器使用基于全局寻优的动态规划算法进行解码，算法的描述见第 6 章。基于全局寻优的方法实现的句法分析器称为 Parser-1，本节中基于动态局部优化的确定性分析方法实现的句法分析器称为 Parser-2。

表5-4 两个模型的结果对比

Table 5-4 The comparison of two models

	依存关系	依存搭配	句子核心词
Parser-1	64.55%	69.57%	70.10%
Parser-2	70.11%	73.39%	68.0%

从对比结果中能够看出，除了句子核心词之外，Parser-2 的性能比 Parser-1 有了较大的提高，尤其是依存关系的准确提高较大。这是因为 Parser-1 主要利用了依存关系概率，而 Parser-2 的动态局部优化能够融合多种信息，充分利用了结构概率，提高了依存分析的准确率。Parser-1 的实现过程是先分析句子的骨架依存结构，然后再分析每个依存搭配的关系类型，而 Parser-2 将依存搭配和依存关系一体分析，关系和搭配的准确率差距较小。

但是 Parser-2 的句子核心词准确率不如 Parser-1，这和确定性分析中的无法回溯有关。一个句子中常含有多个分句，而每个分句中几乎都有动词，全局寻优的搜索算法可以通过比较选出最合适的句子核心词，而确定性分析算法则无法统一比较，导致核心动词准确率偏低。

5.6 本章小结

本章针对汉语语言中句法结构多样的特点，采用分治策略进行依存句法分析。对树库中存在的一些句法构成方式较为复杂的语言结构，包括简单并列结构、动词习语、时间和数字短语，由于简单的统计学习无法掌握其构成规律，或者句法结构的构成规律较为明显，但是在训练数据中出现的次数并不是很多，本章采用统计结合规则的方法进行处理。之后，短语的核心词同句子的其他成分进行统一分析，并构建了融合多种信息的依存概率模型。

在解码方式上，本章提出了一种基于动态局部优化的搜索算法。该算法动态寻找局部最优解,很好地缓解了传统确定性分析算法中的错误蔓延问题。算法将依存树的生成分为四个操作：检验、归约、插入和删除。其中，检验操作能够方便地融合多种信息，增强模型的判别能力。在检验过程中，算法利用了多种结构信息。首先，本章引入词汇支配度的概念，即一个词支配其从属词的能力。支配度可以限制因词汇的过度拟合而产生的一个节点生成过多的子节点，也可以避免子节点不足的情况，有效地避免了非法结构的生成。其次，本章计算了二元结构概率，用以帮助准确地描述依存关系概率的合理性。另外，根据确定性分析算法的特点，本章考虑了依存结构的序列依赖问题，并将其作为约束条件，检验依存弧的合理性。

本章的算法能够以线性时间完成句子的搜索，并能有效地融合多种结构信息。最后，通过实验检验了本章的分治策略以及搜索算法的有效性，获得了较好的句法分析结果。

第6章 基于分步策略的多语依存分析

6.1 引言

近年来，依存语法在句法分析领域受到了广泛的重视，很多有价值的工作和高性能的依存句法分析性器不断涌现[56, 75] [82]。但这些工作只是针对英语或者其他某一种语言的，在将其方法推广到其他语言的过程中，存在这诸多困难，也使得不同工作之间很难进行直接的比较。

已有一些工作对单语句法分析进行改进和扩展，将其应用到其他语言之中，其中应用最多的是Collins97的句法分析模型[38]，因其在英语中达到了较高的水平，目前已被应用于捷克语[143]、德语[144]、西班牙语[145]、法语[146]等语言之中，并被扩展为多语句法分析的分析引擎[147]。

依存句法分析方面，Kudo实现了一个日语的依存分析器[54]，Yamada对其扩展为英语分析[56]。Nivre的依存分析器已分别应用于瑞典语[82]、英语[83]、捷克语[97]、保加利亚语[148]以及汉语[59]。McDonald的依存分析器则分别对英语[92]、捷克语[93]以及丹麦语[95]进行了分析。

以上这些工作对多语句法分析进行了有益的探索，但是由于树库资源不统一，并缺乏一致的评测集和评价标注，使得多语句法分析的研究还存在许多问题。近几年，随着各种语言的依存树库相继标注完成，很多已有的树库也可转化为依存标注形式，解决了多语分析所需要的训练和评测语料问题。因此，研究者将关注的热点自然地放到建立实用的多语依存句法分析系统之上。

国际会议CoNLL的shared task在2006和2007年已经连续两次举行多语依存分析的任务评测[2, 3]，评测组委会提供十几种语言的依存标注语料，要求参加评测的系统以同一方法训练每一种语言，然后用该系统对所有的语言进行依存分析，并提交结果。两次评测均有多家单位参加，其中的一些系统取得了令人满意的分析结果，表明多语依存分析正在受到越来越多的研究者的重视，在研究方法上也在不断取得进步。

为了探索汉语同其他语言在句法分析上的相同和不同之处，并尝试将汉语的依存分析方法扩展到其他语言之中，本文对多语依存分析进行了研究。

在分析方法上，本文使用一个分步策略，即将依存分析分为两个步骤，首先分析句子的依存骨架结构，然后识别骨架结构中每个搭配的关系类型。在实验中，以CoNLL shared task 2006提供的树库作为训练和测试数据，然后对上一章的方法进行修改，去除模型中语言相关的部分，将其改造为另一个多语依存分析器，同本文的方法进行对比，同时也列出了CoNLL-X shared task 2006的多语依存分析结果作为参考。

6.2 依存骨架分析

依存句法分析有两种策略，一种是一体化分析策略，即同时确定依存搭配及其关系类型，如本文第 5 章的分析方法；另一种是分步策略，即先分析句子的依存骨架结构，然后再分析每个搭配的关系类型。图 6-1为一个依存骨架分析树。

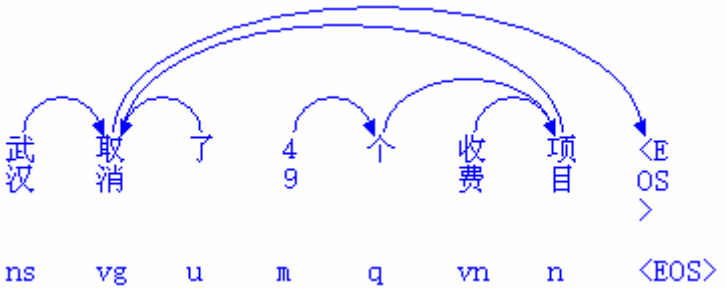


图6-1 依存骨架分析树

Figure 6-1 A skeleton dependency tree

在第一步的骨架分析中，需要完成两个工作：获取依存搭配的概率和设计高效的搜索算法，下面分别对这两个问题进行介绍。

6.2.1 依存搭配概率计算

统计句法分析建模的目的是寻找一个概率评价函数，使其在给定的语法框架下，能够评价某个句法分析结果是当前句子正确语法解释的概率。设 G 为某一种语言的语法，S 为一个句子，T 为根据语法 G 生成 S 的所有可能分析结果所组成的集合。对于任意分析树 $t \in T$，句法分析的概率模型能够计算出 t 的条件概率：

$p(t \mid S,G)$ 且满足 $\sum_{t \in T} p(t|S,G) = 1$

本文的工作是建立一个依存语法的概率分析模型，当输入一个线性的词语序列 *S*，模型输出一棵骨架分析树 *t*，并使树 *t* 符合汉语的依存语法规范。其中，*S* 是经过分词和词性标注的句子，表示为

$$S = \{< w_1，\ t_1>, < w_2，\ t_2>, \ldots , < w_n，\ t_n>\}$$

w_i ($1 \leqslant i \leqslant n$)是句子中第 *i* 个词，t_i ($1 \leqslant i \leqslant n$)是第 *i* 个词的词性。句法分析是一个处理语法歧义的问题，对一个给定的句子，存在数量巨大的符合语法的分析树。可以用概率评价函数对每个分析结果进行评价，把消歧问题转化为一个最优化的过程，即给定句子 *S*，找出一棵概率最大的依存分析树 *t**，该过程可以表示为：

$$\begin{aligned} t^* &= \arg\max_{t \in T} P(t \mid S) \\ &= \arg\max_{t \in T} P(t \mid w_1, w_2, ..., w_n) \end{aligned} \tag{6-1}$$

为了计算分析树 *t* 的概率，对公式（6-1）式做了独立假设，即假设分析树中的各个依存弧相互独立，则整个依存树的概率为 *n*-1 条依存弧的概率乘积，即

$$P(t \mid S) = \prod_{k=1..n-1} P(L_k \mid w_i, w_j) \tag{6-2}$$

公式（6-2）式中，L_k ($1 \leqslant k \leqslant n-1$)是构成 *n* 个节点分析树的依存弧，$w_i$ 和 w_j ($j > i$)是两个词汇节点。除了两端的节点 w_i 和 w_j 之外，依存弧 L_k 还需要另外两个构成要素：*Direction* 和 *Distance*。*Direction* 是依存弧的方向，取 0 和 1 两个值：

$$Direction = \begin{cases} 0 & \text{如果} w_j \text{依存于} w_i \\ 1 & \text{如果} w_i \text{依存于} w_j \end{cases}$$

Distance 是节点 w_i 和 w_j 之间的距离，取 1、2、3 三个值：

$$Distance = \begin{cases} 1 & \text{如果 } j-i=1 \\ 2 & \text{如果 } j-i=2 \\ 3 & \text{如果 } j-i>2 \end{cases}$$

通过对训练数据进行统计，采用极大似然估计计算依存弧 L_k 的概率：

$$\tilde{P}(L_k \mid w_i, w_j) = \frac{C(L_k, w_i, w_j)}{C(w_i, w_j)} \tag{6-3}$$

其中，分子表示节点 w_i 和 w_j 构成依存弧 L_k 的次数，分母表示 w_i 和 w_j 在一个句子中以距离 *Distance* 出现的次数，既包括存在依存关系也包括不存在依存关系的次数。

对词汇关系的稀疏问题，采用词性关系进行插值平滑。上式的词汇依存概率记为 $\tilde{P}_1$，将其中的 w_i 替换成该词的词性 t_i，得到词性依存概率记为 $\tilde{P}_2$，则平滑后依存弧的概率为

$$\tilde{P} = \lambda_1 \tilde{P}_1 + \lambda_2 \tilde{P}_2 + \lambda_3 \xi$$

其中，$\lambda_1+\lambda_2+\lambda_3 = 1$，由训练获得，$\xi$ 是一常量，这里取 0.001。通过平滑，有效地缓解了词汇依存信息的数据稀疏问题。

6.2.2 基于动态规划的搜索算法

句法分析算法是句法分析模型的实现，就是根据句法分析模型对分析结果的概率评价，从全部或部分的分析结果中，找出符合用户要求的结果。本文的问题可归结为在一个有向图中搜索出满足依存文法的最优依存树，类似于 Eisner 的方法[77]，本文采用动态规划的思想，按照自底向上的过程，在 $O(n^3)$时间内从全部分析结果中，搜索出概率值最高的一条路径，以下对算法进行描述。

（1）相关概念的定义

句子的长度为N，由连续i（$1 \leqslant i \leqslant N$)个节点构成的符合依存文法的分析树叫做$i$节点子树，如图6-2(a)和(b)分别为2节点子树和3节点子树。

图6-2 (a) 2 节点子树　　(b) 3 节点子树

Figure 6-2 (a) is a 2-subtree and (b) is a 3-subtree

子树中没有父亲的节点叫做子树的根节点，符合依存文法的每棵子树只有一个根节点，如图6-2 (a)和图6-2(b)中的节点1，分别是对应子树的根节点。

每一棵子树有一个的概率值，叫做子树的概率。

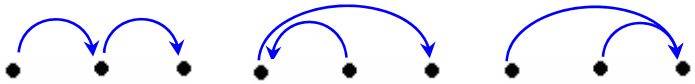
若干节点构成的子树中，根节点相同的子树叫做同根树，则*i*个节点构成*i*组同根树。如图6-3中，由三个节点构成的七棵子树中，每个节点依次作为根节点，构成三组同根树：

图6-3 (a) 以第 1 个节点为根节点的树

Figure 6-3 (a) The trees with the first node as root

(b) 以第 2 个节点为根节点的树

(b) The trees with the second node as root



(c) 以第 3 个节点为根节点的树

(c) The trees with the third node as root

在搜索算法中，对子树有连接和删除两种操作，连接操作是将两棵相邻子树合成一棵新的子树，即通过依存弧将一棵子树的根节点同其相邻子树的根节点进行连接，单个节点记为根节点。对图6-2 (a)和图6-2 (b)的两棵子树通过依存弧a进行连接操作，可得到一棵新的5节点子树，如图6-4所示。

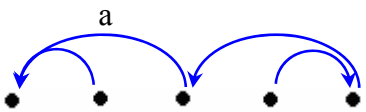


图6-4 子树的连接操作

Figure 6-4 The concatenation of two sub-trees

同根树中，删去其中概率值小的子树，只保留最大概率的子树作为构成上一层子树的中间结果，叫作删除操作。如对图6-3中的三组同根树分别进行删除操作，将在（a）、（b）、（c）中各保留一棵子树，得到以节点1、2和3为根的三棵子树，该操作实质是对中间结果进行无损剪枝，剪枝过程不丢

失最大值信息。

（2）算法描述

该算法采用自底向上的策略，依次生成从2节点到*N*节点的所有子树。在生成*i*节点子树时（共生成*N*-*i*+1次），利用了从1到*i*-1节点的子树进行连接操作，得到*i*组同根树，然后对每组同根树进行删除操作，保留*i*棵子树。由于1到*i*-1节点的所有子树均保留最大概率值，所以能够保证得到*i*节点子树的最大值。以此类推，分析*i*+1、*i*+2直到*N*节点子树，得到根节点依次为由1到*N*的*N*棵树，这*N*棵树中，概率值最大的分析结果即为全局最优的依存树。算法的描述如图6-5所示。

图6-5中，函数Concatenate(begin, i, end) 将节点begin到节点*i*的子树同节点*i*到节点end的子树进行连接操作，生成以*i*为根节点的一组同根树，函数Delete(begin,i,end)对该组同根树进行删除操作，只保留以*i*为根节点中概率值最大的一棵子树。

```
Dependency_Parsing(S)
    for length ← 2 to N
        for begin ← 0 to N - length
        {
            end = begin + length - 1;
            for i ← begin to end
            {
                Concatenate(begin, i, end);
                Delete(begin,i,end);
            }
        }
    Max_tree ← Find maximal tree in N trees;
    Output(Max_tree);
```

图6-5 依存分析的搜索算法

Figure 6-5 The searching algorithm of dependency parsing

6.3 基于 SNoW 的关系识别

6.3.1 SNoW 分类器介绍

SNoW（Sparse Network of Winnows）是建立在预定义或者增量变化的特征空间上的稀疏网络。它是对原始 Winnow 算法的改进，可以高速处理大数据量的多元分类问题。SNoW 的计算性能取决于活跃特征的数量，而不是全部特征空间。另外，它采用多种规则更新方式，包括感知器、Winnow、Bayes 算法等。这些特点保证了 SNoW 算法具有很高的计算效率，这对本文大数据量的多语依存分析非常重要。

SNoW 作为多类分类器，以一组带特征的样例为输入，输出每个样例的标注类别，并且赋予一个置信度值。SNoW 的基本结构框架是一个两层的网络结构，如图 6-6。其中，输入层是特征层，在这一层结点被分配给训练样例的特征。第二层，即目标输出层则对应一个类别标注，它是输入特征的函数。比如，对于一个布尔函数，目标结点只有两个，一个对应正例，另一个对应反例。SNoW 的框架实例可以通过重新定义目标节点数、节点类型、规则更新算法实现。

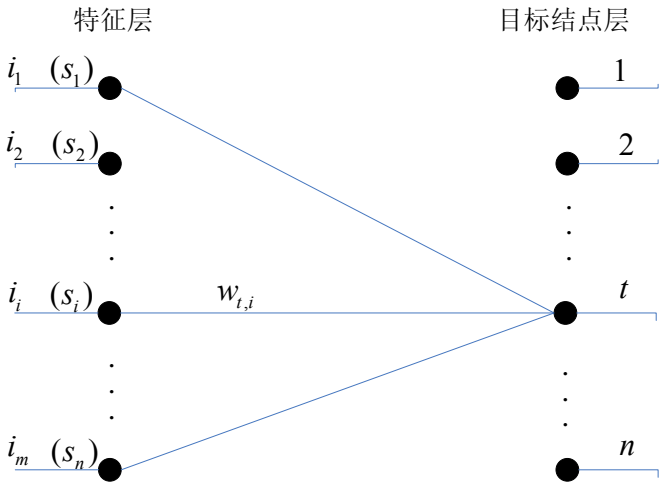


图6-6 SNoW 的网络框架结构图

Figure 6-6 The framework of SNoW

在 SNoW 的输入层，输入的是一组样例。在训练阶段，每个目标节点将利用提供的样例产生一个特征表示函数。在测试阶段，这些函数的表示将用

来预测一个输入样例的类标标注。标准样例的特征可以是布尔值或者实数值，特征空间的每个特征 i 所对应的特征强度为 s_i。在样例里没有出现的特征，其强度值为 0。目标结点是通过带权的边和特征结点连接的。训练过程中，边的连接是动态的，特征 i 和目标结点 t 连接的条件是：当且仅当特征 i 出现在标注为 t 的训练样例中。

设 $A_t=\{i_1,...,i_m\}$ 连接到结点 t 的活跃特征集合，s_i 是样例中特征 i 的强度值，$w_{i,t}$ 是连接第 i 个特征和目标结点 t 的边的权值，θ_t 是结点 t 的阈值。那么结点 t 被预测为正例的条件是：

$$\sum_{i\in A_t} w_{t,i}s_i \geq \theta_t \qquad (6\text{-}4)$$

上式的加权求和也作为目标结点的激活值，用于驱动训练过程中的权值更新和测试过程中的目标预测。设 T 是当前网络结构实例中所有目标结点的集合，对于样例的激活特征集 $A_{t\in T}$，其预测目标 t^* 为：

$$t^*=\arg\max_{t\in T}\sigma(\theta_t,\Omega_t(e)) \qquad (6\text{-}5)$$

其中 $\Omega_t(e)$ 是由公式（6-4）计算的结点 t 对样例 e 的激活值，$\sigma(\theta_t,\Omega_t(e))$ 是一种学习算法函数，一般取 sigmoid 函数，可以把输出转化到 0 到 1 的区间内。

6.3.2 依存关系识别

获得句子的骨架分析结果之后，下一步的工作是判别每个依存搭配的关系类型。本文的方法是依次扫描依存树中的每个依存弧，然后根据依存弧的节点信息、上下文词语以及语法结构，识别当前弧的依存关系类型。

本章将关系识别作为一个多元分类问题，使用上节介绍的 SNoW 分类器进行识别。SNoW 分类器的输入为一组特征向量，本文从依存骨架树中提取的特征如表 6-1所示。

CoNLL-X shared task 提供的数据中，很多树库包含了丰富的信息，如语法、词汇等信息。但由于对其他语言不够了解，本文并没有利用全部信息。本文抽取的特征包括依存关系词汇特征、上下文特征以及结构特征。按照表 6-1列举的特征，图 6-1的骨架分析树中，依存搭配“取消←项目”的特征向量为：

1-取消 2-项目 3-v 4-n 5-vg 6-n 7-n 8-v 9-u 10-EOS 11-1 12-2 13-2 14-0 15-1 16-5

这些特征均是语言无关的，抽取特征之后，针对每种语言分别训练相应的分类器，然后对测试集中的依存骨架结构进行关系标注。

表6-1 关系识别的特征向量

Table 6-1 The feature vector of relation recognization

	特征		特征
1	节点 1 的词形	2	节点 2 的词形
3	节点 1 的粗粒度词性	4	节点 2 的粗粒度词性
5	节点 1 的细粒度词性	6	节点 2 的细粒度词性
7	节点 1 的前一个词性	8	节点 2 的前一个词性
9	节点 1 的后一个词性	10	节点 2 的后一个词性
11	节点 1 的左孩子个数	12	节点 2 的左孩子个数
13	节点 1 的右孩子个数	14	节点 2 的右孩子个数
15	依存弧方向	16	节点间距离

6.4 多语依存分析实验

6.4.1 实验数据

CoNLL-X shared task规定了统一的数据格式：数据文件中句子之间以空格分隔，每个句子由一个或多个符号（tokens）构成，每个符号包含十个域（fields），域之间以空格分隔。表6-2列出了符号的十个域值，其中，测试数据只提供前6项内容，要求评测程序识别第7、8两项内容。

表6-2 CoNLL-X shared task 2006 评测数据格式

Table 6-2 The data format of CoNLL-X shared task 2006

序号	域	含义	序号	域	含义
1	ID	数字标识	6	FEATS	语法/词法特征
2	FORM	词形	7	HEAD	父节点
3	LEMMA	原形/词干	8	DEPREL	依存关系
4	CPOSTAG	词性粗类	9	PHEAD	凸父节点
5	POSTAG	词性细类	10	PDEPREL	凸依存关系

不同的语言相应的域值相差比较大，图6-7是Sinica树库中的一个句子表示形式。

1	他	_	N	Nhaa	_	2	agent	_	_
2	穿	_	V	VC2	_	0	ROOT	_	_
3	著	_	DE	Di	_	2	aspect	_	_
4	破舊	_	V	VH11	_	5	head	_	_
5	的	_	DE	DE	_	6	property	_	_
6	上衣	_	N	Nab	_	2	goal	_	_

图6-7 多语依存分析数据例子

Figure 6-7 An example of multilingual dependency parsing data

图6-7中，下划线表示该域的值为空。可以看到，汉语没有词干信息。另外，该树库也没有标注词法/语法特征。9、10两个域的值在全部13个语言中均为空。

对多语依存分析的评价，CoNLL-X shared task 2006 制订了统一的标准，包括三个评价指标：

（1）父节点-关系得分(labeled attachment score, LAS)。正确识别了某一节点的父节点以及和父节点之间的关系标记，这样的节点所占的比例为父节点-关系得分。

（2）父节点得分(unlabeled attachment score, UAS)。正确识别了某一节点的父节点，但和父节点之间的关系类型未必正确，这样的节点所占的比例为父节点得分。

（3）关系标记准确率(label accuracy, LA)：正确识别了某一节点和父节点之间的关系标记，但父节点未必正确，这样的节点所占的比例为关系标记准确率。

2006 年的评测任务中，标点不作为有效符号参与评价指标的计算，而2007 年的评测则将标点也计算在内。

6.4.2 实验结果

本实验中，关系类型识别所使用的分类器是 SNoW 工具包，由 UIUC 大学的 Roth 等人开发并已共享[1]。使用的版本为 SNoW+(Version 3.2.0)，其参数

[1] http://l2r.cs.uiuc.edu/ danr/snow.html

选择为：Winnow: (1.35, 0.8, 1, 0.3159)。

按照 6.3.2 节中所抽取的特征向量，用 SNoW 分类器对 13 中语言的关系类型进行了识别，识别的准确率如表 6-3所示。

表6-3 关系识别结果

Table 6-3 The results of relation recognization

语言	准确率	语言	准确率	语言	准确率
Arabic	0.7700	Dutch	0.8648	Spanish	0.8514
Bulgarian	0.8847	German	0.896	Swedish	0.8494
Chinese	0.8595	Japanese	0.9571	Turkish	0.7414
Czech	0.7772	Portuguese	0.8894	-	-
Danish	0.7925	Slovene	0.7872	-	-

获得依存关系之后，对完整的依存分析结果进行评价，评价标准为 CoNLL-X shared task 提供的前两个评价标准：父节点-关系得分和父节点得分。

为了对多语依存分析的效果进行比较，本文对第 5 章的汉语句法分析方法进行修改，构建了一个多语依存分析器。主要修改方式是去除模型中的语言相关部分，包括各种特定语言结构的处理。修改后的句法分析器作为 Parser-1，本文的多语分析器作为 Parser-2。作为参考，同时列出了参加 CoNLL-X shared task 全部系统的平均结果，全部结果如表 6-4所示。

表 6-4的结果中，第一个结果是参加评测的结果，第二个是本实验的结果，第三个是 CoNLL-X shared task 的平均结果。其中黑体字为 Parser-2 中的高于 Parser-1 的分析结果。

Parser-1 和 Parser-2 是两种有代表性的多语依存分析方法，在分析结果上，Parser-2 的 LAS 有 4 种语言好于 Parser-1，UAS 有 6 种语言好于 Parer-1。通过观察表 6-4，可以发现两种方法的分析结果有如下特点：

（1）Parser-1 的结果同 CoNLL 评测结果的变化趋势基本一致，而 Parser-2 结果在变化趋势有一些差异，如 Danish、Japanese 等。

（2）Parser-2 的 LAS 和 UAS 结果相差要大一些，主要是因为很多语言的关系类型较多，如表 6-5所示，导致分类器的效果下降。

表6-4 多语依存分析结果

Table 6-4 The results of multilingual dependency parsing

Language	Parser-1		Parser-2		CoNLL	
	LAS	UAS	LAS	UAS	LAS	UAS
Arabic	0.5074	0.6479	0.3893	0.4892	0.5994	0.7348
Bulgarian	0.6764	0.7397	**0.7133**	**0.7858**	0.7998	0.8589
Chinese	0.7529	0.7990	0.7186	**0.8264**	0.7832	0.8485
Czech	0.5852	0.6814	0.5079	0.6287	0.6717	0.7701
Danish	0.7770	0.7990	0.7212	**0.9125**	0.7616	0.8452
Dutch	0.5936	0.6407	0.5649	0.6316	0.7073	0.7507
German	0.6811	0.7300	**0.6835**	**0.7453**	0.7858	0.8260
Japanese	0.7084	0.7264	**0.8413**	**0.8681**	0.8586	0.8905
Portuguese	0.7113	0.7710	0.6466	0.7039	0.8063	0.8646
Slovene	0.5721	0.6894	0.5143	0.6406	0.6516	0.7653
Spanish	0.6508	0.7007	0.6120	0.6963	0.7352	0.7776
Swedish	0.6383	0.7319	0.5891	0.6734	0.7644	0.8421
Turkish	0.4172	0.5690	**0.4534**	**0.5777**	0.5595	0.6935

通过对结果的比较，可以看到，多语依存分析的结果在不同语言上相差很大，在同一种语言上，不同的依存分析器也得到了不同的结果。如能够发现导致这些差别的因素，对多语依存分析将会有很大的帮助。但是无论从本文的分析结果还是 CoNLL 的评测结果，均无法发现影响多语依存分析准确率的确定因素。如表 6-5，无论是训练数据规模最大的捷克语树库，还是句子长度最短的中文树库，均没有取得最高的分析结果。我们只能认为没有哪一种因素能够对依存分析起决定性的影响，只是由多种因素共同发挥作用。

本文对不同语言的语法特点进行统计分析，试图解释多语依存分析结果相差很大的原因。第一个特点是句子的根节点数，多数语言中每个句子中只有一个根节点，但一些语言有多个根节点，表 6-5中列出了每种语言中最大根节点的数量。另一个重要的语法特点是依存关系的方向性，树库中的依存

关系有的只有一个固定的方向，有的既可以指向左，也可以指向右。具有两个方向的关系实际增加了关系的数量，这样的关系和树库的标注规范有关。表 6-5列出了每个树库中有这类关系的数量。另外，一些语言的语法结构非常灵活，且同其他语言相差很大，增加了语言建模的难度。例如有些语言中，主语可以位于在谓语的任意一侧，容易和宾语于发生混淆[149] [19, 150]。

表6-5 多语依存分析的训练数据统计（规模列表示训练数据的词汇量，单位为 1000 词；长度列表示句子的平均词长；数量列表示关系类型的数量，左侧数字为类型的总个数，右侧数字为其中具有双重方向性的关系类型数量；核心列表示每个句子中核心词（即根节点）的最大个数。）

Table 6-5 The statistics of training data in the multilingual dependency parsing

语言	规模	长度	数量	核心	语言	规模	长度	数量	核心
Ar	54	37.2	27/24	17	Ja	151	8.9	7/2	14
Bu	190	14.8	19/34	1	Po	207	22.8	55/40	1
Ch	337	5.9	78/55	1	Sl	29	18.7	26/23	11
Cz	1249	17.2	82/72	28	Sp	89	27.0	21/19	1
Da	94	18.2	54/24	4	Sw	191	17.3	64/54	1
Du	195	14.6	26/17	9	Tu	58	11.5	26/23	5
Ge	700	17.8	46/40	1	-	-	-	-	-

本文的多语依存分析结果中，多数语言都低于 CoNLL-X shared task 评测系统的平均值，其原因主要有如下几方面：

（1）13 种语言中，多数语言都存在交叉结构（non-projective structures），而汉语的依存结构不存在交叉弧现象。我们缺乏交叉语言的背景知识，并且本文的句法分析方法是针对非交叉语言（projective languages）的，无法处理语言中的交叉结构。

（2)树库中所提供的标注信息在本文中并没有全部利用。例如词形变化信息、语法/词法特征等。这主要是因为汉语中没有此类信息，而我们缺乏对其他语言中此类信息的理解，所以暂时没有使用。

（3）本文还没有深入挖掘语言中的结构信息，使本文方法的优势没有发挥。

6.5 本章小结

在前几章汉语依存句法分析的基础之上，本章探索了多语依存分析技术，使用了一种基于分步策略的多语依存分析方法。该方法首先建立一个语言无关的依存概率模型，主要利用了树库中的词汇和词性信息，然后使用基于动态规划的全局寻优算法对句子进行解码，获得句子的依存骨架分析树。第二步针对骨架分析的结果，识别每个依存弧的关系类型。本章使用基于 SNoW 的方法对关系进行分类，首先依次扫描依存树中的每个依存弧，然后根据依存弧的节点信息、上下文词语以及语法结构，识别当前弧的依存关系类型。

最后给出了本章的多语依存分析方法在 CoNLL-X shared task 2006 数据集上的分析结果。作为比较，本文对第 5 章的汉语句法分析方法进行修改，除去模型中的语言相关部分，构建了一个多语依存分析器，对同一数据集进行了分析。结合参加 CoNLL-X shared task 的系统分析结果，本章对多语依存分析存在的问题进行了分析。

结论

对语言的深层处理过程中，句法分析处于一个瓶颈位置。如能将其有效的加以解决，一方面是对相应的语法理论的验证，能够证明某一语法理论的正确性和有效性，促进语法理论的研究和发展，为人类掌握语言的规律提供实践性的检验。另一方面可以作为自然语言处理技术的一个基础，为语言的深层理解提供一个新的平台，有效支撑各种语义、语用等分析技术。也可以直接对各种上层应用，比如机器翻译、信息抽取、自动文摘等产生帮助。

本文针对汉语的特点，全面深入地研究了汉语依存句法分析中的几项关键技术，包括词法分析、短语分析以及句子片段分析。通过对依存树库的统计学习，本文对这几个问题分别给出了相应的解决办法。具体地讲：本论文的贡献主要表现在以下几个方面：

1、提出了一种基于最大熵的动词子类标注方法。该方法首先将动词分为 8 个子类，然后根据句子上下文信息判别动词的子类。子类标注有效地缓解动词引起的句法歧义，改善了依存句法分析的性能。

2、提出了一种基于隐马尔科夫树模型的名词复合短语分析方法。该方法将名词复合短语的边界识别和语法分析进行一体化处理。由于隐马尔可夫树模型能够把短语的边界信息和语法结构统一在一棵树上，避免了分步处理时所存在的问题，使复合短语的边界识别和结构分析均达到了较高的准确率。

3、提出了一种面向句法分析的句子片段识别方法。针对自然语言句子长度大、语法结构复杂的问题，本文先将句子划分为多个片断，通过片断分割来缩减句子的长度，从而简化语法结构。对每个片断进行依存分析之后，再识别各片断之间的依存关系，最后将各个片断组合为一个完整的分析树。

4、提出了一种基于动态局部优化的依存分析搜索算法。该算法动态寻找局部最优解，很好地缓解了以往确定性分析算法中的错误蔓延问题。算法在解码过程中能够方便地融合多种结构信息，增强模型的判别能力。在模型中，本文引入词汇支配度的概念，以限制因词汇的过度拟合而产生的非法结构。其次，计算了二元结构概率，用以帮助准确地描述依存关系概率的合理性。另外，根据确定性分析算法的特点，本文考虑了依存结构的序列依赖问题，将其作为约束条件，检验依存弧的合理性。

本文的研究内容较好地解决了汉语依存句法分析在不同层面所存在的

一些问题。对以下几个问题，还需要做进一步的研究：

1、句法分析的概率模型在特征融合上还有待进一步加强。理想的概率模型能够有效利用树库中标注的各种信息，并能同搜索算法结合在一起，不会增加搜索算法的复杂度。

2、需要进一步挖掘句法的结构信息。在树库规模受限的情况下，启发式的语法知识对句法分析至关重要。下一步将探索以自动或人工的方式发现语言中的规律以及句法的结构特征，改善句法分析的效果。

3、多语依存分析是一个很有发展前景的研究方向，本文在这方面研究得还不够深入。下一步工作将探索多语依存分析的特点和规律，将已有的汉语依存分析方法进行合理的改造和扩展，使汉语和多语的依存分析很好地融合在一起。

参考文献

1. J. Allen. Natural Language Understanding (Second Edition): The Benjamin / Cummings Publishing Company, Inc. 1995
2. B. Sabine and M. Erwin. CoNLL-X shared task on Multilingual Dependency Parsing. Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL-X), New York City. 2006: 149-164
3. J. Nivre, J. Hall, S. Kubler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret. The CoNLL 2007 Shared Task on Dependency Parsing. Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL, Prague. 2007: 915-932
4. 文勖，张宇，刘挺. 类别主特征结合句法特征的中文问题层次分类. 第二届全国信息检索与内容安全学术会议, 上海. 2005: 211-220
5. J.Cowie and W.Lehnert. Information extraction. Commun. ACM, 1996, 39(1): 80-91
6. M. Surdeanu, S. Harabagiu, and J. Williams. Using Predicate-Argument Structures for Information Extraction. Proceedings of the ACL. 2003: 8-15
7. H.P. Zhang, T. Liu, J.S. Ma, and X.T. Liao. Chinese Word Segmentation with Multiple Postprocessors in HIT-IRLab. SIGHAN. 2005: 172-175
8. X.Q. Luo. A Maximum Entropy Chinese Character-Based Parser. Proceedings of Conference on Empirical Methods in NLP. 2003: 192-199
9. P. Fung, G. Ngai, Y. Yang, and B. Chen. A Maximum Entropy Chinese Parser Augmented with Transformation-Based Learning. ACM Transactions on Asian Language Information Processing, 2004, 3(3): 159-168
10. 卢志茂，刘挺，李生. 统计词义消歧的研究进展. 电子学报, 2006, 34(2): 333-343
11. 卢志茂，刘挺，张刚，李生. 基于依存分析改进贝叶斯模型的词义消歧. 高技术通讯, 2003, 13(5): 1-7
12. D.L. Waltz. An English language question answering system for a large

relational database. Commun. ACM, 1978, 21(7): 526-539

13. T. Liu, W.X. Che, S. Li, Y.X. Hu, and H.J. Liu. Semantic role labeling system using maximum entropy classifier. CoNLL2005, Ann Arbor, Michigan. 2005: 189-192
14. 周强. 基于语料库和面向统计学的自然语言处理技术介绍. 计算机科学, 1995, 22(4): 36-40
15. M. Marcus, B. Santorini, and M. Marcinkiewicz. Building a Large Annotated Corpus of English: the Penn Treebank. Computational Linguistics, 1993, 19(2): 313-330
16. S. Brants, S. Dipper, S. Hansen, W. Lezius, and G. Smith. The TIGER Treebank. Proc. of the First Workshop on Treebanks and Linguistic Theories (TLT). 2002: 24–42
17. K. Sima'an, A. Itai, Y. Winter, A. Altman, and N. Nativ. Building a Tree-Bank of Modern Hebrew Text. Journal Traitement Automatique des Langues (t.a.l.) -- Special Issue on Natural Language Processing and Corpus Linguistics, 2001, 42(2): 347-380
18. Y. Kawata and J. Bartels. Stylebook for the Japanese Treebank in VERBMOBIL. 2000
19. M.C. Torruella and M.A.M. Antonín. Design Principles for a Spanish Treebank. Proc. of the First Workshop on Treebanks and Linguistic Theories (TLT). 2002: 1698–1703
20. K. Chen, C. Luo, M. Chang, F. Chen, C. Chen, C. Huang, and Z. Gao. Sinica Treebank: Design Criteria, Representational Issues and Implementation. Treebanks: Building and Using Parsed Corpora, ed. A. Abeill. Vol. 20, Dordrecht. 2003: 231-248
21. N.W. Xue, F. Xia, F.D. Chiou, and M. Palmer. The Penn Chinese Treebank: Phrase Structure Annotation of a Large Corpus. Natural Language Engineering, 2004, 10(4): 1-30
22. 周强, 汉语树库构建--技术报告, 清华大学计算机系智能技术与系统国家重点实验室. 2003
23. J. Hajic. Building a Syntactically Annotated Corpus: The Prague Dependency Treebank. Issues of Valency and Meaning, 1998: 106-132
24. T.H. King, R. Crouch, S. Riezler, M. Dalrymple, and R. Kaplan. The

PARC700 dependency bank. Proceedings of the EACL03: 4th International Workshop on Linguistically Interpreted Corpora (LINC-03). 2003: 1-8

25. I. Boguslavsky, S. Grigorieva, N. Grigoriev, L. Kreidlin, and N. Frid. Dependency Treebank for Russian: Concept, Tools, Types of Information. The 18th International Conference on Computational Linguistics (COLING). 2000: 987-991
26. C. Bosco and V. Lombardo. Dependency and relational structure in treebank annotation. Workshop on Recent Advances in Dependency Grammar. 2004: 1-8
27. T.B.Y. Lai and C.N. Huang. Dependency-based Syntactic Analysis of Chinese and Annotation of Parsed Corpus. the 38th Annual Meeting of the Association for Computational Linguistics. 2000: 255-262
28. F. Pereira and Y. Schabes. Inside-Outside Reestimation from Partially Bracketed Corpora. the 30th Annual Meeting of the Association for Computational Linguistics. 1992: 128-135
29. D. Magerman and M. Marcus. Pearl: A Probabilistic Chart Parser. Proc. of the 1991 European ACL Conference, Berlin, Germany. 1991: 15-20
30. T. Briscoe and J. Carroll. Generalized LR Parsing of Natural Language (Corpora) with Unification-Based Grammars. Computational Linguistics, 1993, 19(1): 25-60
31. D. Klein and C. Manning. Accurate Unlexicalized Parsing. the 41th Association for Computational Linguistics. 2003: 423-430
32. 周强，黄昌宁. 基于局部优先的汉语句法分析方法. 软件学报, 1999, 10(1): 1-6
33. J.S. Ma, Y. Zhang, T. Liu, and S. Li. A statistical dependency parser of Chinese under small training data. Workshop: Beyond shallow analyses - Formalisms and statistical modeling for deep analyses, IJCNLP-04, San Ya. 2004
34. D. Hindle and M. Rooth. Structural Ambiguity and Lexical Relations. Computational Linguistics, 1993, 19(1): 103-120
35. E. Jelinek, J. Lafferty, D. Magerman, R. Mercer, A. Ratnaparkhi, and S. Roukos. Decision Tree Parsing using a Hidden Derivation Model.

Proceedings of the Human Language Technology Workshop, Plainsboro, New Jersey. 1994: 272-277

36. D. Magerman. Statistical Decision-Tree Models for Parsing. Proc. of the 33rd Annual Meeting of the ACL. 1995: 276-283
37. M. Collins. A Statistical Dependency Parser Of Chinese Under Small Training Data. Proc. of the 34th Annual Meeting of the ACL. 1996: 184-191
38. M. Collins. Three Generative, Lexicalized Models for Statistical Parsing. Proceedings of the 35th annual meeting of the association for computational linguistics. 1997: 16–23
39. M. Collins. Head-Driven Statistical Models for Natural Language Parsing. Ph. D.: University of Pennsylvania. 1999: 1-235
40. E. Charniak. A Maximum-Entropy-Inspired Parser. Proceedings of the First Annual Meeting of the North American Association for Computational Linguistics, Seattle, WA. 2000: 132–139
41. D. Chiang and D.M. Bikel. Recovering Latent Information in Treebanks. Proceedings of the 19th International Conference on Computational Linguistics, Taipei. 2002: 183-189
42. M. Johnson. PCFG models of linguistic tree representations. Computational Linguistics, 1998, 2(4): 613-632
43. R. Levy and C. Manning. Is it Harder to Parse Chinese, or the Chinese Treebank? Proceedings of the 41th Association for Computational Linguistics. 2003: 439-446
44. 孙宏林. 现代汉语非受限文本的实语块分析. 北京大学博士研究生学位论文. 2001
45. 吴云芳. 体词性并列结构的结构平行. 全国第七届计算语言学联合学术会议. 2003: 141-146
46. K.J. Chen and Y.M. Hsieh. Chinese Treebanks and Grammar Extraction. Proceedings of the 1st International Joint Conference of NLP, Sanya China. 2004: 655-663
47. X.W. Han and T.J. Zhao. FML-Based SCF predefinition learning for Chinese verbs. Proceedings of the 1st International Joint Conference of NLP, Sanya China. 2004: 115-122

48. A. McCallum, D. Freitag, and F. Pereira. Maximum Entropy Markov Models for Information Extraction and Segmentation. Proceedings of ICML. 2000: 591-598
49. J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. Proceedings of 18th International Conference on Machine Learning 2001: 282-289
50. M. Johnson. Joint and Conditional Estimation of Tagging and Parsing Models. Proceedings of ACL. 2001: 314-321
51. A. Ratnaparkhi. Learning to Parse Natural Language with Maximum Entropy Models. Machine Learning, 1999, 34(1-3): 151-175
52. V.N. Vapnik. The Nature of Statistical Learning Theory, Berlin: Springer-Verlag. 1995
53. T. Joachims. Text categorization with support vector machines: learning with many relevant features. Proceedings of the 10th European Conference on Machine Learning. 1998: 137-142
54. T. Kudo and Y. Matsumoto. Japanese dependency structure analysis based on support vector machines. Proceedings of Empirical Methods in Natural Language Processing and Very Large Corpora, Hong Kong. 2000: 18-25
55. T. Kudo and Y. Matsumoto. Japanese Dependency Analysis using Cascaded Chunking. The 6th Conference on Natural Language Learning. 2002: 63-69
56. H. Yamada and Y. Matsumoto. Statistical Dependency Analysis with Support Vector Machines. Proc. of the 8th Intern. Workshop on Parsing Technologies (IWPT). 2003: 195-206
57. M.X. Jin, M.Y. Kim, and J.H. Lee. Two-Phase Shift-Reduce Deterministic Dependency Parser of Chinese. Proc. of IJCNLP: Companion Volume including Posters/Demos and tutorial abstracts. 2005
58. Y.C. Cheng, M. Asahara, and Y. Matsumoto. Deterministic dependency structure analyzer for Chinese. Proceedings of International Joint Conference of NLP. 2004: 500-508
59. Y.C. Cheng, M. Asahara, and Y. Matsumoto. Chinese Deterministic Dependency Analyzer: Examining Effects of Global Features and Root

Node Finder. Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing. 2005

60. J.R. Quinlan. Induction of Decision Trees. Machine Learning, 1986, 1(1): 81-106
61. M. Collins. Discriminative Reranking for Natural Language Parsing. Proceedings of the 17th International Conference on Machine Learning. 2000: 175-182
62. M. Collins and N. Duffy. New Ranking Algorithms for Parsing and Tagging: Kernels over Discrete Structures, and the Voted Perceptron. Proceedings of the 30th Annual Meeting of the ACL. 2002: 263-270
63. E. Charniak and M. Johnson. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. Proceedings of ACL 2005: 173-180
64. D. Yuret. Discovery of linguistic relations using lexical attraction. Ph. D. Thesis: MIT. 1998
65. 周强，黄昌宁. 汉语概率型上下文无关语法的自动推导. 计算机学报, 1998, 21(5): 387-392
66. A. Sarkar. Applying Co-Training Methods to Statistical Parsing. Second Meeting of the North American Chapter of the Association for Computational Linguistics, Pittsburgh, PA. 2001: 175-182
67. J.F. Gao and H. Suzuki. Unsupervised Learning of Dependency Structure for Language Modeling. Proceedings of the 41th ACL. 2003: 521-528
68. D. Klein. The Unsupervised Learning of Natural Language Structure. Ph. D. Thesis: Stanford University. 2005
69. M. Chitrao and R. Grishman. Statistical Parsing of Messages. Proceedings Speech and Natural Language Workshop: Morgan Kaufman Publishers. 1990: 263-266
70. P. Pantel and D.K. Lin. An Unsupervised Approach to Prepositional Phrase Attachment using Contextually Similar Words. Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics. 2000: 101-108
71. M. Tomita. A Probabilistic Parsing Method for Sentence Disambiguation. Current Issues in Parsing Technology, ed. M. Tomita, Boston, MA: Kluwer Academic Publishers. 1991: 139-152

72. A.V.A.a. J.D.Ullman. The Theory of Parsing, Translation and Compiling: Prentice-Hall. 1972
73. J.E.P.D. Thesis. An efficient Context-free Parsing Algorithm: Carnegie-Mellon Univ. 1968: 94-102
74. 孟遥. 基于最大熵的全局寻优的汉语句法分析模型和算法研究. 博士学位论文, 哈尔滨工业大学. 2003
75. J. Eisner. Three New Probabilistic Models for Dependency Parsing: An Exploration. Proc. of the 16th Intern. Conf. on Computational Linguistics (COLING). 1996: 340--345
76. J. Eisner, An Empirical Comparison of Probability Models for Dependency Grammar, University of Pennsylvania. 1996: 1-18
77. J. Eisner. Bilexical grammars and a cubic-time probabilistic parser. Proceedings of the International Workshop on Parsing Technologies, MIT. 1997: 54-65
78. T.B.Y. Lai, C.N. Huang, M. Zhou, J.B. Miao, and K.C. Siu. Span-based Statistical Dependency Parsing of Chinese. Proc. NLPRS. 2001: 677-684
79. A. Ratnaparkhi. A Linear Observed Time Statistical Parser Based on Maximum Entropy Models. Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing, Brown University, Providence, Rhode Island. 1997: 1-10
80. A. Harri. On parsing binary dependency structures deterministically in linear time. Workshop on dependency-based grammars, COLING-ACL'98, Montreal. 1998: 68-77
81. J. Nivre. An Efficient Algorithm for Projective Dependency Parsing. Proceedings of the 8th International Workshop on Parsing Technologies (IWPT), Nancy, France. 2003: 149-160
82. J. Nivre, J. Hall, and J. Nilsson. Memory-Based Dependency Parsing. Proc. of the Eighth Conf. on Computational Natural Language Learning (CoNLL). 2004: 49-56
83. J. Nivre and M. Scholz. Deterministic Dependency Parsing of English Text. Proc. of the 20th Intern. Conf. on Computational Linguistics (COLING). 2004: 64-70
84. I.A. Melchuk. Dependency Syntax: Theory and Practice, Albany: State

University Press of New York. 1988

85. C. Samuelsson. A Statistical Theory of Dependency Syntax. COLING 2000. 2000: 684-690
86. A. Dikovsky. Grammars for Local and Long Dependencies. Proceedings of the 39th International Conference of ACL, Morgan Kaufman. 2001: 156-163
87. G. Infante-Lopez, M.d. Rijke, and K. Sima'an. A General Probabilistic Model for Dependency Parsing. proceedings of the BNAIC 2002, Leuven, Belgium. 2002: 139-146
88. T. By. English Dependency Grammar. Workshop on Recent Advances in Dependency Grammar, COLING2004. 2004: 64-69
89. A. Nasr and O. Rambow. A Simple String-Rewriting Formalism for Dependency Grammar. Workshop on Recent Advances in Dependency Grammar, COLING2004. 2004: 17-24
90. J. Nivre. Inductive Dependency Parsing: Springer Verlag. 2006
91. H. Isozaki, H. Kazawa, and T. Hirao. A Deterministic Word Dependency Analyzer Enhanced With Preference Learning. COLING-2004. 2004: 275-281
92. R. McDonald, K. Crammer, and F. Pereira. Online Large-Margin Training of Dependency Parsers. Proc. of the 43rd Annual Meeting of the ACL. 2005: 91–98
93. R. McDonald, F. Pereira, K. Ribarov, and J. Hajic. Non-Projective Dependency Parsing using Spanning Tree Algorithms. Joint Conference on Human Language Technologies and Empirical Methods in Natural Language Processing. 2005: 523-530
94. M. Zhou. A block-based dependency parser for unrestricted Chinese text. Proc. of the 2nd Chinese Language Processing Workshop Attached to ACL-2000, Hong Kong. 2000: 78-84
95. R. McDonald and F. Pereira. Online Learning of Approximate Dependency Parsing Algorithms. 11th Conference of the European Chapter of the Association for Computational Linguistics: EACL. 2006: 81-88
96. S. Marinov and J. Nivre. A Data-Driven Parser for Bulgarian. Proceedings

of the Fourth Workshop on Treebanks and Linguistic Theories, Barcelona. 2005: 89-100

97. J. Nivre and J. Nilsson. Pseudo-Projective Dependency Parsing. Proc. of the 43rd Annual Meeting of the ACL. 2005: 99-106
98. R. McDonald, K. Lerman, and F. Pereira. Multilingual dependency analysis with a two-stage discriminative parser. Proceedings of the 10th Conference on Computational Natural Language Learning, New York City. 2006
99. J. Nivre, J. Hall, J. Nilsson, G. Eryigit, and S. Marinov. Labeled Pseudo-Projective Dependency Parsing with Support Vector Machines. Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL). 2006: 221-225
100. S. Riedel, R. Cakıcı, and I. Meza-Ruiz. Multi-lingual dependency parsing with incremental integer linear programming. Proceedings of the 10th Conference on Computational Natural Language Learning. 2006
101. M. Kuhlmann and J. Nivre. Mildly Non-Projective Dependency Structures. Proceedings of COLING-ACL (Companion Volume), Sydney, Australia. 2006: 507-514
102. J. Nivre. Constraints on Non-Projective Dependency Parsing. Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL). 2006: 73-80
103. 孙茂松，邹家彦. 汉语自动分词研究评述. 当代语言学, 2001, 3(1): 22-32
104. X. Luo, M.S. Sun, and B.K. Tsou. Covering Ambiguity Resolution in Chinese Word Segmentation Based on Contextual Information. Proceedings of COLING2002, Taiwan. 2002: 598-604
105. J. Gao, M. Li, and C.N. Huang. Improved source-channel models for Chinese word segmentation. Proceedings of ACL, Sapporo, Japan. 2003: 7-12
106. G. Roger, L. Geoffrey, and S. Geoffrey. The Computational Analysis of English: A Corpus-based Approach, London: Longman Inc. 1987: chapters 1-3
107. 俞士汶，段慧明，朱学锋，孙斌，常宝宝. 北大语料库加工规范：切

分• 词性标注• 注音. Journal of Chinese Language and Computing, 2003, 13(2): 121-158

108. T. Niesler. Category-based statistical language models. Ph.D., London: University of Cambridge. 1997
109. G. Andrew, T. Grenager, and C. Manning. Verb sense and subcategorization: Using joint inference to improve performance on complementary tasks. Proceedings of Empirical Methods in Natural Language Processing, Barcelona, Spain. 2004: 150-157
110. A.D. Beale. Lexicon and grammar in probabilistic tagging of written English. Proceedings of the 26th conference on Association for Computational Linguistics, Buffalo, New York. 1988: 211-216
111. N.W. Xue and M. Palmer. Automatic Semantic Role Labeling for Chinese Verbs. Proceedings of the 19th International Joint Conference on Artificial Intelligence, Edinburgh, Scotland. 2005: 1160-1165
112. J.N. Darroch and D. Ratcliff. Generalized iterative scaling for log-linear models. Annals of Mathematical statistics, 1972, 43(5): 1470-1480
113. M. Lauer. Designing Statistical Language Learners: Experiments on Noun Compounds. Ph.D., Australia: Macquarie University. 1995
114. T. Tanaka and T. Baldwin. Noun-Noun Compound Machine Translation: A Feasibility Study on Shallow Processing. Proceedings of the ACL-03 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment, Sapporo, Japan. 2003: 17-24
115. J. Yoon, K.-S. Choi, and M. Song. A corpus-based approach for Korean nominal compound analysis based on linguistic and statistical information. Natural Language Engineering, 2001, 7(3): 251-270
116. A.M. Buckeridge and R.F.E. Sutcliffe. Disambiguating Noun Compounds with Latent Semantic Indexing. Proceedings of the 2nd International Workshop on Computational Terminology, Taipei, Taiwan. 2002: 71-77
117. K. Takeuchi, K. Kageura, and T. Koyama. An LCS-Based Approach for Analyzing Japanese Compound Nouns with Deverbal Heads. Proceedings of the 2nd International Workshop on Computational Terminology, Taipei, Taiwan. 2002: 1-7
118. K.S. Nam and T. Baldwin. Interpreting Semantic Relations in Noun

Compounds via Verb Semantics. Proceedings of COLING/ACL, Sydney, Australia. 2006: 491-498

119. K.J. Chen and C.J. Chen. Automatic Semantic Classification for Chinese Unknown Compound Nouns. The 18th International Conference on Computational Linguistics, Centre Universitaire, Luxembourg. 2000: 173-179
120. D. Moldovan, A. Badulescu, M. Tatu, D. Antohe, and R. Girju. Models for the semantic classification of noun phrases. Proceedings of HLT/NAACL-2004 Workshop on Computational Lexical Semantics, Boston, USA. 2004: 60-67
121. E. Viegas, W. Jin, R. Dolan, and S. Beale. Representation and Processing of Chinese Nominals and Compounds. Proceedings of Coling-ACL98 workshop : the Computational Treatment of Nominals, Montreal, Canada. 1998: 20-24
122. J. Zhang, J. Gao, and M. Zhou. Extraction of Chinese Compound Words: An Experimental Study on a Very Large Corpus. Proceedings of the Second Workshop on Chinese Language Processing, Hong Kong. 2000: 132-139
123. M. Baroni, J. Matiasek, and H. Trost. Wordform- and Class-based Prediction of the Components of German Nominal Compounds in an AAC System. Proceedings of the 19th International Conference on Computational Linguistics, Taipei, Taiwan. 2002: 1-7
124. P. Nakov and M. Hearst. Search Engine Statistics Beyond the n-gram: Application to Noun Compound Bracketing. Proceedings of Ninth Conference on Computational Natural Language Learning, CoNLL-2005, Ann Arbor, MI, USA. 2005: 17-24
125. 赵军，黄昌宁. 汉语基本名词短语结构分析研究. 计算机学报, 1999, 22(2): 141-146
126. M. Crouse, R. Nowak, and R. Baraniuk. Wavelet-Based Statistical Signal Processing Using Hidden Markov Models. IEEE trans. On Signal Processing, 1998, 46(4): 886~902
127. 田永鸿，黄铁军，高文. 基于多粒度树模型的 Web 站点描述及挖掘算法. 软件学报, 2004, 15(9): 1393-1404

128. 梅家驹，竺一鸣. 同义词词林: 上海辞书出版社. 1983
129. E. Brill. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. Computational Linguistics, 1995, 21(4): 543-565
130. P.L. Shiuan and C.T.H. Ann. A Divide-and-Conquer Strategy for Parsing. Proceedings of the ACL/SIGPARSE 5th International Workshop on Parsing Technologies, Santa Cruz, USA. 1996: 57-66
131. C. Braun, G. Neumann, and J. Piskorski. A Divide-and-Conquer Strategy for Shallow Parsing of German Free Texts. Proceedings of ANLP-2000, Seattle, Washington. 2000: 239-246
132. C. Lyon and B. Dickerson. Reducing the Complexity of Parsing by a Method of Decomposition. International Workshop on Parsing Technology. 1997: 215-222
133. V.J. Leffa. Clause Processing in Complex Sentences. Proceedings of LREC'98, Granada, Spain. 1998: 937-943
134. E.F.T.K. Sang and H. Déjean. Introduction to the CoNLL-2001 Shared Task: Clause Identification. Proceedings of CoNLL-200l. 2001: 53-57
135. M. Li, J. Li, Z. Dong, Z. Wang, and D. Lu. Building a Large Chinese Corpus Annotated with Semantic Dependency. Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing, Sapporo, Japan. 2003: 84-91
136. R. Xu, Q. Lu, Y. Li, and W. Li. The Construction of A Chinese Shallow Treebank. Proceedings of 3rd ACL SIGHAN Workshop. 2004: 94-101
137. 刘挺，马金山，李生. 基于词汇支配度的汉语依存分析模型. 软件学报, 2006, 17(9): 1876-1883
138. T. Liu, J.S. Ma, and S. Li. Building a Dependency Treebank for Improving Chinese Parser. Journal of Chinese Language and Computing, 2006, 16(4): 207-224
139. S.D. Kim, B.-T. Zhang, and Y.T. Kim. Reducing parsing complexity by intra-sentence segmentation based on maximum entropy. Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora. 2000: 164-171
140. M. Jin, K. Mi-Young, D. Kim, and J.-H. Lee. Segmentation of Chinese

Long Sentences Using Commas. Proceedings of 3rd ACL SIGHAN Workshop. 2004

141. 周强. 汉语语料库的短语自动划分和标注研究. 博士学位论文, 北京: 北京大学. 1996
142. E. Black, F. Jelinek, J. Lafferty, D. Magerman, R. Mercer, and S. Roukos. Towards history-based grammars: using richer models for probabilistic parsing. Proceedings of the Fifth DARPA Speech and Natural Language Workshop, San Francisco: Morgan Kaufmann. 1992: 134-139
143. M. Collins, J. Hajic, L. Ramshaw, and C. Tillmann. A Statistical Parser for Czech. Proc. of the 37th Annual Meeting of the ACL. 1999: 505-512
144. A. Dubey and F. Keller. Probabilistic Parsing for German Using Sister-Head Dependencies. Proc. of the 41st Annual Meeting of the ACL. 2003: 96-103
145. B. Cowan and M. Collins. Morphology and Reranking for the Statistical Parsing of Spanish. Proc. of the Joint Conf. on Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP). 2005: 25-70
146. A. Arun and F. Keller. Lexicalization in Crosslinguistic Probabilistic Parsing: The Case of French. Proc. of the 43rd Annual Meeting of the ACL. 2005: 306-313
147. D.M. Bikel. Design of a Multi-lingual, Parallel-processing Statistical Parsing Engine. Proc. of the Human Language Technology Conference(HLT). 2002: 24-27
148. S. Marinov and J. Nivre. A Data-Driven Dependency Parser for Bulgarian. Proc. of the Fourth Workshop on Treebanks and Linguistic Theories (TLT). 2005: 89-100
149. S. Afonso, E. Bick, R. Haber, and D. Santos. "Floresta sint´a(c)tica": a treebank for Portuguese. Proc. of the Third Intern. Conf. on Language Resources and Evaluation (LREC). 2002: 1698-1703
150. K. Oflazer, B. Say, D.Z. Hakkani-Tr, and G. Tr. Building a Turkish Treebank. Treebanks: Building and Using Parsed Corpora, ed. A. Abeill. Vol. 20, Dordrecht. 2003: 261-277

附录

附录 1 汉语角色标记集

（1）人名角色标记集

角色标记	描述	例子
PER_SURNAME	中国人名的姓	*江*/泽/民
PER_SINGLE	二字中国人名的名	李/*鹏*/总理
PER_MIDDLE	三字中国人名的中间的字	张/*会*/鹏/同学
PER_END	三字中国人名的尾字	张/会/*鹏*/同学
PER_PREFIX	中国人名前缀	*老*/王，*小*/张
PER_SUFFIX	中国人名后缀	王/*总*，李/*氏*
PER_WORD	三字中国人名的名字是一词	张/*朝阳*，蔡/*国庆*
TRANS_FIRST	外国人名首字	*帕*/瓦/罗/蒂
TRANS_MIDDLE	外国人名中间的字	帕/*瓦*/*罗*/蒂
TRANS_END	外国人名尾字	帕/瓦/罗/*蒂*
CONJ	人名之间的连接词	乔/石/*和*/李/鹏/一起
PRE_CROSS	与人名上文形成交集歧义的词	现任/主席/*为何*/鲁/丽
NEXT_CROSS	与人名下文形成交集歧义的词	龚/学/*平等*/领导
PRE	人名的上文	*告诉*/李/明/这/件/事
NEXT	人名的下文	这/是/张/浩/*教授*
OTHER	其他的词	*改革*/*开放*/是/邓/小/平/同志/*为*/*我们*/*开创*/*的*/*伟大*/*事业*

（2）地名角色标记集

角色标记	描述	例子
FIRST	地名首字	*厄*/立/特/里/亚/国
MIDDLE	地名中间的字	厄/*立*/*特*/*里*/亚/国
END	地名尾字	厄/立/特/里/*亚*/国
SUFFIX	地名后缀	厄/立/特/里/亚/*国*

PRE	地名上文	*在*/杨/家/沟/召开
NEXT	地名下文	在/杨/家/沟/*召开*
CONJ	地名之间的连接词	刘/家/村/*和*/下/岸/村/相邻
OTHER	其他的词	*我*/*出生*/在/松/原/市

附录 2 863 词性标注集

Tag	Description	Example	Tag	Description	Example
a	adjective	美丽	ni	organization name	保险公司
b	other noun-modifier	大型，西式	nl	location noun	城郊
c	conjunction	和，虽然	ns	geographical name	北京
d	adverb	很	nt	temporal noun	近日，明代
e	exclamation	哎	nz	other proper noun	诺贝尔奖
g	morpheme	茨，甥	o	onomatopoeia	哗啦
h	prefix	阿，伪	p	preposition	在，把
i	idiom	百花齐放	q	quantity	个
j	abbreviation	公检法	r	pronoun	我们
k	suffix	界，率	u	auxiliary	的，地
m	number	一，第一	v	verb	跑，学习
n	general noun	苹果	wp	punctuation	，。！
nd	direction noun	右侧	ws	foreign words	CPU
nh	person name	杜甫，汤姆	x	non-lexeme	萄，翱

附录 3 863 词性标记集与北大词性标记集的对应关系

序号	863 词性标记及解释	北大词性标记及解释
（1）	n　普通名词	n　名词 Ng 名语素
（2）	nh 人名	nr 人名
（3）	ns 地名	ns 地名
（4）	ni 团体、机构、组织的专名词	nt 机构团体
（5）	nz 其它专名	nz 其它专名
（6）	nt 时间名词	t　时间名词 Tg 时间语素
（7）	nl 处所名词	s　处所词
（8）	nd 方位名词	f　方位词
（9）	m 数词	m　数词 Mg 数语素
（10）	q　量词	q　量词 Qg 量语素
（11）	b　区别词	b　区别词 Bg 区别语素
（12）	r　代词	r　代词 Rg 代语素
（13）	v　动词	v　动词 Vg 动语素 vd 副动词 vn 名动词
（14）	a　形容词	a　形容词 Ag 形语素 ad 副形词 an 名形词 z　状态词
（15）	d　副词	d　副词 Dg 副语素

（16）	p　介词	p　介词
（17）	c　连词	c　连词
（18）	u　助词	u　助词 Ug 助语素 y　语气词 Yg 语气语素
（19）	o　拟声词	o　拟声词
（20）	e　叹词	e　叹词
（21）	h　前接成分	h　前接成分
（22）	k　后接成分	k　后接成分
（23）	i　习用语	i　成语 l　习用语
（24）	j　简称	j　简略语
（25）	g　语素字	g　语素
（26）	x　非语素字	x　非语素字
（27）	wp 标点	w　标点符号
（28）	ws 字符串	nx 外文字符

攻读博士学位期间发表的论文

1 Ting Liu, Jinshan Ma and Sheng Li. Building a Dependency Treebank for Improving Chinese Parser. Journal of Chinese Language and Computing. 2006, 16(4): 207-224

2 刘挺，马金山，李生. 基于词汇支配度的汉语依存分析模型. 软件学报. 2006, 17(9):1876-1883（EI: 064410214587）

3 Liu Ting, Ma Jinshan, Zhang Huipeng and Li Sheng. Subdividing Verbs to Improve Syntactic Parsing，Journal of Electronics(China) . 2007, 24(3): 347-352

4 马金山，张宇，刘挺，李生. 利用三元模型及依存分析查找中文文本错误. 情报学报, 2004, Vol.23(6): 723-728

5 马金山, 刘挺, 李生. 基于 n-gram 及依存分析的中文自动查错方法, 第二十届东方语言计算机处理国际学术会议, 2003.8, 585-591

6 Ma Jinshan, Zhang Yu, Liu Ting and Li Sheng. A statistical dependency parser of Chinese under small training data. Workshop: Beyond shallow analyses - Formalisms and statistical modeling for deep analyses, IJCNLP-04, 2004

7 Ting Liu, Jinshan Ma, Huijia Zhu, and Sheng Li. Dependency Parsing Based on Dynamic Local Optimization. In Proceedings of Tenth Conference on Computational Natural Language Learning, CoNLL-shared task, 2006

8 马金山，刘挺，李生. 面向句法分析的句子片段识别. 全国第九届计算语言学学术会议, 2007, 252-257

9 马金山，刘挺，李生. 基于 SVM 的句子片段识别. 哈尔滨工业大学学报（已录用）

10 马金山，刘挺，李生. 基于隐马尔可夫树模型的名词复合短语分析.（已投往计算机学报）