# 一种改进的中文层次句法分析模型研究*

李军辉 周国栋 朱巧明 钱培德
苏州大学计算机科学与技术学院，苏州，215006
E-mail: lijunhui@suda.edu.cn

**摘 要：** 首先提出了层次句法分析模型，该模型先对输入句子进行词性标注和基本组块识别，紧接着循环多次进行复杂组块识别直至得到根结点。该方法本质上属于一种基于移进-归约序列的句法分析模型，因此具有此类模型的各类优点；然后，本文分析了移进-归约句法分析模型中存在的潜在问题，并通过在产生式（LHS→RHS）概率模型中引入 RHS 的分值，进一步提高系统性能。基于 CTB2.0 的实验表明，在正确分词情况下，RHS 分值的引入进一步提高了层次分析模型的性能（对长度≤40 的句子 F1 值提高 1.2），F1 值达到 81.2。

**关键词：** 中文句法分析，层次句法分析模型，层次句法树

# Research on an Improved Chinese Hierarchical Parsing Model

LI Jun-Hui, ZHOU Guo-Dong, ZHU Qiao-Ming, Qian Pei-De
School of Computer Science & Technology, Soochow University, Suzhou 215006

**Abstract:** Hierarchical parsing model is proposed for syntactic parsing and divided into three consequent tasks: (1) POS tagging; (2) identifying basic chunks and (3) recognizing complex constituents until the root node is formed. Hierarchical parsing belongs to shift-reduce based parsing models and owns their advantages, like running with less complexity time and performing with excellent classifiers. Moreover, this paper further improves the parsing performance by introducing the score of RHS into the production (LHS→RHS) probability model. Experiments on Chinese Treebank2.0 with gold word segmentation show the score of RHS helps hierarchical parsing with F1-measure of 1.08 improvement on test sentences (less <=40) and achieves 81.2 in F1-measure.

**Keywords:** Chinese syntactic parsing, hierarchical parsing model, hierarchical tree

## 1 引言

句法分析是自然语言处理的一个基础问题。它指的是根据给定的语法，自动推导出句子的语法结构，即句子所包含的句法单位和这些句法单位之间的关系。近些年来，随着大规模标注语料库的发布，句法分析得到越来越多的关注，基于统计的句法分析已代替原来的基于规则的方法。目前，可将大部分主流的统计句法分析模型分为：基于 PCFG 的句法分析模型和基于移进/归约序列的句法分析模型。基于 PCFG 的句法分析模型将句法树表示为一序列的产生式，模型假设各产生式之间是相互独立的，在计算产生式概率时，词汇化信息的引入往往能够更准确地计算产生式的概率值，得到较好的句法分析性能，详细可参见(Collins, 1999; Charniak, 2000)。而基于移进/归约序列的句法分析模型（Ratnaparkhi, 1999; Wang 等, 2006）将句法树转换为连续的移进/归约动作序列，在每个分析状态，分析器根据当前组块的上下文，调用分类器预测下一个动作类别。基于移进/归约序列的句法分析模型实际是将句法分析任务转化为一序列的分类任务，这使得模型中可以充分利用各种基于机器学习的分类方法，包括：最大熵、SVM 和决策树等。

* 基金项目：国家 863 计划(2006AA01Z147)；国家自然科学基金(60673041, 60873150)；国家教育部博士点基金(20060285008, 200802850006)；江苏省自然科学基金(BK2008160)；

本文主要做了以下工作：首先本文提出了一种新型的句法分析模型—层次句法分析模型，其本质上也是一种基于移进/归约序列的句法分析模型；其次，分析基于移进/归约序列句法分析模型存在的潜在问题，并通过在产生式（LHS→RHS）概率模型中引入 RHS 的分值，进一步提高句法分析的性能。基于 CTB2.0 的实验表明，在正确分词情况下，词汇化 PCFG 模型进一步提高了层次分析模型的性能，对长度≤40 的句子 F1 值达到 81.2。

## 2 层次句法分析模型（Hierarchical Parsing, HP）

### 2.1 层次句法分析流程

本文提出的层次句法分析模型与其他基于移进/归约序列的句法分析模型类似，都是通过预测"动作"来逐步构建句法树。如果一个连续的动作序列 A 能够构建句法树 T，则称动作序列 A 为句法树 T 的推导。值得注意的是，通过使用"动作序列"来表示句法树，看不到 PCFG 模型中经常提到的各类文法规则，句法树的分值也被分解为推导中各个动作的分值。

在自底向上的构建句法树过程中，根据已有的动作序列$\{a_1, \ldots, a_N\}$，预测下一个可能的动作$a_{N+1}$，产生一个新的动作序列$\{a_1, \ldots, a_N, a_{N+1}\}$，并且任意一棵句法树都有一个确切的推导。整个句法分析的过程可分解为三个过程：词性标注(POS Tagging)、基本组块识别(Basic Chunking)和复杂组块识别(Parsing)。对某给定的句子，需分别执行词性标注和基本组块识别各一次，紧接着，循环执行复杂组块识别过程直至识别出根结点。以下是各过程的描述，其中词性标注和基本组块识别过程的更详细描述可参考 Ratnaparkhi (1999)。

词性标注：对输入的句子词串 $S=(word_1, word_2, \ldots, word_n)$，从左至右分别预测每个词的词性，输出词性标注结果 $S=(word_1/pos_1, word_2/pos_2, word_n/pos_n)$。因此，本过程中的动作类别集合为词性标记集合。

基本组块识别：基本组块指的是子结点均为词性结点的组块。基本组块的识别以词性标注结果为输入。从左至右，为每个单词/词性标记对赋予基本组块识别标记。基本组块识别标记类别包括 Start_X, Joint_X 和 Other 三类，其中 X 为任意的组块类别。基本组块识别标记被用于基本组块的检测，如果 $word_m/pos_m$ 被标注为 Start_X，并且 $word_{m+1}/pos_{m+1}, \ldots, word_{m+i}/pos_{m+i}$ 均被标记为 Joint_X，则序列$\{word_m/pos_m, \ldots, word_{m+i}, pos_{m+i}\}$被组合成一个基本组块 X。

复杂组块识别：与基本组块不同的是，复杂组块指的是至少有一个子结点不是词性结点的组块。从左到右，分别为每个单元（此单元即可以是基本组块、也可以是复杂组块或词性结点）赋予复杂组块识别标记。复杂组块标类别包括 Begin_X, Middle_X, End_X, Single_X 和 Other 共五类，其中 X 为任意的组块类别。复杂组块识别标记被用于复杂组块的检测，如果连续的单元被标注为 Begin_X, Middle_X, …, Middle_X, End_X，则此连续单元被组合为一个复杂组块 X；如果某单元被标注为 Single_X，则此单元单独构成复杂组块 X。

不难分析，假设一个包含 n 个词的句法树对应的推导为$\{a_1, \ldots, a_N\}$，则动作序列$\{a_1, \ldots, a_n\}$为词性标记序列，$\{a_{n+1}, \ldots, a_{2n}\}$为基本组块识别标记序列，$\{a_{2n+1}, \ldots, a_N\}$为复杂组块识别标记序列。

### 2.2 特征集和训练集合的构造

**特征集的构造** 如前分析，本文提出的句法分析器在预测下一个动作时，是以已有的动作序列为依据的。使用条件概率 $P_X(a|b)$来表示在已有动作序列为 b 的前提下，下一个动作为 a 的概率。然而，动作序列 b 所构成的上下文包含了大量的信息，参考 Ratnaparkhi(1999)制定的特征和根据多次实验结果，最终分别为基本组块识别和复杂组块识别制定了表 1 和表 2 所示的特征模板：

表 1 基本组块识别过程使用的特征集合

| 一元特征（8 个） |
| --- |
| word(-2), word(-1), word(0), word(1), word(2),<br>pos(0), pos(1), pos(2) |
| 二元特征（8 个） |
| word(-1)&word(0), word(0)&word(1), action(-2)&pos(-2), action(-1)&pos(-1),<br>pos(0)&word(1), word(-1)&pos(0), word(0)&pos(1), pos(0)&pos(1) |
| n(n>2)元特征（2 个） |
| action(-1)&pos(-1)&word(0), action(-1)&word(-1)&pos(0) |

表 2 复杂组块识别过程使用的特征模板

| 一元特征（18 个） |
| --- |
| cons(i), cons(i*), cons(i**), -2≤i≤3 |
| 二元特征（8 个） |
| cons(i, i+1), cons(i*, i+1), cons(i, i+1*), cons(i*, i+1*), i=-1, 0 |
| n(n>2)元特征（6 个） |
| cons(0, 1*, 2*), cons(0, 1, 2*), cons(0, 1*, 2), cons(0*, 1*, 2*),<br>cons(0, 1*, 2*, 3*), cons(0*, 1*, 2*, 3*) |

表 1 和表 2 中使用到的模板函数定义如下:

- word(i)：窗口 i 的词
- pos(i)：窗口 i 的词性标记
- action(i)：窗口 i 的基本组块识别标记
- cons(i)：窗口 i 组块的中心词
- cons(i*)：窗口 i 复杂组块识别标记+组块的类别，若 i≥0，则省略复杂组块识别标记
- cons(i**)：窗口 i 复杂组块识别标记+组块的类别+组块中心词词性，若 i≥0，则省略复杂组块识别标记

**训练集的构造** 不难发现，对句法树 T 及其推导$\{a_1, \ldots, a_N\}$，根据预先制定的词性标注特征集、基本组块识别特征集和复杂组块识别特征集，将得到一个包含 N 条样例的事件集合，表示为 $TS=\{(a_i, b_i)|1\leq i\leq N\}$，其中 $a_i$ 为动作类别，$b_i$ 为根据特征模板得到的上下文特征集合。设句法树 T 包含的单词数为 n，则在事件集合 TS 中，子集合 $TS_1=\{(a_i, b_i)|1\leq i\leq n\}$为词性标注事件集，$TS_2=\{(a_i, b_i)|n+1\leq i\leq 2*n\}$为基本组块识别事件集，$TS_3=\{(a_i, b_i)|2*n+1\leq i\leq N\}$为复杂组块识别事件集。

## 3 层次句法分析的改进

在以上描述的层次句法分析中，产生式 P→C1…Cn 的概率表示为动作 Begin_P, Middle_P, …, End_P 的概率累积，如式(1)所示：

$$P_{action}(C_1 \cdots C_n \Rightarrow P) = P(Begin\_P \mid context(C_1)) * \prod_{i=2}^{n-1} P(Middle\_P \mid context(C_i)) * P(End\_P \mid context(C_n)) \quad \text{式(1)}$$

在式(1)中，context($C_i$)为组块 $C_i$ 的上下文特征。仔细分析可发现式(1)存在以下三个问题：

1). （Collins, 1999; Charniak, 2000）等词汇化 PCFG 模型取得较好的性能充分说明了中心

组块、中心词等信息的重要性，但在层次句法分析过程中，假设当前需要为单元X预测其动作，此时并不能根据上下文判断出单元X是否会是其父结点的中心组块，也无法知道单元X与其他相邻单元之间的修饰与被修饰关系；

2). 对于一些不符合语法规则的产生式，由于产生式的概率转化为动作的概率之积，故此类产生式仍具有一定的概率值，不利于区分和杜绝此类产生式。例如，VP→VV+AS+PP+NP，每个组块对应的动作(依次为Begin_VP、Middle_VP、Middle_VP和 End_NP)仍具有较大的概率值；再比如 VP→VV(期望)+NP，VV(期望)的动作Begin_VP 和 NP 的动作 End_NP 都具有较大的概率值，这使得产生式概率值很高，这与实际情况（动词 “期望”的宾语极少是名词短语NP）相悖论。

3). 层次句法分析模型总是试图找到最优的动作序列，但最优的动作序列并不等价于最优的产生式集合。

为解决上述三个问题，首先为产生式中的每个组块获取中心词信息，形式化地表示为：

$$P(pw,\ pt) \rightarrow C_1(cw_1, ct_1)..C_n(cw_n, ct_n) \qquad \text{式(2)}$$

式(1)的产生式概率计算是假设产生式右侧已存情况下形成父结点 P 的概率，整个句法树的概率为所有产生式概率的累积。这使得在计算整个句法树概率的过程中，都是假设产生式的右侧是已存在的并且是正确的，而并未考虑产生式的右侧是否合理，其存在的概率又是多少。例如，若产生式右侧为“VV(期望)+NP”，则其存在不合理，应具有较低的概率值。为此，定义在已知中心词信息的情况下，产生式右侧的分值 $Score_{right}$，表示为：

$$Score_{right} = P(C_1(cw_1, ct_1)..C_n(cw_n, ct_n) \mid hw, ht) \qquad \text{式(3)}$$

其中，hw 和 ht 分别表示右侧中心组块的中心词和中心词词性，这可以根据中心组块规则而获得。假设在产生式的右侧，不相邻组块之间是相互独立的，这样，式(3)所示的 $Score_{right}$ 可表示为：

$$Score_{right} = P(C_1(cw_1, ct_1)..C_n(cw_n, ct_n) \mid hw, ht) = \prod_{i=1}^{n} P_c(C_i(cw_i, ct_i) \mid C_1(cw_1, ct_1)..C_{i-1}(cw_{i-1}, ct_{i-1}), hw, ht) \qquad \text{式(4)}$$

$$= \prod_{i=1}^{n} P_c(C_i(cw_i, ct_i) \mid C_{i-1}, hw, ht)$$

同时，进一步将各子结点的概率分解为：

$$\begin{aligned} & P_c(C_i(cw_i, ct_i) \mid C_{i-1}, hw, ht) \\ & = P_{c1}(C_i \mid C_{i-1}, hw, ht) * P_{c2}(ct_i \mid C_i, C_{i-1}, hw, ht) * P_{c3}(cw_i \mid ct_i, C_i, C_{i-1}, hw, ht) \end{aligned} \qquad \text{式(5)}$$

将式(2)所示的产生式概率和式(3)所示的产生式右侧分值线性组块，得到产生式的分值 $Score_p$，表示为：

$$Score_p = Score_{right}{}^{\alpha} * P_{action}(C_1 \cdots C_n \Rightarrow P)^{1-\alpha} \qquad \text{式(6)}$$

如式(6)所示，产生式的分值由两部分组成：i. 产生式右侧出现的分值; ii. 产生式存在的情况下，得到父结点的概率。同时，参数 $\alpha$ 用于表示各部分值的权重。这样，整个句法树的分值表示为各产生式分值之积。

在估算式(5)中的 $P_{c1}$、$P_{c2}$ 和 $P_{c3}$ 时，使用 Back-off 平滑方法来缓解数据稀疏带来的问题，表 3 列出了计算各条件概率时使用的条件变量情况。例如，$P_{c2}$ 的估算值为：

$$e = \lambda_1 e_1 + (1-\lambda_1)(\lambda_2 e_2 + (1-\lambda_2)e_3) \quad \text{式(7)}$$

其中，e1= $P_{c2}(ct_i|\ C_i, pt, C_{i-1}, hw)$，e2= $P_{c2}(ct_i|\ C_i, ht)$，e3= $P_{c2}(ct_i|\ C_i,)$。$P_{c1}$ 和 $P_{c3}$ 的估算方法可按此类推。参数 $\lambda_i$ 的计算方法可参考文献 Collins(1999)。

表 3 Back-off 平滑各层的条件变量

| Back-off Level | $P_{c1}(C_i|\ldots)$ | $P_{c2}(ct_i|\ldots)$ | $P_{c3}(cw_i|\ldots)$ |
|---|---|---|---|
| 1 | ht, $C_{i-1}$, hw | $C_i$, ht, $C_{i-1}$, hw | $ct_i$, $C_i$, ht, $C_{i-1}$, hw |
| 2 | ht | $C_i$, ht | $ct_i$, $C_i$, ht |
| 3 | | $C_i$ | $ct_i$, $C_i$ |
| 4 | | | $ct_i$ |

# 4 实验结果与分析

## 4.1 实验设置

本文以宾州大学的 Chinese Treebank2.0 作为实验语料。CTB2.0 是由国际语言数据联盟（LDC）发布的一个语料库，为中文句法分析提供了一个公共的实验数据。CTB2.0 其包括 325 篇文章，含有 4186 个句子，约 100K 词组成。按照中文句法分析实验的常规划分，把 Section001-270（3485 条句子）作为训练集，Section301-325（353 条句子）作为开发集，Section271-300（348 条句子）作为测试集。在为层次句法分析模型抽取训练事件时，预先将训练集中的每棵句法树转换为对应的层次句法树。

以下实验均使用最大熵模型[1]预测各类标记的概率分布。实验的句法分析性能评测采用 EVALB评测程序[2]，本文报告三个常用的评价指标值：标记召回率(LR)、标记准确率(LP)和F1 值。词性标注(POS)性能评价指标为精确率(Acc.)，即在测试集所有词中，有多少词的词性标记被正确标注。同时，使用Dan Bikel提供的脚本程序[3]来验证句法分析性能是否发生显著性变化。

## 4.2 改进的层次句法分析性能

按第 3 节所述的方法对层次句法分析模型进行改进和优化。在式(6)所示的产生式分值计算公式中，$Score_{right}$ 为产生式右侧出现的分值，$P_{action}$ 为在产生式右侧存在的情况下，得到父结点的概率。图 1 给出了 $\alpha$ 取值范围在区间[0, 1.0]对开发集的句法分析性能影响。

从图 1 可以看出：当 $\alpha$ 值位于区间[0, 0.3]时，产生式右侧分值 $Score_{right}$ 能够促进层次句法分析性能的提高，但当 $\alpha$ 值>0.3 时，句法分析性能急剧下降。这是因为，$Score_{right}$ 的统计值远少于 $P_{action}$ 概率值，而当 $\alpha$ 值较大时，式(5)的值主要由 $Score_{right}$ 决定。特别地，当 $\alpha$ 为 0.12 时，在开发集上取得最优的句法分析性能 F1 值，较未使用 PCFG 统计模型提高了 0.56。

[1] The OpenNLP Maximum Entropy Package. http://maxent.sourceforge.net/
[2] The Evalb Program. http://nlp.cs.nyu.edu/evalb/
[3] Randomized Parsing Evaluation Comparator. http://www.cis.upenn.edu/~dbikel/software.html

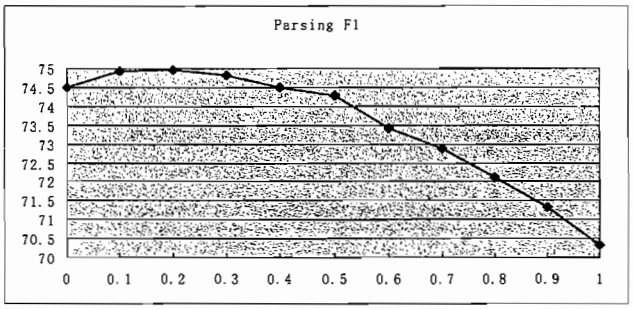


图 1 $\alpha$ 值在[0, 1]区间对开发集的句法分析性能影响

表 4 列出了改进的层次句法分析模型在测试集上的分析性能。

表 4 改进 HP 在测试集的句法分析性能（基于正确分词）

| Model | <=40 words 共 300 条句子 | | | | all sentences 共 348 条句子 | | | |
|---|---|---|---|---|---|---|---|---|
| | LR(%) | LP(%) | F1 | POS(%) | LR(%) | LP(%) | F1 | POS(%) |
| HP | 77.62 | 82.59 | 80.03 | 93.48 | 74.25 | 79.07 | 76.59 | 92.68 |
| 改进 HP | 78.41 | 84.09 | 81.15 | 93.56 | 74.65 | 80.28 | 77.36 | 92.81 |

从表 4 中可以看出，改进的层次句法分析能够进一步提高句法分析的性能，对长度≤40 的句子性能 F1 值达到了 81.15，显著性测试结果显示召回率和准确率均取得了显著性的提高（$p<0.05$）。

### 4.3 相关工作及性能比较

表 5 相关工作的中文句法分析性能比较（基于正确分词）

| Model | <=40 words 共 300 条句子 | | | all sentences 共 348 条句子 | | |
|---|---|---|---|---|---|---|
| | LR(%) | LP(%) | F1 | LR(%) | LP(%) | F1 |
| Bikel & Chiang(2000) | 76.8 | 77.8 | 77.3 | - | - | - |
| Chiang & Bikel(2002) | 78.8 | 81.1 | 79.9 | - | - | - |
| Levy & Manning (2003) | 79.2 | 78.4 | 78.8 | - | - | - |
| Bikel (2004) | 78.0 | 81.2 | 79.6 | - | - | - |
| Jiang (2004) | 80.1 | 82.0 | 81.1 | - | - | - |
| Xiong et al. (2005) | 78.7 | 80.1 | 79.4 | - | - | - |
| Wang et al. (2006) | 79.2 | 81.1 | 80.1 | 76.2 | 78.0 | 77.1 |
| Charniak Parser | 79.2 | 81.7 | 80.4 | 75.3 | 78.6 | 76.9 |
| Berkeley Parser | 77.6 | 80.7 | 79.1 | 75.2 | 78.1 | 76.6 |
| *HP* | *77.6* | *82.6* | *80.0* | *74.3* | *79.1* | *76.6* |
| *改进 HP* | *78.4* | *84.0* | *81.1* | *74.7* | *80.3* | *77.4* |

自LDC发布CTB1.0 以来，中文句法分析愈来愈被人们关注，其性能水平也得到不断的提高。Bikel和Chiang (2000)构建了两个中文句法分析器：基于Collins (1999)模型 2 的词汇化PCFG模型和基于TAG(Tree-Adjoining Grammar)的分析模型，取得的分析性能LR/LP为76.8%/77.8%。Chiang和Bikel (2002)提出了一种自动识别短语中心成分的方法，利用了EM算法自动发现隐藏在树库中的信息，取得了较好的分析性能，LR/LP为 78.8%/81.1%。Levy和Manning (2003)针对常见的句法分析错误进行分析和改进，采用的是Factored模型，取得的句法分析性能LR/LP为 79.2%/78.4%。Bikel (2004)毕业论文实现Collins模型，得到性

能LR/LP为 78.0%/81.2%。同样，Jiang (2004)将Collins模型应用于中文，取得的性能LR/LP为 80.1%/82.0%。Xiong等(2005)基于中心词驱动模型，对基本名词短语作了特殊处理，并且利用外部资源（知网等）中的语义信息提高了模型的性能，最终的性能LR/LP为78.7%/80.1%。Wan等(2006)实现了一个快速的确定性的中文句法分析器，采用一种基本移进-归约序列的句法分析模型，在预测动作的概率分布时，综合了多种分类器的预测结果，最终的性能LR/LP为 79.2%/81.1%。同时，对提供源码的Charniak Parser[4]和Berkeley Parser[5]采用相同的实验数据集，测试其性能。表 5 给出了以上各模型与本文模型的分析性能比较，所有的分析器皆基于正确的分词。

## 5 结束语

句法分析是自然语言处理研究中的关键技术之一，其结果的好坏直接影响到对自然语言句子的理解。本文首先提出了层次句法分析模型，此模型将句法树的构建转化为一个分类问题，这使得能够利用各类性能优秀的分类模型。该句法分析模型属于基于移进-归约序列句法分析模型的一种，因此具有此类模型的各类优点。同时，本文分析了移进-归约句法分析模型中存在的潜在问题，并通过在产生式（LHS→RHS）概率模型中引入 RHS 分值，进一步提高了系统性能。但从表 4 中也发现，本模型虽具有较高的准确率，但召回率却低于其他模型，因此，如何提高本模型的召回率是下一步要解决的问题。

## 参考文献


[1] Daniel M. Bikel and David Chiang. 2000. Two statistical pasing models applied to Chinese Treebank. In *Proceedings of the Second Chinese Lnaguage Processing Workshop, ACL 2000.*

[2] Daniel M. Bikel. 2004. On the parameter space of generative lexicalized statistical parsing models. Ph.D. thesis, *University of Pennsylvania.*

[3] Eugene Charniak. 2000. A maximum-entropy inspired parser. In *Proceedings of NAACL-2000.*

[4] David Chiang and Daniel M. Bikel. 2002. Recovering latent information in treebanks. In *Proceedings of COLING-2002.*

[5] Michael John Collins. 1999. Head-driven statistical models for natural language parsing. Ph.D. thesis. *University of Pennsylvania.*

[6] Zhengping Jiang. 2004. Statisitcal Chinese parsing. Honours thesis, *National Unversity of Singapore.*

[7] Roger Levy and Christopher D. Manning. 2003. Is it harder to parse Chinese, or the Chinese Treebank? In *Proceedings of ACL-2003.*

[8] Adwait Ratnaparkhi. 1999. Learning to parse natural language with maximum entropy models. *Machine Learning, 34(1-3):151-175.*

[9] Mengqiu Wang, Kenji Sagae, and Teruko Mitamura. 2006. A fast, accurate deterministic parser for Chinese. In *Proceedings of ACL-COLING-2006.*

[10] Deyi Xiong, Shuanglong Li, Qun Liu, Shouxun Lin, and Yueliang Qian. 2005. Parsing the Penn Chinese Treebank with semantic knowledge. In *Proceedings of IJCNLP-2005.*


[4] Charniak Parser. http://www.cs.brown.edu/~ec/

[5] Berkeley Parser. http://code.google.com/p/berkeleyparser/