

# 最大生成树算法和 Nivre 算法相结合的中文依存关系解析

周惠巍, 黄德根, 高洁, 杨元生

大连理工大学 计算机科学与技术学院, 大连 116024

E-mail: zhouhuiwei@dlut.edu.cn

**摘要:** 基于最大生成树解析算法和 Nivre 解析算法的互补关系, 提出了最大生成树解析算法和 Nivre 解析算法相结合的中文依存关系解析方法。利用 Nivre 模型的解析结果修正最大生成树模型有向边的权重, 再搜索最大生成树作为依存树。使用宾州中文树库中的 4500 句语料作十折交叉测试, 结合模型的依存关系正确率达到了 86.14%。结果表明本文提出的结合方法有效地提高了中文依存关系解析性能。

**关键词:** 中文依存关系解析; 最大生成树解析算法; Nivre 算法; 支持向量机

## MST Parsing Algorithm and Nivre's Algorithm Integrated Dependency Parser for Chinese

Zhou Huiwei, Huang Degen, Gao Jie, Yang Yuansheng

School of Computer Science and Technology, Dalian University of Technology, Dalian 116024

E-mail: zhouhuiwei@dlut.edu.cn

**Abstract:** We present a maximum spanning tree (MST) algorithm and Nivre's algorithm integrated dependency parser for Chinese based on their complementary strengths. At parsing time, we introduce the results of Nivre parser into MST parser. Our system achieves the accuracy of 86.14% using 10-fold cross-validation on the Penn Chinese Treebank Corpus. Experiments show that the proposed method improved the parsing accuracy significantly.

**Keywords:** Chinese dependency analysis; maximum spanning tree algorithm; Nivre's algorithm; support vector machines

### 1 引言

依存关系解析是句法分析的一个重要方法, 是自然语言处理的一个重要任务, 近年来引起了人们的广泛兴趣。英文依存关系解析<sup>[1][2]</sup>与日语依存关系解析<sup>[3]</sup>已经取得了较好的研究成果。中文与英文等其他语言有着明显区别, 中文没有严格意义上的形态变化, 不同词类之间的界限不十分明显, 这使得中文依存关系解析变得更为困难。

大规模语料库的构建, 使得人们可以使用数据驱动的方法构建依存关系解析器。数据驱动依存关系解析方法可以分为两类: 最大生成树(MST)解析方法和决策式解析方法。MST 解析方法将句子中的词看作图的顶点, 将词间依存关系连线看作图的有向边。机器学习获得依存关系概率(有向边权重)函数, 基于概率函数计算有向边的权重。从而将依存关系解析问题转换为在完全有向图中搜索最大生成树问题。Eisner 将最大生成树应用于英语等非交叉结构语言的依存关系解析( $O(n^3)$ )<sup>[4]</sup>, McDonald 使用 Chu-Liu-Edmonds 最大生成树解析算法<sup>[5]</sup>解析捷克语等交叉结构语言的依存关系( $O(n^2)$ )<sup>[6]</sup>。辛霄将 MST 算法应用于中文依存关系解析, 取得了很好的解析效果<sup>[7]</sup>。刘挺基于词汇支配度构建了汉语依存分析模型, 找出一棵概率最大的依存树<sup>[8]</sup>。决策式解析方法有效地利用了句子的中间解析结果特征, 从一个解析状态转移至下一个解析状态。解析采用贪婪算法, 每一步转移都寻求局部最优转移状态, 直至解析结束。代表性的决策式解析方法有面向英语的 Nivre 算法<sup>[1]</sup>、Yamada 算法<sup>[2]</sup>, 面向日语的组块逐步应用算法<sup>[3]</sup>。Cheng 将 Nivre 算法<sup>[1]</sup>和 Yamada 算法<sup>[2]</sup>应用于中文依存关系解析, 基于最大熵和支持向量机(SVMs)进行确定性依存关系解析<sup>[9]</sup>, 并使用全局特征和根节点解析器提高解析性能<sup>[10]</sup>。Xu 基于 SVMs 构建了中文短语依存关系解析器<sup>[11]</sup>。段湘煜提出了两种模型对句法分析动作进行建模避免了原决策式依存分析方法的贪婪性<sup>[12]</sup>。Cheng

的研究表明基于 SVMs 的 Nivre 算法<sup>[1]</sup>更符合中文的语法特点<sup>[9]</sup>。

最大生成树解析方法和决策式解析方法差别很大，本质上是对立和互补的。最大生成树解析方法采用全句的依存树进行训练，解析时使用最大生成树搜索整句的最优依存树，因此具有全局性和完备性。但是，因为不到最大生成树搜索完毕，无法获得任何中间解析结果，所以无法将解析的中间结果应用于后续解析。而决策式解析方法基于状态转移过程进行训练，解析时搜索局部最优转移状态，直至整句解析结束，因此具有局部性和贪婪性。但是，可以将句子解析的中间结果用于后续解析。McDonald 和 Nivre 实验分析两种模型的解析错误<sup>[13]</sup>，对于距离较远的依存关系，MST 模型的解析正确率远远高于决策式解析模型。而对于距离较近的依存关系，决策式模型的解析结果更好。决策式模型解析短句的依存关系更优，而 MST 模型解析长句的依存关系更佳。Nivre 和 McDonald 基于其互补关系提出在训练时结合两种模型的方法，将一种模型的解析结果作为指导特征引入另一种模型<sup>[14]</sup>。当两种模型的解析正确率相差不大时，结合后的模型显著地提高了原基本模型的依存关系正确率。

基于两种模型的特点，本文提出解析时结合两种模型的方法。基于 Nivre 模型的解析结果，修正 MST 模型有向边的权重，再基于 Chu-Liu-Edmonds 最大生成树解析算法<sup>[6]</sup>搜索得到该句的依存树。既有效地利用了 Nivre 解析模型的中间解析结果特征，又使用了具有全局性的最大生成树搜索算法，避免了因为决策式解析算法贪婪性而引起的解析错误。

## 2 改进的 Nivre 算法

基于 Nivre 算法的解析系统利用机器学习的方法确定 Right, Left 等转移操作，通过一系列转移操作，可将初始的输入词序列  $Sen = \{w_1, w_2, \dots, w_n\}$  转移至终止状态，获得依存关系解析结果  $D = \{Dep(1), Dep(2), \dots, Dep(n)\}$ 。其中， $Dep(k)$  为词  $w_k (k=1, \dots, n)$  所依存的词  $i$ 。

Cheng<sup>[9]</sup>将确定性 Nivre 算法应用于中文时，对 Reduce 操作和 Shift 操作的划分不十分准确。针对此问题，我们提出一种改进的确定性 Nivre 算法。

在 Nivre 算法中，解析器可以表示成一个三元组  $\langle S, I, A \rangle$ 。其中  $S$  和  $I$  是堆栈， $I$  中是待解析的输入序列。 $A$  是一个集合，存放在解析过程中确定下来的依存关系项。假设给定一个输入序列  $Sen$ ，解析器首先被初始化成  $\langle nil, Sen, \varphi \rangle$ 。解析器解析栈  $S$  的栈顶元素  $t$  和栈  $I$  的栈顶元素  $n$  的依存关系，然后采取相应的动作，操作栈中的元素移动，算法迭代进行直至栈  $I$  为空。此时，解析器停止迭代，输出集合  $A$  中的依存关系序列。Nivre 算法一共定义了 4 个操作：

(1) Right. 在当前三元组  $\langle t|S, n|I, A \rangle$  中，假如存在依存关系  $t \rightarrow n$ ，即  $t$  依存于  $n$ ，则在集合  $A$  中添加项  $(t \rightarrow n)$ ，同时弹出  $S$  中的栈顶元素  $t$ ，于是三元组变为  $\langle S, n|I, A \cup \{(t \rightarrow n)\} \rangle$ 。

(2) Left. 在当前三元组  $\langle t|S, n|I, A \rangle$  中，假如存在依存关系  $n \rightarrow t$ ，则在集合  $A$  中添加项  $(n \rightarrow t)$ ，同时把元素  $n$  压入栈  $S$  中，于是三元组变为  $\langle n|t|S, I, A \cup \{(n \rightarrow t)\} \rangle$ 。

如果  $n$  与  $t$  不存在依存关系，改进的 Nivre 算法对 Reduce 操作和 Shift 操作做了明确的定义。

(3) Reduce. 假如两栈顶元素  $n$  与  $t$  不存在依存关系， $t$  有父亲节点在其左侧，并且该父亲节点与  $n$  存在依存关系，解析器从栈  $S$  中弹出  $t$ ，于是三元组变为  $\langle S, n|I, A \rangle$ 。

(4) Shift. 当 Right, Left, Reduce 操作条件都不满足时，将  $n$  压入栈  $S$  中，三元组变为  $\langle n|t|S, I, A \rangle$ 。

基于 SVMs 构建 Nivre 依存关系解析器。解析当前三元组的两个栈顶元素  $(t, n)$  的依存关系时，选取表 1 所示特征。

表 1 Nivre 算法的特征向量

节点 $t-2, t-1, t, n, n+1, n+2$ 的词和词性
节点 $t-2, t-1, t, n, n+1, n+2$ 的孩子节点的词和词性
节点 $t$ 与 $n$ 在句中的距离

### 3 MST 解析模型与 Nivre 解析模型的结合

MST 解析模型和 Nivre 解析模型均为数据驱动模型, 因此 Nivre 和 McDonald 提出一种在学习时进行结合的方法<sup>[14]</sup>。将其中一个模型的解析结果作为指导特征引入另一个模型, 以此促进两个模型的相互学习、融合。这种结合方法简单而且显著提高了依存关系正确率。本文考虑到两种模型的互补关系及其各自较高的解析正确率, 提出一种在解析时结合的方法, 有效地提高了解析正确率, 并弥补了基本模型的不足。

我们将支持向量机 (SVMs) 用于计算 MST 解析模型中有向图各边依存概率。sigmoid 函数可以将 SVMs 输出的距离函数较好地近似为概率函数<sup>[15]</sup>。本文基于 Chu-Liu-Edmonds 算法搜索最大生成树。解析两个词( $i, j$ )依存关系的具体特征见表 2。

表 2 MST 算法的特征向量

节点 $i-2, i-1, i, i+1, i+2$ 的词和词性
节点 $j-2, j-1, j, j+1, j+2$ 的词和词性
节点 $i$ 与 $j$ 在句中的距离

#### 3.1 学习时的结合方法

MST 解析模型和 Nivre 解析模型都是基于机器学习获得一个解析函数  $s: X \rightarrow R$ 。模型不同,  $X$  不同。在 MST 模型中,  $X$  是一系列的依存关系连线; 而在 Nivre 模型中,  $X$  是一系列转移操作。无论哪种模型, 输入都表示为一个  $k$  维特征向量  $f: X \rightarrow R^k$ 。Nivre 和 McDonald<sup>[14]</sup> 将一个模型称为基本模型  $B$ , 另一个称为指导模型  $C$ 。基于指导模型  $C$  解析语料, 然后将获得的解析结果作为特征加入到基本模型  $B$  的训练中, 学习获得结合解析模型  $B_C$ 。这种结合方法的基本思想是认为指导模型是可信任的模型, 希望基本模型在学习时能吸收指导模型的优点, 从而提高其解析性能。结合模型  $MST_{Nivre}$  和  $Nivre_{MST}$  的具体指导特征如表 3 所示。

表 3  $MST_{Nivre}$  与  $Nivre_{MST}$  的指导特征

$MST_{Nivre}(i, j)$	$Nivre_{MST}(t, n)$
$(i, j)$ 是否在 $G_x^{Nivre}$ 中	$(t, n)$ 是否在 $G_x^{MST}$ 中, $(n, t)$ 是否在 $G_x^{MST}$ 中, $t$ 的父亲节点的方向(left/right/root), $n$ 的父亲节点的方向(left/right/root)

基于 13 种语言的实验结果表明, 除瑞典语的  $Nivre_{MST}$  模型外, 其他结合模型的解析性能均好于基本模型。通过观察实验数据发现, 除瑞典语外, 其他语言的指导模型和基本模型的依存关系正确率均很接近 (小于 2%)。且当指导模型的正确率高于基本模型时, 结合模型的正确率较基本模型提高较大。而当指导模型的正确率远远小于基本模型时 (瑞典语), 结合模型的正确率则低于基本模型。

语料的不同, MST 解析模型和 Nivre 解析模型的解析正确率可能相差较大, 这样得到的结合模型的解析性能有可能低于基本模型。因此, 我们提出一种在解析时结合两个模型的方法。

#### 3.2 解析时的结合方法

我们以一种模型为基本模型, 利用另一种模型的解析结果修正基本模型的解析结果。

McDonald 和 Nivre 基于 13 种语言的实验表明, MST 模型的解析正确率普遍高于 Nivre 模型, MST 模型具有全局性<sup>[14]</sup>。我们以 MST 模型为基本模型, 利用 Nivre 模型的解析结果修正完全有向图的权重, 再基于 Chu-Liu-Edmonds 算法搜索依存树。每种算法均输出 1-best 结果。

基于 MST 算法获得 MST 依存关系树; 基于 Nivre 算法获得 Nivre 依存关系树。依据两棵依存树修正完全有向图的权重  $score(i, j, y) \in R$  :

(1) 对于 MST 依存树和 Nivre 依存树中同时存在的边, 将其权重乘以一个权值  $a$ , 获得新的权重  $a \cdot \text{score}(i_{MN}, j_{MN}, y) \in R$ 。

(2) 对于在 MST 依存树中存在、而在 Nivre 依存树中不存在的边, 将其权重乘以一个权值  $b_1$ , 获得新的权重  $b_1 \cdot \text{score}(i_M, j_M, y) \in R$ 。

(3) 对于在 Nivre 依存树中存在、而在 MST 依存树中不存在的边, 将其权重乘以一个权值  $b_2$ , 获得新的权重  $b_2 \cdot \text{score}(i_N, j_N, y) \in R$ 。

最后基于修正过的权重函数  $\text{mod-score}(i, j, y) \in R$  搜索得到依存树  $T = \arg \max_{G=(V,E)} \sum_{(i,j) \in E} \text{mod-score}(i, j, y)$ 。

## 4 实验

实验采用宾州中文树库 (Penn Chinese Treebank, CTB) 5.0。由于该树库为短语结构, 没有依存关系, 我们首先基于中心子节点过滤表将短语结构的宾州中文树库转换为依存结构树库。我们采用 1~4,500 句的语料作十折交叉验证, 语料包含 118,703 词。因为很难制定逗号两侧的短语的依存规则, 我们同 Cheng 一样, 将逗号和句号都作为句子的结束标记<sup>[10]</sup>。

表 4 显示了 4 种模型的解析结果。实验结果表明: 在 CTB 语料库上, MST 模型的依存关系正确率远远高于改进的 Nivre 模型 (1.68%)。结合模型  $\text{Nivre}_{\text{MST}}$  的依存关系正确率较基本模型 (Nivre 模型) 提高了 1.24%; 而结合模型  $\text{MST}_{\text{Nivre}}$  的依存关系正确率却低于基本模型 (MST 模型)。由此可见, 当指导模型的依存关系解析性能远远低于基本模型时, 对基本模型没有正确的指导作用。

表 4 依存关系解析结果

模型	依存关系正确率(%)	句子正确率(%)
Nivre 模型	80.48	60.51
改进 Nivre 模型	83.39	64.54
MST 模型	85.07	62.56
$\text{Nivre}_{\text{MST}}$ 模型	84.63(↑1.24)	63.05(↓1.49)
$\text{MST}_{\text{Nivre}}$ 模型	83.99(↓1.08)	63.40(↑0.84)

采用本文提出的解析时结合的方法, 设权值  $a=1.2$ , 调节权值  $b_1$ 、 $b_2$  的比例关系, 解析结果如表 5 所示。

由表 5 的实验数据可见, 我们提出的解析时结合的解析效果均好于文献[14]学习时结合的解析效果。解析时结合的方法比学习时结合的方法获得了更加稳定的解析性能。同时发现, 当  $b_1$  和  $b_2$  的比值为 1: 1.2 时, 即  $a$  与  $b_2$  的权值相同时, 解析效果最好。可见, Nivre 模型解析获得的依存关系, 无论其在 MST 模型解析结果中是否存在, 影响权值一致时解析效果最好。

表 5 解析时结合的解析结果( $a=1.2$ )

$a$	$b_1$	$b_2$	依存关系正确率(%)	句子正确率(%)
1.2	1	1.0	84.97	62.14
1.2	1	1.1	85.63	63.47
<b>1.2</b>	<b>1</b>	<b>1.2</b>	<b>86.00</b>	<b>64.51</b>
1.2	1	1.3	85.87	64.34
1.2	1	1.4	85.71	64.21

改变 Nivre 模型的影响权值, 实验结果如表 6 所示。通过实验结果发现, 当 Nivre 模型的影响权值为 1.3 时, 结合模型的解析效果最好。

表6 改变Nivre模型影响权值的解析结果

$a$	$b_1$	$b_2$	依存关系正确率(%)	句子正确率(%)
1.1	1	1.1	85.69	63.57
1.2	1	1.2	86.00	64.51
<b>1.3</b>	<b>1</b>	<b>1.3</b>	<b>86.14</b>	<b>65.06</b>
1.4	1	1.4	86.13	65.45

图1显示了三种模型依存关系正确率和句子长度(句长间隔为2: 1-2, 3-4, 等)的关系。

实验结果表明: 在短句子中 MST 模型和 Nivre 模型的解析性能相差不大。但是随着句子长度的增大, Nivre 模型的解析性能明显低于 MST 模型, 这主要是由于 Nivre 算法贪婪性解析错误逐步扩散引起的。结合模型的解析性能好于 Nivre 模型和 MST 模型。这表明, 随着句子长度的增大, 结合模型不仅保留了 MST 算法的全局搜索优势, 又有效地利用了 Nivre 算法丰富的中间解析结果特征, 使得其解析性能比 MST 模型有很明显的提高。

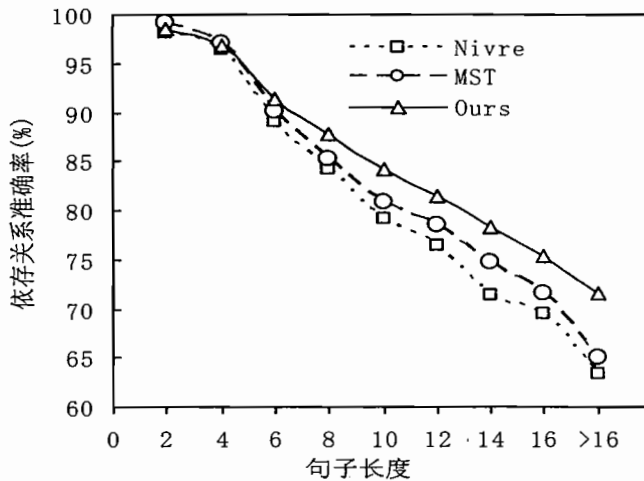


图1 依存关系正确率与句子长度的关系

辛霄采用 CoNLL 2008 共享任务语料, 比较了自顶向下、自底向上、MST 三种依存关系解析算法。在 WSJ 测试语料库上, MST 算法取得了最好的解析效果, 依存关系正确率达到 87.42%<sup>[7]</sup>。基于 CTB 语料, 我们再现了 MST 算法, 解析结果如表 4 所示。刘挺手工建立了 46,000 个句子的汉语依存树库, 将树库中的 40,000 句作为训练集, 在 4,000 句的测试集上, 获得了约 74% 的依存关系正确率<sup>[8]</sup>。Cheng 采用 Nivre 算法, 并使用全局特征和根节点解析器提高解析性能。实验采用 CTB 语料库, 训练语料为 377,408 词, 测试语料为 63,886 词。实验取得了很好的效果, 依存关系解析正确率达到 86.18%, 句子正确率达到 61.33%<sup>[10]</sup>。Xu 构建了中文短语依存关系解析器, 并假设句中各个短语依存且仅依存于其后方某个短语。实验采用 CTB 语料库, 训练语料为 4,000 句, 任意另取 100 句作为测试语料, 依存关系解析正确率为 77.3%<sup>[11]</sup>。段湘煜基于动作建模的依存句法分析器优于原始决策式分析器。实验采用 CTB 语料库, 训练语料为 434,936 词, 测试语料为 50,319 词, 依存关系解析正确率达到 84.05%<sup>[12]</sup>。McDonald 和 Nivre 采用 CoNLL-X 共享任务语料库, 依存关系解析正确率达到 88.43%<sup>[14]</sup>。我们基于 CTB 语料, 再现了他们学习时结合的方法, 解析结果见表 4。

## 5 结论

本文在已有的 MST 算法和 Nivre 算法的基础上, 提出了解析时结合 MST 算法和 Nivre 算法的中文依存关系解析方法。实验证明本文提出的结合方法, 比采用一种算法的单一模型, 以及学习时结合的方法具有更好的解析效果, 并具有稳定的解析性能, 有效地避免了单独一种解析算法的解析错误。

在 CTB 上的实验结果表明, 对于 Nivre 解析结果中存在的依存关系, 无论其在 MST 解析结果中是否存在, 权值一致时解析效果最好。这样我们只需根据 Nivre 解析结果修改 MST 模型中有向边的权重, 再基于新的权重获得依存树。结合方法在训练时间上的消耗也可以通过并行处理加以改进。

MST 模型的训练语料为句中的所有词对, 因此训练样例数量庞大, 且正负样例的比例不均衡。这样训练 MST 模型在时间和空间上消耗都较大, 使得在大规模语料上训练 MST 模型较为困难。今后, 我们可以通过采用其他机器学习方法或合理删减负例的方法解决训练消耗问题。

## 参考文献

- [1] Joakim Nivre and Mario Scholz. Deterministic Dependency Parsing of English Text[C]//Proceedings of COLING'04, Geneva, Switzerland. 2004, 64-70.
- [2] Hiroyasu Yamada and Yuji Matsumoto. Statistical dependency Analysis with Support Vector Machines[C]//Proceedings of the 8th International Workshop on Parsing Technologies, Nancy, France. 2003, 195-206.
- [3] Taku Kuto and Yuji Matsumoto. Japanese dependency analysis using cascaded chunking[C]//Proceeding of CoNLL, Taipei, Taiwan, 2002, 63-69.
- [4] Jason M. Eisner. Three new probabilistic models for dependency parsing: An exploration[C]//Proceedings of COLING, 1996, 340-345.
- [5] Y.J. Chu and T.H. Liu. On the shortest arborescence of a directed graph [J]. Science Sinica, 1965, (14): 1396-1400.
- [6] Ryan McDonald, Koby Crammer and Fernando Pereira. Online large-margin training of dependency parsers[C]//Proceedings of ACL, Ann Arbor, 2005, 91-98.
- [7] 辛霄, 范士喜, 王轩, 王晓龙. 基于最大熵的依存句法分析. 中文信息学报[J]. 2009, 23(2): 18-22.
- [8] 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[J]. 2006, 17(9): 1876-1883.
- [9] Yuchang Cheng. Machine Learning-based Dependency Analyzer for Chinese[C]//Proceedings of the International Conference on Chinese Computing, Singapore. 2005, 66-73.
- [10] Yuchang Cheng, Masayuki Asahara and Yuji Matsumoto. Chinese deterministic dependency analyzer: Examining effects of global features and root node finder[C]//Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing, Juje, Korea, 2005, 17-24.
- [11] Yun Xu, Feng Zhang. Using SVM to construct a Chinese dependency parser [J]. Journal of Zhejiang University SCIENCE A, 2006, 7(2): 199-203.
- [12] 段湘煜, 赵军, 徐波. 基于动作建模的中文依存句法分析. 中文信息学报[J]. 2007, 21(5): 31-35.
- [13] Ryan McDonald and Joakim Nivre. Characterizing the errors of data-driven dependency parsing models[C]//Proceedings of EMNLP-CoNLL, 2007, 122-131.
- [14] Joakim Nivre and Ryan McDonald. Integrating Graph-Based and Transition-Based Dependency Parsers[C]//Proceedings of ACL-08: HLT, Columbus, Ohio, USA, 2008, 950-958.
- [15] John C. Platt. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. In Advances in Large Margin Classifiers. MIT Press. 1999.