数据驱动的依存句法分析方法研究

李正华[1]，李渝勤[2]，刘　挺[1]，车万翔[1]

（1 哈尔滨工业大学 社会计算与信息检索研究中心，哈尔滨 150001；2 北京拓尔思信息技术股份有限公司，北京 100101）

摘　要： 依存句法分析是自然语言处理领域的核心研究课题。依存句法分析的目标是将输入的自然语言文本从序列形式转化为树状结构，从而刻画句子内部词语之间的句法关系。近年来，依存句法分析作为一个研究热点，取得了长足的发展，并且逐渐广泛应用于其他自然语言处理任务中。对前人提出的数据驱动的依存句法分析方法进行总结和比较，进而提出了依存句法分析未来的挑战。

关键词： 自然语言处理；依存句法分析；数据驱动

中图分类号： TP391.2　　**文献标识码：** A　　**文章编号：** 2095－2163(2013)05－0001－04

Research on Data－driven Dependency Parsing

LI Zhenghua[1], LI Yuqin[2], LIU Ting[1], CHE Wanxiang[1]

(1 Research Center for Social Computing and Information Retrieval, Harbin Institute of Technology, Harbin 150001, China;
2 Beijing TRS Information Technology Co., Ltd., Beijing 100101, China)

Abstract: Dependency parsing is one of the key research topics in the field of natural language processing (NLP). Given an input sentence, dependency parsing aims to convert the sequential input words into a tree structure, capturing the syntactic relations between word pairs inside the sentence. In the past few years, dependency parsing has become a hot topic and have made extensive progress. Meanwhile, dependency parsing is more and more widely used in other NLP tasks. This paper tries to surey existing data－driven dependency parsing approaches. Moreover, the paper proposes future directions and challenges for this field.

Key words: Natural Language Processing; Dependency Parsing; Data－driven

1　依存句法分析的定义

句法分析任务是对文本进行分析，将输入句子从序列形式变为树状结构，从而刻画句子内部词语之间的组合或修饰关系。这是自然语言处理领域的核心研究课题，已经广泛应用到其它自然语言处理任务中，如机器翻译、自动问答、信息抽取等。和其他句法分析形式如短语结构句法分析相比，依存句法分析具有形式简单、易于标注、便于学习、分析效率更高等优点[1,2]。另外，依存句法描述词和词之间的关系，因此更适合于表达非连续的、远距离的结构，这对于一些语序相对自由的西方语言非常重要。依存语法历史悠久，最早可能追溯到公元前几世纪 Panini 提出的梵文语法。依存语法存在一个共同的基本假设：句法结构本质上包含词和词之间的关系。这种关系称为依存关系(Dependency Relations)。一个依存关系连接两个词，分别是核心词(Head)和修饰词(Dependent)。依存关系可以细分为不同的类型，表示两个词之间的句法关系(Dependency Relation Types)。目前，依存语法标注体系已经为自然语言处理领域的许多专家和学者所采用，并应用于不同语言中，且对其不断地发展和完善。研究者们提出并实现了多种不同的依存分析方法，达到了较好的准确率。近年来，依存句法分析多已广泛用于统计机器翻译[3]、自动问答[4]和信息抽取[5]等任务，并取得了良好的效果。

依存句法分析任务的输入是一个已完成分词的自然语言句子。形式化地，输入句子可以表示为：$x = W_0 W_2 \cdots W_i \cdots W_n$，其中，$w_i$ 表示输入句子的第 i 个词；W_0 表示一个伪词，指向整个句子的核心词，也就是根节点(ROOT)。图1表示输入句子“刚满19岁的欧文现在效力利物浦队。”的依存树。

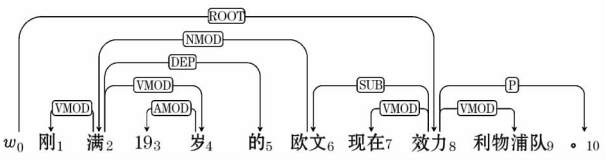

图1 依存树示例

Fig. 1 Example of a dependency parse

最一般地，一个依存句法树由多个依存弧构成，表示为：$d = \{(h,m,l) : 0 \leqslant h \leqslant n, 0 < m \leqslant n, l \in L\}$，其中一条有向弧 (h,m,l) 表示从核心词(Head) w_h 到依存词(Dependent) w_m 的依存弧，标签 l 表示依存弧的句法关系类型，L 是标注规范中定义的依存句法关系类型集合。依存弧的关系类型用来

收稿日期：2013－09－12
基金项目：国家科技支撑计划重点项目(2011BAH11B03)。
作者简介：李正华(1983－)，男，陕西合阳人，博士研究生，主要研究方向：自然语言处理、句法分析；
李渝勤(1963－)，女，四川重庆人，学士，副研究员，主要研究方向：中文信息处理、信息检索；
刘　挺(1972－)，男，黑龙江哈尔滨人，博士，教授，博士生导师，主要研究方向：信息检索、自然语言处理、社会计算；
车万翔(1980－)，男，黑龙江东宁人，博士，副教授，主要研究方向：自然语言处理、语义分析、句法分析。

表示两个词之间的句法或语义关系。例如,从“效力$_8$”到“欧文$_6$”的依存弧表示“效力$_8$”是核心词,“欧文$_6$”是依存词,关系类型 SUB 表示主语(Subject)。一棵合法的依存树需满足三个条件:(1)连通;(2)无环;(3)单核心。

依存句法分析的目标是给定输入句子 x,寻找分值(或概率)最大的依存树 d^*,具体公式为:

$$d^* = \text{argmac}_d \text{Score}(x, d)$$

因此,依存句法分析存在四个基本问题:

(1)如何定义 Score(x,d),即采用哪种方式将依存树的分值分解为一些子结构的分值。这是模型定义问题;

(2)采用哪些特征来表示每一部分子结构,即特征表示问题;

(3)如何获取特征的权重,即模型训练算法问题;

(4)给定模型参数,即已知特征的权重,如何搜索到分值最大的依存树。这是解码问题。

2 依存句法分析的方法

数据驱动的依存句法分析方法主要有两种主流的方法:基于图(Graph-based)的分析方法和基于转移(Transition-based)的分析方法。这两种方法从不同的角度解决这个问题。CoNLL 上的评测结果表明这两种方法各有所长,并且存在一定的互补性[2,4]。下面对各类方法展开细致分析。

2.1 基于图的依存句法分析方法

基于图的依存分析模型将依存句法分析问题看成从完全有向图中寻找最大生成树的问题。一棵依存树的分值由构成依存树的几种子树的分值累加得到。模型通过基于动态规划的解码算法从所有可能的依存树中搜索出分值最高的依存树。相关的研究工作主要包括:

(1)模型定义。根据依存树分值中包含的子树的复杂度,基于图的依存分析模型可以简单区分为一阶、二阶和三阶模型。一阶模型中,依存树的分值由所有依存弧的分值累加得到,即依存弧之间相互独立,互不影响[7]。二阶模型中,依存树的分值中融入了相邻兄弟弧(Sibling)和祖孙弧(Parent-child-grandchild)的分值[8,9]。三阶模型中,进一步增加了祖孙兄弟弧(Grandparent-parent-sibling)等三条依存弧构成的子树信息[10]。

(2)特征表示。在上述模型定义的基础上,研究人员也提出了相应的一阶、二阶、三阶子树特征[7-10]。每种子树特征考虑句子中的词语和词性信息、依存弧的方向和距离信息等。随着高阶子树特征的使用,依存句法分析模型的准确率也有较大幅度的提高。

(3)训练算法。基于图的依存分析方法通常采用在线训练算法(Online Training),如平均感知器算法(Averaged Perceptron)[11]、被动进取算法(Passive-Aggressive)[12]和 Margin Infused Relaxed 算法(MIRA)[13]。在线学习算法以迭代的方式训练特征的权重。一次迭代中遍历整个训练数据集合,每次根据一个训练实例的分析结果对当前的权重向量进行调整。

(4)解码算法。一阶模型对应的解码算法为 Eisner 算法[14]。Eisner 算法的本质是动态规划,不断合并相邻子串的分析结果,直到得到整个句子的结果,其时间复杂度为 O(n^3)。进而,McDonald 和 Pereira(2006)对 Eisner 算法进行扩展,增加了表示相邻兄弟节点的数据类型,时间复杂度仍为 O(n^3)。Carreras(2007)同样对 Eisner 算法进行扩展,得到面向二阶模型的基于动态规划的解码算法,时间复杂度为 O(n^4)。Koo 和 Collins(2010)提出了面向三阶模型的解码算法,时间复杂度为 O(n^4)。一些研究者提出采用基于柱搜索的解码算法,允许模型方便地融入更高阶的解码算法,同时保证较低的时间复杂度[15,16]。

2.2 基于转移的依存句法分析方法

基于转移的依存分析模型将依存树的搜索过程建模为一个动作序列,将依存分析问题转化为寻找最优动作序列的问题。模型通过贪心搜索或者柱搜索的方式找到近似最优的依存树。其优点在于可以充分利用已形成的子树信息,从而形成丰富的特征,以指导模型决策下一个动作。相关的研究工作主要包括:

(1)模型定义。基于转移的依存句法分析方法提出早期,研究者们使用局部分类器(如最大熵分类器)决定下一个动作,选择概率最大的动作[17,18]。这样,一个依存树的概率由其对应的动作序列中每一个动作的概率累乘得到。近年来,研究者们采用线性全局模型来决定下一个动作,一个依存树的分值为对应动作序列中每一个动作的分值的累加[19-21]。

(2)特征表示。基于转移的依存句法分析方法的优势在于可以充分使用已构成的子树信息。Zhang 和 Nivre(2011)在前人工作的基础上,提出了丰富的特征集合,如三阶子树特征,词的配价信息等[21]。

(3)训练算法。早期,研究者们在训练语料上训练出一个局部分类器,在解码过程中重复使用,决定下一个动作。通常采用的分类器有基于记忆的分类器、支持向量机等。近年研究发现采用全局线性模型可以提高句法分析的准确率,通常采用平均感知器在线训练算法。

(4)解码算法。其任务是找到一个概率或分值最大的动作序列。早期采用贪心解码算法,即每一步都根据当前状态,选择并执行概率最大的动作,进入到下一个状态。如此反复直至达到接收状态,形成一棵合法的依存树[17,18]。进而,研究者们提出使用柱搜索的解码方式扩大搜索空间,即同时保留多个分值最高的状态,直到搜索结束时选择最优的动作路径[22,19]。Huang 和 Sagae(2010)提出在柱搜索中加入动态规划,通过合并等价状态进一步扩大搜索空间[20]。随着搜索空间的增大,依存句法分析的准确率有显著提高。

2.3 模型融合的方法

基于图的方法和基于转移的方法从不同的角度解决问题,各有优势。基于图的模型进行全局搜索但只能利用有限的子树特征,而基于转移的模型搜索空间有限但可以充分利用已构成的子树信息构成丰富的特征。McDonald 和 Nivre(2011)通过详细比较发现,这两种方法存在不同的错误分布。因此,研究者们使用不同的方法融合两种模型的优势,常见的方法有:stacked learning[2,23];对多个模型的结果加权

后重新解码[24,25]；从训练语料中多次抽样训练多个模型(Bagging)[26,27]。

2.4　词性标注和依存句法分析联合模型

依存句法分析模型中，词性是非常重要且有效的特征。如果只使用词语特征，会导致严重的数据稀疏问题。自然语言处理中，词性标注和依存句法分析这两个问题通常被当成两个独立的任务，以级联的方式实现。即对于一个输入句子，假定其分词结果已知，先对句子进行词性标注，然后在词性标注结果的基础上进行依存句法分析。这种级联的方法会导致错误蔓延。也就是说，词性标注的错误会严重影响依存分析的准确率。由于汉语缺乏词形变化信息(如英语中的词后缀变化如 - ing，- ed，- es，- ly 等)，因此汉语的词性标注比其他语言如英语更具挑战性。近年来，研究者们通过建立词性标注和依存句法分析联合模型，在同一个模型中解决这两个紧密相关的任务，允许词性信息和句法结构互相影响和帮助，取得了不错的效果。一方面，联合模型中，句法信息可以用来指导词性标注，从而帮助解决一部分需要句法结构才能够消解的词性歧义。另一方面，更准确的词性标注，也可以反过来帮助依存分析。Li 等通过扩展基于图的依存句法分析模型，首次提出汉语词性标注和依存句法分析联合模型[28]，并且提出了适用于联合模型的训练算法[29]，显著提高了词性标注和依存句法分析的准确率。进而，一些研究者们提出基于转移的词性标注和依存句法分析联合模型[30,31]。Ma 等(2012)尝试了基于 Easy - first 的汉语词性标注和依存句法分析联合模型[32]。

2.5　基于多树库融合的方法

对于统计的数据驱动的分析模型而言，标注数据的规模很大程度上影响着分析结果的准确率。依存句法分析是一种结构化分类问题，比二元分类和序列标注问题更具挑战性，因此依存句法分析更容易受到数据稀疏问题的影响，树库规模对依存句法分析的准确率影响很大。然而，标注树库是一件艰巨的任务，通常需要耗费很大的人力和物力。目前的研究结果表明在一个树库上训练出的句法分析的模型似乎很难进一步提高句法分析的准确率。然而，汉语存在多个树库。这些树库由不同的组织或机构标注，遵循不同的标注规范，面向不同的应用。尽管各个树库遵循不同的标注规范，但却都是根据人们对汉语语法的理解而标注，因此包含很多共性的标注结构。同时，不一致的标注结果应该也是有规律可循的。所以，一些研究者们尝试同时利用多个树库，帮助句法分析的准确率。李正华等(2008)曾尝试统计和规则相结合的方法，将短语结构的源树库 CTB 转化为符合 CDT 标注规范的依存结构，然后将转化后的树库和 CDT 合并，提高训练数据的规模，以提高依存句法分析准确率[33]。Niu 等(2009)提出一种基于统计的树库转化方法，将依存结构的 CDT 树库转化为满足 CTB 标注规范的短语结构树库，进而使用语料加权的方式增大训练树库的规模，提高了短语结构句法分析的性能[34]。Li 等(2012)提出一种基于准同步文法的多树库融合方法，不是直接将转化后的树库作为额外的训练数据，而是使用准同步文法特征增强依存句法分析模型，从而柔和地学习标注规范中规律性的不一致，提高依存句法分析的准确率[35]。

3　依存句法分析面临的挑战

自从 2006 年开始，CoNLL 国际评测一直关注依存句法分析，不但提供了多语言、高质量的树库，并通过对各种方法的比较分析，让研究者们对依存分析问题的理解更加清晰，极大地促进了依存句法分析的发展。依存分析已经成为自然语言处理的一个热点问题，方法也越来越成熟，并且在许多领域得到了应用。然而，目前依存句法分析还存在很多挑战，这些挑战也可能是未来依存分析发展的趋势。具体分析如下：

(1) 提高依存分析准确率。目前主流的两种依存分析方法都存在一定的缺陷。基于图的方法很难融入全局特征。而基于转移的方法虽然原理上可以利用丰富的特征，但是实际使用的特征还是属于局部特征，另外也还存在错误级联的问题(柱搜索只能缓解这个问题)。融合不同依存分析模型的方法可以提高分析性能，但是提高幅度比较有限。研究可知，只有从新的角度理解这个问题本身，提出新的建模方法，或者应用新的机器学习方法，才有望大幅度提高依存分析性能。一些学者提出的利用未标注数据帮助依存分析模型是一个很好的思路，值得深入研究。

(2) 提高依存分析效率。基于图的依存分析方法融入高阶特征可以提高性能，但是效率很低，无法适应实际应用的需求。在不明显降低分析性能的前提下，如何提高依存分析效率也是一个很有实际价值的问题。

(3) 领域移植问题。研究发现，当训练数据领域与测试数据领域不相同时，即使差距不大，也会导致句法分析性能下降很大。以英语为例，从华尔街日报树库移植到 Brown 语料时，句法分析性能下降近 8%。目前依存树库所覆盖的领域、规模都很有限，而标注树库的代价很大。因此解决领域移植问题，对于依存分析的实际应用至关重要。

(4) 语言相关的依存分析。目前最主流的两种依存分析方法都是语言无关的，纯粹依靠机器学习方法从数据中学习，加入人类知识只能限于特征选择。然而，每种语言都有其特点。因此语言相关的依存分析研究，如针对每种语言的特点设计更有效的模型和算法，利用一些语言特有的资源等，也是很有必要的。近年来，国内学者已经在汉语依存句法分析上做出了很多成绩，然而如何利用汉语的特点，提高汉语句法分析的准确率和效率，仍然是一个开放的问题。

4　结束语

本文对数据驱动的依存句法分析方法进行深入调研和总结。主流的依存句法分析方法大致可以分为两类。一类是基于图的方法，另一类为基于转移的方法。两种方法从不同的角度解决依存句法分析问题，都取得了较好的效果。进而，研究者们提出各种融合方法，尝试使得两类方法扬长补短，各取优势。另外，本文还探讨了依存句法分析和底层词性标注任务联合求解，以及利用多个树库提高依存句法分析准确率的相关工作。最后，本文展望了依存句法分析的未来研究方向和可能的挑战。

参考文献:

[1] 马金山. 基于统计方法的汉语依存句法分析研究 [D]. 哈尔滨: 哈尔滨工业大学,2003.

[2] MCDONALD R, NIVRE J. Analyzing and integrating dependency parsers [J]. Computational Linguistics, 2011, 37(1): 197 – 230.

[3] DING Yuan, PALMER M. Machine translation using probabilistic synchronous dependencyuinsertion grammars [C] //Proceedings of ACL, 2005: 541 – 548.

[4] PARK J H, CROFT W B. Query term ranking based on dependency parsing of verbose queries [C] //Proceedings of SIGIR, 2010: 829 – 830.

[5] CULOTTA A, SORENSEN J. Dependency tree kernels for relation extraction [C] //Proceedings of ACL, 2004.

[6] BUCHHOLZ S, MARSI E. CoNLL – X shared task on multilingual dependency parsing [C] // Proceedings of CONLL – X, 2006: 149 – 164.

[7] MCDONALD R, CRAMMER K, PEREIRA F. Online large – margin training of dependency parsers [C] //Proceedings of ACL, 2005: 91 – 98.

[8] MCDONALD R, PEREIRA F. Online learning of approximate dependency parsing algorithms [C] // Proceedings of EACL, 2006: 81 – 88.

[9] CARRERAS X. Experiments with a higher – order projective dependency parser [C] //Proceedings of the CoNLL Shared Task Session of EMNLP – CoNLL, 2007: 957 – 961.

[10] KOO T, COLLINS M. Efficient third – order dependency parsers [C] //Proceedings of ACL, 2010: 1 – 11.

[11] COLLINS M. Discriminative training methods for hidden Markov models: theory and experiments with perceptron algorithms [C] // Proceedings of EMNLP, 2002.

[12] CRAMMER K, DEKEL O, KESHET J, et al. Online passive aggressive algorithms [C] // Proceedings of NIPS, 2003.

[13] CRAMMER K, SINGER Y. Ultraconservative online algorithms for multiclass problems [J]. Journal of Machine Learning Research, 2001 (3).

[14] EISNER J. Three new probabilistic models for dependency parsing: An exploration [C] // Proceedings of COLING, 1996: 340 – 345.

[15] 李正华, 车万翔, 刘挺. 基于柱状搜索的高阶依存句法分析 [J]. 中文信息学报, 2010, 24(1): 37 – 41.

[16] ZHANG Hao, MCDONALD R. Generalized higher – order dependency parsing with cube pruning [C] //Proceedings of EMNLP, 2012: 320 – 331.

[17] YAMADA H, MATSUMOTO Y. Statistical dependency analysis with support vector machines [C] //Proceedings of IWPT, 2003: 195 – 206.

[18] NIVRE J. An efficient algorithm for projective dependency parsing [C] //Proceedings of IWPT, 2003: 149 – 160.

[19] ZHANG Yue, CLARK S. A tale of two parsers: Investigating and combining graph – based and transition – based dependency parsing [C] //Proceedings of EMNLP, 2008: 562 – 571.

[20] HUANG Liang, SAGAE K. Dynamic programming for linear – time incremental parsing [C] // Proceedings of ACL, 2010: 1077 – 1086.

[21] ZHANG Yue, NIVRE J. Transition – based dependency parsing with rich non – local features [C] // Proceedings of ACL, 2011: shortpapers: 188 – 193.

[22] DUAN Xiangyu, ZHAO Jun, XU Bo. Probabilistic parsing action models for multi – lingual dependency [C] //Proceedings of the CoNLL Shared Task Session of EMNLP – CoNLL, 2007: 940 – 946.

[23] MARTINS A F T, DAS D, SIMITH N A, et al. Stacking dependency parsers [C] //Proceedings of EMNLP, 2008: 157 – 166.

[24] SAGAE K, LAVIE A. Parser combination by reparsing [C] //Proceedings of NAACL, 2006: 129 – 132.

[25] SURDEANU M, MANNING C D. Ensemble models for dependency parsing: cheap and good? [C] // Proceedings of NAACL, 2010.

[26] SUN Weiwei. Learning Chinese Language Structures with Multiple Views [D]. Saarland University, 2012.

[27] ZHANG Meishan, CHE Wanxiang, LIU Yijia, et al. HIT dependency parsing: Bootstrap aggregating heterogeneous parsers. In Notes of the First Workshop on Syntactic Analysis of Non – Canonical Language (SANCL), 2012.

[28] LI Zhenghua, ZHANG Min, CHE Wanxiang, et al. Joint Models for Chinese POS Tagging and Dependency Parsing [C] //Proceedings of EMNLP, 2011: 1180 – 1191.

[29] LI Zhenghua, ZHANG Min, CHE Wanxiang, et al. A separately passive – aggressive training algorithm for joint POS tagging and dependency parsing [C] //Proceedings of COLING, 2012: 1681 – 1698.

[30] HATORI J, MATSUZAKI T, MIYAO Y, et al. Incremental joint POS tagging and dependency parsing in Chinese [C] //Proceedings of IJCNLP, 2011: 1216 – 1224.

[31] BOHNET B, NIVRE J. A transition – based system for joint part – of – speech tagging and labeled non – projective dependency parsing [C] //Proceedings of EMNLP, 2012: 1455 – 1465.

[32] MA Ji, XIAO Tong, ZHU Jingbo, et al. Easy – first Chinese POS tagging and dependency parsing [C] // Proceedings of COLING, 2012: 1731 – 1746.

[33] 李正华, 车万翔, 刘挺. 短语结构树库向依存结构树库转化研究 [J]. 中文信息学报, 2008, 22(6): 14 – 19.

[34] NIU Zhengyu, WANG Haifeng, WU Hua. Exploiting heterogeneous treebanks for parsing [C] // Proceedings of ACL, 2009: 46 – 54.

[35] LI Zhenghua, CHE Wanxiang, LIU Ting. Exploiting multiple treebanks for parsing with Quasi – synchronous Grammar [C] //Proceedings of ACL, 2012: 675 – 684.