计算机科学2002Vol.29№.10

基于双向策略标记上下文无关文法的图算法*)

BSCFG-Based Bi-directional Parsing

周雅倩　黄萱菁　吴立德

(复旦大学计算机科学与工程系　上海200433)

Abstract Bi-directional strategy-marked context free grammar (BSCFG) is a more flexible grammar, but it is not always complete. This paper proposes an algorithm to make any incomplete BSCFG complete, and proves its correctness. Then we develop a BSCFG-based bi-directional chart parser. Experiment indicates a promotion of 23.4% at speed.

Keywords Context free grammar, Bi-directional strategy-marked context free grammar, Bi-directional chart parsing

一、引言

句法分析是自然语言处理的重要基础，相应的研究很多，但是由于速度问题，句法分析很难大规模运用，因此提高分析速度一直是句法分析的重要研究方向。我们知道，句法分析的过程实际上是一个在句法成分树的空间中寻找最优分析树的过程，因此可以把分析过程看成是个搜索最佳路径的过程。在搜索的过程中给些"启发"将提高效率。双向图算法[1]，是个最佳的例证，由于每条规则都有触发类来规定它的使用时机，使得它无论是时间还是空间上都比传统的图算法有更高的效率。

说到句法分析，总要涉及文法。上下文无关文法，由于其表示的清晰性及简单性，被广泛应用于自然语言处理中。基于上下文无关文法的分析方法有很多，有些自底向上(例如：移进归约[2,3])，有些自顶向下(例如：递归向下[2])。LC(Left-Corner 的缩写)[4]分析方法可以看成是这两种策略的结合，这种结合体现在预测上，文[4]使用这种方法取得了显著的效率。以上这些方法，都假设对句法规则统一操作：要么都自底向上，要么都自顶向下。文[5]提出了双向策略标记上下文无关文法(bi-directional strategy-marked context free grammar，简称 BSCFG)[5]，使纵双向(上下)成为可能，但是它同时指出并非所有的 BSCFG 是完备的。在本文中，我们将提出一种算法，对给定的 BSCFG 进行规范化，使之完备(所谓完备是指基于 BSCFG 的分析器能分析所有基于 CFG 的分析器能分析的句子)，并且实现了基于 BSCFG 的双向图算法，取得了提高速度23.4%的实验结果。

下面给出 BSCFG 的形式定义和基于 BSCFG 的分析方法；然后给出 BSCFG 规范化的算法和完备性证明；并列出实验结果；最后是总结。

二、语法表示及分析算法

2.1　形式定义

双向策略标记上下文无关文法[5]以上下文无关文法为骨架，加上规则的触发信息，定义为：

$G=(V_N,V_T,P,S,T)$

其中，V_N,V_T,P,S 的含义与上下文无关文法中的定义相同，分别为非终结符集、终结符集、规则集和开始符；T 为规则集的触发类集。对规则集中的每条规则 $r:A_0\rightarrow A_1A_2\cdots A_k$，$T$ 满足以下两个条件：

1. $T(r)\neq\Phi$；

2. $\forall\ j\in T(r),0\leqslant j\leqslant k$。

其中 $T(r)$ 为规则 r 的触发类集，$\bigcup_r T(r)=T$。$0\in T(r)$ 表示规则 r 的触发类可以是左部的类，也就是规则 r 可自顶向下分析；$j>0\ \&\ j\in T(r)$ 表示规则 r 的触发类可以是右部第 j 个类。

特别是，当 $\forall\ r\in P,\forall\ t\in T(r):t=0$ 时，文法完全自顶向下分析；当 $\forall\ r\in P,\forall\ t\in T(r)1:t=1$ 时，文法完全自底向上且从左向右分析，即：每条规则都是从它右部的第一个类开始分析；当 $\forall\ r\in P,\forall\ t\in T(r):t=k>0$ 时，文法完全自底向上且双向分析，双向图分析器使用的就是这种语法。

由于当一条规则有多个触发类时，只会引进更多的活动弧；而没有触发类的规则是无法被激发使用的。故本文规定每一条规则有且只有一个触发类，相应地 $T(r)$ 表示一个数：$0\leqslant T(r)\leqslant k$，而不是数集。

2.2　概念和符号的说明

·规则右(左)触发指的是规则的触发类在该规则的右(左)边。右(左)触发规则指的是触发类在右(左)边的规则。规则的右(左)触发类指的是右(左)触发规则的触发类。

·大写字母(可加下标)$\in V_T\cup V_N$；$\alpha,\beta,\gamma,\delta\in(V_T\cup V_N)^*$。

·规则用 $A\rightarrow\alpha,A_0\rightarrow A_1A_2\cdots A_k$ 等的形式表示。

·在分析图中弧表示为：$(A\rightarrow\alpha\cdot\beta\cdot\gamma,i,j)$。$A\rightarrow\alpha\beta\gamma$ 表示弧使用的规则，其中两个点表示规则中已被分析的成分的位置，i,j 表示规则中已被分析的成分在输入句中所对应语段的首尾坐标。完成弧可简略表示为(X,i,j)，X 为这条弧的类。

2.3　分析算法

基于 BSCFG 的分析算法与传统双向图算法[1]类似。不同的是，在初始和扩展时分别增加：

1. 在初始时，对于规则库中所有以开始符 S 为左触发类的规则，把$(S\rightarrow\cdot\cdot\alpha,0,0)$和$(S\rightarrow\alpha\cdot\cdot f,n,n)$加入图中。其中，$n$ 为输入句的长度。

2. 在扩展时，对于所有左边需求 B 的活动弧$(A\rightarrow\alpha B\cdot\beta\cdot$

*)国家自然科学基金(项目号：69873011)、863计划(项目号：863-306-ZD02-02-4)项目。周雅倩　硕士研究生，主要研究方向是自然语言处理。黄萱菁　副教授，主要研究方向是自然语言处理。吴立德　教授，博士生导师，主要研究方向是自然语言处理和计算机视觉。

γ,i,j)和所有以 B 为左触发类的规则 $B\rightarrow\delta$，把($B\rightarrow\cdot\cdot\delta,i,j$)加入图中；对于所有右边需求 B 的活动弧($A\rightarrow\alpha\cdot\beta\cdot B\gamma,i,j$)和所有以 B 为左触发类的规则 $B\rightarrow\delta$，把($B\rightarrow\cdot\cdot\delta,j,j$)加入图中。

2.4 不完备性

然而，并不是所有的 BSCFG 都是完备的[5]。图1的文法就是不完备的。

1. $A\rightarrow C\#BC$
2. $\#A\rightarrow DB$
3. $B\rightarrow\#EA$
4. $\#B\rightarrow CF$
5. $C\rightarrow\#F$

图1 双向策略标记上下文无关文法 G_S 的规则集

这个文法中，字母 A B C 为非终结符，字母 D E F 为终结符，A 为开始符号，前面打 # 的符号为触发类。规则1只有当 B 生成时才可使用；而规则4只有当输入中有词串 ED 时才可使用(因为：E 的输入触发规则3，然后触发规则2，接着由于 D 的输入，才触发规则4)。

对地输入“$FFFF$”，用 G_S 是无法分析的，因为 F 只能触发规则5，生成 C，而 C 无法触发任何一条规则。然而“$FFFF$”是可以用 G_S 对应的 CFG 来归约到开始符 A 的，所以 G_S 不完备。

三、规范化算法及完备性证明

3.1 BSCFG 规范化算法

我们发现引起 BSCFG 不完备的原因是一个类既作为一些规则的右触发类同时又作为另一些规则的左触发类。因此设想，当左触发类集和右触发类集的交集为空时，BSCFG 是完备的。下面将基于这个设想，给出一个 BSCFG 规范化的算法，并且证明用这个算法规范后的 BSCFG 是完备的。

算法思想 改写同时满足下面三个条件的规则 r，使其不同时满足这三个条件：1、r 右触发(触发类为 C)；2、存在以 C 为左部的规则 r'；3、r' 左触发。

途径 改变 r 的触发类：1、使 r 仍然右触发；2、若途径1行不通，使 r 左触发。

对于一个上下文无关文法，右部触发类的初始标定可用文[6]中的方法。一般认为，那些在句中出现频率高的、词性变化难确定的(即：用标注器预标词性时易出错的)、用法灵活的词或短语最好用自顶向下分析。这些特点是可以用统计的方法来确定的。

对图1的例子，规范化后的结果见图2。图3是用图2中的文法分析“$FFFF$”分析树，其中箭头表示触发。若句子能被文法分析，则在分析树中，每个非终结符都有且只有一个箭头指向它。

1. A→#C B C
2. #A→D B
3. B→#E A
4. #B→C F
5. C→#F

图2 图1例子规范化的结果

这个文法就可以分析“$FFFF$”的过程为：($F,i,i+1$)触发规则5，生成($C,i,i+1$)，($i=0,1,2,3$)。由($C,0,1$)触发规则1，接着触发规则4，由于($C,1,2$)和($F,2,3$)的存在，生成($B,1,3$)。最后由($C,0,1$)、($B,1,3$)和($C,3,4$)，生成($A,0,4$)。

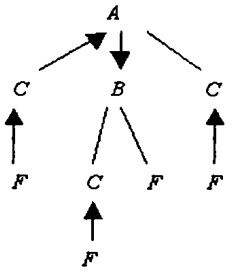

图3 输入“FFFF”的分析树

3.2 概念和符号的说明

为以下证明的方便，先就将要涉及的一些概念和符号作以下说明：

·基于 BSCFG 的双向图算法的分析图中弧表示为：($\#A\rightarrow\alpha\cdot\beta\cdot\gamma,i,j$)或($A\rightarrow\alpha\cdot\beta\#B\gamma\cdot\delta,i,j$)。其中 # 后面的是触发类，若左部类不标 #，则表示触发类在右部。下文证明中的图指的是分析结束后的图。

·CFG 分析树的结点表示：$\langle A_0\rightarrow A_1A_2\cdots A_k,i,j\rangle$。$A_0\rightarrow A_1A_2\cdots A_k$ 表示结点使用的规则，i,j 是结点在输入句中对应语段的首尾坐标。

·根据 CFG 分析树，我们可以建立对应的 BSCFG 的分析树(结点与 CFG 分析树一一对应)，它的结点表示：$\langle\#A_0\rightarrow A_1A_2\cdots A_k,i,j\rangle$(称为左触发结点)或$\langle A_0\rightarrow A_1A_2\cdots\#A_i\cdots A_k,i,j\rangle$(称为右触发结点)。下文中的分析树指的都是 BSCFG 分析树。

3.3 完备性证明

规范化的结果是右触发类集合和左触发类集合的交为空，形式化表示为：

$\{A_i|r:A_0\rightarrow\cdots A_i\cdots,i\in T_r(r),r\in P\}\cap$

$\{A_0|r:A_0\rightarrow\alpha,0\in T_r(r),r\in P\}=\Phi$

可以证明，规范化后的 BSCFG 是完备的：任何一个输入句，若它能被 CFG 归约，那么它也能被规范化后的 BSCFG 归约。为证明完备性，首先证明引理1—4。

引理1 若在图中存在的活动弧($\#A\rightarrow\ddot{\alpha},i,i$)，且在分析树中以$\langle\#A\rightarrow\alpha,i,j\rangle$为根的子树下，所有右触发的结点在图中有完成弧，那么在图中存在完成弧($\#A\rightarrow\cdot\alpha\cdot,i,j$)。

证明：在图算法中，分析树中的叶子就是图中的完成弧。

由于分析树中所有的右触发的结点在图中都有完成弧，那么完全可以把分析树中所有右触发的结点当成叶子结点，这时分析成了完全自顶向下。自顶向下分析中只要分析过程中需要某类，则必定能产生对这个类的归约[2]。如图4中，完全可以把结点$\langle A''\rightarrow\alpha'',i'',j''\rangle$当成叶子，而不必管它下面的结点的组织。

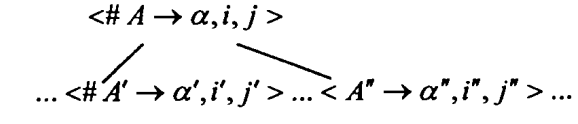

图4 结点$\langle\#A\rightarrow\alpha,i,j\rangle$的分析树

那么若在图中存在弧($\#A\rightarrow\cdot\cdot\alpha,i,j$)，则必然在图中存在弧($\#A\cdot\alpha\cdot,i,j$)。

引理2 若在图中存在活动弧($\#A\rightarrow\cdot\cdot\alpha,i,j$)，且在分析

树中以〈#$A\rightarrow\cdot,a,i,j$〉为根的子树下，所有右触发的结点在图中有完成弧，那么在图中存在完成弧(#$A\rightarrow\cdot\alpha\cdot,i,j$)。

证明：由引理1的证明知，引理2是显然成立的。

引理3 若分析树中以〈$A_0\rightarrow A_1A_2\cdots\#A_t\cdots A_k,i,j$〉为根的子树下，所有右触发的结点在图中有完成弧，那么在图中存在完成弧〈$A_0\rightarrow\cdot A_1A_2\cdots\#A_t\cdots A_k\cdot,i,j$〉。

证明：分析树如图5表示，其中 $i=i_1,j_m=i_{m+1}(m=1,2,\cdots,k-1),j_k=j$。

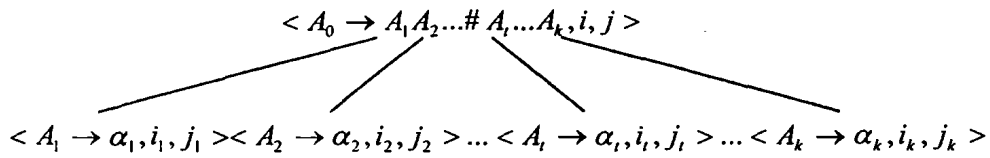

图5 结点〈$A_0\rightarrow A_1A_2\cdots\#A_t\cdots A_k,i,j$〉的分析树

因为规范化的结果是，作为右触发类的类不会作为任何规则的左触发类，所以任何以 A_t(规则 $A_0\rightarrow A_{A2}\cdots\#A_t\cdots A_k$ 的右触发类)为左部的规则必定右触发，那么由条件知，在图中存在类 A_t 的完成弧($A_t\rightarrow\cdot\alpha_t\cdot,i_t,j_t$)，由触发规则 $A_0\rightarrow A_1A_2\cdots\#A_t\cdots A_k$，得弧($A_0\rightarrow A_1A_2\cdots\#A_t\cdot\cdots A_k,i_t,j_t$)。

那么若 $A_{t-1}\rightarrow\alpha_{t-1}$右触发，由条件知，在图中有相应完成弧($A_{t-1}\rightarrow\cdot\alpha_{t-1}\cdot,i_{t-1},j_{t-1}$)，那么得弧($A_0\rightarrow A_1A_2\cdots\cdot A_{t-1}\#A_t\cdots A_k,i_{t-1},j_t$)。

若 $A_{t-1}\rightarrow\alpha_{t-1}$左触发，那么由分析算法的扩展(参阅2.3节)知，在图中有弧(#$A_{t-1}\rightarrow\alpha_{t-1}\cdot,i_t,i_t$)，则由引理2知，在图中有它的完成弧(#$A_{t-1}\rightarrow\cdot\alpha_{t-1}\cdot,i_{t-1},i_t$)，那么也得弧($A_0\rightarrow A_1A_2\cdots\cdot A_{t-1}\#A_t\cdot\cdots A_k,i_{t-1},j_t$)。

由以上两段的证明知，弧($A_0\rightarrow A_1A_2\cdots\#A_t\cdot\cdots A_k,i_t,j_t$)必然可以扩展到完全弧($A_0\rightarrow\cdot A_1A_2\cdots\#A_t\cdots A_k\cdot,i,j$)。

引理4 分析树中所有右触发的结点，在图中都有相应的完成弧。

证明：设树 T 的高度为 $H(T)$，以下用归纳法证明：

1. $H(T)=1$时，T 为叶子结点，在图中显然有相应的完成弧。

2. 假设分析树中所有高度小于 n 的右触发的结点，在图中有相应的完成弧，那么我们要证的是所有高度等于 n 的右触发的结点，在图中有相应的完成弧。由引理3知，显然。

3. 综上1，2，原命题得证。

定理 规范化后的 BSCFG 是完备的(可由分析算法来分析)。

也就是要证明：任何分析树中所有的结点在图中有对应的完成弧。

证明：1. 由引理4知若分析树根结点右触发，那么在图中有它相应的完成弧。

2. 由分析算法的初始步骤(参阅2.3节)知，若分析树根结点左触发，那么存在活动弧($S\rightarrow\cdot\cdot\alpha,0,0$)(或($S\rightarrow\cdot\cdot\alpha,n,n$))，由引理4知所有右触发的结点在图中有相应的完成弧，所以根据引理1(引理2)，分析树根结点在图中有相应的完成弧。

综上1，2，分析树的根结点在图中有对应的完成弧。

所以所有结点在图中有对应的完成弧，因为不然，根结点的完成弧是不可能生成的。

四、实验

我们用基于 BSCFG 的双向图算法，做了一个实验。实验使用包含230多条规则的一个语法，测试集合是200个英文句子，平均长度约为9个单词。以下是186句完全分析的实验结果：

表1 实验结果

算　法	分析终止条件	时间(毫秒)
基于 CFG 的双向图算法	条件1	1011
	条件2	964
基于 BSCFG 的双向图算法	条件1	797
	条件2	738

两个算法使用的规则集完全相同，所不同的是基于 CFG 的双向图使用文[6]中的方法决定触发类，而基于 BSCFG 的双向图算法在这个结果上，令空规则和关于动词的规则左触发，然后用本文的规范化算法进行规划化。分析终止条件1指的是得到所有的分析结果时才结束分析；条件2指的是得到第一个分析结果时就结束分析。显然，只有在终止条件2下，弧的竞争机制才对效率有贡献，使用基于 BSCFG 的双向图算法使分析时间减少23.4%。

小结和展望 本文给出了规范化 BSCFG 的算法，并且证明了使用这种规范算法规范后的 BSCFG 是完备的。在这个基础上，我们实现了基于 BSCFG 的双向图算法，取得23.4%的速度提高。

当我们使用词性和规则的统计信息来预测弧的生成时，将大大提高分析的效率。这也将更充分体现基于 BSCFG 的双向图算法的高效性和灵活性。我们也可用文法分割[7]的方法：当一条活动弧需要某个类时，可调用此类的子分析器，这个子分析器可以使用任何一种分析方法，所以这将是一种高效的方法。

参考文献

1 Gibson E. Bidirectional Active Chart Parsing. CMU-CMT-93-139, 1993

2 陈火旺，钱家骅，孙永强. 程序设计语言编译原理. 国防工业出版社，1984

3 Tomita M. Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems. Kluwer Academic Publishers, 1986

4 Moore R C. Improved Left-Corner Chart Parsing for Large Context-Free Grammars. In: Proc. of IWPT2000

5 Ritche G. Completeness Conditions for Mixed Strategy bidirectional Parsing. Comutational Linguastic, 1999. 457～486

6 吴立德，等. 大规模中文文本处理. 上海：复旦大学出版，1997

7 Weng F, Meng H. Po Chui Luk, Parsing a Lattice with Multiple Grammars. In: Proc. of IWPT2000