Complements and Adjuncts in Dependency Grammar Parsing Emulated by a Constrained Context-Free Grammar

Tom B.Y. Lai
Dept. of Chinese, Translation and Linguistics
City University of Hong Kong
Tat Chee Avenue, Kowloon
Hong Kong
Dept. of Computer Science and Technology
Tsinghua University, Beijing
cttomlai@cityu.edu.hk

Changning Huang
Dept. of Computer Science and Technology
Tsinghua University, Beijing
Beijing 100084
China
hcn@mail.tsinghua.edu.cn

Abstract

Generalizing from efforts parsing natural language sentences using the grammar formalism, Dependency Grammar (DG) has been emulated by a context-free grammar (CFG) constrained by grammatical function annotation. Single-headedness and projectivity are assumed. This approach has the benefit of making general constraint-based context-free grammar parsing facilities available to DG analysis. This paper describes an experimental implementation of this approach using unification to realize grammatical function constraints imposed on a dependency structure backbone emulated by a context-free grammar. Treating complements of a head word using subcategorization lists residing in the head word makes it possible to keep phrase-structure-rule-like mechanisms to the minimum. Adjuncts are treated with a syntactic mechanism that does not reside in the lexicon in this generally lexical approach to grammar.

Introduction

The mathematical properties of Dependency Grammar (Tesniere (1959)) are studied by Gaifman (1965) and Hays (1964). Following their footsteps, Robinson (1970) formulates four axioms to govern the well-formedness of dependency structures:

(a) One and only one element is independent;
(b) All others depend directly on some element;
(c) No element depends directly on more than one other;
(d) If A depends directly on B and some element C intervenes between them (in linear order of string), then C depends directly on A or on B or some other intervening element.

These axioms require that all words should depend on only one word and that, arranging words in linear order, crossing of dependency links as in Fig.1 should not be allowed.

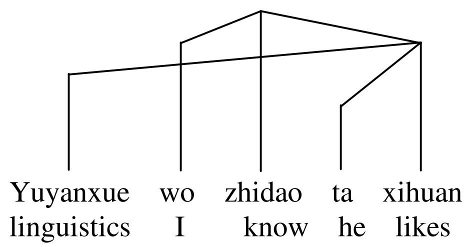

Fig.1

These are effectively the requirements of single-headedness and projectivity.

While there are some schools of DG that do not follow Robinson's axioms in their entirety (e.g. Hudson (1984, 1990), Melcuk (1988)), many computational linguists working on DG-based parsing have based their work on these assumptions (e.g. Hellwig (1986), Covington (1990)). DG parsing of Chinese have used statistical corpus-based algorithms (Huang et al. (1992), Yuan and Huang. (1992)), rule-based algorithms (Zhou and Huang (1994)) and mixed algorithms (Li et al. (1993)). They may work with syntactic categories or with the words directly; they may or may not take word order into consideration; but they all observe Robinson's axioms in their entirety. They also label dependency relations with grammatical functions like *subject* and *object*.

Generalizing over DG-based parsing of Chinese, Lai and Huang (1994) note that, taking linear word-order into consideration, this approach to DG can be emulated by a model having a context-free constituent component that is constrained by a grammatical function component, very much in the spirit of the constituent and functional structures of Lexical Functional Grammar (LFG, Bresnan ed. (1982)). The syntactic dependency structure in this approach to DG is however different from context-free phrase structure in that non-lexical phrasal nodes are not allowed. As in LFG, the grammatical function structure, which provides the constraining mechanism, is mathematically a graph rather than a tree. This relieves the syntactic dependency component of any need to be multiple-headed and non-projective (Lai and Huang 1995). Following this approach, Lai and Huang (in press) describes a unification-based (Shieber (1986)) experimental parser adapted from the PATR parser in Gazdar and Mellish (1989). Control and simple semantic analysis are handled.

The present paper discusses issues of using a constrained CFG to emulate DG. Section 1 explains the implications of Robinson's axioms and describes Hays' CFG-like formulation of dependency rules. Section 2 formulates the "dependency rule with grammatical function annotation" model and describes its emulation with PATR. Section 3 discusses how the lexical orientation of DG motivates a proper distinction between complements subcategorized for by the head and adjuncts that are not, and describes how this can be accomplished in a constrained CFG emulation using subcategorization lists in the lexicon. A distinction between grammatical information residing and not residing in the lexicon is noted. Section 4 discusses the real nature of the constrained CFG emulation of DG. Though DG can be usefully emulated by a constrained CFG model, the formalism, as least as defined by Robinson's axioms, does have aspects that cannot be modelled elegantly by PSG.

1 Representation of Dependency

The governor-dependent (head-modifier) relationship between words in an utterance can be represented as for the Chinese sentence (from Yuan and Huang (1992) in Fig. 2:

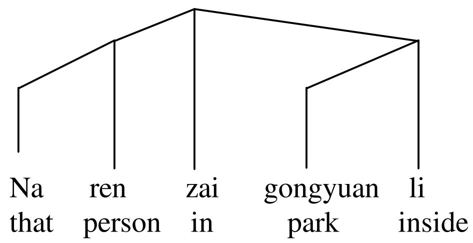

Fig. 2

The *main* (or *central*) element, using Hays' (1964) terminology, of the sentence is *zai*. Its immediate dependants are *ren* and *li*, which, in turn, have dependants of their own. This sentence can also be represented as in Fig. 3 (Tesniere's *stemma*):

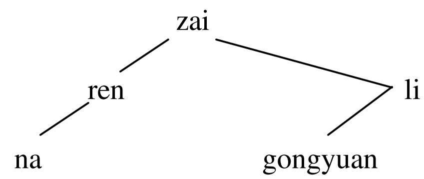

Fig. 3

If we do not mangle up word order in the *dependency structure* of Fig. 3, it can be seen that it is equivalent to the *tree structure* in Fig. 2.

Based on the work of Gaifman (1965), Hays (1964) proposes rules of the following form for the generation of dependency structures:

(a) X(A,B,C, ... ,H, * ,Y, ... ,Z)
(b) X(*)
(c) * (X)

Fig. 4

In Fig.4, (a) states that the governing *auxiliary alphabet* X has dependent A, B, C, ..., H, Y, ..., Z and that X itself (the governor) is situated between H and Y. (b) says that the *terminal alphabet* X occurs without any dependants. (c) says that X occurs without any governor, i.e. it is the *main* or *central* element. Gaifman (1965) establishes that a Dependency Grammar obtained in this way is equivalent to a phrase structure grammar in the sense that:

- they have the same terminal alphabet;
- for every string over that alphabet, every structure attributed by either grammar corresponds to a structure attributed by the other.

Robinson's (1970) four axioms (*v. supra*) license the same kinds of dependency structures as Hays′ rules. Unlike these rules, they do not contain unnecessary stipulation involving linear word order. It is easy to see that the third axiom is a requirement of *single-headedness*. As for the fourth axiom, consider Fig. 5:

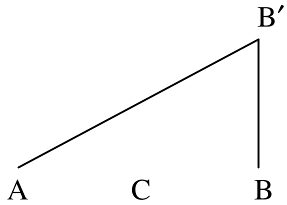

Fig. 5

The axiom stipulates that the governor of C must be located between A and B (which are themselves possible governors). Seen in the Baysian cast, this effectively requires that the link between C and its governor should not cross the lines AB′ and BB′. This is thus a requirement of *projectivity*.

2 Dependency Rules and Grammatical Function Annotation

Generalizing over approaches adopted in DG-based parsing of Chinese, Lai and Huang (1994) noted that grammatical functions like *subject* and *object* are generally used to label dependency links. These labels are not found in Hays' dependency rules and Robinson's axioms. In a sense, they are entities on a second level. Borrowing the idea of *functional annotation* from LFG, they proposed annotated dependency rules of the following form:

X(A(fa), B(fb), ..., * , ... Z(fz))

For example, the following rules account for the transitive verbs:

(a) * (TV)
(b) TV(N(subj), * , N(obj))
(c) N(*)

Fig. 6

While this is not a phrase-structure grammar (PSG), Lai and Huang (in press) exploited the obvious affinity between Robinson-style DG and PSG and implemented a DG parser using the PATR of Gazdar and Mellish (1989). In this implementation, the Chinese sentence

Zhang San	kanjian	Li Si
name	saw	name

is accounted for by the annotated rules (simplified and semantic analysis mechanisms stripped for brevity) in Fig. 7:

```
R ule   TVP ---> [PN1, TV, PN2] :-
        TVP:cat    ===   tv,
        TV:cat     ===   tv,
        PN1:cat    ===   n,
        PN2:cat    ===   n,
        TVP:ds       ===   TV:ds,
        TVP:ds:subj   ===   PN1:ds,
        TVP:ds:obj    ===   PN2:ds.
W ord   kanjian :-
        W:cat     ===   tv,
        W:ds:head    ===   kanjian.
W ord   'zhangsan' :-
        W:cat     ===   n,
        W:ds:head    ===   'zhangsan'.
W ord   'lisi' :-
        W:cat     ===   n,
        W:ds:head    === 'lisi'.
```

Fig. 7

Besides outputting a grammatical function structure, the following dependency structure is produced:

[tv, [[n, [zhangsan]], [tv, [kanjian]], [n, [lisi]]]]

PATR outputs a phrase-structure tree, but after applying a pruning operation on the two *tv*'s, a structure equivalent to Fig. 8 is obtained:

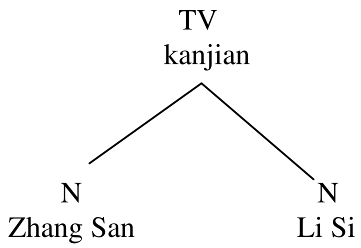

Fig. 8

The more complicated sentence, involving a subject-control verb,

Zhang San	xiang	da	Li Si
name	want	hit	name

is accounted for the following rules (necessary details only, for brevity):

```
R ule CVP ---> [N, CV, VS] :-
    CVP:slash:here   ===   no,
    CVP:slash:down   ===   yes,
    CVP:cat   ===   CV:cat,
    CV:cat    ===   v,
    CV:subcat:tran   ===   tv,
    CV:subcat:ctrl   ===   subj,
    N:cat   ===   n,
    VS:cat    ===   v,
    VS:slash:here    ===   subj,
    CVP:ds    ===   CV:ds,
    CV:ds:subj:fill    ===   N:ds,
    CV:ds:obj:fill     ===   VS:ds,
    %Subject-control information follows
    CV:ds:subj:fill=== CV:ds:obj:fill:subj:fill.
R ule VS ---> [V, N] :-
    VS:slash:here    ===   subj,
    VS:cat   ===   V:cat,
    V:cat    ===   v,
    V:subcat:tran    ===   tv,
    N:cat    ===   n,
    VS:ds    ===   V:ds,
    V:ds:obj:filler ===   N:ds.
W ord   xiang3 :-
    W:cat     ===   v,
    W:subcat:tran    ===   tv,
    W:subcat:ctrl    ===   subj,
    W:ds:head    ===   xiang.
```

Control sentences are one of the motivations for DG grammarians like Hudson (1994) to give up the *single-headedness* and *non-projectivity* requirements, finding it difficult, for example, not to allow the controlled verb *da* to have both *Zhang San* and *xiang* as its heads, violating *single-headedness* and *projectivity* at the same time as shown in Fig. 9:

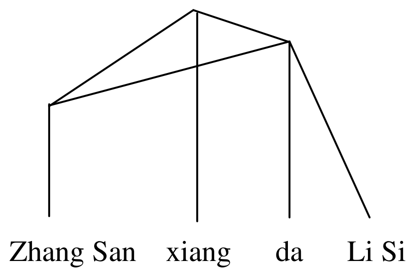

Fig. 9

By introducing a level of grammatical function to accommodate such complications, Lai and Huang (1995; in press) preserve *single-headedness* and *projectivity* in the syntactic dependency structure as in Fig. 10:

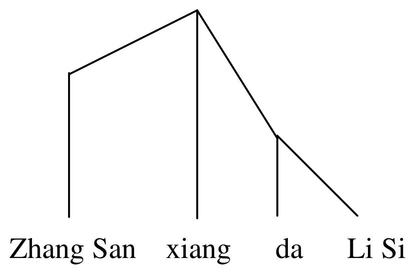

Fig. 10

Other difficulties involving raising, extraction, tough-movement and extraposition (Hudson (1994)) can be dealt with similarly.

This two-level approach to DG parsing is essentially a context-free PSG constrained by grammatical function annotations. A grammatical function structure accompanies the dependency structure of a legal sentence just as a functional structure is associated with a constituent structure in LFG. Morphological and semantic constraints (Melcuk (1988)) can also be dealt with on additional levels.

3 Complements and Adjuncts

The constrained CFG emulation of DG described in the previous section inevitably prompts the question whether it is still a DG. In his foreword in Starosta (1988), Hudson mentioned three characteristics of DG. First, DG should be *monostratal* in the sense that there should be no transformations. Second, dependency should be basic, not derived. Third, the rules of grammar should not be formally distinct from subcategorization facts. Lai and Huang′s approach meets the first two criteria. While the proper treatment of adjuncts will be discussed below, the close coupling of the phrase-structure-rule-like dependency rules and subcategorization properties discussed in the previous section also gives the approach the third characteristic.

One may feel somewhat uncomfortable about phrase-structure rules or phrase-structure-rule like mechanisms playing an important role in an emulation of DG. After all, although it is true that Hays′ rules work like phrase structure rules, conformation to Robinson′s axioms does not imply that the process of sentence recognition will necessarily have an image in a PSG. The situation is particularly critical in the treatment of adjuncts that are not subcategorized for by a head word. We could quite easily deal with adjunct in the manner of the following annotated phrase structure rule in LFG:

$$\begin{array}{llll} VP \rightarrow & V & NP & ZP \\ & & (\uparrow obj) = \downarrow & (\uparrow adjunct) = \downarrow \end{array}$$

But we would then have to let a large number of phrase structure rules not related to subcategorization facts slip into the grammar. This violates the third criterion mentioned above and is obviously undesirable.

This being a critical problem of constrained CFG emulation of DG, we adopt another approach by exploiting the fact that the categorical labels of a head word and it′s dominating nodes are the same. Using (simplified) PATR notations, the two generic rules:

```
X → X Y
  Y: fun = adjunct

X → Y X
  Y:fun = adjunct
```

Fig. 11

will be able to cover all kinds of adjunct rules. The two X's on the two sides of the arrow are short-hand for two different symbols, say X1 and X2, constrained by the condition X1:cat = X2:cat.

As subcategorization information has to be encoded in the lexical items anyway, there is nothing seriously wrong with phrase-structure-rule-like stipulations about complements. However, we should note that, in Chinese and English, adjuncts generally do not come between a head word and its "unmoved" non-subject complements. This could be taken care of by adding a bar-level feature to the rules in Fig. 11 as in Generalized Phrase Structure Grammar (GPSG, Gazdar et al. (1985)). But then we would be relying more heavily on phrase-structure-rule-like mechanisms. Instead of this, we find the alternative method of treatment in Head-Driven Phrase Structure Grammar (HPSG, Pollard and Sag (1994)) convenient.

First, lexical entries are as in Fig. 12:

```
gei   ('give')
   cat = v
   subcat.left = [n(subj)]
   subcat.right = [n(iobj), n(obj)]
```

Fig. 12

Rules like the following (necessary details only, for brevity) will take care of complements subcategorized for by the head word:

```
V  →  V  X
  {cat(fun) = pop(V:subcat.right)}
  % fails if V:subcat = []
  X:cat = cat
  X:fun = fun

V  →  X  V
```

```
{cat(fun) = pop(V:subcat.left)}
% fails if V:subcat = []
X:cat = cat
X:fun = fun
```

Fig. 13

Adjuncts are kept from getting in between a head word and its unmoved non-subject complements by adding constraints like

X: subcat.right = []

to rules in Fig. 11

As shown in Fig. 12, a lexical entry has two subcategorization lists, one for complements on its left and one for complements on its right, an inspiration from Yuan and Huang (1992). The elements in a subcategorization list is arranged so that the one that occurs closest to the head word is at the head. The rules in Fig. 13 are presented in a form that is easily understood by readers. The *pop* operation, which is procedural in nature (hence the braces), hands over to the caller the head elements of a the subcategorization list, removing it from the list at the same time. It is actually implemented in a PATR-compatible manner.

This scheme works for Chinese and English, in which unmoved non-subject complements follow the verb. Adjustments are required for moved complements. Adjustments are also required for other languages.

It should be noted that the rules, in the spirit of Robinson′s axioms, try not to meddle with word order as far as possible. In this respect, PATR is inelegant in that it has to have two symmetrical adjunct rules in Fig. 11 and two symmetrical complement rules in Fig. 13. This inelegance seems to be inherent in the PSG nature of PATR.

4 Nature of the Emulation Model

An examination of the real nature of our emulation model is in order. As a computational emulation of a DG conforming to Robinson's four axioms and using grammatical functions to label dependency links, it sanctions sentences with dependency structures that satisfy the *single-headed* and *non-projective* conditions. Well-formed dependency structures are accompanied by a grammatical function structures that, *inter alia*, ensure that subcategorization properties of lexical items are satisfied. Grammatical function structures do not have to conform to the *single-headed* and *projective* conditions. Morphological and semantic constraints can be accommodated similarly.

Most grammatical mechanism in the emulation are triggered by lexical information. Hays-style dependency rules, which are emulated by phrase-structure mechanisms in PATR. Rules for complements of the head word derive their real power from lexical subcategorization information. They thus meet the criterion that rules of grammar of a DG should not be formally distinct from subcategorization facts.

Adjunct rules are not related to any subcategorization facts. We believe that their existence (in small numbers) is justified in our emulation model. Even in a DG formalism that does not have phrase-structure-like rules, there have to be some general facilities to take care of such non-lexical grammatical mechanisms. DG is lexically orientated, but it has to cope with non-lexical grammatical mechanisms, where they exist, in language.

The dependency rule cum functional constraint emulation in Lai and Huang (1994; 1995; in press) has obviously been influenced by LFG. With the introduction of mechanisms to handle complements and adjuncts in the previous section, the emulation model has moved towards HPSG. Grammatical function constraints, which work like functional annotations in LFG, provide the main facilities to resolve grammatical problems like control. On the other hand, dependency rules, emulated by PATR phrase-structure rules, are kept to a minimum and deals with subcategorization and adjoining with a HPSG-like mechanism.

The emulation model, however, remains an emulation. The Chinese parsing experiments from which the generalization has been made do not all use (context-free) phrase structure rules (e.g. Yuan et al. (1992); Zhou and Huang (1994)). PATR rules are useful only in so far as they can produce structures that can be transformed to dependency structures.

Conclusion

We have thus been committed in our efforts to emulate a Robinson-style Dependency Grammar with a lexically oriented Context-Free Grammar constrained by grammatical function annotations. Besides providing a formalism for valid and illuminating linguistic analysis, this emulation has enabled us to implement a unification-based parser in PATR. We are however not necessarily committed to the claim that Dependency Grammar is a notational variant of Phrase Structure Grammar. Robinson-style dependency structures have great affinity with phrase structure trees, but they do not have to be generated by phrase structure rules. In fact, phrase structure rule-based emulation is inelegant in handling some phenomena that DG can deal with elegantly.

Acknowledgements

Our thanks go to the National Science Foundation of China for supporting the research reported in this paper.

References

Bresnan J.W. ed. (1982) *The Mental Representation of Grammatical Relations*. MIT Press, Cambridge, U.S.A., 874p.

Covington M.A. (1990) *Parsing Discontinuous Constituents in Dependency Grammar*. Computational Linguistics, 16/4, pp. 234-236.

Gaifman H. (1965) *Dependency Systems and Phrase-Structure Systems*. Information and Control, 8, pp. 304-337.

Gazdar G., Klein E., Pullum E. and Sag I. (1985) *Generalized Phrase Structure Grammar*. Blackwell, Oxford, 276p.

Gazdar G. and Mellish C. (1989) *Natural Language Processing in Prolog*. Addision Wesley, Wokingham, 504p.

Hays D.G. (1964) *Dependency Theory: A Formalism and Some Observations*. Language, 40, pp. 511-525.

Hellwig P. (1986) *Dependency Unification Grammar*. Proc. COLING 86, pp. 195-199.

Huang C.N., Yuan C.F. and Pan S.M. (1992) *Yuliaoku, Zhishi Huoqu He Jufa Fenxi (Corpora, Knowledge Acquisition and Syntactic Parsing)*. Journal of Chinese Information Processing, 6/3, pp. 1-6.

Hudson R. (1984) *Word Grammar*. Blackwell, Oxford, 267p.

Hudson R. (1990) *English Word Grammar*. Blackwell, Oxford, 445p.

Hudson R. (1994) Discontinuous Phrases in Dependency Grammar. University of London Working Papers in Linguistics, 6, pp. 89-124.

Hudson R. (1995) *Dependency Counts*. In "Functional Description of Language", E. Hajicova, ed., Faculty of Mathematics and Physics, Charles University, Prague, pp. 85-115.

Lai B.Y. and Huang C.N. (1994) *Dependency Grammar and the Parsing of Chinese Sentences*. The Proceedings of the 1994 Kyoto Conference (Joint ACLIC8 and PACFoCoL2), 10-11 Aug. 1994, pp. 63-71.

Lai B.Y.T. and Huang C.N. (1995) *Single-Headedness and Projectivity for Syntactic Dependency*. The Linguistics Association of Great Britain Spring Conference, University of New Castle , 10-12 August, 1995.

Lai T.B.Y. and Huang C.N. (in press) *An Approach to Dependency Grammar for Chinese*. In "Theoretical Explorations in Chinese Linguistics", Y. Gu, ed., Hong Kong: Linguistic Society of Hong Kong, Hong Kong, in press.

Li J.K., Zhou M. and Huang C.N. (1993) *Tongji Yu Guize Jiehe De Hanyu Jufa Fenxi Yanjiu (Study on Using Statistics and Rules at the Same Time in Syntactic Analysis)*. Proc. JSCL93, Xiamen University, pp. 176-181.

Maxwell D. (ms) *Unification Dependency Grammar*.

Melcuk I.A. (1988) *Dependency Syntax: Theory and Practice*. State University of New York Press, Albany, 428p.

Pollard C. and Sag I. (1994) *Head-Driven Phrase Structure Grammar*. University of Chicago Press, Chicago, 440p.

Robinson J.J. (1970) *Dependency Structures and Transformation Rules*. Language, 46, pp. 259-285.

Shieber S.M. (1986) *An Introduction to Unification-Based Approach to Grammar*. Chicago University Press, Chicago, 105p.

Starosta S. (1988) *The Case for Lexicase*. Pinter, London.

Tesniere L. (1959) *Elements de Syntaxe Structurale*, Klincksieck, Paris.

Yuan C.F. and Huang C.N. (1992) *Knowledge Acquisition and Chinese Parsing Based on Corpus*. Proc. COLING 92, Nantes, France, pp. 13000-13004.

Zhou M. and Huang C.N. (1994) *An Efficient Syntactic Tagging Tool for Corpora*. Proc. COLING 94, Kyoto, pp. 949-955.