IPASJ International Journal of Information Technology (IIJIT)
Web Site: http://www.ipasj.org/IIJIT/IIJIT.htm
A Publisher for Research Motivation ....
Email:editoriijit@ipasj.org
Volume 5, Issue 11, November 2017
ISSN 2321-5976

Multiple Adjunction in Feature-Based Tree-Adjoining Grammar

Alcott, Louisa May

University of Mannheim

ABSTRACT

In parsing with Tree Adjoining Grammar (TAG), independent derivations have been shown by Schabes and Shieber (1994) to be essential for correctly supporting syntactic analysis, semantic interpretation, and statistical language modeling. However, the parsing algorithm they propose is not directly applicable to Feature-Based TAGs (FB-TAG). We provide a recognition algorithm for FB-TAG that supports both dependent and independent derivations. The resulting algorithm combines the benefits of independent derivations with those of Feature-Based grammars. In particular, we show that it accounts for a range of interactions between dependent vs. independent derivation on the one hand, and syntactic constraints, linear ordering, and scopal vs. nonscopal semantic dependencies on the other hand.

1. INTRODUCTION

A Tree Adjoining Grammar (TAG; Joshi and Schabes 1997) consists of a set of elementary trees and two combining operations, substitution and adjunction. Consequently, a TAG derivation can be described by a tree (called a derivation tree) specifying which elementary TAG trees were combined using which operations to yield that derivation. In this tree, each vertex is labeled with a tree name and each edge with a description of the operation (node address and operation type) used to combine the trees labeling its end vertices. As we shall see in Section 3.2, in TAG, each derivation tree specifies a unique parse tree, also called derived tree. In previous work, it has been argued that TAG derivation trees provide a good approximation of semantic dependencies between the words of a sentence (Kroch 1989; Rambow, Vijay-Shanker, and Weir 1995; Candito and Kahane 1998; Kallmeyer and Kuhlmann 2012). As shown by Schabes and Shieber (1994), however, there are several possible ways of defining TAG derivation trees, depending on how multiple adjunction of several auxiliary trees at the same tree node is handled. The standard notion of derivation proposed by Vijay-Shanker (1987) forbids multiple adjunction, thus enforcing dependent derivations. In contrast, the extended notion of derivation proposed by Schabes.

Shieber (1992, 1994) allows multiple adjunction at a single node, thereby yielding so-called independent derivations (i.e., derivations where the relation between the adjoining trees is left unspecified). The difference between the two types of derivations is illustrated in Figure 1. While in the standard (dependent) derivation, one adjective tree is adjoined to the other adjective tree, which itself is adjoined to the noun tree for pepper; in the extended (independent) derivation, both adjective trees adjoin to the noun tree. Schabes and Shieber (1994) argue that allowing both for dependent and independent derivations better reflects linguistic dependencies. Making use of the distinction introduced in TAG between predicative and modifier auxiliary trees (Schabes and Shieber (1994), Section 3.1), they define a parsing algorithm that assigns dependent derivations to predicative auxiliary trees but independent derivations to multiple modifier auxiliary trees adjoining to the same node. In case both predicative and modifier auxiliary trees adjoin to the same node, their parsing algorithm ensures that predicative trees appear above the modifier trees in the derived tree. This parsing algorithm is defined for featureless variants of TAG. In contrast, in implemented TAGs (e.g., XTAG [The XTAG Research Group 2001], SemXTAG [Gardent 2008], or XXTAG1 [Alahverdzhieva 2008]) feature structures and feature unification are central. They are used to minimize the size of the grammar; to model linguistic phenomena such as verb/subject agreement; and to encode a unification-based syntax/semantics interface (e.g., Gardent and Kallmeyer 2003). In this article, we extend Schabes and Shieber's proposal to Feature-Based TAG (FB-TAG); and we show that the resulting parsing algorithm naturally accounts for the interplay of dependent vs. independent derivation structures with syntactic constraints, linear ordering, and scopal vs. nonscopal semantic dependencies. The article is organized as follows. In Section 2, we recap the motivations for independent derivations put forward by Schabes and Shieber (1994) and we briefly discuss the interactions that may arise between dependent and independent derivations. Section 3 summarizes their approach. In Section 4, we present the intuitions and motivations underlying our proposal and we highlight the differences with Schabes and Shieber's approach. Section 5 presents our proposal. Section 6 concludes

2. WHY ARE INDEPENDENT DERIVATIONS DESIRABLE?

We start by summarizing Schabes and Shieber's motivations for independent derivations. We then discuss the interactions between dependent and independent derivations.

Movement verbs such as to wander allow for directional modifiers such as through the meerkats, whereas verbs such as to remind do not. In TAG, such restrictions can be modeled using selective adjoining constraints to specify which modifier tree may or may not be adjoined at a particular node in a given tree. Therefore, it is possible to license (1) and to rule out (2c). In Example (2a), however, under the dependent notion of adjunction, the tree for the directional adverbial through the meerkats will adjoin to the modifier tree for yesterday, which itself will adjoin to the tree selected by reminded. Thus, constraints placed by the verb on its modifiers must be passed through by modifier trees (here, the tree for yesterday) to also rule out sentences such as Example (2a). Propagating selective adjunction constraints in TAG would lead to a formalism for which derivation trees are no longer context-free (Schabes and Shieber 1994). The second motivation for independent adjunction stems from probabilistic approaches. Stochastic lexicalized TAG specifies the probability of an adjunction of a given auxiliary tree at a given node in another elementary tree (Resnik 1992; Schabes 1992). Thus, under the standard notion of derivation, the overall probability of the string roasted red pepper would be determined by the probability of red adjoining to pepper and the probability of roasted adjoining to red. In contrast, independent adjunction would result in a derivation such that the overall probability of the string roasted red pepper would be determined by the probability of both red and roasted adjoining to pepper. Schabes and Shieber (1994, page 97) argue that it is plausible that "the most important relationships to characterize statistically are those between modifier and modified, rather than between two modifiers." A third motivation comes from semantics and, more particularly, from scope ambiguities involving modifiers. Given a sentence such as Example (3), where the relative scope of the modifiers twice and intentionally is ambiguous,3 Shieber (1994) shows that, under the extended definition of adjunction, a synchronous TAG modeling the relation between syntactic trees and logical formulae can account for both readings.

2.1 Dependent, Independent, and Mixed Derivations

To capture the different types of semantic dependencies and morpho-syntactic constraints that may hold between multiple auxiliary trees adjoining to the same entity, both dependent and independent derivations are needed. As argued earlier, because there are no constraints or semantic relation holding between each of them, multiple intersective modifiers applying to the same entity (Example (4)) are best modeled using an independent derivation.

In contrast, because they may involve strong scopal and morpho-syntactic constraints, stacked predicative verbs (i.e., verbs taking a sentential complement, Example (5a)) and non-intersective modifiers (Example (5c)) require dependent derivations. Consider sentences (5a–b), for instance. If predicative trees were assigned an independent derivation, sentence (5a) would be judged ungrammatical (because want requires an infinitival complement but would adjoin to the finite verb slept) and conversely, sentence (5b) would incorrectly be judged grammatical (because both want and try require an infinitival complement). Similarly, in Example (5c), the church is Syrian Orthodox, not Syrian and Orthodox. Assigning a dependent rather than an independent derivation to such cases straightforwardly captures the distinction between intersective and non-intersective modifiers.

As we shall see in Section 5.3, the parsing algorithm we propose licenses dependent, independent, and mixed derivations but is restricted to appropriately distinguish between various types of modifiers. Moreover, the feature information encoded in the grammar further restricts the derivation structures produced, thereby accounting for the interactions between adjunction, linear ordering, and morpho-syntactic constraints.

3. SCHABES AND SHIEBER'S PROPOSAL: MOTIVATIONS AND INTUITIONS

Tree Adjoining Grammar distinguishes between two types of auxiliary trees, namely, modifier vs. predicative auxiliary trees (Joshi and Vijay-Shanker 2001). Whereas predicative trees are assigned to verbs taking a sentential argument, modifier trees are assigned to all other auxiliary trees (e.g., verbal auxiliaries, adjectives, adverbs, prepositions, and determiners). More generally, the difference between a predicative and a modifier tree is that in a predicative tree, the foot node, like the substitution nodes, corresponds to an argument node selected by its lexical anchor (i.e., the word that selects that tree) whereas in a modifier auxiliary tree, the foot node is an open slot corresponding to the phrase being modified. When associating semantic entities with tree nodes (as proposed, for example, by Joshi and Vijay-Shanker [2001] and Gardent and Kallmeyer [2003]), this difference can be seen by noting the entities associated with root and foot nodes: These are distinct in a predicative tree but identical in modifier trees. In their approach, Schabes and Shieber specify a TAG-to-LIG conversion that systematically associates dependent derivations with predicative

IPASJ International Journal of Information Technology (IIJIT)
Web Site: http://www.ipasj.org/IIJIT/IIJIT.htm
A Publisher for Research Motivation .... Email:editoriijit@ipasj.org
Volume 5, Issue 11, November 2017 ISSN 2321-5976

auxiliary trees and independent derivations with modifier auxiliary trees. In addition, they introduce two mechanisms to ensure that each derivation tree unambiguously specifies a linguistically plausible derived tree.

First, they enforce ordering constraints between modifier trees adjoining at the same node (which are thus ambiguous with respect to the derived tree they describe) by assuming that derivation trees are ordered and that linear precedence (LP) statements can be used to constrain the order of siblings in a derivation tree. For instance, given the independent derivation shown in Figure 1, an LP statement stating that βred must occur before βroasted in the derivation tree will ensure that βroasted appears above βred in the derived tree and therefore that the resulting derived tree yields the phrase roasted red pepper rather than red roasted pepper. Second, when both predicative and modifier trees adjoin at the same address, predicative trees always occur above all modifier trees in the derived tree ("outermost predication"). When allowing for independent derivations, however, several derived trees are possible, depending on the order in which the auxiliary trees are adjoined. To ensure a unique mapping from derivation to derived tree, Schabes and Shieber (1994) therefore introduce the notion of ordered derivation trees. Ordered derivation trees differ from standard TAG derivation trees in that (i) they may contain sibling edges labeled with the same address, and (ii) they specify a total order on such siblings.

4 TAG TO LIG COMPILATION

The TAG-to-LIG compilation proposed by Vijay-Shanker and Weir (1991) produces LIG rules that simulate a traversal of the derived tree produced by the original TAG grammar. In these LIG rules, each node η of a TAG elementary tree is viewed as having both a top t[..η] and a bottom b[..η] component to account for the possibility of an adjunction. Figure 3 illustrates the traversal of the TAG-derived trees specified by the LIG resulting fromVijay-Shanker and Weir (1991) TAG-to-LIG compilation. Figure 4 lists the LIG rules resulting from the TAG to LIG compilation process. Each nonterminal (t[..η] or b[..η]) with the top of the stack symbol in a LIG rule corresponds to a unique node in some elementary tree of the grammar. The inner stack symbols are used to keep track of the nodes higher in the derived tree where an auxiliary tree has been adjoined. Rules of Types 1 and 2 capture immediate dominance between the bottom of a node η and the top of its immediate daughters in two configurations, depending on whether η dominates the foot node (Type 1) or not (Type 2). Rules of Type 3 handle nodes that require neither substitution nor adjunction. This rule handles cases where no adjunction occurs at a node by rewriting the top of this node to its bottom. Rules of Type 6 model substitution. Finally, rules of Types 4 and 5 handle adjunction. They specify that, for any given node η and any auxiliary tree β that may adjoin to η, the top of η rewrites to the top of the root node of β; and the bottom of the foot of β to the bottom of η. It follows that there can be no multiple adjunction in this LIG version of TAG.

5 MODIFYING THE TAG TO LIG COMPILATION TO ALLOW FOR MULTIPLE ADJUNCTIONS

To associate predicative tree adjunctions with dependent derivations and multiple modifier adjunctions with independent derivations, Schabes and Shieber (1994) modify the compilation of TAG to LIG, proposed by Vijay-Shanker and Weir (1991) as sketched in Figure 5. Type 4(a) rules apply to adjunctions involving predicative trees. They are identical to Type 4 rules in the Vijay-Shanker and Weir's approach and therefore enforce a standard (dependent) derivation for predicative trees. In contrast, Type 4(b) rules apply to adjunctions involving modifiers and result in an independent derivation.

As shown in Figure 10, this approach can incorrectly lead to derivation failures in the case of an independent multiple adjunction. Intuitively, the reason for this is that, in Schabes and Shieber's approach, multiple adjunction starts and ends from the bottom component of the node being adjoined to. This is fine when no features are involved because the category of the node being adjoined to is always identical to the root and foot node of the auxiliary trees being adjoined. When nodes carry feature structures, however, a unification clash can occur that makes derivation fail. Thus, in our example, derivation incorrectly fails because the bottom feature structures of the root node of the auxiliary tree for all and the bottom feature structure of the root node of the auxiliary tree for the should unify but have conflicting value. As shown by the dependent derivation for all the meerkats depicted in Figure 8, this is incorrect.

Proposal:

Intuition and Motivations As we just saw, in the case of multiple independent adjunctions, a straightforward extension of Schabes and Shieber's LIG framework to FB-TAG fails to correctly capture the unification constraints encoded in the grammar. More generally, when extending multiple independent adjunction to FB-TAG, it is crucial that the feature constraints encoded by the linguist describe the same set of derived trees no matter One key point illustrated by Figure 11 is that whereas multiple adjunction operates on a single node (here ηo), the unification constraints of FB-

TAG require that the bottom feature structure of the foot of an auxiliary tree which appears higher in the derived tree (here, β2) unifies with the bottom feature structure of the root of the auxiliary tree appearing immediately below it in the derived tree (here β1)—not with that of the root of the node to which it adjoins (here ηo). In other words, while a multiple adjunction on ηo operates on ηo only, a correct implementation of FB-TAG unification constraints requires keeping track of the feature structures associated with the auxiliary trees successively adjoining to ηo . In our proposal, we capture this bookkeeping requirement by associating tree nodes not with feature structures but with reference variables pointing to feature structures. The parsing algorithm is then specified so as to support dependent, independent, and mixed derivations while enforcing the same unifications as would be performed under a dependent adjunction.

To specify these constraints on similar cases (soft ordering constraints on adjectives and strict ordering constraints on temporal and spatial adverbial phrases in German), Schabes and Shieber (1994) suggest the use of LP constraints on derivation tree siblings. As illustrated by the derivation of Example (11c–d) in Figures 8 and 9, in the FBTAG approach, such additional constraints are unnecessary: They simply fall out of the feature constraints encoded in the grammar

6. CONCLUSION

Although independent derivations have been shown by Schabes and Shieber (1994) to be essential for correctly supporting syntactic analysis, semantic interpretation, and statistical language modeling, the parsing algorithm they propose is restricted to TAG and is therefore not directly applicable to large scale implemented Feature-Based TAGs. We have provided a recognition algorithm for FB-TAGs that supports both dependent and independent derivations under certain restrictions enforced jointly by feature constraints and by side conditions in the parsing algorithm. The resulting algorithm combines the benefits of independent derivations with those of Feature-Based Grammars. In particular, we showed that it accounts for a range of interactions between dependent vs. independent derivation on the one hand, and syntactic constraints, linear ordering, and scopal vs. nonscopal semantic dependencies on the other hand.

REFERENCES

[1]. Alahverdzhieva, Katya. 2008. XTAG using XMG. Master's thesis, Universit´e de Nancy.
[2]. Candito, Marie-H´el`ene and Sylvain Kahane. 1998. Can the TAG derivation tree represent a semantic graph? An answer in the light of Meaning-Text Theory. In Proceedings of the Fourth Workshop on Tree-Adjoining Grammars and Related Frameworks (TAG+4), pages 21–24, Philadelphia, PA.
[3]. Crabb´e, Benoˆıt, Denys Duchier, Claire Gardent, Joseph Le Roux, and Yannick Parmentier. 2013. XMG: eXtensible MetaGrammar. Computational Linguistics, 39(3):1–66.
[4]. Gardent, Claire. 2008. Integrating a unification-based semantics in a large scale Lexicalised Tree Adjoining Grammar for French. In Proceedings of the 22nd International Conference on Computational Linguistics (COLING), pages 249–256, Manchester.
[5]. Gardent, Claire and Laura Kallmeyer. 2003. Semantic construction in Feature-Based TAG. In Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL), pages 123–130, Budapest.
[6]. Gazdar, Gerald. 1988. Applicability of indexed grammars to natural languages. In Uwe Reyle and Christian Rohrer, editors, Natural Language Parsing and Linguistic Theories, volume 35 of Studies in Linguistics and Philosophy. Springer, Netherlands, pages 69–94.
[7]. Joshi, Aravind K. 2000. Relationship between strong and weak generative power of formal systems. In Proceedings of the 5th International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+5), pages 107–114, Paris