Flexible Composition in LTAG: Quantifier Scope and Inverse Linking

Aravind K. Joshi
IRCS, University of Pennsylvania,
3401 Walnut Street, Philadelphia, PA 19104–6228, USA.
joshi@linc.cis.upenn.edu

Laura Kallmeyer
TALaNa/LaTTICe, University Paris 7,
2 place Jussieu, 75251 Paris Cedex 05, France.
laura.kallmeyer@linguist.jussieu.fr

Maribel Romero
Department of Linguistics, 610, Williams Hall, University of Pennsylvania,
Philadelphia, PA, 19104-6305, USA.
romero@ling.upenn.edu

Abstract

This paper addresses the problem of constraints for relative quantifier scope, in particular in inverse linking readings where certain scope orders are excluded. We show how to account for such restrictions in the Tree Adjoining Grammar (TAG) framework by adopting a notion of flexible composition. In the semantics we use for TAG we introduce quantifier sets that group quantifiers that are 'glued' together in the sense that no other quantifier can scopally intervene between them. The flexible composition approach allows us to obtain the desired quantifier sets and thereby the desired constraints for quantifier scope.

1 Introduction

The scope of quantifiers within a sentence can be in principle arranged in different orderings, making the sentence potentially ambiguous. For example, sentence (1a), with two quantifiers, has two logically possible scope orderings: the surface order *some* > *every*, and an "inverse" order *every* > *some*. These two orderings yield two actual readings of the sentence, spelled out in Predicate Logic in (1b-c) respectively.

(1) a. An FBI agent is spying on every professor.

b. $\exists\,\forall$: $\exists x\ [\mathsf{agent}(x)\ \wedge\ \forall y[\mathsf{professor}(y)\ \rightarrow\ \mathsf{spy}(x,y)]\ ]$

c. $\forall\,\exists$: $\forall y\ [\mathsf{professor}(y)\ \rightarrow\ \exists x[\mathsf{agent}(x)\ \wedge\ \mathsf{spy}(x,y)]\ ]$

Both scope orderings must be made available by the grammar also when the two quantifiers appear nested within each other, that is, when one of the quantifiers appears within the Noun Phrase headed by the other quantifier. This is illustrated in (2) and (3). In (2a), the surface ordering *no>a* corresponds to the existing reading spelled out in (2b). In (3a), the inverse ordering *every>a* gives us the existing reading in (3b). The inverse reading in nested quantifier constructions is called "inverse linking reading". Note, that, in the inverse linking reading, the nested Qu$_2$ does not only take scope over its host NP, but over the clause in general, as it can bind the variable *it* in (4) (May 1985, p. 68).

(2) a. No representative from an African country came to the meeting.

b. $\neg\exists\,\exists$: $\neg\exists x\ \exists y\ [\mathsf{representative}(x,y)\ \wedge\ \mathsf{Afrcountry}(y)\ \wedge\ \mathsf{came}(x)\ ]$

(3) a. A representative from every African country came to the meeting.

b. $\forall\,\exists$: $\forall y\ [\ \mathsf{Afrcountry}(y)\ \rightarrow \exists x\ [\mathsf{representative}(x,y)\ \wedge\ \mathsf{came}(x)]\ ]$

(4) Somebody from every city despises it.

When we turn to sentences with three quantifiers Qu$_1$, Qu$_2$ and Qu$_3$, we have six logically possible scope combinations. One of these six combinations, namely the ordering Qu$_3$ Qu$_1$ Qu$_2$, yields an actual reading for sentences like (5), as spelled out in (5b).[1]

(5) a. (At least) two social workers gave a doll to each/every child.

b. $\forall\,2\,\exists$: $\forall y\ [\ \mathsf{child}(y)\ \rightarrow\ \exists x\ [\mathsf{social\text{-}workers}(x)\ \wedge\ |x| \geq 2\ \wedge$
$\forall x'[x' \subset_i x\ \rightarrow\ \exists z[\mathsf{doll}(z)\ \wedge\ \mathsf{give}(x', z, y)]\ ]\ ]\ ]$

However, in nested quantifier configurations where Qu$_3$ appears within Qu$_2$, this same ordering Qu$_3$ Qu$_1$ Qu$_2$ does not yield an actual reading (Larson 1987, Heim & Kratzer 1998, Sauerland 2000). Neither does Qu$_2$ Qu$_1$ Qu$_3$ yield an actual reading (Hobbs & Shieber 1987). This can be seen in (6)-(7). Take, e.g., the six logically possible scope orderings in (6a). The claim is that the nested quantifiers $\exists$ and $\forall$ can in principle take scope together under 2 (orders $2\exists\forall$ and $2\,\forall\,\exists$) or they can take scope together over 2 ($\exists\,\forall\,2$ and $\forall\,\exists\,2$), but they cannot take scope separately with the quantifier 2 intervening between them (orderings * $\exists\,2\,\forall$, * $\forall\,2\,\exists$).

(6) a. Two politicians spy on someone from every city. Larson (1987)

b. $2\,\exists\,\forall$, $2\,\forall\,\exists$, $\exists\,\forall\,2$, $\forall\,\exists\,2$, * $\exists\,2\,\forall$, * $\forall\,2\,\exists$

c. * $\forall\,2\,\exists$: $\forall y\ [\ \mathsf{city}(y)\ \rightarrow$
$2\ x\ [\mathsf{politicians}(x), \exists z[\mathsf{person}(z)\ \wedge\ \mathsf{from}(z,y)\ \wedge\ \mathsf{spy}(x,z)]\ ]$

[1] In (5b), x ranges over singular and plural individuals (see Link 1983), the formula $|x| \geq 2$ says that x consists of at least 2 atoms and the universal quantification $\forall x'$ corresponds to the distributive interpretation optionally available for plural Noun Phrases (Link 1983 among many others). For convenience, we will use the notation $2\ x[p_1\ \wedge\ p_2]$ for $\exists x\ [p_1\ \wedge\ |x| \geq 2\ \wedge\ \forall x'[x' \subset_i x\ \rightarrow\ p_2]\ ]\ ]$ in subsequent examples.

(7) Two engineers repaired some exits from every freeway in California city. Larson (1987)

Although the missing reading corresponding to $\exists\, 2\, \forall$ may be banned due to general architectural reasons, the unavailability of the reading $*\forall\, 2\, \exists$ formalized in (6b) is puzzling.[2] The aim of the present paper is to provide an LTAG account of why no quantificational NP can intervene between an inverse linked quantifier and its host NP. The paper is part of a larger project concerned with the development of a compositional semantics for LTAG. We first provide some background on LTAG and compositional semantics in section 2. Section 3 develops a flexible composition approach to quantification. Section 4 spells out the semantics for it, generating only the correct scopal combinations for nested quantifier constructions.

2 LTAG and compositional semantics

2.1 Lexicalized Tree Adjoining Grammars (LTAG)

An LTAG (Joshi & Schabes 1997) consists of a finite set of trees (elementary trees) associated with lexical items and of composition operations of substitution (replacing a leaf with a new tree) and adjunction (replacing an internal node with a new tree). The elementary trees represent extended projections of lexical items and encapsulate all syntactic/semantic arguments of the lexical anchor. They are minimal in the sense that only the arguments of the anchor are encapsulated, all recursion is factored away.

LTAG derivations are represented by derivation trees that record the history of how the elementary trees are put together. A derived tree is the result of carrying out the substitutions and adjoinings. See Fig. 1 for an example. The numbers in the derivation tree are the node positions where substitution/adjunction takes place: *John* is substituted for the node at position (1) and *always* is adjoined at position (2).

2.2 Compositional semantics with LTAG

Because of the localization of the arguments of a lexical item within elementary trees TAG derivation trees express predicate argument dependencies. Therefore it is generally assumed that the proper way to define compositional semantics for LTAG is with respect to the derivation tree, rather than the derived tree

[2] $*\exists\, 2\, \forall$ involves having the quantifier $\exists z$ separated from its restrictor $from(z,y)$ as in (8), a configuration that should perhaps be banned on grounds orthogonal to the present paper (it has very weak truth conditions in Predicate Logic). But note that such configuration does not arise in $*\forall\, 2\, \exists$. The unavailability of this reading is, hence, puzzling and needs an explanation.

(8) $*\exists\, 2\, \forall$: $\exists z[person(z) \wedge 2x\ [politicians(x) \wedge$
$\forall y\ [\ city(y) \wedge from(z,y) \rightarrow spy(x,z)]\ ]\ ]$

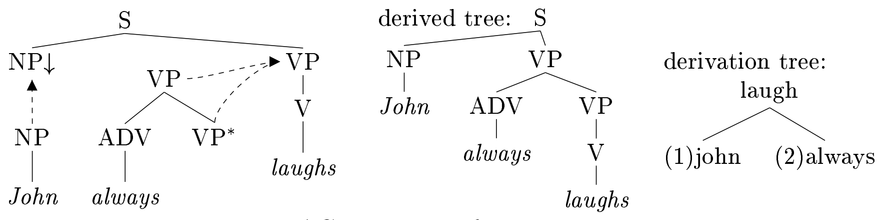


Figure 1: TAG derivation for *John always laughs*

(see, e.g., Candito & Kahane, 1998; Joshi & Vijay-Shanker, 1999; Kallmeyer & Joshi 1999, 2002).

The overall idea is as follows. Each elementary tree is linked to a semantic representation. The way the semantic representations combine with each other depends on the derivation tree. Following Kallmeyer & Joshi (1999, 2002), in this paper, we will adopt 'flat' semantic representations as in, for example, Minimal Recursion Semantics (MRS, Copestake et al., 1999). (9) shows the elementary semantic representations for *John always laughs.*[3]

(9) $\begin{array}{|c|}\hline l_1 : \mathsf{laugh}(x_1) \\ h_1 \geq l_1 \\ \hline \text{arg: } \langle x_1, (1) \rangle \\ \hline \end{array}$ $\begin{array}{|c|}\hline \mathsf{john}(x) \\ \hline \text{arg: } - \\ \hline \end{array}$ $\begin{array}{|c|}\hline l_2 : \mathsf{always}(h_2) \\ g_1 \geq l_2, h_2 \geq s_1 \\ \hline \text{arg: } g_1, s_1 \\ \hline \end{array}$

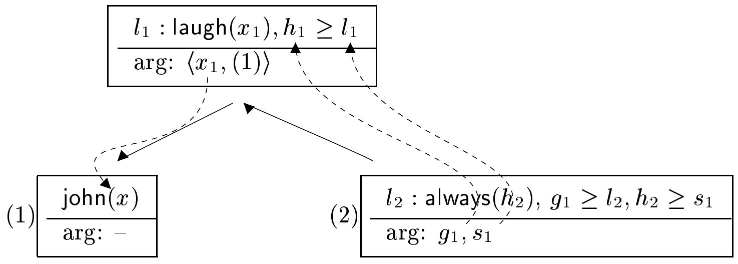
Roughly, a semantic representation consists of a conjunctively interpreted set of formulas (typed lambda-expressions), scope constraints and a set of argument variables. The formulas may contain labels and holes (metavariables for propositional labels). In the following, $l_1, l_2, \ldots$ are propositional labels, $h_1, h_2, \ldots$ are propositional holes, $s_1, s_2, \ldots$ are propositional and $x_1, x_2, \ldots$ individual argument variables (whose values must be propositional labels/free individual variables) and $g_1, g_2, \ldots$ are hole variables (special argument variables whose values must be holes). Argument variables may be linked to positions in the elementary tree, as it is the case for x_1 in (9).

The use of holes is motivated by the desire to generate underspecified representations (as in, e.g., Bos, 1995) for scope ambiguities. After having constructed a (possibly underspecified) semantic representation with holes and labels, disambiguation is done which consists of finding bijections from holes to labels that respect the scope constraints. E.g., in the semantic representation for *laugh*, there is a hole h_1 above l_1 (constraint $h_1 \geq l_1$). Between h_1 and l_1, other labels and holes might intervene (introduced for example by quantifiers or adverbs) or, if this is not the case, l_1 will be assigned to h_1 in the disambiguation(s). The constraints $k_1 \geq k_2$ differ from the qeq conditions $k_1 =_q k_2$ in MRS (see Copestake et al. 1999, p.10) in that they allow any element having a propositional argument (quantifiers, scope-taking adverbs ...) to intervene

[3] $\mathsf{john}(x)$ is not a standard unary predicate but it is supposed to signify "there is a unique individual called John and x refers to that individual".

between k_1 and k_2 while in MRS, just quantifiers can intervene between k_1 and k_2.

When combining semantic representations, values are assigned to argument variables and the union of the semantic representations is built. The values for the argument variables of a certain (elementary) semantic representation must come from semantic representations that are linked to it in the derivation tree. The linking of argument variables and syntactic positions restricts the possible values as follows: In a substitution derivation step at a position p, only argument variables linked to p get values. In an adjunction step, only argument variables that are not linked to any positions can get values. In the case of a substitution, a new argument is inserted and therefore a value is assigned to an argument variable in the old semantic representation. However, in the case of an adjunction, a new modifier is applied and therefore a value is assigned to a variable in the semantic representation that is added. In other words, in case of a substitution, semantic composition is downwards, while in case of an adjunction, semantic composition is upwards. For a formal definition of the semantic composition operation see Kallmeyer & Joshi (2002). The algebra introduced there is close to what Copestake et al. (2001) introduce for MRS except for details of the formalization and for the fact that in Copestake et al. (2001) each semantic representation contains just one "hook", i.e. just one element that can be assigned as possible value to an argument (if equations are viewed as variable assignments). This is different in our approach, e.g. in (9) h_1 and l_1 are contributed by the same elementary representations and they are both used as values when combining *laugh* and *always*.



l_1 : laugh(x_1), $h_1 \geq l_1$
arg: $\langle x_1, (1) \rangle$
(1)
john(x)
arg: –
(2)
l_2 : always(h_2), $g_1 \geq l_2, h_2 \geq s_1$
arg: g_1, s_1

Figure 2: Semantic composition for *John always laughs*

Fig. 2 shows the semantic composition for *John always laughs*. The directed edges signify semantic application while the dotted links signify variable assignments. *John* is substituted into *laugh*, therefore the corresponding semantic composition is downwards, while the composition of *always* and *laugh* is upwards. Furthermore, the value of x_1 needs to come from *John* since x_1 is linked to the node address where *John* is substituted, and the values of g_1 and s_1 need to come from *laugh* since they are not linked to any node addresses. Consequently, $x_1 \to x, g_1 \to h_1$ and $s_1 \to l_1$. The result is (10).

(10)

$l_1 : \mathsf{laugh}(x), \mathsf{john}(x), l_2 : \mathsf{always}(h_2), h_1 \geq l_1, h_1 \geq l_2, h_2 \geq l_1$
arg: –

A disambiguation δ is a bijection from holes to labels such that: After having applied δ (i.e., replaced the holes by their corresponding labels), the reflexive transitive closure $\geq^*$ of the order $\geq$ with (i) $l_1 \geq l_2$ if $l_1 \geq l_2$ is a constraint and (ii) $l_1 \geq l_2$ and $l_1 \neq l_2$ if l_2 labels a subformula of the formula labelled l_1 must be such that (a) $\geq^*$ is a partial order and (b) $l_1 \not\geq^* l_2$ and $l_2 \not\geq^* l_1$ if l_1 and l_2 are different arguments of the same predicate (e.g., the restrictive and the nuclear scope of a quantifier).

In (10), $h_1 \geq l_2$, $l_2 > h_2$ (because h_2 appears inside a formula labelled l_2) and $h_2 \geq l_1$. Consequently $h_1 \neq l_1$ and the only possible disambiguation is $h_1 \to l_2, h_2 \to l_1$. This leads to the semantics $\mathsf{john}(x) \wedge \mathsf{always}(\mathsf{laugh}(x))$.

2.3 Separating scope and predicate argument information

A central aspect of (Kallmeyer & Joshi, 1999, 2002) is the separation of the contribution of a quantifier into a scope and a predicate argument part: Quantifiers have a set of two elementary trees and tree-local multicomponent TAGs are used. (This means that if a new elementary tree set is added, all trees of the set are added simultaneously and they are added to nodes belonging all to the same elementary tree.) An auxiliary tree consisting of a single node is linked to the scope part of the semantics, while an initial tree is linked to the predicate argument part. E.g., consider the syntactic analysis of *every dog barks* in Fig. 3. The corresponding elementary semantic representations are shown in (11).

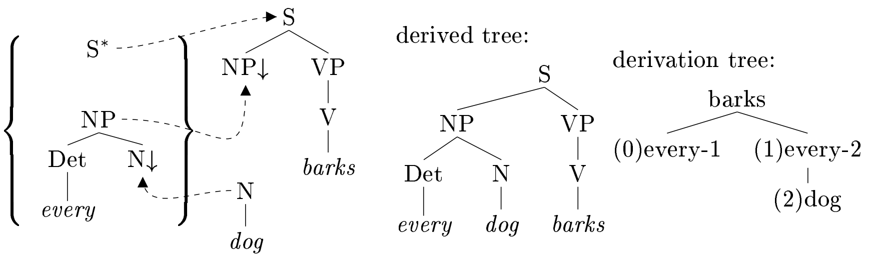


Figure 3: Syntactic analysis of *every dog barks*

(11)

$l_1 : \mathsf{bark}(x_1)$	$l_2 : \forall x(h_2, h_3)$	$l_3 : p_1(x)$	$q_1 : \mathsf{dog}$
$h_1 \geq l_1$	$h_3 \geq s_1$	$h_2 \geq l_3$	
arg: x_1	arg: s_1	arg: p_1	arg: –

The scope part of the quantifier (second representation in (11)) introduces a proposition containing the quantifier, its variable and two holes for its restrictive and nuclear scope. The proposition this semantic representation is applied to (variable s_1) is in the nuclear scope of the quantifier ($h_3 \geq s_1$). The

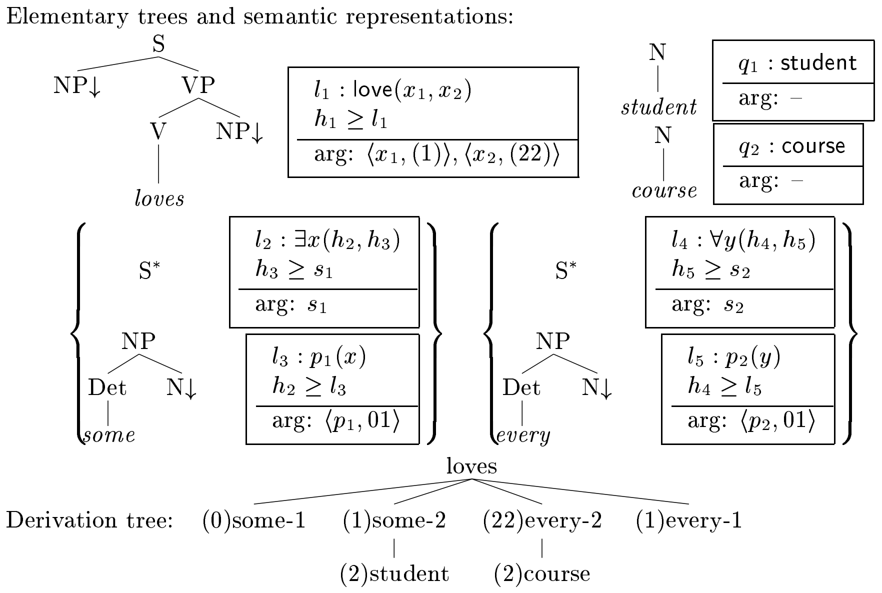


Figure 4: Scope ambiguity and underspecification

predicate argument part (third representation in (11)) introduces a proposition $p_1(x)$ where p_1 will be the noun predicate dog. This proposition is in the restrictive scope of the quantifier ($h_2 \geq l_3$). The values for the argument variables are $x_1 \to x, s_1 \to l_1, p_1 \to q_1$ which gives (12). The only disambiguation is $h_1 \to l_2, h_2 \to l_3, h_3 \to l_1$ which leads to the semantics $\forall x(\mathsf{dog}(x), \mathsf{bark}(x))$.

(12)

$l_1 : \mathsf{bark}(x)$, $l_2 : \forall x(h_2, h_3)$, $l_3 : \mathsf{dog}(x)$, $h_1 \geq l_1$, $h_3 \geq l_1$, $h_2 \geq l_3$
arg: –

To account for cases with more than one quantifier, a restricted use of multiple adjunctions (for the scope parts) is necessary.

As already mentioned above, the use of holes and labels allows to generate underspecified representations for quantifier scope ambiguities as in *some student loves every course*. The elementary trees and elementary semantic representations and the derivation tree are shown in Fig. 4. The assignments are $x_1 \to x, x_2 \to y, s_1 \to l_1, p_1 \to q_1, s_2 \to l_1, p_2 \to q_2$. The result is (13).

(13)

$l_2 : \exists x(h_2, h_3)$, $l_4 : \forall y(h_4, h_5)$, $l_1 : \mathsf{loves}(x, y)$, $l_3 : \mathsf{student}(x)$, $l_5 : \mathsf{course}(y)$, $h_2 \geq l_3, h_3 \geq l_1$, $h_4 \geq l_5, h_5 \geq l_1$, $h_1 \geq l_1$
arg: –

According to (13), $\mathsf{student}(x)$ is in the restriction of $\exists$, $\mathsf{course}(y)$ in the restriction of $\forall$, and $\mathsf{loves}(x, y)$ is in the body of $\exists$ and the body of $\forall$. This leaves open whether $\exists$ is in the body of $\forall$ or $\forall$ in the body of $\exists$. The corresponding two disambiguations are $h_1 \rightarrow l_2, h_2 \rightarrow l_3, h_3 \rightarrow l_4, h_4 \rightarrow l_5, h_5 \rightarrow l_1$ (wide scope of $\exists$) and $h_1 \rightarrow l_4, h_2 \rightarrow l_3, h_3 \rightarrow l_1, h_4 \rightarrow l_5, h_5 \rightarrow l_2$ (wide scope of $\forall$).

There are many related works on computational models for scope representation, e.g., Reyle (1993) that introduces scope constraints and underspecification into DRT. One that has a specific connection to our work is Alshawi (1992). In this work there is an intermediate level of scope representation (Quasi Logical Form (QLF)). At this level underspecified representation of scope is allowed (among other things). This form is computed from a prior phase of syntactic analysis and is produced by an initial semantic analysis phase.

The fact that we provide in our representation a level of underspecification is not the novel part of our system. One of the novel aspects of the compositional semantics developed in Kallmeyer & Joshi (2002) is that the derivation tree (which is the syntactic derivational history in the LTAG system) already represents the underspecified scope relations. Computation of this representation is not a separate level. This is a crucial point of departure from the traditional compositional systems. The other distinguishing aspect is the factoring of the composition of the predicate-argument semantics from the scope composition semantics.

3 LTAG and flexible composition

In a context-free grammar, CFG, a rule such as $A \rightarrow BC$ can be interpreted in two ways. We can regard B as a function and C as its argument, producing the result A. Alternatively, C can be regarded as a function and B as its argument, producing the same result. We have flexible composition here, in the sense that the direction of composition is flexible. In the case of CFGs it is easily seen that providing such flexibility does not affect the weak generative capacity of the grammar (i.e., the set of strings generated by the grammar) as well as the strong generative capacity (i.e., the set of derivation trees generated by the grammar). In other words flexible composition does not buy us anything new. This is due to the fact that CFG is a string rewriting system and function and argument are 'string-adjacent'. For a TAG and, in particular, for the multi-component TAG it can be shown that flexible composition allows the possibility of increasing both the strong and weak generative capacities. This is due to the fact that when TAG trees are composed (interpreting them either as functions or arguments) the function and argument trees are 'tree-adjacent' (rather than 'tree-adjacent). The fact that complex topological objects are composed allows the possibility of increasing strong and weak generative capacities using flexible composition.

We will use some simple examples to illustrate what we mean by flexible composition in a TAG or MC-TAG (Multi-Component TAG). Instead of the two operations, substitution and adjoining, we will use the term 'attachment'.

In Fig. 5 β_1 can be attached to α_1 at the interior S node of α_1 resulting in the tree corresponding to *who*$_i$ NP thinks NP likes ϵ_i. In this case β_1 composes with α_1. Alternatively, we can regard α_1 as a multicomponet tree (with two components) as shown in α_2 with the two components α_{21} and α_{22}. Now we can compose α_2 with β_1 such that α_{21} attaches to the root node of β_1 and α_{22} attaches to the footnode S of β_1, resulting in the same string as before, but with a different derivation (different structural description).

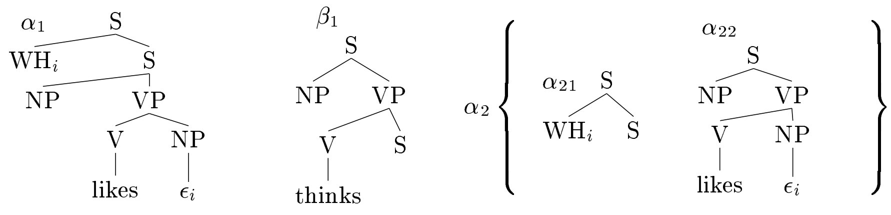


Figure 5: Example of flexible composition

In flexible composition if a tree t composes with a tree u then we require that u is an elementary tree. This assures 'tree locality' in the composition. Given two trees t and u composition can go in either direction if both t and u are elementary. If both t and u are derived trees then they cannot compose with each other. If only one of the trees is elementary then the other tree can compose into it but not vice versa. Given this constraint on locality the composition can proceed in a flexible manner. Of course, several derived trees can be added simultaneously to an elementary tree. This is necessary in order not to exclude standard TAG derivations. Basically, flexible composition allows one to traverse the derivation tree starting at any node and moving up and down the derivation tree in a 'flexible' manner until all nodes of the derivation tree have been visited. Actually, in our present paper we will not use this more general notion of flexible composition. We will traverse the derivation tree (for a MC-TAG) from bottom up maintaining the requirements of flexible composition as described above.

4 The quantifier set approach

In this section we propose a way to obtain the desired scope restrictions for inverse linking constructions making use of the flexible composition approach. Consider again (6), repeated as (14). The reading we want to exclude is $\forall\, 2\, \exists$. (The order $\exists\, 2\, \forall$ will be excluded anyway.)

(14) two politicians spy on someone from every city

In the flexible composition approach, at some point the QPs *someone* and *every city* are composed. In this step, the two scope parts (the S auxiliary trees) of these quantifiers are identified (one adjoins to the other). The result is the complex QP *someone from every city.* Later, this QP and *two politicians* are both added to *spy*, i.e., their scope parts adjoin to the S node of *spy.* In other words, in this latter step the scope parts of the complex QP and of *two* *politicians* are identified. It seems that whenever an identification of scope parts takes place (i.e., either one adjoins to the other or all adjoin to the same node),

- all scope orders are possible between the quantifiers involved in that identification, and
- no other quantifier can intervene (i.e., have scope over one of the quantifiers while being in the scope of another of the quantifiers involved in this identification).

To formalize this, we introduce quantifier sets in our semantic representations. The idea is the following: Whenever several quantifiers are identified, a new set is built containing the scope parts of these quantifiers. Eventually, these scope parts are already sets (as in the case of the complex QP in (14)). E.g., the representation for (14) contains a quantifier set $\{l_1 : 2 \ldots, \{l_3 : \exists \ldots, l_5 : \forall \ldots\}\}$.

The elements of one quantifier set (e.g., $\exists$ and $\forall$ in (14)) are considered being 'glued together' in the sense that no other quantifier can intervene. This is obtained by putting a condition on the scope order that makes sure that if one part of a quantifier set Q_1 is subordinated by one part of another quantifier set Q_2, then all quantifiers in Q_1 must be subordinated by all quantifiers in Q_2. More formally, to the conditions on the relation "$\geq^*$" one obtains after having applied a disambiguation (see (a), (b), p; 6), we add the following: (c) for each quantifier set Q, for all $Q_1, Q_2 \in Q$: if there are labels l_1 in Q_1 and l_2 in Q_2 such that $l_1 >^* l_2$, then for all l_1 in Q_1 and l_2 in Q_2 $l_1 >^* l_2$ holds.

For (14), this excludes $l_3 >^* l_1 >^* l_5$ (excluded anyway because of condition (b) on disambiguations and the separation between scope and restriction of a quantifier) and $l_5 >^* l_1 >^* l_3$, the inverse linking reading we want to exclude.

Let us go through the derivation of (14). Fig. 6 shows its derivation tree. For the scope parts of quantifiers we allow now non-local multicomponent attachments. This does not affect the generative capacity of the grammar.

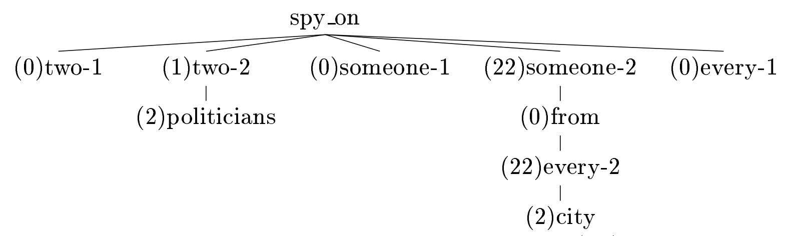


Figure 6: Derivation tree of (14)

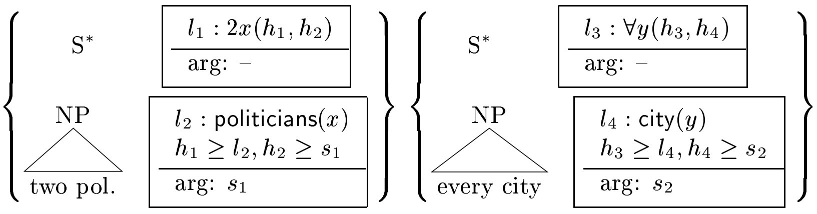
The flexible composition view corresponds roughly to a bottom-up derivation where derived trees are added to elementary trees, i.e., the derivation steps are the following:

1. *politicians* attaches to the lower part of the multicomponent (MC) set of *two* building a larger MC set
2. similarly, *city* attaches to *every* building a MC set

3. the lower part of the MC set of *every city* is substituted into *from*. The result is a new MC set.
4. the MC set of *from every city* is added to the MC set of *someone* with adjunction of the upper component at the scope part and an adjunction of the lower component at the NP. At this point, the first identification of two scope parts takes place. The result is a new MC set.
5. the two MC sets of *two politicians* and *someone from every city* are added to *spy_on* where the two scope parts are adjoined to the root node and the two lower comonents are substituted into the corresponding leaves. At this point, the second identification of scope parts takes place.

Compared to section 2, we slightly modify the semantic representation of quantifiers: the scope part contains only the quantifier with the holes for restriction and body. The scope constraint linking the quantifier to its proposition is part of the lower part of the quantifier. In particular, the variable for the proposition in the nuclear scope of the quantifier (s_1 and s_2 in (15)) is now part of the lower part. This is necessary, since we allow non-local multicomponent adjunction for the scope auxiliary trees. Consequently, the scope part and the predicate-argument part of a quantifier are not necessarily added to the same elementary tree. But the tree the predicate argument part is added to is the tree that contributes the proposition that must be in the nuclear scope of the quantifier. E.g., in (14), the scope part of *every* is identified with the scope part of *someone* and finally added to *spy*. But the proposition that must be part of the nuclear scope of *every* comes from the *from* tree, the tree the predicate argument part of *every* is added to.

(15) shows the multicomponent sets derived for *two politicians* and *every city*. (16) shows the elementary tree for *from*.

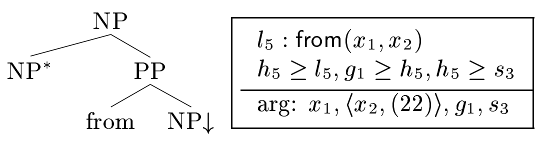
(15)


$l_1 : 2x(h_1, h_2)$
arg: –

l_2 : politicians(x)
$h_1 \geq l_2, h_2 \geq s_1$
arg: s_1

$l_3 : \forall y(h_3, h_4)$
arg: –

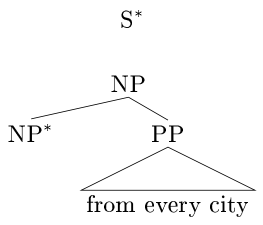
l_4 : city(y)
$h_3 \geq l_4, h_4 \geq s_2$
arg: s_2

(16)


l_5 : from(x_1, x_2)
$h_5 \geq l_5, g_1 \geq h_5, h_5 \geq s_3$
arg: $x_1, \langle x_2, (22)\rangle, g_1, s_3$

Adding *every city* to *from* by substitution of the lower component at the NP leaf inside the PP leads to $x_2 \to y$ and $s_2 \to l_5$. The result is (17).

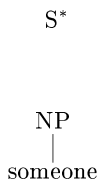
(17)



$$\begin{array}{|l|}\hline l_3 : \forall y(h_3, h_4) \\ \hline \text{arg: } - \\ \hline \end{array}$$

$$\begin{array}{|l|}\hline l_4 : \mathsf{city}(y),\ l_5 : \mathsf{from}(x_1, y) \\ h_3 \geq l_4, h_4 \geq l_5, h_5 \geq l_5, g_1 \geq h_5, \\ h_5 \geq s_3 \\ \hline \text{arg: } x_1, g_1, s_3 \\ \hline \end{array}$$

(18)

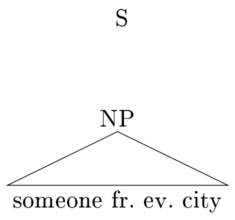


$$\begin{array}{|l|}\hline l_6 : \exists z(h_6, h_7) \\ \hline \text{arg: } - \\ \hline \end{array}$$

$$\begin{array}{|l|}\hline l_7 : \mathsf{person}(z) \\ h_6 \geq l_7, h_7 \geq s_4 \\ \hline \text{arg: } s_4 \\ \hline \end{array}$$

(18) is the MC set for *someone*. When adding *from every city* to (18), the two scope parts are put into one quantifier set. The assignments are $x_1 \to z, g_1 \to h_6, s_3 \to l_7$. ($s_3 \to l_7$ is the only possibility, and $g_1 \to h_7$ would lead to $h_7 \geq h_5 \geq l_7$ and $h_6 \geq l_7$, i.e., to l_7 being in the restrictive and the nuclear scope of *someone*.) One obtains (19).

(19)



$$\begin{array}{|l|}\hline \{l_3 : \forall y(h_3, h_4), l_6 : \exists z(h_6, h_7)\} \\ \hline \text{arg: } - \\ \hline \end{array}$$

$$\begin{array}{|l|}\hline l_4 : \mathsf{city}(y),\ l_5 : \mathsf{from}(z, y),\ l_7 : \mathsf{person}(z) \\ h_3 \geq l_4, h_4 \geq l_5, h_5 \geq l_5, h_6 \geq h_5, \\ h_5 \geq l_7, h_6 \geq l_7, h_7 \geq s_4 \\ \hline \text{arg: } s_4 \\ \hline \end{array}$$

When adding the two QPs, *two politicians* and *someone form every city* to *spy*, the two scope parts are adjoined to the same node and thereby identified. Therefore a large quantifier set is built. The result is (20).

(20)
$$\begin{array}{|l|}\hline \{l_1 : 2x(h_1, h_2), \{l_3 : \forall y(h_3, h_4), l_6 : \exists z(h_6, h_7)\}\} \\ l_2 : \mathsf{politicians}(x),\ l_4 : \mathsf{city}(y),\ l_5 : \mathsf{from}(z, y),\ l_7 : \mathsf{person}(z),\ l_8 : \mathsf{spy}(x, z) \\ h_1 \geq l_2, h_2 \geq l_8, h_3 \geq l_4, h_4 \geq l_5, h_5 \geq l_5, \\ h_6 \geq h_5, h_5 \geq l_7, h_6 \geq l_7, h_7 \geq l_8, h_8 \geq l_8 \\ \hline \text{arg: } - \\ \hline \end{array}$$

The inverse linking reading with the third quantifier intervening is correctly excluded: this reading would mean $l_3 > l_1 > l_6$. Let $Q_1 := \{l_3 : \forall \ldots, l_6 : \exists \ldots\}$ and $Q_2 := l_1 : 2 \ldots$. Then the scope order condition on quantifier sets is not satisfied because $l_3 > l_1$ and $l_6 \not> l_1$.

Other scope taking elements, such as adverbs or modals, are not involved into the quantifier set mechanism since they do not have a separate scope tree.

Therefore, they can intervene freely between quantifiers belonging to the same set. E.g., a reading $\forall$ *want* $\exists$ for (21) is allowed.

(21) John wanted to meet someone from every city   Sauerland (2000)

5 Conclusion

In this paper we provided an LTAG account for certain restrictions on quantifier scope. The approach is part of a larger project on compositional semantics in LTAG. The constructions considered are inverse linking readings for nested quantifiers. I.e., sentences with one quantifying phrase Qu_1 embedded in another quantifying phrase Qu_2 where Qu_1 takes scope over Qu_2. In this case no other quantifier that is on the same level as Qu_2 can scopally intervene between Qu_1 and Qu_2.

In order to explain the fact that some quantifiers seem to be more closely connected than others, we adopted another perspective on TAG derivation, namely a perspective of flexible composition. This allowed to combine first those quantifiers that are closer with respect to scope and that do not allow intervening quantifiers and then to combine larger sets of quantifiers. In our semantics we built corresponding smaller and larger sets of quantifiers that express the constraints on relative quantifier scope that can be observed in inverse linking readings. The flexible composition approach as used in this paper does not increase the generative capacity of the TAG formalism, it is just a specific way of ordering the derivations in a TAG.

Acknowledgments

We would like to thank Chung-Hye Han for valuable discussions of the subject of this paper. Furthermore, we are grateful to two anonymous reviewers for many helpful comments and for drawing our attention to further work on the problems we addressed.

References

Alshawi, H. (ed.): 1992. *The Core Language Engine.* MIT Press.

Bos, J.: 1995. Predicate logic unplugged. *in* P. Dekker and M. Stokhof (eds), *Proceedings of the 10th Amsterdam Colloquium.* pp. 133–142.

Candito, M.-H. & Kahane, S.: 1998. Can the TAG Derivation Tree represent a Semantic Graph? an Answer in the Light of Meaning-Text Theory. *Fourth International Workshop on Tree Adjoining Grammars and Related Frameworks, IRCS Report 98–12.* University of Pennsylvania, Philadelphia. pp. 25–28.

Copestake, A., Flickinger, D., Sag, I. A. and Pollard, C.: 1999. Minimal Recursion Semantics. An Introduction. Manuscript, Stanford University.

Copestake, A., Lascarides, A. and Flickinger, D.: 2001. An Algebra for Semantic Construction in Constraint-based Grammars. *Proceedings of ACL.*

Heim, I. & Kratzer, A.: 1998. *Semantics in Generative Grammar.* Blackwell.

Hobbs, J. & Shieber, S. 1987. An algorythm for generating quantifier scopings. *Computational Linguistics* **13**, 47–63.

Joshi, A. K. & Schabes, Y.: 1997. Tree-Adjoning Grammars. *in* G. Rozenberg and A. Salomaa (eds), *Handbook of Formal Languages.* Springer. Berlin. pp. 69–123.

Joshi, A. K. & Vijay-Shanker, K.: 1999. Compositional Semantics with Lexicalized Tree-Adjoining Grammar (LTAG): How Much Underspecification is Necessary?. *in* H. C. Blunt and E. G. C. Thijsse (eds), *Proceedings ot the Third International Workshop on Computational Semantics (IWCS-3).* Tilburg. pp. 131–145.

Kallmeyer, L. & Joshi, A. K.: 1999. Factoring Predicate Argument and Scope Semantics: Underspecified Semantics with LTAG. *in* P. Dekker (ed.), *12th Amsterdam Colloquium. Proceedings.* Amsterdam. pp. 169–174.

Kallmeyer, L. & Joshi, A. K. 2002. Factoring Predicate Argument and Scope Semantics: Underspecified Semantics with LTAG. *Journal of Language and Computation.* To appear.

Larson, R.: 1987. Quantifying into np. Ms. MIT.

Link, G.: 1983. The logical analysis of plurals and mass terms. *in* R. B. et al. (ed.), *Meaning, Use and Interpretation of Language.* de Grüyter. Berlin. pp. 302–323.

May, R.: 1985. *Logical Form. Its Structure and Derivation.* MIT Press. Cambridge, Mass.

Reyle, U. 1993. Dealing with ambiguities by underspecification: Construction, representation and deduction. *Journal of Semantics* **10**, 123–179.

Sauerland, U.: 2000. Syntactic economy and quantifier raising. Ms. University of Tübingen.