Semantic Interpretation of Unrealized Syntactic Material in LTAG

Olga Babko-Malaya
University of Pennsylvania
malayao@ldc.upenn.edu

Abstract

This paper presents a LTAG-based analysis of gapping and VP ellipsis, which proposes that resolution of the elided material is part of a general disambiguation procedure, which is also responsible for resolution of underspecified representations of scope.

1 Introduction

The problem of ellipsis resolution is to recover the interpretation of the elided material. For example, in (1), the elided VP is interpreted as being identical to the verb in the preceding sentence. Likewise, in the gapping structures, as shown in (2), the interpretation of a gap is being identified with the interpretation of the preceding verb.

(1) Mary likes Bill. Jane does too.
(2) Mary ate beans and others -- rice.

Whereas some approaches assume syntactic identity between the antecedent and the elided material (e.g. Fiengo and May 1994), others suggest that VP ellipses are proforms, semantically identified with their antecedents (see Dalrymple et al 1991, Shieber et al 1996, Hardt 1993, 1999).

This paper follows semantic approaches to ellipsis resolution. It adopts the LTAG semantics of Kallmeyer and Romero 2004 and proposes that resolution of ellipsises and gaps is part of a general disambiguation procedure, which is also responsible for resolution of underspecified representations of scope.

2 LTAG Semantics with Semantic Unification

In LTAG framework (Joshi and Schabes 1997), the basic units are (elementary) trees, which can be combined into bigger trees by substitution or adjunction. LTAG derivations are represented by derivation trees that record the history of how the elementary trees are put together. Given that derivation steps in LTAG correspond to predicate-argument applications, it is usually assumed that LTAG semantics is based on the derivation tree, rather than the derived tree (Kallmeyer and Joshi 2003).

Semantic composition which we adopt is based on LTAG semantics with semantic unification (Kallmeyer and Romero 2004). In the derivation tree, elementary trees are replaced by their semantic representations and corresponding feature structures. Semantic representations are as defined in Kallmeyer and Joshi 2003, except that they do not have argument variables. These representations consist of a set of formulas (typed λ-expressions with labels) and a set of scope constraints.

Each semantic representation is linked to a feature structure. Feature structures, as illustrated by different examples below, include a feature i whose values are individual variables and features p and MaxS, whose values are propositional labels. Semantic composition consists of feature unification. After having performed all unifications, the union of all semantic representations is built.

Consider, for example, the semantic representations and feature structures associated with the elementary trees of the sentence shown in (3).

(3) Mary dates Bill

\[
S \\
\begin{array}{c}
NP \\
[v_1] \\
date \\
NP \\
[i: v_2]
\end{array}
\begin{array}{c}
NP \\
[i: x]
\end{array}
\begin{array}{c}
Bill (y) \\
[i: y]
\end{array}
\]

\[
\begin{array}{c}
Mary (x) \\
[i: x]
\end{array}
\]

\[
\begin{array}{c}
date \\
[i: v_1]
\end{array}
\begin{array}{c}
[i: v_2]
\end{array}
\]

\[
date (v_1, v_2)
\]

\[
S \\
\begin{array}{c}
NP \\
[v_1] \\
date \\
NP \\
[i: v_2]
\end{array}
\begin{array}{c}
NP \\
[i: x]
\end{array}
\begin{array}{c}
Bill (y) \\
[i: y]
\end{array}
\]

\[
\begin{array}{c}
Mary (x) \\
[i: x]
\end{array}
\]

\[
\begin{array}{c}
date (v_1, v_2)
\end{array}
\]

\[
\begin{array}{c}
do ...
The final representation of this sentence is underspecified for scope, given that there are no constraints which restrict the relative scope of every and some. In order to obtain one of the readings, a disambiguation mapping is needed: Disambiguations:
1. $R_2 \to l_s, R_3 \to l_s, N_2 \to l_s, N_1 \to l_s;\;\text{some}(y,\text{course}(y)), \text{every}(x,\text{student}(x), \text{like}(x, y))$
2. $R_2 \to l_s, R_3 \to l_s, N_2 \to l_s, N_1 \to l_s;\;\text{every}(x,\text{student}(x), \text{some}(y,\text{course}(y), \text{like}(x, y))$

Disambiguations are functions from propositional variables to propositional labels that respect the scope constraints, such that after having applied this mapping, the transitive closure of the resulting scope is a partial order.

3 The Problem of Ellipsis Resolution in LTAG semantics

Given LTAG semantics, there are two possible approaches to resolution of the elided material: reconstruction can be done as part of the unification process or as part of the disambiguation procedure. If reconstruction was done as unification, the semantic representation of the elided material would be disambiguated in the final representation. On the other hand, it is well known that resolution of ellipses and gaps can be ambiguous. For example, the sentence in (6), discussed in Siegel 1987 and Johnson 2003 among others, has 2 interpretations:\footnote{Other cases of ambiguous interpretations of the elided material are discussed in section 7.}

(6) Ward can’t eat caviar and his guests -- dried beans
Can’t (eat (ward, caviar)) & eat (his guests, dried beans))
Can’t (eat(ward, caviar)) & can’t (eat(his guests, dried beans))

As this example shows, the gap in (6) can be reconstructed by selecting either the verb or the negated modal as its antecedent. The two interpretations represent different scope readings between the conjunction and negation, which should be analyzed as underspecified in LTAG semantics. Resolution of gaps, therefore, cannot be done as part of unification, since it depends on the disambiguated interpretation. The question is whether it is possible to define an underspecified representation of these two readings, and what kind of resolution mechanism can be used to disambiguate these interpretations?
4 LTAG Semantics of Gapping

In LTAG semantics, semantic representations are introduced by lexicalized trees. In order to account for the analysis of gapping and VP ellipsis, this paper proposes that semantics should be defined on both lexicalized and non-lexicalized trees. Specifically, we propose that

**Interpretation of a gap (or elided VP) is the semantic interpretation of a non-lexicalized S tree.**

The semantic representations of lexicalized S trees under this new approach are derived compositionally, given the meaning of a non-lexicalized S tree and the meaning of a verb.

(7) S
NP[iv_1] VP
V [Ag: v_3, Pat: v_4, MaxS: C_1]
NP[iv_2] V
[Ag: v, Pat: u, MaxS: C]
l_1: \lambda u_1 \lambda v. C (v) (v_1)
NP[iv_2]
[Ag: v, Pat: u, MaxS: C]
l_2: \lambda u_1 \lambda v_1, C_1 (y) (x)

Non-lexicalized trees introduce a propositional label and a propositional variable, illustrated by l_1 and C above. If a tree is a transitive S-tree, there are two lambda bound variables, which correspond to the Agent and Patient features of the verb. Performing feature unifications (v_i=v, v_e=u, C_1=C) and scope constraint disambiguations (C->l_0), the proposition l_1 will be reduced to: \lambda u_1 \lambda v. date(v, u)(v_1)=date(v_1, v_2).

Given this proposal, we suggest that the semantics of gaps, VPE and other types of ellipses are identified with their antecedents as part of the disambiguation procedure. The sentence in (7), shown below, differs from the previous one in the presence of a negated modal. The interpretation of this modal introduces a proposition l_0: can’t(N_0) and a constraint P_2 ≤ N_0. After P_2 is unified with the proposition l_0, the final representation has two constraints on the variable l_0: l_0 ≤ C and l_0 ≤ N_0, and therefore two possible disambiguations. In the disambiguation 1, C is mapped to l_0, introduced by the modal, and l_2 and l_3 are reduced to can’t(eat(x, y)) and can’t(eat(z, w)). These disambiguations yield the desired interpretations of this sentence.

(8) Can’t eat caviar and his guests -- dried beans
5 LTAG Analysis of VP Ellipsis

The analysis of gapping presented above can be easily extended to the analysis of VP ellipsis. VPE differs from gapping in that it is not restricted to coordinated structures. Whereas in the examples above resolution of gaps was enforced by the feature structure of ‘and’, in the case of VPE, a similar unification, forced by pragmatic constraints, results in recovering the elided material.

As the example in (9) illustrates, our analysis of VPE assumes the following modification of the semantics of non-lexicalized trees: propositions introduced by non-lexicalized trees have one lambda-bound variable, so that each argument is introduced by a separate proposition. For example, the interpretation of a transitive tree below has two propositions \( l_1 \) and \( l_2 \), and two propositional variables \( C_1 \) and \( C_2 \). The proposition \( l_2 \) corresponds to the meaning of a VP, which is missing in the standard TAG-based analyses. This decomposition of the meaning of a non-lexicalized tree, therefore, can be independently motivated by the existence of modifiers which predicate of VPs. We further assume that the MaxS feature of the S tree corresponds to the variable introduced by the agent (or the highest-ranked argument).

(9) Mary likes Bill. Jane does too.

Applying disambiguations \( C_2 \rightarrow l_0 \), \( C_1 \rightarrow l_2 \), we derive the following propositions:

\[
\begin{align*}
l_1 &: \lambda v C_1 (x) \\
l_2 &: \lambda u C_2 (y) \quad l_2 \leq C_1 \\
l_0 &: \lambda u C_2 (y) \quad l_0 \leq C_2 \\
l_0 &: \lambda (v, u) \quad l_0 \leq C_2 \\
l_1 &: \lambda v C_1 (x) \quad l_1 \leq C_1 \\
l_0 &: \lambda (v, u) \quad l_0 \leq C_2 \\
l_1 &: \lambda v C_1 (x) \quad l_1 \leq C_1 \\
l_0 &: \lambda u C_2 (y) \quad l_0 \leq C_2 \\
l_0 &: \lambda (v, u) \quad l_0 \leq C_2 \\
l_1 &: \lambda v C_1 (x) \quad l_1 \leq C_1 \\
l_0 &: \lambda u C_2 (y) \quad l_0 \leq C_2 \\
l_0 &: \lambda (v, u) \quad l_0 \leq C_2 \\
\end{align*}
\]

Resolution of gaps under this analysis is done as part of the scope resolution procedure on underspecified representations. A crucial feature of this analysis is that the propositions \( l_2 \) and \( l_1 \) are ‘underspecified’ in the final representation and the variable \( C \) is computed during the disambiguation, i.e. when all scope ambiguities are being resolved. In this respect this analysis differs from previous approaches, where the final representation did not include any variables, except for the arguments of quantifiers or other scopal elements.\(^2\)

\(^2\) However, see Babko-Malaya 2004, where a similar analysis is proposed to account for the semantics of coordinated structures with quantified NPs.
Now consider the second sentence: Jane does too:

\[
\text{NP}[i; v_s] \quad \text{VP} \quad l_1; \lambda v_s, C_3 (v_s)
\]

This sentence introduces an intransitive tree and one propositional variable \( C_3 \). This variable is not constrained within the sentence, and parallel to other pro-forms, it gets its interpretation from the previous discourse. Specifically, the interpretation of the second sentence is derived by unification of the S features of the second and the first S-trees in (9): \( C_3 = C_1 \), \( v_s = v \). Given that \( C_1 \) is mapped to \( l_1 \) above, it corresponds to the proposition being reconstructed: \( C_3 (\equiv C_1) \rightarrow l_2 \).

\[
l_1; \lambda v, \text{like}(v, u) (r) = \text{like}(r, u)
\]

### 6 Scope Parallelism

Many previous approaches impose parallelism constraints on the interpretation of the elided material (e.g., Fox 2000, Asher et al 2001 among others). Under the present analysis, scope parallelism comes for free. Consider, for example, the following sentence discussed in Dalrymple et al 1991, among others, where ambiguity is resolved in the same way in both the antecedent and at the ellipsis site: *John gave every student a test, and Bill did too.* The final interpretation of the first sentence is given in (10) and has 2 possible disambiguations.

\[
(10) \text{John gave every student a test.}
\]

\[
l_0; \text{give}(v, u, w)
\]

\[
l_1; \lambda v, C_1 (x) \quad l_2; \lambda u, C_2 (y) \quad l_2 \leq C_1
\]

\[
l_1; \lambda w, C_3 (z) \quad l_1 \leq C_2
\]

\[
l_1; \text{every}(y, R_7, N_7) \quad l_2; \text{some}(z, R_3, N_3)
\]

\[
l_1; \text{student}(y) \quad l_1; \text{test}(z) \quad \text{john(x)}
\]

\[
l_2 \leq C_1, l_2 \leq N_4, l_2 \leq N_7, l_2 \leq R_7, l_1 \leq R_4
\]

The surface reading (every >> some) is derived by the following mapping: \( C_1 \rightarrow l_0 \), \( C_2 \rightarrow l_3 \), \( R_7 \rightarrow l_8 \), \( N_7 \rightarrow l_9 \), \( C_3 \rightarrow l_5 \), \( R_3 \rightarrow l_6 \), \( N_3 \rightarrow l_9 \).

The interpretation of the second sentence is derived by unifying the S-features of the S-trees (as shown in the previous section). As the result, the variables \( C_3 \) and \( v_3 \) are unified with the variables \( C_1 \) and \( v \). Given that \( C_1 \) is being mapped to the proposition \( l_1 \) above, \( C_3 \) is being reconstructed as the proposition \( \lambda v, \text{like}(v, u) \) (r) = \( \text{like}(r, u) \). The second sentence is derived by unification of the S-features of the second sentence is derived by unification of the S-features of the second sentence.

\[
(11) \text{Bill did too.}
\]

\[
l_1; \lambda v, \text{like}(v, u) (r) = \text{like}(r, u)
\]

\[
C_3 (\equiv C_1) \rightarrow l_7
\]

The interpretation of the second sentence can be obtained by the following mapping: \( C_3 \rightarrow l_0 \), \( C_2 \rightarrow l_3 \), \( R_7 \rightarrow l_8 \), \( N_7 \rightarrow l_9 \), \( C_7 \rightarrow l_5 \), \( R_8 \rightarrow l_6 \), \( N_8 \rightarrow l_9 \), \( l_2; \text{give}(v, y, z)
\]

\[
l_1; \text{every}(y, \text{student}(y), \text{give}(v, y, z))
\]

Now, when the second sentence is interpreted, \( C_3 \) is unified with \( C_1 \), which is being mapped to \( l_0 \). The proposition \( l_1 \), then, is reduced to: \( \lambda v, \text{like}(v, u) \) (r) = \( \text{like}(r, u) \). As this example illustrates, scope parallelism follows from the present analysis, given that \( C_3 \) is unified with a disambiguated interpretation of a VP. It can also be shown that the wide scope puzzle (Sag 1980), shown in (12) is not unexpected under this approach, however, the analysis of this phenomenon is beyond the scope of this paper.

\[
(12) \text{A nurse saw every patient. Dr. Smith did too.}
\]

\[
\text{some}(x, \text{nurse}(x), \text{every}(y, \text{patient}(y), \text{see}(x, y)))
\]

\[
\ast \text{every}(y, \text{patient}(y), \text{some}(x, \text{nurse}(x), \text{see}(x, y)))
\]

---

3 As Hirschbuhler 1982, Fox 2000 among others noted, there are constructions where subjects of VPE can have narrow scope relative to nonsubjects. For example, the sentence *A Canadian flag was hanging in front of every building. An American flag was too* has a reading in which each building has both an American and a Canadian flag standing in front of it. The existence of such readings does not present a problem for the present analysis, if we adopt an analysis of quantificational NPs proposed in Babko-Malaya 2004.
7 Antecedent Contained Deletion (ACD)


\[
(13) \text{John wants Mary to read every book Bill does.}
\]

The elided material in this sentence is understood as either “Bill reads” or “Bill wants Mary to read”. Given that ‘want’ and ‘every’ can take different scope, four possible readings are expected. However, puzzling in this case is the unavailability of one of these readings: “John wants that for every book that Bill wants Mary to read, she reads it. Let us consider the final interpretation of this sentence:

\[
\begin{align*}
&\text{l}_0: \text{want}(v_0, N_o) & &\lambda v_0. C_2(\text{want})(r)
\\&\text{l}_1: \text{read}(v, u) & &\lambda v. C_1(\text{read})(x)
\\&\text{l}_c: \lambda x. C_4(\text{read})(y) & &\text{l}_c \leq C_1 & &\text{l}_c: \text{book}(y) \land l_3
\\&\text{l}_3: \text{every}(y, R_5, N_5) & &\lambda v_3. C(z)
\\&\text{mary}(x), \text{john}(r), \text{bill}(z)
\\&l_1 \leq C_1, l_3 \leq C_2, l_1 \leq N_5, l_1 \leq R_5, l_1 \leq N_4
\end{align*}
\]

The non-lexicalized S tree introduces a proposition \( l_1 \) and variables \( C \) and \( v_3 \). These variables can be unified with either \( S \) features of the ‘read’-tree (i.e. \( C_1 \) and \( v \)), or \( S \) features of the ‘want’-tree (i.e. \( C_2 \) and \( v_0 \)). In the first case, the small ellipsis interpretation is derived, and both scope readings are available: \( C = C_1, v_3 = v \)

\[
\text{C/C}_1 \rightarrow l_0, C_2 \rightarrow l_1
\]

\[
\text{C/C}_1 \rightarrow l_0, C_2 \rightarrow l_1
\]

\[
\text{every} >> \text{want}:
\]

\[
\text{N}_5 \rightarrow l_0, C_2 \rightarrow l_1, N_5 \rightarrow l_1, R_3 \rightarrow l_8
\]

\[
\text{l}_5: \text{every}(y, \text{book}(y) \land \text{read}(z, y), \text{want}(r, \text{read}(x, y)))
\]

\[
\begin{align*}
&\text{l}_1 \leq C_1, l_3 \leq C_2, l_1 \leq N_5, l_1 \leq R_5, l_1 \leq N_4
\end{align*}
\]

The fourth possible reading, where \( \text{want} >> \text{every} \), however, is predicted to be unavailable under the present assumptions. This reading, \( \text{want}(r, \text{every}(y, \text{book}(y) \land \text{want}(z, \text{read}(x, y)), \text{read}(x, y))) \), cannot be derived, since it requires the proposition \( l_3 \) to be ‘inserted’ within the proposition \( l_0 \).

References


Babko-Malaya O. 2004 “LTAG Semantics of NP-Coordination” in Proc. of TAG+7, Vancouver


Hardt, D. 1993. VP Ellipsis: Form, Meaning and Processing. PhD Diss. Univ. of PA


In linguistics, semantic analysis is the process of relating syntactic structures, from the levels of phrases, clauses, sentences and paragraphs to the level of the writing as a whole, to their language-independent meanings. It also involves removing features specific to particular linguistic and cultural contexts, to the extent that such a project is possible. The elements of idiom and figurative speech, being cultural, are often also converted into relatively invariant meanings in semantic analysis. Parts of the sentence are notional sentence constituents, which are in certain syntactic relations to other constituents or to the sentence as a whole. Accordingly we distinguish between principal parts of the sentence, constituting the predication, or the basic structure of the sentence, and secondary parts of the sentence, extending, or expanding the basic structure. Parts of the sentence are notional constituents as they name elements of events, or situations denoted by the sentence; actions or states, different participants and circumstances. The formal properties of parts of the sentence are notional sentence constituents as they name elements of events, or situations denoted by the sentence; actions or states, different participants and circumstances. The formal properties of parts of the sentence:

```
Answer: c Explanation: Semantic interpretation is the process of associating a FOL expression with a phrase. 3. What is meant by compositional semantics? a) Determining the meaning b) Logical connectives c) Semantics d) None of the mentioned View Answer. Answer: a Explanation: Compositional semantics is the process of determining the meaning of P\*Q from P, Q and "\*". Answer: c Explanation: Some kind of sentence in the semantic interpretation can't be logical term nor a complete logical sentence. 6. How many verb tenses are there in the English language? a) 1 b) 2 c) 3 d) 4 View Answer. Answer: c Explanation: There are three types of tenses available in English language are past, present and future. The usage of the method of semantic interpretation based on component analysis when interpreting the meaning of the word does not answer the question on the procedure for the allocation of semantic features. In this sense, a special place is occupied by experimental method. At its core lies.
```

The first stage of the test is the selection of the material. As an example, take the verbs of vision. First factual materials were collected, which consisted of separate proposals containing verbs of vision. Mother tongue of the researcher is Russian, therefore, in the selection of the verbs of the language criterion for inclusion of a verb into a set of verbs for analysis was a linguistic intuition. The material sources were fiction, publicistic literature and a number of scientific texts.