Disambiguation and Optional Co-Composition

Pablo Gamallo (1), Gabriel P. Lopes (1), Alexandre Agustini (2)
(1) CITI, Faculdade de Ciências e Tecnologia,
Universidade Nova de Lisboa, Portugal
{gamallo,gpl}@di.fct.unl.pt
(2) Pontifícia Universidade Católica Rio Grande do Sul (PUCRS), Brazil.
agustini@inf.pucrs.br

Mots-clefs – Keywords

Désambiguïsation du sens des mots, Désambiguïsation structurale, Co-composition, Acquisition de restrictions de sélection
Word sense disambiguation, Structural disambiguation, Co-composition, Selection restrictions acquisition

Résumé - Abstract

Cet article décrit une propriété sémantique propre aux dépendances syntaxiques binaires: la co-composition. On proposera ici une définition plus générale que celle donnée par Pustejovsky et que nous appelons "co-composition optionnelle". L'objet de cet article est de montrer les avantages apportées par la co-composition optionnelle dans deux tâches particulières en TAL: la désambiguïsation du sens des mots et la désambiguïsation structurale. Concernant cette deuxième tâche, nous décrirons les expériences faites sur un corpus.

This paper describes a specific semantic property underlying binary dependencies: co-composition. We propose a more general definition than that given by Pustejovsky, what we call "optional co-composition". The aim of the paper is to explore the benefits of optional co-composition in two disambiguation tasks: both word sense and structural disambiguation. Concerning the second task, some experiments were performed on large corpora.

1 Introduction

The objective of this paper is to describe the role of binary syntactic dependencies (i.e., "head-dependent" relations) in two specific NLP tasks: both word and structural disambiguation. Our work is mainly based on two assumptions: first, there are two semantic structures underlying a dependency: one where the dependent word is semantically required by the head, and another where the head word is semantically required by the dependent. Second, for some particular tasks such as word disambiguation or syntactic attachment, we also assume that one of both structures can be more discriminant. So, in order to select a word sense or a syntactic attachment, the most discriminant structure will be retained, in particular, the one containing the least ambiguous word. In special cases, for instance when the two related words are highly ambiguous, both structures can be retained. We call this phenomenon *optional co-composition*

The main contribution of this paper is to define some properties of optional co-composition as well as how it is involved in two disambiguation tasks: word sense disambiguation and attachment resolution (syntactic structure disambiguation).

This paper is organized as follows. First, section 2 will introduce what we consider to be the restrictive structure of a dependency. This structure will be defined on the basis of the notion of optional co-composition. The description will be focused on a particular task: word sense disambiguation. Finally, in section 3, we will focus on a different task: the acquisition of selection restrictions from corpora, and the use of selection restrictions to solve structural ambiguity. Some empirical results will be given.

2 Optional Co-composition

We assume that two words related by a syntactic dependency impose semantic restrictions on each other. Not only verbs and adjectives may select different senses of nouns, but also nouns must be taken as active selectors of senses of verbs, adjectives, and other nouns. We call this property of dependencies "optional co-composition".

In (5), the co-composition operation is activated only in some specific binary dependencies. In particular, it is triggered off if both the verb and the noun contain very specific lexical information. In Generative Lexicon, the scope of this particular operation is then very narrow. We consider, however, that co-composition is a general semantic property underlying any syntactic dependency between two words. In this section, we will propose a more general notion of co-composition than the one proposed by Pustejovsky. To do it, functional application will not be driven by relational words such as verbs and adjectives, but by syntactic dependencies.

We consider dependencies as active objects that control and regulate the selection requirements imposed by the two related words. So, they are not taken here as merely passive syntactic cues related in a particular way (linking rules, syntactic-semantic mappings, syntactic assignments, etc.) to thematic roles or lexical entailments of verbs (1). They are conceived of as the main functional operations taking part in the process of sense interpretation.

On this basis, we associate functional application, not to relational expressions (verbs, adjectives, ...), but to dependencies. In functional terms, a dependency can be defined as a binary λ-expression:

$$\lambda x \lambda y \, dep(x, y) \tag{1}$$

where x and y are variables for word meanings. The meaning of the head word, x, will be in the first position, while the meaning of the dependent, y, will be in the second one. The different types of dependency we consider are the following: nominal verb complement situated to the left of the verb ($lobj$), or to the right of the verb ($robj$), prepositional complement of the verb ($iobj\_prep - name$), prepositional complement of the noun ($prep - name$), and attributive function of the adjective ($attr$).

The objective of this subsection is to show how dependencies can be used to disambiguate words in a co-compositional way. Take the expression "drive the tunnel". In WordNet, "drive" has 21 senses; one of them represents the event of making a passage by excavating. By contrast, "tunnel" merely has 2 very related senses. In order to interpret any composite expression, we argue that the hearer/reader uses the least ambiguous word as disambiguator. In "drive the tunnel", it is the noun that selects for a specific verb sense: the *making* sense. The word disambiguation strategy we propose here consists of the following 3 steps:

1. Identifying a dependency function:
From the verb-noun expression, the $robj$ binary function is proposed:

$$\lambda x \lambda y \, robj(x, y) \tag{2}$$

2. Choice of a word disambiguator:
The dependency function is applied first to the word considered to be the best discriminator. By default, it will be the word with the least number of senses, that is, the least polysemous word. As has been said before, the chosen word must be "tunnel". As a result, this word is assigned to the dependent position of $robj$:

$$\begin{array}{l} [\lambda x \lambda y \, robj(x, y)] \ (tunnel) \\ \lambda x \, robj(x, tunnel) \end{array} \tag{3}$$

This is still a predicative function likely to be applied to the word in the head position. Consequently, word "tunnel", in the dependent position, is taken here as the active predicate.
3. Restrictions of the predicate and Final Application:
The selection restrictions imposed by "tunnel" in the $robj$ dependency represent the classes of verbs with which that noun can combine in this dependency. In the next section, we will outline how word classes can be learned from corpus data. The predicative function associated to "tunnel" is applied to verb "drive":

$$\begin{array}{l} [\lambda x \, robj(x, tunnel : \sigma)] \ (drive) \\ robj(drive_{\sigma}, tunnel) \end{array} \tag{4}$$

The requirements imposed by the nominal predicate, and noted σ, allow to select a particular sense of the verb if and only if, at least, one of the 21 senses of "drive" belongs to σ. $drive_{\sigma}$ represents the particular sense of "drive" that is compatible with restrictions σ. Such a procedure is independent of the way we represent (as features, word clusters, probabilities, etc.) word senses and selection restrictions.

This strategy is more efficient than the standard compositional approaches, since here the disambiguation process is controlled by the word that is considered to be the most appropriate to discriminate the sense of the other one. Moreover, optional co-composition makes functional application more flexible, since it allows to choose as predicative function whatever word within a dependency, or even, if necessary, both words. Any word of a binary dependency may become the lexical function and, then, be used to disambiguate the meaning of the other word.

Nevertheless, word disambiguation should not be restricted to a single binary dependency. The target word is actually disambiguated by all words to which it is syntactically related. So, the disambiguating context of a word is not only a single dependency, but also the set of dependencies it participates in. This remains beyond the scope of the paper.

We have described in this section the internal structure of syntactic dependencies and how they can be used to disambiguate words in a flexible way. In the following section, we will see the benefits of optional co-composition in a different task: syntactic disambiguation.

3 Using Co-composition to Solve Syntactic Ambiguity

This section describes a method to solve syntactic attachment. First, we acquire selection restrictions from corpora, then the acquired information is used to build a subcategorization lexicon. Finally, a specific heuristic is used to propose correct syntactic attachments. The main characteristic of the method is the use of the assumption on optional co-composition introduced in the previous section. This method has been accurately described in (2).

3.1 Selection Restrictions Acquisition

An experiment to automatically acquire selection restrictions was performed on Portuguese corpora[1]. We used an unsupervised and knowledge-poor method. It is unsupervised because no training corpora semantically labeled and corrected by hand is needed. It is knowledge-poor since no handcrafted thesaurus such as WordNet nor no MRD is required (3). The method consists of the following steps. First, raw text is automatically tagged and then analyzed in binary syntactic dependencies using a simple heuristic based on Right Association. For instance, the expression “the salary of the secretary" gives rise to the relation:

$$of(salary, secretary) \tag{5}$$

Then, following the assumption on co-composition, we extract two different functional predicates from every binary dependency. From (5), we extract:

$$\lambda y\, of(salary, y),\ \ \lambda x\, of(x, secretary) \tag{6}$$

Finally, we generate clusters of predicates by computing their word distribution. We assume, in particular, that different predicates are considered to impose the same selection restrictions if they have similar word distribution. Similarity is calculated by using a particular version of the Lin coefficient (4). As a result, a predicate like $\lambda y\ of(salary, y)$ may be aggregated into the following cluster:

$$\lambda y\, of(salary, y),\ \ \lambda y\, of(post, y),\ \ \lambda y\, lobj(resign, y),\ \ \lambda x\, attr(x, competent) \tag{7}$$

which is associated to those words co-occurring at least once with each predicate of the cluster, e.g.:

[1] 3 million words belonging to the P.G.R. (*Portuguese General Attorney Opinions*) corpora, which is constituted by case-law documents. Due to space restrictions, we will give only English translations.

secretary (*secretary*)

· $\lambda x\ of(x, secretary)$ = (*post, career, category, qualification, rank, status, function, remuneration, job, salary*)
· $\lambda y\ of(secretary, y)$ = (*administration, assembly, authority, council direction, company, entity, state, government , institute, judge, minister, ministery, president, service, tribunal organ*)
· $\lambda x\ iobj\_to(x, secretary)$ = (*allude, apply, attend, assign, concern, correspond, determine, resort, refer, relate*)
· $\lambda x\ iobj\_to(x, secretary)$ = (*concern, be-incombent, concede, confer, trust, send, be-incombent, belong*)
· $\lambda x\ iobj\_by(x, secretary)$ = (*sign, concede, confer, homologate, compliment, subscribe*)
· $\lambda x\ lobj(x, secretary)$ = (*define, establish, make, fix, indicate, foresee, refer*)

Table 1: Excerpt of lexicon entry *secretary*

```
secretary, president, minister, manager, worker, journalist
```

We use these words to extensionally define the selection restrictions imposed by the similar predicates of cluster (7). In fact, the set of words required by similar predicates represents the extensional description of their semantic preferences.

3.2 Building a Subcategorization Lexicon

The acquired clusters of predicates and their associated words are used to build a lexicon with syntactic and semantic subcategorization information. Table 1 shows an excerpt of the information learned concerning the entry *secretary*. This entry defines six different predicative structures. Notice that it is the notion of co-composition that allows us to define a great number of predicates that are not usual in the standard approaches to subcategorization. Five of the six predicates with *secretary* do not subcategorize standard dependent complements, but different types of heads. This is a significant novelty of our approach.

3.3 Attachment Heuristic

Optional co-composition is also at the center of syntactic disambiguation. It underlies the heuristic we use to check if two phrases are dependent or not. This heuristic states that two phrases are syntactically attached only if one of these two conditions is verified: either the *dependent* is semantically required by the *head*, or the *head* is semantically required by the *dependent*. Take the expression:

correspond to the secretary of the minister

There exist at least three possible attachments: 1) *correspond* is attached to *secretary* by means of preposition *to*; 2) *correspond* is attached to *minister* by means of preposition *of*; 3) *secretary* is attached to *minister* by means of preposition *of*. Each attachment is verified using the co-compositional information stored in the lexicon. For instance, the first attachment is verified if

only if, at least, one of the two following conditions is satisfied:
Dependent Condition: predicate $\lambda y\ iobj\_to(correspond, y)$ subcategorizes a class of nouns to which *secretary* belongs;
Head Condition: context $\lambda x\ iobj\_to(x, secretary)$ subcategorizes a class of verbs to which *correspond* belongs.
According to the lexical information illustrated in Table 1, the attachment is allowed because the Head Condition is satisfied by the verb. Note that, even if we had not learned information on the verb restrictions, the attachment would be allowed since the restrictions imposed by one of the two possible predicative structures (the nominal one) are satisfied. Following this attachment procedure, we are able to decide that *secretary* and *minister* are dependent, but not *correspond* and *minister*. An evaluation protocol is described in (2).

4 Conclusion

This paper has introduced a particular property of syntactic dependencies, namely optional co-composition, and its role in the process of disambiguation. This property allows learning two complementary semantic structures of a dependency, even if only one of them contains enough information to select a word sense or a specific syntactic attachment. The theoretical background underlying many works on NLP is often far from most recent and innovative approaches to lexical semantics, cognitive linguistics, or other linguistic areas. The main contribution of the paper is to merge different theoretical approaches (generative lexicon and cognitive grammar) in order to define a sound notion, optional co-compositionality, and describe how it can be used in different NLP applications. In sum, our aim is to use some ideas taken from current linguistic approaches to improve NLP applications.

Acknowledgement

The work by Pablo Gamallo was supported by a grant of FCT, MCT, Portugal. The work by Alexandre Agustini is supported by CAPES and PUCRS, Brazil.

References

DOWTY D.R. (1989), On the semantic content of the notion of Thematic Role, *Properties, Types, and Meaning, vol. 2*, Kluwer Academic Publisher, 69–130.

GAMALLO P., LOPES G.P., AGUSTINI A. (2003), Learning Subcategorisation Information to Model a Grammar with Co-Restrictions, *Traitement Automatic de la Langue*, Vol 44(1), 93–117.

GREFENSTETTE G (1994), *Explorations in Automatic Thesaurus Discovery*, USA, Kluwer Academic Publisher.

LIN D. (1998), Automatic Retrieval and Clustering of Similar Word, *COLING-ACL'98*.

PUSTEJOVSKY J. (1995), *The Generative Lexicon*, Cambridge, MIT Press.