Using Lexical Acquisition to Enrich a Predicate Argument Reusable Database

Paula Cristina Vaz, David Martins de Matos, Nuno J. Mamede

Spoken Language Laboratory, INESC-ID / IST
Lisboa 1000-029, Portugal
{paula.vaz,david.matos,nuno.mamede}@i2i.inesc-id.pt

Abstract
The work described in this paper aims to enrich the noun classifications of an existing database of lexical resources (de Matos and Ribeiro, 2004) adding missing information such as semantic relations. Relations are extracted from an annotated and manually corrected corpus. Semantic relations added to the database are retrieved from noun-appositive relations found in the corpus. The method uses clustering to generate labeled sets of words with hypernym relations between set label and set elements.

1. Introduction
We constantly create new words and new uses for old words. Even if we could, at some point in time, completely describe a language through some resource such as a dictionary, it would inevitably become incomplete in a matter of months (Manning and Schutze, 1999). Moreover, since building lexical resources is a time consuming and expensive task, reusing and improving existing resources is easier and more productive than building new ones.
The aim of the work described in this paper is to take an existing database of lexical resources (de Matos and Ribeiro, 2004) and enrich the noun classifications it holds by adding missing information such as semantic relations (e.g. hypernym). This new information includes noun categorization and restrictions on predicate argument selection retrieved from noun-appositive relations. A database with a rich predicate argument structure is important in natural language processing (NLP) tasks like generation, summarization, language analysis, and automatic machine translation, among others.
Section 2. talks about related work; section 3.1. explains the corpus structure; section 3.2. describes the categorizations of nouns and relations; section 4. discusses the results; and section 5. contains the conclusions.

2. Lexical acquisition using apposition
Apposition is a syntactical construction, in which two noun phrases are placed side by side, commonly used by newspaper writers to categorize subjects. Nouns in appositive are semantically related, as discussed in (Riloff and Shepherd, 1997). Two noun phrases related by apposition explain each other and help the reader to understand the nouns involved. Moreover, apposition can also be used to introduce acronyms or to complete technical definitions.
Automatic lexical acquisition is the ability that computers have to learn lexical information from machine-readable texts, and has been widely developed in recent years. Extracting methodologies, that include text mining and mapping items to a set of properties, emerged from research (Nicholson et al., 2006).
In particular, some authors use apposition to define lexical relations between nouns. This paper explores apposition as a source of automatic lexical acquisition.

Caraballo proposed a method where nouns are clustered together based on conjunction and appositive data collected from the Wall Street Journal corpus. The method uses bottom-up clustering to built a hierarchy of related nouns. The internal nodes of the resulting tree are then labeled with hypernyms for the nouns clustered underneath them (Caraballo, 1999). This work was intended to extend the WordNet (Fellbaum, 1998) with domain specific text.
Pantel and Pennacchiotti have an algorithm that automatically converts semantic relations in an ontology. The algorithm uses patterns extracted from a corpus to determine relations between nouns (Pantel and Pennacchiotti, 2006; Pennacchiotti and Pantel, 2006).
McIntosh and Curran uses noun-appositive relations to extract technical definitions from biological texts. Their work generates tables of biological compound definitions (McIntosh and Curran, 2007).

This work uses apposition to relate nouns and clustering to define classes.

3. Building semantic relations
The program, first analyzes the corpus and identifies the noun-appositive pairs. Then uses clustering to categorize relations between nouns in hypernym and synonymy.

3.1. Corpus analyzes
Floresta Sintáctica (FS) (Afonso et al., 2002) is a Portuguese language treebank created from CETEMPúblico, a collection of articles from the Portuguese daily Público. FS has 41,406 syntactic trees and about 1 million words, automatically annotated using PALAVRAS (Bick, 2000).
A subset of FS, the first 184,773 words, are manually corrected. This subset is called Bosque Sintáctico (BS). We use BS as our source of annotated noun-apposition pairs.
A corpus analysis found apposition chunks included in subject and direct object chunks. Co-occurrent nouns related by apposition could be of the type \{proper_name, common_name.indicating_title\}, \{common_name, common_name.indicating_function\}, and \{proper_name, acronym\}, as shown in sentence 1.1, 1.2, and 1.3.

Example 1. The following sentence uses apposition to indicate that “Yasser Arafat” is the “president of Palestina”:
1.1 (SUBJ Yasser Arafat, (APP presidente da Palestina)),
comeu com Bill Clinton na Casa Branca.
(SUBJ Yasser Arafat, (APP the Palestinian president)), ate with Bill Clinton at the White House.

The following sentence uses apposition to indicate that “manufacture” is a “company”:

1.2 (SUBJ O mundialmente conhecido fabricante de ratos, (APP a empresa Suíça Logitech)), encerrou o fabrico de ratos mecânicos.
(SUBJ A worldwide known mice manufacturer, (APP the Swiss Logitech company)), closed the production of mechanical mice.

The following sentence uses apposition to introduce the acronym for the United Nations:

1.3 (SUBJ A Organização das Nações Unidas, (APP ONU)), enviou tropas para África.
(SUBJ United Nations, (APP UN)), sent soldiers to Africa.

3.2. Categorizing nouns

We searched the corpus extracting every chunk containing an apposition sub-chunk. The program saves the two nouns that are heads of the noun and apposition sub-chunk in two-element sets. We use two element sets instead of pairs, because the same noun can appear in the noun or apposition sub-chunk, as shown in example 2.

Example 2. In the following sentences the noun presidente (president) is used as the appositive in sentence 2.1 and as the subject in sentence 2.2.

2.1 (SUBJ Yasser Arafat, (APP presidente da Palestina)),
comeu com Bill Clinton na Casa Branca.
(SUBJ Yasser Arafat, (APP the Palestinian president)), ate with Bill Clinton at the White House.

2.2 (SUBJ O presidente da Palestina, (APP Yasser Arafat)),
deu uma conferência de imprensa.
(SUBJ The president of the Palestine, (APP Yasser Arafat)), gave a press conference.

The noun set that implements the relation between chunk heads is saved. For instance, the sentence of example 1 produce the set \{Yasser\_Arafat, presidente (president)\} and sentence of example 2 produce the set \{presidente (president), Yasser\_Arafat\}, but only the first one is used. Note that only the word presidente (president) is included in the two element set, and not the expression presidente da Palestina (president of Palestine), because presidente (president) is the appositive chunk head.

3.2.1. Hypernyms

The system searches for two noun sets with a common elements and generates sets of related nouns. Sets with similar nouns are clustered together. The noun included in all the sets is the cluster label and the related nouns are the elements of each cluster.

Nouns like Bill Clinton, Boris Ieltsin, and Antônio Cavaco Silva occurred in sets with the noun presidente (president). These nouns originated a set labeled presidente (president), as shown in example 3.

Example 3. Noun sets:

3.1 \{presidente, Bill Clinton\}
3.2 \{Boris Ieltsin, presidente\}
3.3 \{presidente, Antônio Cavaco Silva\}

presidente = \{Bill Clinton, Boris Ieltsin, Antônio Cavaco Silva\}

Elements of the cluster are related to the cluster label by an is-a relation and the cluster label is a hypernym of cluster elements (figure 1).

Figure 1: is-a relations

Multi-word organization names, eligible for acronym use, were grouped in sets tagged after the common word, as shown in example 4. Labels hold a hypernym relation with set elements, i.e., the label is an abstraction of the set elements.

Example 4. The sets

4.1 \{Museu\_do\_Ar (Air Museum), MA \}
4.2 \{Museu\_da\_Cidade (City Museum), MC\}

generated the set

museu = \{Museu\_da\_Cidade (City Museum), Museu\_do\_Ar (Air Museum)\}

Figure 2 shows the word organização (organization) as the hypernym of concrete organization names. Each organization is also related by a is-a relation to the corresponding acronym. This way, when processing corpus, the program can refer to same entity if it either finds acronyms or complete names and assumptions made for an organization are also true for each organization in particular.

3.2.2. Synonym

Words like empresa (company) co-occurred with proper nouns (company names) and with common nouns such as fabricante (manufacturer) raising the possibility of creating synset (Fellbaum, 1998) entries. We noticed that relations between nouns are transitive: it is, thus, possible to create a network of semantic relations between the extracted nouns. Figure 3 shows the semantic relations between six nouns. Each connection represents a set extracted from the corpus and implements the relation is-a.
4. Results

We were able to extract over 486 sets of related nouns from *Bosque Sintáctico*. We compared the classification of our system with the manually annotated examples to calculate recall and precision. Recall was calculated using the number of annotations correctly marked (ch for hypernyms and cs for synonyms), and the number of annotations that were consider as hypernyms/synonyms by the system, but were manually annotated as synonyms/hypernyms (mas/mah):

\[
\text{recall}_{\text{hypernym}} = \frac{\text{ch}}{\text{ch} + \text{mas}} \quad \text{and} \quad \text{recall}_{\text{synonym}} = \frac{\text{ch}}{\text{ch} + \text{mah}}
\]

To calculate precision, we also used ch/cs, and the number of annotations that the system classified with synonyms/hypernyms but were manually annotated as hypernyms/synonyms (mah/mas).

\[
\text{precision}_{\text{hypernym}} = \frac{\text{ch}}{\text{ch} + \text{mah}} \quad \text{and} \quad \text{precision}_{\text{synonym}} = \frac{\text{ch}}{\text{ch} + \text{mas}}
\]

Table 1 shows the obtained precision, recall and F-measure.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Hypernym</th>
<th>Synonym</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recall</td>
<td>90.48%</td>
<td>43.75%</td>
</tr>
<tr>
<td>Precision</td>
<td>51.35%</td>
<td>87.50%</td>
</tr>
<tr>
<td>F</td>
<td>65.52%</td>
<td>58.33%</td>
</tr>
</tbody>
</table>

Table 1: Evaluating measures.

5. Conclusion

Automatic lexical acquisition is a hard task and lexical resources are essential for NLP tasks. We proposed a simple method to categorize nouns using a small syntactically annotated corpus. The method uses apposition and clustering to define relations between nouns.

We conclude that nouns in apposition are related by synonymy and that multi-word nouns that are not proper names, but share common words, are related by hyponymy with the common words. These results are an important step toward the objective of our research work: enrichment of the information on predicate argument structure contained in our lexical resources database. Other corpus will be explored to extract noun apposition relations.

Acknowledgments

This work was partially funded by PRIME National Project TECNOVOZ (number 03/165) and FCT project CMU-P/0005/2007.

6. References


