An Empirical Approach to Spanish Anaphora Resolution

A. Ferrández; M. Palomar
Dept. Languages and Information Systems
Alicante University - Apt. 99
03080 - Alicante - Spain
antonio@dlsi.ua.es mpalomar@dlsi.ua.es

L. Moreno
Dept. Information Systems and Computation
Valencia University of Technology
lmoreno@dsic.upv.es

Abstract
This paper documents the development of an empirically-based system implemented in Prolog that automatically resolves several kinds of anaphora in Spanish texts. These are pronominal references, surface-count anaphora, one-anaphora and elliptical zero-subject constructions (i.e. sentences that omit their pronominal subject). The resolution proposed by the authors can be based on representations resulting from either partial or full parsing. The system developed can also work on the output of a part-of-speech tagger or with different dictionaries, without changing the grammar. This grammar represents the syntactic information of each language by means of the Slot Unification Grammar formalism. This paper also documents the different kinds of information that were used for anaphora resolution in full and partial parsing, as well as evaluation results. Finally, the system has been adapted to English texts, obtaining encouraging results that prove that it can be applied with only a very few refinements to other languages as well as Spanish and English. In addition, the differences between English and Spanish anaphora are noted.

1. Introduction
Anaphora resolution is an important component in any Machine Translation (MT) system dedicated to accuracy in translation. One example that proves the importance of anaphora resolution in MT is the different set of pronouns in different languages. These differences consist of several discrepancies in gender or number (most Spanish pronouns supply more morphological information than their equivalents in English, e.g. the pronoun *they* has two equivalent pronouns in Spanish: *ellos*:masculine and *ellas*:feminine) or differences between the structure of the languages themselves (for example in Spanish sentences it is usual to omit their pronominal subject, whereas in English it is compulsory to include a subject) which requires the anaphora to be solved before making the translation.

This paper proposes an empirical approach to Spanish anaphora resolution. This resolves several kinds of anaphora such as pronouns, one-anaphora, surface-count anaphora and elliptical zero-subject constructions for Spanish texts. This empirical approach can be considered as the first computational approach to Spanish anaphora resolution.

In the following section the state-of-the-art of anaphora resolution will be summarised. This is followed by a description of the proposed

---

1 This paper has been supported by the CICYT number TIC97-0671-C02-01 / 02
computational system that includes an anaphora resolution module. The following section will show how this module can deal with all these kinds of anaphora. Finally, in the last section, the evaluation results are presented. These results have been obtained from a Telecommunication Manual (The Blue Book) and they cover the pronominal anaphora evaluation and elliptical zero subject constructions. This section will also document the results obtained in English anaphora resolution on a text of the same genre (the English version of The Blue Book), and describe the differences that have been observed between Spanish and English pronominal anaphora resolution.

2. Background in anaphora resolution

The anaphora resolution module developed by the authors is based on the integration of different kinds of knowledge, e.g. syntactic or semantic information, unlike other approaches that are based on statistics, neural networks or the principles of reasoning with uncertainty: e.g. Connoly et al. (1994) or Mitkov (1995). In these approaches the semantic and domain knowledge information is very expensive in relation to computational processing. As a consequence, current anaphora resolution implementations mainly rely on constraints\(^2\) and preference heuristics\(^3\) which employ information that originated from morphosyntactic or shallow semantic analysis, e.g. in Baldwin (1997). These approaches, however, perform notably well. Lappin and Leass (1994) described an algorithm for pronominal anaphora resolution with a high rate (85%) of correct analyses. This primarily operates on syntactic information only. Kennedy and Boguraev (1996) proposed an algorithm for anaphora resolution which is a modified and extended version of the one developed by Lappin and Leass (1994). This work does not necessarily require in-depth, full, syntactic parsing of text. The modifications enable the resolution process to work from the output of a POS tagger, enriched only with annotations of grammatical function of lexical items in the input text stream. The advantage of this approach is that anaphora resolution can be achieved within Natural Language Processing (NLP) frameworks which do not (or cannot) employ robust and reliable parsing components. Quantitative evaluation shows the anaphora resolution algorithm described here to run at a rate of 75% accuracy.

There are some other approaches that work on the output of a POS tagger, e.g. that of Mitkov and Stys (1997), in which another knowledge-poor approach to resolving pronouns in technical manuals in both English and Polish is proposed. This approach is a modification of the one reported in Mitkov (1997a). Here, the knowledge is limited to a small noun phrase grammar, a list of terms and a set of antecedent indicators (definiteness, giveness, term preference, lexical reiteration, etc.).

The anaphora resolution system developed by the present authors differs from these approaches, in that it is included in a general-purpose computational system. This is a modular system that can be applied to different applications (e.g. Machine Translation or Information Extraction). One of the advantages of this system is that it can carry out a partial or full parse of the text with the same parser and grammar. The authors adapted their anaphora resolution module in order to work with partial syntactic information. Furthermore, the module can deal with several kinds of anaphora as well as pronominal; for example one-anaphora and surface-count anaphora, and it can also resolve both Spanish and English anaphora.

3. The proposed computational system

In this section the developed computational system is described in full detail. This section consists of four subsections. The first one describes an overview of the whole system. In the following one the Slot Unification Grammar formalism is described since it is used to store the syntactic information in the system. This is followed by a description of how one can choose between partial or full parsing of the text and, finally, an example of how the system operates is shown.

\(^2\) Information that allows us to discount possible candidates for the antecedent of an anaphor.

\(^3\) Information that sorts the remaining candidates after constraints have been applied in order to choose the antecedent of the anaphor.
3.1 Overview

The proposed computational system is described graphically in Figure 1. This system represents the syntactic information by means of the Slot Unification Grammar (SUG) formalism which will be described in the following subsection. The authors have developed a translator that turns SUG rules into Prolog clauses. This translator has been run under SICStus Prolog 2.1 and Arity Prolog 5.1. It provides a Prolog program that will parse each sentence.

During parsing, the system will access the dictionary or part-of-speech tagger (POS tagger) only once in order to avoid repeated access of the same word from the dictionary. It stores the information of each word on a list before starting the parsing and it will work with this structure instead of the list of words of a DCG parser in Prolog. For example: DCG list: [this, book, is, mine], SUG list: [word (this, [adj (sing, dem), pron (sing, dem)]), word (book, [noun (...)]), ...]. Each element from the SUG list is a structure with the name word and with two arguments. The first one corresponds to the same word of the sentence in the same way as a Prolog atom. The second one corresponds to a structure list, which refers to the lexical entries of each word. That is to say that every time the parser has to access a lexical entry of a word, it will look it up in this list.

During parsing, the system will access the dictionary or part-of-speech tagger (POS tagger) only once in order to avoid repeated access of the same word from the dictionary. It stores the information of each word on a list before starting the parsing and it will work with this structure instead of the list of words of a DCG parser in Prolog. For example: DCG list: [this, book, is, mine], SUG list: [word (this, [adj (sing, dem), pron (sing, dem)]), word (book, [noun (...)]), ...]. Each element from the SUG list is a structure with the name word and with two arguments. The first one corresponds to the same word of the sentence in the same way as a Prolog atom. The second one corresponds to a structure list, which refers to the lexical entries of each word. That is to say that every time the parser has to access a lexical entry of a word, it will look it up in this list.

4 Lexical ambiguity (e.g. between this as a pronoun or as adjective) is solved by the parser. It proves each one from left to right, and chooses the first one that satisfies the grammatical rules of the constituent to parse.
The output of the parsing module is what the authors have termed *slot structure* (henceforth *SS*). This *SS* stores the lexico-syntactic, morphological, and semantic information of every constituent of the grammar. Each *SS* consists of a structure with function the name of the constituent (*np*, *vp*, ...). Its first argument corresponds to another structure *conc* which includes all the arguments of the constituent (**Number, Gender, SemanticType**). The second one corresponds to the variable (*Ap*) of the final logical formula of the constituent. It should be noted that this *Ap* is also used to link an anaphor with its referring antecedent. The remaining arguments correspond to the *SS* of its sub-constituents. In this *SS* the parser leaves Prolog variables uninstantiated5 and these represent the slots corresponding to the optional constituents that do not appear in the sentence. In this way, it becomes evident, which constituents have not been parsed. From now on, each *SS* will be shown with *Ap* and *conc* only if it is necessary for the sake of simplicity. In **Figure 3** an example of a slot structure can be seen with its final logical formula that uses the Prolog variable stored in each *SS*.

---

5 In this paper, Prolog variables uninstantiated are represented as the symbol: “_”.

This *SS* is the input of the following module in which the authors deal with anaphora resolution as well as other NLP problems such as extrapolation, ellipsis or PP-attachment. As has been previously mentioned, this *SS* stores all the necessary information for this resolution. This module also receives a list of candidate antecedents from previous sentences in order to solve intersentential anaphora. This list consists of the *SS* of the different noun phrases that have been parsed in previous sentences. It is similar to the list of discourse representation structures described in the Discourse Representation Theory (Kamp 1981, Guenthner 1987), since each *SS* stores a discourse marker (the Prolog variable used for the logical formula and for linking an anaphor with its referring antecedent) and morphological information. So if the sentence *It bit Peter* appears after the sentence in **Figure 3** (*A dog barked*), the pronoun *it* would be linked with its referring antecedent by means of sharing the same discourse marker. The final result can be seen in **Figure 4**.

---

**Figure 4.** An example of anaphora resolution.

The output of this module is a *SS* where all the anaphors have been resolved. This *SS* is then used in the last module of the system, in which the final logical formula of the sentence is obtained. An example of the whole system can be seen in section 3.4.

It should be emphasised that this computational system is based on the separation of the different modules of a typical NLP system: lexical treatment, syntactic information, parsing, resolution of NLP problems and semantic analysis. This fact allows one to produce modular NLP systems in which grammatical rules, logical
formulas and the module of resolution of NLP problems are quite independent of each other. In this way, one can modify one module without affecting the remaining modules of the system. For example, one can change either the grammatical formalism, the dictionary or the semantic notation without affecting the anaphora resolution module. The general architecture of this system is sequential but it allows interaction between different NLP problems, and between different modules.

3.2 Slot Unification Grammar (SUG)

In this section the SUG formalism is briefly described. The SUG is a logical formalism based on unification, which is an extension of Definite Clause Grammars (DCG). It is called Slot Unification Grammar due to the slot structures in which it represents information (see Figure 3 and Figure 4). SUG has been developed with the aim of extending DCG in order to facilitate the resolution of Natural Language Processing (NLP) problems in a modular way. This formalism has been previously applied to anaphora resolution in Ferrández et al. (1997b).

The authors used SUG instead of other well known formalisms such as Head Driven Phrase Structure Grammar (HPSG, Pollard and Sag 1995), Lexical Functional Grammar (LFG, Bresnan 1982) or Slot Grammars (SG, McCord 1990), because SUG allows a modular and computational treatment of NLP problems, and it facilitates its integration with any POS tagger or dictionary.

SUG can be defined as the quadruple: \((NT,T,P,H)\), where \(NT\) and \(T\) are a finite set of nonterminal and terminal symbols respectively; moreover \(NT \cap T = \emptyset\). \(P\) is a finite set of pairs \(\alpha++>\beta\) where \(\alpha\in NT\), \(\beta\in (T\cup NT)\cup\{\text{procedures calls}\}\), and these pairs are called production rules. Finally \(H\) is a set of production rules which only contains the first member of the production rule, i.e. \(\alpha\), and \(\alpha\)'s name is either coordinated, juxtaposition, fusion, basicWord or isWord.

SUG’s production rules add to those of DCG such that each sub-constituent of \(\beta\) could be omitted in the sentence if it is noted between the optional operator: \(<<\text{ constituent }>>\). It is a well-known fact that optional constituents in DCG can be obtained by using a nonterminal symbol (e.g. \(\text{optA}\), with \(\text{optA} \rightarrow A\) and \(\text{optA} \rightarrow \{\}\)). However this skill necessitates the addition of new nonterminal symbols, whereas SUG allows us to achieve this without adding any new ones. The reduction of grammatical rules in SUG can be seen in Figure 5.

Furthermore, this optional operator has the possibility of reminding one whether the optional constituent has been parsed in the sentence or not. This information can be of use in the resolution of NLP problems such as ellipsis or extraposition. It is carried out by adding a label to the optional constituent, e.g. \(<<\text{ SSNP } : np >>\). This label will be an uninstantiated Prolog variable if constituent \(np\) is missing, in order to make Prolog predicate \(\text{var (SSNP)}\) successful.

3.3 Partial parsing with SUG

One of the main advantages of this system is that it allows one to carry out either partial or full parsing of the text. One can choose between these two kinds of parsing by selecting the appropriate
initial symbol of the SUG grammar, as is explained in this section.

The importance of partial parsing is noted in Abney (1997). It is considered better to carry out partial parsing on unrestricted text rather than complete parsing, because of possible errors and the unavoidable lack of completion of lexicon and grammar. It is also difficult to perform a global search efficiently with unrestricted text, due to the length of sentences and the ambiguity of grammars. Partial parsing is considered to be a response to these difficulties. Partial parsing techniques aim to recover syntactic information efficiently and reliably from unrestricted text. They do this however, at the cost of sacrificing completeness and depth of analysis.

Figure 6. SUG partial grammar.

The initial symbol that carries out a full parsing of the text in any grammar is usually: sentence ++> np, vp. In the case of carrying out a partial parsing, one only has to use the new initial symbol sentencePP, whose grammatical rule is shown in Figure 6. This symbol consists of a sequence of coordinated prepositional phrases (pp), coordinated noun phrases (np), pronouns (p), conjunctions (conj) and verbs (verb) in whatever order they appear in the text. The slot structure (SS) returned by the parser consists of a sequence of these constituents: pp, np, p, conj, verb and free words. The attachments (e.g. of the pp) are postponed to the module of resolution of NLP problems, which could work jointly with the algorithm for anaphora resolution (in a similar way to the approach proposed in Azzam 1995). The free words consist of constituents that are not covered by this partial parsing (e.g. adverbs). It is important to note that this is a flexible way of establishing a partial parsing of the text, since one can easily determine the set of constituents to parse, by including them in the grammatical rule sentencePP. In Figure 6, only constituents considered necessary for anaphora resolution are parsed.

3.4 An example

The system developed by the authors solves anaphora by means of the slot structure (SS) returned after parsing the sentence. Later in this paper several examples of anaphora resolution will be shown. Figure 7 shows an example of how the whole system operates. In this example, the sentence Jane saw a dog yesterday is parsed. Firstly, the sentence is translated into the previously mentioned SUG list with all the information obtained from the dictionary or from the tagger. In the case of accessing the dictionary, all the lexical entries of each word could be stored by means of the SUG fact isWord. Otherwise, if the sentence is tagged, the process described in Figure 2 will be applied. In either case, the SUG list with all the necessary information for the following modules of the system will be obtained.

Figure 7
to choose the appropriate initial symbol of the grammar: *sentencePP* (partial parsing shown in Figure 6), or *sentence ++> np, vp* (full parsing). The output of the parsing module would be a SS that stores all the necessary information for anaphora resolution. This SS is used to obtain the logical formula (obviously, in the case of partial parsing, we can only obtain partial logical formulas). The variables $X$ and $Y$ used in the logical formula are obtained from each SS.

4. The anaphora resolution module

In this section the anaphora resolution module of the system is described in full detail. This section consists of three subsections. In the first one the proposed algorithm for anaphora resolution is shown. In the second, the detection of elliptical zero-subject constructions in Spanish is described. Finally, some evaluation results of this module are presented.

4.1 The algorithm

What follows is a description of an algorithm that deals with discourse anaphora using partial or full parsing. It is based on the process described in Figure 1. This algorithm is applied after the parsing of a sentence and before obtaining its logical formula.

This algorithm deals with pronominal references, surface-count anaphora and one-anaphora. It uses a slot structure (SS) corresponding to the output of the parsing module as well as a list of candidates. This list consists of the slot structures of all the previously parsed noun phrases. For each case of anaphora in this SS, several constraints and preferences are applied. The output of this algorithm consists of an SS, where each anaphor has been resolved.

The detection of anaphors and possible antecedents is easily carried out by means of the information stored in each SS, i.e. its name and arity. For example, antecedents have an SS with *np* as their name, the pronouns have *pron*, and one-anaphora has *np* with this structure in English: *determiner + adjective + “one” (the red one)*, and in Spanish they are noun phrases in which the noun has been omitted: *el rojo*. One-anaphora in Spanish could have the following SUG rule:

$$ np ++> <\text{determiner}>, \text{adjective}, <pp>.$$  

And the following SS:

$$ np (\text{conc}(\ldots), X, \text{determiner}(\ldots), \text{adjective}(\ldots), pp(\ldots))$$

One-anaphora and surface-count anaphora in English do not usually bear gender information whereas in Spanish they do, e.g. *the red one* can be translated into both *el rojo* (masculine) and *la roja* (feminine); or *the second one* can be translated into Spanish for both *el segundo* (masculine) and *la segunda* (feminine).

The number of previous sentences considered in the resolution of an anaphor are determined by the kind of anaphor itself. For pronominal references the antecedents in the same sentence or in the two previous sentences are considered, unlike one-anaphora which have more lexical information. The antecedents in the same or previous paragraph are taken into consideration. The number of sentences is known because this information is stored jointly with the SS of every antecedent: a different Prolog variable is assigned for each sentence and all the antecedents in this sentence will have this variable in the list of antecedents.

The algorithm applies a set of constraints to the list of possible antecedents in order to discount candidates. If there is only one candidate, this one will be the antecedent of the anaphor. Otherwise, if more than one candidate is left remaining, a set of preferences will be applied. These preferences sort the list of remaining candidates, and the first of these will be the selected antecedent. There are other approaches that do not use constraints since they consider all types of information as preferences. Mitkov (1997b), compared these two different approaches: one with constraints and preferences, and another one with only preferences. His results show that preferences may be more reliable since no antecedent is discarded, but on the other hand, constraints could make the procedure faster and more accurate. These results are confirmed in the authors’ experiments (see section 4.3) where if constraints are not used, then the accuracy rate is reduced significantly. These constraints and preferences are described in more detail in Ferránández et al (1998a) for full parsing, and in
Ferrández et al (1998b) for partial parsing. A brief description of these constraints (morphosyntactic agreement, C-command and semantic consistency) and preferences (surface structure, lexical information and syntactic parallelism) is shown:

a) Morphosyntactic agreement (person, gender and number) are checked by unification of the structure conc. For example, in the sentence:

*John saw Jane at the party. She was very happy.*

There are two possible candidates whose SS are:

* np (conc (sing, masc), X, John), np (conc (sing, fem), X, Jane)*

 Whereas the SS of the pronoun:

* pron (conc (sing, fem), X, she)*

In order to decide between both antecedents, the unification of both conc structures is carried out.

b) C-command and Minimal Governing Category constraints as formulated in Reinhart (1983). They are applied to the syntactic information stored in the SS of each constituent.

For example, in the sentence:

*John went to the party although Peter didn’t see him*

If one were to run a full parsing of the text, one would obtain the SS shown in Figure 8, where *him* cannot corefer with *Peter* because it would require that a non-reflexive pronoun be interpreted as non-coreferential with any c-commanding NP in its Minimal Governing Category\(^9\) (MGC) (in this case \(SENT_3\)), whereas \(NP_1\) (*John*) is not included in this MGC and the pronoun does not c-command it.

However, if one were to run a partial parsing one would also have enough information to carry out these constraints. One only has to apply the following heuristic: whenever one finds a free conjunction\(^10\) it is considered that one has entered into a new clause, and the pronouns in the new clause can corefer with the antecedents in the previous sentence. However, they cannot corefer with antecedents in the new clause. In the example shown in Figure 8, one would obtain the following sequence of constituents:

* np (John), v (went), pp (to the party), conj (although), np (Peter), v (didn’t saw), pron (him)*

From this it can be stated that *him* appears in the second clause (there is the free conjunction *although*), therefore it cannot corefer with *Peter* although it can corefer with *John*.

In the sentence:

*John and him bought a book*

One can also state that *him* can not corefer with *John* with partial parsing, since one would obtain the following sequence of constituents where there is no free conjunction:

* np (John and him), v (bought), np (a book)*

However in:

*The woman who looks at the present that she bought…*

The pronoun *she* can corefer with the *woman* although no free conjunction was found. However, in this example, one does

---

\(^9\) The minimal governing category of a node A is the minimal (lowest) *Sentence* or *NP* dominating A.

\(^{10}\) The term *free conjunctions*, refers to those conjunctions that do not join coordinated prepositional or noun phrases.
In Figure 10 the anaphora resolution process for the sentence The monkey ate the banana because it was hungry is described. As shown in this figure, there are two possible antecedents that agree in number and gender with the anaphor it. The third argument of the conc structures of these SS is the semantic type, according to the IRSAS ontology described in Figure 9. Consequently the monkey has the semantic type of nonhuman, whereas the banana has vegetal. The pronoun gets its semantic type from the verb and its modifiers, and in this particular case it is animal. Due to this semantic type, the monkey is preferred to the banana, and it is unified with the variable Z. However, if we apply this algorithm to the sentence The monkey ate the banana because it was ripe, then the banana will be chosen as antecedent, since the pronoun obtains the semantic type of vegetal from the verb.

d) Surface structure of the sentence for the surface-count anaphora (the former, the latter, ...).

The SS returned by the SUG fact coordinated is used. This fact allows the coordination of constituents with the same or different form: Peter, your daughter and she and it will allow us to access any coordinated constituent in the desired order (see Figure 11). That is to say, its SS will tell us the number of coordinated constituents. In this way, the anaphor: the first one will choose the second coordinated noun phrase, in this case Peter.

---

11 This NP is one of the constituents that is fully parsed (see Figure 6).
Figure 11. Information used in resolution of surface-count anaphora.

e) Lexical information.
This is carried out by means of counting the number of times that the same candidate occurs in the text.

f) Finally, syntactic information stored in the SS is returned by the parser: syntactic parallelism, i.e. preference for those antecedents in the same position with reference to the verb.

Moreover, in order to solve one-anaphora, the following preference is applied: the antecedents with a similar syntactic structure are chosen. For example, the sentence in Figure 12:

Wendy didn’t give either boy a green tie-dyed T-shirt, but she gave Sue a blue one.

<table>
<thead>
<tr>
<th>List of antecedents:</th>
</tr>
</thead>
<tbody>
<tr>
<td>np (properNoun(Wendy))</td>
</tr>
<tr>
<td>np (X, det (a), adj ([green, tie-dyed]), noun (T-shirt))</td>
</tr>
</tbody>
</table>

SS of the anaphor:
np (Y, det (a), adj ([blue]), pron (one)).

Final SS of the anaphor:
np (Y, det (a), adj ([blue, tie-dyed]), noun (T-shirt))

List of antecedents after solving the anaphor:
np (X, det (a), adj ([green, tie-dyed]), noun (T-shirt))
np (Y, det (a), adj ([blue, tie-dyed]), noun (T-shirt))

Since they have different variables in their SS (discourse markers), it means that they represent different objects in discourse

Figure 12. Resolution of one-anaphora in: “Wendy didn’t give either boy a green tie-dyed T-shirt, but she gave Sue a blue one.”

The antecedent a green tie-dyed T-shirt is chosen instead of Wendy or Sue because they have similar SS (a determiner, a common noun and an adjective):

np (noun(Wendy)), np (X, det (a), adj ([green, tie-dyed]), noun (T-shirt))

And the SS of the one-anaphora:
np (Y, det (a), adj ([blue]), pron (one))

This SS allows decomposition of the description (i.e. green can be broken off) and the solution of the anaphora will be:

np (Y, det (a), adj ([blue, tie-dyed]), noun (T-shirt))

It is important to note that the solution will have a different variable13 (Y) than its antecedent (X). It means the anaphor and its antecedent do not co-refer, and the anaphor refers to a new entity in the discourse. However in:

John bought a dark red apple and a green pear. He ate the red one.

The anaphor will co-refer with a dark red apple, and they will share the same variable since they refer to the same entity in the discourse14. The distinction between both cases is made because in the second one the anaphor and its antecedent share the same modifiers15 (red) and they agree in number.

In relation to partial parsing and one-anaphora, it does not represent a drawback since we parse full noun phrases (see Figure 6), and in this way all the necessary information described above is disposed of.

---

12 This list of adjectives is provided by the SUG fact juxtaposition.

13 This variable corresponds to the λp of the final logical formula of the constituent.

14 In Figure 4 an example of correferance can be seen.

15 In the future we intend to include more semantic information in order to solve these anaphors.
4.2 The adaptation of this algorithm to elliptical zero-subject constructions in Spanish

It is usual in Spanish sentences to omit their pronominal subject. In these cases we obtain the person and gender information from the verb of the sentence, and once we obtain the appropriate pronoun, we apply the same pronoun resolution algorithm that we have described in the previous subsection.

We can check the omission of the pronominal subject of a sentence by means of the SS of the sentence as shown in Figure 13. From this, it is known that the subject of the sentence has been omitted due to the Prolog variable that we find (‘_’). Given that the typical SUG rule for a sentence in Spanish is:

\[ \text{sent} \ (\text{Person}, \ \text{Number}, \ \text{Gender}) \ \rightarrow \ \langle \langle \ \text{np} \ (\text{Person}, \ \text{Number}, \ \text{Gender}) \ \rangle, \ \text{vp} \ (\text{Person}, \ \text{Number}, \ \text{Gender}) \ \rangle. \]

The subject is optional. When it is omitted in the sentence, the SS stores a Prolog variable in the slot corresponding to this noun phrase. One can obtain the information corresponding to the subject from the verb of the sentence. In this figure, it would be third person, singular, masculine or feminine and it has the semantic feature of animal. With these omitted pronominal anaphors one applies the preference for the subject of the previous sentence (if it agrees in person and number, and if it is semantically consistent). This information is used to find its antecedent, in this case Peter.

\[ \text{SS of the sentences:} \]
\[ \text{sent}(\text{np}(“\text{Pedro”}), \text{vp}(“\text{compró un regalo”)}) \]
\[ \text{sent}(\_ \_ \_, \text{vp}(“\text{Luego se fue a casa”)}) \]

\[ \text{SS of the verb of the second sentence:} \]
\[ \text{v(conc(past,thirdPerson,sing,animal,inorganic)}, X, “fue”) \]

\[ \text{SS of the omitted anaphor:} \]
\[ \text{pron(conc(\_, thirdPerson, sing, animal)}, Z \_ “he or she”) \]

Figure 13

It is necessary to check if the verb is either impersonal or imperative, since in Spanish it would not have a subject. For example, in: \textit{Llueve (it is raining)}, since the verb \textit{llueve (rain)} is impersonal, we do not have to search for its antecedent, or in:

\textit{Haz un nudo en el pañuelo (tie a knot in your handkerchief)}

Where the verb haz \textit{(tie)}\(^{17}\) is imperative. In these cases, it is essential not to parse the subject of these verbs by means of this SUG rule:

\[ \text{sent} \ \rightarrow \ \langle \langle \ \text{SUBJ:np} \ \rangle, \ \text{vp} \ (\text{Tense}), (((\text{Tense == \text{imperative}}; \text{Tense == \text{impersonal}}), \text{var(SUBJ)})) ; \text{true})) \]

The feature imperative or impersonal is stored in the SS of the verb haz \textit{(tie)}:

\[ \text{v(conc(imperative)}, X, “haz”) \]

It also shows that it is not necessary to search for any pronoun or subject.

Sometimes gender information of the pronoun can be obtained when the verb is copulative. For example, in:

\textit{Pedro, vio a Ana, en el parque. Ø Estaba muy guapa} (Peter, saw Ann, in the park. She, was very beautiful)

In this example, the verb \textit{estaba (was)} is copulative, so its subject has to agree in gender and number with its object. In this way, one can obtain the gender information from the object, guapa \textit{(beautiful woman)}, that has feminine gender, so the omitted pronoun would be \textit{she} instead of \textit{he}.

4.3 Evaluation of the anaphora resolution module

This section shows some evaluation results of the anaphora resolution module, both in pronominal anaphora and elliptical zero-subject constructions. The last point of this section explains how the authors have adapted the system to English texts.

4.3.1 Pronominal anaphora evaluation

As in Ferrández et. al (1998b), the authors ran the system on part of the Spanish version of The Blue

\[ ^{17}\]In this case, we are comparing both verbs, haz and tie, and whilst we accept that the translation is not faithful, the sense of the imperative is implicit.
Book corpus (98,206 words, 3,089 sentences with an average of 31.8 words per sentence). This corpus contains the International Telecommunications Union CCITT handbook in English, French and Spanish versions. This corpus is one of the most important collections of telecommunications texts and contains 5M words, automatically tagged by the Xerox tagger.

 Semantic information was not used since the tagger does not provide this information, but despite this, the following figures were obtained. The system detected 100% of the pronominal anaphors\(^\text{18}\), and the partial parsing described in Figure 6 parsed 81% of words with a very simple grammar\(^\text{19}\). The average length of sentences is 32 words. For pronominal references an 83% accuracy rate was obtained with 100 third-person pronouns. These are demonstrative or personal pronouns, and they are classified into three categories: 53 complement pronouns\(^\text{20}\) (la, lo, le, ...), 26 pronouns inside a prepositional phrase\(^\text{21}\) and 21 pronouns that are not inside a prepositional phrase (él, ella, ...). The success rate for each class of pronoun was: 85%, 85% and 76% respectively. Another interesting fact is that the system found 2,210 possible candidate antecedents for all the pronouns. They were reduced to 906 candidates after applying constraints (morphological agreement and c-command). This fact means that these constraints have eliminated 59% of the possible candidates, leaving 22 possible candidates on average for each antecedent before restrictions, and 9 after restrictions. The correct antecedent was obtained for all these pronouns by only applying constraints in 4 pronouns. It was necessary to apply preferences in order to resolve the remaining pronouns. All these antecedents are located in the same sentence as the anaphor or in the two previous sentences, and all of them correspond to noun phrases. For example, in sentence (2) (page 13), there are eight possible antecedents in the same sentence as the anaphor them:

\[
\text{np}_1 (\text{noise}), \text{np}_2 (\text{the power of noise}), \text{np}_3 (\text{analogue transmission}), \text{np}_4 (\text{the systems of analogue transmission}), \text{np}_5 (\text{the circuit}), \text{np}_6 (\text{noise from the circuit}), \text{np}_7 (\text{the main source of noise from the circuit}), \text{np}_8 (\text{long or medium connections}).
\]

These constraints are very important since they considerably reduce the number of candidates to consider. With reference to c-command constraints, in all the sentences that were parsed, the heuristic that the authors applied with partial parsing worked correctly 100% of the time.

In order to measure the effect of c-command and morphological agreement, the authors ran the experiment implementing only one of these constraints. If only morphological agreement is considered, the number of candidates remaining after this constraint was reduced to 993 (55% reduction). If one only considers c-command they are reduced to 2062 (7% reduction). Hence morphological agreement plays a very important role for Spanish texts, since if one does not apply this constraint the success rate is reduced to 58% because there are more possible candidates for each anaphor before applying preferences. If only morphological agreement is used, a 73% success rate is achieved.

This 83% accuracy is similar to the 75% obtained by Kennedy and Boguraev (1996) although it is difficult to compare both measures since the authors worked on different texts and different languages. With reference to the work of Mitkov (1998) in which he obtains a success rate of 89.7% in English technical manuals, (a better rate than that achieved by the present authors - 83% for Spanish texts, and 87% for English texts as is shown in section 4.3.3), it should be pointed out that he used information that is very close to the genre of the text: e.g. the antecedent indicator section heading preference (if a NP occurs in the heading of the section, part of which is the current sentence, then we consider it as the preferred candidate). In this study, this information was not used, so the authors consider their approach to be more easily adapted to other kinds of texts.

\(^{18}\) It should be noted that identifying anaphoric pronouns in English is not as simple as in Spanish because of the many cases of pleonastic pronouns (e.g. non-anaphoric it).

\(^{19}\) This percentage could easily be improved by adding more constituents to the grammar (e.g. adverbs or punctuation marks).

\(^{20}\) One can see an example of this kind of pronoun in the sentence: Peter lo vio (Peter saw it).

\(^{21}\) An example: John fue por él (John went for him).
The reason why some of the references have failed is due to the lack of semantic information. One can see an example in the sentence (1) where our system has selected the antecedent *el usuario de un auricular telefónico* (the user of a phone receiver) instead of *un auricular telefónico* (a phone receiver), since only a phone receiver can produce a rise of the acoustic pressure (and not the user).

(1) Conjunto de perturbaciones pasajeras o permanentes del funcionamiento del oído, o del sistema nervioso, que puede sufrir el usuario de un auricular telefónico, como consecuencia de una brusca e importante elevación de la presión acústica producida por éste.

A set of passing or permanent disruptions of the operation of the ear, or of the nervous system, that the user of a phone receiver can suffer as a consequence of a sudden and important rise of the acoustic pressure produced by it.

As an example of an anaphor correctly solved by this system, one can see the sentence (2) where the antecedent *los sistemas de transmisión analógica* (the systems of analogue transmission) is chosen for the pronoun *them*.

(2) En las conexiones largas o de longitud media, es probable que la fuente principal de ruido de circuito estriben en los sistemas de transmisión analógica, ya que en ellos la potencia de ruido suele ser proporcional a la longitud del circuito.

In long or medium connections, it is probable that the main source of noise from the circuit relies on the systems of analogue transmission, since the power of noise in them is usually proportional to the length of the circuit.

### 4.3.2 Elliptical zero-subject evaluation

The system was also run on part of the *Lexesp* corpus22 (2,123 words, 99 sentences and an average number of 21 words per sentence). This corpus contains Spanish texts from different genres. These texts are mainly obtained from newspapers and they are also tagged automatically as in *The Blue Book*. The partial parsing described in Figure 6 was run on these texts, and the system parsed 83% with the same grammar that was used for *The Blue Book*. It parsed 471 noun phrases and 209 prepositional phrases. The authors ran the proposed algorithm on these texts to detect the elliptical zero-subject constructions in Spanish. The system detected 181 verbs where 75% of these verbs have the subject omitted. The success rate of the algorithm was found to be 97%, where failures did occur, this was mainly due to the lack of semantic information. For example, in the sentence (3), the system parsed the following sequence of constituents: pp (prep (in), np (the naked branches, the sparrows)), v (looked), np (fatter), producing an incorrect coordination of the noun phrases *branches* and *the sparrows* (the subject of the verb *looked*) due to the lack of semantic information. Since the system didn’t detect a noun phrase before the verb, it is returned as a verb that has its subject omitted.

(3) …en la desnudez de las ramas, los gorriones parecían más gordos…

… in the naked branches, the sparrows looked fatter…

Another reason for some of the failures was found to be the result of some mistakes in the tagging of the sentence. This occurs in sentence (4) for example, where the personal pronoun *I* is tagged as a complement personal pronoun (*me*) that cannot be the subject of a verb. This system concludes that this verb does not have an explicit subject.

(4) Yo quería conocer aquello, …

I wanted to know that, …

The verbs detected by our system to have an omitted subject can be classified into the following categories: 13% of first person pronouns, 11% of second person pronouns, 19% that have the subject moved to a position after the verb (cataphora), and 45% in which the subject can be substituted by a third person pronoun, whose antecedent is in the previous sentences (anaphora). Since semantic information is not available, one cannot detect whether the noun phrase after the verb is its direct object or its subject, but from this 19% one can detect that it is the subject 20% of the time since there are at least two noun phrases after the verb.

---

22 This corpus belongs to the project of the same name carried out by the Departamento de Psicología of the University of Oviedo and developed by the Grupo de Lingüística Computacional of the Universitat de Barcelona, with the collaboration of the Grupo de Tratamiento del Lenguaje of the Universitat Politècnica of Catalunya.
4.3.3 Pronominal anaphora evaluation for English texts

The approach described for Spanish texts was adapted in order to apply it to English. There were two main changes made to the Spanish system. The first one is the new SUG for English in which the grammatical rules of the constituents were written and which were partially parsed (np and pp). It appears that this new grammar is formed by different sets of terminal and non-terminal symbols, and given that the English Xerox tager also has different tags, the second change has been the interface between the grammar and the tagger (see Figure 2).

The system was also run on the English version of The Blue Book, in order to make a comparison with previous results. This can be considered as a text of the same genre. Eighty-one pronouns were dealt with (it:41, they:31, themselves:9) with an accuracy of 87%. In this text there are on average: 22 words per sentence, 11 candidates per pronoun before restrictions and 5.5 candidates after restrictions.

With reference to the differences between English and Spanish pronoun resolution, a greater number of possible antecedents were observed for Spanish pronouns (22) than for English (11). This fact could be due to the larger overall size of Spanish sentences.

Another difference is that the constraints (c-command and morphological agreement) played a more important role for Spanish texts in the detection of the antecedent: the total number of possible antecedents is reduced from 2,210 to 906 (a reduction of 59%), whereas for English texts it has only a reduction of 49%. This is mainly due to the fact that the Spanish language has more morphological information than English24, although this information also has an important role in English (it has reduced the number of possible candidates by 39%).

In spite of the fact that Spanish pronouns have more morphological information than English pronouns, the authors obtained a better accuracy for the English (87%) than for the Spanish (83%) version of the same text. This fact is mainly due to Spanish sentences being longer than English sentences. Hence there are more possible antecedents for each pronoun in Spanish (9) than in English (5.5). Another reason is because the structure of Spanish sentences is more flexible than in English. For example, the direct object can usually occupy the position of the subject, and this can only be known by means of semantic information: e.g. La trama de supervisión la utiliza ... (direct object: la trama, pronoun: la, verb: utiliza). This fact means that the position of the antecedent with reference to the verb is less important for Spanish texts, and Spanish syntactic parallelism is less important than in English texts. Moreover, there are some special cases of Spanish complement pronouns (la, le, ...) in which c-command constraints do not work. These pronouns are those that function as a complement of the verb, e.g. A Peter le gusta Ana, where the pronoun (le) has its antecedent (Peter) in the same sentence and clause. These cases are explained by a simple rule: “when the indirect/direct object is moved from its typical place, then the pronoun refers to this object”. However, this fact makes it more difficult to solve these pronouns, since semantic information is needed to detect when the direct object has been moved from its original place after the verb.

5. Conclusions

This paper proposes a computational system that can deal with intersentential anaphora resolution both in English and Spanish texts. The different kinds of anaphora that it deals with are pronominal references, surface-count anaphora, one-anaphora and elliptical zero-subject constructions.

This system can work on the output of a POS tagger, on which one can automatically apply a partial parsing from the Slot Unification Grammar formalism. In order to evaluate this anaphora resolution module in unrestricted texts, only

---

23 Pleonastic pronouns it (i.e. non-anaphoric it) have not been included in these results.

24 For example the English pronoun it does not have morphological information, but its equivalents in Spanish do (este: masculine, esta: femenine). The same is true of they (ellos: masc., ellas: fem) but not of he and she (él / ella).
lexical, morphological and partial syntactic information were used. The authors obtained an accuracy of 83% for pronominal references, 97% in detection of zero subject pronouns, and 87% accuracy for English pronouns on the English version of the same corpus (The Blue Book).

As a future aim the authors will include semantic information in their algorithm in order to check the improvement obtained by it. This information will be stored in a dictionary (e.g. WordNet) which could be automatically consulted (since the tagger does not provide this information).

6. References


Ferrández A., Palomar M. and Moreno L. (1997b) Slot Unificacion Grammar and anaphora resolution. Recent Advances in Natural Language Processing. RANLP. Tzigov Chark (Bulgaria)


Mitkov, R. (1997b) Factors in anaphora resolution: they are not the only things that matter. A case study based on two different approaches. In Proceedings of ACL/ EACL workshop on Operational factors in practical, robust anaphora resolution. Madrid (Spain)

Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, COLING - ACL. Montreal (Canada)


