Improving Prepositional Phrase Attachment Disambiguation Using the Web as Corpus*

Hiram Calvo [1] and Alexander Gelbukh [1,2]

[1] Center for Computing Research, National Polytechnic Institute,
Av. Juan de Dios Bátiz s/n, esq. Av. Mendizábal, México, D.F., 07738. México
hcalvo@sagitario.cic.ipn.mx, gelbukh@cic.ipn.mx; www.gelbukh.com

[2] Department of Computer Science and Engineering, Chung-Ang University,
221 Huksuk-Dong, DongJak-Ku, Seoul, 156-756, Korea

Abstract. The problem of Prepositional Phrase (PP) attachment disambiguation consists in determining if a PP is part of a noun phrase, as in *He sees the room with books*, or an argument of a verb, as in *He fills the room with books.* Volk has proposed two variants of a method that queries an Internet search engine to find the most probable attachment variant. In this paper we apply the latest variant of Volk's method to Spanish with several differences that allow us to attain a better performance close to that of statistical methods using treebanks.

1 Introduction

In many languages, prepositional phrases (PP) such as *in the garden* can be attached to noun phrases (NP): *the grasshopper in the garden*, or verb phrases (VP): *plays in the garden.* Sometimes there are several possible variants for attachment of a given PP. For example, in *The police accused the man of robbery* we can consider two possibilities:

(1) *The police* [*accused* [*the man of robbery*]]
(2) *The police* [*accused* [*the man*] *of robbery*]

In the case (1) the object of the verb is *the man of robbery*, and in (2) the object is *the man*, and the accusation is *of robbery*. An English speaker knows that the second option is the correct one, whereas for a computer we need a method to automatically determine which option is correct.

There are several methods to find the correct PP attachment place that are based on treebank statistics. These methods have been reported to achieve up to 84.5% accuracy [1], [2], [3], [4], [5], [6]. However, resources such as treebanks are not available for many languages and they are difficult to port, so that a less resource-demanding

* Work done under partial support of Mexican Government (CONACyT, SNI), IPN (PIFI, CGEPI), RITOS-2, and Korean Government (KIPA Professorship for Visiting Faculty Positions in Korea). The second author is currently on Sabbatical leave at Chung-Ang University.

H. Calvo, A. Gelbukh. Improving Prepositional Phrase Attachment Disambiguation Using the Web as Corpus. In: A. Sanfeliu, J. Ruiz-Shulcloper (Eds.). Progress in Pattern Recognition, Speech and Image Analysis (CIARP 2003). Lecture Notes in Computer Science, N 2905, Springer-Verlag, 2003, pp. 604–610.

method is desirable. Ratnaparkhi [7] describes a method that requires only a part-of-speech tagger and morphological information. His method uses raw text to be trained.

The quality of the training corpus significatively determines the correctness of the results. Specially, to reduce the effects of noise in a corpus and to consider most of the phenomena, a very large corpus is desirable. Eric Brill [8] shows that it is possible to achieve state-of-the-art accuracy with relatively simple methods whose power comes from the plethora of texts available to such systems. His paper also gives examples of several NLP applications that benefit from the use of very large corpora.

Nowadays, large corpora comprise more than 100 million words, whereas the Web can be seen as the largest corpus with more than one billion documents. Particularly for Spanish, Bolshakov and Galicia-Haro [9] report approximately 12,400,000 pages that can be found through Google. We can consider the Web as a corpus that is big and diverse enough to obtain better results with statistical methods for NLP.

Using the Web as corpus is a recently growing trend; an overview of the existing research that tries to harness the potential of the web for NLP can be found in [10]. In particular, for the problem of finding the correct PP attachment, Volk [11], [12] proposes variants of a method that queries an Internet search engine to find the most probable PP attachment.

In this paper we show the results of applying the latest variant of Volk's method with several differences to Spanish. In Section 2 we explain the variants of Volk's method. In Section 3 we present the differences of the method we use with regard to his method. In Section 4 we explain the details of our experiment and the results we obtained, and finally we draw the conclusions.

2 Volk's Method

Volk proposes two variants of a method to decide the attachment of a PP to a NP or a verb. In this Section we explain both variants and their results.

2.1 First Variant

Volk [11] proposes disambiguating PP attachments using the web as corpus by considering the co-occurrence frequencies (*freq*) of verb + preposition against those of noun + preposition. The formula used to calculate the co-occurrence is:

```
cooc(X,P) = freq(X,P) / freq (X)
```

where X can be either a noun or a verb. For example, for *He fills the room with books,* N = *room*, P = *with*, and V = *fill*. The value of `cooc(X,P)` is between 0 (no co-occurrences found) and 1 (the words always occur together)

The value of `freq (X,P)` is calculated by querying the AltaVista search engine using the `NEAR` operator: `freq(X,P) = query("X NEAR P")`.

To choose an attachment variant, `cooc(N+P)` and `cooc(V+P)` are calculated, and the variant with the higher value is chosen. If some of the `cooc` values are lower than a

Table 1. Coverage and Accuracy for Volk's 2000 algorithm

threshold	coverage	accuracy
0.1	99%	68%
0.3	36.7%	75%
0.5	7.7%	82%

minimum co-ocurrence threshold, the attachment cannot be desambiguated, and thus it is not covered. By adjusting the *minimum co-occurrence threshold*, Volk's 2000 algorithm can attain very good coverage but poor accuracy, or good accuracy with low coverage. Table 1 shows the coverage / accuracy values for Volk's experiments.

Volk [11] also concludes that using full forms is better than using lemmas.

The same experiment has been done for Dutch by Vandeghinste [13], reaching for a coverage of 100% an accuracy of 58.4%. To obtain an accuracy of 75%, Vandeghinste used a threshold of 0.606, yielding the coverage of only 21.6%.

2.2 Second Variant

In a subsequent paper [12], Volk uses a different formula to calculate co-occurrences. Now the head noun of the PP is included within the queries. The formula used is:

cooc(X,P, N_2) = freq(X, P, N_2) / freq(X)

where freq (X,P,N_2) is calculated by querying the AltaVista search engine using the NEAR operator: freq(X,P,N_2) = query("X NEAR P NEAR N_2"). X can be N_1 or V. For example, for *He fills the room with books,* N_1 = *room*, P = *with*, N_2 = *books* and V = *fill.*

Volk experiments first by requiring that both cooc(N_1,P,N_2) and cooc(V,P,N_2) can be calculated to determine a result. Then, he considers using a threshold to determine the PP attachment when one of cooc(N_1,P,N_2) or cooc(V,P,N_2) is not known. That is, if cooc(N_1,P,N_2) is not known, cooc(V,P,N_2) must be higher than the threshold to decide that the PP is attached to the verb, and *vice versa.* Afterwards, by including both lemmas and full forms in queries, Volk attains a better performance, and by defaulting to noun attachment for previously uncovered attachments, he attains the coverage of 100%. The results he found are shown as Table 2.

Table 2. Results of Volk's 2001 Method

coverage	accuracy	requires both cooc(N_1,P,N_2) and cooc(V,P,N_2)?	threshold when cooc(N_1,P,N_2) or cooc(V,P,N_2) is not known	includes both lemmas and full forms in queries?	defaults to noun attachment for uncovered attachments?
55%	74.32%	yes			
63%	75.04%		0.001		
71%	75.59%		0.001	yes	
85%	74.23%		0	yes	
100%	73.08%		0	yes	yes

For Dutch, requiring both cooc(N_1,P,N_2) and cooc(V,P,N_2), Vandeghinste achieves a coverage of 50.2% with an accuracy of 68.92. Using a threshold and including both lemmas and full forms in queries, he reaches 27% coverage for an accuracy of 75%. For 100% coverage, defaulting the previously uncovered cases to noun attachments, an accuracy of 73.08% is obtained.

3 Improving Performance

Methods to resolve PP attachment ambiguity based on treebank statistics achieve by far a better performance than the experiments described above. Nonethless, we think that there are several elements that could be changed to improve methods based on Web queries. One of the elements to consider is the size of the document database of search engines. Indeed, this is relevant for finding representative co-occurrence frequencies for certain language. It is known that not every search engine yields the same results. For example, Table 3 shows the number of co-occurrences found from different search engines for the same words:

Table 3. Number of co-ocurrences found in several search engines

	leer en el metro	*read in the subway*
Google	104	30
All-the-Web	56	23
Altavista	34	16
Teoma	15	19

Google is ranked as search engine with the largest database size by the search engine showdown.[1] Because of its greater document database size, we have determined that using Google to obtain word co-occurrence frequencies can yield to better results.

Another element to consider is the use of the NEAR operator. We decided do not using it the since it does not guarantee that the query words appear in the same sentence. Let us consider the following queries from AltaVista:

(1) wash NEAR with NEAR door 6,395 results
(2) wash NEAR with NEAR bleach 6,252 results

(1) yields 6,395 pages found, even when books are unrelated to the wash operation. Compared to (2) that yields 6,252 pages found, we can see that there is no clear distinction of when is a preposition + noun related to a verb. On the other hand, using an exact phrase search yields 0, which marks out a clear distinction between *wash with door* and *wash with bleach*. The numbers of the pages found are as follows:

Exact phrase search	AltaVista	Google
"*wash with door*"	0	0
"*wash with bleach*"	100	202

[1] Information taken from www.searchengineshowdown.com, update of December 31st, 2002.

Table 4. Queries to determine the PP attachment of
Spanish *Veo al gato con un telescopio* and English *I see the cat with a telescope*

Veo al gato con un telescopio	hits	I see the cat with a telescope	hits
ver	296,000	see	194,000,000
"ver con telescopio"	8	"see with telescope"	13
"ver con telescopios"	32	“see with telescopes”	76
"ver con un telescopio"	49	"see with a telescope"	403
"ver con el telescopio"	23	"see with the telescope"	148
"ver con unos telescopios"	0	“see with some telescopes”	0
"ver con los telescopios"	7	“see with the telescopes”	14
veo	642,000		
"veo con telescopio"	0		
"veo con telescopios"	0		
"veo con un telescopio"	0	*(no such forms in English)*	
"veo con unos telescopios"	0		
"veo con el telescopio"	1		
"veo con los telescopios"	0		
freq(veo,con,telescopio) =	**1.279×10^{-4}**	**freq(see,with,telescope) =**	**3.371×10^{-6}**
gato	185,000	cat	24,100,000
"gato con telescopio"	0	"cat with telescope"	0
"gato con telescopios"	0	“cat with telescopes”	0
"gato con un telescopio"	3	“cat with a telescope"	9
"gato con unos telescopios”	0	“cat with some telescopes”	0
"gato con el telescopio"	6	"cat with the telescope"	2
"gato con los telescopios"	0	“cat with the telescopes”	0
freq(gato,con,telescopio) =	**0.486×10^{-4}**	**freq(cat,with,telescope) =**	**0.456×10^{-6}**

Following [12], we use jointly full forms and lemmatized forms of nouns and verbs to obtain better performance. However, as we are not using the `NEAR` operator, we must consider the determiners that can be placed between the noun or verb and the preposition. Also we consider that the nucleus of the PP might appear in plural, without affecting its use. To illustrate this, consider the following sentence[2]:

Veo al gato con un telescopio ‘I see the cat with a telescope”

The attachments are calculated by the queries shown in Table 4. Since `freq(veo,con,telescopio) > freq(gato,con,telescopio)`, the attachment is disambiguated as *veo con telescopio* ‘see with telescope’.

4 Experiment and Results

For our evaluation we extracted randomly 100 sentences from the LEXESP corpus of Spanish [15] and the newspaper *Milenio Diario*[3]. All searches were restricted to Spanish pages.

[2] Example borrowed from [14].

[3] www.milenio.com

First, we considered not restricting queries to a specific language, given that a benefit could be obtained from similar words across languages, such as French and Spanish. For example, the phrase *responsables de la debacle* ‘responsibles of the rout’ is used in both languages varying only in its accentuation (*débâcle* in French, *debacle* in Spanish). As Google does not take into account word accentuation, results for both languages are returned by the same query. However, with an unrestricted search, Google returns different count-ups in its API[4] and in its GUI.[5] For example, for *ver* ‘to see’, its GUI shows 270,000 results, whereas its API returns more than 20,000,000, even enabling the “group similar results” filter. This enormous deviation can be reduced by restricting language to a specific language. For Spanish, a restricted search for *ver* ‘to see’ in the GUI returns 258,000 results, whereas in the API it returns 296,000. Currently we are not aware of the reason for this difference; in any case it does not have any serious impact on our experiments.

The sentences of our experiment bear 181 cases of preposition attachment ambiguity. From those, 162 could be automatically resolved. They were verified manually and to determine that 149 of them were resolved correctly and 13 were incorrect.

In terms of coverage and accuracy used by Volk, we obtain the coverage of 89.5% with an accuracy of 91.97%. Without considering coverage, the overall percentage of attachment ambiguities resolved correctly is 82.3%.

5 Conclusions

We have found an increase in performance using Volk’s method with the following differences:

- using exact phrase searches instead of `NEAR` operator;
- using a search engine with a larger document database;
- searching combinations of words that include definite and indefinite articles; and
- searching for singular and plural forms of words when possible.

The results obtained with this method (89.5% coverage, 91.97% accuracy, 82.3% overall) are very close to those obtained by using treebank statistics, without the need of such expensive resources.

A demo version of a program implementing our method can be found at the website `likufanele.com/ppattach`.

References

[1] Adwait Ratnaparkhi, Jeff Reynar, and Salim Roukos. A Maximum Entropy Model for Prepositional Phrase Attachment. In *Proceedings of the Human Language Technology Workshop*, pp 250-255, Plainsboro, N.J. ARPA, 1994.

[4] Google API is a web service that uses the SOAP and WSDL standards to allow a program to query directly the Google search engine. More information can be found at api.google.com.

[5] www.google.com

[2] Eric Brill and Phil Resnik. A Rule Based Approach to Prepositional Phrase Attachment Disambiguation. In *Proceedings of the Fifteenth International Conference on Computational Linguistics* (COLING), 1994.

[3] Michael Collins and James Brooks. Prepositional Phrase Attachment trhough a Backed-of Model. In David Yarouwsky and Kenneth Church, eds, *Proceedings of the Third Workshop on Very Large Corpora*, pages 27-38, Cambridge, Massachussets, June 1995.

[4] Paola Merlo, Matthew W. Crocker, and Cathy Berthouzoz. Attaching Multiple Prepositional Phrases: Generalized Backer-off Estimation. In Claire Cardie and Ralph Weischedel, editors, *Second Conference on Empirical Methods in Natural Language Processing*, pp 149-155, Providence, R.I., August 1-2, 1997.

[5] Jakub Zavrel and Walter Daelemans. Memory-Based Leraning: Using Similarity for Smoothing. In ACL, 1997.

[6] Alexander Franz. Independence Assumptions Considered Harmful. In ACL, 1997.

[7] Adwait Ratnaparkhi. Statistical Models for Unsupervised Prepositional Phrase Attachment, In *Proceedings of the 36th ACL and 17th COLING*, pp. 1079-1085, 1998.

[8] Eric Brill. Processing Natural Language without Natural Language Processing, In Alexander Gelbukh, ed. *CICLing 2003, Computational Linguistics and Intelligent Text Processing*, Lecture Notes in Computer Science 2588:360–369, Springer-Verlag, 2003.

[9] Igor A. Bolshakov and Sofia N. Galicia-Haro. Can We Correctly Estimate the Total Number of Pages in Google for a Specific Language?, In Alexander Gelbukh, ed. *CICLing 2003, Computational Linguistics and Intelligent Text Processing,* Lecture Notes in Computer Science 2588:415–419, Springer-Verlag, 2003.

[10] Frank Keller and Mirella Lapata, Using the Web to Obtain Frequencies for Unseen Bigrams. To appear in *Computational Linguistics* 29:3, 2003.

[11] Martin Volk. Scaling up. Using the WWW to resolve PP attachment ambiguities. In *Proceedings of Konvens* 2000*,* Ilmenau, October 2000.

[12] Martin Volk: Exploiting the WWW as a corpus to resolve PP attachment ambiguities. In *Proceeding of Corpus Linguistics* 2001. Lancaster: 2001.

[13] Vincent Vandeghinste. Resolving PP Attachment Ambiguities Using the WWW. In the *Thirteenth meeting of Computational Linguistics in the Netherlands*, CLIN 2002 Abstracts, Groningen, 2002.

[14] Sofia Galicia Haro, Alexander Gelbukh, and Igor A. Bolshakov. Una aproximación para resolución de ambigüedad estructural empleando tres mecanismos diferentes. In *Procesamiento de Lenguaje Natural*, No 27, September 2001. Sociedad Española para el Procesamiento de Lenguaje Natural (SEPLN), Spain, pp. 55-64.

[15] Fernando Cuetos, Miguel Angel Martí, and Valiña Carreiras, *Léxico informatizado del Español*. Edicions de la Universitat de Barcelona, 2000.