Statistical Models for Unsupervised Prepositional Phrase Attachment

Adwait Ratnaparkhi
Dept. of Computer and Information Science
University of Pennsylvania
200 South 33rd Street
Philadelphia, PA 19104-6389
adwait@unagi.cis.upenn.edu

Abstract

We present several unsupervised statistical models for the prepositional phrase attachment task that approach the accuracy of the best supervised methods for this task. Our unsupervised approach uses a heuristic based on attachment proximity and trains from raw text that is annotated with only part-of-speech tags and morphological base forms, as opposed to attachment information. It is therefore less resource-intensive and more portable than previous corpus-based algorithm proposed for this task. We present results for prepositional phrase attachment in both English and Spanish.

1 Introduction

Prepositional phrase attachment is the task of deciding, for a given preposition in a sentence, the attachment site that corresponds to the interpretation of the sentence. For example, the task in the following examples is to decide whether the preposition *with* modifies the preceding noun phrase (with head word *shirt*) or the preceding verb phrase (with head word *bought* or *washed*).

1. I bought the shirt with pockets.
2. I washed the shirt with soap.

In sentence 1, *with* modifies the noun *shirt*, since *with pockets* describes the *shirt*. However in sentence 2, *with* modifies the verb *washed* since *with soap* describes how the shirt is *washed*. While this form of attachment ambiguity is usually easy for people to resolve, a computer requires detailed knowledge about words (e.g., *washed* vs. *bought*) in order to successfully resolve such ambiguities and predict the correct interpretation.

2 Previous Work

Most of the previous successful approaches to this problem have been statistical or corpus-based, and they consider only prepositions whose attachment is ambiguous between a preceding noun phrase and verb phrase. Previous work has framed the problem as a classification task, in which the goal is to predict N or V, corresponding to noun or verb attachment, given the head verb v, the head noun n, the preposition p, and optionally, the object of the preposition $n2$. For example, the $(v, n, p, n2)$ tuples corresponding to the example sentences are

1. bought shirt with pockets
2. washed shirt with soap

The correct classifications of tuples 1 and 2 are N and V, respectively.

(Hindle and Rooth, 1993) describes a partially supervised approach in which the FIDDITCH partial parser was used to extract (v, n, p) tuples from raw text, where p is a preposition whose attachment is ambiguous between the head verb v and the head noun n. The extracted tuples are then used to construct a classifier, which resolves unseen ambiguities at around 80% accuracy. Later work, such as (Ratnaparkhi et al., 1994; Brill and Resnik, 1994; Collins and Brooks, 1995; Merlo et al., 1997; Zavrel and Daelemans, 1997; Franz, 1997), trains and tests on quintuples of the form $(v, n, p, n2, a)$ extracted from the Penn treebank(Marcus et al., 1994), and has gradually improved on this accuracy with other kinds of statistical learning methods, yielding up to 84.5% accuracy(Collins and Brooks, 1995). Recently, (Stetina and Nagao, 1997) have reported 88% accuracy by using a corpus-based model in conjunction with a semantic dictionary.

While previous corpus-based methods are highly accurate for this task, they are difficult to port to other languages because they require resources that are expensive to construct or simply nonexistent in other languages. We present an unsupervised algorithm for prepositional phrase attachment in English that requires only an part-of-speech tagger and a morphology database, and is therefore less resource-intensive and more portable than previous approaches, which have all required either treebanks or partial parsers.

3 Unsupervised Prepositional Phrase Attachment

The exact task of our algorithm will be to construct a classifier cl which maps an instance of an ambiguous prepositional phrase $(v, n, p, n2)$ to either N or V, corresponding to noun attachment or verb attachment, respectively. In the full natural language parsing task, there are more than just two potential attachment sites, but we limit our task to choosing between a verb v and a noun n so that we may compare with previous supervised attempts on this problem. While we will be given the candidate attachment sites during testing, the training procedure assumes no a priori information about potential attachment sites.

3.1 Generating Training Data From Raw Text

We generate training data from raw text by using a part-of-speech tagger, a simple chunker, an extraction heuristic, and a morphology database. The order in which these tools are applied to raw text is shown in Table 1. The tagger from (Ratnaparkhi, 1996) first annotates sentences of raw text with a sequence of part-of-speech tags. The chunker, implemented with two small regular expressions, then replaces simple noun phrases and quantifier phrases with their head words. The extraction heuristic then finds head word tuples and their likely attachments from the tagged and chunked text. The heuristic relies on the observed fact that in English and in languages with similar word order, the attachment site of a preposition is usually located only a few words to the left of the preposition. Finally, numbers are replaced by a single token, the text is converted to lower case, and the morphology database is used to find the base forms of the verbs and nouns.

The extracted head word tuples differ from the training data used in previous supervised attempts in an important way. In the supervised case, both of the potential sites, namely the verb v and the noun n are known before the attachment is resolved. In the unsupervised case discussed here, the extraction heuristic only finds what it thinks are *unambiguous* cases of prepositional phrase attachment. Therefore, there is only one possible attachment site for the preposition, and either the verb v or the noun n does not exist, in the case of noun-attached preposition or a verb-attached preposition, respectively. This extraction heuristic loosely resembles a step in the bootstrapping procedure used to get training data for the classifier of (Hindle and Rooth, 1993). In that step, unambiguous attachments from the FIDDITCH parser's output are initially used to resolve some of the ambiguous attachments, and the resolved cases are iteratively used to disambiguate the remaining unresolved cases. Our procedure differs critically from (Hindle and Rooth, 1993) in that we do not iterate, we extract unambiguous attachments from unparsed input sentences, and we totally ignore the ambiguous cases. It is the hypothesis of this approach that the information in *just the unambiguous* attachment events can resolve the *ambiguous* attachment events of the test data.

3.1.1 Heuristic Extraction of Unambiguous Cases

Given a tagged and chunked sentence, the extraction heuristic returns head word tuples of the form $(v, p, n2)$ or $(n, p, n2)$, where v is the verb, n is the noun, p is the preposition, $n2$ is the object of the preposition. The main idea of the extraction heuristic is that an attachment site of a preposition is usually within a few words to the left of the preposition. We extract :

$(v, p, n2)$ if

- p is a preposition ($p \neq of$)
- v is the first verb that occurs within K words to the left of p
- v is not a form of the verb *to be*
- No noun occurs between v and p

Tool	Output
Raw Text	The professional conduct of lawyers in other jurisdictions is guided by American Bar Association rules or by state bar ethics codes , none of which permit non-lawyers to be partners in law firms .
$\Downarrow$	
POS Tagger	The/DT professional/JJ conduct/NN of/IN lawyers/NNS in/IN other/JJ jurisdictions/NNS is/VBZ guided/VBN by/IN American/NNP Bar/NNP Association/NNP rules/NNS or/CC by/IN state/NN bar/NN ethics/NNS codes/NNS ,/, none/NN of/IN which/WDT permit/VBP non-lawyers/NNS to/TO be/VB partners/NNS in/IN law/NN firms/NNS ./.
$\Downarrow$	
Chunker	conduct/NN of/IN lawyers/NNS in/IN jurisdictions/NNS is/VBZ guided/VBN by/IN rules/NNS or/CC by/IN codes/NNS ,/, none/NN of/IN which/WDT permit/VBP non-lawyers/NNS to/TO be/VB partners/NNS in/IN firms/NNS ./.
$\Downarrow$	
Extraction Heuristic	(n =lawyers, p =in, $n2$ =jurisdictions) (v =guided, p =by, $n2$ =rules)
$\Downarrow$	
Morphology	(n =lawyer, p =in, $n2$ =jurisdiction) (v =guide, p =by, $n2$ =rule)

Table 1: How to obtain training data from raw text

- $n2$ is the first noun that occurs within K words to the right of p
- No verb occurs between p and $n2$

$(n, p, n2)$ if

- p is a preposition $(p \neq of)$
- n is the first noun that occurs within K words to the left of p
- No verb occurs within K words to the left of p
- $n2$ is the first noun that occurs within K words to the right of p
- No verb occurs between p and $n2$

Table 1 also shows the result of the applying the extraction heuristic to a sample sentence.

The heuristic ignores cases where $p = of$, since such cases are rarely ambiguous, and we opt to model them deterministically as noun attachments. We will report accuracies (in Section 5) on both cases where $p = of$ and where $p \neq of$. Also, the heuristic excludes examples with the verb *to be* from the training set (but not the test set) since we found them to be unreliable sources of evidence.

3.2 Accuracy of Extraction Heuristic

Applying the extraction heuristic to 970K unannotated sentences from the 1988 Wall St. Journal[1] data yields approximately 910K unique head word tuples of the form $(v, p, n2)$ or $(n, p, n2)$. The extraction heuristic is far from perfect; when applied to and compared with the annotated Wall St. Journal data of the Penn treebank, only 69% of the extracted head word tuples represent correct attachments.[2] The extracted tuples are meant to be a noisy but *abundant* substitute for the information that one might get from a treebank. Tables 2 and 3 list the most frequent extracted head word tuples for unambiguous verb and noun attachments, respectively. Many of the frequent noun-attached $(n, p, n2)$ tuples, such as *num to num*,[3] are incorrect. The prepositional phrase *to num* is usually attached to a verb such as *rise* or *fall* in the Wall St. Journal domain, e.g., *Profits rose 46 % to 52 million.*

[1] This data is available from the Linguistic Data Consortium, `http://www.ldc.upenn.edu`

[2] This accuracy also excludes cases where $p = of$.

[3] Recall the *num* is the token for quantifier phrases identified by the chunker, like *5 million*, or *6 %*.

Frequency	Verb	Prep	Noun2
8110	close	at	num
1926	reach	for	comment
1539	rise	to	num
1438	compare	with	num
1072	fall	to	num
970	account	for	num
887	value	at	million
839	say	in	interview
680	compare	with	million
673	price	at	num

Table 2: Most frequent $(v, p, n2)$ tuples

Frequency	Noun	Prep	Noun2
1983	num	to	num
923	num	from	num
853	share	from	million
723	trading	on	exchange
721	num	in	num
560	num	to	month
519	share	on	revenue
461	num	to	day
417	trading	on	yesterday
376	share	on	sale

Table 3: Most frequent $(n, p, n2)$ tuples

4 Statistical Models

While the extracted tuples of the form $(n, p, n2)$ and $(v, p, n2)$ represent unambiguous noun and verb attachments in which *either* the verb or noun is known, our eventual goal is to resolve ambiguous attachments in the test data of the form $(v, n, p, n2)$, in which *both* the noun n and verb v are always known. We therefore must use any information in the unambiguous cases to resolve the ambiguous cases. A natural way is to use a classifier that compares the probability of each outcome:

$$cl(v, n, p, n2) = \begin{cases} N & \text{if } p = of \\ \arg\max_{a \in \{N,V\}} Pr(v, n, p, a) & \text{otherwise} \end{cases} \tag{1}$$

We do not currently use $n2$ in the probability model, and we omit it from further discussion.

We can factor $Pr(v, n, p, a)$ as follows:

$$\begin{aligned} Pr(v, n, p, a) &= Pr(v)Pr(n) \\ &\quad Pr(a|v, n) \\ &\quad Pr(p|a, v, n) \end{aligned}$$

The terms $Pr(n)$ and $Pr(v)$ are independent of the attachment a and need not be computed in cl (1), but the estimation of $Pr(a|v, n)$ and $Pr(p|a, v, n)$ is problematic since our training data, i.e., the head words extracted from raw text, occur with either n or v, but never both n, v. This leads to make some heuristically motivated approximations. Let the random variable ϕ range over $\{true, false\}$, and let it denote the presence or absence of any preposition that is unambiguously attached to the noun or verb in question. Then $p(\phi = true|n)$ is the conditional probability that a particular noun n in free text has an unambiguous prepositional phrase attachment. ($\phi = true$ will be written simply as $true$.) We approximate $Pr(a|v, n)$ as follows:

$$\begin{aligned} Pr(a = N|v, n) &\approx \frac{Pr(true|n)}{Z(v, n)} \\ Pr(a = V|v, n) &\approx \frac{Pr(true|v)}{Z(v, n)} \\ Z(v, n) &= Pr(true|n) + Pr(true|v) \end{aligned}$$

The rationale behind this approximation is that the tendency of a v, n pair towards a noun (verb) attachment is related to the tendency of the noun (verb) alone to occur with an unambiguous prepositional phrase. The $Z(v, n)$ term exists only to make the approximation a well formed probability over $a \in \{N, V\}$.

We approximate $Pr(p|a, v, n)$ as follows:

$$\begin{aligned} Pr(p|a = N, v, n) &\approx Pr(p|true, n) \\ Pr(p|a = V, v, n) &\approx Pr(p|true, v) \end{aligned}$$

The rationale behind these approximations is that when generating p given a noun (verb) attachment, only the counts involving the noun (verb) are relevant, assuming also that the noun (verb) has an attached prepositional phrase, i.e., $\phi = true$.

We use word statistics from both the tagged corpus and the set of extracted head word tuples to estimate the probability of generating $\phi = true$, p, and $n2$. Counts from the extracted set of tuples assume that $\phi = true$, while counts from the corpus itself may correspond to either $\phi = true$ or $\phi = false$, depending on if the noun or verb in question is, or is not, respectively, unambiguously attached to a preposition.

4.1 Generate ϕ

The quantities $Pr(true|n)$ and $Pr(true|v)$ denote the conditional probability that n or v will occur with some unambiguously attached preposition, and are estimated as follows:

$$Pr(true|n) = \begin{cases} \frac{c(n,true)}{c(n)} & c(n) > 0 \\ .5 & \text{otherwise} \end{cases}$$
$$Pr(true|v) = \begin{cases} \frac{c(v,true)}{c(v)} & c(v) > 0 \\ .5 & \text{otherwise} \end{cases}$$

where $c(n)$ and $c(v)$ are counts from the tagged corpus, and where $c(n, true)$ and $c(v, true)$ are counts from the extracted head word tuples.

4.2 Generate p

The terms $Pr(p|n, true)$ and $Pr(p|v, true)$ denote the conditional probability that a particular preposition p will occur as an unambiguous attachment to n or v. We present two techniques to estimate this probability, one based on bigram counts and another based on an interpolation method.

4.2.1 Bigram Counts

This technique uses the bigram counts of the extracted head word tuples, and backs off to the uniform distribution when the denominator is zero.

$$Pr(p|true, n) = \begin{cases} \frac{c(n,p,true)}{c(n,true)} & c(n, true) > 0 \\ \frac{1}{|\mathcal{P}|} & \text{otherwise} \end{cases}$$
$$Pr(p|true, v) = \begin{cases} \frac{c(v,p,true)}{c(v,true)} & c(v, true) > 0 \\ \frac{1}{|\mathcal{P}|} & \text{otherwise} \end{cases}$$

where $\mathcal{P}$ is the set of possible prepositions, where all the counts $c(\ldots)$ are from the extracted head word tuples.

4.2.2 Interpolation

This technique is similar to the one in (Hindle and Rooth, 1993), and interpolates between the tendencies of the (v, p) and (n, p) bigrams and the tendency of the type of attachment (e.g., N or V) towards a particular preposition p. First, define $c_N(p) = \sum_n c(n, p, true)$ as the number of noun attached tuples with the preposition p, and define $c_N = \sum_p c_N(p)$ as the number of noun attached tuples. Analogously, define $c_V(p) = \sum_v c(v, p, true)$ and $c_V = \sum_p c_V(p)$. The counts $c(n, p, true)$ and $c(v, p, true)$ are from the extracted head word tuples. Using the above notation, we can interpolate as follows:

$$Pr(p|true, n) = \frac{c(n, p, true) + \frac{c_N(p)}{c_N}}{c(n, true) + 1}$$
$$Pr(p|true, v) = \frac{c(v, p, true) + \frac{c_V(p)}{c_V}}{c(v, true) + 1}$$

5 Evaluation in English

Approximately 970K unannotated sentences from the 1988 Wall St. Journal were processed in a manner identical to the example sentence in Table 1. The result was approximately 910,000 head word tuples of the form $(v, p, n2)$ or $(n, p, n2)$. Note that while the head word tuples represent correct attachments only 69% of the time, their quantity is about 45 times greater than the quantity of data used in previous supervised approaches. The extracted data was used as training material for the three classifiers cl_{base}, cl_{interp}, and cl_{bigram}. Each classifier is constructed as follows:

cl_{base} This is the "baseline" classifier that predicts N of $p = of$, and V otherwise.

cl_{interp}: This classifier has the form of equation (1), uses the method in section 4.1 to generate ϕ, and the method in section 4.2.2 to generate p.

cl_{bigram}: This classifier has the form of equation (1), uses the method in section 4.1 to generate ϕ, and the method in section 4.2.1 to generate p.

Table 4 shows accuracies of the classifiers on the test set of (Ratnaparkhi et al., 1994), which is derived from the manually annotated attachments in the Penn Treebank Wall St. Journal data. The Penn Treebank is drawn from the 1989 Wall St. Journal data, so there is no possibility of overlap with our training data. Furthermore, the extraction heuristic was developed and tuned on a "development set", i.e., a set of annotated examples that did not overlap with either the test set or the training set.

Subset	Number of Events	cl_{bigram}	cl_{interp}	cl_{base}
$p = of$	925	917	917	917
$p \neq of$	2172	1620	1618	1263
Total	3097	2537	2535	2180
Accuracy	-	81.91%	81.85%	70.39%

Table 4: Accuracy of mostly unsupervised classifiers on English Wall St. Journal data

Attachment	$Pr(a\|v,n)$	$Pr(p\|a,v,n)$
Noun($a = N$)	.02	.24
Verb($a = V$)	.30	.44

Table 5: The key probabilities for the ambiguous example *rise num to num*

Table 5 shows the two probabilities $Pr(a|v,n)$ and $Pr(p|a,v,n)$, using the same approximations as cl_{bigram}, for the ambiguous example *rise num to num*. (Recall that $Pr(v)$ and $Pr(n)$ are not needed.) While the tuple (num, to, num) is more frequent than (rise, to, num), the conditional probabilities prefer $a = V$, which is the choice that maximizes $Pr(v,n,p,a)$.

Both classifiers cl_{interp} and cl_{bigram} clearly outperform the baseline, but the classifier cl_{interp} does not outperform cl_{bigram}, even though it interpolates between the less specific evidence (the preposition counts) and more specific evidence (the bigram counts). This may be due to the errors in our extracted training data; supervised classifiers that train from clean data typically benefit greatly by combining less specific evidence with more specific evidence.

Despite the errors in the training data, the performance of the unsupervised classifiers (81.9%) begins to approach the best performance of the comparable supervised classifiers (84.5%). (Our goal is to replicate the supervision of a treebank, but not a semantic dictionary, so we do not compare against (Stetina and Nagao, 1997).) Furthermore, we do not use the second noun $n2$, whereas the best supervised methods use this information. Our result shows that the information in imperfect but abundant data from unambiguous attachments, as shown in Tables 2 and 3, is sufficient to resolve ambiguous prepositional phrase attachments at accuracies just under the supervised state-of-the-art accuracy.

6 Evaluation in Spanish

We claim that our approach is portable to languages with similar word order, and we support this claim by demonstrating our approach on the Spanish language. We used the Spanish tagger and morphological analyzer developed at the Xerox Research Centre Europe[4] and we modified the extraction heuristic to account for the new tagset, and to account for the Spanish equivalents of the words *of* (i.e., *de* or *del*) and *to be* (i.e., *ser*). Chunking was not performed on the Spanish data. We used 450k sentences of raw text from the Linguistic Data Consortium's Spanish News Text Collection to extract a training set, and we used a non-overlapping set of 50k sentences from the collection to create test sets. Three native Spanish speakers were asked to extract and annotate ambiguous instances of Spanish prepositional phrase attachments. They annotated two sets (using the full sentence context); one set consisted of all ambiguous prepositional phrase attachments of the form $(v, n, p, n2)$, and the other set consisted of cases where $p = con$. For testing our classifier, we used only those judgments on which all three annotators agreed.

6.1 Performance

The performance of the classifiers cl_{bigram}, cl_{interp}, and cl_{base}, when trained and tested on Spanish language data, are shown in Table 6. The Spanish test set has fewer ambiguous prepositions than the English test set, as shown by the accuracy of cl_{base}. However, the accuracy improvements of cl_{bigram} over cl_{base} are statistically significant for both test sets.[5]

[4] These were supplied by Dr. Lauri Kartunnen during his visit to Penn.

[5] Using proportions of changed cases, $P = 0.0258$ for the first set, and $P = 0.0108$ for the set where $p = con$

Test Set	Subset	Number of Events	cl_{bigram}	cl_{interp}	cl_{base}
All p	$p = de\|\|del$	156	154	154	154
	$p \neq de\|\|del$	116	103	97	91
	Total	272	257	251	245
	Accuracy	-	94.5%	92.3%	90.1%
$p = con$	**Total**	192	166	160	151
	Accuracy	-	86.4%	83.3%	78.6%

Table 6: Accuracy of mostly unsupervised classifiers on Spanish News Data

7 Conclusion

The unsupervised algorithm for prepositional phrase attachment presented here is the only algorithm in the published literature that can significantly outperform the baseline without using data derived from a treebank or parser. The accuracy of our technique approaches the accuracy of the best supervised methods, and does so with only a tiny fraction of the supervision. Since only a small part of the extraction heuristic is specific to English, and since part-of-speech taggers and morphology databases are widely available in other languages, our approach is far more portable than previous approaches for this problem. We successfully demonstrated the portability of our approach by applying it to the prepositional phrase attachment task in the Spanish language.

8 Acknowledgments

We thank Dr. Lauri Kartunnen for lending us the Spanish natural language tools, and Mike Collins for helpful discussions on this work.

References

ACL. 1997. *Proceedings of the 35th Annual Meeting of the ACL, and 8th Conference of the EACL*, Madrid, Spain, July.

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

Michael Collins and James Brooks. 1995. Prepositional Phrase Attachment through a Backed-off Model. In David Yarowsky and Kenneth Church, editors, *Proceedings of the Third Workshop on Very Large Corpora*, pages 27–38, Cambridge, Massachusetts, June.

Alexander Franz. 1997. Independence Assumptions Considered Harmful. In ACL (ACL, 1997).

Donald Hindle and Mats Rooth. 1993. Structural Ambiguity and Lexical Relations. *Computational Linguistics*, 19(1):103–120.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1994. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

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

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

Adwait Ratnaparkhi. 1996. A Maximum Entropy Part of Speech Tagger. In Eric Brill and Kenneth Church, editors, *Conference on Empirical Methods in Natural Language Processing*, University of Pennsylvania, May 17–18.

Jiri Stetina and Makoto Nagao. 1997. Corpus Based PP Attachment Ambiguity Resolution with a Semantic Dictionary. In Jou Zhou and Kenneth Church, editors, *Proceedings of the Fifth Workshop on Very Large Corpora*, pages 66–80, Beijing and Hong Kong, Aug. 18 – 20.

Jakub Zavrel and Walter Daelemans. 1997. Memory-Based Learning: Using Similarity for Smoothing. In ACL (ACL, 1997).