Attaching Multiple Prepositional Phrases: Generalized Backed-off Estimation

Paola Merlo
IRCS-U. of Pennsylvania
LATL-University of Geneva
2 rue de Candolle
1204 Geneva, Switzerland
merlo@latl.unige.ch

Matthew W. Crocker
Centre for Cognitive Science
University of Edinburgh
2 Buccleuch Place,
Edinburgh, UK EH8 9LW
mwc@cogsci.ed.ac.uk

Cathy Berthouzoz
LATL
University of Geneva
2 rue de Candolle
1204 Geneva, Switzerland
berthouzoz@latl.unige.ch

Abstract

There has recently been considerable interest in the use of lexically-based statistical techniques to resolve prepositional phrase attachments. To our knowledge, however, these investigations have only considered the problem of attaching the first PP, i.e., in a [V NP PP] configuration. In this paper, we consider one technique which has been successfully applied to this problem, backed-off estimation, and demonstrate how it can be extended to deal with the problem of multiple PP attachment. The multiple PP attachment introduces two related problems: sparser data (since multiple PPs are naturally rarer), and greater syntactic ambiguity (more attachment configurations which must be distinguished). We present and algorithm which solves this problem through re-use of the relatively rich data obtained from first PP training, in resolving subsequent PP attachments.

1 Introduction

Ambiguity is the most specific feature of natural languages, which sets them aside from programming languages, and which is at the root of the difficulty of the parsing enterprise, pervading languages at all levels: lexical, morphological, syntactic, semantic and pragmatic. Unless clever techniques are developed to deal with ambiguity, the number of possible parses for an average sentence (20 words) is simply intractable. In the case of prepositional phrases, the expansion of the number of possible analysis is the Catalan number series, thus the number of possible analyses grows with a function that is exponential in the number of Prepositional Phrase (Church and Patil, 1982). One of the most interesting topics of debate at the moment, is the use of frequency information for automatic syntactic disambiguation. As argued in many pieces of work in the AI tradition (Marcus, 1980; Crain and Steedman, 1985; Altmann and Steedman, 1988; Hirst, 1987), the exact solution of the disambiguation problem requires complex reasoning and high level syntactic and semantic knowledge. However, current work in part-of-speech tagging has succeeded in showing that it is possible to carve one particular subproblem and solve it *by approximation* — using statistical techniques — independently of the other levels of computation.

In this paper we consider the problem of prepositional phrase (PP) ambiguity. While there have been a number of recent studies concerning the use of statistical techniques for resolving single PP attachments, i.e. in constructions of the form [V NP PP], we are unaware of published work which applies these techniques to the more general, and pathological, problem of multiple PPs, e.g. [V NP PP1 PP2 ...]. In particular, the multiple PP attachment problem results in sparser data which must be used to resolve greater ambiguity: a strong test for any probabilistic approach.

We begin with an overview of techniques which have been used for PP attachment disambiguation, and then consider how one of the most successful of these, the backed-off estimation technique, can be applied to the general problem of multiple PP attachment.

2 Existing Models of Attachment

Attempts to resolve the problem of PP attachment in computational linguistics are numerous, but the problem is hard and success rate typically depends on the domain of application. Historically, the shift from attempts to resolve the problem completely, by using heuristics developed using typical AI techniques (Jensen and Binot, 1987; Marcus, 1980; Crain and Steedman, 1985; Altmann and Steedman, 1988) has left the place for attempts to solve the problem by less expensive means, even if only approximately. As shown by many psycholinguistic and practical studies (Ford et al., 1982; Taraban and McClelland, 1988; Whittemore et al., 1990), lexical information is one of the main cues to PP attachment disambiguation.

In one of the earliest attempts to resolve the problem of PP attachment ambiguity using lexical measures, Hindle and Rooth (1993) show that a measure of mutual information limited to lexical association can correctly resolve 80% of the cases of PP attachment ambiguity, confirming the initial hypothesis that lexical information, in particular co-occurrence frequency, is central in determining the choice of attachment.

The same conclusion is reached by Brill and Resnik (1994). They apply transformation-based learning (Brill, 1993) to the problem of learning different patterns of PP attachment. After acquiring 471 patterns of PP attachment, the parser can correctly resolve approximately 80% of the ambiguity. If word classes (Resnik, 1993) are taken into account, only 266 rules are needed to perform at 80% accuracy.

Magerman and Marcus (1991) report 54/55 correct PP attachments for Pearl, a probabilistic chart parser, with Earley style prediction, that integrates lexical co-occurrence knowledge into a probabilistic context-free grammar. The probabilities of the rules are conditioned on the parent rule and on the trigram centered at the first input symbol that would be covered by the rule. Even if the parser has been tested only in the direction giving domain, where the behaviour of prepositions is very consistent, it shows that a mixture of lexical and structural information is needed to solve the problem successfully.

Collins and Brooks (1995) propose a 4-gram model for PP disambiguation which exploits backed-off estimation to smooth null events (see next section). Their model achieves 84.5% accuracy. The authors point out that prepositions are the most informative element in the tuple, and that taking low frequency events into account improves performance by several percentage points. In other words, in solving the PP attachment problem, backing-off is not advantageous unless the tuple that is being tested is not present in the training set (it has zero counts). Moreover, tuples that contain prepositions are the most informative.

The second result is roughly confirmed by Brill and Resnik, (ignoring the importance of n2 when it is a temporal modifier, such as *yesterday, today*). In their work, the top 20 transformations learned are primarily based on specific prepositions.

3 Back-off Estimation

The PP attachment model presented by Collins and Brooks (1995) determines the most likely attachment for a particular prepositional phrase by estimating the probability of the attachment. We let C represent the attachment event, where $C = 1$ indicates that the PP attaches to the verb, and $C = 2$ indicates attachment to the object NP. The attachment is conditioned by the relevant head words, a 4-gram, of the VP.

- Tuple format: (C, v, n1, p, n2)
- So: *John read [[the article] [about the budget]]*
- Is encoded as: (2, read, article, about, budget)

Using a simple maximal likelihood approach, the best attachment for a particular input tuple (v,n1,p,n2) can now be determined from the training data via the following equation:

$$argmax_i \; \hat{p}(C = i|v, n1, p, n2) = \frac{f(i, v, n1, p, n2)}{f(v, n1, p, n2)} \quad (1)$$

Here f denotes the frequency with which a particular tuple occurs. Thus, we can estimate the probability for each configuration $1 \leq i \leq 2$, by counting the number of times the four head words were observed in that configuration, and dividing it by the total number of times the 4-tuple appeared in the training set.

While the above equation is perfectly valid in theory, sparse data means it is rather less useful in practice. That is, for a particular sentence containing a PP attachment ambiguity, it is very likely that we will never have seen the precise (v,n1,p,n2) quadruple before in the training data, or that we will have only seen it rarely.[1] To address this problem, they employ backed-off estimation when zero counts occur in the training data. Thus if $f(v, n1, p, n2)$ is zero, they 'back-off' to an alternative estimation of $\hat{p}$ which relies on 3-tuples rather than 4-tuples:

$$\hat{p}_3(C = i|v, n1, p, n2) = \quad (2)$$
$$\frac{f(i, v, n1, p) + f(i, v, p, n2) + f(i, n1, p, n2)}{f(v, n1, p) + f(v, p, n2) + f(n1, p, n2)}$$

Similarly, if no 3-tuples exist in the training data, they back-off further:

$$\hat{p}_2(C = i|v, n1, p, n2) = \quad (3)$$
$$\frac{f(i, v, p) + f(i, n1, p) + f(i, p, n2)}{f(v, p) + f(n1, p) + f(p, n2)}$$
$$\hat{p}_1(C = i|v, n1, p, n2) = \frac{f(i, p)}{f(p)} \quad (4)$$

The above equations incorporate the proposal by Collins and Brooks that only tuples including the preposition should be considered, following their results that the preposition is the most informative lexical item. Using this technique, Collins and Brooks achieve an overall accuracy of 84.5%.

[1] Though as Collins and Brooks point out, this is less of an issue since even low counts are still useful.

4 The Multiple PP Attachment Problem

Previous work has focussed on the problem of single PP attachment, in configurations of the form [V NP PP] where both the NP and the PP are assumed to be attached within the VP. The algorithm presented in the previous section, for example, simply determines the maximally likely attachment event (to NP or VP) based on the supervised training provided by a parsed corpus. The broader value of this approach, however, remains suspect until it can be demonstrated to apply more generally. We now consider how this approach - and the use of lexical statistics in general - might be naturally extended to handle the more difficult problem of multiple PP attachment. In particular, we investigate the PP attachment problem in cases containing two PPs, [V NP PP1 PP2], and three PPs, [V NP PP1 PP2 PP3], with a view to determining whether n-gram based parse disambiguation models which use the backed-off estimate can be usefully applied. Multiple PP attachment presents two challenges to the approach:

1. For a single PP, the model must make a choice between two structures. For multiple PPs, the space of possible structural configurations increases dramatically, placing increased demands on the disambiguation technique.
2. Multiple PP structures are less frequent, and contain more words, than single PP structures. This substantially increases the sparse data problems when compared with the single PP attachment case.

4.1 Materials and Method

To carry out the investigation, training and test data were obtained from the Penn Tree-bank, using the `tgrep` tools to extract tuples for 1-PP, 2-PP, and 3-PP cases. For the single PP study, VP attachment was coded as 1 and NP attachment was coded as 2. A database of quadruples of the form *(configuration,v,n,p)* was then created. The table below shows the two configurations and their frequencies in the corpus.

Configuration	Structure	Counts
1	$[_{VP}$NP PP]	7740
2	$[_{VP}[_{NP}$PP]]	12223

The same procedure was used to create a database of 6-tuples *(configuration,v,n1,p1,n2,p2)* for the attachment of 2 PPs. The values for the configuration varies over a range 1..5, corresponding to the 5 grammatical structures possible for 2 PPs, shown and exemplified below with their counts in the corpus.[2]

Config	Structure	Counts
1	$[_{VP}$V NP PP PP]	535
2	$[_{VP}$V $[_{NP}$NP PP] PP]	1160
3	$[_{VP}$V $[_{NP}[_{PP}$P $[_{NP}$NP PP]]]]	1394
4	$[_{VP}$V NP $[_{PP}[_{NP}$NP PP]]]	1055
5	$[_{VP}$V $[_{NP}$NP PP PP]]	539

1. The agency said it will **keep** the **debt under review for** possible further downgrade.
2. Penney decided to **extend** its **involvement with** the **service for** at least five years.
3. The bill was then sent back to the House to resolve the question of how to **address** budget **limits on** credit **allocations for** the Federal Housing Administration.
4. Sears officials insist they don't intend to **abandon** the everyday pricing **approach in** the **face of** the poor results.
5. Mr. Ridley hinted at this motive in **answering questions from** members of **Parliament after** his announcement

Finally, a database of 8-tuples *(configuration,v,n1,p1,n2,p2,n3,p3)* was created for 3 PPs. The value of the configuration varies over a range 1..14, corresponding to the 14 structures possible for 3 PPs, shown in Table 1 with their counts in the corpus.

The above datasets were then split into training and test sets by automatically extracting stratified samples. For PP1, we extracted quadruples of about 5% of the total (1014/19963). We then created a test set for PP2 which is a subset of the PP1 test set, and approximately 10% of the 2 PP tuples (464/4683). Similarly, the test set for PP3 is a subset of the PP2 test set of approximately 10% (94/907). It is important that the test sets are subsets to ensure that, e.g., a PP2 test case doesn't appear in the PP1 training set, since the PP1 data is used by our algorithm to estimate PP2 attachment, and similarly for the PP3 test set.

4.2 Does Distance Matter?

In exploring multiple PP attachment, it seems natural to investigate the effects of the distance of the PP from the verb. The following table reports accuracy of noun-attachment, when the attachment decision is conditioned only on the preposition and on the distance - in other words, when estimating $\hat{p}(1|p,d)$ where 1 is the coding of the attachment to the noun, p is the preposition and $d = \{1,2,3\}$.[3]

[2]We did not consider the left-recursive NP structure for the 2 PP (or indeed 3 PP) cases. Checking the frequency of their occurrences revealed that there were only 2 occurrences of $[_{VP}[_{NP}[_{NP}[_{NP}$PP] PP]]] structures in the corpus.

[3]These figures are to be taken only as an indication of a trend, as they represent the accuracy obtained by

Configuration	Structure	Counts
1	$[_{VP}$V NP PP PP PP]	15
2	$[_{VP}$V NP PP $[_{PP}$P $[_{NP}$PP]]]	86
3	$[_{VP}$V $[_{NP}$NP PP] PP PP]	63
4	$[_{VP}$V $[_{NP}$NP PP] $[_{PP}$P $[_{NP}$PP]]]	168
5	$[_{VP}$V $[_{NP}[_{PP}$P $[_{NP}$NP PP]]] PP]	81
6	$[_{VP}$V $[_{NP}[_{PP}$P $[_{NP}$NP PP]]PP]]	31
7	$[_{VP}$V $[_{NP}[_{PP}$P $[_{NP}$NP PP PP]]]]	47
8	$[_{VP}$V $[_{NP}[_{PP}$P $[_{NP}$NP $[_{PP}$P $[_{NP}$PP]]]]]]	142
9	$[_{VP}$V NP $[_{PP}[_{NP}$NP PP]] PP]	47
10	$[_{VP}$V NP $[_{PP}[_{NP}$NP PP PP]]]	34
11	$[_{VP}$V NP $[_{PP}[_{NP}$NP $[_{PP}$P $[_{NP}$PP]]]]]	80
12	$[_{VP}$V $[_{NP}$NP PP PP] PP]	20
13	$[_{VP}$V $[_{NP}$NP PP PP PP]]	21
14	$[_{VP}$V $[_{NP}$NP PP $[_{PP}$P $[_{NP}$PP]]]]	72

Table 1: Corpus counts for the 14 structures possible for 3-PP sequences.

	1 PP	2 PP	3 PP	Total	All
Count	20299	4711	939	25949	25949
Correct	15173	3525	755	19453	19349
%	74.7	74.8	80.4	75	74.5

It can be seen from these figures that conditioning the attachment according to both preposition and distance results in only a minor improvement in performance, mostly because separating the biases according to preposition *distance* increases the sparse data problem. It must be noted, however, that counts show a steady increase in the proportion of low attachments for PP further from the verb, as shown in the table below. The simplest explanation of this fact is that more (inherently) noun-attaching prepositions must be occurring in 2nd and 3rd positions. This predicts that the distribution of preposition occurrences changes from PP1 to PP3, with an increase in the proportion of low attaching PPs. Globally, failure to use position results in 41.3% of correct configurations, while use of position results in 45% correct attachments.

	1 PP	2 PP	3 PP	Total
Count	20299	4711	939	25949
Low	12223	3063	706	15992
% Low	60.2	65.0	75.1	61.6

Having established that the distance parameter is not as influential a factor as we hypothesized, we exploit the observation that attachment preferences do not significantly change depending on the distance of the PP from the verb. In the following section, we discuss an extension of the back-off estimation model that capitalizes on this property.

testing on the training data. Moreover, we are only considering 2 attachment possibilities for each preposition, either it attaches to the verb or it attaches to the lowest noun.

5 The Generalized Backed-Off Algorithm

The algorithm for attaching the first preposition is almost identical to that of Collins and Brooks (1995), and we follow them in including only tuples which contain the preposition. We do not, however, use the final noun (following the preposition) in any of our tuples, thus basing our model of PP1 on three, rather than four, head words.

Procedure B1:

The most likely configuration is:

$arg\ max_i\ \hat{p}_1(C_2 = i|v,n,p)$, where $1 \leq i \leq 2$

1. IF $f(v,n,p) > 0$ THEN
 $\hat{p}_1(i|v,n,p) = \frac{f(i,v,n,p)}{f(v,n,p)}$
2. ELSEIF $f(v,p) + f(n,p) > 0$ THEN
 $\hat{p}_1(i|v,n,p) = \frac{f(i,v,p)+f(i,n,p)}{f(v,p)+f(n,p)}$
3. ELSEIF $f(p) > 0$ THEN
 $\hat{p}_1(i|v,n,p) = \frac{f(i,p)}{f(p)}$
4. ELSE $\hat{p}_1(1|v,n,p) = 0, \hat{p}_1(2|v,n,p) = 1$

In this case i denotes the attachment configuration: $i = 1$ is VP attachment, $i = 2$ is NP attachment. The subscript on C_2 is used simply to make clear that C has 2 possible values. In the subsequent algorithms, C_5 and C_{14} are used to indicate the larger sets of configurations.

The algorithm used to handle the cases containing 2PPs is shown in Figure 1, where j ranges over

Procedure B2

The most likely configuration is:
$arg\ max_j\ \hat{p}_2(C = j|v, n1, p1, n2, p2)$, where $1 \leq j \leq 5$

1. IF $f(v, n1, p1, n2, p2) > 0$ THEN
$$\hat{p}_2(j) = \frac{f(j,v,n1,p1,n2,p2)}{f(v,n1,p1,n2,p2)}$$
2. ELSEIF $f(n1, p1, n2, p2) + f(v, p1, n2, p2) + f(v, n1, p1, p2) > 0$ THEN
$$\hat{p}_2(j) = \frac{f(j,n1,p1,n2,p2)+f(j,v,p1,n2,p2)+f(j,v,n1,p1,p2)}{f(n1,p1,n2,p2)+f(v,p1,n2,p2)+f(v,n1,p1,p2)}$$
3. ELSEIF $f(p1, n2, p2) + f(v, p1, p2) + f(n1, p1, p2) > 0$ THEN
$$\hat{p}_2(j) = \frac{f(j,p1,n2,p2)+f(j,v,p1,p2)+f(j,n1,p1,p2)}{f(p1,n2,p2)+f(v,p1,p2)+f(n1,p1,p2)}$$
4. ELSE **Competitive Backed-off Estimate**

Figure 1: Procedure B2

the five possible attachment configurations outlined above.

The first three steps use the standard backed-off estimation, again including only those tuples containing *both* prepositions. However, after backing-off to three elements, we abandon the standard backed-off estimation technique. The combination of sparse data, and too few lexical heads, renders backed-off estimation ineffective. Rather, we propose a technique which makes use of the richer data available from the PP1 training set. Our hypothesis is that this information will be useful in determining the attachments of subsequent PPs as well. This is motivated by our observations, reported in the previous section, that the distribution of high-low attachments for specific prepositions did not vary significantly for PPs further from the verb. The **Competitive Backed-Off Estimate** procedure, presented below, operates by initially fixing the configuration of the first preposition (to either the VP or the direct object NP), and then considers how the second preposition would be optimally attached into the configuration.

Procedure Competitive Backed-off Estimate

1. C'_2 is the most likely configuration for PP1,
$arg\ max_i\ \hat{p}_1(C'_2 = i|v, n1, p1)$
2. C''_2 is the preferred configuration for PP2 w.r.t n2,
$arg\ max_i\ \hat{p}_1(C''_2 = i|v, n2, p2)$
3. C'''_2 is the preferred configuration for PP2 w.r.t n1,
$arg\ max_i\ \hat{p}_1(C'''_2 = i|v, n1, p2)$
4. **Find Best Configuration**

First we determine C'_2, on which depends the attachment of p1. We then determine C''_2, which indicates the preference for p2 to attach to the VP or to n2, and C'''_2, which is the preference for p2 to attach to the VP or to n1. Given the preferred configurations C'_2, C''_2, and C'''_2, we now must determine the best of the five possible configurations, C_5, for the entire VP.

Procedure Find Best Configuration

1. IF $C'_2 = 1$ and $C''_2 = 1$ THEN
$C_5 \leftarrow 1$
2. ELSEIF $C'_2 = 1$ and $C''_2 = 2$ THEN
$C_5 \leftarrow 4$
3. ELSEIF $C'_2 = 2$ and $C''_2 = 1$ and $C'''_2 = 1$ THEN
$C_5 \leftarrow 2$
4. ELSEIF $C'_2 = 2$ and $C''_2 = 2$ and $C'''_2 = 1$ THEN
$C_5 \leftarrow 3$
5. ELSEIF $C'_2 = 2$ and $C''_2 = 1$ and $C'''_2 = 2$ THEN
$C_5 \leftarrow 2$
6. ELSEIF $C'_2 = 2$ and $C''_2 = 2$ and $C'''_2 = 2$ THEN
tie-break
   (a) IF $f(2, v, n2, p2) < f(2, v, n1, p2)$ THEN
   $C_5 \leftarrow 5$
   (b) ELSE $C_5 \leftarrow 3$

The tests 1 to 5 simply use the attachment values C'_2, C''_2, and C'''_2 to determine C_5: the best configuration. In the final instance, step 6, where the C''_2 indicates a preference for n2 attachment, and C'''_2 indicates a preference for n1 attachment a tie-break is necessary to determine which noun to attach to. As a first approximation, we use the frequency of occurrence used in determining these preferences, rather than the probability for each preference. That is, we favour the bias for which there is more evidence, though whether this is optimal remains an empirical question. For example, if C_2'' is based on 4 observations, and C_2''' is based on 7, then the C_2''' preference is considered stronger.

Having constructed the algorithm to determine the best configuration for 2 PPs, we can similarly generalize the algorithm to handle three. In this case k denotes one of fourteen possible attachment configurations shown earlier. The pseudo code for procedure B3 is shown below, simplified for reasons of space.

Procedure B3

The most likely configuration is:

$arg\ max_k\ \hat{p}_3(C_{14} = k|v, n1, p1, n2, p2, n3, p3)$, where $1 \leq k \leq 14$

1. IF $f(v, n1, p1, n2, p2, n3, p3) > 0)$ THEN $\hat{p}_3(k) = \frac{f(k,v,n1,p1,n2,p2,n3,p3)}{f(v,n1,p1,n2,p2,n3,p3)}$
2. ELSE Try backing-off to 6 or 5 items ...
3. ELSE Competitive Backed-off Estimate:
   (a) Use **Procedure B2** to determine C_5', the configuration of p1 and p2
   (b) Compute C_2'', C_2''', C_2'''', the preferred attachment of p3 w.r.t n1, n2, n3 respectively
   (c) Determine the best configuration

Again, we back-off up to two times, always including tuples which contain the three prepositions. After this, backing-off becomes unstable, so we use the **Competitive Backed-off Estimate**, as above, but scaled up to handle the three prepositions and fourteen possible configurations.

5.1 Results

To evaluate the performance of our algorithm, we must first determine what the expected baseline, or lower-bound on, performance would be. Given the variation in the number of possible configurations across the three cases, the performance expected due to chance would be 50% for 1 PP, 20% for 2 PPs, and 7% for 3 PPs. A better baseline is the performance that would be expected by simply adopting the most likely configuration, without regard to lexical items. This is shown in the table below, with the most frequent configuration shown in parentheses.

	PP1(2)	PP2(5)	PP3(14)
Total	19963	4683	907
Most Frequent	12223(2)	1394(3)	168(4)
Percent Correct	61.2%	29.8%	18.5%

Table 2 presents the performance of the competitive backed-off estimation algorithm on the test data. As can be seen, the performance for PP1 replicates the findings of Collins and Brooks, who achieved 84.5% (using 4 lexical items, compared to our three). For PP2 performance is again high, recalling that the algorithm is discriminating five possible attachment configurations, and the baseline expectation was only 29.8%. Similarly for PP3, our performance of 43.6% accuracy (discriminating fourteen configurations) far out strips the baseline of 18.5%.

6 Conclusions

The backed-off estimate has been demonstrated to work successfully for single PP attachment, but the sparse data problem renders it impractical for use in more complex constructions such as multiple PP attachment; there are too many configurations, too many head words, too few training examples. In this paper we have demonstrated, however, that the relatively rich training data obtained for the first preposition can be exploited in attaching subsequent PPs. The algorithm incrementally fixes each preposition into the configuration and the more informative PP1 training data is exploited to settle the competition for possible attachments for each subsequent preposition. Performance is considerably better than both chance and the naive baseline technique. The generalized backed-off estimation approach which we have presented constitutes a practical solution to the problem of multiple PP disambiguation. This further suggests that backed-off estimation may be successfully integrated into more general syntactic disambiguation systems.

Acknowledgments

We gratefully acknowledge the support of the British Council and the Swiss National Science Foundation on grant 83BC044708 to the first two authors, and on grant 12-43283.95 and fellowship 8210-46569 from the Swiss NSF to the first author. We thank the audiences at Edinburgh and Pennsylvania for their useful comments. All errors remain our responsibility.

References

Gerry Altmann and Mark Steedman. 1988. Interaction with context during human sentence processing. *Cognition*, 30(3):191–238.

Eric Brill and Philip Resnik. 1994. A rule-based approach to prepositional phrase attachment disambiguation. International Conference on Computational Linguistics (COLING).

Eric Brill. 1993. *A Corpus-based Approach to Language Learning*. Ph.D. thesis, University of Pennsylvania, Philadelphia, PA.

	PP1		PP2		PP3	
	Total	Correct	Total	Correct	Total	Correct
No back-off	300	285	36	35	1	1
Back-off 1	614	510	61	54	1	1
Back-off 2	100	60	232	161	3	3
Competitive	NA	NA	135	73	89	36
Total	1014	855	464	323	94	41
Percent	84.3%		69.6%		43.6%	

Table 2: Performance of the competitive backed-off estimation algorithm on the test data.

Ken Church and R. Patil. 1982. Coping with syntactic ambiguity or how to put the block in the box on the table. *American Journal of Computational Linguistics*, 8(3-4):139–149.

Michael Collins and James Brooks. 1995. Prepositional phrase attachment through a backed-off model. Third Workshop on Very Large Corpora.

Stephen Crain and Mark Steedman. 1985. On not being led up the garden path: The use of context by the psychological parser. In David R. Dowty, Lauri Kartunnen, and Arnold M. Zwicky, editors, *Natural Language Processing: Psychological, Computational, and Theoretical Perspectives*, pages 320–358. Cambridge University Press, Cambridge.

Marylin Ford, Joan Bresnan, and Ron Kaplan. 1982. A competence-based theory of syntactic closure. In Joan Bresnan, editor, *The Mental Representations of Grammatical Relations*, pages 727–796. MIT Press, Cambridge, MA.

Donald Hindle and Mats Rooth. 1993. Structural ambiguity and lexical relations. *Computational Linguistics*, 19(1):103–120.

Graheme Hirst. 1987. *Semantic Interpretation and the Resolution of Ambiguity*. Cambridge University Press.

Karen Jensen and Jean-Louis Binot. 1987. Disambiguating prepositional phrase attachments by using on-line dictionary definitions. *Computational Linguistics*, 13(3-4):251–260.

David Magerman and Mitch Marcus. 1991. Pearl: A probabilistic parser. In *Proceedings of the European Chapter of the Association for Computational Linguistics*, Berlin.

Mitch Marcus. 1980. *A Theory of Syntactic Recognition for Natural Language*. MIT Press, Cambridge, MA.

Philip Resnik. 1993. *Selection and Information: A Class-Based Approach to Lexical Relationships*. Ph.D. thesis, University of Pennsylvania.

Roman Taraban and James L. McClelland. 1988. Constituent attachment and thematic role assignment in sentence processing: Influences of content-based expectations. *Journal of Memory and Language*, 27:597–632.

Greg Whittemore, Kathleen Ferrara, and Hans Brunner. 1990. Empirical study of predictive power of simple attachment schemes for post-modifiers prepositional phrases. pages 23–30, Pittsburgh, PA. Association for Computational Linguistics.