Thesauruses for Prepositional Phrase Attachment

Mark McLauchlan
Department of Informatics
University of Sussex
Brighton, BN1 9RH
mrm21@sussex.ac.uk

Abstract

Probabilistic models have been effective in resolving prepositional phrase attachment ambiguity, but sparse data remains a significant problem. We propose a solution based on similarity-based smoothing, where the probability of new PPs is estimated with information from similar examples generated using a thesaurus. Three thesauruses are compared on this task: two existing generic thesauruses and a new specialist PP thesaurus tailored for this problem. We also compare three smoothing techniques for prepositional phrases. We find that the similarity scores provided by the thesaurus tend to weight distant neighbours too highly, and describe a better score based on the rank of a word in the list of similar words. Our smoothing methods are applied to an existing PP attachment model and we obtain significant improvements over the baseline.

1 Introduction

Prepositional phrases are an interesting example of syntactic ambiguity and a challenge for automatic parsers. The ambiguity arises whenever a prepositional phrase can modify a preceding verb or noun, as in the canonical example *I saw the man with the telescope*. In syntactic terms, the prepositional phrase attaches either to the noun phrase or the verb phrase. Many kinds of syntactic ambiguity can be resolved using structural information alone (Briscoe and Carroll, 1995; Lin, 1998a; Klein and Manning, 2003), but in this case both candidate structures are perfectly grammatical and roughly equally likely. Therefore ambiguous prepositional phrases require some kind of additional context to disambiguate correctly. In some cases a small amount of lexical knowledge is sufficient: for example *of* almost always modifies the noun. Other cases, such as the telescope example, are potentially much harder since discourse or world knowledge might be required.

Fortunately it is possible to do well at this task just by considering the lexical preferences of the words making up the PP. Lexical preferences describe the tendency for certain words to occur together or only in specific constructions. For example, *saw* and *telescope* are more likely to occur together than *man* and *telescope*, so we can infer that the correct attachment is likely to be verbal. The most useful lexical preferences are captured by the quadruple (v, n_1, p, n_2) where v is the verb, n_1 is the head of the direct object, p is the preposition and n_2 is the head of the prepositional phrase. A benchmark dataset of 27,937 such quadruples was extracted from the Wall Street Journal corpus by Ratnaparkhi et al. (1994) and has been the basis of many subsequent studies comparing machine learning algorithms and lexical resources. This paper examines the effect of particular smoothing algorithms on the performance of an existing statistical PP model.

A major problem faced by any statistical attachment algorithm is sparse data, which occurs when plausible PPs are not well-represented in the training data. For example, if the observed frequency of a PP in the training is zero then the maximum likelihood estimate is also zero. Since the training corpus only represents a fraction of all possible PPs, this is probably an underestimate of the true probability. An appealing course of action when faced with an unknown PP is to consider similar known examples instead. For example, we may not have any data for *eat pizza with fork*, but if we have seen *eat pasta with fork* or even *drink beer with straw* then it seems reasonable to base our decision on these instead.

Similarity is a rather nebulous concept but for our purposes we can define it to be *distributional similarity*, where two words are considered similar if they occur in similar contexts. For example, *pizza* and *pasta* are similar since they both often occur as the direct object of *eat*. A thesaurus collects together lists of such similar words. The first step in constructing a thesaurus is to collect co-occurrence statistics from some large corpus of text. Each word is assigned a probability distribution describing the probability of it occurring with all other words, and by comparing distributions we can arrive at a similarity score. The corpus, co-occurrence relationships and distributional similarity metric all affect the nature of the final thesaurus.

There has been a considerable amount of research comparing corpora, co-occurrence relations and similarity measures for general-purpose thesauruses, and these thesauruses are often compared against wide-coverage and general purpose semantic resources such as WordNet. In this paper we examine whether it is useful to tailor the thesaurus to the task. General purpose thesauruses list words that tend to occur together in free text; we want to find words that behave in similar ways specifically within prepositional phrases. To this end we create a PP thesaurus using existing similarity metrics but using a corpus consisting of automatically extracted prepositional phrases.

A thesaurus alone is not sufficient to solve the PP attachment problem; we also need a model of the lexical preferences of prepositional phrases. Here we use the back-off model described in (Collins and Brooks, 1995) but with maximum likelihood estimates smoothed using similar PPs discovered using a thesaurus. Such *similarity-based* smoothing methods have been successfully used in other NLP applications but our use of them here is novel. A key difference is that smoothing is not done over individual words but over entire prepositional phrases. Similar PPs are generated by replacing each component word with a distributionally similar word, and we define a similarity functions for comparing PPs. We find that using a score based on the rank of a word in the similarity list is more accurate than the actual similarity scores provided by the thesaurus, which tend to weight less similar words too highly.

In Section 2 we cover related work in PP attachment and smoothing techniques, with a brief comparison between similarity-based smoothing and the more common (for PP attachment) class-based smoothing. Section 3 describes Collins' PP attachment model and our thesaurus-based smoothing extensions. Section 4 discusses the thesauruses used in our experiment and describes how the specialist thesaurus is constructed. Experimental results are given in Section 5 and we show statistically significant improvements over the baseline model using generic thesauruses. Contrary to our hypothesis the specialist thesaurus does not lead to significant improvements and we discuss possible reasons why it underperforms on this task.

2 Previous work

2.1 PP attachment

Early work on PP attachment disambiguation used strictly syntactic or high-level pragmatic rules to decide on an attachment (Frazier, 1979; Altman and Steedman, 1988). However, work by Whittemore et al. (1990) and Hindle and Rooth (1993) showed that simple lexical preferences alone can deliver reasonable accuracy. Hindle and Rooth's approach was to use mostly unambiguous (v, n_1, p) triples extracted from automatically parsed text to train a maximum likelihood classifier. This achieved around 80% accuracy on ambiguous samples.

This marked a flowering in the field of PP attachment, with a succession of papers bringing the whole armoury of machine learning techniques to bear on the problem. Ratnaparkhi et al. (1994) trained a maximum entropy model on (v, n_1, p, n_2) quadruples extracted from the Wall Street Journal corpus and achieved 81.6% accuracy. The Collins and Brooks (1995) model scores 84.5% accuracy on this task, and is one of the most accurate models that do not use additional supervision. The current state of the art is 88% reported by Stetina and Nagao (1997) using the WSJ text in conjunction with WordNet. The next section discusses other specific approaches that incorporate smoothing techniques.

2.2 Similarity-based smoothing

Smoothing for statistical models involves adjusting probability estimates away from the maximum likelihood estimates to avoid the low probabilities caused by sparse data. Typically this involves mixing in probability distributions that have less context and are less likely to suffer from sparse data problems. For example, if the probability of an attachment given a PP $p(a|v, n_1, p, n_2)$ is undefined because that quadruple was not seen in the training data, then a less specific distribution such as $p(a|v, n_1, p)$ can be used instead. A wide range of different techniques have been proposed (Chen and Goodman, 1996) including the backing-off technique used by Collins' model (see Section 3).

An alternative but complementary approach is to mix in probabilities from distributions over "similar" contexts. This is the idea behind both similarity-based and class-based smoothing. Class-based methods cluster similar words into classes which are then used in place of actual words. For example the class-based language model of (Brown et al., 1992) is defined as:

$$p(w_2|w_1) = p(w_2|c_2)p(c_2|c_1) \qquad (1)$$

This helps solve the sparse data problem since the number of classes is usually much smaller than the number of words.

Class-based methods have been applied to the PP attachment task in several guises, using both automatic clustering and hand-crafted classes such as WordNet. Li and Abe (1998) use both WordNet and an automatic clustering algorithm to achieve 85.2% accuracy on the WSJ dataset. The maximum entropy approach of Ratnaparkhi et al. (1994) uses the mutual information clustering algorithm described in (Brown et al., 1992). Although class-based smoothing is shown to improve the model in both cases, some researchers have suggested that clustering words is counterproductive since the information lost by conflating words into broader classes outweighs the benefits derived from reducing data sparseness. This remains to be proven conclusively (Dagan et al., 1999).

In contrast, similarity-based techniques do not discard any data. Instead the smoothed probability of a word is defined as the total probability of all similar words $S(w)$ as drawn from a thesaurus, weighted by their similarity $\alpha(w, w')$. For example, the similarity-based language model of (Dagan et al., 1999) is defined as:

$$p(w_2|w_1) = \sum_{w'_1 \in S(w_1)} \alpha(w_1, w'_1) p(w_2|w'_1) \qquad (2)$$

where $\sum_{w'_1 \in S(w_1)} \alpha(w_1, w'_1) = 1$. The similarity function reflects how often the two words appear in the same context. For example, Lin's similarity metric (Lin, 1998b) used in this paper is based on an information-theoretic comparison between a pair of co-occurrence probability distributions.

This language model was incorporated into a speech recognition system with some success (Dagan et al., 1999). Similarity-based methods have also been successfully applied word sense disambiguation (Dagan et al., 1997) and extraction of grammatical relations (Grishman and Sterling, 1994). Similarity-based smoothing techniques of the kind described here have not yet been applied to probabilistic PP attachment models. The memory-based learning approach of (Zavrel et al., 1997) is the closest point of contact and shares many of the same ideas, although the details are quite different. Memory-based learning consults similar previously-seen examples to make a decision, but the similarity judgements are usually based on a strict feature matching measure rather than on co-occurrence statistics. Under this scheme *pizza* and *pasta* are as different as *pizza* and *Paris*. To overcome this Zavrel et al. also experiment with features based on a reduced-dimensionality vector of co-occurrence statistics and note a small (0.2%) increase in performance, leading to a final accuracy of 84.4%.

Our use of specialist thesauruses for this task is also novel, although in they have been used in the somewhat related field of selectional preference acquisition by Takenobu et. al. (1995). Different thesauruses were created for different grammatical roles such as subject and object, and used to build a set of word clusters. Clusters based on specialist thesauruses were found to predict fillers for these roles more accurately than generic clusters.

$p(a|v, n_1, p, n_2) =$

1. $\frac{f(a,v,n_1,p,n_2)}{f(v,n_1,p,n_2)}$

2. $\frac{f(a,v,n_1,p)+f(a,v,p,n_2)+f(a,n_1,p,n_2)}{f(v,n_1,p)+f(v,p,n_2)+f(n_1,p,n_2)}$

3. $\frac{f(a,v,p)+f(a,n_1,p)+f(a,p,n_2)}{f(v,p)+f(n_1,p)+f(p,n_2)}$

4. $\frac{f(a,p)}{f(p)}$

5. Default: noun attachment

Figure 1: Collins and Brooks (1995) backing off algorithm. A less specific context is used when the denominator is zero or $p(a|v, n_1, p, n_2) = 0.5$.

3 Smoothing

Our baseline model is Collins and Brooks (1995) model, which implements the popular and effective backing-off smoothing technique. The idea is to initially use $p(a|v, n_1, p, n_2)$, but if there isn't enough data to support a maximum likelihood estimate of this distribution, or $p(a|v, n_1, p, n_2) = 0.5$, then the algorithm backs off and uses a distribution with less conditioning context. The backing off steps are shown in Figure 1.

If we use the similarity-based language model shown in (2) as a guide, then we can create a smoothed version of Collins' model using the weighted probability of all similar PPs (for brevity we use c in to indicate the context, in this case an entire PP quadruple):

$$p(a|c) = \sum_{c' \in S(c)} \alpha(c, c') p(a|c') \qquad (3)$$

In contrast to the language model shown in (2), the set of similar contexts $S(c)$ and similarity function $\alpha(c, c')$ must be defined for multiple words (we abuse our notation slightly by using the same α and S for both PPs and words, but the meaning should be clear from the context). Thesauruses only supply neighbours and similarity scores for single words, but we can generate distributionally similar PPs by replacing each word in the phrase independently with a similar one provided by the thesaurus. For example, if *eat* has two neighbours: $S(eat) = \{drink, enjoy\}$, and *pizza* has just one: $S(pizza) = \{pasta\}$, then the following examples will be generated for *eat pizza with fork*:

eat pasta with fork
drink pizza with fork
drink pasta with fork
enjoy pizza with fork
enjoy pasta with fork

Clearly this strategy of generates some nonsensical or at least unhelpful examples. This is not necessarily a serious problem since such instances should occur at best infrequently in the training data. Unfortunately our baseline model will back off and attempt to provide a reasonable probability for them all, for example by using $p(a|with)$ in place of $p(a|enjoy, pasta, with, fork)$. This introduces unwanted noise into the smoothed probability estimate.

Our solution is to apply smoothing to the counts used by the probability model. The smoothed frequency of a prepositional phrase $f_s(a, c)$ is the weighted average frequency of the set of similar PPs $S(c)$:

$$f_s(a, c) = \sum_{c' \in S(c)} \alpha(c, c') f(a, c') \tag{4}$$

These smoothed frequencies are used to calculate the conditional probabilities for the model. For example, the probability distribution in step one is defined as:

$$p(a|v, n_1, p, n_2) = \frac{f_s(a, v, n_1, p, n_2)}{f_s(v, n_1, p, n_2)}$$

Distributionally similar triples are generated for step two using the same word replacement strategy and smoothed frequency estimates for triples are calculated in the same way as quadruples. We back off to a smaller amount of context if the smoothed denominator is less than 1. This is done for empirical reasons, since decisions based on very low frequency counts are unreliable. The distributions used in steps three and four are not smoothed. Attempting to disambiguate a PP based on just two words is risky enough; introducing similar PPs found by replacing these two words with synonyms introduces too much noise.

Quadruples and triples are more reliable since the context rules out those unhelpful PPs. For example, our model automatically deals with polysemous words without the need for explicit word sense disambiguation. Although thesauruses do conflate multiple senses in their neighbour lists, implausible senses result in infrequent PPs. The similarity set for the PP *open plant in Korea* might contain *open tree in Korea* but the latter's frequency is likely to be zero. Generating triples is riskier since there is less context to rule out unlikely PPs: the triple *tree in Korea* is more plausible and possibly misleading. But our model does have a natural preference for the most frequent sense in the thesaurus training corpus, which is a useful heuristic for word sense disambiguation (Pedersen and Bruce, 1997). For example, if the thesaurus is trained on business text then *factory* will be ranked higher than *tree* when the thesaurus trained on a business corpus (this issue is discussed further in Section 5.2).

Finally, to complete our PP attachment scheme we need to define a similarity function between PPs, expressed fully as $\alpha\big((v, n_1, p, n_2), (v', n_1', p', n_2')\big)$. The raw materials we have to work with are the similarity scores for matching pairs of verbs and nouns as given by the thesaurus. We do not smooth preposition counts. In this paper we compare three similarity measures:

- **average**: The average similarity score of all word pairs in the PP using the similarity measure provided by the thesaurus. For example, $\alpha(c, c')$ when $c = (eat, pizza, with, fork)$ and $c' = (enjoy, pasta, with, fork)$ is defined as:

  $$\frac{1}{3}\alpha(eat, enjoy) + \alpha(pizza, pasta) + \alpha(fork, fork)$$

  The similarity score of identical words is assumed to be 1.

- **rank**: The rank score of the nth neighbour w' of a word w is defined as:

  $$rs(w, w') = \beta n$$

  where $0 \leq \beta \leq 1$. The rank similarity scores for the pizza example above when $\beta = 0.1$ are $rs(eat, enjoy) = 0.2$ and $rs(pizza, pasta) = 0.1$. The combined score for a PP is found by summing the rank score for each word pair and subtracting this total from one:

  $$\alpha(c, c') = 1 - \sum_{w \in v, n_1, n_2} rs(w, w')$$

  We impose a floor of zero on this score. Continuing with the pizza example, the rank similarity score between $(eat, pizza, with, fork)$ and $(enjoy, pasta, with, fork)$ is $\alpha(c, c') = 1 - 0.2 - 0.1 = 0.7$. Note that the similarity score provided by the thesaurus is used to determine the ranking but it otherwise not used.

- **single best**: Instead of smoothing using several similar contexts, we can set $\alpha(c, c') = 1$ for the closest context for which $f(c') > 0$ and ignore all others, thereby just replacing an unknown feature with a similar known one. This simplified form of smoothing may be appropriate for non-statistical models or situations where relative frequency estimates are hard to incorporate.

4 Thesauruses

As noted above, a thesaurus is a resource that groups together words that are distributionally similar. Although we refer to such resources using the singular, a thesaurus has several parts for different word categories such as nouns, verbs and adjectives.

We compare three thesauruses on this task. The first two are large-scale generic thesauruses, both constructed using the similarity metric described in (Lin, 1998b), but based on different corpora. The first, which we call **Lin**, is derived from 300 million words of newswire text and is available on the Internet[1]. The second, which we call **WASPS**, forms part of the WASPS lexicographical workbench developed at Brighton University [2] and is derived from the 100 million word BNC. The co-occurrence relations for both are a variety of grammatical relations such as direct object, subject and modifier. **WASPS** also includes prepositional phrase relations but without attempting to disambiguate them. All possible attachments are included under the assumption that correct attachments will tend to have higher frequency (Adam Kilgarriff, p.c.).

These thesauruses are designed to find words that are similar in a very general sense, and are often compared against hand-crafted semantic resources such as WordNet. However for the PP attachment task semantic similarity may be less important. We are more interested in how words behave in particular syntactic roles. For example, *eat* and *bake* are rather loosely related semantically but will be close neighbours in PP terms if they both often occur with prepositional phrase contexts such as *pizza with anchovies*.

The third thesaurus is designed to supply such specialised, task-specific neighbours. It consists of three sub-thesauruses, one for the each of the v, n_1 and n_2 words in the PP (a preposition thesaurus was also constructed with plausible-looking neighbours but was found not to be useful in practice). The co-occurrence relations used in each case consist of all possible subsets of the three remaining words together with the attachment decision. For example, given *eat pizza with fork* the following co-occurrences will be included in the thesaurus training corpus:

eat – n1-pizza,p-with,n2-fork,N
eat – n1-pizza,p-with,N
eat – n1-pizza,n2-fork,N
eat – p-with,n2-fork,N
eat – n1-pizza,N
eat – p-with,N
eat – n2-fork,N

The training corpus is created from 3.3 million prepositional phrases extracted from the British National Corpus. These PPs are identified semi-automatically using a version of the weighted GR extraction scheme described in (Carroll and Briscoe, 2001). The raw text is parsed and any PPs that occur in a large percentage of the highly ranked candidate parses are considered reliable and added to the thesaurus training corpus. Mostly these are unambiguous (v, p, n_1) or (n_1, p, n_2) triples from phrases such as *we met in January*. The dataset is rather noisy due to tagging and parsing errors, so we discarded any co-occurrence relations occurring fewer than 100 times.

We use the similarity metric described in Weeds (2003). This is a parameterised measure that can be adjusted to suit different tasks, but to ensure compatibility with the two generic thesauruses we chose parameter settings that mimic Lin's measure.

5 Experiments

For our experiments we use the Wall Street Journal dataset created by Ratnaparkhi et al. (1994). This is divided into a training set of 20,801 words, a development set of 4,039 words and a test set of 3,097 words. Each word was reduced to its morphological root using the morphological analyser described in (Minnen et al., 2000). Strings of four digits beginning with a 1 or 2 are replaced with *YEAR* and all other digit strings including those including commas and full stops were replaced with *NUM*. Our implementation of Collins' algorithm only achieves 84.3% on the test data, with the shortfall of 0.2% primarily due to the different morphological analysers used[3]

5.1 Smoothing

Firstly we compare the different PP similarity functions. Figure 2 shows the accuracy of each as a function of k, the number of examples in $S(c)$. The **WASPS** thesaurus was used in all cases. The best smoothed model is **rank** with 85.1% accuracy when $\beta = 0.05$ and $k = 15$. The accuracy of **rank** with the smallest β value drops off rapidly when $k > 10$, showing that neighbours beyond this point are providing unreliable evidence and should be discounted more aggressively. More interestingly, this problem also affects **average**, suggesting that the similarity scores provided by the thesaurus are also misleadingly high for less similar words. The same effect was also observed when we used the harmonic mean of all similarity scores, so it is unlikely that the problem is an artifact of the averaging operation.

On the other hand, if β is set quite low (for example

[1] http://www.cs.ualberta.ca/ lindek/downloads.htm

[2] http://wasps.itri.brighton.ac.uk

[3] This result is interesting since this analyser is more accurate than the one used by Collins. We chose to use this analyser because it matches the word forms in the thesauruses better.

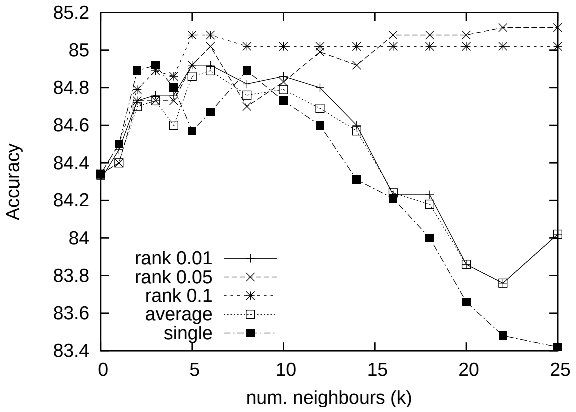

Figure 2: Accuracy for different smoothing functions on the development set plotted against k, the number of similar words used for smoothing

$\beta = 0.01$) then accuracy levels off very quickly as less similar neighbours are assigned zero frequency. The middle value of $\beta = 0.05$ appears to offer a good trade-off. Regardless of the similarity function we can see that relatively small values for k are sufficient, which is good news for efficiency reasons (each attachment decision is an $O(k)$ operation).

Figure 3 shows the combined coverage of the triple and quadruple features in Collins' model, which are the only smoothed features in our model. For example, almost 75% of attachment decisions are resolved by 3- or 4-tuples using the **average** function and setting $k = 25$. Again, **average** is comparable to **rank** with $\beta = 0.01$. Table 1 compares the accuracy of the smoothed and unsmoothed models at each backing off stage. Smoothing has a negative effect on accuracy, but this is made for by an increase in accuracy.

The reduction in the error rate with the **single best** policy on the development set is somewhat less than with the smoothed frequency models, and the results more error-prone and sensitive to the choice of k. These models are more likely to be unlucky with a choice of feature than with the smoothed frequencies. As noted above, this technique may still be useful for algorithms which cannot easily incorporate smoothed frequency estimates.

	Smoothed		Unsm.	
Stage	Acc.	Cov.	Acc.	Cov.
1	90.9	12.4	91.2	8.5
2	87.3	49.7	87.5	33.5
3	80.8	34.2	82.1	54.2
4	73.4	3.6	73.9	3.7

Table 1: Accuracy and coverage of the first two backing off stages on the development data. The smoothed model uses WASPS with $\beta = 0.5$ and $k = 5$.

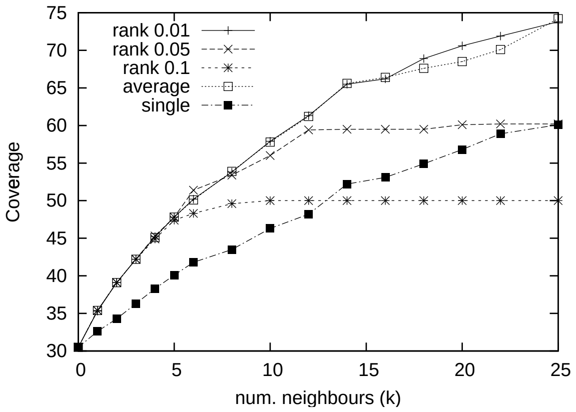

Figure 3: Coverage for different smoothing functions against the number of neighbours used for smoothing

5.2 Thesauruses

A thesaurus providing better neighbours should do better on this task. Figure 4 shows the accuracy of the three thesauruses using **rank** smoothing and $\beta = 0.05$ on the development data. Final results using $k = 5$ and $\beta = 0.05$ on the data is shown in Table 2, together with the size of the noun sections of each thesaurus (the direct object thesaurus in the case of **specialist**) and coverage of 3- and 4-tuples.

Clearly both generic thesauruses consistently outperform the specialist thesaurus. The latter tends to produce neighbours with have less obvious semantic similarity, for example providing *pour* as the first neighbour of *fetch*. We hypothesised that using syntactic rather than semantic neighbours could be desirable, but in this case it often generates contexts that are unlikely to occur: *pour price of profit* as a neighbour of *fetch price of profit*, for example. Although this may be a flaw in the approach, we may simply be using too few contexts to create a reliable thesaurus. Previous research has found that using more data leads to better quality thesauruses (Curran and Moens, 2002). We are also conflating attachment preferences, since a word must appear with similar contexts in both noun and verb modifying PPs to achieve a high sim-

Thesaurus	Acc.	Size (N)	Cov.
None	84.30	-	30.5
Lin	85.02	13,850	72.1
WASPS	85.05	17,843	60.1
Specialist	84.50	5,669	61.0

Table 2: Accuracy on the test data using $\beta = 0.05$ and $k = 5$; the size of the noun section of each thesaurus, and coverage of smoothed 4- and 3-tuples

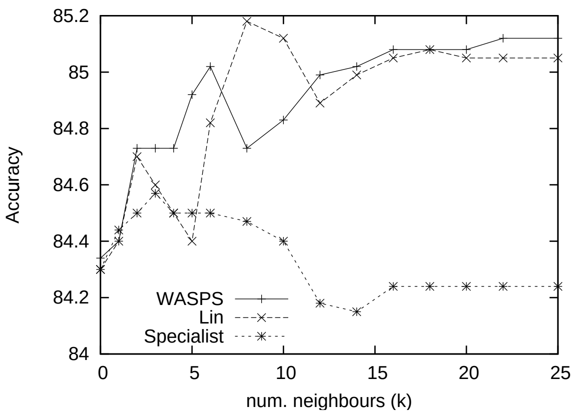

Figure 4: Accuracy of the three different thesauruses on the development set using rank smoothing with $\beta = 0.05$

Method	Accuracy	WN?
Zavrel et. al. (1997)	84.1	No
WASPS	85.1	No
Li & Abe (1998)	85.2	Yes
Stetina & Nagao (1997)	88.1	Yes

Table 3: Accuracy of various attachment models using WordNet or automatic clustering algorithms

ilarity score. There may be merit in creating separate thesauruses for noun-attachment and verb-attachment, since there may be words that are strongly similar in only one of these cases.

Interestingly, although **Lin** is smaller than **WASPS** it has better coverage. This is most likely due to the different corpora used to construct each thesaurus. **Lin** is built using newswire text which is closer in genre to the Wall Street Journal. For example, the first neighbour for *fetch* in **WASPS** is *grab*, but none of the top 25 neighbours of this word in **Lin** have this sporting sense. Both **WASPS** and **specialist** are derived from the BNC and have similar coverage, although the quality of **specialist** neighbours is not as good.

The **WASPS** and **Lin** models produce statistically significant ($P < 0.05$) improvements over the vanilla Collins model using a paired t-test with 10-fold cross-validation on the entire dataset[4]. The **specialist** model is not significantly better. Table 3 compares our results with other comparable PP attachment models.

On the face of it, these are not resounding improvements over the baseline, but this is a very hard task. Ratnaparkhi (1994) established a human upper bound of 88.2% but subsequent research has put this as low as 78.3% (Mitchell, 2003). At least two thirds of the remaining errors are therefore likely to be very difficult.

An inspection of the data shows that many of the remaining errors are due to poor neighbouring PPs being used for smoothing. For example, the PP in *entrust company with cash* modifies the verb, but no matching quadruples are present in the training data. The only matching (n_1, p, n_2) triple using **WASPS** is (*industry, for, income*), which appears twice in the training data modifying the noun. The model therefore guesses incorrectly even though the thesaurus is providing what appear to be semantically appropriate neighbours. Another example is *attend meeting with representative*, where the (v, p, n_2) triple (*talk, with, official*) convinces the model to incorrectly guess verb attachment.

Part of the problem is that words in the PP are replaced independently and without consideration to the remaining context. However we had hoped the specialist thesaurus might alleviate this problem by providing neighbours that are more appropriate for this specific task. Finding good neighbours for verbs is clearly more difficult than for nouns since subcategorisation and selectional preferences also play a role.

6 Conclusion

Our results show that the similarity-based smoothing of frequency estimates significantly improves an already respectable probabilistic PP attachment model. However our hypothesis that a task-specific thesaurus would outperform a generic thesaurus was not borne out by our experiments. The neighbours provided by the specialist thesaurus are not as informative as those supplied by the generic thesauruses. Of course, this negative result is naturally good news for developers of generic thesauruses.

We described ways of finding and scoring distributionally similar PPs. A significant number of errors in the final model can be traced to the way individual words in the PP are replaced without regard to the wider context, producing neighbouring PPs that have conflicting attachment preferences. The specialist thesaurus was not able to overcome this problem. A second finding is that distributional similarity scores provided by all thesauruses weight dissimilar neighbours too highly, and more aggressive weighting schemes are better for smoothing.

Our aim is to apply similarity-based smoothing with both generic and specialist thesauruses to other areas in lexicalised parse selection, particularly other overtly lexical problems such as noun-noun modifiers and conjunction scope. Lexical information has a lot of promise for parse selection in theory, but there are practical problems such as sparse data and genre effects (Gildea, 2001). Appropriately trained thesauruses and similarity-based techniques should help to alleviate both problems.

[4] The Collins model achieves 84.50±1.0% accuracy and the smoothed model 84.90±1.0% accuracy by this measure.

Acknowledgements

Many thanks to Julie Weeds and Adam Kilgarriff for providing the specialist and WASPS thesauruses, and for useful discussions. Thanks also to the anonymous reviewers for many helpful comments.

References

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

Edward Briscoe and John Carroll. 1995. Developing and evaluating a probabilistic lr parser of part-of-speech and punctuation labels. In *Proceedings of the IWPT '95*, pages 48–58.

Peter F. Brown, Vincent J. Della Pietra, Peter V. de Souza, Jenifer C. Lai, and Robert L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):468–479.

John Carroll and Ted Briscoe. 2001. High precision extraction of grammatical relations. In *Proceedings of the IWPT '01*.

Stanley Chen and Joshua Goodman. 1996. An empirical study of smoothing techniques for language modelling. In *Proceedings of ACL '96*, pages 310–318.

Michael Collins and James Brooks. 1995. Prepositional phrase attachment through a backed-off model. In *Proceedings of EMNLP '95*, pages 27–38.

James Curran and Mark Moens. 2002. Scaling context space. In *Proceedings of the ACL '02*, pages 222–229.

Ido Dagan, Lillian Lee, and Fernando Pereira. 1997. Similarity-based methods for word sense disambiguation. In *Proceedings of ACL '97*, pages 56–63.

Ido Dagan, Lillian Lee, and Fernando Pereira. 1999. Similarity-based models of word cooccurrence probabilities. *Machine Learning*, 34(1-3):43–69.

L. Frazier. 1979. *On comprehending sentences: Synactic parsing strategies*. Ph.D. thesis, University of Connecticut.

Daniel Gildea. 2001. Corpus variation and parser performance. In *Proceedings of EMNLP '01*, Pittsburgh, PA.

Ralph Grishman and John Sterling. 1994. Generalizing automatically generated selectional patterns. In *Proceedings of COLING '94*, pages 742–747.

Don Hindle and Mats Rooth. 1993. Structural ambiguity and lexical relations. *Computational Linguistics*, 19:103–120.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalised parsing. In *Proceedings of ACL '03*.

Hang Li and Naoki Abe. 1998. Word clustering and disambiguation based on co-occurrence data. In *Proceedings of COLING '98*, pages 749–755.

Dekang Lin. 1998a. Dependency-based evaluation of MINIPAR. In *Workshop on the Evaluation of Parsing Systems*.

Dekang Lin. 1998b. An information-theoretic measure of similarity. In *Proceedings of ICML '98*, pages 296–304.

Guido Minnen, John Carroll, and Darren Pearce. 2000. Robust, applied morphological generation. In *Proceedings of INLG 2000*, pages 201–208.

Brian Mitchell. 2003. *Prepositional phrase attachment using machine learning algorithms*. Ph.D. thesis, University of Sheffield.

Ted Pedersen and Rebecca Bruce. 1997. Distinguishing word senses in untagged text. In *Proceedings of EMNLP '97*.

Adwait Ratnaparkhi, Jeff Reynar, and Salim Roukos. 1994. A maximum entropy model for prepositional phrase attachment. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 250–255.

Jiri Stetina and Makoto Nagao. 1997. Corpus based PP attachment ambiguity resolution with a semantic dictionary. In *Proceedings of WVLC '97*, pages 66–80.

Tokunaga Takenobu, Iwayama Makoto, and Tanaka Hozumi. 1995. Automatic thesaurus construction based on grammatical relations. In *Proceedings of IJCAI '95*, pages 1308–1313.

Julie Weeds. 2003. A general framework for distributional similarity. In *Proceedings of the EMNLP '03*.

G. Whittemore, K. Ferrara, and H. Brunner. 1990. Empirical study of predictive powers of simple attachment schemes for post-modifier prepositional phrases. In *Proceedings of ACL '90*, pages 23–30.

Jakub Zavrel, Walter Daelemans, and Jorn Veenstra. 1997. Resolving PP attachment ambiguities with memory-based learning. In *Proceedings of CoNLL '97*, pages 136–144.