Corpus Statistics Meet the Noun Compound: Some Empirical Results

Mark Lauer
Microsoft Institute
65 Epping Road,
North Ryde NSW 2113
Australia
t-markl@microsoft.com

Abstract

A variety of statistical methods for noun compound analysis are implemented and compared. The results support two main conclusions. First, the use of conceptual association not only enables a broad coverage, but also improves the accuracy. Second, an analysis model based on dependency grammar is substantially more accurate than one based on deepest constituents, even though the latter is more prevalent in the literature.

1 Background

1.1 Compound Nouns

If parsing is taken to be the first step in taming the natural language understanding task, then broad coverage NLP remains a jungle inhabited by wild beasts. For instance, parsing noun compounds appears to require detailed world knowledge that is unavailable outside a limited domain (Sparck Jones, 1983). Yet, far from being an obscure, endangered species, the noun compound is flourishing in modern language. It has already made five appearances in this paragraph and at least one diachronic study shows a veritable population explosion (Leonard, 1984). While substantial work on noun compounds exists in both linguistics (e.g. Levi, 1978; Ryder, 1994) and computational linguistics (Finin, 1980; McDonald, 1982; Isabelle, 1984), techniques suitable for broad coverage parsing remain unavailable. This paper explores the application of corpus statistics (Charniak, 1993) to noun compound parsing (other computational problems are addressed in Arens *et al*, 1987; Vanderwende, 1993 and Sproat, 1994).

The task is illustrated in example 1:

Example 1

(a) [woman$_N$ [aid$_N$ worker$_N$]]
(b) [[hydrogen$_N$ ion$_N$] exchange$_N$]

The parses assigned to these two compounds differ, even though the sequence of parts of speech are identical. The problem is analogous to the prepositional phrase attachment task explored in Hindle and Rooth (1993). The approach they propose involves computing lexical associations from a corpus and using these to select the correct parse. A similar architecture may be applied to noun compounds.

In the experiments below the accuracy of such a system is measured. Comparisons are made across five dimensions:

- Each of two analysis models are applied: adjacency and dependency.
- Each of a range of training schemes are employed.
- Results are computed with and without tuning factors suggested in the literature.
- Each of two parameterisations are used: associations between words and associations between concepts.
- Results are collected with and without machine tagging of the corpus.

1.2 Training Schemes

While Hindle and Rooth (1993) use a partial parser to acquire training data, such machinery appears unnecessary for noun compounds. Brent (1993) has proposed the use of simple word patterns for the acquisition of verb subcategorisation information. An analogous approach to compounds is used in Lauer (1994) and constitutes one scheme evaluated below. While such patterns produce false training examples, the resulting noise often only introduces minor distortions.

A more liberal alternative is the use of a co-occurrence window. Yarowsky (1992) uses a fixed 100 word window to collect information used for sense disambiguation. Similarly, Smadja (1993) uses a six content word window to extract significant collocations. A range of windowed training schemes are employed below. Importantly, the use of a window provides a natural means of trading off the amount of data against its quality. When data sparseness undermines the system accuracy, a wider window may

admit a sufficient volume of extra accurate data to outweigh the additional noise.

1.3 Noun Compound Analysis

There are at least four existing corpus-based algorithms proposed for syntactically analysing noun compounds. Only two of these have been subjected to evaluation, and in each case, no comparison to any of the other three was performed. In fact all authors appear unaware of the other three proposals. I will therefore briefly describe these algorithms.

Three of the algorithms use what I will call the ADJACENCY MODEL, an analysis procedure that goes back to Marcus (1980, p253). Therein, the procedure is stated in terms of calls to an oracle which can determine if a noun compound is acceptable. It is reproduced here for reference:

Given three nouns n_1, n_2 and n_3:

- If either $[n_1\ n_2]$ or $[n_2\ n_3]$ is not semantically acceptable then build the alternative structure;
- otherwise, if $[n_2\ n_3]$ is semantically preferable to $[n_1\ n_2]$ then build $[n_2\ n_3]$;
- otherwise, build $[n_1\ n_2]$.

Only more recently has it been suggested that corpus statistics might provide the oracle, and this idea is the basis of the three algorithms which use the adjacency model. The simplest of these is reported in Pustejovsky *et al* (1993). Given a three word compound, a search is conducted elsewhere in the corpus for each of the two possible subcomponents. Whichever is found is then chosen as the more closely bracketed pair. For example, when *backup compiler disk* is encountered, the analysis will be:

Example 2

(a) [backup$_N$ [compiler$_N$ disk$_N$]]
when *compiler disk* appears elsewhere
(b) [[backup$_N$ compiler$_N$] disk$_N$]
when *backup compiler* appears elsewhere

Since this is proposed merely as a rough heuristic, it is not stated what the outcome is to be if neither or both subcomponents appear. Nor is there any evaluation of the algorithm.

The proposal of Liberman and Sproat (1992) is more sophisticated and allows for the frequency of the words in the compound. Their proposal involves comparing the mutual information between the two pairs of adjacent words and bracketing together whichever pair exhibits the highest. Again, there is no evaluation of the method other than a demonstration that four examples work correctly.

The third proposal based on the adjacency model appears in Resnik (1993) and is rather more complex again. The SELECTIONAL ASSOCIATION between a predicate and a word is defined based on the contribution of the word to the conditional entropy of the predicate. The association between each pair of words in the compound is then computed by taking the maximum selectional association from all possible ways of regarding the pair as predicate and argument. Whilst this association metric is complicated, the decision procedure still follows the outline devised by Marcus (1980) above. Resnik (1993) used unambiguous noun compounds from the parsed *Wall Street Journal* (WSJ) corpus to estimate the association values and analysed a test set of around 160 compounds. After some tuning, the accuracy was about 73%, as compared with a baseline of 64% achieved by always bracketing the first two nouns together.

The fourth algorithm, first described in Lauer (1994), differs in one striking manner from the other three. It uses what I will call the DEPENDENCY MODEL. This model utilises the following procedure when given three nouns n_1, n_2 and n_3:

- Determine how acceptable the structures $[n_1\ n_2]$ and $[n_1\ n_3]$ are;
- if the latter is more acceptable, build $[n_2\ n_3]$ first;
- otherwise, build $[n_1\ n_2]$ first.

Figure 1 shows a graphical comparison of the two analysis models.

In Lauer (1994), the degree of acceptability is again provided by statistical measures over a corpus. The metric used is a mutual information-like measure based on probabilities of modification relationships. This is derived from the idea that parse trees capture the structure of semantic relationships within a noun compound.[1]

The dependency model attempts to choose a parse which makes the resulting relationships as acceptable as possible. For example, when *backup compiler disk* is encountered, the analysis will be:

Example 3

(a) [backup$_N$ [compiler$_N$ disk$_N$]]
when *backup disk* is more acceptable
(b) [[backup$_N$ compiler$_N$] disk$_N$]
when *backup compiler* is more acceptable

I claim that the dependency model makes more intuitive sense for the following reason. Consider the compound *calcium ion exchange*, which is typically left-branching (that is, the first two words are bracketed together). There does not seem to be any reason why *calcium ion* should be any more frequent than *ion exchange*. Both are plausible compounds and regardless of the bracketing, *ions* are the object of an *exchange*. Instead, the correct parse depends on whether *calcium* characterises the *ions* or mediates the *exchange*.

Another significant difference between the models is the predictions they make about the proportion

[1] Lauer and Dras (1994) give a formal construction motivating the algorithm given in Lauer (1994).

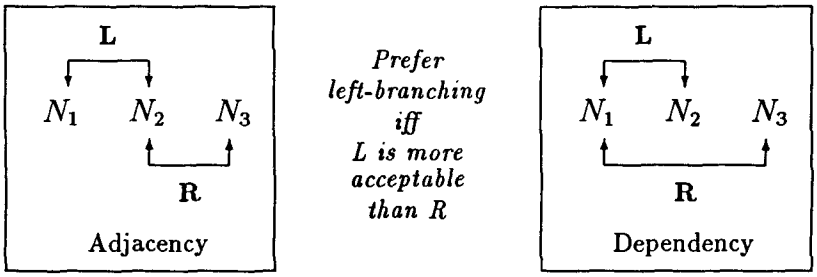


Figure 1: Two analysis models and the associations they compare

of left and right-branching compounds. Lauer and Dras (1994) show that under a dependency model, left-branching compounds should occur twice as often as right-branching compounds (that is two-thirds of the time). In the test set used here and in that of Resnik (1993), the proportion of left-branching compounds is 67% and 64% respectively. In contrast, the adjacency model appears to predict a proportion of 50%.

The dependency model has also been proposed by Kobayasi *et al* (1994) for analysing Japanese noun compounds, apparently independently. Using a corpus to acquire associations, they bracket sequences of Kanji with lengths four to six (roughly equivalent to two or three words). A simple calculation shows that using their own preprocessing hueristics to guess a bracketing provides a higher accuracy on their test set than their statistical model does. This renders their experiment inconclusive.

2 Method

2.1 Extracting a Test Set

A test set of syntactically ambiguous noun compounds was extracted from our 8 million word Grolier's encyclopedia corpus in the following way.[2] Because the corpus is not tagged or parsed, a somewhat conservative strategy of looking for unambiguous sequences of nouns was used. To distinguish nouns from other words, the University of Pennsylvania morphological analyser (described in Karp *et al*, 1992) was used to generate the set of words that can only be used as nouns (I shall henceforth call this set $\mathcal{N}$). All consecutive sequences of these words were extracted, and the three word sequences used to form the test set. For reasons made clear below, only sequences consisting entirely of words from Roget's thesaurus were retained, giving a total of 308 test triples.[3]

These triples were manually analysed using as context the entire article in which they appeared. In some cases, the sequence was not a noun compound (nouns can appear adjacent to one another across various constituent boundaries) and was marked as an error. Other compounds exhibited what Hindle and Rooth (1993) have termed SEMANTIC INDETERMINACY where the two possible bracketings cannot be distinguished in the context. The remaining compounds were assigned either a left-branching or right-branching analysis. Table 1 shows the number of each kind and an example of each.

Accuracy figures in all the results reported below were computed using only those 244 compounds which received a parse.

2.2 Conceptual Association

One problem with applying lexical association to noun compounds is the enormous number of parameters required, one for every possible pair of nouns. Not only does this require a vast amount of memory space, it creates a severe data sparseness problem since we require at least some data about each parameter. Resnik and Hearst (1993) coined the term CONCEPTUAL ASSOCIATION to refer to association values computed between groups of words. By assuming that all words within a group behave similarly, the parameter space can be built in terms of the groups rather than in terms of the words.

In this study, conceptual association is used with groups consisting of all categories from the 1911 version of Roget's thesaurus.[4] Given two thesaurus categories t_1 and t_2, there is a parameter which represents the degree of acceptability of the structure $[n_1 n_2]$ where n_1 is a noun appearing in t_1 and n_2 appears in t_2. By the assumption that words within a group behave similarly, this is constant given the two categories. Following Lauer and Dras (1994) we can formally write this parameter as $Pr(t_1 \rightarrow t_2)$ where the event $t_1 \rightarrow t_2$ denotes the modification of a noun in t_2 by a noun in t_1.

2.3 Training

To ensure that the test set is disjoint from the training data, all occurrences of the test noun compounds have been removed from the training corpus.

[2]We would like to thank Grolier's for permission to use this material for research purposes.

[3]The 1911 version of Roget's used is available on-line and is in the public domain.

[4]It contains 1043 categories.

Type	Number	Proportion	Example
Error	29	9%	In *monsoon regions rainfall* does not ...
Indeterminate	35	11%	Most advanced aircraft have *precision navigation systems.*
Left-branching	163	53%	...escaped punishment by the Allied *war crimes tribunals.*
Right-branching	81	26%	Ronald Reagan, who won two *landslide election victories*, ...

Table 1: Test set distribution

Two types of training scheme are explored in this study, both unsupervised. The first employs a pattern that follows Pustejovsky (1993) in counting the occurrences of subcomponents. A training instance is any sequence of four words $w_1w_2w_3w_4$ where $w_1, w_4 \notin \mathcal{N}$ and $w_2, w_3 \in \mathcal{N}$. Let $\text{count}_p(n_1, n_2)$ be the number of times a sequence $w_1n_1n_2w_4$ occurs in the training corpus with $w_1, w_4 \notin \mathcal{N}$.

The second type uses a window to collect training instances by observing how often a pair of nouns co-occur within some fixed number of words. In this study, a variety of window sizes are used. For $n \geq 2$, let $\text{count}_n(n_1, n_2)$ be the number of times a sequence $n_1w_1 \ldots w_in_2$ occurs in the training corpus where $i \leq n-2$. Note that windowed counts are asymmetric. In the case of a window two words wide, this yields the mutual information metric proposed by Liberman and Sproat (1992).

Using each of these different training schemes to arrive at appropriate counts it is then possible to estimate the parameters. Since these are expressed in terms of categories rather than words, it is necessary to combine the counts of words to arrive at estimates. In all cases the estimates used are:

$$\Pr(t_1 \rightarrow t_2) = \frac{1}{\eta} \sum_{\substack{w_1 \in t_1 \\ w_2 \in t_2}} \frac{\text{count}(w_1, w_2)}{\text{ambig}(w_1)\text{ambig}(w_2)}$$

$$\text{where } \eta = \sum_{\substack{w_1 \in \mathcal{N} \\ w_2 \in t_2}} \frac{\text{count}(w_1, w_2)}{\text{ambig}(w_1)\text{ambig}(w_2)}$$

Here $\text{ambig}(w)$ is the number of categories in which w appears. It has the effect of dividing the evidence from a training instance across all possible categories for the words. The normaliser ensures that all parameters for a head noun sum to unity.

2.4 Analysing the Test Set

Given the high level descriptions in section 1.3 it remains only to formalise the decision process used to analyse a noun compound. Each test compound presents a set of possible analyses and the goal is to choose which analysis is most likely. For three word compounds it suffices to compute the ratio of two probabilities, that of a left-branching analysis and that of a right-branching one. If this ratio is greater than unity, then the left-branching analysis is chosen. When it is less than unity, a right-branching analysis is chosen.[5] If the ratio is exactly unity, the analyser guesses left-branching, although this is fairly rare for conceptual association as shown by the experimental results below.

For the adjacency model, when the given compound is $w_1w_2w_3$, we can estimate this ratio as:

$$R_{adj} = \frac{\sum_{t_i \in \text{cats}(w_i)} \Pr(t_1 \rightarrow t_2)}{\sum_{t_i \in \text{cats}(w_i)} \Pr(t_2 \rightarrow t_3)} \quad (1)$$

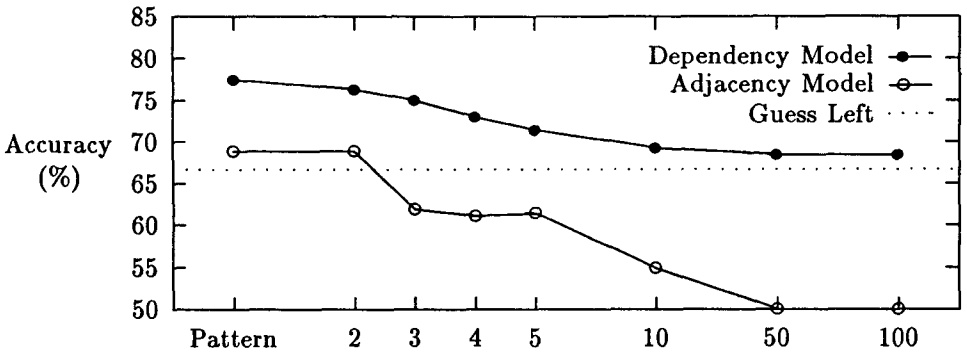
For the dependency model, the ratio is:

$$R_{dep} = \frac{\sum_{t_i \in \text{cats}(w_i)} \Pr(t_1 \rightarrow t_2)\Pr(t_2 \rightarrow t_3)}{\sum_{t_i \in \text{cats}(w_i)} \Pr(t_1 \rightarrow t_3)\Pr(t_2 \rightarrow t_3)} \quad (2)$$

In both cases, we sum over all possible categories for the words in the compound. Because the dependency model equations have two factors, they are affected more severely by data sparseness. If the probability estimate for $\Pr(t_2 \rightarrow t_3)$ is zero for all possible categories t_2 and t_3 then both the numerator and the denominator will be zero. This will conceal any preference given by the parameters involving t_1. In such cases, we observe that the test instance itself provides the information that the event $t_2 \rightarrow t_3$ can occur and we recalculate the ratio using $\Pr(t_2 \rightarrow t_3) = k$ for all possible categories t_2, t_3 where k is any non-zero constant. However, no correction is made to the probability estimates for $\Pr(t_1 \rightarrow t_2)$ and $\Pr(t_1 \rightarrow t_3)$ for unseen cases, thus putting the dependency model on an equal footing with the adjacency model above.

The equations presented above for the dependency model differ from those developed in Lauer and Dras (1994) in one way. There, an additional weighting factor (of 2.0) is used to favour a left-branching analysis. This arises because their construction is based on the dependency model which predicts that left-branching analyses should occur twice as often. Also, the work reported in Lauer and Dras (1994) uses simplistic estimates of the probability of a word given its thesaurus category. The equations above assume these probabilities are uniformly constant. Section 3.2 below shows the result of making these two additions to the method.

[5] If either probability estimate is zero, the other analysis is chosen. If both are zero the analysis is made as if the ratio were exactly unity.



Training scheme (integers denote window widths)

Figure 2: Accuracy of dependency and adjacency model for various training schemes

3 Results

3.1 Dependency meets Adjacency

Eight different training schemes have been used to estimate the parameters and each set of estimates used to analyse the test set under both the adjacency and the dependency model. The schemes used are:

- the pattern given in section 2.3; and
- windowed training schemes with window widths of 2, 3, 4, 5, 10, 50 and 100 words.

The accuracy on the test set for all these experiments is shown in figure 2. As can be seen, the dependency model is more accurate than the adjacency model. This is true across the whole spectrum of training schemes. The proportion of cases in which the procedure was forced to guess, either because no data supported either analysis or because both were equally supported, is quite low. For the pattern and two-word window training schemes, the guess rate is less than 4% for both models. In the three-word window training scheme, the guess rates are less than 1%. For all larger windows, neither model is ever forced to guess.

In the case of the pattern training scheme, the difference between 68.9% for adjacency and 77.5% for dependency is statistically significant at the 5% level ($p = 0.0316$), demonstrating the superiority of the dependency model, at least for the compounds within Grolier's encyclopedia.

In no case do any of the windowed training schemes outperform the pattern scheme. It seems that additional instances admitted by the windowed schemes are too noisy to make an improvement.

Initial results from applying these methods to the EMA corpus have been obtained by Wilco ter Stal (1995), and support the conclusion that the dependency model is superior to the adjacency model.

3.2 Tuning

Lauer and Dras (1994) suggest two improvements to the method used above. These are:

- a factor favouring left-branching which arises from the formal dependency construction; and
- factors allowing for naive estimates of the variation in the probability of categories.

While these changes are motivated by the dependency model, I have also applied them to the adjacency model for comparison. To implement them, equations 1 and 2 must be modified to incorporate a factor of $\frac{1}{|t_1||t_2||t_3|}$ in each term of the sum and the entire ratio must be multiplied by two. Five training schemes have been applied with these extensions. The accuracy results are shown in figure 3. For comparison, the untuned accuracy figures are shown with dotted lines. A marked improvement is observed for the adjacency model, while the dependency model is only slightly improved.

3.3 Lexical Association

To determine the difference made by conceptual association, the pattern training scheme has been retrained using lexical counts for both the dependency and adjacency model, but only for the words in the test set. If the same system were to be applied across all of $\mathcal{N}$ (a total of 90,000 nouns), then around 8.1 billion parameters would be required. Left-branching is favoured by a factor of two as described in the previous section, but no estimates for the category probabilities are used (these being meaningless for the lexical association method).

Accuracy and guess rates are shown in figure 4. Conceptual association outperforms lexical association, presumably because of its ability to generalise.

3.4 Using a Tagger

One problem with the training methods given in section 2.3 is the restriction of training data to nouns in $\mathcal{N}$. Many nouns, especially common ones, have verbal or adjectival usages that preclude them from being in $\mathcal{N}$. Yet when they occur as nouns, they still provide useful training information that the current system ignores. To test whether using tagged

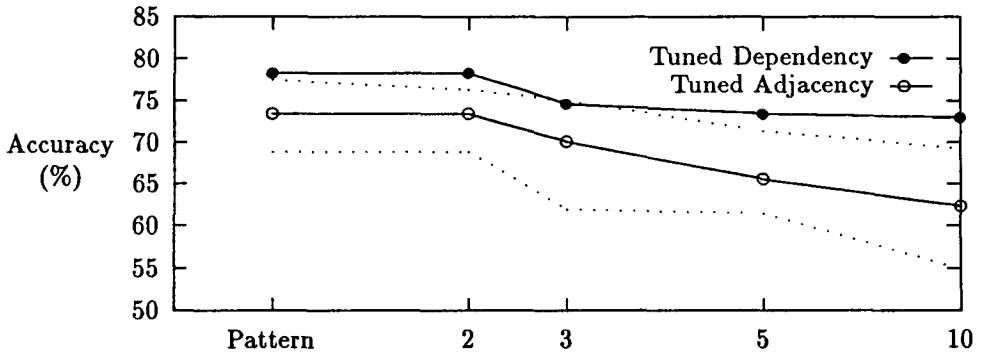


Figure 3: Accuracy of tuned dependency and adjacency model for various training schemes

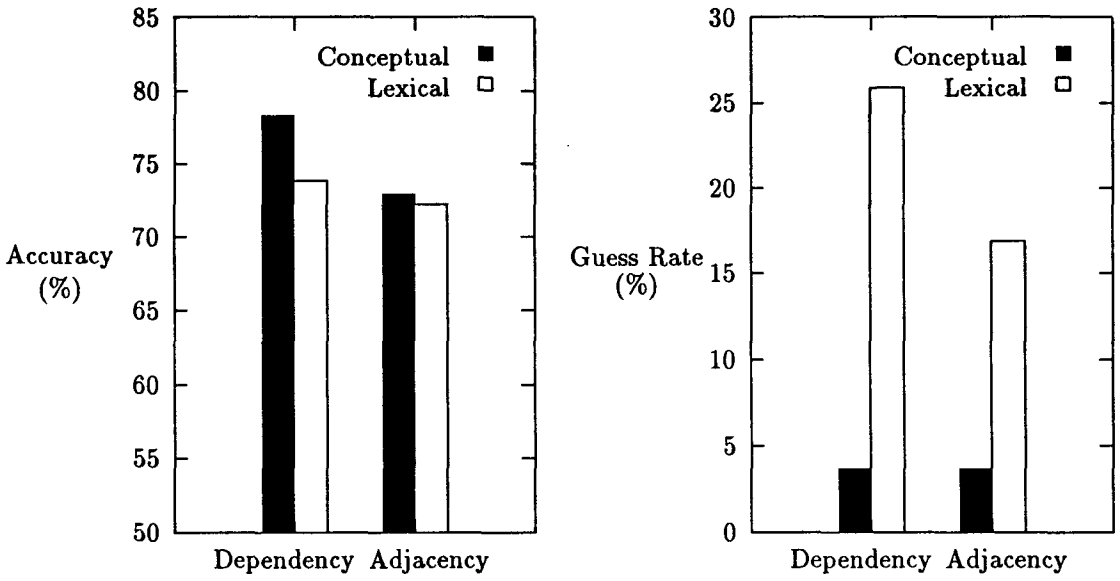


Figure 4: Accuracy and Guess Rates of Lexical and Conceptual Association

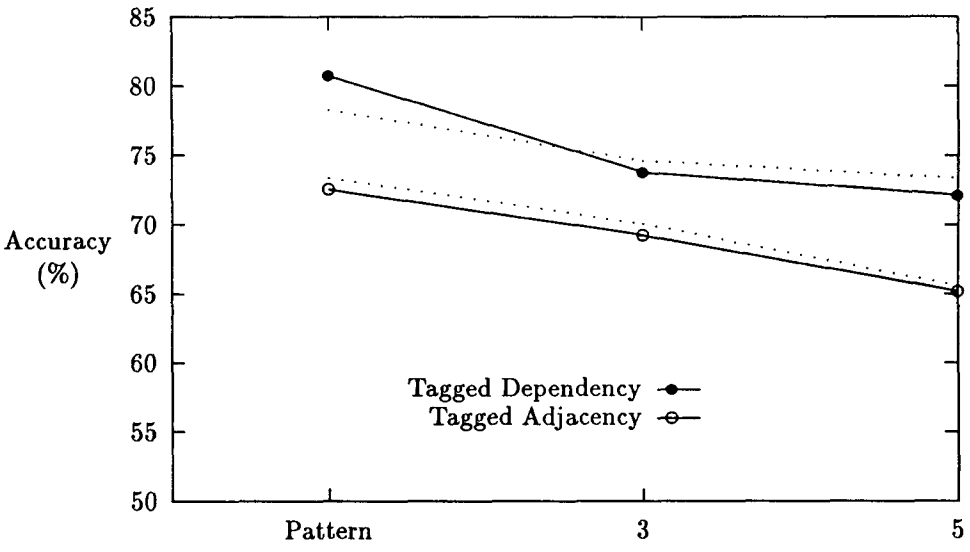


Figure 5: Accuracy using a tagged corpus for various training schemes

data would make a difference, the freely available Brill tagger (Brill, 1993) was applied to the corpus. Since no manually tagged training data is available for our corpus, the tagger's default rules were used (these rules were produced by Brill by training on the Brown corpus). This results in rather poor tagging accuracy, so it is quite possible that a manually tagged corpus would produce better results.

Three training schemes have been used and the tuned analysis procedures applied to the test set. Figure 5 shows the resulting accuracy, with accuracy values from figure 3 displayed with dotted lines. If anything, admitting additional training data based on the tagger introduces more noise, reducing the accuracy. However, for the pattern training scheme an improvement was made to the dependency model, producing the highest overall accuracy of 81%.

4 Conclusion

The experiments above demonstrate a number of important points. The most general of these is that even quite crude corpus statistics can provide information about the syntax of compound nouns. At the very least, this information can be applied in broad coverage parsing to assist in the control of search. I have also shown that with a corpus of moderate size it is possible to get reasonable results without using a tagger or parser by employing a customised training pattern. While using windowed co-occurrence did not help here, it is possible that under more data sparse conditions better performance could be achieved by this method.

The significance of the use of conceptual association deserves some mention. I have argued that without it a broad coverage system would be impossible. This is in contrast to previous work on conceptual association where it resulted in little improvement on a task which could already be performed. In this study, not only has the technique proved its worth by supporting generality, but through generalisation of training information it outperforms the equivalent lexical association approach given the same information.

Amongst all the comparisons performed in these experiments one stands out as exhibiting the greatest contrast. In all experiments the dependency model provides a substantial advantage over the adjacency model, even though the latter is more prevalent in proposals within the literature. This result is in accordance with the informal reasoning given in section 1.3. The model also has the further commendation that it predicts correctly the observed proportion of left-branching compounds found in two independently extracted test sets.

In all, the most accurate technique achieved an accuracy of 81% as compared to the 67% achieved by guessing left-branching. Given the high frequency of occurrence of noun compounds in many texts, this suggests that the use of these techniques in probabilistic parsers will result in higher performance in broad coverage natural language processing.

5 Acknowledgements

This work has received valuable input from people too numerous to mention. The most significant contributions have been made by Richard Buckland, Robert Dale and Mark Dras. I am also indebted to Vance Gledhill, Mike Johnson, Philip Resnik, Richard Sproat, Wilco ter Stal, Lucy Vanderwende and Wayne Wobcke. Financial support is gratefully ack-

nowledged from the Microsoft Institute and the Australian Government.

References

Arens, Y., Granacki, J. and Parker, A. 1987. Phrasal Analysis of Long Noun Sequences. In *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, Stanford, CA. pp59-64.

Brent, Michael. 1993. From Grammar to Lexicon: Unsupervised Learning of Lexical Syntax. In *Computational Linguistics*, **Vol 19(2)**, Special Issue on Using Large Corpora II, pp243-62.

Brill, Eric. 1993. *A Corpus-based Approach to Language Learning.* PhD Thesis, University of Pennsylvania, Philadelphia, PA..

Charniak, Eugene. 1993. *Statistical Language Learning.* MIT Press, Cambridge, MA.

Finin, Tim. 1980. *The Semantic Interpretation of Compound Nominals.* PhD Thesis, Co-ordinated Science Laboratory, University of Illinois, Urbana, IL.

Hindle, D. and Rooth, M. 1993. Structural Ambiguity and Lexical Relations. In *Computational Linguistics* **Vol. 19(1)**, Special Issue on Using Large Corpora I, pp103-20.

Isabelle, Pierre. 1984. Another Look At Nominal Compounds. In *Proceedings of COLING-84*, Stanford, CA. pp509-16.

Karp, D., Schabes, Y., Zaidel, M. and Egedi, D. 1992. A Freely Available Wide Coverage Morphological Analyzer for English. In *Proceedings of COLING-92*, Nantes, France, pp950-4.

Kobayasi, Y., Tokunaga, T. and Tanaka, H. 1994. Analysis of Japanese Compound Nouns using Collocational Information. In *Proceedings of COLING-94*, Kyoto, Japan, pp865-9.

Lauer, Mark. 1994. Conceptual Association for Compound Noun Analysis. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, Student Session, Las Cruces, NM. pp337-9.

Lauer, M. and Dras, M. 1994. A Probabilistic Model of Compound Nouns. In *Proceedings of the 7th Australian Joint Conference on Artificial Intelligence*, Armidale, NSW, Australia. World Scientific Press, pp474-81.

Leonard, Rosemary. 1984. *The Interpretation of English Noun Sequences on the Computer.* North-Holland, Amsterdam.

Levi, Judith. 1978. *The Syntax and Semantics of Complex Nominals.* Academic Press, New York.

Liberman, M. and Sproat, R. 1992. The Stress and Structure of Modified Noun Phrases in English. In Sag, I. and Szabolcsi, A., editors, *Lexical Matters CSLI Lecture Notes* **No. 24**. University of Chicago Press, pp131-81.

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

McDonald, David B. 1982. *Understanding Noun Compounds.* PhD Thesis, Carnegie-Mellon University, Pittsburgh, PA.

Pustejovsky, J., Bergler, S. and Anick, P. 1993. Lexical Semantic Techniques for Corpus Analysis. In *Computational Linguistics* **Vol 19(2)**, Special Issue on Using Large Corpora II, pp331-58.

Resnik, Philip. 1993. *Selection and Information: A Class-Based Approach to Lexical Relationships.* PhD dissertation, University of Pennsylvania, Philadelphia, PA.

Resnik, P. and Hearst, M. 1993. Structural Ambiguity and Conceptual Relations. In *Proceedings of the Workshop on Very Large Corpora: Academic and Industrial Perspectives*, June 22, Ohio State University, pp58-64.

Ryder, Mary Ellen. 1994. *Ordered Chaos: The Interpretation of English Noun-Noun Compounds.* University of California Press Publications in Linguistics, **Vol 123**.

Smadja, Frank. 1993. Retrieving Collocations from Text: Xtract. In *Computational Linguistics*, **Vol 19(1)**, Special Issue on Using Large Corpora I, pp143-177.

Sparck Jones, Karen. 1983. Compound Noun Interpretation Problems. In Fallside, F. and Woods, W.A., editors, *Computer Speech Processing.* Prentice-Hall, NJ. pp363-81.

Sproat, Richard. 1994. English noun-phrase accent prediction for text-to-speech. In *Computer Speech and Language*, **Vol 8**, pp79-94.

Vanderwende, Lucy. 1993. SENS: The System for Evaluating Noun Sequences. In Jensen, K., Heidorn, G. and Richardson, S., editors, *Natural Language Processing: The PLNLP Approach.* Kluwer Academic, pp161-73.

ter Stal, Wilco. 1995. Syntactic Disambiguation of Nominal Compounds Using Lexical and Conceptual Association. Memorandum UT-KBS-95-002, University of Twente, Enschede, Netherlands.

Yarowsky, David. 1992. Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora. In *Proceedings of COLING-92*, Nantes, France, pp454-60.