Class-Based Probability Estimation using a Semantic Hierarchy

Stephen Clark
Division of Informatics
University of Edinburgh
2 Buccleuch Place, Edinburgh
EH8 9LW, UK
stephenc@cogsci.ed.ac.uk

David Weir
School of Cognitive and Computing Sciences
University of Sussex
Falmer, Brighton
BN1 9QH, UK
david.weir@cogs.susx.ac.uk

Abstract

This paper concerns the acquisition of a particular kind of lexical knowledge, namely the knowledge of which noun senses can fill argument slots of predicates. Probabilities are used to represent the knowledge, and classes from a semantic hierarchy are used to estimate the probabilities. There is a particular focus on the problem of how to determine a suitable class, or level of generalisation, in the hierarchy. A pseudo disambiguation task is used to compare different class-based estimation methods.

1 Introduction

This paper concerns the problem of how to estimate the probability of a noun sense appearing as a particular argument to a predicate. The problem with estimating a probability model over senses is that this involves a huge number of parameters, which results in a sparse data problem. The proposal here is to define a probability model over senses in a semantic hierarchy, and exploit the fact that senses can be grouped into classes consisting of semantically similar senses. Defining probabilities in terms of classes means that the number of parameters is reduced. The assumption underlying this approach is that the probability of a sense can be approximated by a probability based on a suitably chosen class.

The hierarchy used is the noun hypernym hierarchy of WordNet (Fellbaum, 1998), which consists of senses, or 'lexicalised concepts', related by the 'is-a-kind-of' relation. If c is-a-kind-of c', then c' is a *hypernym* of c. To estimate the probability of a concept, c, appearing as an argument of a predicate, a set of concepts dominated by a hypernym of c is chosen to represent c. We develop a novel solution to the problem of how to determine a suitable hypernym, or level of generalisation, in the hierarchy. A pseudo disambiguation task is used to compare our class-based estimation method with some alternative proposals.

2 The Semantic Hierarchy

We use the noun hypernym hierarchy of WordNet, version 1.6. A sense, or concept, in WordNet is represented by a 'synset', which is the set of synonymous words that can be used to denote that concept.[1] For example, the synset for the concept $\langle\texttt{cocaine}\rangle$ is $\{\, \textit{cocaine}, \textit{cocain}, \textit{coke}, \textit{snow}, C \,\}$. Let $\mathrm{syn}(c)$ be the synset for concept c, and let $\mathrm{cn}(n) = \{\, c \mid n \in \mathrm{syn}(c) \,\}$ be the set of concepts that can be denoted by noun n.

The hierarchy has the structure of a DAG, with what we call the 'direct-isa' relation connecting nodes in the graph. Let $\mathrm{isa} = \text{direct-isa}^*$ be the transitive reflexive closure of direct-isa, so that $(c, c') \in \mathrm{isa} \Rightarrow c'$ is a hypernym of c. We use $\overline{c'} = \{\, c \mid (c, c') \in \mathrm{isa} \,\}$ to denote the set consisting of c' and those concepts dominated by c'. For example, $\overline{\langle\texttt{animal}\rangle}$ is the set consisting of those concepts that denote kinds of animals.

The probability of a concept appearing as an argument of a predicate is written $p(c|v, r)$, where c is a concept in WordNet, v is a predicate and r is an argument position. The focus in this paper is on verbs, but the techniques can be applied to any predicate that takes nominal arguments. The probability $p(c|v, r)$ is to be interpreted as follows: this is the probability that some noun n in $\mathrm{syn}(c)$, when denoting concept c, appears in position r of verb v (given v and r). The example used throughout the paper is $p(\langle\texttt{dog}\rangle|\textit{run}, \texttt{subj})$, which is the conditional probability that some noun in the synset

[1] Note that we are using *concept* to refer to a lexicalised concept or sense, and not a set of senses. Angled brackets are used to denote concepts in the hierarchy.

of $\langle\texttt{dog}\rangle$, when denoting $\langle\texttt{dog}\rangle$, appears in the subject position of the verb *run*.

The data used to estimate the probabilities is assumed to be in the form of (n, v, r) triples: a noun, verb and argument position. Such data can be obtained from a treebank or from a robust parser. All the data used here have been obtained using the system of Briscoe and Carroll (1997). Note that no distinction is made between the different senses of a verb, and each noun is assumed to denote exactly one concept.

3 Class-Based Probability Estimation

Using the method of maximum likelihood to estimate $p(\langle\texttt{dog}\rangle|\textit{run}, \text{subj})$ would not be appropriate. The problem with maximum likelihood estimation is that it tends to *over-fit* the data, giving too much probability mass to cases seen in the data, and no probability mass to unseen cases. The solution proposed here is to base the probability estimate on a suitably chosen class, such as $\overline{\langle\texttt{animal}\rangle}$, so that the probabilities of unseen cases can be inferred from the seen cases.

Before explaining how a suitable class is chosen, we first explain how a set of concepts $\overline{c'}$ can be used to estimate $p(c|v, r)$. An inappropriate strategy would be to simply substitute $\overline{c'}$ for the individual concept c, since $p(\overline{c'}|v, r)$ is the conditional probability that some noun denoting a concept in $\overline{c'}$ appears in position r of verb v. For example, $p(\overline{\langle\texttt{animal}\rangle}|\textit{run}, \text{subj})$ is the probability that some noun denoting a kind of animal appears in the subject position of *run*. Probabilities of sets of concepts are obtained by summing over the concepts in the set:

$$p(\overline{c'}|v, r) = \sum_{c'' \in \overline{c'}} p(c''|v, r) \qquad (1)$$

This means that $p(\overline{\langle\texttt{animal}\rangle}|\textit{run}, \text{subj})$ is likely to be much greater than $p(\langle\texttt{dog}\rangle|\textit{run}, \text{subj})$, and not a good approximation of $p(\langle\texttt{dog}\rangle|\textit{run}, \text{subj})$.

$$\begin{aligned}
p(v|\overline{c'}, r) &= p(\overline{c'}|v, r)\frac{p(v|r)}{p(\overline{c'}|r)} \\
&= \frac{p(v|r)}{p(\overline{c'}|r)}\left(\sum_i p(\overline{c'_i}|v, r) + p(c'|v, r)\right) \\
&= \frac{p(v|r)}{p(\overline{c'}|r)} \times \\
&\quad \sum_i p(v|\overline{c'_i}, r)\frac{p(\overline{c'_i}|r)}{p(v|r)} + (v|c', r)\frac{p(c'|r)}{p(v|r)} \\
&= \frac{1}{p(\overline{c'}|r)}\left(\sum_i k\ p(\overline{c'_i}|r) + k\ p(c'|r)\right) \\
&= \frac{k}{p(\overline{c'}|r)}\left(\sum_i p(\overline{c'_i}|r) + p(c'|r)\right) \\
&= k
\end{aligned}$$

Figure 1: Demonstration of how probabilities can remain constant when moving up the hierarchy.

What can be done, though, is to *condition* on sets of concepts. If it can be shown that $p(v|\overline{c'}, r)$, for some hypernym, c', of c, is a reasonable approximation of $p(v|c, r)$, then we have a way of estimating $p(c|v, r)$. The probability $p(v|c, r)$ can be obtained from $p(c|v, r)$ using Bayes' theorem:

$$p(c|v, r) = p(v|c, r)\frac{p(c|r)}{p(v|r)} \qquad (2)$$

Since $p(c|r)$ and $p(v|r)$ are conditioned on the argument slot only, it is more likely these can be estimated satisfactorily using maximum likelihood estimation. This leaves $p(v|c, r)$. Continuing with the $\langle\texttt{dog}\rangle$ example, the proposal is to estimate $p(\textit{run}|\langle\texttt{dog}\rangle, \text{subj})$ using a maximum likelihood estimate of $p(\textit{run}|\overline{\langle\texttt{animal}\rangle}, \text{subj})$, or something similar. In Figure 1, it is shown that if $p(v|\overline{c'_i}, r) = k$ for each child c'_i of c', and $p(v|c', r) = k$, then $p(v|\overline{c'}, r)$ will also be equal to k.[2]

Figure 1 shows how probabilities conditioned on sets of concepts can remain constant when moving up the hierarchy, and this suggests a way of finding a suitable set, $\overline{c'}$, for concept c: initially set c' equal to c, and move up the hierarchy, changing the value of c', until there is a significant change in $p(v|\overline{c'}, r)$.[3] Estimates of

[2] The proof only applies to a tree, whereas WordNet is a DAG. However, since WordNet is a close approximation to a tree, we assume this will not be a problem in practice.

[3] We assume that $p(v|\overline{c}, r)$ is close to $p(v|c, r)$; in fact,

$$\hat{p}(c|r) = \frac{f(c,r)}{f(r)} = \frac{\sum_{v' \in \mathcal{V}} f(c,v',r)}{\sum_{v' \in \mathcal{V}} \sum_{c' \in \mathcal{C}} f(c',v',r)}$$

$$\hat{p}(v|r) = \frac{f(v,r)}{f(r)} = \frac{\sum_{c' \in \mathcal{C}} f(c',v,r)}{\sum_{v' \in \mathcal{V}} \sum_{c' \in \mathcal{C}} f(c',v',r)}$$

$$\hat{p}(v|\overline{c'},r) = \frac{f(\overline{c'},v,r)}{f(\overline{c'},r)} = \frac{\sum_{c'' \in \overline{c'}} f(c'',v,r)}{\sum_{v' \in \mathcal{V}} \sum_{c'' \in \overline{c'}} f(c'',v',r)}$$

Table 1: Maximum Likelihood Estimates – $f(c,v,r)$ is the number of (n,v,r) triples in the data in which n is being used to denote c; $\mathcal{V}$ is the set of verbs in the data, and $\mathcal{C}$ is the set of concepts.

$p(v|\overline{c'_i},r)$, for each child c'_i of c', can be compared to see if $p(v|\overline{c'},r)$ has significantly changed. (We ignore the probability $p(v|c',r)$, and consider the probabilities $p(v|\overline{c'_i},r)$ only.)

Before giving the details of the generalisation procedure, we give the maximum likelihood estimates of the relevant probabilities, and deal with the problem of ambiguous data. The estimates are given in Table 1. The problem is that the estimates are defined in terms of frequencies of senses, whereas the data consists of nouns. In response to this, we estimate $f(c,v,r)$ by simply distributing the count for each noun n in syn(c) evenly among all senses of the noun:

$$\hat{f}(c,v,r) = \sum_{n \in \mathrm{syn}(c)} \frac{f(n,v,r)}{|\,\mathrm{cn}(n)|} \qquad (3)$$

where $|\,\mathrm{cn}(n)|$ is the cardinality of $\mathrm{cn}(n)$. Resnik (1998) explains how this apparently crude technique works surprisingly well.

4 Finding a suitable Level of Generalisation

In this section we give the details of how to find a suitable class to represent a concept. We first show how to test if $p(v|\overline{c'},r)$ changes significantly by moving up a node in the hierarchy.

Consider the problem of deciding if $p(run|\overline{\langle\texttt{canine}\rangle},\mathrm{subj})$ is a good approximation of $p(run|\overline{\langle\texttt{dog}\rangle},\mathrm{subj})$. ($\langle\texttt{canine}\rangle$ is the parent of $\langle\texttt{dog}\rangle$ in WordNet.) To do this, the probabilities $p(run|\overline{c'_i},\mathrm{subj})$ are compared using a chi-squared test, where the c'_i are the children of $\langle\texttt{canine}\rangle$. In this case, the null hypothesis of the test is that the probabilities $p(run|\overline{c_i},\mathrm{subj})$ are the same for each child c_i. By judging the strength of the evidence against the null hypothesis, it can be determined how similar the true probabilities are likely to be. If the test indicates that the probabilities are likely to be very different, then the null hypothesis is rejected, and the conclusion is that $p(run|\overline{\langle\texttt{canine}\rangle},\mathrm{subj})$ is not a good approximation of $p(run|\overline{\langle\texttt{dog}\rangle},\mathrm{subj})$.

An example contingency table, based on counts obtained from a subset of the BNC, is given in Table 2. One column contains estimates of counts arising from concepts in $\overline{c_i}$ appearing in the subject position of *run*: $\hat{f}(\overline{c_i},run,\mathrm{subj})$. A second column contains estimates of counts arising from concepts in $\overline{c_i}$ appearing in the subject position of a verb other than *run*. The figures in brackets are the expected values, given that the null hypothesis is true. There is a choice of which statistic to use in conjunction with the test. The usual statistic encountered in text books is the Pearson chi-squared statistic, denoted X^2. However, Dunning (1993) claims that the log-likelihood chi-squared statistic (G^2) is more appropriate for corpus-based NLP. In Section 6, we compare the two statistics in a task-based evaluation.

For Table 2, the value of G^2 is 3.8 and the value of X^2 is 2.5. Assuming a level of significance of $\alpha = 0.05$, the critical value is 12.6 (for 6 degrees of freedom). Thus, for this α value, the null hypothesis would not be rejected for either statistic, and the conclusion would be that there is no reason to suppose $p(run|\overline{\langle\texttt{canine}\rangle},\mathrm{subj})$ is not a reasonable approximation of $p(run|\overline{\langle\texttt{dog}\rangle},\mathrm{subj})$.

A key question is how to select the value for α. We could just select a value, such as 0.05, but any value determined in this way is to some extent arbitrary. An alternative solution is to treat α as a parameter and set it empirically, by taking a held-out test set and choosing the value of α that maximises performance on the relevant task. Note that this approach sets no constraints on the value of α: the value could be as high as 0.995, or as low as 0.0005, depending on the particular application.

The procedure for finding a suitable class,

$p(v|\overline{c},r)$ is equal to $p(v|c,r)$ when c is a leaf node.

$\overline{C}$	$\hat{f}(\overline{C}, run, \text{subj})$		$\hat{f}(\overline{C}, \text{subj})$ $-\hat{f}(\overline{C}, run, \text{subj})$		$\hat{f}(\overline{C}, \text{subj}) =$ $\sum_{v \in \mathcal{V}} \hat{f}(\overline{C}, v, \text{subj})$
$\overline{\langle\texttt{bitch}\rangle}$	0.3	(0.5)	26.7	(26.6)	27.0
$\overline{\langle\texttt{dog}\rangle}$	12.8	(10.5)	620.4	(622.7)	633.2
$\overline{\langle\texttt{wolf}\rangle}$	0.3	(0.6)	38.7	(38.4)	39.0
$\overline{\langle\texttt{jackal}\rangle}$	0.0	(0.3)	20.0	(19.7)	20.0
$\overline{\langle\texttt{wild\_dog}\rangle}$	0.0	(0.0)	3.0	(3.0)	3.0
$\overline{\langle\texttt{hyena}\rangle}$	0.0	(0.2)	10.0	(9.8)	10.0
$\overline{\langle\texttt{fox}\rangle}$	0.0	(1.2)	72.3	(71.1)	72.3
	13.4		791.1		804.5

Table 2: Contingency table for the children of ⟨`canine`⟩ in the subject position of *run*

$\overline{c'}$, to represent concept c in position r of verb v works as follows. (We refer to $\overline{c'}$ as the 'similarity-class' of c with respect to v and r, and the hypernym c' as $\text{top}(c, v, r)$.) Initially, a variable `top` is assigned to the concept c itself. Then, by working up the hierarchy, `top` is reassigned to be successive hypernyms of c. This continues until the probabilities associated with the sets of concepts dominated by `top` and the siblings of `top` are significantly different. Once a node is reached that results in a significant result for the chi-squared test, the procedure stops, and `top` is returned as $\text{top}(c, v, r)$. In cases where a concept has more than one parent, the parent is chosen that results in the lowest value of the chi-squared statistic, as this indicates the probabilities are more similar. The set $\overline{\text{top}(c, v, r)}$ is the similarity-class of c for verb v and position r.

There may be cases where the conditions for the appropriate application of a chi-squared test are not met. One condition that is likely to be violated is the requirement that expected values in the contingency table should not be too small. (A rule of thumb often found in text books is that the expected values should be greater than 5.) One response to this problem is to apply some kind of thresholding, and either ignore counts below the threshold, or only apply the test to tables that do not contain low counts. The problem with this approach is that any threshold is, to some extent, arbitrary, and there is evidence to suggest that, for some tasks, low counts are important (Collins and Brooks, 1995). Another approach would be to use Fisher's exact test, which can be applied to tables regardless of the size of the counts. The main problem with this test is that it is computationally expensive, especially for large contingency tables (and it only applies to tables with whole number counts).

What we have found in practice is that applying the chi-squared test to tables with low counts tends to produce an insignificant result, and the null hypothesis is not rejected. The consequences of this for the generalisation procedure are that low count tables tend to result in the procedure moving up to the next node in the hierarchy. But given that the purpose of the generalisation is to overcome the sparse data problem, this behaviour is desirable.

Table 3 shows some example generalisation levels for a small number of hand-picked verbs, over a range of values for α. The G^2 statistic was used in the chi-squared tests, and the data were extracted from a subset of the BNC using the system of Briscoe and Carroll. The number of times that each verb in the table occurred in the data (with some object) is shown. The table indicates that the extent of generalisation increases with a decrease in the value of α. This is to be expected, since, given a contingency table chosen at random, a higher value of α is more likely to lead to a significant result than a lower value of α.

The point of the table is not to argue that the example generalisation levels are 'correct', but simply to show some examples, and give some indication of how the generalisation level changes with values in α. We argue that, since the purpose of this work is probability estimation, the most suitable level is the one that leads to the best estimate. So if ⟨`hotdog`⟩, for example, generalises to ⟨`sandwich`⟩(in the object po-

(c, v, r), $f(v, r)$	α	
(⟨hotdog⟩, *eat*, obj)	0.0005	⟨hotdog⟩⟨sandwich⟩⟨snack_food⟩⟨DISH⟩ ... ⟨food⟩ ... ⟨entity⟩
	0.05	⟨hotdog⟩⟨sandwich⟩⟨snack_food⟩⟨DISH⟩ ... ⟨food⟩ ... ⟨entity⟩
$f(eat, \text{obj})$	0.5	⟨hotdog⟩⟨sandwich⟩⟨snack_food⟩⟨DISH⟩ ... ⟨food⟩ ... ⟨entity⟩
$= 1,703$	0.995	⟨hotdog⟩⟨SANDWICH⟩⟨snack_food⟩⟨dish⟩ ... ⟨food⟩ ... ⟨entity⟩
(⟨belief⟩, *abandon*, obj)	0.0005	⟨belief⟩⟨mental_object⟩⟨cognition⟩⟨PSYCHOLOGICAL_FEATURE⟩
	0.05	⟨belief⟩⟨MENTAL_OBJECT⟩⟨cognition⟩⟨psychological_feature⟩
$f(abandon, \text{obj})$	0.5	⟨BELIEF⟩⟨mental_object⟩⟨cognition⟩⟨psychological_feature⟩
$= 673$	0.995	⟨BELIEF⟩⟨mental_object⟩⟨cognition⟩⟨psychological_feature⟩
(⟨Socrates⟩, *kiss*, obj)	0.0005	⟨Socrates⟩ ... ⟨person⟩⟨life_form⟩⟨CAUSAL_AGENT⟩⟨entity⟩
	0.05	⟨Socrates⟩ ... ⟨person⟩⟨life_form⟩⟨CAUSAL_AGENT⟩⟨entity⟩
$f(kiss, \text{obj})$	0.5	⟨Socrates⟩ ... ⟨person⟩⟨life_form⟩⟨CAUSAL_AGENT⟩⟨entity⟩
$= 345$	0.995	⟨Socrates⟩ ... ⟨PERSON⟩⟨life_form⟩⟨causal_agent⟩⟨entity⟩

Table 3: Example levels of generalisation for different values of α; the selected level is shown in upper case

sition of *eat*), rather than the 'more intuitive' ⟨food⟩, this should not be considered a failure. If there exist plenty of data about sandwiches, why generalise any higher? Indeed, we show in Section 6 that to generalise unnecessarily can be harmful for some tasks.

5 Alternative Approaches

The approaches used for comparison are those of Resnik (1993), subsequently developed by Ribas (1995), and Li and Abe (1998), which has been adopted by McCarthy (2000). These have been chosen because they directly address the question of how to find a suitable level of generalisation in WordNet.

The first alternative is to use the 'association score', which is a measure of how well a set of concepts, C, satisfies the selectional preferences of a verb, v, for argument position, r:

$$\mathrm{A}(C, v, r) = p(C|v, r) \log_2 \frac{p(C|v, r)}{p(C|r)} \qquad (4)$$

An estimate of the association score, $\hat{\mathrm{A}}(C, v, r)$, can be obtained using maximum likelihood estimates of the probabilities. The key question is how to find a suitable set to represent concept c, assuming the choice is from those sets dominated by hypernyms of c. Resnik's suggestion is to choose the set that maximises the association score.

The second alternative is to use the Minimum Description Length (MDL) Principle. Li and Abe use MDL to estimate probabilities of the form $p(n|v, r)$, where n is a noun. Their use of MDL requires the hierarchy to be viewed as a thesaurus, in which nouns are represented at leaf nodes, and each internal node represents a class, which is the set of all the nouns at leaf nodes dominated by the internal node.

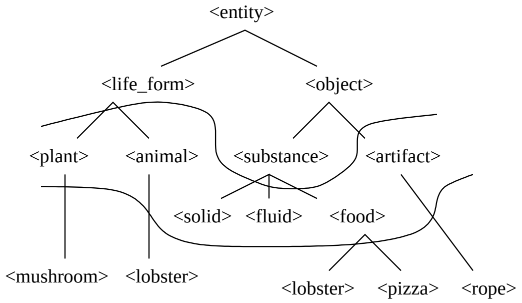

Figure 2: Possible cut returned by MDL

Once the hierarchy is viewed in this way, MDL can be used to determine a 'cut' across the hierarchy, where a cut defines a partition of the leaf nodes. The appropriate class for n is the class in the cut that contains n. The probabilities are estimated by dividing the probability of the class evenly among the nouns in the class.

An example cut showing part of the hierarchy, based on an example from Li and Abe (1998), is given in Figure 2. This example is for the direct object slot of *eat*. In this case, the appropriate class for *pizza* is FOOD, and the probability $p(pizza|eat, \text{obj})$ is estimated by dividing the probability mass for FOOD evenly among all the nouns in the synsets dominated by ⟨food⟩.

We used McCarthy's (2000) implementation of MDL. In order that every noun is represented at a leaf node, McCarthy creates new leaf nodes for each synset at an internal node. However, unlike Li and Abe, McCarthy does not transform WordNet into a tree, which is strictly required for Li and Abe's application of MDL. This did create a problem, in that many of the cuts returned by MDL were over-generalising at the ⟨`entity`⟩ node. The reason is that ⟨`person`⟩, which is close to ⟨`entity`⟩, and dominated by ⟨`entity`⟩, has two parents: ⟨`life_form`⟩ and ⟨`causal_agent`⟩. This DAG-like property was responsible for the over-generalisation, and so we removed the link between ⟨`person`⟩ and ⟨`causal_agent`⟩. This appeared to solve the problem, and the results presented later for the average degree of generalisation do not show an over-generalisation compared with those given in Li and Abe (1998).

6 Pseudo Disambiguation Experiments

The task we used to compare different generalisation techniques is similar to that used by Pereira et al. (1993) and Rooth et al. (1999). The task is to decide which of two verbs, v and v', is more likely to take a given noun, n, as an object. The test and training data were obtained as follows. A number of verb direct object pairs were extracted from a subset of the BNC, using the system of Briscoe and Carroll. All those pairs containing a noun not in WordNet were removed, and each verb and argument was lemmatised. This resulted in a data set of around 1.3 million (v, n) pairs.

To form a test set, $3,000$ of these pairs were randomly selected, such that each selected pair contained a fairly frequent verb. (Following Pereira et al., only those verbs that occurred between 500 and $5,000$ times in the data were considered.) Each instance of a selected pair was then deleted from the data. This was to ensure that the test data were unseen. The remaining pairs formed the training data. To complete the test set, a further fairly frequent verb, v', was randomly chosen for each (v, n) pair. The random choice was made according to the verb's frequency in the original data set, subject to the condition that the pair (v', n) did not occur in the training data. Given the set of (v, n, v') triples, the task is to decide whether (v, n) or (v', n) is the correct pair.

Using our approach, the disambiguation decision for each (v, n, v') triple was made as follows:

```
If $\max_{c \in \mathrm{cn}(n)} \hat{p}(c|v, \mathrm{obj}) > \max_{c \in \mathrm{cn}(n)} \hat{p}(c|v', \mathrm{obj})$
  then choose $(v, n)$
else if $\max_{c \in \mathrm{cn}(n)} \hat{p}(c|v', \mathrm{obj}) > \max_{c \in \mathrm{cn}(n)} \hat{p}(c|v, \mathrm{obj})$
  then choose $(v', n)$
else choose at random
```

If n has more than one sense, the sense is chosen that maximises the relevant probability estimate; this explains the maximisation over $\mathrm{cn}(n)$. The probability estimates were obtained using our class-based method, and the G^2 statistic was used for the chi-squared test.

Using the association score, the decision for each test triple was made as follows:

```
If $\max_{c \in \mathrm{cn}(n)} \max_{c' \in \mathrm{h}(c)} \hat{\mathrm{A}}(\overline{c'}, v, \mathrm{obj}) >$
    $\max_{c \in \mathrm{cn}(n)} \max_{c' \in \mathrm{h}(c)} \hat{\mathrm{A}}(\overline{c'}, v', \mathrm{obj})$
  then choose $(v, n)$
else if $\max_{c \in \mathrm{cn}(n)} \max_{c' \in \mathrm{h}(c)} \hat{\mathrm{A}}(\overline{c'}, v', \mathrm{obj}) >$
    $\max_{c \in \mathrm{cn}(n)} \max_{c' \in \mathrm{h}(c)} \hat{\mathrm{A}}(\overline{c'}, v, \mathrm{obj})$
  then choose $(v', n)$
else choose at random
```

We use $\mathrm{h}(c)$ to denote the set consisting of the hypernyms of c. The inner maximisation is over $\mathrm{h}(c)$, assuming c is the chosen sense of n, which corresponds to Resnik's method of choosing a set to represent c. The outer maximisation is over the senses of n, $\mathrm{cn}(n)$, which determines the sense of n by choosing the sense that maximises the association score.

Using MDL, the disambiguation decision was made as follows ($\tilde{p}$ is used to denote an estimate using the MDL approach):

```
If $\max_{n' \in \mathrm{sep}(n)} \tilde{p}(n'|v, \mathrm{obj}) > \max_{n' \in \mathrm{sep}(n)} \tilde{p}(n'|v', \mathrm{obj})$
  then choose $(v, n)$
else if $\max_{n' \in \mathrm{sep}(n)} \tilde{p}(n'|v', \mathrm{obj}) > \max_{n' \in \mathrm{sep}(n)} \tilde{p}(n'|v, \mathrm{obj})$
  then choose $(v', n)$
else choose at random
```

Since some nouns appear more than once in WordNet, the instance of n is chosen that maximises the relevant probability estimate. We use $\mathrm{sep}(n)$ to denote the separate instances of n.

Generalisation technique	% correct	av.gen.	sd.gen
Similarity-class			
$\alpha = 0.0005$	73.5	3.3	2.0
$\alpha = 0.05$	73.2	2.8	1.9
$\alpha = 0.3$	72.5	2.4	1.8
$\alpha = 0.75$	73.5	1.9	1.6
$\alpha = 0.995$	72.8	1.2	1.2
Low-class	72.1	0.9	1.0
MDL	68.3	4.1	1.9
Assoc	63.9	4.2	2.1

Table 4: Results for the pseudo disambiguation task

Generalisation technique	% correct	av.gen.	sd.gen
Similarity-class			
$\alpha = 0.0005$	66.4	4.5	1.9
$\alpha = 0.05$	68.1	4.1	1.9
$\alpha = 0.3$	69.8	3.7	1.9
$\alpha = 0.75$	72.1	3.0	1.9
$\alpha = 0.995$	71.8	1.9	1.6
Low-class	71.0	1.1	1.1
MDL	62.9	4.7	1.9
Assoc	62.6	4.1	2.0

Table 5: Results for the pseudo disambiguation task with 1/5th training data

The first set of results is given in Table 4. Our technique is referred to as the 'similarity-class' technique, and the approach using the association score is referred to as 'Assoc'. The results are given for a range of α values, and demonstrate clearly that the performance of similarity-class varies little with changes in α, and similarity-class outperforms both MDL and Assoc.

We also give a score for our approach using a simple generalisation procedure, which we call "Low-class". The procedure is to select the first class that has a count greater than zero (relative to the verb and argument position), which is likely to return a low level of generalisation, on the whole. The results show that our generalisation technique only narrowly outperforms the simple generalisation procedure. Note that "Low-class" is still using our class-based estimation method, by applying Bayes' theorem and conditioning on a class, as described in Section 3; the difference is in how class is chosen.

To investigate the results, we calculated the average number of generalised levels for each approach. The number of generalised levels for a concept c (relative to a verb v and argument position r) is the difference in depth between c and $\text{top}(c, v, r)$. To give an example of how the difference in depth was calculated, suppose $\langle$`dog`$\rangle$ generalised to $\langle$`placental_mammal`$\rangle$ via $\langle$`canine`$\rangle$ and $\langle$`carnivore`$\rangle$; in this case the difference would be 3. For each test case, the number of generalised levels for both verbs, v and v', was calculated, but only for the chosen sense of n. The results are given in the third column of Table 4, and demonstrate clearly that both MDL and Assoc are generalising to a greater extent than similarity-class. (The fourth column gives a standard deviation figure.) These results suggest that MDL and Assoc are over-generalising, at least for the purposes of this task.

To investigate why the value for α had no impact on the results, we repeated the experiment, but with 1/5th of the data. A new data set was created by taking every 5th pair of the original 1.3 million pairs. A test set of $3,000$ triples was created from this new data set, as before, but this time only verbs that occurred between 100 and $1,000$ times were considered. The results using these test and training data are given in Table 5.

These results show a variation in performance across values for α, with an optimal performance when α is around 0.75. (Of course, in practice, the value for α would need to be optimised on a held-out set.) But even with this variation, similarity-class is still out-performing MDL and Assoc across the whole range of α values. Note that the α values corresponding to the lowest scores lead to a significant amount of generalisation, which provides additional evidence that MDL and Assoc are over-generalising for this task. The Low-class method scores highly for this data set also, but given that the task is one that apparently favours a low level of generalisation, the high score is not too surprising.

As a final experiment, we compared the task performance using the X^2, rather than G^2, statistic in the chi-squared test. The results are given in Table 6 for the complete data set. The

α value	% correct – G^2	% correct – X^2
0.0005	73.5 (3.3)	73.9 (3.0)
0.05	73.2 (2.8)	73.7 (2.5)
0.3	72.5 (2.4)	73.6 (2.2)
0.75	73.5 (1.9)	73.8 (1.8)
0.995	72.8 (1.2)	72.3 (1.2)

Table 6: Disambiguation results for G^2 and X^2

figures in brackets give the average number of generalised levels. The X^2 statistic is performing at least as well as G^2, throwing doubt on the claim by Dunning (1993) that the G^2 statistic is better suited for use in corpus-based NLP. The results show clearly that the average level of generalisation is slightly higher for G^2 than X^2. This suggests a possible explanation for the results presented here, and those in Dunning (1993), which is that the X^2 statistic provides a less conservative test when counts in the contingency table are low. A less conservative test is better suited to the pseudo disambiguation task, since this results in a low level of generalisation, on the whole, which is good for this task. In contrast, the task that Dunning considers, the discovery of bigrams, is better served by a more conservative test.

7 Conclusion

We have presented a class-based estimation method that incorporates a procedure for finding a suitable level of generalisation in WordNet. This method has been shown to provide superior performance on a pseudo disambiguation task, compared with two alternative approaches. One of the features of the generalisation procedure is the way that α, the level of significance in the chi-squared test, is treated as a parameter. This allows some control over the extent of generalisation, which can be tailored to particular tasks.

8 Acknowledgements

This work was supported by an EPSRC studentship to the first author. We would like to thank Diana McCarthy for suggesting the pseudo disambiguation task and providing the MDL software, John Carroll for supplying the data, and the anonymous reviewers for their helpful comments.

References

E. Briscoe and J. Carroll. 1997. Automatic extraction of subcategorisation from corpora. In *Proceedings of the Fifth ACL Conference on Applied Natural Language Processing*, pages 356–363, Washington, DC.

M. Collins and J. Brooks. 1995. Prepositional phrase attachment through a backed-off model. In *Proceedings of the Third Workshop on Very Large Corpora*, pages 27–38, Cambridge, MA.

T. Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

C. Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database.* The MIT Press.

H. Li and N. Abe. 1998. Generalizing case frames using a thesaurus and the MDL principle. *Computational Linguistics*, 24(2):217–244.

D. McCarthy. 2000. Using semantic preferences to identify verbal participation in role switching. In *Proceedings of the first Conference of the North American Chapter of the Association for Computational Linguistics*, pages 256–263, Seattle, WA.

F. Pereira, N. Tishby, and L. Lee. 1993. Distributional clustering of English words. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 183–190, Columbus, OH.

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

P. Resnik. 1998. Wordnet and class-based probabilities. In Christiane Fellbaum, editor, *WordNet: An Electronic Lexical Database*, chapter 10, pages 239–263. The MIT Press.

F. Ribas. 1995. On learning more appropriate selectional restrictions. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics*, pages 112–118, Dublin, Ireland.

M. Rooth, S. Riezler, D. Prescher, G. Carroll, and F. Beil. 1999. Inducing a semantically annotated lexicon via EM-based clustering. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 104–111, University of Maryland, MD.