# Investigating Semantic Knowledge for Text Learning

Anupriya Ankolekar
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA, 15213, USA
anupriya@cs.cmu.edu

Young-Woo Seo
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA, 15213, USA
ywseo@cs.cmu.edu

Katia Sycara
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA, 15213, USA
katia@cs.cmu.edu

## ABSTRACT

Recent work has made much of using semantic knowledge, derived in particular from domain ontologies, for improving text learning tasks. Semantic knowledge is assumed to capture more in-depth knowledge of the text domain in comparison with conventional statistics-based methods that can only rely on more surface vocabulary-specific characteristics of a data set. Therefore, using semantic knowledge instead of statistics-based methods should improve performance in text learning tasks significantly. We believe that this claim needs careful scrutiny and examine the validity of this assumption in this paper. We explore the usefulness of ontologies for a text classification task and the use of feature selection methods to extract terms that can function as candidate ontological concepts for building or extending ontologies. We point to a number of issues that arise when trying to use semantic knowledge for text classification. One particularly troublesome issue is that semantic knowledge encoded in ontologies simply may not correspond to the concepts and terms significant for text classification.

## Keywords

semantic knowledge, ontologies, feature selection, classification

## 1. INTRODUCTION

As the Semantic Web has grown in prominence in recent years, its natural application to the fields of information retrieval and text learning[1] is being recognized. A primary objective of the Semantic Web is to describe the (semantic) structure and content of Web-based textual data, while the field of text learning aims to detect some underlying characteristics of the content of a given data set to be used for a given text learning task, such as classification.

[1]Text learning [1] is an umbrella term that refers to the application of machine learning techniques to text and hypertext data.

Clearly, both areas can benefit from each other's techniques. There has been work to explore various aspects of the application of semantic knowledge and annotations to the fields of information retrieval and text learning. For example, Hotho et al. [7] improved text clustering with the use of words from a domain concept hierarchy. Shah et al. [16] explored the performance of information retrieval techniques on semantically annotated data. Ontology building, in part by text learning methods, was outlined in [9].

To date, however, we feel there have been few studies to *quantitatively* and *empirically* examine the value of techniques from Semantic Web areas to text learning and vice versa. Specifically, we seek an improved understanding of the trade-offs involved with techniques from each field to be able to better combine them in the future. As we will discuss in the following section, we believe that feature selection, in particular, is key to gaining this understanding, since the best performance in text learning tasks, such as classification, can be achieved through the selection of a good subset of the features in the original feature space.

As a first step towards achieving this goal, this paper focuses on two questions: how useful are the existing feature selection methods for (semi-) automated ontology learning and how useful is the word feature set derived from domain ontologies for improving the performance of text classification. Therefore the contributions of this paper are two-fold:

- This paper provides a quantitative analysis of the word feature coverage of the concepts in a domain ontology, provided by existing feature selection methods.
- It compares the performance of classifiers on the same data set with and without knowledge of a domain ontology, making simple use of the structure of the ontology.

In the next Section 2, we present some of our observations on the fields of text learning, information retrieval and the Semantic Web that motivated our work in this paper. In the following Section 3, we present the feature selection methods used in this paper. Section 4 covers several methods for text classification that we use in our experiments. In Section 5, the experiments on feature selection and classification are described in detail and the results of the experiments presented. Finally, in Section 6, we discuss the results and possible extensions of this work.

## 2. MOTIVATIONS

It is well known that, for text learning and information retrieval fields, maximum performance is often achieved not by using all available features (words or phrases) of a document from the given text data set, but by using only a "good" subset of those. Feature selection refers to the way of extracting a set of features which is more informative for executing a given text learning task or for reducing the dimensionality of the original data set. The selected feature set should contain more reliable or sufficient information from the original data set. Feature selection methods have relied heavily on statistical and information-theoretical measures to capture vocabulary-specific characteristics of a given textual data set to identify good word features. These methods have proved to be useful for text learning tasks, esp. classification and clustering. However, the vocabulary of any data set will be dependent on the context: the time period when the data was generated, the purpose for which the data was generated, and the organization that generated the data. Therefore, the words selected by statistics-based feature selection methods may not extract words corresponding to the high-level concepts.

These concepts can encompass several low-level context dependent concepts. Given, for instance, a set of news articles of a certain period, the word feature set identified by existing feature selection methods may contain word features such as `BMW`, `Mercedes` and `Saab` as good indicators for the automobile category. However, if instead the term `automobile manufacturers` were included as a feature this would encompass the previous three words. Therefore, using a set of higher-level concepts as features can lead to a reduction in the dimension of original feature space and to the generation of a more semantically meaningful word feature set. Since a domain ontology comprises a set of higher-level concepts, such as `automobile manufacturers`, with representation in different words such as `Mercedes` and `Saab`, this indicates that using concept words in a domain ontology as features could help improve on the feature selection methods.

There is, however, a danger if we only rely on concept words of an domain ontology to identify features in a given data set. Since an ontology usually reflects the inherent knowledge and biases of the creator of the ontology, the concept words of the ontology may not accurately reflect the terms that are important for a given text learning task, in the sense of being distinguishing terms for the data set. In other words, there could be a gap between what a human thinks are useful words for a given text learning task and the words that are statistically significant for that text data set. This problem is exacerbated if the user does not come from the same community of people that generated the data set, since vocabularies and jargon vary across communities. It is clear that the words in the ontology should be close to vocabulary of the given data set to be useful for a text learning task. In other words, a generic ontology may be less useful for a text learning task than one designed specifically to represent the information content of the documents in the corpus.

There is thus a potential problem in reconciling manually created ontologies for the tasks of text learning and information retrieval with the semantic structures that are actually useful for these tasks for a particular set of documents. This observation suggests that techniques from text learning and information retrieval can themselves be used to build ontologies (semi-) automatically [3] [7]. This would be highly desirable, as ontology-building is still largely conducted by hand, in a costly, labor-intensive and error-prone process. With the help of techniques from text learning and information retrieval fields, statistically significant terms that could serve as potential concepts in a domain ontology can be presented as candidate concepts words to the domain expert constructing the ontology.

In order to evaluate the use of the existing feature selection methods for ontology building, the first empirical test of our paper will be to compare the word set returned by the existing feature selection methods and by the domain ontology. Since the other key objective of the paper is to explore how semantic knowledge can benefit text learning tasks, the next step is to evaluate the usefulness of the word feature set derived from a domain ontology for text classification.

In the next section, we briefly discuss the two approaches towards feature selection that will form the core of the comparison between text learning with and without semantic knowledge.

## 3. FEATURE SELECTION

Feature selection is the problem of identifying the most informative word features within a set of documents for a given text learning task. Statistics-based feature selection methods, briefly discussed in section 3.1 rely on the vocabulary characteristics of given data set, whereas ontology-based feature selection, discussed in section 3.2, relies on a set of ontologies associated with the document set.

### 3.1 Statistics-based Feature Selection

In our study, we considered two methods, mutual information (MI) and $\chi^2$ statistic. Each of these methods uses a criterion to determine a subset of the original feature space that seems to best capture the characteristics of a given data set.

The mutual information $I(C_j; X_i)$ is the relative entropy between the joint distribution and the product distribution $P(X_i)P(C_j)$ [5].

$$\begin{aligned} I(C_j; X_i) &= H(C_j) - H(C_j|X_i) \\ &\approx log\frac{P(C_j \wedge X_i)}{P(C_j) \times P(X_i)} \end{aligned} \quad (1)$$

where $C_j$ is a class and $X_i$ is a word feature. In particular, it is the reduction in the uncertainty of one random variable $C_j$ due to knowledge about another random variable, $X_i$. The less dependent $X_i$ and $C_j$ are, the closer $I(C_j; X_i)$ is to zero. This information-theoretical measure is commonly used in the statistical modeling of word associations.

The $\chi^2$ statistic measures the lack of independence between $C_j$ and $X_i$ by comparing the observed co-occurrence frequencies in the 2-way contingency table of a word feature $X_i$ and a class $C_j$ with the frequencies expected for independence. $\chi^2(C_j, X_i)$ is estimated by:

$$\chi^2(C_j, X_i) = \frac{|D| \times (ad - cb)^2}{(a+c) \times (b+d) \times (a+b) \times (c+d)} \quad (2)$$

where $a$ is the number of times $X_i$ and $C_j$ co-occur; $b$ is the number of times $X_i$ occurs, but not $C_j$; $c$ is the number of times $C_j$ occurs, but not $X_i$; $d$ is the number of times neither $C_j$ nor $X_i$ occur, and $|D|$ is the total number of documents. $C_j$ and $X_i$ are dependent on each other if the difference between the observed and expected frequencies is large, whereas they are independent if the $\chi^2$ statistics score is close to zero [10]. The scores derived from mutual information and $\chi^2$ statistics should be interpreted with care. In the case of mutual information, low-frequency word features can have higher scores than more common ones, while the scores by $\chi^2$ statistics are known not to be reliable for low-frequency word features [17].

## 3.2 Ontology-based Feature Selection

Ontologies can be used to improve the task of text learning, especially through feature selection. Given an ontology or a set of ontologies associated with a textual data set, there are various ways the ontology can be used for feature selection.

If the data set has an associated comprehensive domain ontology, such that each classification category corresponds to a portion of that ontology, then concepts from the entire ontology could be used as features for the data set. On the other hand, each category could have a dedicated ontology, such as one for `cocoa` and another for `gold`. These ontologies would then be used separately to provide features for each category. These two alternatives are roughly equivalent for our tasks, but the first one will require more time and effort to construct.

At the outset of our experiment, we attempted to construct a comprehensive ontology for the categories under consideration. However, constructing such an ontology is a difficult task, as it necessitates finding connections between large numbers of terms within different categories. As a result, we then turned to constructing a single ontology for each category, a far simpler task. Each ontology was therefore constructed manually, through a process described in Section 5.2.

An orthogonal choice to be made concerns how exactly a portion of an ontology is used to provide features for classification. A straightforward idea is to use the ontological concepts as features. Since the concepts represent the terms, primarily the words–unigrams (single words) or bigrams (two-word phrases)–with most semantic content, they are naturally suited to being used as features for classification. Besides concepts, ontologies also define properties (relations) between these concepts. Ontology-based feature selection methods could make use of certain relations that have names that are meaningful to the category. In this paper, we again focus on the simpler alternative of purely using words in concept names as features. Another point that ought to be mentioned is that ontologies are primarily collections of triples, corresponding to a trigram model whereas most classifiers assume a unigram or bigram word model. Since we are not yet sure about how the ontology trigram model can be reconciled with the uni-/bigram word model, we chose to treat the ontological trigrams as unigrams.

# 4. CLASSIFICATION

Text classification is the problem of assigning documents to one of some set of predefined classes. In a classification task, we are given a training set, $D_{train} = \{\vec{d}_1, ..., \vec{d}_n\}$ where $D_{train} \subset D, \vec{d} \in \mathbb{R}^t$ with preassigned class labels, $C = \{C_1, ..., C_m\}$. Given this data, classification involves inducing a model from the training data that will be effective at predicting the class label in test data, $D_{test} \subset D$.

A Bayesian network classifier is simply a Bayesian network applied to a classification problem. It contains a node, $C_j$, for the class label and a node, $X_i$, for each of the domain features (i.e. words). Given a specific document vector $\vec{d}$, the Bayesian network allows us to compute the posterior probability $p(C_j|\vec{d})$ for each possible class $C_j$.

Bayesian networks are a kind of directed acyclic graph (DAG) that graphically represents the dependencies in a probability distribution. Each variable (feature) in a given document is represented as a node in the graph. Arcs between nodes represent dependencies between the respective variables. If two variables are independent, there is no arc connecting the two corresponding nodes. Furthermore, a node for a variable $X_i$ represents the probability of $X_i$ given the probabilities of the variables that are immediate parents of $X_i$, denoted by $\Pi(X_i)$. Nodes without parents simply represent the prior probability for those variables [13].

Given this framework, we would like to determine the probability distribution $P(C|\vec{d})$, where $C$ is the class variable and $\vec{d}$ is a $t$-dimensional vector $< X_1, ..., X_t >$, representing an observed instance (i.e. document).

## 4.1 Zero-dependence Bayesian Classifier

Within this framework, the simplest classifier is the naive Bayesian classifier. As a Bayesian method, it predicts the class $C_j$ that maximizes the posterior probability, $P(C_j|\vec{d})$, for a document vector $\vec{d}$, under a strong assumption that each feature $X_i$ is conditionally independent of every other feature given the class label.

$$\begin{aligned} P(C|\vec{d}) &= \arg\max_j P(C_j)P(\vec{d}|C_j) \\ &= \arg\max_j P(C_j)\prod_i^t P(X_i|C_j) \end{aligned} \quad (3)$$

The assumption on the feature independence is clearly unrealistic in a textual domain, as well as in other domains. Still, the naive Bayesian classification has been shown to be surprisingly effective on such problems [12] [14]. Due to the feature independence assumption, naive Bayes classifiers are quite computationally efficient [2] [11]. In other words, training such a classifier only requires time that is linear in the number of features and data instances, meaning that they do not use word combinations as predictors and are thus far more efficient than the exponential non-Bayes approaches.

Several approaches have been proposed for augmenting the naive Bayesian classification with limited interactions between the feature nodes, in order to build more accurate classification models. To be more specific, these approaches

allow each node in a Bayesian network to have some parents beyond the class variable. However there are two obstacles which prevent us from exploring this idea. Firstly, it is a NP-hard problem to induce an optimal Bayesian classifier [4]. Second, it could take exponential time in the number of features for any algorithm to construct such a classifier. Two solutions have been proposed to this problem: the Tree-Augmented Naive (TAN) Bayesian classifier and the $k$-dependence Bayesian classifier (KDB). The TAN classifier learns a Bayesian network, where the class variable has no parents and each node has as parents the class variable and at most one other node, meaning that each node can have one augmenting edge pointing to it [6]. The KDB algorithm [15] is used for finding Bayesian classifiers where each node has at most $k$ parents for arbitrary values of $k$. It chooses as the parents of a feature node $X_i$ the $k$ other features that $X_i$ is most dependent on, using a metric of class conditional mutual information.

Since KDB is able to take into account a limited dependency between features, it can make use of the relational information in an ontology unlike naive Bayes classifiers. We therefore chose to use KDB as the basis for our experiments.

## 4.2 *k*-dependence Bayesian Classifier

A *k-dependence Bayesian classifier* is a Bayesian network (BN) which contains the structure of the naive Bayesian classifier and furthermore allows each feature $X_i$ to have a maximum of $k$ feature nodes as parents. In other words, $\Pi(X_i) = \{C, X_{DP_i}\}$ where $X_{DP_i}$ is a set of at most $k$ dependent feature nodes and $\Pi(C) = \emptyset$. With this definition, the full Bayesian classifier is a $(n-1)$-dependence Bayesian classifier, if $n$ is the number of domain features.

Given a $k$ value for the maximum allowable degree of feature dependence, KDB outputs a $k$-dependence Bayesian classifier using conditional probability tables determined from a given data set of pre-classified instances (i.e. documents).

The algorithm itself is as follows:

Do while $j \leq$ (number of target classes):

1. For each feature $X_i$, compute the mutual information, $I(X_i; C_j)$ and rank the features in descending order.
2. For each pair of features $X_i$ and $X_l$, where $i \neq l$, compute the class conditional mutual information $I(X_i; X_l|C_j)$.

$$I(X_i; X_l|C) = \sum_{x_i, x_l, c} P(x_i, x_l, c) log \frac{P(x_i, x_l|c)}{P(x_i|c)P(x_l|c)}$$

3. Let the Bayesian network being constructed $BN$, begin with a single class node with class label $C_j$.
4. For each feature $X_i$, sort the other features $X_l$ by decreasing conditional mutual information $I(X_i; X_l|C_j)$ and select the top two most dependent features as parents of $X_i$, given $C_j$.
5. Compute the conditional probability tables inferred by the structure of $BN$ by using counts from the dataset and output $BN$.

# 5. EXPERIMENTS

As mentioned earlier, there are two objectives that we want to pursue through our experiments:

- To compare the word feature sets identified by statistics-based and ontology-based feature selection methods.
- To compare the performance of Bayesian classifications with and without knowledge of a domain ontology.

To achieve these objectives, for the feature selection experiment, we measured the overlap between words that occurred in the ontology corresponding to a category and their ranking in descending order by the feature selection methods of mutual information and $\chi^2$ statistics. We used the automatic feature selection methods to first rank all the words in the lexicon of a category by their scores. Then, we noted the rank for those words that also appeared in the ontology.

Assuming the class label is the core concept in an ontology, words (i.e. concepts) in the ontology have a natural rank based on their distances from the core concept. However, we do not consider their distance as ranks for comparison because this does not address our objective of investigating how useful the existing methods for feature selection are in selecting words for ontology building.

For the classification experiment, we compared three variants of Bayesian classification. The first case is the naive Bayes classifier which used the word features set identified by the ontology and the feature selection methods (e.g. mutual information and $\chi^2$), and assumed no dependence between word features. The second one is the KDB algorithm, used to construct a 2-dependence Bayesian classifier using word features set obtained from feature selection methods. The limit of 2 for $k$ was chosen primarily to keep the computational complexity down. For the third variant, we again constructed a 2-dependence Bayesian classifier. This time, however,s the KDB algorithm made use of a word feature set from the domain ontologies. Both KDB algorithms made use of conditional mutual information to construct 2-dependence Bayesian classifiers.

## 5.1 Text Data set

To make our evaluation results more comparable to most of the published results in text categorization, we chose the Reuters-21578 dataset.[2]

This data set, which consists of world news stories from 1987, has become a benchmark in text categorization evaluations. It has been partially labelled by experts with respect to a list of categories. These categories have been grouped into super-categories of people, topics, places, organizations etc. The category distribution is skewed: the most common category has a training-set frequency of 2877, but 82% of the categories have less than 100 instances and 33% of the categories have less than 10 instances.

For this paper, we use the "ModLewis" split of Reuters-21578, which contains 13,625 training documents and 6,188

[2]It is publicly available at [8].

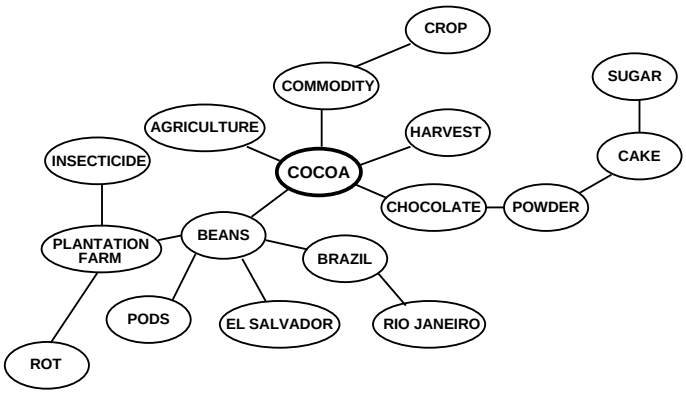


**Figure 1: A section of the ontology manually constructed for the `cocoa` category**

testing documents leaving 1765 unused documents. There are 135 overlapping topic categories, but we used only those 4 for which there exists a relatively large set of documents across training and test set: `cocoa`, `copper`, `cotton`, and `nat-gas` (natural-gas). We chose these four categories because we believe the vocabulary of these categories best reflects that of the whole data set. We limited ourselves to four categories for this paper because building corresponding ontologies is a time-consuming and knowledge-intensive task.

## 5.2 Ontologies

In the absence of existing ontologies for these categories, we constructed ontologies for each of the categories `cocoa`, `copper`, `cotton` and `nat-gas`. This was done as follows: first, for each category, a lexicon of all the distinct words that appeared in documents within that category after eliminating stop words was compiled. Next, the lexicon associated with the category was examined manually and a subset of the words in the lexicon picked as being most closely associated semantically with the category. This list was then used to create an ontology.

A part of the ontology constructed for `cocoa` is shown in Figure 1. The ovals represent concepts and the lines connecting them represent properties or relations between concepts. This is a highly simplified version of the actual ontology. In particular, we do not consider the different kinds of relations connecting the concepts. Furthermore, we are primarily making use of the ontological structure, as opposed to the deductive capabilities a full ontology enables.

## 5.3 Experimental Results

### *5.3.1 Feature Selection*

The results of the feature selection experiment for the Reuters-21578 data set for four categories, `cocoa`, `copper`, `cotton` and `nat-gas` are presented in Figures 2 to 5. The horizontal axis indicates the rank of words that were present in the category lexicon, and the vertical axis represents the cumulative number of words that occurred in the ontology. Thus, the graphs are to be read as follows. For any rank $x$ on the horizontal axis, the graph shows the number of words in the ontology that occurred in the top $x$ number of features as determined by the mutual information and $\chi^2$ methods.

| $\chi^2$ | Mutual Information | Ontology |
|---|---|---|
| tonnes | patterson | cocoa |
| sugar | vengeance | cocoas |
| production | eshleman | chocolate |
| mln | melicias | powder |
| export | muscles | powdered |
| prices | roldan | crushed |
| imports | flex | butter |
| cocoa | flaws | bean |
| pct | nastro | beans |
| week | maccia | agriculture |
| total | demico | commodity |
| buffer | kusumaatmadja | harvest |
| traders | barros | harvests |
| report | wrangles | cake |
| oil | practised | cakes |
| ⋮ | ⋮ | ⋮ |

**Table 1: For the category `cocoa`, word features extracted by statistics- and ontology-based feature selection**

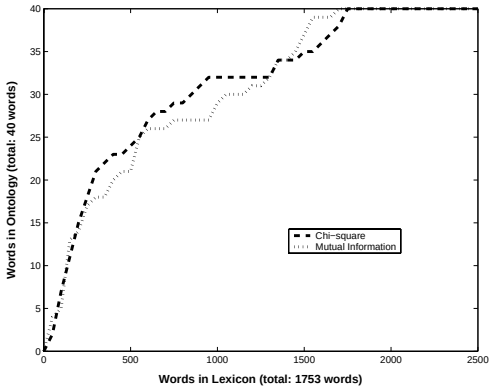


**Figure 2: Feature selection for the `cocoa` category. The dashed line plots the number of features obtained by $\chi^2$ statistics, while the dotted line indicates the number of features obtained by mutual information feature selection.**

As can be seen from the graphs, the performance of the feature selection is relatively good, as in, a large number of words in the ontology show up relatively high in the ranking by both automatic feature selection methods. The results are best for the `copper` category, then the `cotton` category and finally the `cocoa` category. Generally, in order to find half of the words in the ontology, one needs only go through about 200 words in the lexicon instead of 2000 words. As we discuss in Section 6, this may reflect less on the performance of the automatic feature selection methods and be related more to the ontology not fitting the task and domain satisfactorily.

The result for the `nat-gas` category was not as good as that of either of the other three categories and we go more into the possible reasons when discussing the next experiment.

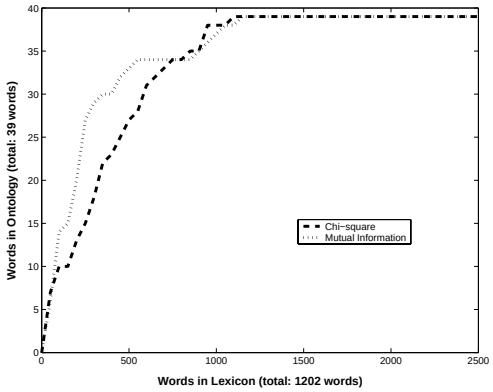


**Figure 3: Feature selection for the `copper` category**

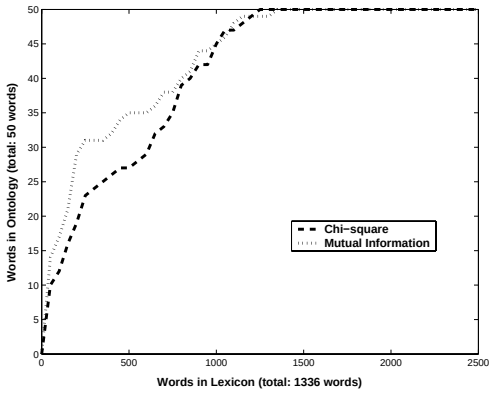


**Figure 4: Feature selection for the `cotton` category**

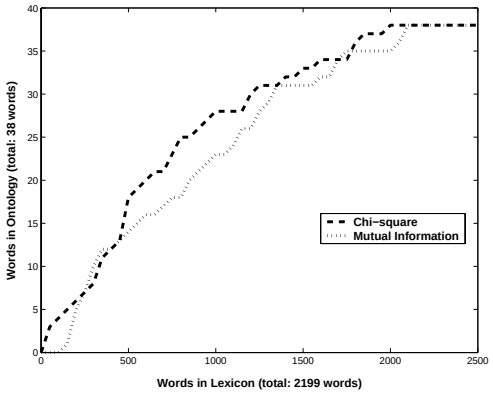


**Figure 5: Feature selection for the `nat-gas` category**

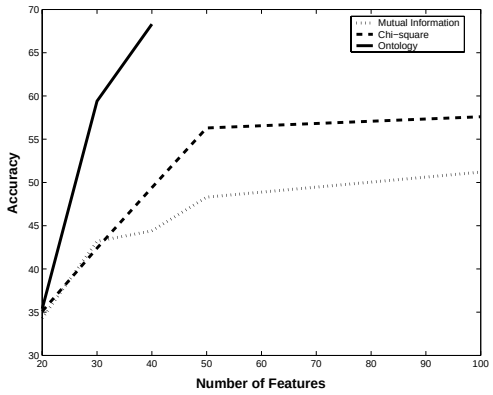


**Figure 6: Classification for the `cocoa` category**

### *5.3.2 Classification*

We conducted two different experiments on classification. The first experiment attempted to show how using word features set from ontology affects the performance of naive Bayes classifier.

The results of the classification accuracy for the Reuters-21578 data set for the four categories, `cocoa`, `copper`, `cotton` and `nat-gas` are presented in Figures 6 to 9. The horizontal axis indicates the number of features that were used in the classification model, and the vertical axis the percentage of the test documents that were correctly classified. Notice that with only 40 word features ontology-based feature selection achieves 68.3% accuracy in `cocoa` category, higher than that of $\chi^2$-based feature selection with 100 features.

The number of features for the ontology-based feature selection is bounded by the number of terms that were deemed to be semantically associated with the category and were thus included in the ontology. For the `cocoa` category, the ontology included 48 terms and this figure differed across categories. In each case, we compared the top 20, 30 or 40 features of the category to ensure equal feature dimensionality. At equal feature dimensionality, for three out of the four categories, the ontology-based feature selection method achieves higher accuracy than two other feature selection algorithms: feature selection by mutual information and chi-square statistics with the class label.

The performance of the classifier for `nat-gas` as in the earlier experiment is not as good as that for the other categories. There are a number of possible reasons that can account for the bad performance. Since the lexicon for the `nat-gas` category had a significant overlap with the lexicons for other categories in the corpus, such as `oil` and `pet-chem` (petrochemicals), the terms included in its ontology have less predictive power for classification. Also, the manually constructed ontology for natural gas itself is suspect, as its construction required specialised domain knowledge.

We conducted another experiment to demonstrate how using a word features set from an ontology affects the performance of a 2-dependence Bayesian classifier because one of our hypotheses is that capturing dependence between word features could achieve a better performance than zero-dependence (i.e. naive Bayesian classifier.) We determined the conditional mutual information for features extracted by both statistics-based and ontology-based feature selection. The Figure 10 shows the same ontology shown in Figure 1 with the arcs in the original ontology shown by thin dotted lines. The additional arcs with continuous lines indicate dependencies identified by conditional mutual information not originally present within the ontology. Furthermore, the thicker the arc, the stronger the dependency identified by conditional mutual information.

The dependencies shown in the graph were determined as

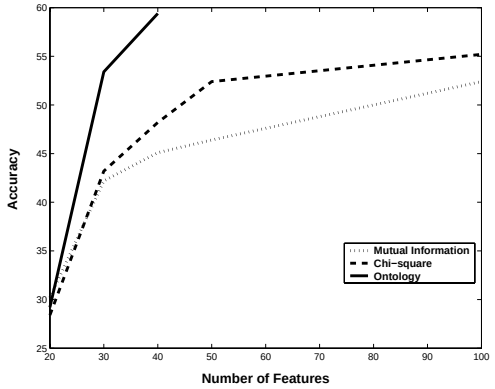


**Figure 7: Classification for the `copper` category**

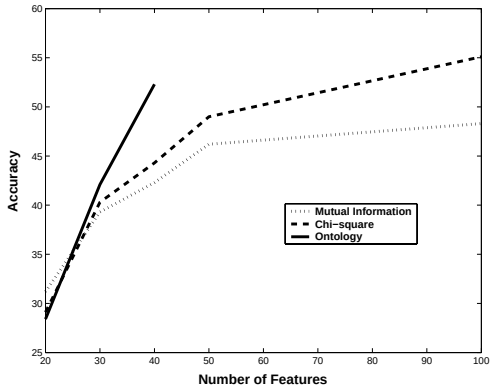


**Figure 8: Classification for the `cotton` category**

follows: first the conditional mutual information was calculated between all word features in the ontology. Then, for each feature, the two most dependent word features were determined and used by the $k$-dependence classifier. These are the word feature dependencies shown in the graph.

The next Figure 11 shows that a Bayesian classifier with a limited (i.e. 2) dependence between features seems better than a Bayesian classifier with zero-dependence. However the performance using a word features set from ontology is comparable, but not better than a word features set from statistics-based feature selection methods. In case of "nat-gas" category, the performance of ontology word feature set and 2-dependence Bayesian classifier is particularly bad. Note that the results of this experiment are similar to that of feature selection experiment. These results are caused by two facts; one is that the ontology for "nat-gas" was not as good as the others and the other is that sufficient statistics were available for that category to the feature selection methods. The overall performance of the combination of ontology word feature set and 2-dependence Bayesian classifier was worse than expected. We suspected this is partially because the vocabulary used in the ontologies does not necessarily reflect the primary word features of the data set, even though the vocabulary came from the same data set, and partially because KDB made good use of statistics of the given data set.

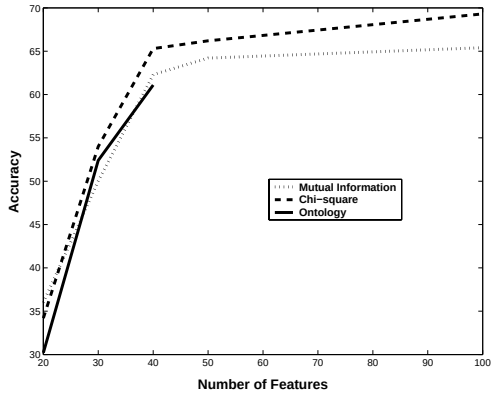


**Figure 9: Classification for the `nat-gas` category**

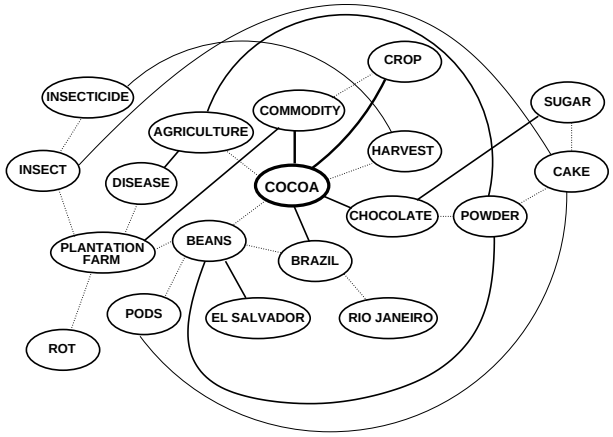


**Figure 10: A section of the ontology for the `cocoa` category with both ontology-based and statistics-based dependencies**

## 6. DISCUSSION AND FUTURE WORK

In this paper, we presented two experiments to evaluate the goodness of the word feature sets identified by statistics-based and ontology-based feature selection methods and to compare the performance of Bayesian classifications with and without knowledge of a domain ontology. Our hypotheses were that a word feature set identified by the existing feature selection methods would be useful for automatic ontology learning and that a word feature set identified by ontology-based feature selection would improve the performance of text classification.

The experimental results on feature selection showed that there was a good overlap between the word feature set identified by existing feature selection methods and the word feature set derived from a domain ontology. With the sole exception of the `nat-gas` category, half of the words in the category ontologies were placed approximately in the top 200 word features identified by the statistics-based feature selection. This indicates that the existing feature selection methods could be useful for identifying a set of candidate words for a domain ontology.

The first experiment on classification was intended to inves-

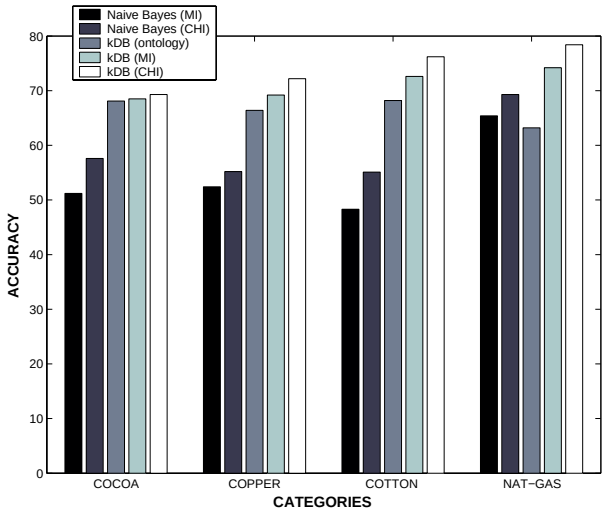


**Figure 11: Results of classification.**

tigate how an ontology-derived word feature set affects the performance of a naive Bayes classifier. The results generally show an improvement in classification performance using an ontology-derived word feature set as opposed to one identified by feature selection methods. However, the size of that improvement is limited, which could be attributed to various factors. For one, although several terms in the ontology were bigrams, such as `saudi arabia` and `rio janeiro`, our underlying model uses unigrams and would therefore split these terms and treat them as single words. Furthermore, during the construction of a domain ontology, the ontology concept names were chosen from words in the lexicon of a category without examining the actual news reports. Thus, while the chosen concept words and relationships may be important from the point of view of encyclopaedic knowledge, they do not present the kind of knowledge required for better classification of news reports. In other words, the vocabulary of the ontology reflects the content of the text data set, but not in a way that is relevant to text classification. We believe this is likely to be a general problem when using ontologies for text classification. Therefore, a semi-automatic approach to constructing ontologies for text learning would be desirable. Our experiments indicate that this approach is also likely to be fruitful.

The other classification experiment attempted to demonstrate how an ontology-derived word feature set affects the performance of 2-dependence Bayesian classifier, since one of our hypotheses was that a classifier capturing the dependence between word features would achieve a better performance than a zero-dependence (i.e. naive Bayesian) classifier. The experimental results show that capturing the dependence between word features does indeed produce a better classification performance than zero-dependence Bayesian classifier. The performance using a ontology-derived word feature set is also comparable, but not far better than using a word feature set identified by statistics-based feature selection methods. We suspect this is primarily because the vocabulary used for ontology does not necessarily reflect the primary word features of the data set, even though the vocabulary came from the same data set. Another possible reason is that KDB was able to make good use of the instances of the given data set. It performs particularly well in the `nat-gas` category, which had a much larger set of associated documents which were themselves divided evenly between the training and the test set. The superior performance of KDB in the `nat-gas` category seems to suggest that ontology-based feature selection might be more useful for cases of insufficient training data. Therefore, KDB makes better use of statistical dependencies between (word) features than of semantic dependencies.

In summary, the experimental results seem to support our hypothesis on the usefulness of the existing feature selection methods, in particular $\chi^2$, for ontology learning. Our hypothesis on the usefulness of ontology-based feature selection for text classification was only partially supported. Therefore, although using ontologies does seem to provide some improvement in classifier performance, we believe more work needs to be done before the trade-offs involved in the use of semantic knowledge for text classification can be understood and the use of ontologies for text classification becomes really worthwhile.

Our future work will include the replication of these experiments for more categories of the Reuters-21578 data set, as well as for several other data sets. Furthermore, we will explore the effect of insufficient data on the performance of the two kinds of classifiers. An interesting application area would be in the classification of multimedia data with text captions. These captions are usually concise, include words with a high degree of semantic content and would probably show significantly improved classification performance from ontology-based feature selection. In addition, we will explore a hybrid approach that combines statistics-based and ontology-based feature selection to identify a set of word feature that is useful for text classification.

## 7. ACKNOWLEDGMENTS

The research was funded by the Defense Advanced Research Projects Agency as part of the DARPA Agent Markup Language (DAML) program under the Air Force Research Laboratory contract F30601-00-2-0592 to Carnegie Mellon University.

## 8. REFERENCES

[1] CMU Text Learning Group. http://www-2.cs.cmu.edu/afs/cs/project/theo-4/text-learning/www/.

[2] L. D. Baker and A. K. McCallum. Distributional clustering of words for text classification. In *Proceedings of ACM International Conference on Research and Development in Information Retrieval (SIGIR-98)*, pages 96–103. ACM Press, 1998.

[3] B. Berendt, A. Hotho, and G. Stumme. Towards semantic web mining. In *Proceedings of International Semantic Web Conference (ISWC-2002)*, pages 264–278, 2002.

[4] D. Chickering. *Lecture Notes in Statistics*, chapter Learning bayesian networks is NP-complete. Springer Verlag, 1995.

[5] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, 1991.

[6] N. Friedman and M. Goldszmidt. Building classifiers using bayesian networks. In *Proceedings of National Conferrence on Artificial Intelligence (AAAI-96)*, pages 1277–1284, 1996.

[7] A. Hotho, S. Staab, and A. Maedche. Ontology-based text clustering. In *Proceedings of the IJCAI-2001 Workshop Text Learning: Beyond Supervision*, Seattle, USA, August 2001.

[8] D. Lewis. The reuters-21578 data set. http://www.daviddlewis.com/resources/testcollections/-reuters21578/.

[9] A. Maedche and S. Staab. Ontology learning for the semantic web. *IEEE Intelligent Systems*, pages 72–79, March/April 2001.

[10] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

[11] A. McCallum and K. Nigam. Employing EM and pool-based active learning for text classification. In *Proceedings of International Conference on Machine Learning (ICML-98)*, pages 359–367, 1998.

[12] M. Pazzani and D. Billsus. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27(3):313–331, 1997.

[13] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[14] J. Rennie and A. McCallum. Using reinforcement learning to spider the web efficiently. In *Proceedings of International Conference on Machine Learning (ICML-99)*, pages 335–343, 1999.

[15] M. Sahami. Learning limited dependence bayesian classifiers. In *Proceedings of International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pages 335–338. AAAI Press, 1996.

[16] U. Shah, T. Finin, A. Joshi, R. S. Cost, and J. Mayfield. Information retrieval on the semantic web. In *Proceedings of Conference on Information and Knowledge Management (CIKM-2002)*, pages 461–468, 2002.

[17] Y. Yang and J. O. Pederson. A comparative study on feature selection in text categorisation. In *Proceedings of International Conference on Machine Learning (ICML-97)*, pages 412–420. Morgan Kaufmann Publishers, 1997.