Two-Stage Text Classification Using Bayesian Networks

ABSTRACT

The "curse of dimensionality" provides a powerful impetus to explore alternative data structures and representations for text processing. This paper presents a method for preparing a dataset for classification by determining the utility of a very small number of related dimensions via a Discriminative Multinomial Naive Bayes process, then using these utility measurements to weight these dimensions for use in a Bayesian network classifier. We show that the use of this two-stage methodology provides significant improvements over both Discriminative Multinomial Naive Bayes and Bayesian network classifiers alone.

1. INTRODUCTION

Text processing (classification, information retrieval, etc.), in many areas provides an excellent example of the "curse of dimensionality" [1]. Proper, useful, and efficient solutions to this curse, i.e., solutions that allow data to be effectively represented using a smaller number of dimensions are desirable because they allow, for example, for data to be indexed more efficiently for retrieval and classification. To this end, a variety of data structures now include semantic data, usage statistics, and many other details to enhance the efficacy of a smaller number of dimensions.

An enhancement to text classification efficiency using a constrained dimensionality can be achieved using an enhanced data structure, the cardinality of whose dimensions are influenced by an initial distribution which is obtained via the Discriminative Multinomial Naive Bayes classifier. For final classification, a Bayesian network classifier is be brought to bear on this enhanced data structure.

This paper is organized as follows: The next section is a review of work relating to text classification and Bayesian networks. This review is followed by the definition of some prerequisite concepts. Then a detailed account of our approach to the classification task, including the general system architecture is given. Following this, the experimental methodology and results are given. Lastly, some conclusions from these experiments, details of the contributions of this paper, and suggestions for future work are given.

2. BACKGROUND

2.1 Bayesian Networks

Bayesian networks [2] are directed acyclic graphs that represent joint probability distributions over a set of random variables [3]. They are convenient in that they provide an intuitive and compact representation of the joint distribution of this set of variables, and they expose ways to utilize their dependencies to perform statistical inference.

Each node in a Bayesian network corresponds to a random variable in the domain and the directed edges between the nodes each correspond to a node's parents' influence on that node. Built into the graphical structure are ways to exploit the conditional independence properties of the distribution. Equation 1 shows the general assumption of conditional independence in Bayesian networks.

$$X_i : (X_i \perp NonDescendants_{X_i} \mid Pa(x_i)) \tag{1}$$

That is, each variable X_i is conditionally independent of its non-descendants given its parents. Then the joint distribution of each random variable X_i in a Bayesian network may be represented as in equation 2.

$$P(x_1, \ldots, x_n) = \prod_i P(x_i \mid Pa(x_i)) \tag{2}$$

Using a Bayesian network as a classifier is simply a matter of calculating the probability of a particular class given the values of the remaining variables in the network. Inference over a Bayesian network classifier proceeds by the calculation shown here.

$$\text{argmax}_y P(y|\mathbf{x}) \tag{3}$$

where

$$P(y \mid \mathbf{x}) = \frac{P(U)}{P(\mathbf{x})} \propto P(U) = \prod_{u \in U} p(u \mid Pa(u)) \qquad (4)$$

where y is the class variable, $\mathbf{x}$ is the set of observed attribute variables (the set of givens), and U is a set of variables [4].

For inference where there are unobserved or missing values for variables, other methods such as Expectation Maximization, Gibbs Sampling, and Gradient ascent are used [3]. These methods rely on partial data and conditional probability tables to conduct inference over these incomplete representations.

2.2 Related Work

2.2.1 Probabilistic Text Classification

Lam and Low [5] constructed a Bayesian network text classifier automatically from text training documents. The conditions that they place on the Bayesian network structure are the same conditions to which we will adhere. These conditions are that the edges in the network that exist between a class and a feature must run from the class to the feature, i.e., the class nodes have no parents.

Klopotek et. al. [6, 7] presented a method for using tree-like large Bayesian networks to conduct efficient inference for text classification. Tree-like Bayesian networks are Bayesian networks that contain root nodes and single parent nodes for each non-root. They utilized these networks to manage the enormous number of variables involved in the text classification task, since reasoning in these structures is "orders of magnitude simpler than in general-type Bayesian networks" [7]. They showed that these networks enabled them to relax the vocabulary restrictions that were necessary, based on the high dimensionality that characterizes text information, to make the aforementioned Bayesian inference tractable.

2.2.2 Link-Based Classifiers

The majority of the recent work involving probabilistic graphical treatments of text classification pertains to link analysis, i.e. hyperlinks in web documents, for the classification of web pages.

Motivated by the then-emerging interest in hypertext mining, Getoor and Lu [8] proposed "...a framework for modeling link distributions" and used it to "improve classification accuracy" [8]. Popescul, et al. [9] used statistical relational learning to predict where (i.e. in which journals, conference proceedings, etc.) scientific papers will be published. Their approach was based on word counts, citations, co-citations, and word co-occurrences. As with the aforementioned approach, this approach emphasized link analysis above the other dimensions.

2.2.3 Taxonomically-Enhanced Data Structures

Caragea, et al. [10] proposed a classifier based on an ontology-extended data structure for classifying semi-structured data. Their ontology essentially consisted of an "Abstraction Hierarchy" that the authors do not specify but define mathematically. We used the WordNet [11] taxonomy for this purpose, without considering links between documents.

Hossain, et al. [12] used WordNet to create "document graphs" which are a form of instantiated taxonomy used for graph-based hierarchical agglomerative clustering. [omitted for anonymity], et al. used abstraction analysis based on WordNet hypernyms for both information retrieval and text classification. The work of Hossain, [omitted for anonymity], etc. was geared towards unstructured data, as this work will be also.

3. APPROACH

This investigation approaches the text classification problem from a probabilistic perspective. In the beginning, the dimensionality of a dataset is substantially reduced. As one example, using two separate techniques of dimensionality reduction (for comparison) on the 20 Newsgroups dataset [13], the dimensionality is reduced from 15,947 to 1, 3, 17, and 194. These particular numbers were chosen because they correspond to taxonomic abstraction levels in WordNet [11] (see section 3.1.1). After this reduction we use the resulting distribution of the dimensions, obtained from the training data via the Discriminative Multinomial Naive Bayes Classifier for Text (DMNBtext, see section 3.3.3), as they relate to the classifications themselves to perform final classification.

3.1 Dimensionality Reduction Strategies

Two techniques for dimensionality reduction are used in this investigation: taxonomic abstraction and Latent Semantic Analysis.

3.1.1 Taxonomic abstraction

Taxonomic abstraction is the process of determining superordinate and subordinate relationships between words. For indexing and classification purposes, abstraction considers the superordinate words corresponding to the keywords in the corpus. The general idea is that profiles of combinations of superordinate concepts may yield a degree of referential precision that is useful for classification, since their subordinate terms influence the degree to which these superordinate terms are represented in a corpus.

WordNet [11] is one such hierarchy, and its acyclicity with regard to subordinate and superordinate relationships is ensured by its authors. The acyclicity property is important because it allows the easy construction of the abstraction paths, or, paths of taxonomic derivation that are used in this investigation. There are many other types of relationships that WordNet represents using all standard parts of speech, but for simplicity this investigation makes use, solely, of the structure relating to nouns. Another advantage of using the noun hierarchy is that the entire hierarchy has a single root. Abstraction in a single-rooted hierarchy tends towards the creation of term-document matrices that are less sparse, since all abstractions tend towards the same root.

Figure 1 depicts four of the types of relationships among nouns in WordNet. These nouns are organized in synsets, i.e. sets of synonyms. Shown are *hypernymy* / *hyponymy*, which is analogous to superordinate / subordinate relationships in that a hypernym is a more general form of its hyponyms; synonymy, which characterizes words that have similar meanings; and polysemy, which characterizes a word (or synset) having a multitude of meanings.

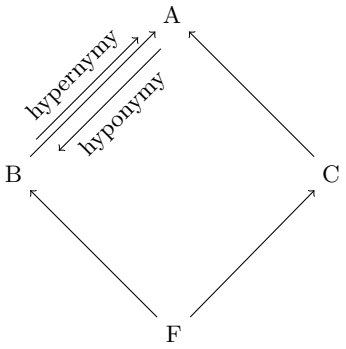

Figure 1: The types of relationships in WordNet.

3.1.2 Latent Semantic Analysis

[14] Latent Semantic Analysis (LSA) may be used to collapse the dimensionality of an index for a corpus of documents by statistically inferring relationships between words, and using the words having greater variance, in terms of term-document representation, for the items in the new index. It uses singular value decomposition (SVD) applied to a term-document matrix to determine the dimensions that are more likely not to represent noise in the corpus and allows the construction of an index whose elements are more likely to be useful in the differentiation of documents.

For reducing the dimensionality of a document corpus LSA produces a low-rank approximation of a term-document matrix. Using SVD it prioritizes the dimensions of the corpus from those having greatest variance to those having the least. When this is completed the user (the classification process) has the option of choosing the top n dimensions to be included in the resulting classification data structure.

3.2 System Architecture

Figure 2 depicts a high-level overview of the process. The connections between each of the processes are labeled with the entity that is transferred between the processes. The contents of each of these labels are defined as follows.

The object passed from the Preprocessing module to the Dimensionality Reduction module is the Term-Document Matrix (TDM). This is a matrix M of term-document vectors m where $m_{i,j}$ is the measure of word relevance of word i to document j. For this investigation, the TDM is comprised of word vectors that use the Term Frequency × Inverse Document Frequency (TF×IDF) measure as their elements.

$$TF(w_d, d) = \frac{|w_d|}{|d|} \tag{5}$$

$$IDF(w_d) = \log\left(\frac{|D|}{|D_w|}\right) \tag{6}$$

$$TFxIDF(w_d, d) = TF(w_d, d) \times IDF(w_d) \tag{7}$$

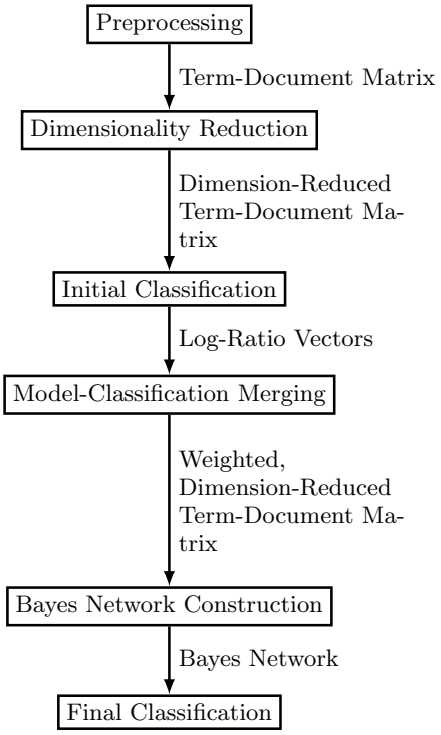

Figure 2: Process Overview

It should be noted here that the TDM is a very sparse matrix because documents normally use only a very small fraction of the words in the language in which they have been written. Indeed, an effect of a dimensionality reduction using both taxonomic abstraction and LSA is to lower the sparseness of this representation.

For dimensionality reduction via taxonomic abstraction the Document Graph data structure is utilized. A Document Graph ($DG(V, E)$) is a structure created by aggregating the hypernym superstructures (paths to the root) of all nouns in a document. Each vertex $v \in V$ represents a word in WordNet and each edge $e \in E$ represents its connections from its hypernyms and to its hyponyms.

Figure 3 shows a document graph for a hypothetical document containing two words: 'bug' and 'microphone'. Notice that there are two very different senses of the word 'bug' shown here; that of the insect and that of the miniature microphone from spy movies. Both sense paths are traced to the root. The assemblage of all of these paths for a document is the Document Graph.

The values in the graph reflect artificial TF×IDF values for the sentence containing the words 'bug' and 'microphone'. Before graphing, the word 'bug' has the TF×IDF value 0.85 and the word 'microphone' has the TF×IDF value of 0.35. Since, in this example, the word 'bug' has two hypernyms, namely 'insect' and 'microphone', 'bug' divides its support (initially the TFxIDF value) evenly among both, adding half of its support (0.425) to each hypernym. Since 'microphone' already has a support of 0.35, its support becomes 0.35 +

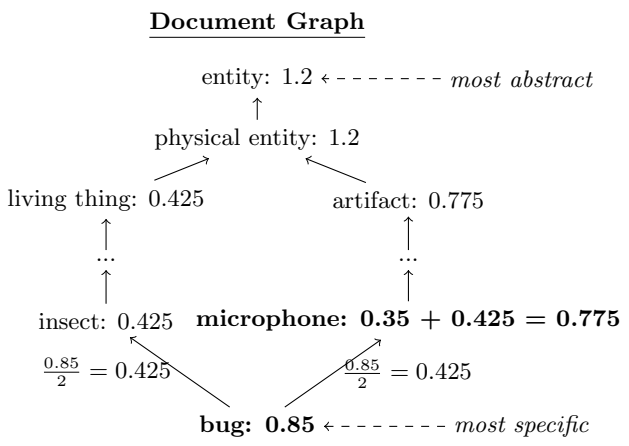

Figure 3: Example of a Document Graph for a document containing two keywords: 'bug' and 'microphone'.

	1	2	3
	entity	entity	entity
	physical entity	physical entity	physical entity
	object	object	object
	unit	unit	unit
	living thing	artifact	artifact
	organism	instrumentation	instrumentation
	animal	device	device
	invertebrate	electrical device	electrical device
	arthropod	transducer	transducer
	insect	electroacoustic transducer	electroacoustic transducer
	bug	microphone	microphone
		bug	
support:	**0.425**	**0.425**	**0.35**

Table 1: Full Abstraction Paths for 'bug' and 'microphone'

	1	2	3
	entity	entity	entity
	physical entity	physical entity	physical entity
	object	object	object
	unit	unit	unit
	living thing	artifact	artifact
	organism	instrumentation	instrumentation
support:	**0.425**	**0.775**	

Table 2: Paths for 'bug' and 'microphone' truncated at a depth of 3. Columns 2 and 3 represent the same path at this truncation level, therefore the support of this combined path becomes the sum of the original paths shown in columns 2 and 3 of table 1.

0.425 = 0.775 due to the contribution of half of the support of the word 'bug'. The rest of the values represent a propagation of TF×IDF values from hyponym to hypernym, with all values converging at the word 'physical entity' since both 'living thing' and 'artifact' are physical entities. 'entity' is the most abstract noun in WordNet, so every noun will converge there.

In this paradigm, the presence of the word 'microphone' has the effect of adding a layer of semantic detail to the word 'bug.' In this situation, a probable disambiguation of the word 'bug' would place it in the path containing the word 'artifact' rather than the one describing a form of a living thing. This is because the word 'microphone' adds its support, or TF×IDF value of 0.35 to that path, enhancing the support of all words above it.

Abstraction Paths are extracted from the DGs in order to obtain a profile for the desired level of abstraction. An Abstraction Path (AP) is a path in a document graph from leaf to root or from a location of changing support to root containing all vertices along that path and the weight of the vertex most distant from the root. Table 1 shows the three abstraction paths that emanate from the word 'bug' and the word 'microphone' in the document graph in figure 3.

The Dimension-Reduced Term-Document Matrix ($DRTDM$) is the object that is passed between the Dimensionality Reduction module and the Initial Classification module. The $DRTDM$ is a TDM in which some dimensions have been deleted or combined, resulting in a TDM of lower dimensionality. Given that the objective in this investigation is to achieve a useful index of a manageable dimensionality, we seek to create a usable DRTDM.

Another concept appropriate to dimensionality reduction via taxonomic abstraction is that of Abstraction Level. This is the level in the hierarchy, coming from the root, that is used in the abstraction analysis. As an illustration, table 2 shows the three abstraction paths that emanate from the word 'bug' and the word 'microphone' truncated at a depth of 6. As can be seen, at this level paths 2 and 3 lead to the same word. The internal representation of this phenomenon is the coalescence of both paths into one single path, using the sum of their original supports as the support for this new path. This is internally represented as a coalescence of paths because each member of the path may be weighted differently, though the progression of the weights from most specific to most abstract remains monotonic.

Here, the word 'instrumentation' ends up representing a greater proportion of the words in the document than the word 'organism' as both 'bug' and 'microphone' have contributed their supports to this path. Also, notice that this is one level after any distinction is made in the paths representing microphones and insects respectively. In other words, this is the level at which the distinction between the words 'bug' and 'microphone' become apparent.

The Log-Ratio Matrix (LRM) obtained via the processes in the Initial Classification module is a matrix in which each

element $l_{i,j}$ is the result of the calculation of equation 8 where $P(t_i \mid c_j)$ is the probability that a document contains term t_i given the document is in class c_j and $P(t_i \mid \neg c_j)$ is the probability that a document contains term t_i willgiven the document is not in class c_j. The LRM is obtained via the DMNBtext classification algorithm (see section 3.3).

$$l_{i,j} = \log\left(\frac{P(t_i \mid c_j)}{P(t_i \mid \neg c_j)}\right) \tag{8}$$

For construction of the Bayesian network for final classification, the Weighted, Dimensionally Reduced Term-Document Matrix (WDM) is used. The WDM is a $DRTDM$ in which each element has been multiplied by its corresponding entry in the LRM. It is the element-by-element product, rather than the matrix product.

Equations 9 through 11 depict the LRM, $DRTDM$, and WDM. $C(d_n)$ is the class into which document n belongs. $R(l_{t_0,C(d_0)}, w_{t_0,d_0})$ is the term support for document d_0 times the weight of that term for the class as determined by the LRM.

$$\mathbf{LRM} = \begin{bmatrix} l_{t_0,c_0} & \cdots & l_{t_k,c_0} \\ \vdots & \ddots & \vdots \\ l_{t_0,c_m} & \cdots & l_{t_k,c_m} \end{bmatrix} \tag{9}$$

$$\mathbf{DRTDM} = \begin{bmatrix} w_{t_0,d_0} & \cdots & w_{t_k,d_0} \\ \vdots & \ddots & \vdots \\ w_{t_0,d_n} & \cdots & t_{t_k,d_n} \end{bmatrix} \tag{10}$$

$$\mathbf{WDM} = \begin{bmatrix} R(l_{t_0,C(d_0)}, w_{t_0,d_0}) & \cdots & R(l_{t_0,C(d_0)}, w_{t_k,d_0}) \\ \vdots & \ddots & \vdots \\ R(l_{t_0,C(d_n)}, w_{t_0,d_n}) & \cdots & R(l_{t_k,C(d_n)}, w_{t_k,d_n}) \end{bmatrix} \tag{11}$$

3.3 Implementation

3.3.1 Preprocessing

The preprocessing step proceeds as usual for any text classification or information retrieval task. The corpus is read into a database and the TDM is extracted via calculations of the TF×IDF values of each of the dimensions in relation to each of the documents. For this process we used RapidMiner's WVTool (Word Vector Tool) [15].

3.3.2 Dimensionality Reduction

Dimensionality reduction, as stated in section 3, proceeds apriori, i.e., before any classification task has been brought to bear. Both dimensionality reduction techniques are applied individually and not in combination with one another for separate lines of experimentation.

Taxonomic Abstraction: Abstraction proceeds primarily by constructing Document Graphs for all documents in the training set and extracting abstraction paths from them. After this, the paths are cut at a predetermined level that is passed into the system by the user. Due to the system resource constraints, the system was only tested up to abstraction depth of 4.

depth	dimensionality
1	1
2	3
3	17
4	194

Table 3: WordNet Dimensionality By Taxonomy Depth

Algorithm 1 Log-Ratio Matrix / Training Data Merging

Require: LRM $\mathbf{W}$ contains entries $\mathbf{w}$ for each attribute-class pair and TDM $\mathbf{T}$ contains entries $\mathbf{t}$ for each term-document pair

Ensure: In the resulting matrix, each element is weighted corresponding to its relevant entry in the LRM

```
for each t ∈ T do
    let w = the LRM entry for class
    for each t ∈ t do
        t = t × w_t
    end for
    WDM = WDM ∪ t
end for
return WDM
```

Table 3 shows the dimensionality of the words contained in WordNet as the depth increases. The actual dimensionality may differ because not all paths are represented for all corpi.

Latent Semantic Analysis (LSA): For dimensionality reduction via latent semantic analysis [14] the SSpace package from Airhead Research [16] was used. The same dimensionality that would result from cutting the abstraction paths at the levels described in table 3 was used (1, 3, 17, and 194 dimensions). The reasoning here is that a direct comparison of the effectiveness of the number of dimensions can be achieved if the two techniques are compared using the same number of dimensions. Since dimensionality reduction via taxonomic abstraction at, for example, level 2 produces 3 dimensions, the same number of dimensions will be selected from the dimensional reduction via latent semantic analysis.

After dimensionality reduction, each entry in the respective $DRTDM$ is replaced with its corresponding z-value.

3.3.3 Initial Classification

Initial classification proceeds using Discriminative Multinomial Naive Bayes for Text Classification (DMNB) [17]. DMNB differs from traditional Naive Bayes classification in that it considers both likelihood information ($P(x \mid Pa(x))$) and prediction error ($P(class \mid instance) - \hat{P}(class \mid instance)$) in its frequency estimation for parameter learning. Therefore it is a combination of both generative and discriminative learning [17].

This step provides information about how each of the dimensions, refined in the dimensionality reduction step, influences the class membership of each document.

3.3.4 Model-Classification Merging

Algorithm 1 creates the Weighted, Dimensionally Reduced Term-Document Matrix. This matrix is the final representation for the terms and documents going into the Bayesian network classifier.

3.3.5 Bayes Network Construction

The K2 search algorithm [18] is used to learn the structure of the Bayesian network. This is a hill climbing algorithm that adds arcs with a fixed ordering of the variables. Parameterization of this algorithm in Weka allows it to start the network as a Naive Bayes network and proceed from there. It also allows the user to designate the maximum number of parents that each node may have. For these experiments a maximum of 10 parents was chosen. This had the effect that was the same as choosing no limit on the number of parents.

Figure 4 is a graphical depiction of the Bayesian network that resulted from the analysis at WordNet's 3rd level. Using hypernyms at this level results in a dimensionality of 17. As can be seen, each synset has an edge leading directly from the *group* vertex. This is a result of this network starting as a naive Bayes network.

3.3.6 Final Classification

SimpleEstimator is used for estimating the conditional probability tables of a Bayes network once the structure has been learned. This method estimates probabilities directly from data [4]. It does this by calculating the class membership properties given each test instance.

The Bayesian network in figure 4 expresses some interesting relationships among hypernyms in the corpus. According to the network, there is a V structure with, for example, the word 'set' at the junction. This would suggest, as expected, that the presence of the word 'process' as a hypernym representation in a document activates the path from 'process' to 'group' via the word 'set' even though 'process' is also directly connected to 'group'. Being siblings in the WordNet hierarchy, 'process' and 'set' have a relationship that does not seem to portend this new relationship. Such properties may, in the future, be used to form a new, data-driven taxonomy that does not rely on lexicographers (domain experts) for its topology.

4. EXPERIMENTS AND EVALUATION

4.1 Datasets

Our experiments were conducted on two datasets: the 20 Newsgroups dataset [13] and the Reuters-21578 dataset. The use of the 20 Newsgroups dataset is convenient because it provides a default document labeling via the newsgroups to which each conversation was posted. Also, it provides an interesting example of noisy, unstructured data.

The Reuters-21578 Text Categorization Test Collection [19] is a collection of documents that appeared on the Reuters newswire service in the year 1987. The Collection provides categories of general subject matter for the documents. The use of this dataset was less intuitive for this investigation, as the categorization is not mutually exclusive, i.e. there may be more than one category per document. To mitigate this, we used the approach of Qian et al. [20], where documents from the top-10 largest categories of the 'ModApte' split of this dataset are used.

For training and testing, the datasets were divided in half with one half being used to obtain the log-ratio matrix via the DMNBtext classifier, and the other being used for 10 fold cross-validation with the Bayesian network classifier.

4.2 Results

Figure 5 shows the values of the correctly classified instances for each experiment we ran for the 20 Newsgroups dataset. Figure 6 shows the same for the Reuters dataset. Depicted in both charts for comparison are the results when only the Bayesian network classifier was run on both abstraction and latent semantic analysis configurations (Abstraction BayesNet and LSA BayesNet, respectively). As can be seen, the Bayesian network classifier, when using a Weighted, Dimensionally Reduced Term-Document Matrix (WDM) radically increases the percentage of correctly classified instances. For just 194 dimensions, abstraction indexing produces a correctly classified percentage of 96.64%. The same process with dimensionality reduction via LSA produces comparable results, but peaks at a dimensionality of 3 with a correctly classified percentage of 93.3%.

On a class-by-class basis, the F-measure was instructive.

$$F_\beta = \frac{(\beta^2 + 1) * recall * precision}{\beta^2 * precision + recall} \tag{12}$$

where $\beta = 0$ means that F_β is the same calculation as precision, and $\beta = 1$ means that F_β puts equal weight on precision and recall. Since we are interested in enhancing both measures we used $\beta = 1$.

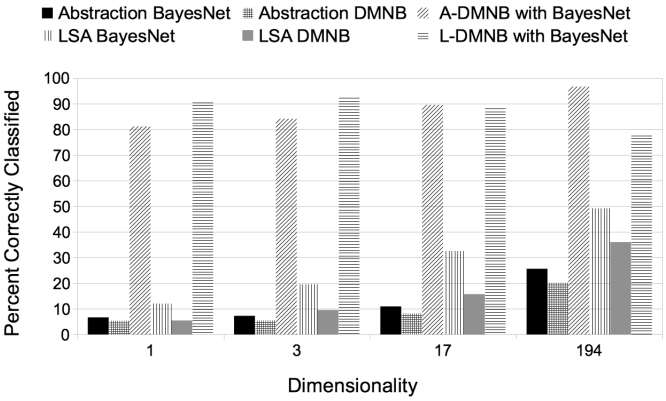

Figure 5: Classification Accuracy for the 20 Newsgroups dataset

Tables 4 through 7 show the class-by-class F-measure comparison for the experiments we conducted using the 20 News-

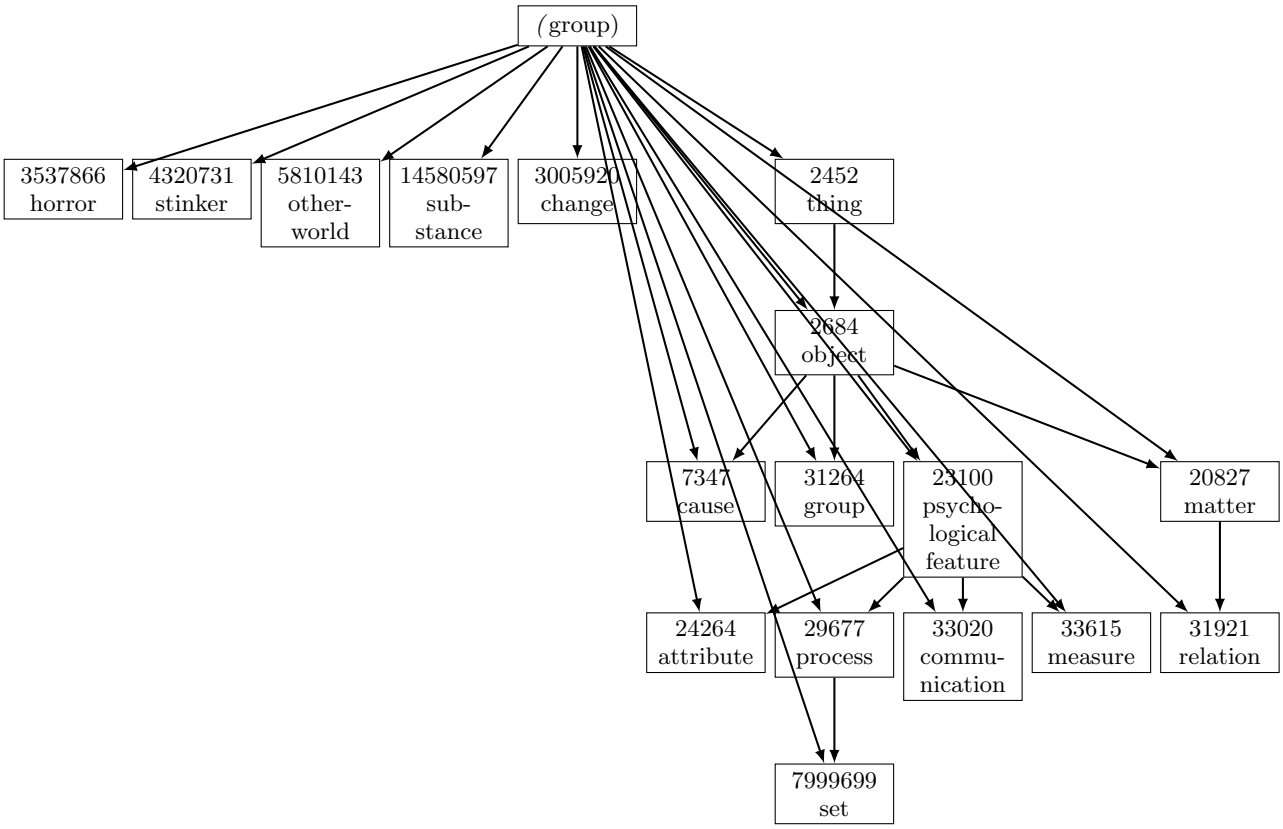

Figure 4: Bayesian Network Depicting Hypernym Relationships

groups dataset. Notice that, even at a dimensionality of 1, data model enhancement using the Bayesian network classifier allowed the correct classification of some documents into some classes that had not even been represented without the *a-posteriori* Bayesian network classification. This is exactly the desired effect. This suggests that the posterior distribution of the classes (after DMNB model construction) affected classification with respect to the entire collection of classifications.

As can be seen in tables 4 and 6, the majority of the F-measure values are 0 for a dimensionality of 1. The configuration of entries of 0 is different in both of these tables for a dimensionality of 2. Sometimes, as in the case of the 'rec.autos' row of table 4, the F-measure drops to 0 as the dimensionality increases from 1 to 3. This can be explained by the fact that the 3 dimensions represented in the second column do not include the dimension represented in the column corresponding to 1 dimension. The same is the case for 6. The dimensions represented in the configuration of 3 dimensions do not necessarily include the dimension represented in the configuration of 1 dimension.

Tables 8 through 11 show the class-by-class F-measure comparison for the experiments we conducted using the Reuters 21578 dataset. These experiments are in line with what was expected following the experiments with the 20 Newsgroups dataset. It should be noted, however, that class membership in the Reuters 21578 dataset is not mutually exclusive. The results for the experiments with this dataset are summarized in tables 8 through 11.

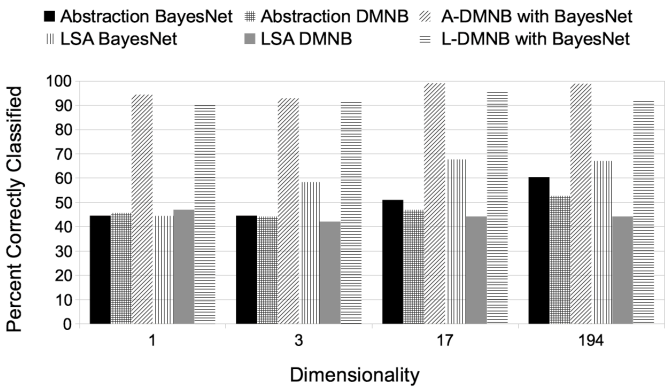

Figure 6: Classification Accuracy for the Reuters-21578 Dataset

5. CONCLUSIONS AND CONTRIBUTIONS

We conclude that Bayesian network classifiers provide an excellent option for text classification, when the importance of each term to each class can be established. Weighting the terms properly, after the initial model construction, can

	Dimensionality			
	1	**3**	**17**	**194**
rec.motorcycles	0.094	0.086	0.107	0.11
comp.sys.mac. hardware	0	0	0.02	0.134
talk.politics.misc	0	0	0	0.064
soc.religion.christian	0.084	0.053	0.129	0.221
comp.graphics	0	0	0.098	0.216
sci.med	0	0.01	0	0.131
talk.religion.misc	0	0	0	0
comp.windows.x	0	0	0.13	0.328
comp.sys.ibm.pc. hardware	0	0	0.004	0.205
talk.politics.guns	0	0	0	0.113
alt.atheism	0	0	0	0.01
comp.os.ms-windows. misc	0	0	0.076	0.213
sci.crypt	0	0.039	0.043	0.256
sci.space	0.069	0.102	0.035	0.355
misc.forsale	0	0	0.142	0.224
rec.sport.hockey	0	0.091	0.071	0.264
rec.sport.baseball	0	0	0.095	0.182
sci.electronics	0	0	0.097	0.153
rec.autos	0.097	0	0.011	0.143
talk.politics.mideast	0	0	0.04	0.274
weighted average:	0.019	0.021	0.058	0.186
percent correctly classified:	5.47	5.60	8.32	20.29

Table 4: DMNBtext on Abstraction Indexing for the 20 Newsgroups Dataset

	Dimensionality			
	1	**3**	**17**	**194**
rec.motorcycles	0.551	0.705	0.832	0.98
comp.sys.mac. hardware	0.59	0.579	0.7	0.948
talk.politics.misc	1	1	0.998	0.996
soc.religion.christian	0.984	0.984	0.985	0.98
comp.graphics	0.911	0.911	0.952	0.947
sci.med	0.918	0.941	0.938	0.986
talk.religion.misc	1	1	1	0.998
comp.windows.x	0.583	0.574	0.818	0.962
comp.sys.ibm.pc. hardware	0.683	0.672	0.662	0.971
talk.politics.guns	0.989	0.992	0.991	0.992
alt.atheism	0.996	0.999	0.991	0.985
comp.os.ms-windows. misc	0.751	0.786	0.871	0.952
sci.crypt	0.798	0.899	0.944	0.988
sci.space	0.957	0.972	0.975	0.991
misc.forsale	0.587	0.613	0.774	0.834
rec.sport.hockey	0.793	0.887	0.931	0.967
rec.sport.baseball	0.645	0.764	0.848	0.96
sci.electronics	0.859	0.894	0.896	0.984
rec.autos	0.722	0.773	0.85	0.967
talk.politics.mideast	0.999	0.991	0.998	0.987
weighted average:	0.81	0.84	0.894	0.968
percent correctly classified:	81.10	84.12	89.46	96.64

Table 5: DMNBtext Preprocessing Abstraction Indexing With Bayes Net Classifier for the 20 Newsgroups Dataset

also allow the classification, by Bayesian network classifier, to correctly classify documents into classes that would have been neglected by the initial classification procedure. We show that proceeding in this fashion enables the production of excellent results at a very low dimensionality.

In this work we have provided two main contributions. The first is a method of enhancing accuracy of text classification where the classification dimensions have been drastically reduced. This is important in that it may save index space and facilitate pruning of the document search space. The second contribution is information about how the accuracy of classification varies as the level of abstraction varies. This contribution may be seen in tables 4 through 11, where the results are broken down by dimensionality.

For those informed by work in ontologies and taxonomies, this work also has implications in the area of extracting semantic information from the Bayesian networks used in the final classification. In other words, probabilistic information extracted using the Bayesian network construction, we hypothesize, may be used to enhance the taxonomies that are used to restrict the dimensionality. Use of the probabilities in this way may lead to the construction of data structures that are more useful for information retrieval and the classification task.

6. FUTURE WORK

Abstraction and the use of ontologies provides an excellent basis for application in information retrieval. Figure 4 shows interesting lexical relationships exposed by the construction of the network. An addition to the WordNet taxonomy that includes these relationships may be instructive and may lead to some advances in classification inference that can be conducted using the enhanced taxonomy as a reference.

Also, from a use case perspective, exploration into integrating both classification steps into a comprehensive structure for inference perhaps may lead to the construction of a viable system for classification refinement in domains that require such things. For example, the Bayesian treatment of derivational taxonomies may be brought to bear to yield useful insight into the structural profile and usage characteristics of specific text corpora. Since text processing is a field that deals with very large and dynamic datasets, exploration of dynamic updating options for large, constantly changing text corpora would also be useful.

7. REFERENCES

[1] R. Bellman, *Dynamic Programming*, ser. Dover Books on Mathematics. Dover Publications, 2003. [Online]. Available: http://books.google.com/books?id=fyVtp3EMxasC

[2] J. Pearl, *Probabilistic reasoning in intelligent systems - networks of plausible inference*, ser. Morgan

	Dimensionality			
	1	**3**	**17**	**194**
rec.motorcycles	0	0.074	0.023	0.448
comp.sys.mac. hardware	0	0	0.142	0.367
talk.politics.misc	0	0	0	0.09
soc.religion.christian	0.069	0	0.166	0.479
comp.graphics	0	0	0.123	0.371
sci.med	0	0.023	0	0.218
talk.religion.misc	0	0	0	0
comp.windows.x	0.087	0.166	0.402	0.468
comp.sys.ibm.pc. hardware	0	0.048	0.335	0.332
talk.politics.guns	0	0	0	0.393
alt.atheism	0	0	0	0.07
comp.os.ms-windows. misc	0	0.416	0.398	0.461
sci.crypt	0.087	0.095	0.032	0.647
sci.space	0.084	0.093	0.028	0.378
misc.forsale	0	0.011	0.125	0.337
rec.sport.hockey	0.086	0.122	0.131	0.538
rec.sport.baseball	0.071	0	0.288	0.297
sci.electronics	0	0	0	0.087
rec.autos	0	0	0.131	0.471
talk.politics.mideast	0	0	0	0.53
weighted average:	0.026	0.057	0.124	0.364
percent correctly classified:	5.45	9.53	15.72	36.01

Table 6: DMNBtext on Latent Semantic Analysis for the 20 Newsgroups Dataset

	Dimensionality			
	1	**3**	**17**	**194**
rec.motorcycles	0.955	0.902	0.947	0.743
comp.sys.mac. hardware	0.82	0.842	0.733	0.613
talk.politics.misc	1	1	0.995	0.986
soc.religion.christian	1	1	0.981	0.95
comp.graphics	0.984	0.984	0.915	0.625
sci.med	0.999	0.996	0.996	0.914
talk.religion.misc	1	1	1	1
comp.windows.x	0.718	0.8	0.851	0.743
comp.sys.ibm.pc. hardware	0.697	0.781	0.711	0.514
talk.politics.guns	1	1	0.993	0.96
alt.atheism	1	1	0.988	0.967
comp.os.ms-windows. misc	0.759	0.899	0.834	0.582
sci.crypt	0.946	0.98	0.881	0.839
sci.space	1	1	0.985	0.916
misc.forsale	0.695	0.742	0.718	0.49
rec.sport.hockey	0.983	0.979	0.817	0.869
rec.sport.baseball	0.964	0.915	0.856	0.853
sci.electronics	0.954	0.989	0.968	0.816
rec.autos	0.872	0.897	0.748	0.649
talk.politics.mideast	1	0.999	0.992	0.975
weighted average:	0.914	0.933	0.892	0.791
percent correctly classified:	91.46	93.30	89.10	78.58

Table 7: DMNBtext on Latent Semantic Analysis With Bayes Net Classifier for the 20 Newsgroups Dataset

	Dimensionality			
	1	**3**	**17**	**194**
earn	0.628	0.614	0.642	0.751
wheat	0	0	0	0
money-fx	0	0	0	0
corn	0	0	0	0
trade	0	0	0	0
acq	0	0	0.197	0.368
grain	0	0	0	0.178
interest	0	0	0	0
crude	0	0	0	0
ship	0	0	0	0
weighted average:	0.287	0.272	0.329	0.43
percent correctly classified:	45.74	44.26	47.02	52.77

Table 8: DMNBtext on Abstraction Indexing for the Reuters ModApte Split

Kaufmann series in representation and reasoning. Morgan Kaufmann, 1989.

[3] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[4] (2011, April) Weka documentation. [Online]. Available: http://www.cs.waikato.ac.nz/ml/weka/index_documentation.html

[5] *Automatic document classification based on probabilistic reasoning: model and performance analysis*, vol. 3, 1997. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=635349

[6] M. A. Klopotek and M. Woch, "Very large bayesian networks in text classification," in *Proceedings of the 1st international conference on Computational science: PartI*, ser. ICCS'03. Berlin, Heidelberg: Springer-Verlag, 2003, pp. 397–406. [Online]. Available: http://portal.acm.org/citation.cfm?id=1764172.1764219

[7] M. A. Klopotek, "Very large Bayesian multinets for text classification," *Future Gener. Comput. Syst.*, vol. 21, no. 7, pp. 1068–1082, 2005. [Online]. Available: http://dx.doi.org/10.1016/j.future.2004.03.007

[8] Q. Lu and L. Getoor, "Link-based text classification," in *IJCAI Workshop on Text Mining and Link Analysis*, 2003.

[9] A. Popescul, L. H. Ungar, S. Lawrence, and D. M. Pennock, "Statistical relational learning for document mining," 2003, pp. 275–282.

[10] C. Caragea, D. Caragea, and V. Honavar, "Learning link-based classifiers from ontology-extended textual data," in *IEEE International Conference On Tools With Artificial Intelligence*. IEEE Computer Society, 2009, pp. 354–361.

	Dimensionality			
	1	**3**	**17**	**194**
earn	0.995	0.957	0.998	0.99
wheat	0.938	0	0.973	1
money-fx	0.977	0.97	0.958	0.96
corn	0	0.69	0.923	0.97
trade	0.902	0.962	1	1
acq	0.934	0.94	0.979	0.99
grain	0.954	0.947	1	0.986
interest	0.857	0.974	1	0.971
crude	0.955	0.957	1	0.982
ship	0.706	0.96	1	1
weighted average:	0.93	0.909	0.989	0.987
percent correctly classified:	94.26	92.77	98.94	98.72

Table 9: DMNBtext Preprocessing Abstraction Indexing With Bayes Net Classifier for the Reuters ModApte Split

	Dimensionality			
	1	**3**	**17**	**194**
earn	0.639	0.592	0.614	0.614
wheat	0	0	0	0
money-fx	0	0	0	0
corn	0	0	0	0
trade	0	0	0	0
acq	0	0	0	0
grain	0	0	0	0
interest	0	0	0	0
crude	0	0	0	0
ship	0	0	0	0
weighted average:	0.3	0.249	0.274	0.274
percent correctly classified:	46.92	42.04	44.16	44.16

Table 10: DMNBtext on Latent Semantic Analysis for the Reuters ModApte Split

[11] Wordnet: a lexical database for the english language. [Online]. Available: http://wordnet.princeton.edu/

[12] M. S. Hossain and R. A. Angryk, "Gdclust: A graph-based document clustering technique." in *ICDM Workshops*. IEEE Computer Society, 2007, pp. 417–422. [Online]. Available: http://dblp.uni-trier.de/db/conf/icdm/icdmw2007.html#HossainA07

[13] (2009, May) Home page for 20 newsgroups data set. [Online]. Available: http://people.csail.mit.edu/jrennie/20Newsgroups/

[14] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Jounal of the American Society for Information Science*, vol. 41-6, pp. 391–407, 1990.

[15] M. Wurst. (2009, June) The word vector tool and the rapidminer text plugin. GNU Public License. [Online]. Available: http://wvtool.sf.net

[16] (2011, April) airhead-research - the s-space package: A scalable software library for semantic spaces. [Online]. Available: http://code.google.com/p/airhead-research/

	Dimensionality			
	1	**3**	**17**	**194**
earn	0.985	0.966	1	1
wheat	0.952	0.774	1	1
money-fx	0.75	0.895	0.974	0.769
corn	0	0	0.696	0.643
trade	0.87	0.895	0.963	1
acq	0.917	0.937	0.96	0.89
grain	0.944	0.907	0.986	0.983
interest	0.629	0.857	0.889	0.773
crude	0.884	0.898	0.875	0.764
ship	0.9	0.875	0.824	0.8
weighted average:	0.897	0.913	0.963	0.926
percent correctly classified:	91.08	92.36	96.39	92.78

Table 11: DMNBtext on Latent Semantic Analysis With Bayes Net Classifier for the Reuters ModApte Split

[17] J. Su, H. Zhang, C. Ling, and S. Matwin, "Discriminative parameter learning for bayesian networks," in *ICML 2008'*, 2008.

[18] G. Cooper and E. Herskovits, "A bayesian method for the induction of probabilistic networks from data." *Machine Learning*, vol. 9(4), no. 309-347, 1992.

[19] Reuters-21578 text categorization test collection. [Online]. Available: http://www.daviddlewis.com/resources/testcollections/reuters21578/

[20] T. Qian, H. Xiong, Y. Wang, and E. Chen, "On the strength of hyperclique patterns for text categorization âŃŞ," *Jounal of Information Science*, 2007.