In Lorenza Saitta, ed., *Machine Learning: Proc. of the Thirteenth International Conference*, Morgan Kaufmann, 1996.

Applying the Multiple Cause Mixture Model to Text Categorization

Mehran Sahami*
Gates Building 1A
Computer Science Department
Stanford University
Stanford, CA 94305-9010
sahami@cs.stanford.edu

Marti Hearst
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304
hearst@parc.xerox.com

Eric Saund
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304
saund@parc.xerox.com

Abstract

This paper introduces the use of the Multiple Cause Mixture Model to automatic text category assignment. Although much research has been done on text categorization, this algorithm is novel in that is unsupervised, that is, does not require pre-labeled training examples, and it can assign multiple category labels to documents. In this paper we present very preliminary results of the application of this model to a standard test collection, evaluating it in supervised mode in order to facilitate comparison with other methods, and showing initial results of its use in unsupervised mode.

1 Introduction

The popularity of searching the contents of the Internet has recently increased recognition of the need for automatic assignment of category labels to documents in large text collections. Web interfaces such as Stanford's Yahoo web search system (Yahoo! 1995) make use of manually-assigned category labels to help users understand the structure of its text collection. However, manual information is time-consuming to produce and so automated methods of category assignment are needed.

There has been a great deal of research on automatic category label assignment and great strides are being made in this arena. However, most existing algorithms are supervised; that is, they require a large training set of pre-labeled documents. (Although there has been interesting work on how to intelligently reduce the amount of training data required (Lewis & Gale 1994).)

In this paper we describe an algorithm that can automatically induce multiple category structure underlying an unlabeled document corpus. Following an unsupervised learning phase, newly presented documents can be evaluated according to the multiple category descriptors learned. This algorithm makes use of the Multiple Cause Mixture Model (Saund 1995), which has been shown to have strong results when applied to various test data sets including a small-scale mock-up of the text categorization application. In this paper we describe the application of the Multiple Cause Mixture Model (MCMM) to the automatic discovery of document categories.

Besides operating in an unsupervised manner, this algorithm differs from many text categorization models in that it provides a novel method for representing the fact that documents can often be best described by more than one topic label. Most algorithms assign only one label per document, or else treat the classification task as a sequence of dichotomous decisions where, for a given document, a binary yes-or-no decision is made for each category label in turn (Apte, Damerau, & Weiss 1994). By contrast, the MCMM attempts to place documents into multiple categories when appropriate. (Hearst 1994) discusses why multiple category assignment is important for information access.

2 Topical Clustering

We intend to capture the topics inherent in a text corpus by finding coherent clusters of lexical entries (words) within a high-dimensional space. Our *vector-based* representation encodes documents as *J*-

* This work was done while the first author was at the Xerox Palo Alto Research Center.

dimensional vectors, where each vector entry corresponds to one word. Due to constraints in the formulation of the mathematical model, these vector entries are binary; beyond a threshold number, the number of occurrences of the word does not matter. Thus, we can then view each document in this representation as simply being a point on a vertex of the J-dimensional hypercube.

Our theoretical model of document corpora is that when a document is "about" a given topic, that topic *causes* certain words to become likely to appear in the document. Typically, each topic will be associated with a substantial number of words, so over a corpus, clusters of words will become indicative of topics. If several topics are present, or true of a document, the collection of words expected to appear will be the conjunction of the words associated with each individual topic. Moreover, if more than one topic suggests a given word (e.g., a document about rivers and finance might both be expected to contain the word, "bank"), then that word would be predicted to appear with even greater likelihood than the causation contributed by either topic alone.

A mathematical model reflecting this structure is provided by the *Multiple Cause Mixture Model* (Saund 1995). Figure 1 illustrates. Activity m in cluster-layer topic nodes flows top-down to "cause" activity in nodes r, which reflect predictions of how likely individual words are to appear in the document.

When presented with word-vector data whose distribution reflects the underlying multiple cause assumption, the Multiple Cause Mixture Model can be trained in unsupervised fashion to tease apart the various constituent word-cluster subspaces. The learning algorithm is analgous to the EM algorithm for training the standard single cause Mixture Model (Duda & Hart 1973).

Although the MCMM is presented in detail elsewhere (Saund 1995), we give a brief overview here for completeness and present the assumptions of the model focusing on text categorization.

2.1 The Multiple Cause Mixture Model

The Multiple Cause Mixture Model (MCMM) shares with other cluster style models the core idea that regularities in high dimensional data may be captured by associating data vectors with unobserved (hidden) clusters. It differs from other models by permitting clusters not only to compete for data points, but also to cooperate with one another in accounting for observed data.

The fundamental operation of the MCMM is propagation from a vector of K beliefs or activities m ($0 \leq m_k \leq 1$) at the cluster layer to a J-vector of data value predictions, r, according to weights c. In general many different options are available for the *mixing function* $r_j = r_j(m, c)$. For the present purposes we choose for the mixing function, *soft disjunction*, also known as Noisy Or (Pearl 1988):

$$r_j = 1 - \prod_k (1 - m_k c_{j,k}).$$

Thus $0 \leq c_{j,k} \leq 1$ and $0 \leq r_j \leq 1$. This choice of mixing function reflects the interpretation that the propensity for any given word to appear in a document only increases with the presence of activity in multiple topic nodes capturing the document's topical content.

Although the inherent directionality of the model is from beliefs in clusters (topics) to predictions of data (words), the inverse, *measurement*, computation can be performed as follows. Any prediction vector r can be compared with a binary vector d representing the words actually present in a document ($d_j \in \{0, 1\}$). The accuracy of the prediction is reflected by a log-likelihood objective function:

$$g = \sum_j \log[d_j r_j + (1 - d_j)(1 - r_j)].$$

Given an observed binary data vector d, it is possible through gradient ascent to find the vector of cluster activities m that optimizes this objective function.

Consider a corpus of I documents (indexed by i) represented by J-dimensional binary word vectors d_i. Then the objective of unsupervised training of the model is to arrive at a set of weights c reflecting clusters of words that co-occur in documents such that the global objective function G is maximized over the entire corpus,

$$G = \sum_i g(i).$$

Training begins by initializing a single cluster centroid as a random point in $[0, 1]^J$ space. For this initial cluster centroid and for the later stages in which there are several cluster centroids (that is, $K > 1$), the maximization is performed in two steps. First, the cluster centroid(s), c, are fixed and gradient ascent is used to find appropriate values of $m_{i,k}$ which yield a local maximum in the gain function over all data points. In the second step, the cluster activation values, $m_{i,k}$, are

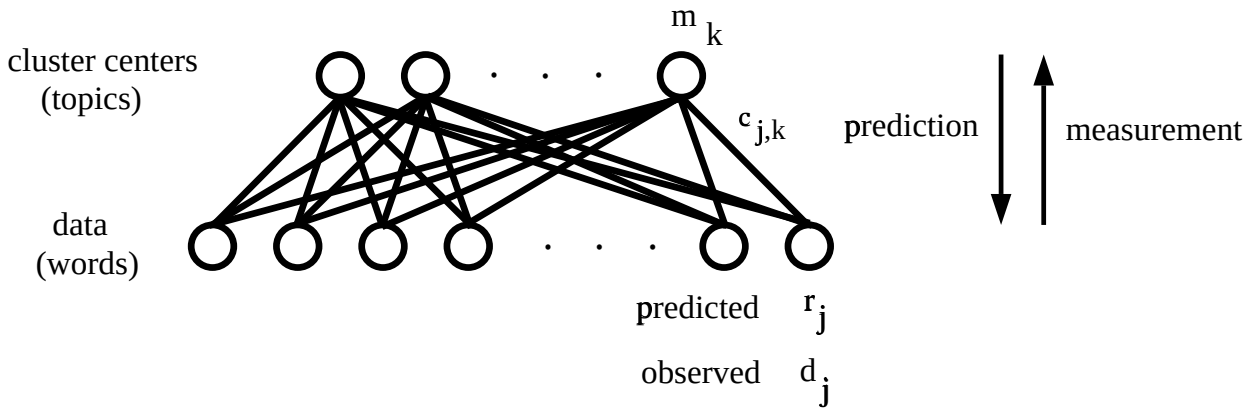

Figure 1: Architecture of the Multiple Cause Mixture Model.

fixed at the values found in the previous step and gradient ascent over the cluster centers, $c_{j,k}$, is employed to further increase the gain function until a local maximum is again reached. These steps are repeated until the gain function cannot be maximized further. Once such a plateau in the objective function is achieved, one of the existing cluster centers is split into two (i.e., the number of cluster centers is increased by one) and the above optimization is repeated over the whole set of cluster centroids. The process of incrementing the number of clusters continues until the addition of a new cluster is found to no longer increase the value of the gain function. The algorithm then returns the set of cluster centroids found as well as the activation vectors for each data point.

It is also possible to employ the MCMM as a supervised learning algorithm where the activity vectors m_i are provided by a teacher so the algorithm only needs to optimize the appropriate cluster centroids, c.

Note that contrary to conventional neural network models (Rumelhart, Hinton, & Williams 1986), there exists no direct "feedforward" computation of cluster layer activities from observed data. Also, unlike the conventional single cause mixture model, the activities m are not constrainted to sum to unity. Thus multiple cluster units may be fully active, reflecting belief that a combination of several topics best accounts for the observed word vector. Cluster units may also be only partially active, analogous to fuzzy set membership.

The MCMM also differs from recent Bayesian network (Pearl 1988) approaches to text categorization. First, the direction of causality is *from* the underlying hidden causes (topics) *to* observed document features (similar to the approach of (Fung & DelFavero 1995), but unlike that of (Croft & Turtle 1992)). Second, while Bayesian networks may employ the same Noisy Or mixing function as the MCMM, they are to be viewed in strict probabilistic terms. The MCMM refrains from committing to a formal probabilistic interpretation and its associated tools and constraints. Moreover, the MCMM network is densely connected, rendering unsuitable factorization methods for probabilistic inference (Li & D'Ambrosio 1994); instead, gradient ascent solution methods leave our approach open to employing mixing functions more complex than Noisy Or when appropriate. Furthermore, our formulation places no constraints on the combinations of topics that can occur (Fung & DelFavero 1995).

2.2 Model Assumptions

Several assumptions made by the current version of the MCMM are worth noting, especially in the context of text categorization. Foremost, our model assumes that all input data is binary. For text categorization, this means that we simply denote the occurrence or absence of a word with each component of the data vector. However, this may lead to large variations in what information is encoded by a document vector as longer documents will have a higher propensity to have spurious appearances of single words that may be unindicative of the document's topic, but are nevertheless included in the document vector. Moreover, very short documents may contain only a few distinctive terms. As a result, we employ Zipf's Law to help us select the terms we denote as "occurring" in a document vector.

According to Zipf's Law, the number of words that occur a given number of times is inversely proportional

to that number of times. This defines a Zipf curve, Z, relating the number of occurrences of a word, r, with the number of words with r occurrences. We seek to find a point, r_θ, along Z as a threshold for the number of times a word must appear in a document before we indicate the word as "occurring" in the binary document vector. For each document, we produce a Z curve based on a least squares fit of the document's actual word frequencies and use the "knee" of this curve to automatically set the r_θ value for the document. Since longer documents will tend to have larger r_θ values, this serves to eliminate spurious word appearances from the document vector.

It should be noted, however, that some other categorization research hints that simply using unprocessed word occurrence in document vectors may not dramatically alter results. For example, (Yang & Chute 1994), when comparing binary vs. weighted representations of terms, finds very little difference in the results for the two methods. Also, (Apte, Damerau, & Weiss 1994) do not make use of term weight information at all, but nevertheless achieve strong categorization results.

Another assumption of the model is the independence between components of the input vector (i.e., words). This is analogous to a neural network model which contains no hidden units in that each input unit is directly connected to each output unit. In this sense, no hidden features representing conglomerations of simple input features are learned during training to aid in finding cluster centroids. The transfer function in the MCMM is, however, very different from the sigmoid function employed in most neural networks, even though both models employ gradient ascent as their general optimization method. This makes it difficult to employ hidden units in this model, although this would be an interesting venue for further research.

2.3 Related Work in Supervised Categorization

As mentioned above, there has been extensive work on automatic text categorization using pre-labeled training text. This subsection briefly describes some of this work.

The work of (Masand, Linoff, & Waltz 1992) is quite successful given an extremely large training set of 80,000 hand-labeled instances chosen from 350 codes (in 6 categories). This approach uses Memory Based Reasoning (Stanfill & Waltz 1986), in which a highly parallel machine is used to compare a new text against all previously seen training texts in order to determine which are most similar.

(Apte, Damerau, & Weiss 1994) use a large training set to learn rules (expressed in first order logic) about which combinations of terms must be present in a document in order to classify correctly in all cases. This is not a greedy algorithm, and so in principle must try an exponential number of combinations of words, but in practice they use local computations in order to approximate the optimal results. Their algorithm achieves impressive results on the Reuters collection (see below).

(Yang & Chute 1994) learns associations between words in documents and category labels, learning a transformation from one to the other using singular value decomposition. As mentioned above, this work included experiments with variations in term weighting schemes.

(Jacobs & Rau 1990), (Hayes 1992), (Fung *et al.* 1990) and (McCune *et al.* 1985) all require the system designer to hand-code extensive information about what terms in what combinations indicate which categories and the systems only work in limited domains. (Riloff & Lehnart 1994) also requires hand-coded knowledge which is incorporated into a parser; by training on a small number of pre-labeled texts, the terms that are important for the recognition of a particular predetermined category are identified; this work is also domain-sensitive.

2.4 Related Work in Unsupervised Categorization

The MCMM can be used as both a supervised or unsupervised model which relies on maximizing the ability to predict back the input data from a given set of cluster centroids, whereas classical neural networks are supervised models which simply attempt to minimize mean squared error. Although neural network models have been developed for clustering, they often do not allow for data points to be members of more than one cluster at a time.

Text clustering is an unsupervised method for placing documents into groups, and there is a large literature on document clustering ((Willett 1988) provides a good survey of recent work). Clusters can be used to categorize documents if a document is considered to be classified by virtue of its inclusion in a cluster. However, even if the known problems with clustering algorithms are overlooked, and if we consider clusters to represent meaningful categories, it is still the case

that for a given clustering, a document can belong to only one cluster (i.e., can be assigned only one topic).

Another unsupervised method is that of latent semantic indexing (LSI), usually used in document comparison tasks (Deerwester *et al.* 1990), (Dumais & Nielsen 1992). LSI does not require pre-encoded knowledge or pre-labeled training data; word co-occurrences are counted and recorded in a matrix which is then reduced via singular value decomposition. Documents that are near one another in this space can be said to belong to the same category. In this type of approach, a document is said to occupy one point in a multidimensional space. It can be argued that multiple categories can be determined based on which dimensions of the space are examined; however, to our knowledge the application of this idea has not yet been explored.

3 Experimental Results

To test the viability of the MCMM applied to information access domains, we ran a number of experiments with different goals in mind. From the outset of our study, it became clear that the computational costs associated with the MCMM on high-dimensional data (as is the case with most machine learning algorithms) forced us to consider approaches to dimensionality reduction. To this extent, we employed the Zipf's Law scheme for word "occurrences" mentioned earlier as well as a simply frequency based method that eliminated words that occurred in very few or very many of the documents in the corpus. We believe these terms would either be too infrequent to lend to generalization in learning or would be so frequent as to not effectively differentiate any categories.

3.1 Datasets

Our study uses a standard text categorization evaluation set, the Reuters document collection, a large set of documents with assigned topic labels (in most cases only one label is assigned per document).[1] For our experiments, two subsets of the Reuters dataset were created.

The first dataset (DS1) is comprised of 983 documents from the Reuters collection, split 70%/30% into training and testing sets, respectively. These documents represent 9 labels: *gold, silver, copper, coffee, soybean, livestock, treasury bill, gross national product,* and *yen*, which we number 1 through 9, respectively. After applying our dimensionality reduction scheme, this dataset contained 372 dimensions (words).

The second dataset (DS2) is formed from a 240 article subset of the Reuters collection, again split 70%/30% into training and testing sets. This dataset represents 3 labels: *cocoa, jobs,* and *reserves*, numbered 1 through 3, respectively. After dimensionality reduction, this dataset contained 360 dimensions (words).

This is a very small subset of the Reuters collection, which has over 100 category labels and 20,000 documents; we chose a small subset both to reduce computing time and to facilitate the evaluation of the individual cluster contents.

3.2 Supervised Results

Since the MCMM had not previously been applied to text data, we first chose to conduct experiments with the algorithm run in a supervised fashion so that its clustering effectiveness could be more readily evaluated using standard accuracy metrics. For these runs, we trained the model using the DS1 training set and tested on the DS1 test set. Table 1 presents the confusion matrix of the MCMM when we categorize each test document according to the cluster which it has the highest activity for. We also report the precision and recall percentages associated with each cluster (category).

To help analyze the trade-off between precision and recall as well as making use of the multiple cause nature of the MCMM algorithm, Table 2 shows the confusion matrix when we consider a test document d_i to be a member of a cluster k for which the cluster activity $m_{i,k} \geq 0.9$. For purposes of comparison, we also ran a Naive-Bayesian classifier (Duda & Hart 1973) on DS1 and report these results in Table 3.

Here we see that although the MCMM was not originally intended for supervised classification, it gives respectable performance according to the precision and recall metrics. Moreover, the continuous valued outputs of the MCMM allows for a trade-off between precision and recall since the cluster activations, $m_{i,k}$, produce a ranking for category assignment rather than a simple binary decision. For example, increasing the threshold for $m_{i,k}$ would increase precision at the price of reducing recall.

It is important to note here that the relatively poor

[1] This collection can be obtained by anonymous ftp from /pub/reuters1 on ciir-ftp.cs.umass.edu. According to (Apte, Damerau, & Weiss 1994), free distribution for research purposes has been granted by Reuters and Carnegie Group. Arrangements for access were made by David Lewis.

	Label										
Cluster	1	2	3	4	5	6	7	8	9	Precision	Recall
1	43	12	16	1	0	1	2	1	1	55%	93%
2	0	1	1	2	3	3	1	1	7	5%	6%
3	1	1	10	0	4	4	1	0	0	48%	36%
4	2	2	1	33	0	2	0	0	0	83%	89%
5	0	1	0	1	33	6	2	0	1	75%	83%
6	0	0	0	0	0	17	2	0	1	85%	49%
7	0	0	0	0	0	1	38	3	2	86%	83%
8	0	0	0	0	0	1	0	38	1	95%	88%
9	0	0	0	0	0	0	0	0	11	100%	46%

Table 1: Confusion matrix of most active clusters for DS1.

	Label										
Cluster	1	2	3	4	5	6	7	8	9	Precision	Recall
1	43	11	15	1	0	2	3	1	1	56%	93%
2	6	1	1	4	5	4	2	2	7	3%	6%
3	20	9	25	1	4	5	1	1	1	37%	89%
4	1	1	3	34	3	1	0	1	0	77%	92%
5	1	3	1	2	39	14	4	1	1	59%	98%
6	0	1	2	2	17	27	6	1	9	42%	77%
7	7	1	1	3	8	2	44	4	9	56%	96%
8	0	0	0	0	0	0	1	36	3	90%	84%
9	1	2	4	2	4	5	7	2	24	47%	100%

Table 2: Confusion matrix for DS1 where $m_{i,k} \geq 0.9$.

	Label										
Cluster	1	2	3	4	5	6	7	8	9	Precision	Recall
1	39	11	9	0	0	0	1	0	1	64%	85%
2	2	4	3	0	0	0	0	0	0	44%	24%
3	3	0	14	0	0	1	0	0	0	78%	50%
4	1	0	0	35	1	0	0	0	0	95%	95%
5	0	1	0	1	34	2	1	1	0	85%	85%
6	0	1	0	1	3	31	0	0	0	86%	89%
7	0	0	0	0	0	0	43	0	0	100%	93%
8	1	0	1	0	2	1	0	39	3	83%	91%
9	0	0	1	0	0	0	1	3	20	80%	83%

Table 3: Confusion matrix for DS1 using *Naive Bayesian* classification.

performance of the algorithm on the first three categories (*gold, silver,* and *copper*) stems from the fact that there is a high degree of overlap between these categories as several of these documents are given multiple labels in the Reuters collection. If we consider these three categories as the super-category *precious metals* then the precision for this category is 68% and recall climbs to 73% in the case of assigning the most active cluster as the category. Nevertheless, the Naive-Bayesian classifier yields slightly better results, but a paired t-test reveals that this is not significant.

For the MCMM evaluation of Table 1, the mean precision is 0.79 and mean recall is 0.77. Studies of symbolic rule induction methods using the Reuters collection give comparable precision and recall results to those given for the MCMM. For example, (Apte, Damerau, & Weiss 1994) measure precision and recall at four different points using several different system configurations, with the closest to the above score being a precision of 0.83 and recall of 0.77. However, it is important to note that their study dealt with many more categories and a much larger subset of the Reuters collection (for both testing and training), so a direct comparison cannot be made with that work.

3.3 Unsupervised Results

We also conducted experiments to see the effectiveness of the MCMM as an unsupervised algorithm. This is where we believe the real strength of the MCMM lies. We ran the MCMM in unsupervised mode on the DS2 training set to see what clusters it would find without being given any label information. For comparison, we also ran K-Means clustering (Krishnaiah & Kanal 1982) on the DS2 training set. K was set to 4 to match the number of clusters found by the MCMM. We also tried setting $K = 3$ since DS2 has 3 labels, but we found *significantly* worse performance for K-Means in this case, so those results are not reported here.

We then compared how well the clusters found by each algorithm during the unsupervised training matched the labels provided by Reuters for the training data. Finally, we categorized the documents using the clusters found during unsupervised training (using both the maximum and thresholded categorization schemes for the MCMM) and saw how well these aligned with the hand-assigned labels from Reuters. The results of these experiments are presented in Table 4 and Table 5 for the MCMM and K-Means, respectively. In order to calculate the accuracy measurements, we consider cluster 1 to correspond to label 1, cluster 2 to correspond to label 2, and clusters 3 and 4 to correspond to label 3, since these are the majority labelings in each of the 4 clusters.

Strikingly, 3 of the 4 clusters found by the unsupervised algorithms align very well with the labels given by Reuters, achieving very high precision scores, especially considering that label information was unavailable during training. Note, however, that the MCMM had to determine both the cluster centers as well as the *number* of clusters, whereas K-Means was simply given the number of clusters a priori. The most indicative words in each cluster created by the MCMM are given in Table 6 in order to show that the clusters are in fact meaningfully related to the topic labels.

Note that the algorithm currently stops after it has added an additional cluster center that does not increase the gain function. This helps account for the fact that the first three clusters seem to have much more clearly delineated topics than the last one, although the fourth cluster nevertheless appears to be meaningful. Interestingly enough, if we consider both clusters 3 and 4 as being indicative of topic 3, then their combined precision and recall in the maximum categorization case are 77% and 100%, respectively. Due to multiple categorizations of a single document in the thresholded case, we cannot compute the corresponding values.

A close examination of the actual documents used in this experiment reveals that many of the incorrect categorizations in cluster 4 (which we consider to be about the topic *reserves*) are actually labeled as category 2, *jobs*. These documents are placed in this cluster because their primary topic is economic growth and they are only peripherally related to employment. Thus these articles make use of many monetary terms that are very related to the notion of financial reserves. On the other hand, cluster 2 (which we consider to be the cluster indicative of the *jobs* category) contains articles that are clearly discussing trends in unemployment, as opposed to the economy in general. For K-Means, however, we did not witness a similar trend as the distribution of topics in cluster 4 was much more even.

Thus it appears that the fourth cluster found by MCMM is a general economic subtopic within the dataset. This is a small piece of evidence suggesting that the MCMM might be effective in determining what kinds of labels should be considered for an unlabeled document collection, especially since it may be able to determine which documents belong in more than one category.

At this point, however, we should note the difficulty of

	Label (Maximum)					Label (Thresholded: $m_{i,k} \geq 0.1$)				
Cluster	1	2	3	Precision	Recall	1	2	3	Precision	Recall
1	42	4	0	91%	81%	42	3	0	93%	81%
2	7	40	0	85%	68%	5	29	1	83%	49%
3	0	0	27	100%	45%	0	0	27	100%	45%
4	3	15	33	65%	55%	3	9	27	69%	45%

Table 4: Confusion matrix for unsupervised MCMM on DS2.

	Label				
Cluster	1	2	3	Precision	Recall
1	36	0	0	100%	69%
2	0	31	0	100%	53%
3	3	0	31	91%	52%
4	13	28	29	41%	48%

Table 5: Confusion matrix for K-Means clustering on DS2.

Cluster	12 most active words in each cluster
1	they, icco, buffer, price, will, stock, delegate, cocoa, rule, international, producer, tonnes
2	year, february, unemployment, compare, end, march, seasonally, last, workforce, fall, adjust
3	hold, february, foreign, end, mln, fall, gold, reserve, exchange, march, currency, bank
4	exchange, bank, dlrs, government, foreign, official, last, taiwan, reserve, year, u, fall

Table 6: Most active words in clusters found by the unsupervised MCMM for DS2.

evaluating unsupervised methods, especially one such as the MCMM which allows documents to be classified in multiple categories. In previous experiments on different text datasets, we found that some of the clusters formed by the MCMM did not align well with the topic labels of the articles. To understand why this was the case, we conducted a closer examination of these clusters and found that some of the clusters were in fact meaningful, but simply grouped documents in a different way than the original labeling scheme meant to classify the documents. For example, many documents were clustered at different levels of semantic granularity than the original labeling scheme provided for. Surprisingly, we also found that it appeared that some documents had even been mislabeled!

Thus simply looking at how well the MCMM forms clusters that align well with pre-assigned labels is not doing justice to the underlying power of the model. Unfortunately, we have not discovered any solid methods to evaluate the results of clustering in document spaces which are not extremely labor intensive (i.e., having an unbiased group read all the documents assigned to a given cluster by the model and determine if the documents appear to be about a coherent theme).

4 Conclusions

The results of these experiments are quite preliminary but from them it appears that the MCMM is a viable machine learning method for text categorization. Moreover, it has several capabilities not available in other approaches that make it worth exploring further. The ability to run essentially the same algorithm in both supervised and unsupervised mode allows us to make direct comparisons about how readily identifiable topics are within a corpus. Furthermore, the multiple cause nature of this model gives us the flexibility to include documents in more than one category and also produce a ranked list of the degree of inclusion a document has in each cluster.

Unfortunately, such flexibility does not come without an associated cost. The current formulation of the MCMM is very computationally expensive, with the supervised runs presented here taking on the order of a few minutes, and unsupervised runs on larger collections (with more categories and higher dimensionality) taking on the order of several hours. To help address the problem, we are currently pursuing a wide variety of methods for dimensionality reduction (Koller & Sa-

hami 1996) as we hope to integrate this algorithm in a real-time information access architecture in the future.

Acknowledgements

This work was supported by NSF/ARPA/NASA under Stanford's Digital Libraries grant.

References

Apte, C.; Damerau, F.; and Weiss, S. M. 1994. Automated learning of decision rules for text categorization. *Transactions of Office Information Systems* 12(3). Special Issue on Text Categorization.

Croft, W. B., and Turtle, H. R. 1992. Text retrieval and inference. In Jacobs, P. S., ed., *Text-Based Intelligent Systems: Current Research and Practice in Information Extraction and Retrieval*. Lawrence Erlbaum Associates. 127–156.

Deerwester, S.; Dumais, S. T.; Furnas, G. W.; Landauer, T. K.; and Harshman, R. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science* 41(6):391–407.

Duda, R., and Hart, P. 1973. *Pattern Classification and Scene Analysis*. Wiley.

Dumais, S. T., and Nielsen, J. 1992. Automating the assignment of submitted manuscripts to reviewers. In *Proceedings of the 15th Annual International ACM/SIGIR Conference*, 233–244.

Fung, R., and DelFavero, B. 1995. Applying bayesian networks to information retrieval. *Communications of the ACM* 38(3):42–48.

Fung, R. M.; Crawford, S. L.; Appelbaum, L. A.; and Tong, R. M. 1990. An architecture for probabilistic concept-based information retrieval. In *Proceedings of the 13th International ACM/SIGIR Conference*, 455–467.

Hayes, P. J. 1992. Intelligent high-volume text processing using shallow, domain-specific techniques. In Jacobs, P. S., ed., *Text-Based Intelligent Systems: Current Research and Practice in Information Extraction and Retrieval*. Lawrence Erlbaum Associates. 227–242.

Hearst, M. A. 1994. Using categories to provide context for full-text retrieval results. In *Proceedings of RIAO '94; Intelligent Multimedia Information Retrieval Systems and Management*, 115–130.

Jacobs, P., and Rau, L. 1990. SCISOR: Extracting information from On-Line News. *Communications of the ACM* 33(11):88–97.

Koller, D., and Sahami, M. 1996. Toward optimal feature selection. In Saitta, L., ed., *Machine Learning: Proceedings of the Thirteenth International Conference*. Morgan Kaufmann Publishers.

Krishnaiah, P. R., and Kanal, L. N. 1982. *Classification, Pattern Recognition, and Reduction in Dimensionality*. Amsterdam: North Holland.

Lewis, D. D., and Gale, W. A. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM/SIGIR Conference*, 3–11.

Li, Z., and D'Ambrosio, B. 1994. Efficient inference in bayes nets as a combinatorial optimization problem. *Int'l, Journal of Approximate Reasoning* 11(1):55–81.

Masand, B.; Linoff, G.; and Waltz, D. 1992. Classifying news stories using memory based reasoning. In *Proceedings of ACM/SIGIR*, 59–65.

McCune, B.; Tong, R.; Dean, J.; and Shapiro, D. 1985. Rubric: A system for rule-based information retrieval. *IEEE Transactions on Software Engineering* 11(9).

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems*. San Mateo, CA: Morgan Kaufmann.

Riloff, E., and Lehnart, W. 1994. Information extraction as a basis for high-precision text classification. *Transactions of Office Information Systems* 12(3). Special Issue on Text Categorization.

Rumelhart, D. E.; Hinton, G. E.; and Williams, R. J. 1986. *Learning Internal Representations by Error Propagation*. MIT Press. chapter 8.

Saund, E. 1995. A multiple cause mixture model for unsupervised learning. *Neural Computation* 7:51–71.

Stanfill, C., and Waltz, D. 1986. Toward memory-based reasoning. *Communications of the ACM* 29(12):1213–1228.

Willett, P. 1988. Recent trends in hierarchical document clustering: A critical review. *Information Processing and Management* 24(5):577–597.

Yahoo! 1995. On-line guide for the internet. http://www.yahoo.com/.

Yang, Y., and Chute, C. G. 1994. An example-based mapping method for text categorization and retrieval. *Transactions of Office Information Systems* 12(3). Special Issue on Text Categorization.