A SIMPLE PROBABILISTIC APPROACH TO CLASSIFICATION AND ROUTING

Louise Guthrie
James Leistensnider

Lockheed Martin Corporation

P.O. Box 8048
Philadelphia, PA 19101

guthrie,leistens@mds.lmco.com

1. ABSTRACT

Several classification and routing methods were implemented and compared. The experiments used FBIS documents from four categories, and the measures used were the tf.idf and Cosine similarity measures, and a maximum likelihood estimate based on assuming a Multinomial Distribution for the various topics (populations). In addition, the SMART program was run with 'lnc.ltc' weighting and compared to the others.

Decisions for both our classification scheme (documents are put into any number of disjoint categories) and our routing scheme (documents are assigned a 'score' and ranked relative to each category) are based on the highest probability for correct classification or routing. All of the techniques described here are fully automatic, and use a training set of relevant documents to produce lists of distinguishing terms and weights. All methods (ours and the ones we compared to) gave excellent results for the classification task, while the one based on the Multinomial Distribution produced the best results on the routing task.

2. INTRODUCTION

One of the goals of the TIPSTER Phase II Extraction Project [Contract Number 94-F133200-000] has been to integrate extraction and detection technologies. In this paper we extend previous work (Guthrie, et al) [1] on classifying texts into categories, and develop a methodology based on the classification technique for routing documents.

By classifying and routing texts into categories we mean to include a variety of applications; categorizing texts by topic, by the language the text is written in, or by relevance to a specified task. The techniques used here are not language specific and can be applied to any language or domain.

2.1. The Intuitive Model

The mathematical model we use in this paper formalizes the intuitive notion that humans can identify the topic of an unfamiliar article based on the occurrence of topic specific words and phrases. Note that most people can tell that the first passage below is about music, even though the word 'music' is not in the passage. Similarly, most people can tell that the second passage is from a sports article, even though the word 'sport' is never mentioned.

"Before the release of his last studio album, 1993's 'Ten Summoner's Tales', Sting commented that he could no longer put his whole heart into his work; it left him feeling too vulnerable. Not surprisingly, that disc was well-crafted, but a bit void of feeling--unfortunate, considering the wondrous synergy of heart and craft on Sting's masterwork, 1987's 'Nothing Like the Sun'. Sadly, 'Mercury Falling' makes 'Ten Summoner's Tales' seem brilliant by comparison. It's as if Sting only made it because he looked at his calendar one day and realized, by golly, that it was time to make another record. Easily the worst album of what has until now been a remarkably successful career, the disc is aptly named: the temperature never seems to rise on this turgid effort." [2]

"Walter McCarty scored 24 points and Antoine Walker had 14 and nine rebounds as Kentucky pulled away in the second half to beat upstart San Jose State, 110-72, in the first round of the Midwest Regional in Dallas.

The Wildcats (28-3), who are seeking their first national championship since 1978, will meet the winner of the Wisconsin-Green Bay-Virginia Tech game on Saturday at Reunion Arena.

San Jose State, which was making its first NCAA Tournament appearance, gave Kentucky all it could handle in the first half, tying the game at 37-37 with 2:50 to play. The Wildcats then closed out the first half

with an 11-4 run to build a 47-41 advantage at the intermission.

Olivier Saint-Jean finished with 18 points and seven rebounds for the Spartans (13-17), who were one of two teams in the NCAA Tournament with a losing record." [3]

The music passage has many music related words such as 'studio', 'album', 'disc', and 'record', and the sports passage has many sports related words such as 'scored', 'beat', 'championship', 'game', and 'rebounds'. Any of these words taken singly would not necessarily give a strong indication about the passage topic, but taken together they can predict with a high degree of certainty the topic of the passage.

2.2. The Mathematical Model

The mathematical model used here is to represent each category as a multinomial distribution. Parameters are estimated from the frequency of certain sets of words and phrases (the 'distinguishing word sets') found in the training collections.

Previous results (Guthrie et al 1994) indicate that the simple statistical technique of the maximum likelihood ratio test would, under certain conditions, give rise to an excellent classification scheme for documents. Previous theoretical results were verified using two classes of documents, and excellent recall and precision scores were achieved for distinguishing topics (previous tests were conducted in both Japanese and English). In this paper we both extend the classification scheme to include any number of topics and modify the scheme to also perform routing.

In modeling a class of text, our technique requires that we identify a set of key concepts, or distinguishing words and phrases. The intuition is given in the example above, but in this work we want to automate the process of choosing word sets in a way that results in sets of 'distinguishing concepts'.

In (Guthrie et al 1994), it was shown that if the probabilities of the distinguishing word sets in each of the classes is known, we can predict the probability of correct classification. Our goal eventually is to define an algorithm for choosing 'distinguishing word sets' in an optimal way; i.e. a way that will maximize the probability of correct classification. The method we use now (described in section 4.1.) is empirical, but allows us to guarantee excellent classification results.

2.3. Common Approaches

Schemes for classification and routing all tend to follow a particular paradigm:

1. Represent each class (or topic or profile or bucket) as a numerical object.
2. Represent each new document that arrives as a numerical object.
3. Measure the 'similarity' between the new document and each of the classes.
4. For Classification - Place the new document in the category corresponding to the class (or bucket or profile) to which it is most similar. For Routing - Rank the document in the class using some function of the similarity measure.

Although many similarity measures have been studied, two of them seem to have gained popularity in the recent literature: the Cosine and tf.idf measures. The Cosine measure is used when a document is represented as a multi-dimensional vector, and a document is defined as more similar to Class 1 than Class 2 if its corresponding vector is closer to that of Class 1 than to that of Class 2. In tf.idf a document is more similar to Class 1 than Class 2 if more terms match the Class 1 terms than do the Class 2 terms. In our work a document is more similar to Class 1 than Class 2 if the probability of it belonging to Class 1 is greater than the probability of it belonging to Class 2.

In choosing a representation of a class or a representation of a document, much of the current research in classification and routing is focused on choosing the best set of terms (in our case, we call them Distinguishing Terms) to represent it. Many systems start with prevalent but not common (so that words such as 'the' and 'to' are not used) words and phrases in the class training set. The training set may be as small as the initial query which defined the class or as large as all of the documents which are available which are deemed to be relevant to the class. If this set of terms is too small, feedback is generally employed in which the full corpus of documents to be classified and routed is compared to the set, prevalent words and phrases from highly ranked retrieved documents are added to the set, and the full corpus is run again against the larger set of terms.

2.4. Probabilistic Classification Approach Using Multinomial Distribution

A probabilistic method for classification was proposed by Guthrie and Walker [1], which assumed each class was distributed by the multinomial distribution. Elementary statistics tells us that a maximum likelihood ratio test is the best way to calculate the probability that a set of outcomes was produced by a given input. In the example below, we assume a multinomial distribution for our dice and find the largest conditional probability of getting a certain output given a certain input. For ex-

ample, consider the set of outcomes produced by rolling one of two single six-sided dice. One of the dice is fair and one is loaded to be more likely to give a '6' outcome. Let us assign the expected probabilities for the outcomes for each of the two dice.

	Outcome					
	1	2	3	4	5	6
Die	Probability					
Fair	1/6	1/6	1/6	1/6	1/6	1/6
Loaded	1/10	1/10	1/10	1/10	1/10	1/2

Table 2.3-1. Expected Probabilities

Now let us define three sets of outputs.

	Outcome					
	1	2	3	4	5	6
Output	Count					
set 1	5	4	4	6	5	4
set 2	2	3	1	2	4	10
set 3	3	4	2	5	4	8

Table 2.3-2. Outputs

Using the multinomial distribution, we may calculate which is the more likely die to have produced each of the outputs. The multinomial equation is shown below, for the case of 6 possible outcomes.

$$P = \frac{n!}{n_1!\, n_2!\, n_3!\, n_4!\, n_5!\, n_6!} \left[p_1^{n_1}\, p_2^{n_2}\, p_3^{n_3}\, p_4^{n_4}\, p_5^{n_5}\, p_6^{n_6} \right]$$

Using the probabilities assigned to each die for p_1 through p_6, and the number of times each outcome occurred for n_1 through n_6, and the total number of outcomes for n, the following probabilities of producing each output given that a particular die was used are calculated.

Output	Fair Die	Loaded Die
set 1	3.46×10^{-4}	1.33×10^{-7}
set 2	4.09×10^{-6}	5.25×10^{-4}
set 3	7.07×10^{-5}	4.71×10^{-5}

Table 2.3-3. Probability of Output

The most likely die to produce each output is the one with the maximum probability. We can see that these probabilities are an excellent measure for determining which of the dice was more likely to be used to generate each of the sets of outcomes. Set 1, which has a fairly uniform distribution, is much more likely to have been created with the fair die than the loaded one. Set 2, which has nearly half of the outcomes as '6', is much more likely to have been created with the loaded die than the fair one. Set 3 does not have an obvious distribution. It has more '6' outcomes than would be expected with the fair die, but not as many as would be expected with the loaded die. As it turns out, it is just slightly more likely that the fair die was used to generate set 3.

Applying this approach to the document classification problem, we may define the outcomes to be the sets of Distinguish Terms which define the classes. The expected probabilities are then the sum of the frequencies of the Distinguishing Terms in each of the classes divided by the training set lengths. The outputs are the counts of how many of the Distinguishing Terms from each class are evident in a document. Since to create a multinomial distribution all possible outcomes must be accounted for, an additional count is kept of all of the words in a document are not members of any of the Distinguishing Term sets. The expected probability for this set of words is 1.0 minus the sum of the probabilities of all of the Distinguishing Terms in the training set.

2.5. Probabilistic Routing Approach Using Multinomial Distribution

Expanding this approach to the routing problem, we want to find the most likely class given the probabilities of the outputs. This can be calculated with Bayes' Theorem, using the assumption that all classes have equally likely occurrences.

$$P(\text{class}_i \mid \text{output}) = \frac{P(\text{output} \mid \text{class}_i)}{P(\text{output})}$$

Continuing the example with the fair and the loaded die, the sets are assigned probabilities that they belong to each of the classes given the fact that they have a certain set of outcomes. This would result in the following probabilities.

Output	Fair Die	Loaded Die
set 1	0.999616	0.000384
set 2	0.007730	0.992270
set 3	0.600170	0.388830

Table 2.3-4. Probability of Class

Sorting these probabilities, we get the expected results; set 1 is the output most likely to have been created with the fair die and set 2 the least, and set 2 is the output most likely to have been created with the loaded die and set 1 the least.

Comparing these routing results to the classification results, the question may be raised why the probability that a set is from a class needs to be calculated. Ranking with the probability of getting the outputs (Table 2.3-3) would have given the same ranking. But now consider the case in which set 3 was ten times larger, as shown in the table below.

	Outcome					
	1	2	3	4	5	6
Output	Count					
set 1	5	4	4	6	5	4
set 2	2	3	1	2	4	10
set 3	30	40	20	50	40	80

Table 2.3-5. Outputs

Our expectation is still that set 3 should be ranked in the middle, between sets 1 and 2 for each die. Calculating the probabilities of getting these outputs, we get the following table.

Output	Fair Die	Loaded Die
set 1	3.46×10^{-4}	1.33×10^{-7}
set 2	4.09×10^{-6}	5.25×10^{-4}
set 3	1.96×10^{-16}	3.39×10^{-18}

Table 2.3-6. Probability of Output

Using these probabilities directly for ranking would place set 3 on the bottom of each list, which does not agree with intuition. Note that this problem is the same problem that document retrieval systems have with documents of varying lengths; longer documents are ranked lower than they should be. But now we take the second step of calculating the probability that an output is in a class.

Output	Fair Die	Loaded Die
set 1	0.999616	0.000384
set 2	0.007730	0.992270
set 3	0.982998	0.017002

Table 2.3-7. Probability of Class

We can see that now the rankings are as we expect; set 1 is the output most likely to have been created with the fair die and set 2 the least, and set 2 is the output most likely to have been created with the loaded die and set 1 the least. So using this multinomial distribution to rank documents is less likely to be adversely affected by varying document lengths.

3. APPROACH

Below is a description of the different approaches implemented for calculating the match between a document and a class profile. The class scores are then compared to each other to determine the classification and routing results.

3.1. Class Scoring Techniques

tf.idf

The weight associated with each term in the training set is the log of the number of classes divided by the number of classes which contain the term.

The class score is calculated by the following equation [2]. This equation has been modified from the reference by dividing by the sum over the class of the term weights, to normalize the results when Distinguishing Term sets are used which have different lengths.

$$\text{score} = \left[\frac{\sum_{\text{document}} (\text{weight} \times (\frac{1}{2} + \frac{1}{2} \frac{\text{count}}{\text{max. count}}))}{\sum_{\text{class}} \text{weight}} \right]$$

Cosine

The weight associated with each term in the training set is calculated by the following equation [1].

$$\text{weight} = \log \left[\frac{\text{number of classes}}{\text{number of classes with term}} + 1 \right]$$

The class score is calculated by the following equation [1].

$$\text{score} = \left[\frac{\sum_{\text{document}} (\text{weight} \times \log(\text{count} + 1))}{\sqrt{\sum_{\text{class}} (\text{weight})^2 \times \sum_{\text{document}} (\log(\text{count} + 1))^2}} \right]$$

Multinomial Distribution

A number of weights are associated with each term in the training set. A weight is calculated for each of the classes for each term, and the weight is the probability of the term occurrence in the class. This is approximated by taking the frequency of the term occurrence in the training set divided by the size of the training set. The weights for all of the Distinguishing Terms in a set are combined into a single value, called the set weight. An additional weight is calculated, which is necessary for the multinomial distribution. This is the probability that a term is not a Distinguishing Term, and is calculated as 1.0 minus the sum of the probabilities of all of the Distinguishing Terms in the training set. Since the class scores calculated with this approach are exceedingly small, the log of the probability equation is used to avoid computational difficulties.

The class score is calculated by the following equation [3].

$$\text{score} = \left[\log \left(\frac{n!}{n_1! \ldots n_k! \, n_{k+1}!} \right) + \sum_{i=1}^{k+1} (n_i \times \log(\text{weight}_i)) \right]$$

n = number of words in document

k = number of classes

n_i = number of terms from the i^{th} set

n_{k+1} = number of words which do not match any set

For routing, the score is the probability for each class calculated given the words in the document. This is done with the following equation for each class.

$$\text{routing score} = \frac{\text{score}}{\text{sum of all scores}}$$

SMART

The SMART program independently calculates the scores for the Distinguishing Terms and for the document based upon the word frequencies in the entire collection available for classification and routing, and takes the score as the sum of the products of the Distinguishing Term and document weights. A variety of weighting schemes are possible, and a common one is called 'lnc.ltc'. The weight associated with each term in the Distinguishing Term set is calculated by the following equation [6].

$$\text{weight} = \frac{\log \left[k / m \right]}{\sqrt{\sum_{\text{class}} \log \left[k / m \right]}}$$

k = number of classes

m = number of classes with term

The class score is calculated by the following equation [6].

$$\text{score} = \sum_{\text{document}} \left[\frac{(\log(\text{count}) + 1)}{\sqrt{\sum_{\text{class}} (\log(\text{count}) + 1)}} \times \text{weight} \right]$$

3.2. Classification and Routing Techniques

Classification

For classification the document is classified into the class which has the maximum score.

Routing

In routing the top ranked documents for each class are returned. For the tf.idf, Cosine, and SMART methods the class score is used to rank the documents, for the Multinomial Distribution method the routing score is used.

4. IMPLEMENTATION

The following methods were used to determine the Distinguishing Terms, calculate the weights associated with those terms, and to compare documents to the Distinguishing Terms to get class scores and classification and routing determinations.

4.1. Selection of Distinguishing Terms

Each class has a set of Distinguishing Terms, which are those individual terms which occur more often in the class than in other classes, and which can be used to distinguish the class from the other classes. The better this set of Distinguishing Terms is, the better the results will be for routing and classification.

The Distinguishing Terms are found by processing a training set of documents which are representative of the class. This training set must be of a sufficient size to produce good statistics of the terms in the class and the frequencies of the terms.

In each document, the header information up to the headline is removed. This eliminates the class and source information which is added by the collection agent, which would bias the word set. The remaining words are separated at blank spaces onto individual lines, and stemming is performed to remove embedded SGML syntax, possessives, punctuation, and some suffixes (see Appendix A).

The words are then counted and sorted by frequency, and the word probability in the class is calculated by dividing the frequency by the number of words in the training set.

At this point the Distinguishing Terms for each class can be chosen. For this report, three different methods were implemented and experimented with.

1. Use all of the words in the training set.
2. Use the high frequency words in each list which are not the high frequency words in any other list, by selecting the words which

are in the highest so many on the list and not in the highest so many on any other list.

3. Use the high frequency words in each list which occur with low frequency on all of the other lists, by selecting only the words which occur more often in one list than in all other lists combined, until enough words have been chosen.

4.2. Calculation of Term Weights

Each of the selection methods requires a weight to be calculated for each Distinguishing Term. The tf.idf and Cosine methods all calculate the weight using the number of classes which contain the term, while the Multinomial Distribution method calculates the weight using the term probabilities.

tf.idf

$$\text{weight} = \log\left[\frac{\text{number of classes}}{\text{number of classes with term}}\right]$$

Cosine

$$\text{weight} = \log\left[\frac{\text{number of classes}}{\text{number of classes with term}} + 1\right]$$

Multinomial Distribution

Each term has a weight for each class.

$$\text{weight}_{\text{class i}} = \text{probability in class i}$$

SMART

$$\text{weight} = \frac{\log\left[k/m\right]}{\sqrt{\sum_{\text{class}} \log\left[k/m\right]}}$$

k = number of classes

m = number of classes with term

4.3. Document Classification

Each document to be classified is processed the same as the training sets are up to the selection of Distinguishing Terms; the header information is removed, remaining words are separated at blank spaces onto individual lines, and stemming is performed to remove embedded SGML syntax, possessives, punctuation, and many suffixes. The words are then counted and sorted by frequency.

The document words are compared to each of the Distinguishing Terms sets, and a class score is calculated according to the selection method being used. For classification, the document is classified into the class which has the maximum score.

For routing, the routing score is calculated from the class scores. After all of the documents have been classified the routing scores are sorted, with the highest ranking documents being those which are the most like the class profile than any other profile.

5. EXAMPLE SELECTION OF DISTINGUISHING WORDS AND WEIGHTS

To help illustrate the procedure, a small example is described. Consider two different classes, each represented by a training set. Each training set consists of a single document. Class 1 is 'Nursery Rhymes', represented with 'Mary Had a Little Lamb', and Class 2 is 'U.S. Documents', represented with the 'The Pledge of Allegiance'. These documents are shown below.

<article num=1>
<pub>NR-96
<bktype>Nursery Rhyme
<h1>Mary Had A Little Lamb
<txt>Mary had a little lamb whose fleece was white as snow.
Everywhere that Mary went, her lamb was sure to go.
<txt>It followed her to school one day, that was against the rule.
It made the children laugh and play to see a lamb at school.
</article>

Figure 5-1. Text of Class 1

<article num=46>
<pub>US-96
<bktype>US Document
<h1>The Pledge of Allegiance
<txt>I pledge allegiance to the flag of the United States of America and to the Republic for which it stands,
one Nation under God, indivisible, with liberty and justice for all.
</article>

Figure 5-2. Text of Class 2

After removing the header material, separating the words, stemming, sorting by frequency, and calculating the probabilities, the following lists would result. Notice that the stemming does not always work perfectly; 'united' is shortened to 'unite', but 'followed' is shortened to 'followe'. Overall, though, the stemming works much more often than it fails.

0.07843	LAMB	0.11429	THE
0.05882	WAS	0.08571	OF
0.05882	TO	0.05714	TO
0.05882	MARY	0.05714	PLEDGE
0.05882	A	0.05714	FOR
0.03922	THE	0.05714	AND
0.03922	THAT	0.05714	ALLEGIANCE
0.03922	SCHOOL	0.02857	WITH
0.03922	LITTLE	0.02857	WHICH
0.03922	IT	0.02857	UNITE
0.03922	HER	0.02857	UNDER
0.03922	HAD	0.02857	STATES
0.01961	WHOSE	0.02857	STAND
0.01961	WHITE	0.02857	REPUBLIC
0.01961	WENT	0.02857	ONE
0.01961	SURE	0.02857	NATION
0.01961	SNOW	0.02857	LIBERTY
0.01961	SEE	0.02857	JUSTICE
0.01961	RULE	0.02857	IT
0.01961	PLAY	0.02857	INDIVISIBLE
0.01961	ONE	0.02857	I
0.01961	MADE	0.02857	GOD
0.01961	LAUGH	0.02857	FLAG
0.01961	GO	0.02857	AMERICA
0.01961	FOLLOWE	0.02857	ALL
0.01961	FLEECE		
0.01961	EVERYWHERE		
0.01961	DAY		
0.01961	CHILDREN		
0.01961	AT		
0.01961	AS		
0.01961	AND		
0.01961	AGAINST		

Table 5-1. Word Lists

The Distinguishing Terms are then chosen, by one of three methods. The first is to choose all of the words in each list. The second is to select the words which are in the highest so many on each list and not in the highest so many on the other list. For this example, let us choose the words that are in the top 15 on each list and not in the top 10 on the other list. This would produce the following lists. The words 'the' and 'to' were eliminated from each list.

0.07843	LAMB	0.08571	OF
0.05882	WAS	0.05714	PLEDGE
0.05882	MARY	0.05714	FOR
0.05882	A	0.05714	AND
0.03922	THAT	0.05714	ALLEGIANCE
0.03922	SCHOOL	0.02857	WITH
0.03922	LITTLE	0.02857	WHICH
0.03922	IT	0.02857	UNITE
0.03922	HER	0.02857	UNDER
0.03922	HAD	0.02857	STATES
0.01961	WHOSE	0.02857	STAND
0.01961	WHITE	0.02857	REPUBLIC
0.01961	WENT	0.02857	ONE

Table 5-2. Highest Ranking Words

The third way to choose Distinguishing Terms is to select only the words which occur more often in one list than in all other lists combined until enough words have been chosen. For this example, let us choose words which occur more often in one list than in the other list until the sum of the probabilities of the chosen words is at least 40%. This would produce the following lists.

0.07843	LAMB	0.11429	THE
0.05882	WAS	0.08571	OF
0.05882	TO	0.05714	PLEDGE
0.05882	MARY	0.05714	FOR
0.05882	A	0.05714	AND
0.03922	THAT	0.05714	ALLEGIANCE
0.03922	SCHOOL		
0.03922	LITTLE		

Table 5-3. Most Likely Words

Then the weight for each word is calculated. This is done here for each selection method for the last set of distinguishing words.

tf.idf

0.69	LAMB	0.00	THE
0.69	WAS	0.69	OF
0.00	TO	0.69	PLEDGE
0.69	MARY	0.69	FOR
0.69	A	0.00	AND
0.69	THAT	0.69	ALLEGIANCE
0.69	SCHOOL		
0.69	LITTLE		

Table 5-4. tf.idf Weighting on Most Likely Words

Cosine

1.10	LAMB	0.69	THE
1.10	WAS	1.10	OF
0.69	TO	1.10	PLEDGE
1.10	MARY	1.10	FOR
1.10	A	0.69	AND
1.10	THAT	1.10	ALLEGIANCE
1.10	SCHOOL		
1.10	LITTLE		

Table 5-5. Cosine Weighting on Most Likely Words

Multinomial Distribution

Each word has a weight for each class.

0.078	0.000	LAMB	0.039	0.114	THE
0.059	0.000	WAS	0.000	0.086	OF
0.059	0.057	TO	0.000	0.057	PLEDGE
0.059	0.000	MARY	0.000	0.057	FOR
0.059	0.000	A	0.020	0.057	AND
0.039	0.000	THAT	0.000	0.057	ALLEGIANCE
0.039	0.000	SCHOOL			
0.039	0.000	LITTLE			

Table 5-6. Multinomial Distribution Weighting on Most Likely Words

SMART

Weights are not kept from the training set, only the list of words is kept. New weights are calculated from the corpus of documents to be classified and routed. But making the assumption that the training set and the corpus have the same distribution of words, the following weights would be calculated.

0.31	LAMB	0.00	THE
0.31	WAS	0.42	OF
0.00	TO	0.42	PLEDGE
0.31	MARY	0.42	FOR
0.31	A	0.00	AND
0.31	THAT	0.42	ALLEGIANCE
0.31	SCHOOL		
0.31	LITTLE		

Table 5-7. SMART Weighting on Most Likely Words

6. TESTING

The methods were tested against a small set of available documents. These were FBIS documents from June and July of 1991 on four different topics.

Number	Topic	Number of Documents
1	Vietnam: Tap Chi Cong San	20
2	Science and Technology / Japan	25
3	Arms Control	57
4	Soviet Union / Military Affairs	36

Table 6-1. Document Classes

6.1. Selection of Distinguishing Terms

Ten documents randomly chosen from each class were used as training. These training documents were then eliminated from the set of documents to be classified. The following table shows some information about the training documents.

Set	Number of Words		
	Shortest	Longest	Total
1	53	4445	16810
2	181	479	3118
3	161	1059	5498
4	145	6446	18191

Table 6.1-1. Document Classes

Set 1 contained editorials from Vietnam. Some extremely short documents were included which were no longer than the header information (which was stripped before use), the title, author and source, and a note that the article was in Vietnamese and had not been translated. Many of the high frequency words were political or economic.

Set 2 contained abstracts from Japanese technical papers. Many of the high frequency words were technological or were Japanese locations and companies.

Set 3 contained articles about arms control from all over the world. Many of the high frequency words were location, military, or negotiation related.

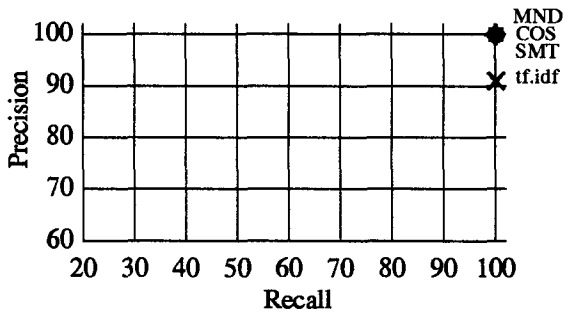
Set 4 contained articles from the Soviet Union about various military affairs, including those in other countries. Many of the high frequency words were Soviet Union locations or military related.

After experimenting with the Distinguishing Term selection methods, it was found that using the most frequent 300 words which were not the most frequent 300 words in any other class worked best for the tf.idf method. The Cosine method worked best when the Distinguishing Terms for each class were the words which were more likely to be in the class than in the sum of the rest of the classes, until the sum of the probabilities of the chosen words was at least 20%. The Multinomial Distribution method works best if the Distinguishing Terms for each class are more likely to be in the class than in another class, so the method which worked best was to choose the words which occur more often in one list than in all other lists combined until the sum of the probabilities of the chosen words was at least 25%.

6.2. Results for Classification

Topics 3 and 4 had a significant overlap in distinguishing words, and this created the most difficulty in choosing the proper class. For example, one topic 4 document described arms control efforts in France, and this was always misclassified as topic 3.

The following charts show the classification precision and recall for each of the classes. The tf.idf method gave the poorest results, while the SMART, Cosine, and Multinomial Distribution methods produced better results.



MND Multinomial Distribution
COS Cosine
tf.idf tf.idf
SMT SMART

Figure 6.2-1. Set 1 Classification Results

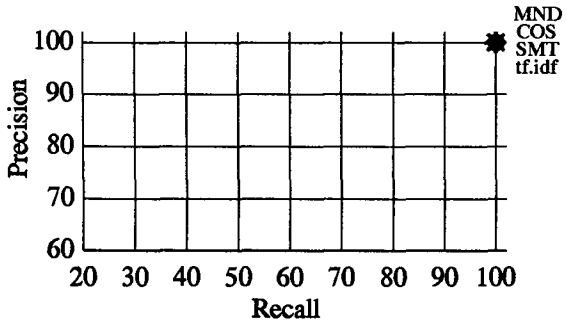


Figure 6.2-2. Set 2 Classification Results

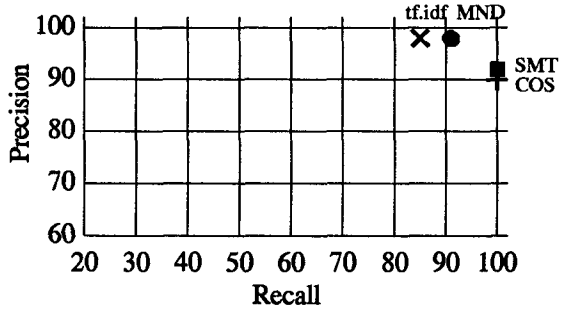


Figure 6.2-3. Set 3 Classification Results

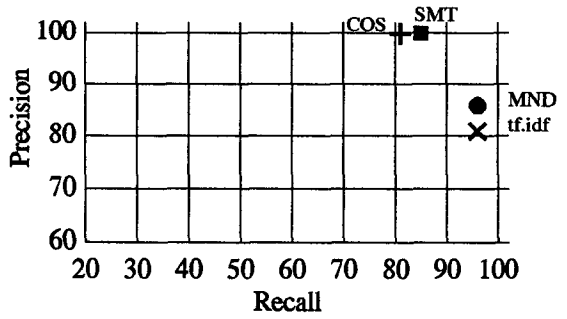


Figure 6.2-4. Set 4 Classification Results

Simplifying the charts to a single number F measure (average of precision plus recall) gives the following comparison.

Method	F measure
SMART	194
Multinomial Distribution	193
Cosine	193
tf.idf	188

Table 6.2-1. Classification F Measures

6.3. Results for Routing

The TREC precision versus recall curves are shown below.

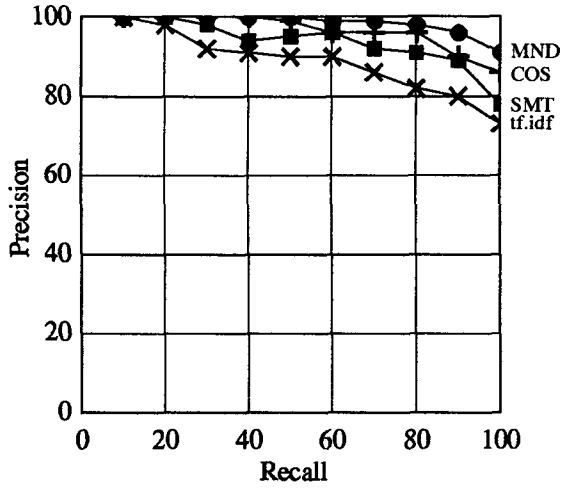


Figure 6.3-1. Routing Results

Simplifying the chart to a single number measure (area under the curve) gives the following comparison.

Method	Area
Multinomial Distribution	983
Cosine	963
SMART	933
tf.idf	882

Table 6.3-1. Routing Areas

7. CONCLUSIONS AND FUTURE WORK

For the small test performed, all of the methods produced about the same classification result, and the Multinomial Distribution method produced the best routing result. Future work with TREC data will determine whether these are repeatable results or whether the small test data was particularly well tuned to the Multinomial Distribution method.

Although we anticipate improvements to all of the methods through the use of phrases, feedback, term expansion and clustering, these have not yet been implemented. Future efforts will investigate these modifications.

This test for classification and routing was much simpler than the TREC task, since the size of the corpus was significantly smaller and less diverse and every document was relevant to a single category. This produced results which were close to perfect for all of the methods, and the Multinomial Distribution method was less than 1% different than the SMART method in clas-

sification, and only 5% better in routing. However, since the TREC data is very diverse and is classified into fifty classes, the Multinomial Distribution method is expected to perform even better than the other methods, as it is particularly good at distinguishing fine detail between classes.

8. REFERENCES

1. Guthrie, L., Walker, E., and Guthrie, J.; "Document Classification By Machine: Theory and Practice", in Proceedings of the 16th International Conference on Computational Linguistics (COLING 94); Kyoto, Japan; 1059-1063; 1994.
2. Mr. Showbiz, Starwave Corporation; 1996.
3. SportsLine, SportsTicker Enterprises L.P.; 1996.
4. Wilkenson, R., Zobel, J., and Sacks-Davis, R.; "Similarity Measures for Short Queries", in Text Retrieval Conference (TREC-4); 1995.
5. Schutze, H., and Pederson, J.; "A Cooccurrence-Based Thesaurus and Two Applications to Information Retrieval"; 1994.
6. SMART on-line documentation.

APPENDIX A. STEMMING PROCEDURE

1. Discard a word if it is an embedded statement (surrounded by < and >).

2. Change it to upper case.

3. Scan for and remove any remaining embedded statements.

4. Remove possessives.
   If the last character is an apostrophe, remove it.
   If the last two characters are 's, remove them.

5. Remove any remaining punctuation.

6. Discard the word if the previous steps have removed all of it.

7. Remove 'ies'.
   If the last three characters are 'ies', change them to 'y'.

8. Remove 'ied'.
   If the last three characters are 'ied', change them to 'y'.

9. Remove plural 's'.
   If the last character is 's' and the next to last is any consonant except 's', remove the 's'.
   Examples: winds -> wind, pass -> pass.

10. Remove 'ing'.
    Do nothing if the word is 'during' or 'th' precedes the 'ing'.
    If the last three characters are 'ing', remove them.
    Examples: winding -> wind.
    If the two characters prior to the 'ing' are the same and not 's', remove the second one.
    Examples: stepping -> step, passing -> pass.
    If the character prior to the 'ing' is a consonant except 'y', the previous character is a vowel, and the next character is not a vowel, add an 'e' to the end of the word.
    Examples: mining -> mine, keying -> key, joining -> join.

11. Remove 'ed'.
    Do nothing if the word is four characters or less.
    If the last two characters are 'ed', remove them.
    Examples: winded -> wind.
    If the two characters prior to the 'ed' are the same and not 's', remove the second one.
    Examples: stepped -> step, passed -> pass.
    If the character prior to the 'ed' is a consonant except 'y', the previous character is a vowel, and the next character is not a vowel, add an 'e' to the end of the word.
    Examples: mined -> mine, keyed -> key, joined -> join.