A MAXIMUM LIKELIHOOD MODEL FOR TOPIC CLASSIFICATION OF BROADCAST NEWS

Richard Schwartz, Toru Imai, Francis Kubala, Long Nguyen, John Makhoul*

BBN Systems and Technologies, Cambridge, MA, 02138, USA
*NHK (Japan Braodcasting Corp.) Sci. & Tech. Res. Labs., Tokyo 157, Japan
Tel: 1-617-873-3360, FAX: 1-617-873-2534, E-mail: schwartz@bbn.com

ABSTRACT

We describe a new algorithm for topic classification that allows discrimination among thousands of topics. A mixture of topics explicitly models the fact that each story has multiple topics, that different words are related to different topics, and that most of the words are not related to any topic. The resulting model, trained by EM, has sharper distributions of words that result in more accurate topic classification. We tested the algorithm on transcribed broadcast news texts. When trained on one year of stories containing over 5,000 different topics and tested on new (later) stories the first choice topic was among the manually annotated choices 76% of the time.

1. INTRODUCTION

This paper deals with the problem of classifying the topic or topics in a document among a large set of predefined topics. Topic classification can be used for skimming, categorizing, or retrieving documents. We distinguish topic classification from typical document retrieval, in which a query consisting of a set of words is compared with documents using a comparison based on weighted word similarity. In particular, here we assume that we have a substantial number of documents that have previously been classified as to the topics contained, allowing for a wide variety of more powerful probabilistic algorithms to be applied.

Previous efforts at topic classification [1-6] dealt with only tens of topics. The method most commonly used to model a topic is simply to count the number of times each word occurs in all the stories that are labeled with that topic. Then, to classify a new story, one multiplies the relative frequencies of all the words in the story for each topic and chooses the topic with the highest product. Various smoothing [3] and word selection techniques [1] have been developed to try to make this method work better.

The fundamental problem with the relative frequency approach is that a particular word in a story may be related to one, or sometimes two of the topics, but rarely does one word indicate *all* of the topics. In addition, most of the words in a story are not directly related to *any* of the main topics, but rather are just general words in the language, or are related to other minor topics. Finally, a real story (such as the news stories in broadcast news data) typically has several topics. For example, a story discussing U.S. policies on loans to Mexico is labeled with four topics: “Clinton, Bill”, “Mexico”, “Money”, and “Economic assistance, American”. The relative frequency approach falsely assumes that each word is related to all the topics. This incorrect model results in high likelihoods for words that are not related to topics. It also results in severely overlapping distributions of words for different topics.

Here we describe a new method that assumes from the start that documents (we use “stories” from now on) have several topics. We have adopted a mixture model of words given the topics of a story as a more realistic model of language. This allows us to distinguish a large number of diverse topics more easily.

In Section 2, we review the traditional model and then describe the new model. In Section 3, we explain how the model parameters are estimated. We present the classification (decoding) algorithm in Section 4, including a 2-pass approximation that reduces the computation needed. Finally, we present some large-scale experiments comparing the two methods on Broadcast News in Section 5.

2. GENERATIVE TOPIC MODELS

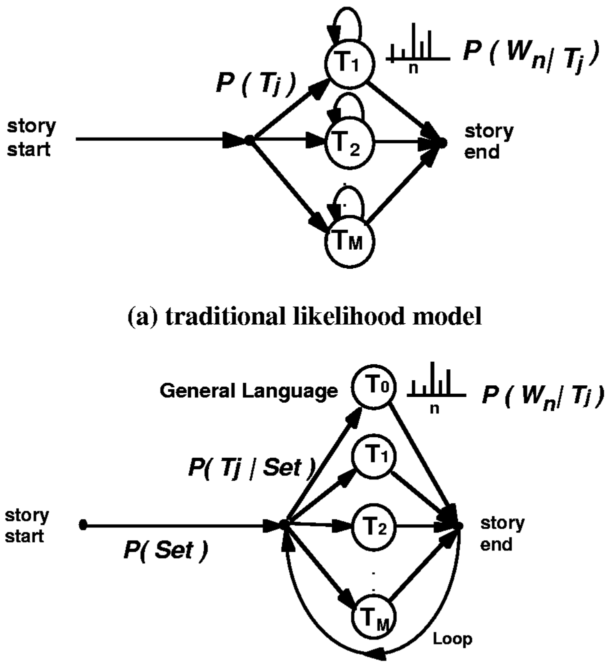
First, we present an interpretation of both the traditional likelihood model and the new model as simple finite state (hidden Markov) models. What we call the traditional method [1-6] starts by counting the number of occurrences of each word in stories with each of the topics. This allows us to estimate the *a priori* probability of each topic label, $P(T_j)$, the conditional likelihood of each word given the topic, $p(W_n \mid T_j)$, and the unconditioned probability of each word, $p(W_n)$. Using Bayes’ rule and the assumption that the words in the story ($W_1 \mathrm{K}\ W_T$) are independent, we get the *posterior* probability of the topic given the words

$$P(T_j \mid Story) \approx P(T_j) \prod_t \frac{P(W_t \mid T_j)}{P(W_t)} . \qquad (1)$$

There are two obvious problems with this formula. First, some words are clearly not related to any topic. Common “stop” words can be eliminated. Effective keywords can also be selected using a χ^2 test [1]. The second problem is that if one of the words that occurs in a test passage has never occurred in the training examples for that topic, the resulting product will be zero. This is easily solved by preventing the probabilities from being zero. McDonough showed in [3] that interpolating the conditional likelihoods with the unconditioned likelihoods as in (2) works as well as keyword selection.

$$P(W_n \mid T_j) \approx \tfrac{3}{4} P(W_n \mid T_j) + \tfrac{1}{4} P(W_n) \qquad (2)$$

Here we interpolated the probabilities using [7]. The traditional model can also be viewed as a very simple hidden Markov model (HMM) model that generates the words, given a topic. Figure 1a illustrates such a model. First, we choose the topic of the story based on the *a priori* probabilities for topics. Then, given that topic, we go to a state that generates words according to their conditional likelihoods for the chosen topic. We continue to choose words until the story is over. It is obvious that this model does not easily generalize to the case of more than one topic for a story. And we must assume that all of the words are related to all of the topics (which is clearly not a good model of language). When we estimate the conditional likelihoods, many words are (falsely) associated with many topics, resulting in topic dependent word distributions that overlap considerably, making it hard to distinguish topics in new (test) stories. In addition, the vast majority of the words in a story, which are not really related to any particular topic, are shared (but not equally) in all of the topic models.



(a) traditional likelihood model

(b) new mixture model

Figure 1: HMMs for (a) the traditional and (b) the new model

Figure 1b illustrates the new model we have adopted. First, to generate a story, we must choose the *set* of topics for the story according to the probabilities of different sets. Note that we always add the special topic of General Language, T_0, to the set of chosen topics. Then, given the set of topics, we choose the topic (or General Language model) that will generate the first word, according to $P(T_j \mid Set)$, the probability that the next word will be about T_j., given the set of topics. A word is then generated according to $P(W_t \mid T_j)$, which is now interpreted as the probability of W_t given that it is a word related to topic *j*. Then, we loop back to where we must choose the topic for the next word. This model clearly allows different words to be about different topics, and also allows us to model (rather than discard) the words that are related to General Language.

3. MODEL ESTIMATION (TRAINING)

To estimate the traditional model we simply count the number of occurrences of each word in stories that have each of the topic labels assigned, and then normalize the counts to form probability distributions. We smooth the distributions [7] to avoid estimation problems. To estimate the parameters of the new model, we must use the Estimate-Maximize (EM) algorithm because, although we know the set of topics assigned to a story, we do not know which of the words is related to each of the topics. We find the parameters (both$P(T_j|Set)$ and $P(W_t|T_j)$) that maximize the likelihood of the observed words given the sets of topics in the training.

We initialize $P(T_j|Set)$ to the *a priori* probability of the topics. $P(T_0 \mid Set)$is initialized with a probability of 1. Note that this means the probabilities of choosing the next word do not sum to 1. Therefore we rescale them for each story so that they sum to 1. We also initialize the $P(W_t|T_j)$ to those obtained from the traditional algorithm. Given the model structure shown in Figure 1b, the probability of a particular word given the set of topics is

$$P(W_t \mid Set) = \sum_{j \in Set} \left[P(T_j \mid Set) P(W_t \mid T_j) \right] \qquad (3)$$

The EM algorithm dictates that we must distribute the count for this word in the training among the possible topics.

$$C(W_t \mid T_j) = \frac{P(T_j \mid Set) P(W_t \mid T_j)}{\sum_{j \in Set} P(T_j \mid Set) P(W_t \mid T_j)} \qquad (4)$$

The counts for words given topics are normalized to produce new estimates of word distributions.

$$P(W_n \mid T_j) = \frac{C(W_n \mid T_j)}{\sum_n C(W_n \mid T_j)} \qquad (5)$$

$P(T_j \mid j \in Set)$, the average percentage of words generated by a topic state given it is in the set of topics, is found by dividing all the counts for that topic by the number of words in stories with that topic label.

$$P(T_j \mid j \in Set) = \frac{\sum_n C(W_n \mid T_j)}{\#\ \text{words in stories about topic } j} \qquad (6)$$

4. CLASSIFICATION (DECODING)

In the conventional method, we compute the log *posterior* probability of each topic, where α is optimized to adjust for the incorrect independence assumption.

$$\log P(T_j \mid Story) = \log P(T_j) + \alpha \sum_t \log\left[\frac{P(W_t \mid Tj)}{P(W_t)}\right] \quad (7)$$

To decode with the new method, we must, in principle, consider all possible *sets* of topics, and compute

$$P(Set \mid Story) = P(Set)\prod_t \frac{\sum_{j \in Set} P(T_j \mid Set)^{\beta} P(W_t \mid T_j)}{P(W_t)} \quad (8)$$

$P(T_j|Set)$ are derived by scaling the $P(T_j \mid j \in Set)$ to sum to 1 for each set considered. The exponential weight, β, is used to counteract the effects of the incorrect independence assumption. This is clearly not feasible when there are several thousand topics. Instead we first consider each topic independently using (9) to choose a small set of likely topics. Then we can rescore all subsets of the top-N topics using (9).

$$\log P(T_j \mid Story) = \log P(T_j) + \sum_t \phi\left\{\log\left[P(T_j \mid j \in Set)^{\beta} \frac{P(W_t \mid T_j)}{P(W_t)}\right]\right\} \quad (9)$$

The new model in (8) avoids the problem of some words being unobserved for a particular topic, because the probability for each word is the *sum* over the topics in the set. So if one of the topic models has a low probability, then perhaps another will have a higher probability. However, when we train the models using this assumption, each topic is effectively trained on only a relatively small number of words. When we use (9) to score topics independently, we violate the assumptions of the new model. The majority of the words, which are related to General Language or other topics, will have a very low probability for the topic we are considering. To avoid this effect, we use a simple filter function, ϕ,

$$\phi(x) = \begin{cases} x & \text{if } (x \geq 0) \\ 0 & \text{if } (x < 0) \end{cases} \quad (10)$$

so we ignore all words with negative likelihood ratios.

After finding a small number of topics that have high scores, we could consider all 2^N-1 subsets of those topics explicitly using (8). We approximated the probability of the set of topics by the product of the joint probabilities of all of the pairs of topics in the set. To avoid a bias for smaller sets, we de-exponentiated the product by the number of terms used.

$$P(Set) = \left[\prod_{k \in Set} \prod_{m \in Set(m>k)} P(T_k, T_m)\right]^{\frac{1}{\binom{N}{2}}} \quad (11)$$

5. EXPERIMENTS

5.1. Corpus

We performed experiments on a corpus of broadcast news transcribed by Primary Source Media available on CDROM[8]. Although the corpus contains over 4 years of transcriptions, we used only 1 year, since that resulted in higher accuracy (topics changed considerably over time). Each story has from 1 to 13 topic labels (depending on the completeness of the person who annotated the data), for an average of 4.5 topics per story. The training data contained 42,502 stories from July 1995 through June 1996. The test data consisted of 989 stories from the first half of July, 1996. The stories averaged about 1,000 words in length, but varied considerably. Even in this one year training period, there were over 5,000 unique topic labels.

5.2. Training

We estimated models for the 4,627 topic labels that occurred more than once. About 2.5% of the topic labels that appeared in the test stories never occurred in these 4,627 labels and so they could not be found. We removed 215 "stop" words (from a list in [9]) and removed suffixes like "ing" and "ed" using an algorithm like Porter's [10]. The 42,502 training stories contained 95,597 unique words after these changes. The average number of unique words in each topic was about 3,000, but it varied greatly depending on the number of training stories for each topic.

As predicted, the new model results in sharper more distinguished distributions. Table 1a shows the probabilities of the most likely words for the topic "Clinton, Bill" using the traditional model. The words "Clinton" and "president" are clearly. However, the words "go", "think", and "say" are not directly related – they are just common words. Table 1b shows the most likely words after EM training. The first four words are clearly relevant, and are 10 times more likely than before, while "go" and "think" are much less likely.

Rank	Word	P(W | T)
1	**president**	**0.013**
2	go	0.011
3	think	0.010
4	**Clinton**	0.009
5	say	0.008

a) Traditional Model

Rank	Word	P(W | T)
1	**president**	**0.104**
2	**Clinton**	**0.096**
3	**house**	**0.036**
4	**white**	**0.034**
.	.	.
.	.	.
36	go	0.003
44	think	0.003

b) New Method

Table 1: Word likelihoods for the topic "Clinton, Bill".

Topic	P (T | Set)
General English	**0.935**
Music, Black	**0.085**
...	
Politics and government	**0.018**
Clinton, Bill	**0.020**
Politics and government	**0.018**

Table 2: Percentage of words produced by some topics.

Table 2 shows the average percentage of words that are directly related to the topic for a few topics. The General Language model accounts for 93.5% of the words, while for most other topics, only a few percent of the words are about that topic.

5.3. Classification Results

Figure 2 shows the classification results comparing the traditional and the new method, scoring topics independently. Based on preliminary experiments, we used α=0.25 and β=0.35. For the basic experiments, the number of topics chosen was varied from 1 to 5. The precision is the percentage of topics chosen that are among the topics annotated manually. The "at-least-one accuracy" shows the percentage of stories in which at least one of the topics found was among the annotated topics. As can be seen, the precision of the new method is considerably higher than that of the traditional method. In particular, the precision of the top choice increases from 63.6% to 75.7%. When we examined a few errors (for both methods), we found that, in many cases, the topics chosen by the algorithm were indeed relevant. Perhaps it was difficult for unaided human annotators to remember 5,000 labels.

We also applied the full scoring method using (8) and (11) with the greedy algorithm described. This was often able to remove one or two topics that were not consistent with the others, thus increasing the precision of the 4th and 5th choices. However, the effect, as shown in Figure 2, was not dramatic. Perhaps a better model of dependence would improve this rescoring step.

In a later experiment to understand the contributions of the new model, we ran an experiment in which we omitted the General Language model, T_0, but kept the mixture model. We found that half of the gain in the new method was due to having the General Language model, and half was due to the use of a mixture model.

6. CONCLUSIONS

We have described a new method for topic classification that models topics within documents more realistically. It accounts for the facts that most stories have several topics, with different words related to different topics, and that most of the words in stories are not directly related to any of the topics. Experiments on broadcast news transcriptions with 5,000 topics, verified that the word distributions obtained were more distinguished, and that classification results were significantly improved.

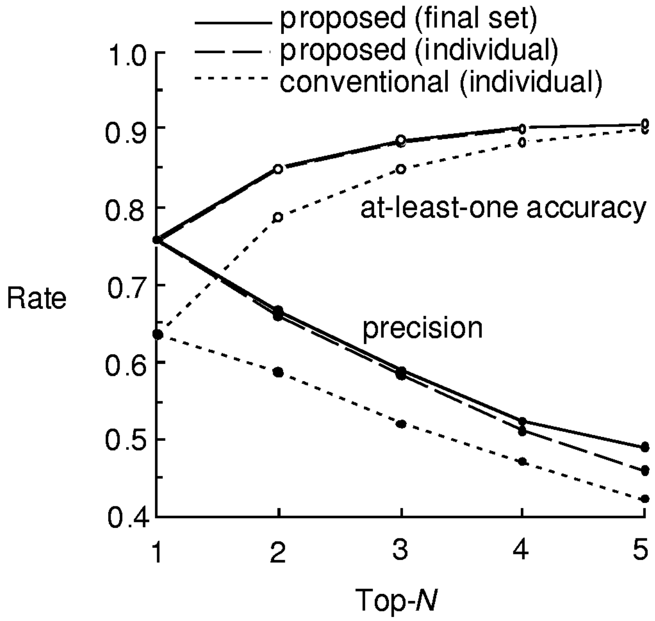


Figure 2: Comparison of Topic Classification Accuracy

REFERENCES

[1] L. Gillick, et. al., "Application of Large Vocabulary Continuous Speech Recognition to Topic and Speaker Identification Using Telephone Speech," Proc. ICASSP-93, Vol. II, pp. 471-474, 1993.

[2] B. Peskin, et. al., "Improvements in Switchboard Recognition and Topic Identification," Proc. ICASSP-96, Vol. II, pp. 303-306, 1996.

[3] J. McDonough, et. al., "Issues in Topic Identification on the Switchboard Corpus," Proc. ICSLP-94, pp. 2163-2166, 1994.

[4] J.H. Wright, et. al., "Improved Topic Spotting through Statistical Modelling of Keyword Dependencies," Proc. ICASSP-95, pp. 313-316, 1995.

[5] R.C. Rose, et. al., "Techniques for Information Retrieval from Voice Messages," Proc. ICASSP-91, pp. 317-320, 1991.

[6] Y. Yamashita, et. al., "Next Utterance Prediction Based on Two Kinds of Dialog Models," Proc. Eurospeech-93, pp. 1161-1164, 1993.

[7] P. Placeway, et. al., "The Estimation of Powerful Language Models from Small and Large Corpora," Proc. ICASSP-93, Vol. II, pp. 33-36, 1993.

[8] http://www.thomson.com/psmedia/bnews.html

[9] http://www.perseus.tufts.edu/Texts/engstop.html

[10] M.F. Porter, "An Algorithm for Suffix Stripping," Program, 14(3), pp. 130-137, 1980.