PROBABILISTIC MODELS FOR TOPIC DETECTION AND TRACKING

F. Walls *H. Jin* *S. Sista* *R. Schwartz*

GTE/BBN Technologies
70 Fawcett St, Cambridge, MA 02138 USA
fwalls@bbn.com, hjin@bbn.com

ABSTRACT

We present probabilistic models for use in detecting and tracking topics in broadcast news stories. Our information retrieval (IR) models are formally explained. The Topic Detection and Tracking (TDT) initiative is discussed. The application of probabilistic models to the topic detection and tracking tasks is developed, and enhancements are discussed. We discuss four variations of these models, and we report our preliminary test results from the current TDT corpus.

1. INTRODUCTION

The Topic Detection and Tracking (TDT) program deals with broadcast news stories. The goals are to be able to track old topics or detect new ones. The problem starts with a large corpus of news stories that are either original text (i.e., newswire) or transcribed text from an audio source. Although seemingly similar, the problems of tracking and detection differ significantly.

In tracking, the system is given a small number (1, 2, 4, 8, or 16) stories about a particular topic. The system is also given other stories (usually a large number) which are known to be irrelevant to the topic. The goal is to produce a score for each remaining story in the corpus that indicates how likely it is to be on the target topic.

In detection, the system must partition all stories into mutually exclusive clusters, and each cluster should contain all the stories on one and only one topic. A detection system is offered no prior knowledge about the topics and must process stories causally or with minimal lookahead.

For both tasks, most algorithms require a metric that evaluates whether a single story belongs with another set of stories on a topic. Therefore, we shall develop a set of four probabilistic metrics that can be applied to detection and/or tracking.

The paper is organized into three main sections. First, we discuss a probabilistic framework for comparing one news story with a group of news stories. The second section contains the specific techniques we used to perform topic detection and tracking. Finally, we present our research results in the third section.

2. PROBABILISTIC IR MODELS

Classical IR measures compare queries with stories by using somewhat ad hoc measures that are related to how many times each query word occurs in a document. We propose to use probabilistic measures wherever possible so that we can formally express what quantities we are computing.

We use two different fundamental models for comparing a story to a group of stories on a topic:

1. The group is the model, and the words in the story were generated according to the word distribution of the group (e.g., the BBN topic spotting model).
2. The story is the model, and the words in the group were generated according to the word distribution of the story (e.g., the BBN IR metric).

In the first case, we are trying to calculate $p(T|S)$ where S is the story and T represents the group of stories on a topic. In case two, we calculate $p(SisR|T)$, which is the probability that S is relevant given the topic model.

We enhance this model further by allowing the words in a story to be generated from two word distributions: the topic-specific distribution and the general English distribution. This model is depicted in figure 1.

BBN topic spotting metric

The BBN topic spotting metric (TS) is one method of estimating of $p(T|S)$; in other words, we want to compute the posterior probability that story S comes from the distribution of topic T. By Bayes' Rule:

$$p(T|S) = p(T) \cdot \frac{p(S|T)}{p(S)} \tag{1}$$

where $p(T)$ is the *a priori* probability that any new story will be on topic T. Furthermore, by making an assumption that the story words are conditionally independent, we get:

$$p(T|S) \approx p(T) \cdot \prod_n \frac{p(s_n|T)}{p(s_n)} \tag{2}$$

where s_n corresponds to the individual words in the story, and $p(s_n|T)$ is the probability that a word in a story on topic T would be s_n.

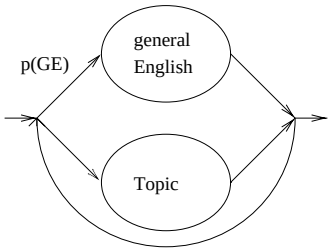

Figure 1. Two-state model of a topic

We model $p(s_n|T)$ with a two-state mixture model, where one state is a distribution of the words in all of the stories in the group, and the other state is a distribution from the whole corpus. That is, we have a generative model for the words in the new story.

To calculate the distributions of the states, we use the Maximum Likelihood (ML) estimate, which is the number of occurrences of s_n among the topic stories divided by the number of words in topic stories. This estimate can be corrected for two main weaknesses:

1. The "stop words" (e.g., the, to) dominate the score. These words can simply be eliminated.
2. The unobserved words for the topic have zero probability. Therefore, the model can be smoothed with a "back-off" to the General English model:

$$p'(s_n|T) = \alpha \cdot p(s_n|T) + (1-\alpha) \cdot p(s_n) \qquad (3)$$

The estimates for the general English distribution and topic distributions can be refined using the Expectation-Maximization (EM) algorithm. This process allows new words to be added to the distributions and emphasizes topic-specific words. Therefore, the EM algorithm assigns higher probabilities to words that are more likely to be in the topic. [3]

BBN IR metric

The BBN IR metric looks at the problem in exactly the opposite way. Given a query Q, we want to know the probability that any new story S is relevant to the query. But in this case, we assume that the query was generated by a model estimated from the story.

$$p(SisR|Q) = p(SisR) \cdot \frac{p(Q|S)}{p(Q)} \qquad (4)$$

Dropping $p(Q)$ and assuming independence of words in the query, we have:

$$p(SisR|Q) \approx p(SisR) \cdot \prod_n p(q_n|S) \qquad (5)$$

Again, we use a two-state model, where one state is a unigram distribution estimated from the story S, and the other is the unigram distribution from the whole corpus.

For the tracking problem, we use all of the stories given to be on the topic as the query. Thus, the query is a very long sequence of words – typically much longer than the new story.

Relevance Feedback

The Relevance Feedback (RF) measure is similar to the IR measure, except we do not use all of the words in the topic stories. Instead, we only use those words that are common to at least two of the stories. Each common word is used only once, but the "back-off" weight from the story state to the general English state is estimated as a function of the number of topic stories that have that query word.

Word Feature

The Word Feature (WF) measure is similar to the RF measure, in that it starts with the words common to two or more of the topic stories. But instead of a two-state mixture model, we use a simple likelihood ratio score:

$$score = \prod_n \frac{p(q_n\_in\_S|S\_on\_T)}{p(q_n\_in\_any\_story)} \qquad (6)$$

3. TRACKING AND DETECTION

3.1. Tracking algorithms

Our tracking system utilizes scores from all four methods: TS, WF, IR, and RF. We utilize an automatic prodedure to normalize the scores within topic and combine different methods to acheive better results.

Score normalization

Because one threshold is used for all topics, score normalization across topics is important for optimizing system performance. Therefore, we collect statistics on the scores by using the training stories as the test stories, then normalize the test scores based on these statistics for each topic.

Model combinations

Different systems focus on different features of the stories. Thus, it seems reasonable to combine the probability scores from many tracking systems with a time-decayed prior probability score. This reflects that a test story is less likely to be on-topic as its age increases.

We use a linear combination of the log scores from the above four systems and the time decay to form the BBN tracking system. Our experiments show a significant reduction of both miss and false alarm rates with a combined system.

3.2. Detection algorithms

Our detection system utilizes many of the same ideas as the tracking system. Detection uses the TS metric to compare individual stories to clusters, but not the IR metric (for reasons that will be explained). The detection system has two main components: clustering and decision metrics.

Clustering

Our detection system utilizes incremental clustering, a simple clustering algorithm. Incremental clustering involves processing stories sequentially and one at a time, and it makes each clustering decision immediately. When a story is encountered, incremental clustering executes two steps: 1) decide which cluster the story is closest to (*selection*), and 2) decide whether to merge the story with the closest cluster or start a new cluster (*thresholding*).

Decision metrics

Our detection system utilizes the TS metric. We build the topic model from the stories within a cluster. Each story's likelihood is calculated based on the two-state model discussed in Section 2.

The choice of decision metrics is important in detection. Because of the incremental clustering approach, we need a decision metric that is comparable across stories and clusters. More specifically, in selection, the metric must be compared using different-sized clusters. Furthermore, thresholding requires the metric be comparable to a constant threshold, independent of both story and cluster size. Note that because the two steps require different types of comparison, two different metrics may be used.

Unfortunately, the probabilistic framework described in Section 2 is not inherently conducive to making such comparisons. For instance, the IR metric is not useful for selection, because one score is generated for each story word in the cluster. On the other hand, the TS metric produces a score for each word in the current story, regardless of cluster size. Hence, the TS metric is an appropriate measure for selection.

For thresholding, the TS metric is still inadequate because the story lengths also differ. At this point, we resort to an *ad hoc* normalization approach to make comparisons with a fixed threshold. One such normalization technique is called average log likelihood. Average likelihood involves simply dividing the final log likelihood by the number of words in the story.

Finally, some techniques used in tracking are also applicable to detection. The cluster models can be adapted as the stories age, and old clusters can be decayed over time. We adapt the cluster models by multiplying system cluster counts by a constant (slightly less than 1) after each day of stories is processed.

4. RESULTS

4.1. Corpora

The Linguistic Data Consortium (LDC) has released two corpora for the purpose of expanding research in TDT. The first, originally used for a pilot evaluation conducted in 1996-1997, consists of about 16,000 stories from newswire sources, collected over one year. [2]

The second corpus, referred to as TDT-2, consists of about 60,000 stories collected over a six month period from both newswire (nwt) and audio sources. The audio sources are transcribed two ways: manually, based on the closed captioning of the news broadcasts (ccap); and automatically, based on the output of Dragon Systems' speech recognizer (asr). The TDT-2 corpus is subdivided in three two-month sets: a training set (train), a development test set (devtest), and an evaluation set. The training set and development test set, which both contain training and test data, can be used freely in the research and system design.

The data is annotated at LDC by human annotators, who listen to audio data or view text transcripts. The annotators are given a set of predefined topics to look for, which can vary from specific events (e.g., John Glenn's space shuttle trip) to broad sequences of events (e.g., Asian economic crisis). For each story, an annotator determines which topics are relevant to the story. Note that only a small percentage of the stories are labeled for any topic. [1]

4.2. Evaluation

The evaluation for the tracking and detection tasks are given below.

Tracking

The tracking task is concerned with finding news stories relevant to N_t given stories on a topic, where N_t is 1, 2, 4, 8, or 16. Except for these training stories, each story in the corpus is scored for its relevance to the target topic. A Detection Error Tradeoff (DET) curve can be generated for each topic by sweeping a decision threshold through the range of possible scores. For the evaluation, the system generates a decision threshold for which a cost function is evaluated:

$$C_{track} = C_{miss} \cdot P_{miss} \cdot P_{topic} + C_{FA} \cdot P_{FA} \cdot (1 - P_{topic}) \quad (7)$$

where $C_{miss} = 1$ and $C_{FA} = 1$ are the costs of a miss and false alarm, and $P_{topic} = 0.02$ is the prior probability of a story being on some topic.

Finally, the C_{track} for each topic is averaged over either the stories or the topics to yield a final result.

Detection

The detection task involves partitioning the corpus into groups of stories related by topic. After the system outputs a clustering of the data, the evaluation software matches each reference topic with the best scoring system cluster. Finally, each topic/cluster pair is scored using the cost function used in tracking given by equation (7). The costs for each topic can be averaged over either the stories or the topics to yield a final result. [1]

Unless otherwise noted, results are reported according to the evaluation specification for TDT using the TDT-2 corpus.

4.3. Tracking results

The first part of table 1 shows the effect of changing the number of training stories; namely, performance degrades when n_t falls below 4. We also compare the performance on an two identical sets of stories from the devtest set, the first

descrip.	cond.	P_{miss}	P_{FA}	C_{track}
TS-train n_t=16	ccap+nwt	24%	0.57%	.0104
TS-train n_t=8	ccap+nwt	23%	0.59%	.0104
TS-train n_t=4	ccap+nwt	22%	0.55%	.0098
TS-train n_t=2	ccap+nwt	32%	0.45%	.0108
TS-train n_t=1	ccap+nwt	47%	0.35%	.0128
RF-devtest	asr	3%	0.68%	.0073
RF-devtest	ccap	3%	0.69%	.0074

Table 1. Effect of training and source conditions on performance

descrip.	cond.	P_{miss}	P_{FA}	C_{track}
WF-devtest	asr	15.4%	0.09%	.0114
IR-devtest	asr	1.95%	0.11%	.0110
RF-devtest	asr	4.56%	0.08%	.0086
TS-devtest	asr	3.04%	0.08%	.0086
combo-devtest	asr	6.07%	0.01%	.0022

Table 2. Performance of different systems (Word Feature, Information Retrieval, Relevance Feedback, Topic Spotting, and combined) when $n_t = 4$

transcribed using closed captioned (ccap) text and the second using the automatic speech recognition (asr) text. Our results (bottom of table 1) show little difference in performance between asr and ccap, despite word error rates in asr text of about 23%.

Table 2 shows the performance improvement of using a logistic regression of all system outputs (combo) versus the RF, IR, TS, and WF systems.

4.4. Detection results

The detection results are shown in table 3. In the first two lines, table 3 clearly shows the improved performance of using a probabilistic metric (TS) over a cosine distance metric (cos) [4]. The third and fourth lines show that using automatically recognized speech (asr) is a significant detriment to performance compared to using closed captioning transcripts (ccap). Adaptation for the time varying nature of topics (TSa) does not show improvement over no adaptation, as shown in the last line. The devtest set gives much better results than the train set. This difference is probably because the devtest topics contain fewer stories and the topics show less variation.

5. CONCLUSION

Although these results demonstrate good performance, both the topic detection and tracking problems leave room for considerable improvement. In light of the ultimate ambition of the TDT initiative (detecting and tracking topics across multiple languages), more research is necessary in these areas. Furthermore, variance between the data sets is significant (even within the TDT-2 corpus), and results can be vary substantially. Understanding these differences is an important goal of the research effort.

Although using more training stories usually improves tracking performance, there is little difference between

descrip.	cond.	P_{miss}	P_{FA}	story C_{det}
TS-devtest	nwt+asr	30.9%	0.05%	.0067
TS-devtest	nwt+ccap	15.0%	0.14%	.0043
TS-train	nwt+ccap	36.2%	0.28%	.0099
cos-train	nwt+ccap	71.5%	0.13%	.0156
TSa-train	nwt+ccap	43.0%	0.16%	.0101

descrip.	cond.	P_{miss}	P_{FA}	topic C_{det}
TS-train	nwt+ccap	28.1%	0.10%	.0066
cos-train	nwt+ccap	32.8%	0.04%	.0069
TS-devtest	nwt+asr	18.8%	0.10%	.0048
TS-devtest	nwt+ccap	11.3%	0.09%	.0031
TSa-train	nwt+ccap	24.7%	0.16%	.0065

Table 3. Detection results for TDT-2. (story and topic weighted)

training on 16, 8 or 4 on-topic stories for our topic tracking system. The system degrades somewhat when it is trained on 1 or 2 stories. When tested on the same stories, our tracker shows no degradation for using automatic speech recognition output instead of closed captioning or other manually-transcribed text. Finally, the system performs much better when output from several tracking algorithms is combined with a time decay.

Detection achieves good performance without resorting to a sophisticated clustering algorithm. Therefore, a significant goal of the research should be to generate better decision metrics. Also, we need to find ways of combining metrics, although this is complicated by the need for normalized scores.

ACKNOWLEDGMENTS: This work was supported by the Defense Advanced Research Projects Agency and monitored by Ft. Huachuca under contract No. DABT63-94-C-0063 and by the Defense Advanced Research Projects Agency and monitored by NRaD under contract No. N66001-97-D-8501. The views and findings contained in this material are those of the authors and do not necessarily reflect the position or policy of the Government and no official endorsement should be inferred.

REFERENCES

[1] "The Topic Detection and Tracking Phase 2 (TDT2) Evaluation Plan," Version 3.7. August 3, 1998.

[2] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang. "Topic Detection and Tracking Pilot Study Final Report." *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*. February, 1998.

[3] R. Schwartz, T. Imai, L. Nguyen, and J. Makhoul. "A Maximum Likelihood Model for Topic Classification of Broadcast News." *Eurospeech '97*, Rhodes, Greece. September, 1997.

[4] G. Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, NY. 1983.

[5] T. Leek, D. Miller, and R. Schwartz, "Labrador: A Hidden Markov Model Information Retrival System", *Technical Talk, GTE Internetworking*, Sept., 1998.