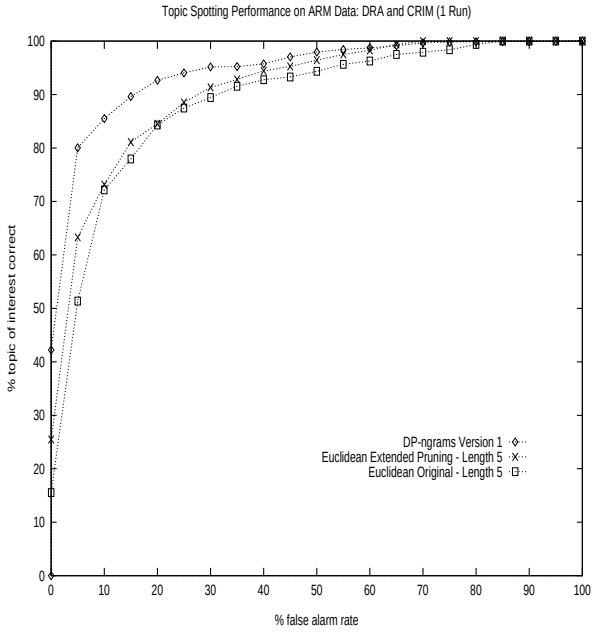
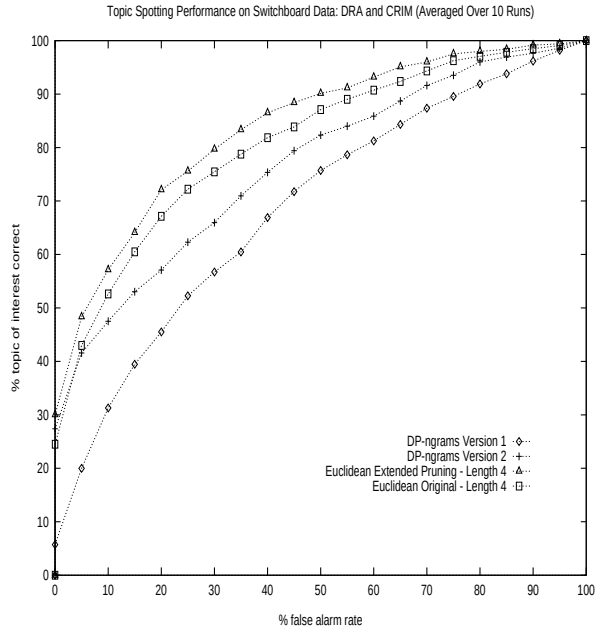
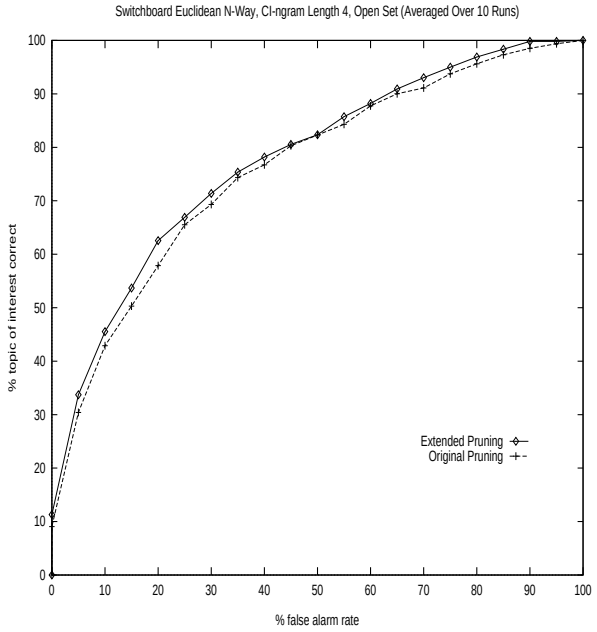
- Discriminative training gives good results: in a sense, minimum error rate selection for DP-ngrams and ENWN pruning are analogous.
- How good is phoneme-based topic spotting in practice - *e.g.*, when the ‘‘background’’ topics aren’t known in advance? Last graph shows ENWN open-set results: training = topic of interest + 5 topics, test = topic of interest + **other** 4 topics. The performance is surprisingly good!

6. ACKNOWLEDGEMENTS

Roland Kuhn and Caroline Drouin wish to acknowledge the financial support of Canada’s Ministry of Defense, and to thank Luc Gagnon and Karl Boutin for several good ideas.







7. REFERENCES

[1] R.C. Rose *et al*, “Techniques for Information Retrieval from Voice Messages”, *ICASSP-91*, Vol. I, pp. 317-320, Toronto, Canada, May 1991.

[2] L. Gillick *et al*, “Application of Large Vocabulary Continuous Speech Recognition to Topic and Speaker Identification Using Telephone Speech”, *ICASSP-93*, Vol. II, pp. 471-474, Minneapolis,USA, April 1993.

[3] J. McDonough *et al*, “Approaches to Topic Identification on the Switchboard Corpus”, *ICASSP-94*, Vol. I, pp. 385-388, Adelaide, Australia, April 1994.

[4] B. Peskin *et al*, “Improvements in Switchboard Recognition and Topic Identification”, *ICASSP-96*, Vol. 1, pp. 303-306, Atlanta, USA, May 1996.

[5] P. Nowell and R. Moore, ‘‘A Non-Word Based Approach to Topic Spotting in Speech’’, *DRA Memorandum No. 4815*, Oct. 1993.

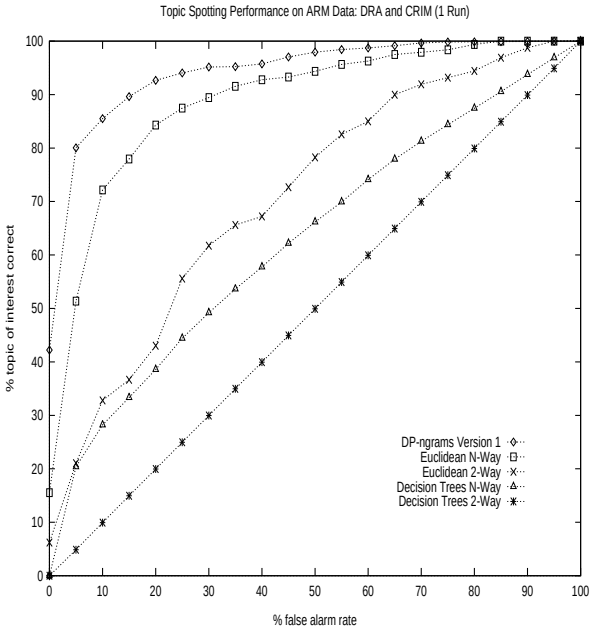
[6] P. Nowell and R. Moore, ‘‘The Application of Dynamic Programming Techniques to Non-Word Based Topic Spotting’’, *Eurospeech-95*, V. 2, pp. 1355-1358, Madrid, Spain, Sept. 1995.

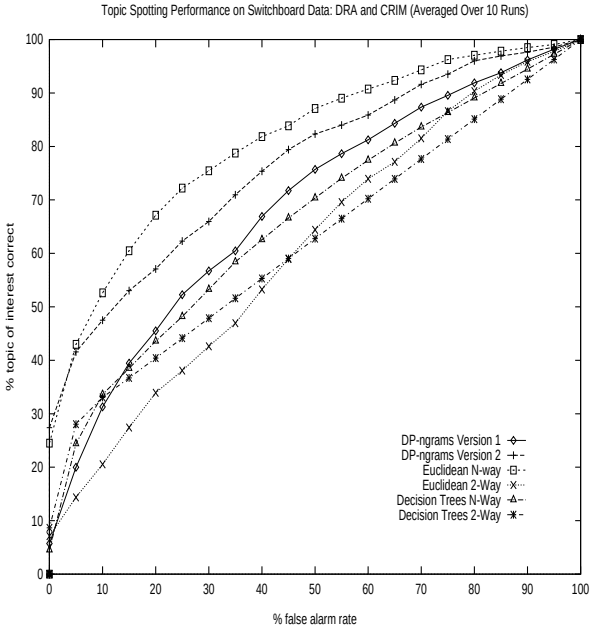
[7] J. H. Wright, M.J. Carey and E.S. Parris, ‘‘Topic discrimination using high-order statistical models of spotted keywords’’, *Computer Speech and Language*, Vol. 9, pp. 381-405, 1995.

[8] P. Nowell, N. Millner and A. Skilling, ‘‘Non-Word Based Topic Spotting Experiments on Switchboard’’, *Proc. of Institute of Acoustics*, V. 18 Part 9, pp. 391-397, Nov. 1996.

[9] L. Breiman, J. Friedman, R. Olshen, and C. Stone, ‘‘Classification and Regression Trees’’, Wadsworth, 1984.

decided to abandon decision trees, and to explore modifications of the Euclidean (ENWN) N-way approach.





4. MODIFICATIONS TO ENWN

4.1 Usefulness Transformation

Recall that for sequence *c* with frequency of occurrence *f(c)*, and topic of interest *T*, asymmetric usefulness is

$$U(c,T) = f(c) \log [f(c \mid T) / f(c \mid \textbf{not } T)].$$

The symmetric usefulness is

$$U'(c,T) = \max\{f(c) \log [f(c \mid T) / f(c \mid \textbf{not } T)],$$
$$f(c) \log [f(c \mid \textbf{not } T) / f(c \mid T)]\}.$$

Usefulness of either kind indicates how strongly a given sequence is correlated with the current topic of interest. Instead of using frequency vectors in the ENWN algorithm, new vectors obtained from the frequency vectors via a usefulness weighting were employed instead in experiments not shown here. Alas! both usefulness transformations yielded a marked deterioration in performance.

4.2 Varying Wordlengths

To generate the initial vocabulary, the maximum length of sequences must be specified. In previous experiments this maximum length had been arbitrarily fixed at 5 for ARM and 4 for Switchboard; now experiments in which this length was changed (not shown here) were carried out. Length 2 yields bad performance for both data sets; lengths 3-6 yield the same results for ARM. Surprisingly, length 3 yielded the best results for Switchboard. We speculate that inclusion of longer sequences creates an initial vocabulary so large that overtraining takes place - i.e., ENWN has too many degrees of freedom for the amount of training data.

4.3 Extended Pruning of Lexicon

This was the most successful modification to CRIM's original ENWN algorithm. Instead of using the full set of training files to decide when to stop pruning the lexicon, training data were split in the proportion 4:1 and held-out data used to estimate the appropriate size for the final lexicon, as a percentage. Once this value had been obtained, the ENWN algorithm was run on the full training set (using frequency vectors also obtained from the full set) and the lexicon pruned until it attained the appropriate size.

It turned out that this size was much smaller than in the original approach. The pruned lexicon averaged about 7% to 10% of the size of the initial lexicon compared with 94% to 98% in the original approach. For Switchboard, the initial lexicon contained about 55,000 sequences; after extended pruning, this went down to 4,000-5,000 sequences.

The next two graphs show that extended pruning of the lexicon yields excellent results. In fact, ENWN with extended pruning is clearly superior to all other approaches on Switchboard (DRA's Version 1 still yields the best result on ARM).

5. DISCUSSION

- ENWN is very crude: it has a primitive distance measure, and no probabilistic model. Why does it work so well?
- DP-ngrams are more sophisticated, and work well on ARM - but are computationally expensive.

data by the usefulness criterion, the DRA researchers noticed that the lists of the most 'useful' DP-ngrams for some topics were dominated by strange DP-ngrams representing long stretches of silence. The reasons for this were investigated; it was found that these DP-ngrams occurred with high frequency in a small number of files and rarely elsewhere. Although the 'usefulness' score of these DP-ngrams was high their dominance posed a problem: they only occurred in a few files, and were thus poor topic predictors.

DRA therefore designed the ''minimum error rate selection'' procedure for selecting DP-ngrams, which explicitly takes account of the distribution of occurrences across training files [8]. Even if it has a high usefulness score, a DP-ngram will not be included unless it decreases the expected error rate. The only novelty in this approach is the selection procedure: once the DP-ngrams have been selected the subsequent processing is the same. In the graphs that follow, the previous DP-ngram approach is called ''Version 1'' and the minimum error rate variant is called ''Version 2''.

2. WORK CARRIED OUT AT CRIM

In CRIM's experiments, a large set of phoneme sequences is first generated from the training files. All sequences that do not exceed a certain length (*e.g.*, that are 5 phonemes long or shorter) and that occur more than a certain number of times in the training files (*e.g.*, that occur more than 4 times) are put into this initial vocabulary.

2.1 Decision Trees

The CRIM team first tried decision trees [9] on both ARM data supplied by DRA Malvern and Switchboard data. The decision trees contain questions of form ''is *f(c) < k*?'' where *c* is a phoneme sequence. Both ARM and Switchboard contain several different topics, so it is not clear *a priori* whether it is better to lump all topics **not** of interest into a single class (the 2-way approach) or to keep them as separate classes (the N-way approach). For decision trees, it turns out to make little difference which we do (see following graphs).

2.2 Euclidean Nearest Wrong Neighbour Approach (ENWN)

Surprisingly, CRIM's best results were obtained with ENWN, in which each file and each topic is represented by a vector of frequencies (sequence count divided by file length), and a new file is assigned to the topic whose vector is nearest to it (Euclidean distance). The trick is the selection of sequences: those which, on average, tend to reduce the relative distance between a training file and the nearest topic vector that is a **wrong** topic are eliminated.

Let *C* be the current training conversation, and let *R* be the right topic and *F* the wrong (false) topic nearest to *C*. Let *S()* be the squared Euclidean distance between two frequency vectors, so that we wish to decrease

$$E(C,R,F) = S(C,R) - S(C,F)$$

Let C_i be the frequency of sequence *i* in *C*, and let R_i and F_i be its frequency in *R* and *F* respectively (an abuse of notation since *R* and *F* are really *R(C)* and *F(C)*). Then

$$E(C,R,F) = \sum_i \left(\left(C_i - R_i \right)^2 - \left(C_i - F_i \right)^2 \right)$$

Consider a vector *V* whose *i*th component is

$$V_i = \left(C_i - R_i \right)^2 - \left(C_i - F_i \right)^2$$

If we eliminate sequence *i*, the new value of *E(C,R,F)* will be $E(C,R,F) - V_i$. Let

$$I_i = \sum_C V_i = \sum_C \left(C_i - R_i \right)^2 - \left(C_i - F_i \right)^2$$

The ENWN algorithm calculates vector *I* with the current set of sequences on the training conversations, and then removes the sequence with the highest value of I_i. At the end of an iteration, the identity of the nearest-wrong topic *F(C)* may change for some conversations *C*. In the original version of this algorithm, our stopping criterion for the pruning of the initial vocabulary was that every I_i be negative.

Only the sequences that have survived this pruning process are used for classification of new conversations. Topics not of interest can be pooled into a single topic (2-way Euclidean) or kept separate (N-way Euclidean).

3. INITIAL RESULTS

The ROC graphs for ARM (7 topics) and Switchboard (10 topics) show the tradeoff between the false alarm rate and the % topic of interest correct (averaged over all topics). The 385 ARM files were divided into 224 training files (32 files per topic) and 161 test files (23 files per topic); only one run was done. For Switchboard, 507 files were used; these were arranged in the ratio 9:1 training:test in 10 different ways to obtain 10 runs. After obtaining the two graphs shown on the next page, the CRIM researchers

APPROACHES TO PHONEME-BASED TOPIC SPOTTING: AN EXPERIMENTAL COMPARISON

Roland Kuhn[1], Peter Nowell[2], and Caroline Drouin[3]

[1]Speech Technology Laboratory, Panasonic Technologies Inc., 3888 State St., Santa Barbara, California 93105, U.S.A. (email: kuhn@STL.research.panasonic.com)

[2]DRA Malvern, St. Andrews Road, Malvern, Worcestershire, United Kingdom WR14 3PS

[3]Centre de recherche informatique de Montréal (CRIM), 1801 McGill College, Montréal, Canada H3A 2NA

ABSTRACT

Topic spotting is often performed on the output of a large vocabulary recognizer or a keyword spotter [1-4]. However, this requires detailed knowledge about the vocabulary, and transcribed training data. If portability to new topics and languages is important, then a topic spotter based on phoneme recognition is preferable [5]. A phoneme recognizer is run on training data consisting of audio files labeled by topic alone - no word transcripts are required. Phoneme sub-sequences which help to predict the topic are then extracted automatically. The work described here was carried out by two teams exploring three very different approaches to phoneme-based topic spotting: the ''DP-ngram'', the ''decision tree'', and the ''Euclidean'' approach. Results obtained by each team on the ARM (Airborne Reconnaissance Mission) and Switchboard data sets were compared by means of Receiver Operating Characteristic (ROC) curves. The best performance for each team was obtained via a similar type of discriminative training.

1. CI-NGRAMS AND DP-NGRAMS

1.1 Early Work

Early work on non-word-based topic spotting was carried out at DRA Malvern and used variable-length context-independent (CI) ngrams for topic spotting on selected Airborne Reconnaissance Mission (ARM) reports [5,6]. The CI-ngrams used in these experiments are sequences of phonemes observed in multiple contexts in the training data. A list of CI-ngrams is generated using conventional string matching techniques; CI-ngrams are then ranked according to their 'usefulness' scores [7]. For CI-ngram *c* with frequency of occurrence *f(c)*, the 'usefulness' of *c* for topic of interest *T* is defined as

$$U(c,T) = f(c) \log [f(c \mid T) / f(c \mid \textbf{not } T)].$$

To decide whether a new file belongs to the topic of interest, the DRA algorithm accumulated the scores of a small number of the most useful CI-ngrams observed in the file and compared the total with a threshold.

Although this approach can work well, it is limited by the length of the CI-ngrams, which are typically only a few phonemes long. Although there is no theoretical maximum length, in practice the length is limited because of recognition errors and variations in pronunciation. Utterances of the same word or phrase are unlikely to be recognized in the same way each time, and the probability of observing any particular CI-ngram decreases exponentially with its length.

DRA therefore turned to DP-ngrams, defined by means of a dynamic programming algorithm which allows partial as well as exact matches. Similar sub-sequences are picked from the training data, and then clustered. Each cluster is defined by its centroid and a permissible distance from that centroid expressed as a function of the number of insertions, deletions, and substitutions. Employing the same selection techniques as before, the DRA researchers were able to obtain longer fragments and significantly improved topic-spotting performance on ARM, at the expense of extra computation [6].

1.2 DP-ngrams on Switchboard Data

In the fall of 1995, DRA Malvern and the Centre de recherche informatique de Montréal (CRIM) began to exchange phoneme sequences. The DRA Malvern team now had access to Switchboard transcripts from the CRIM phoneme recognizer. As it turned out, the much greater length of Switchboard files (5-10 min., as opposed to ~30 sec. for ARM reports) necessitated complete reimplementation of the dynamic programming algorithms to reduce memory and computational requirements. Initial results of the DP-ngram approach on Switchboard were disappointing.

1.3 Minimum Error Rate Selection

Studying the DP-ngrams chosen from Switchboard training