TALKER-INDEPENDENT KEYWORD SPOTTING FOR INFORMATION RETRIEVAL

J. T. Foote[1] *G. J. F. Jones*[1,2] *K. Sparck Jones*[2] *S. J. Young*[1]

[1]Cambridge University Engineering Department, Cambridge, CB2 1PZ, UK
[2]Cambridge University Computer Laboratory, Cambridge, CB2 3QG, UK

ABSTRACT

The goal of the Video Mail Retrieval (VMR) project is to integrate state-of-the-art information retrieval (IR) methods with high-accuracy word spotting to yield a robust and efficient multimedia retrieval system. This paper concerns open-talker and arbitrary-keyword retrieval based on talker-independent subword models. Because talker-independent subword models can not be expected to work as well as the talker-dependent whole-keyword models used in previous VMR experiments, speaker adaptation is investigated as a means of improving performance (especially for talkers with non-British accents). Both standard FOM word spotting measures and actual retrieval results are computed. The results show that the FOM is not necessarily a good indicator of retrieval performance, and that talker adaptation can substantially improve both spotting and retrieval results.

1. THE VIDEO MAIL RETRIEVAL TASK

The last few years have seen an increasing use of multimedia applications, including video and audio mail. Searching a large archive of video messages poses a significant problem, because, unlike text, there are no simple ways to search for a particular reference. The Cambridge Video Mail Retrieval (VMR) project is addressing this problem by developing a system to retrieve stored video messages by voice.

1.1. The VMR message corpus

For the initial development of the VMR system, it was necessary to create an archive of messages with known audio and information characteristics to evaluate both word spotting and message retrieval performance [1]. A fixed set of 35 keywords was chosen to cover ten "categories" chosen to reflect the anticipated messages of actual users; for example, the "management" category includes the keywords "staff," "time," and "meeting." Keywords may be associated with more than one category; and the keyword set includes 11 difficult monosyllabic words (e.g "date" and "mail"), as well as overlapping words (e.g. "word" and "keyword") and word variants (e.g. "locate" and "location").

Fifteen talkers (11 men and 4 women) each provided about 45 minutes of speech data for a total of 5 hours of read training data and 5 hours of spontaneous speech messages. Data was recorded at 16 kHz from a Sennheiser HMD 414 close-talking microphone. The training data, used as enrolment data for talker adaptation, consisted of isolated keywords, read sentences containing keywords in context, and phonetically-rich sentences not containing keywords (only the keyword-containing read sentences were used here). For the message data, talkers were asked to compose a spontaneous response to a prompt. A total of 50 unique prompts were used, five for each category. For each prompt, 6 messages were recorded (from 6 different users). The resulting 300 spontaneous messages, along with their text transcriptions, serve as a test corpus for both the keyword spotting and IR experiments. In addition, the WSJCAM0 British English speech corpus provided training data for the baseline talker-independent models [2].

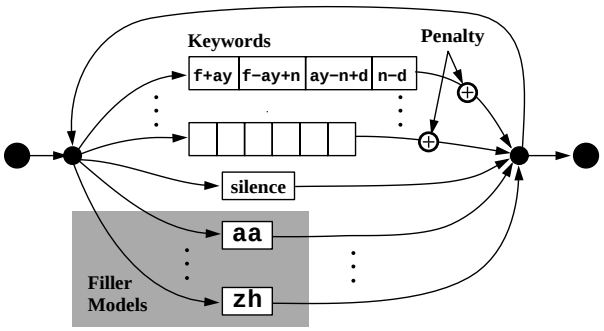

Figure 1. Keyword recognition network.

2. KEYWORD SPOTTING

The initial VMR system depends on talker-dependent whole-keyword models [3], which imposes an unrealistic constraint: messages can be retrieved only for talkers and keywords that are modelled. This paper presents preliminary work on using subword HMMs for talker-independent (though still fixed-vocabulary) spotting.

A full set of 8-mixture word-internal triphone HMMs was trained on British English speech using a tree-based clustering technique [4]. An advantage of this training method is that all possible triphones, biphones and monophones can be modelled, yet because most states are tied, the full model set is relatively compact. Given such a model set, a particular keyword may be easily modelled by concatenating the appropriate sequence of subword models (obtained from a phonetic dictionary). Biphones are used at the beginning and end of the keyword, while triphones model the internal structure. For example, the keyword "find" is represented by the model sequence `f+ay f-ay+n ay-n+d n-d`. Non-keyword speech is modelled by an unconstrained network of monophones (denoted "filler models"), as in Figure 1.

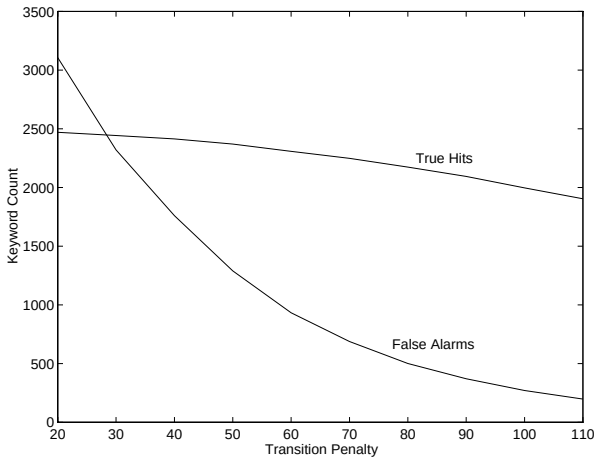

Figure 2. False alarms vs. keyword transition penalty (*R75* models).

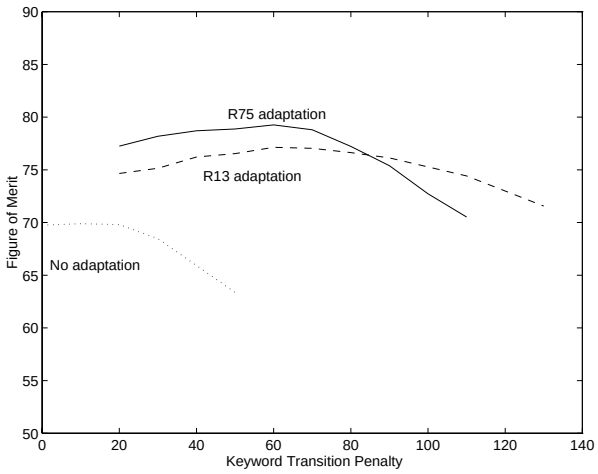

Figure 3. Figure of merit vs. keyword transition penalty.

2.1. Keyword Recognition

Keyword spotting is done with a two-pass recognition procedure. First, Viterbi decoding is performed on a network of just the filler models, yielding a time-aligned sequence of the maximum-likelihood filler monophones and their associated log-likelihood scores. Secondly, another Viterbi decoding is done using a network of the keywords, silence, and filler models in parallel (Figure 1). In a manner similar to Rose [5], keywords are rescored by normalizing each hypothesis score by the the average filler model score over the keyword interval. Unlike [5], however, the average log likelihoods are divided rather than subtracted, which results in somewhat better performance [6].

2.2. Talker Adaptation

The VMR corpus is realistic in that it contains talkers with varied accents. This is problematic when using models trained exclusively on British English talkers. In an attempt to ameliorate this problem, and increase spotting performance in general, talker-adaptation was investigated. The chosen approach was maximum-likelihood linear regression because it has been shown to improve recognition with a comparatively small amount of enrolment data [7]. This method involves adapting the means of the HMM gaussian mixtures to increase the likelihood of the enrolment data. Varying amounts of the VMR corpus training data were used as enrolment data for talker-adaptation experiments.

An unfortunate consequence of talker adaptation is that it increases the number of false alarms, as the average model likelihood is increased by the adaptation process. A solution was to introduce a separate transition penalty to the keyword models; adjusting this value changes the likelihood of the keywords relative to the filler models and thus allows the false alarm rate to be controlled. Figure 2 shows that increasing the penalty dramatically reduces the number of false alarms, while only mildly impacting the number of correctly identified keywords. The net effect is similar to, but better than, increasing the threshold on keyword scores (such that those with low scores are ignored). While increasing the transition penalty has only a small effect on the FOM (which is designed to be relatively threshold-independent), it appreciably increases the IR performance, as in Figure 4.

Talker	*Depen*	*Indep*	*R13*	*R75*
51	66.67	64.52	69.18	67.50
52	75.72	56.15	63.89	69.79
53	70.45	72.82	77.14	77.84
54	83.81	63.23	69.72	73.37
55	84.55	84.55	92.58	91.97
56	77.62	65.71	72.65	73.00
57	76.63	71.06	70.50	69.79
58	88.79	69.87	78.83	84.35
59	88.73	80.34	89.75	89.76
60	84.85	80.42	84.04	90.49
61	65.41	53.99	59.11	65.22
62	88.49	80.75	87.06	89.05
63	95.29	85.92	87.34	85.95
64	87.22	41.98	80.07	77.94
65	82.90	77.07	73.80	82.86
Mean	81.14	69.89	77.13	79.26

Table 1. Keyword spotting figures of merit (using *a posteriori* best transition penalty).

2.3. Keyword Spotting Results

An accepted figure-of-merit (FOM) for word spotting is defined as the average percentage of correctly detected keywords as the threshold is varied from one to ten false alarms per keyword per hour. Keyword spotting results were scored against aligned text transcriptions containing the keywords. The FOM results for the different models and talkers are shown in Table 1. For comparison, the average FOM from the best talker-dependent whole-keyword models is 81.1% (row *Depend* in Table 2, from [8]). The talker-independent subword-model results are in column *Indep*, while the remaining columns present talker-adaptation results using various amounts of adaptation data. The *R13* column used 13 utterances of enrolment data, while the *R75* column used 75. Enrolment utterances were keyword-rich read sentences; the *R13* data contained at least two utterances of each of the 35 keywords, while the *R75* data had at least five[1]. Though adaptation does not uniformly improve all talkers, the average increase is substantial, and is particularly dramatic

[1] Additional experiments (not presented here) indicate that performance increased only slightly when using enrolment data not containing keywords.

for talker 64 (who has an American accent ill-suited to the British English models). Using just a small amount of enrolment data improved the FOM performance substantially, though increasing the enrolment data by a large factor did not yield FOM improvements of the magnitude expected from large-vocabulary recognition experiments [9]. Still, the improvements nearly match the talker-dependent results.

3. INFORMATION RETRIEVAL

Information retrieval (IR) techniques are used to satisfy an operator's information need by retrieving relevant messages from an archive. In practice, the operator composes a search *query* by typing in a sequence of words; from this a group of messages, ranked by *relevance* score, is returned. The operator can then browse the high-scoring messages to find the desired information.

3.1. IR Experiments

Information retrieval experiments require message *queries*, expressing a user's information need, and assessments of message *relevance* to the queries. Queries and assessments were simulated for our tests as follows. Queries were constructed from the message prompts used in the database recording. To reduce variations in word form that inhibit retrieval matching, query words were suffix-stripped to stems using a standard algorithm [10]. Queries were formed from the prompts by selecting those stems also found in the keyword stem list. For example, given the prompt

```
Your current project is lagging behind schedule.
Send a message pointing this out to the other
project management staff. Suggest some days and
times over the next week when you would be
willing to hold a meeting to discuss the
situation.
```

the following query was obtained:

```
project messag project manag staff time meet
```

Word fragments such as "`messag`" are the suffix-stripped keyword roots. The 6 recorded messages generated in response to each prompt were assumed relevant to the query constructed from that prompt. Note that the 24 other messages in the same category, which are quite likely to contain similar keywords, are assumed to be not relevant; retrieval of one of these messages is construed as an error. Thus, the retrieval task is comparatively difficult.

Given a query, the query-message score is a weighted sum of the keywords common to both query and message [11]. A common weighting scheme is the *inverse document frequency* (idf) weight

$$idf_i = \log \frac{N}{n[i]},$$

where N is the total number of messages and $n[i]$ is the number of messages that contain keyword i. Thus keywords occurring in a small number of messages are favoured, and keywords common to many messages (especially those prone to false alarms) are discouraged. This weighting scheme does not depend on the total number or acoustic scores of keyword hits; the query-message score is simply the sum of keyword weights for every above-threshold keyword found at least once in both the query and the message.

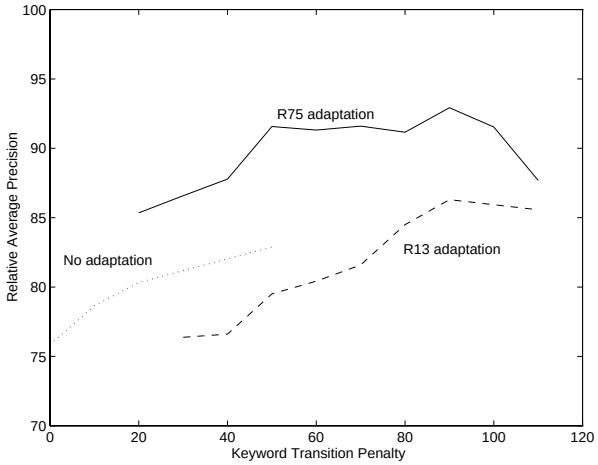

Figure 4. Average retrieval precision vs. keyword transition penalty (at best *a posteriori* score threshold).

3.1.1. Measuring IR performance

Retrieval performance is often measured by *precision*, the proportion of retrieved messages that are relevant to a particular query. One conventional single-number performance figure, *average precision*, is derived as follows: the precision values obtained at each new relevant message in the ranked output for an individual query are averaged, and the results are then averaged across the query set. Other retrieval evaluation metrics are available and generally preferable, but this single-number performance measure is useful for comparing text and word spotting results.

Acoustic word spotting is prone to false alarms and missed keywords which adversely affects retrieval performance. The degradation due to imperfect word spotting can be measured by comparing retrieval performance for spotting with that for text transcriptions of the messages. A particular problem with word spotting is that unrelated acoustic events will often resemble valid keywords. For example, the last part of "hello Kate" is acoustically quite similar to the keyword "locate." Because even the most accurate acoustic models cannot discriminate between homophones, the output of an ideal word spotter that reports all keyword phone sequences provides a more legitimate standard of comparison than text.

3.2. Information Retrieval Results

When using keyword spotting results in an application, typically a threshold is set on the acoustic score. Keywords with scores above the threshold are considered true hits, while those with scores below are considered false alarms and ignored. Choosing the appropriate threshold is a tradeoff between the number of Type I (missed keywords) and Type II (false alarm) errors, with the usual problem that reducing one increases the other. Figure 5 shows how the IR performance varies with the choice of score threshold and transition penalty. At low threshold values, retrieval performance is somewhat impaired

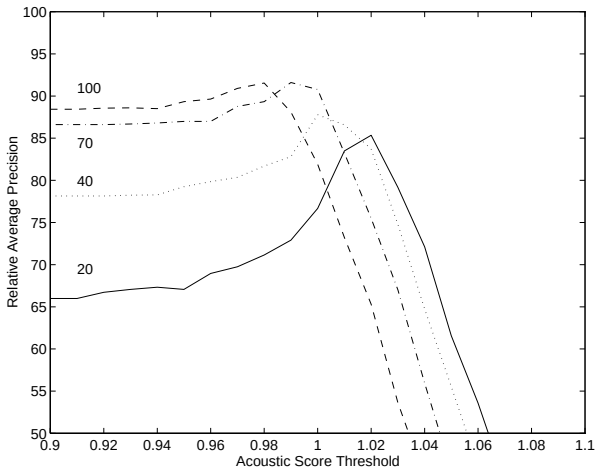

Figure 5. Relative average precision vs. keyword score threshold ($R75$ models).

by a high proportion of false alarms (Type II errors); conversely, higher thresholds (towards the right) remove a significant number of true hits (Type I errors), also degrading performance. The higher transition penalties remove a large proportion of the false alarms even before the threshold is applied, resulting in IR performance that is both better and flatter (less dependent on the score threshold). A major disadvantage of the FOM is that because it is threshold-independent, it can't be used to find an appropriate operating point. In addition, the FOM score is not sufficiently correlated with retrieval performance to allow the appropriate transition penalty to be chosen from the FOM alone.

Table 2 compares acoustic retrieval performance with ideal text and phonetic standards, at the *a posteriori* best transition penalties and score thresholds. Talker adaptation clearly benefits the IR performance, as does increasing the transition penalty. For comparison with the IR results, the average FOM of Table 1 is reproduced in the last column.

4. CONCLUSIONS

This paper has described experiments comparing speaker dependent and speaker independent keyword spotting. As in all our work, we are particularly interested in understanding the relationship between word spotting accuracy and retrieval performance. The results show that the performance drop when going from speaker-dependent to speaker-independent operation is substantial, but adaptation can recover most if not all of this, even using a comparatively small amount of enrolment data (provided enough keyword instances are included). Though not addressed in these experiments, using subword models allows arbitrary keywords to be easily modeled; using an open keyword set should substantially improve the absolute IR performance.

Though the best word spotting FOM and the best retrieval performance are related, the IR performance (like most applications of word spotting) is sensitive to the chosen operating point. It has been shown that the FOM, which is calculated over a range of possible thresholds, is not necessarily a good measure of the potential IR performance.

	Abs.	Text	Phon.	FOM
Text	0.332	100%	—	—
Phonetic	0.317	95.3%	100%	—
Depen	0.295	88.8%	93.2%	81.14%
Indep	0.263	79.0%	82.9%	69.89%
R13	0.271	82.2%	86.3%	77.13%
R75	0.290	87.3%	92.9%	79.26%

Table 2. Relative retrieval performance.

5. ACKNOWLEDGEMENTS

This project is supported by the UK DTI Grant IED4/1/5804 and EPSRC Grant GR/H87629. We wish to thank Julian Odell for baseline triphone models, and Chris Leggetter and Phil Woodland for talker-adaptation software.

REFERENCES

[1] G. J. F. Jones, J. T. Foote, K. Sparck Jones, and S. J. Young. VMR report on keyword definition and data collection. Technical Report 335, Cambridge University Computer Laboratory, May 1994.

[2] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals. WSJCAM0: A British English speech corpus for large vocabulary continuous speech recognition. In *Proceedings of ICASSP 95*, Detroit, 1995. IEEE.

[3] J. T. Foote, G. J. F. Jones, and S. J. Young. Video Mail Retrieval Using Voice: Report on whole word based keyword spotting. Technical report, Cambridge University Engineering Department, September 1994.

[4] S. J. Young, J. J. Odell, and P. C. Woodland. Tree-based state tying for high accuracy acoustic modelling. In *Proc. ARPA Spoken Language Technology Workshop*, Plainsboro, NJ, 1994.

[5] R. C. Rose. Techniques for information retrieval from speech messages. *Lincoln Laboratory Journal*, 4(1):45–60, 1991.

[6] K. M. Knill and S. J. Young. Speaker dependent keyword spotting for hand-held devices. Technical Report 193, Cambridge University Engineering Department, July 1994.

[7] C. Leggetter and P. Woodland. Flexible speaker adaptation for large vocabulary speech recognition. In *Proceedings of Eurospeech 95*. ESCA, 1995.

[8] G. J. F. Jones, J. T. Foote, K. Sparck Jones, and S. J. Young. Video Mail Retrieval: the effect of word spotting accuracy on precision. In *Proceedings of ICASSP 95*, Detroit, 1995. IEEE.

[9] C. Leggetter and P. Woodland. Flexible speaker adaptation using maximum likelihood linear regression. In *Proc. ARPA Spoken Language Technology Workshop*, Barton Creek, 1995.

[10] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, July 1980.

[11] S. E. Robertson and K. Sparck Jones. Simple, proven approaches to text retrieval. Technical Report 335, Cambridge University Computer Laboratory, Dec 1994.