**Techniques for automatically transcribing unknown keywords for open keyword set HMM-based word-spotting**

K. M. Knill & S. J. Young

**CUED/F-INFENG/TR 230**

September 1995

Cambridge University Engineering Department
Trumpington Street
Cambridge CB2 1PZ
England

Email: kmk@eng.cam.ac.uk

**Abstract**

Many word-spotting applications require an open keyword vocabulary, allowing the user to search for any term in an audio document database. In conjunction with this, an automatic method of determining the acoustic representation of an arbitrary keyword is needed. For a HMM-based system, where the keyword is represented by a concatenated string of phones, the keyword phone string (KPS), the phonetic transcription must be estimated. This report describes automatic transcription methods for orthographically spelt, spoken, and combined spelt and spoken, keyword input modes.

The spoken keyword example case is examined in more detail for the following reasons. Firstly, interaction with an audio-based system is more natural than typing at a keyboard or speaking the orthographic spelling. This is of particular interest for hand-held devices with no, or a limited, keyboard. Secondly, there is likely to be a high occurrence of real names and user-defined jargon in retrieval requests which are difficult to cover fully in spelling based systems. The basic approach considered is that of using a phone level speech recogniser to hypothesise one or more keyword transcriptions. The effect on the KPSs of the number of pronunciation strings, the HMM complexity, and the language model used in the phone recogniser, and the number of sample keyword utterances is evaluated through a series of speaker dependent word-spotting experiments on spontaneous speech messages from the Video Mail Retrieval database.

Overall it was found that speech derived KPSs are less robust than phonetic dictionary defined KPSs. However, since the speech-based system does not use a dictionary it has the advantage that it can handle any word or sound. It also requires less memory. Given a single keyword utterance, producing multiple keyword pronunciations using a 7 N-best recogniser was found to give the best word-spotting performance, with a 9.3% drop in performance relative to the phonetic dictionary defined system for a null grammar, monophone HMM-based KPS recogniser. If two utterances are available, greater robustness can be achieved as the problem of poor keyword examples is partially overcome. Again, a 7 N-best approach yielded the best performance (6.1% relative drop), but good performance was also achieved using the Viterbi string for each utterance (8.5% relative drop), which has a lower computational cost.

**Keywords:** transcription, word-spotting, speech recognition, information retrieval.

# Contents

# 1 Introduction

There are many applications, such as personal computing devices, where the use of a fixed keyword set to retrieve documents would be a hindrance, restricting the functionality of the device. This report, therefore, considers the open keyword set case, where the user may search for any word (or sound) in an audio document database. In particular, this report addresses the problem of automatically determining an acoustic representation for an arbitrary keyword in a HMM-based word-spotting system.

Defining a new keyword is akin to adding a new word to a speech recognition system, except that there is no need to spell the new word or incorporate it into a language model. The search request may be specified using text and/or speech, dependent on the device. As the keyword vocabulary is unlimited, no *a-priori* assumptions about the keyword can be made. When specifying a keyword verbally, a single example of the keyword will generally be given, with an expected upper limit of three examples, so there will be too little data to train a wholeword HMM for the keyword. If the keyword is specified textually, a dictionary or letter-to-sound translator must be used. For both input types, the keyword is modelled by concatenating a string of pre-trained sub-word (phone) HMMs, referred to as the *keyword phone string* (KPS).

As indicated, the determination of the KPS depends upon the form in which the keyword is represented in the search request. If a text input is used, then, the keyword is initially defined in terms of its orthographic spelling. This spelling can also be obtained by speaking the letters in turn if a speech input is used. Since confusions are common between similar sounding letters, e.g. "b" and "e", the letters have to be entered using the phonetic alphabet, even so recognition errors are likely to occur. A far more natural approach is simply to speak the keyword request, so that the keyword is defined in terms of its acoustic features. Methods for determining the KPS from an orthographic spelling (assumed perfect), and from one or more spoken example of the keyword are discussed in sections 2 and 3 respectively. Combined methods are discussed in section 4.

Once the KPS has been determined, the audio document database is searched for occurrences of the keyword using a word-spotter, as illustrated in figure 1. The aim of the KPS determination stage is to produce a string that is a good acoustic match of the keyword examples contained in the database, to yield good word-spotting performance. This is therefore used as a measure to compare different approaches. Experimental results are presented in section 5 for the spoken keyword case.

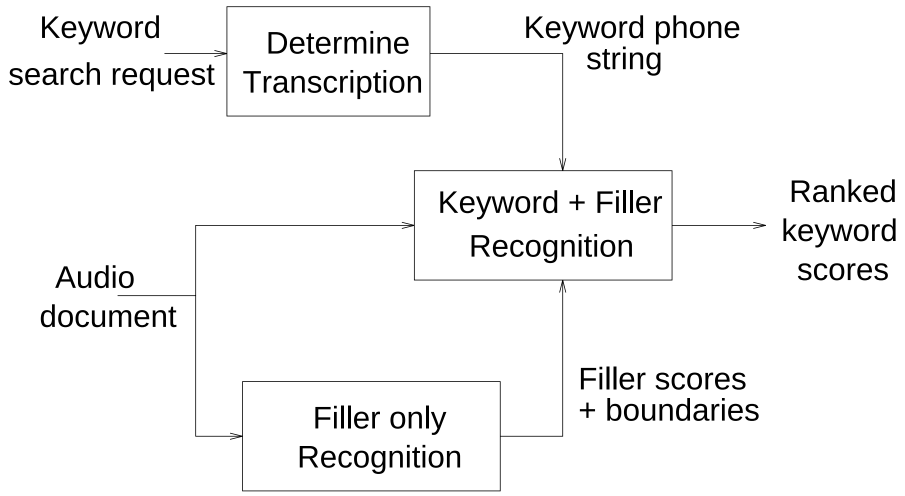


Figure 1: HMM-based open keyword-spotting system

The spoken keyword example case has been focused on in this work for the following reasons. Firstly, many word-spotting devices are user-specific, such as personal memo and dictation tools. There is, therefore, likely to be a high occurrence of real names and user-specific jargon in retrieval requests. It is difficult to get complete coverage of such terms in spelling based systems, which are either based on a pronunciation dictionary or a text-to-phone mapper. Secondly, speech input is the only possibility on hand-held devices where there is a limited or no keyboard. The technology behind these devices is expected to make wide use of word-spotting systems. Although orthographic spellings may be entered by voice, the process is relatively unnatural and requires good error checking with the user.

## 2 Orthographically spelt keyword request

Earliest attempts to construct a phonetic transcription for a new word were in the area of speech synthesis. Various approaches are reviewed in Klatt [7]. Given an orthographic transcription of the new word, most systems look it up in a phonetic dictionary. If the word does not exist in the dictionary, then some set of human-derived text-to-phone rules is generally used to model the word. The best text-to-speech systems produced correct transcriptions for 95-98% of words in running text.

Asadi et al [1] used the text-to-sound rules from DECtalk[1] to provide transcriptions for new words in a continuous speech recognition (CSR) system that were not found in their pronunciation dictionary. Their work showed that DECtalk was insufficient on its own to recognise new words. For example, when 41 words in the 1000 word Resource Management corpus were transcribed by DECtalk (the remainder were hand-transcribed), recognition errors were observed for 9 out of the 41 words, and 5 words were misrecognised all the time.

Lucassen et al [11], and Bahl et al [2] used an information theoretic approach to generate a set of spelling-to-sound rules. These consisted of a set of probabilistic mappings between series' of letters and corresponding pronunciation strings. For the pronunciation string, $R = r_1, \ldots, r_m = r_1^m$, and string of letters, $L = l_1, \ldots, l_n = l_1^n$,

$$\begin{aligned} P(R|L) &= \prod_{i=1}^{n} p(r_i|r_1^{i-1}, l_1^n) \qquad (1) \\ &\approx \prod_{i=1}^{n} p(r_i|r_{i-5}^{i-1}, l_{i-5}^{i+5}) \end{aligned}$$

This assumes that the probability of pronunciation $r_i$ only depends on the current letter, the 5 previous and following letters, and the 5 previous pronunciations, referred to as the context of the pronunciation.

To estimate $p(R|L)$, the contexts are mapped onto a relatively small number of equivalence classes with the aid of a decision tree. The data is partitioned into 2 sub-collections for each context element of each data item by asking binary questions, such as "Is the next letter a vowel?". There is a probability distribution over the pronunciations $r$ corresponding to each sub-collection. The question and context element which minimise the average entropy of the two distributions corresponding to the two sub-collections is selected. The process is repeated until some termination criterion is reached. A recursive smoothing scheme is then used to smooth the distributions at the leaves of the tree. Preparing the data for the estimation process is a labour intensive task, involving the derivation of consistent pronunciations over a number of data sources.

On a 20K continuous speech recognition task, with 500 new words a 77% drop in performance was noted for this scheme, relative to the performance given by hand-derived transcriptions. A third of the text derived transcriptions had errors in them, where an error is considered to be a substantial discrepancy between the artificially generated and hand-written transcriptions so that there would be a fair chance of recognition errors resulting from the automatic transcription. The spelling-to-sound rules are, therefore, insufficient on their own to provide good transcriptions.

The reasons for the poor performance of the spelling derived transcriptions in CSR was surmised by Bahl [2] to be due to a number of reasons. The most significant being that the new words

[1]DECtalk is a registered trademark of Digital Equipment Corporation.

in a CSR system are likely to be names and task-specific jargon, which tend to have highly irregular pronunciations and present substantial difficulties for text-to-speech systems. This is equally true of an open-keyword system, where a high occurrence of names and user-specific jargon can be expected in the search requests. In addition, Bahl noted that talkers develop idiosyncratic pronunciations of many words, especially proper names, and that the spelling of a word sometimes has little correlation with its pronunciation.

# 3 Spoken keyword request

Spoken keyword requests have the obvious advantage of being a more natural method of interacting with an audio-based system than typing at a keyboard. For hand-held devices with no, or a limited, keyboard they are a more appropriate form of input than speaking the new word's spelling. In addition, several of the drawbacks of spelling-based transcription systems, described in the previous section, do not apply, such as the problem of coverage of real names.

Given a spoken keyword request, then the simplest approach to determining the KPS is to pass the request through an isolated word recogniser. The optimal word acoustic match is then used to model the keyword in the word-spotter. This approach has similar limitations to its text equivalent, dictionary look-up, in that complete coverage of the keyword vocabulary will be difficult to achieve. Some steps can be taken to overcome this by allowing the keyword to be built up from more than one word, for example $Philip \rightarrow fill \;\; lip$. The recogniser vocabulary can, however, still grow to a large size, and is relatively inefficient computationally.

Alternatively, a phone level speech recogniser can be used, as shown in figure 2. The speech frames corresponding to the keyword are first identified (section 3.1), then, the KPS is hypothesised using the phone level recogniser. The phone string determined is dependent on the type of recogniser and number of keyword examples, and the set of acoustic sub-word models and grammar constraints employed in the recogniser. These are discussed in sections 3.2 to 3.4.

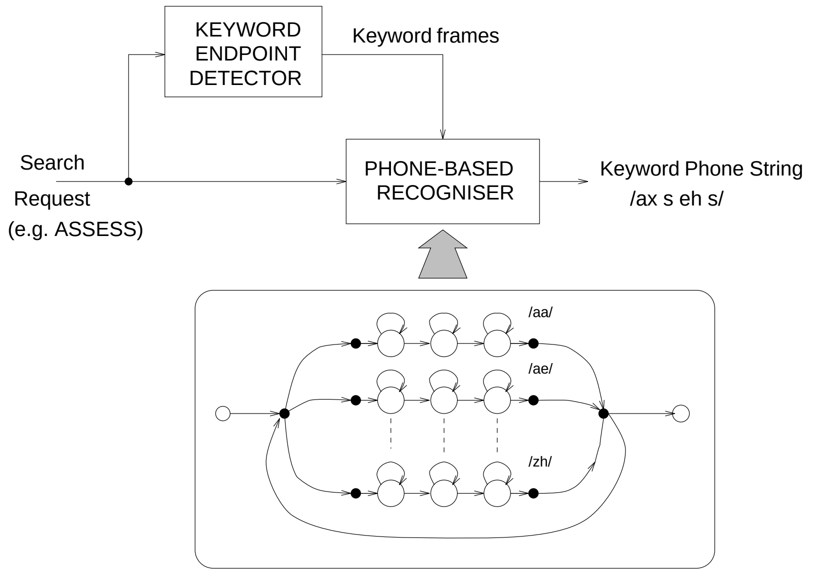


Figure 2: Determination of KPS from spoken keyword request using phone-level speech recogniser

## 3.1 Location of keyword frames

Determining the speech frames corresponding to the keyword is an endpoint detection problem, that is the detection of the transition point between speech and non-speech events. Reliable endpoint detection is difficult due to speech sounds which can be 'lost' against the background noise level, such as weak fricatives (e.g. /f/) or plosives (e.g. /p/), and non-speech noises, such as tongue clicks, lip smacks, and environmental noise, which have similar acoustic characteristics to speech.

A widely used method is that of Rabiner and Sambur [12], which looks for changes in energy and zero-crossing rate. Thresholds are assigned to each indicator, above which speech is said to be present. The endpoint of the speech frames corresponds to the point at which one of the indicators is below its threshold, and the other is crossing the threshold point. Another approach that has been found to be effective is to use a speech recogniser, where non-speech events are modelled and incorporated into the recognition network with speech events. The transition boundaries between the non-speech and speech models are taken as the speech endpoints [4].

As the keyword can be either entered alone or as part of the request, for example "look for Liverpool", any method must be applicable to both isolated and continuous speech input. The speech recogniser approach is better suited to this case. For the isolated keyword case, a network similar to the following is used (using standard HTK notation [15])

```
$PHONE = aa | ae | ... | zh ;

( sil < $PHONE > sil )
```

and for the full request case

```
( sil LOOK [ sil ] FOR [ sil ] < $PHONE > sil )
```

The latter, also, allows the device to detect a search request from a set of voice commands at the same time as the keyword frames are determined.

## 3.2 Phone level recogniser

The choice of the type of recogniser used to determine the KPS is dependent on the number of utterances available.

### 3.2.1 Single utterance

A single utterance of any word will not necessarily be representative of its general pronunciation. However, if it is assumed to be representative then the simplest method of obtaining the KPS is to take the optimal Viterbi path recognised [2, 1, 14]. Errors in the Viterbi KPS caused by recognition errors and non-standard pronunciation, led Kupiec et al [10] to propose the application of a pronunciation dictionary and confusion matrices to produce an N-best phone string list from the original Viterbi string. This provided multiple pronunciations of the keyword to counter some of the errors and the effect of the assumption. Confusion matrices were used to produce multiple pronunciations from the Viterbi string. Phonetic dictionary look-up was then performed for each string. The pronunciations that existed in the dictionary were retained.

Kupiec's method has the disadvantage that it will only detect words that exist in the dictionary. In addition, it has a large storage requirement that is unsuited to the hand-held devices where voice input will be most important. Knill and Young [9], therefore, proposed constructing an N-best list directly from the recognition process, by using one or more string hypotheses output by an N-best phone recogniser to represent the keyword. Figure 3 shows the general method. The top N phone strings, in terms of log-likelihood score, are selected to represent the keyword. The 1-best case is equivalent to the Viterbi case. Preliminary tests showed that word-spotting performance could be improved by increasing the number of phone strings in this way [9].

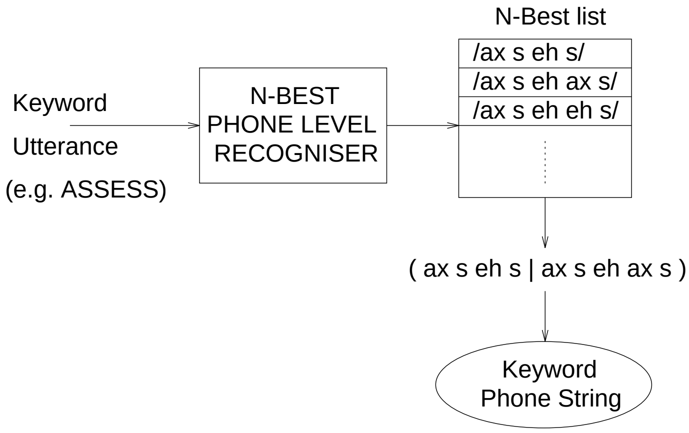


Figure 3: Determination of keyword phone strings using an N-best recogniser

### 3.2.2 Multiple utterance

The effects of a non-standard keyword utterance are lessened if several examples of the keyword are provided to build the KPS from. When multiple utterances have been used to generate transcriptions for continuous speech recognition reduced error rates have been observed, improving with the number of unknown word utterances [2, 5]. However, asking the user to repeat a keyword any more than 3 times would probably be unreasonable.

In the multiple-candidate method [2] the phone transcription was obtained by finding the top m-scoring pronunciations for each example. From the union of all pronunciations across all utterances for each word, the pronunciation $\hat{T}$ was found which maximised

$$\hat{T} = \arg\max_{T_i} \prod_{n=1}^{N} P(U_n|T_i)P(T_i) \tag{2}$$

where $U_n$ was the $n$th of $N$ utterances, and $\{T_i; i = 1, \ldots, Nm\}$ the $i$th transcription. $P(U_n)$ is assumed constant and $P(T_i)$ is given by a language model. This method has a simple extension to the N-best case, where the transcription is taken to be the top N-scoring transcriptions from the product term of equation (2).

Haeb-Umbach et al [5] have recently proposed a method in which an 'average' utterance is obtained from multiple utterances of the unknown word. The average utterance is then used as the input to the phone recogniser. This gave a lower word error rate than the multiple-candidate transcription method. A combination of the two methods achieved the lowest error rate. Several utterances are, however, required to produce the 'average' utterance.

## 3.3 Acoustic sub-word models

Since there is very little keyword data, the acoustic sub-word HMMs need to be well trained in advance. Further training may be used to adapt the models to the new word. Sub-word models used to date on the new word task fall into two classes, phonetic and acoustic segmentation. Phonetic models are constructed by partitioning the acoustic space into regions based on human linguistic knowledge. Alternatively, the acoustic space can be partitioned using automatic clustering techniques, such as Vector Quantisation (VQ).

Both model types have been used for open keyword set word-spotting. Wilcox and Bush [14] used sub-word HMMs based on general acoustic units. An arbitrary segment of the user's speech was used to learn the statistics for a pool of Gaussian distributions, using fuzzy k-means clustering. Each frame of the keyword utterance was quantised using these clusters. The KPS was built up by creating a HMM state for each frame where the cluster index was distinct from the previous frame. The output distribution of the state corresponded to the relevant Gaussian cluster. Once hypothesised, the means and transition probabilities within the KPS were adapted to the keyword by Baum-Welch re-estimation. This approach yields a very natural interaction between the user and device, but its performance is limited by the small number of keyword utterances available to adapt the relatively broad acoustic units.

In contrast, Kupiec et al [10] used standard phonetic HMMs in the keyword recogniser. Each model was a three state HMM with continuous density Gaussian output distributions, trained from hand-transcribed data. Since the KPS, and potential confusions there-of, was compared to an orthographic dictionary, the use of phonetically based models was essential.

Bahl et al [2, 3], compared the use of phonetic and fenonic-based Markov models for determining acoustic transcriptions of words for use in a speech recogniser. The phonetic system was standard, except that the HMMs had seven states, with context-independent HMMs pre-trained on a set of hand-transcribed data. The phone string hypothesised by the acoustic processor is taken to be the KPS. A fenone corresponds to a single frame of speech. In the fenone system, each speech sample vector is compared to a set of 200 VQ prototype vectors using Euclidean distance, and a VQ label string produced. Each label is then replaced by a Markov model, typically 2 states. Before the fenonic models can be used the transition and output probabilities have to be trained. Due to implicit tying between the VQ vectors, training of the models does not necessarily require each word to be seen.

On a number of recognition tasks, both isolated word and continuous speech, the fenonic system was found to give a lower word error rate in general. This is due to the fenones being able to implicitly model co-articulatory effects. When there were limited examples of the new words in the training data, however, the phonetic models achieved the lowest word error rate. This shows one of the disadvantages of fenones, in that in practice several examples of each word are required to train each model. Haeb-Umbach et al [5] have shown a similar result using phonetically based segments, namely that smaller sub-word units tend to provide better automatically derived transcriptions if there is a reasonable amount of transcription material.

Bahl's work used context-independent phonetic HMMs. Asadi et al [1] have shown that improved transcriptions can be achieved by using context-dependent models. These models incorporate some of the co-articulation effects that are lost in context-dependent models. Their performance should be closer to that of fenones.

As the number of keyword utterances is expected to be very small, the acoustic segment approaches described above are not well suited to the KPS determination task. For this work, therefore, pre-trained phonetic HMMs have been used. Context dependency is compared through using monophone and biphone HMMs. Each HMM is a standard three state continuous Gaussian density model (as shown in figure 2), trained using Baum-Welch re-estimation. This model class has the additional advantage that during development the KPSs can be compared to the corresponding phonetic dictionary transcription to assess the accuracy of the automatically transcribed KPS.

## 3.4 Grammar constraints

In the simplest case, no grammar constraints are applied to the phone models in the recogniser. Each phone can be followed by any other phone with equal probability.

Asadi et al [1] have shown that the error rate in a continuous speech recogniser can be lowered by applying a phone-level language model to constrain the recognition paths. They applied a statistical class grammar, where each phone was placed in a separate class. The grammar was computed from the phonetic transcriptions of 600 training sentences, which were representative of the recognition task domain (Resource Management) that they were using. Since the domain is unspecified in the open-keyword word-spotting task considered here, an alternative source of inter-phone statistics is required. Hence, in this work statistical language models were constructed from a 160,000 word pronunciation dictionary (BEEP-0.6 [13]). Using this dictionary has the advantage that it is entirely domain-free. However, since each word is weighted equally, the frequency of occurrence of a particular phone combinations is lost.

A much stronger constraint can be placed on the phone transitions if the phone loop in the recogniser is replaced by a word recogniser, using phonetic dictionary pronunciations for each word in the network, for example

```
$AFTERNOON = WD_BEGIN%AFTERNOON aa f t ax n uw n WD_END%AFTERNOON;
$BBC = WD_BEGIN%BBC b iy b iy s iy WD_END%BBC;
    ......
$ZOO = WD_BEGIN%ZOO z uw WD_END%ZOO;

( $AFTERNOON | $BBC | ... | $ZOO | NEW_WORD )
```

To handle out-of-vocabulary (OOV) words, a new word model is included in the network. When a keyword is requested that is not in the dictionary, it is (hopefully) mapped to the new word model [1], and the phone string determined using one of the techniques described. A tradeoff has to be made between dictionary size and the number of OOV words. Increasing the dictionary reduces the likelihood of OOV words being encountered, but at a cost of higher memory requirements. As previously mentioned, a high proportion of keywords are likely to be names or task-specific jargon which require a large dictionary to give adequate coverage, so this approach is not particularly suited to the open-keyword case. It would, however, be suitable for a combined fixed and open-keyword system.

# 4 Combined orthographic spelling and spoken keyword request

From the above, it is clear that orthographic spelling and spoken automatic transcription methods are complementary, the problems arising from the spelt input being handled by spoken input, and vice-versa. Combining the information present in the two knowledge sources should, therefore, lead to improved phonetic transcriptions. This has been observed in practice.

To make the combination tractable, the information gained from one input mode can be used to constrain the search space in the other input mode. Typically, a series of pronunciations are obtained from the spelt input. These are then used to form the recognition network, forcing the speech derived KPS to match one of the text pronunciations, i.e. the goal is to find the phonetic string $P$ to maximise

$$p(P|L,U) \approx \arg\max_P p(U|P)p(P|L) \tag{3}$$

where $U$ represents the spoken utterance(s) and $L$ the word spelling.

The orthographic spelling-based systems discussed in section 2 give a single pronunciation string, but a pronunciation network is required (to constrain the phonetic recognition process) in this case. A probabilistic transformation method is used to create a pronunciation network from the output of a text-to-speech mapper. The transformation is based on the confusion probabilities between phones in the mapper. For the information theoretic derived spelling-to-sound rules the pronunciation network is formed by taking the top $m$ pronunciations of the $n$th letter, where the score for the $n$th letter of a word is defined by

$$\triangle(l_1^n, r_1^n) = \log p(r_n | r_{n-5}^{n-1}, l_{n-5}^{n+5}) + \triangle(l_1^{n-1}, r_1^{n-1}) \tag{4}$$

This typically gives rise to a recognition network of around 130 pronunciations. In both cases, when a single utterance was used the combined spelling and spoken transcription systems were seen to approach the hand-transcribed speech recognition performance. Using DECtalk, for example, the total number of new words not recognised by the system dropped from 41% to 3% [1]. When multiple utterances are used, the performance was observed to match that of the hand-transcribed systems, for example this was shown for a 20K continuous speech recognition task by Bahl et al [2] given 4 utterances of the new word.

Alternatively, the spelt and spoken derived strings can be used in parallel in the word-spotter or speech recogniser. This has not been shown to lead to any clear improvement, probably due to the amount of variation it introduces increasing the probability of a false alarm [5].

# 5 Experimental results

The KPS from the speech word-spotting system has been evaluated on the VMR database. One and two utterance keyword inputs, various levels of N-best, the use of context-modelling, and language models, have been investigated.

## 5.1 System description

The HMMs used were speaker dependent, with continuous density multiple component Gaussian distributions. Each model had 3 emitting states, with a left-to-right topology, no skips, except for the silence and pause models, both of which were ergodic with 3 and 1 states respectively. The parameters used in the system were; 12 Mel-frequency Cepstral Coefficients, normalised log energy, and first and second derivatives of all parameters, giving a vector size of 39. The BEEP phone set was used, which consists of 46 monophones [13]. Monophone HMMs, with 8 mixture components per state, and right-context state-clustered biphone HMMs, with 3 mixture components, were used in the KPS recogniser. Approximately 350 states were used in the biphone HMMs.

The Video Mail Retrieval (VMR) DATABASE 1 [6] was used to evaluate the system. This database has been designed for word-spotting and information retrieval in spontaneous speech, based around a set of 35 pre-defined keywords. For each speaker, there are 5 examples of each keyword spoken in isolation, a set of $\sim$ 200 read training sentences, and 20 spontaneous speech messages on topics related to the keywords. The HMMs were trained on the read speech, and the isolated keyword examples used as input to the keyword phone string recogniser. Since the keywords are assumed unknown, no distinction is made in training between keyword and non-keyword speech.

The set of monophone HMMs were used in the word-spotter as both the background filler models and the sub-word units in the keyword models. Putative keyword hits were re-scored by dividing the maximum log likelihood keyword score by the average filler model score over the same time frames. This has been shown to yield a better keyword ranking than other proposed schemes [8].

Performance is assessed in terms of the percentage of false alarms per $i$th false alarm, i.e.

$$\%\text{ hits / }i\text{th FA} = \frac{\sum_{j=1}^{K}(H_{i,j}/K_j)}{K} \times 100 \tag{5}$$

where $K$ is the total number of true hits of all keywords, $K_j$ the number of true hits of the $j$th keyword, and $H_{ij}$ is the number of true hits scored for the $j$th keyword before the $i$th false alarm. The performance at the 1st and 3rd false alarm is reported, along with the average performance over the first 10 false alarms. Results are averaged over 4 keyword utterances, and 14 speakers.

If a keyword is known *a priori* then the KPS can be constructed from the phonetic dictionary definition of the keyword. These KPSs should be more robust than automatically derived KPSs, so their word-spotting performance provides a measure for the automatic systems to aim at. The dictionary (BEEP) defined KPS word-spotting performance is referred to as the Baseline performance in the results below.

All training and testing used version 1.5 of the HTK HMM toolkit [15], with suitable extensions to perform N-best decoding and word-spotting.

### 5.2 Results

| System | HMM class | N | No. of False Alarms 1 | 3 | ave |
|---|---|---|---|---|---|
| Baseline | - | - | 66.0 | 81.4 | 84.0 |
| Null Grammar | mono | 1 | 55.7 | 70.7 | 74.1 |
| | mono | 7 | 56.8 | 72.0 | 76.2 |
| | biph | 7 | 56.4 | 71.8 | 75.4 |

Table 1: Word-spotting performance given a single keyword utterance input to the unconstrained KPS recogniser

Table 1 shows the performance of the baseline, and unconstrained (Null Grammar) automatic keyword transcription system. The latter approach led to a relative drop in word-spotting performance of 11.8% for the Viterbi string, and 9.3% for the 7 N-best string, with monophone HMMs. Previous work [9] had shown that word-spotting performance improved with an increase in N in the KPS recogniser up to 7 N-best. For the Null Grammar case, it can be seen that, on average, a slight improvement in performance is gained by using multiple pronunciations in the KPS. This was found to be true for 10/14 speakers in the database. For the remaining speakers, increasing N led to a decrease in performance. However, the average decrease in performance was far less, 1.08% compared to 3.45%, than the average increase observed for the other speakers. This suggests that increasing N makes the system more robust. One disadvantage in increasing N is that the number of operations in the word-spotting pass increases, slowing down the retrieval. This can be partially overcome by using tree structuring so that identical phones are not repeated in the keyword network.

It is known that increasing the complexity of the acoustic models, by incorporating context, can improve performance. The 7 N-best test was repeated using right-context biphones. Table 2 shows that in this instance using biphones leads to a slight drop in performance. This is probably due to having insufficient data to train the biphone models well.

| System | N | No. of False Alarms 1 | 3 | ave |
|---|---|---|---|---|
| Baseline | - | 66.0 | 81.4 | 84.0 |
| Null Grammar | 7 | 56.8 | 72.0 | 76.2 |
| Bigram | 7 | 54.1 | 69.0 | 72.5 |
| Four-gram | 7 | 55.8 | 70.5 | 75.0 |

Table 2: Word-spotting performance for baseline system, and 7 N-best derived KPSs using a null grammar, bigram, and fourgram.

The effect of language modelling on word-spotting performance was investigated by using a phone level bigram and fourgram to constrain the 7 N-best recognition. Table 2 shows that increasing the context from 2 to 4 adjacent phones helped improve performance. However, this was still below that of the null-grammar. This is contrary to results seen for CSR, so is probably due to the poor usage weighting within the pronunciation dictionary.

The major cause of the drop in performance of the speech derived KPS system relative to the baseline, dictionary, system is poor transcriptions of $\sim$ 8% of the keywords. There are a number of causes for this. In particular, breath effects at the start and end of each keyword were found to add unwanted phones. Unclear speech and the use of a different pronunciation by the speaker to that used in the test data also affected the KPS. It is possible that there is a mis-match between the training and test data sets, due to the difference in speaking styles between isolated, read and spontaneous speech. However, the database is too small to investigate this.

| Initial N | Final N | No. of False Alarms 1 | 3 | ave |
|---|---|---|---|---|
| 20 | 1 | 58.5 | 74.1 | 77.0 |
| 20 | 7 | 58.8 | 74.6 | 78.9 |
| 1 | 1 | 58.0 | 73.9 | 76.9 |

Table 3: Word-spotting performance for KPS system based on 2 utterances of the keyword, null grammar, monophone N-best recogniser: 20 N-best reduced to best 1 and 7 transcriptions, and best of 1 N-best

The effect on word-spotting performance when the number of keyword examples is increased was tested for the case of two keyword utterances. Two utterances were chosen as it was felt to be reasonable for an input request to include a repetition of the keyword. From table 3, it can be seen that using 7 N-best transcriptions in the KPS again improves performance over the 1 N-best case. This was true for 11/14 speakers, with an average increase of 2.65% compared to a 1.0% average decrease for the other speakers. Relative to the baseline, the word-spotting performance dropped by 8.3% and 6.1% for the 1 and 7 N-best cases, given 20 initial transcriptions per utterance. In both cases, this is better than the best single utterance case.

To investigate if multiple initial transcriptions were needed, the test was repeated using one transcription per utterance. As shown in table 3, the performance was very similar to the 20 N-best reduced to 1 N-best case, with a 8.5% drop relative to the baseline. This implies that N-best computation is not necessary.

# 6 Conclusions

This report has described methods for automatically determining the phonetic transcription of a keyword (the KPS) for use in an open-keyword set HMM-based word-spotting system. A review of the literature was provided for orthographically spelt, spoken, and combined spelt and spoken, keyword input modes.

In spelling-based systems, the new word transcription is obtained from a phonetic dictionary look-up or from a text-to-phone mapper. These systems have difficulty modelling real names and task-specific jargon, both of which are expected to feature highly in any retrieval task. Hence, a spelling-based system is insufficient on its own to determine the KPS for word-spotting. The phonetic transcription from a spoken input is, typically, acquired by passing the example of the new word through a phone level speech recogniser, and taking the hypothesised output string as the KPS. Unlike the spelt input case, this approach can handle any input word. It also provides a more natural interface to the word-spotting tool, particularly for hand-held devices in which any spelling would have to be orally delivered. However, due to the inherent variation in the speech signal, the input strings are not necessarily consistent or representative of the keyword pronunciation used in the audio document database. This leads to a drop in performance compared to a system based on phonetic dictionary keyword transcriptions. As spoken input is imperative for hand-held devices, it was considered in more detail, summarised below. If both input modes are available then the problems of coverage and inconsistency can be overcome to a certain extent by combining the two knowledge sources to give the KPS. The resulting strings are far more robust, and previous work has shown that recognition performance equal to the level achieved using hand-transcribed strings is possible.

A number of methods for obtaining the KPS from spoken examples were described. The retrieval request is initially passed through an endpoint detector to locate the speech frames corresponding to the keyword. A speech recogniser was chosen to perform this task, as it is relatively robust and applicable to both isolated and embedded keyword requests. Once the speech frames have been found, the spoken utterance(s) were passed through a phone level speech recogniser, with a continuous phone loop network. A multiple pronunciation approach, in which a N-best recogniser is used to hypothesise multiple paths, each of which is combined in the KPS, was investigated. This method has the advantage that it is applicable to any word, and has a relatively low memory requirement as no pronunciation dictionary is used. The latter is an important consideration for hand-held devices, due to their small size. When the KPS had been determined, it was output to the keyword-spotter and the keyword-spotting recognition pass performed.

Word-spotting tests were performed on the VMR database to evaluate the effect of: the number of strings, N; the complexity of the HMMs; language modelling; and the number of sample keyword utterances. As a baseline, the performance of a phonetic dictionary defined KPS was taken. Overall, it was found that the speech derived KPSs were less robust compared to the phonetic dictionary system. Given a single utterance, the best approach was found to be to use a 7 N-best, null grammar, monophone HMM-based, phone recogniser to determine the KPS. This showed a 9.3% relative drop in performance relative to the baseline. If two utterances are available, then it was found that the problem of poor KPSs, due to bad pronunciation or breath effects, was partially overcome. Again, the 7 N-best case yielded the best performance (6.1% relative drop compared to the baseline). However, using simply the two Viterbi strings for each utterance gave almost as good performance (8.5% relative drop).

# 7 Acknowledgements

This work was funded by Hewlett Packard Laboratories, Bristol. The N-best recognition was performed using modified programs, originally written by C.J.Leggetter (null-grammar) and V.Valtchev (language model).

# References

[1] Asadi, A., Schwartz, R., and Makhoul, J. *Automatic modelling for adding new words to a large-vocabulary continuous speech recognition system.* Proc ICASSP'91, pp 305-308, Toronto, 1991.

[2] Bahl, L.R., Das, S., de Souza, P.V., Epstein, R.L., Mercer, R.L., Merialdo, B., Nahamoo, D., Picheny, M.A., and Powell, J. *Automatic Phonetic Baseform Determination.* Proc ICASSP'91, Toronto, 1991.

[3] Bahl, L.R., Brown, P.F., de Souza, P.V., Mercer, R.L., and Picheny, M.A. *A method for construction of acoustic markov models for words.* IEEE Trans Speech and Audio Processing, 1(4), pp 443-452, Oct, 1993.

[4] Deller, J.R., Proakis, J.G., and Hansen, J.H.L. *Discrete-time processing of speech signals*, Macmillan Publishing Co., NY, 1993.

[5] Haeb-Umbach, R., Beyerlein, P., and Thelen, E. *Automatic transcription of unknown words in a speech recognition system.* Proc ICASSP'95, pp 840-843, Detroit, 1995.

[6] Jones, G. J. F., Foote, J. T., Sparck Jones, K., and Young, S. J. *Video Mail Retrieval Using Voice: Report on Keyword Definition and Data Collection (Deliverable Report on VMR Task No 1)*, University of Cambridge Computer Laboratory, Tech. Report No. 335, May, 1994.

[7] Klatt, D.H. *Review of text-to-speech conversion for English*, Journal of the Acoustical Society of America, vol. 82, pp 737-793, Sept. 1987.

[8] Knill, K. M. and Young, S.J. *Speaker Dependent Keyword Spotting for Accessing Stored Speech*, Cambridge University Engineering Dept., Tech. Report No. CUED/F-INFENG/TR 193, 1994. Available by anonymous ftp from svr-ftp.eng.cam.ac.uk.

[9] Knill, K. M. and Young, S.J. *Keyword training using a single spoken example for applications in audio document retrieval.* Proc ICSST, Perth, Australia, 1994.

[10] Kupiec, J., Kimber, D., and Balabsubramanian, V. *Speech-Based Retrieval Using Semantic Co-Occurence Filtering.* Proc ARPA Human Language Technology Workshop, 1994.

[11] Lucassen, J. M. and Mercer, R. L. *An Information Theoretic Approach to the Automatic Determination of Phonetic Baseforms*, Proc. ICASSP'84, San Diego, March 1984, paper no. 42.5.

[12] Rabiner, L.R., and Sambur. M.R. *An algorithm for determining the endpoints of isolated utterances*, Bell System Technical Journal, vol. 54, pp 297-315, Feb. 1975.

[13] Robinson, T., Fransen, J., Pye, D., Foote, J., and Renals, S. *WSJCAM0: A British English speech corpus for large vocabulary continuous speech recognition* Proc ICASSP'95, Detroit, 1995.

[14] Wilcox, L.D. and Bush, M.A. *HMM-Based Wordspotting for Voice Editing and Indexing.* Proc Eurospeech, Vol. 1, pp 25-28, Genoa, Sept, 1991

[15] Young, S. J. Woodland, P. C., and Byrne, W. J. *HTK: Hidden Markov Model Toolkit V1.5.* Entropic Research Laboratories Inc., 1993.