# A Phonetic Search Approach to the 2006 NIST Spoken Term Detection Evaluation

*Roy Wallace, Robbie Vogt and Sridha Sridharan*

Speech and Audio Research Laboratory,
Queensland University of Technology (QUT), Brisbane, Australia

royw@ieee.org, {r.vogt,s.sridharan}@qut.edu.au

## Abstract

This paper details the submission from the Speech and Audio Research Lab of Queensland University of Technology (QUT) to the inaugural 2006 NIST Spoken Term Detection Evaluation. The task involved accurately locating the occurrences of a specified list of English terms in a given corpus of broadcast news and conversational telephone speech. The QUT system uses phonetic decoding and Dynamic Match Lattice Spotting to rapidly locate search terms, combined with a neural network-based verification stage. The use of phonetic search means the system is open vocabulary and performs usefully (Actual Term-Weighted Value of 0.23) whilst avoiding the cost of a large vocabulary speech recognition engine.

**Index Terms**: spoken term detection, phonetic search, keyword spotting

## 1. Introduction

Providing intelligent access to large corpora of spoken audio is one of speech technology's most important challenges. There is direct demand from a diverse range of areas such as security and defense, media monitoring, personal entertainment and as a component of other research areas such as spoken document retrieval and understanding.

For these reasons, in 2006 the National Institute of Standards and Technology (NIST) established a new initiative called Spoken Term Detection (STD) Evaluation, to encourage research and development of technology to detect short word sequences rapidly and accurately in large heterogeneous audio archives [1].

The STD task (also known as keyword spotting) involves the detection of all occurrences of a specified search term, which may be a single word or multiple word sequence. A score is usually output accompanying each putative match as a measure of how confident the system is that it is a true occurrence. This enables the adjustment of the system's operating point to trade-off between errors due to missing true occurrences and errors due to detecting spurious false alarms.

One popular approach [2] is to use a large vocabulary continuous speech recognition (LVCSR) engine to generate a word-level transcription or lattice, which is then indexed in a searchable form. This approach has resulted in good performance, provided a suitable LVCSR engine with low word error rate is available. However, these systems are critically restricted in that the terms that are able to be located are limited to the recogniser vocabulary used at decoding time, meaning that occurrences of out-of-vocabulary (OOV) terms cannot be detected. This is especially important since search terms are typically rare words and practical search engines have been found to experience OOV rates of up to 12–15% [3]. Also, the run-time requirements of LVCSR systems have been suggested to be prohibitive for some large-scale applications [4].

An alternative method which supports an open-vocabulary is phonetic search [5, 6]. This approach does not use LVCSR, but instead performs decoding and indexing at the level of phones. Searching requires the translation of the term into a phonetic sequence, which is then used to detect exact or close matching phonetic sequences in the index. The decoding speed of phonetic systems is much faster than LVCSR, however, phonetic search is not as straightforward and therefore typically slower than the simple word look-up used with LVCSR systems. Additionally, it is prone to high levels of false alarms, especially for short terms [7]. Phonetic systems do, however, show promise for other languages with limited training resources [8], for which phonetic decoding may be the only feasible option in the absence of a thoroughly trained LVCSR engine.

The apparent complementary strengths and weaknesses of the two above approaches lead to the suggestion of fusion. This has been consistently shown to improve performance and also allows for open-vocabulary search [9, 10]. However, this approach does not avoid the costly training, development and run-time requirements associated with LVCSR engines, that is, assuming such resources even exist for the language in question.

QUT chose a phonetic approach to provide fast, open-vocabulary search without the need for an LVCSR engine. The system developed in QUT's Speech and Audio Research Laboratory uses Dynamic Match Lattice Spotting to detect occurrences of phonetic sequences which closely match the target term. The results of the approach used in the 2006 NIST STD evaluation are presented. Section 2 provides an overview of the system. In section 3, the evaluation measures used are explained, followed by results and discussion of performance in section 4.

## 2. Spoken term detection system

The system consists of two distinct stages; indexing and search (Fig. 1). This approach is used to allow as much processing as possible to be performed offline, and thereby enable rapid queries at search time. During indexing, phonetic decoding is used to generate lattices, which are then compiled into a searchable database. At search time, a dynamic matching procedure is used to locate phonetic sequences which closely match the target sequence. The system is based on the Dynamic Match Lattice Spotting technique described in [6]. A brief description of the system components follow.

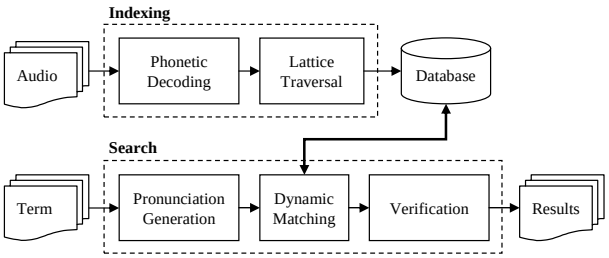


Figure 1: *System architecture*

### 2.1. Indexing

First, Perceptual Linear Prediction feature extraction is performed, followed by a segmentation stage. The speech is then decoded using a Viterbi phone recogniser to generate a recognition phone lattice. Tri-phone HMM's and a bi-gram phone language model are used during decoding, and a 4-gram phone language model is used during rescoring.

The resulting phone lattices could be searched directly, however, this would require a computationally intensive lattice traversal for each new search term, which would severely impact search speed. Instead, a significant portion of the traversal is performed offline during indexing. A modified Viterbi traversal is used to emit all phone sequences of a fixed length, $N$, which terminate at each node in the lattice. A value of $N = 11$ was found to provide a suitable trade-off between index size and search efficiency.

The resulting collection of phone sequences is then compiled into a sequence database. A mapping from phones to their corresponding phonetic classes (vowels, nasals, etc.) is used to generate a hyper-sequence database, which is a constrained domain representation of the sequence database. The resulting two-tier, hierarchical database structure is used at search time to significantly reduce the search space and allow for rapid search.

### 2.2. Search

When a search term is presented to the system, the term is first translated into its phonetic representation using a pronunciation dictionary (for multi-word terms, the pronunciation of each word is concatenated). If any of the words in the term are not found in the dictionary, letter-to-sound rules are used to estimate the corresponding phonetic pronunciations.

Given the resulting target phone sequence, and the collection of indexed phone sequences stored in the database, the task is to compare the target and indexed sequences and emit putative occurrences where a match or near-match is detected. To allow for phone recognition errors, the Minimum Edit Distance (MED) is used to measure inter-sequence distance by calculating the minimum cost of transforming an indexed sequence to the target sequence.

A simplified version of the MED calculation is used which only allows for phone substitutions. Previous experiments have shown that this greatly increases search speed with a minimal effect on performance. Variable substitution costs, $C_s(x, y)$, are used which depend on $x$, the indexed phone and $y$, the target phone. This allows for the incorporation of prior knowledge about the typical misrecognitions made by the speech recognition engine. In particular, the cost of a substitution is inversely related to the likelihood of it occurring. The substitution costs are defined as

$$C_s\left(x,y\right) = \begin{cases} I\left(R_y|E_x\right) & x \neq y \\ 0 & x = y \end{cases}. \tag{1}$$

Here $I\left(R_y|E_x\right)$ represents the information associated with the event that $y$ was actually uttered given that $x$ was recognised;

$$I\left(R_y|E_x\right) = -\log\left(p\left(R_y|E_x\right)\right). \tag{2}$$

Using Bayes Theorem, these statistics are easily computed from the recognition likelihoods, $p\left(E_x|R_y\right)$, phone prior probabilities, $p\left(R_y\right)$, and emission probabilities, $p\left(E_x\right)$, estimated from a recognition confusion matrix generated during development.

The MED score associated with transforming the indexed sequence $X = \left(x_i\right)_{i=1}^{N}$ to the target sequence $Y = \left(y_i\right)_{i=1}^{N}$, is defined as the sum of the cost of each necessary substitution;

$$\text{MED}\left(X,Y\right) = \sum_{i=1}^{N} C_s\left(x_i, y_i\right). \tag{3}$$

For each indexed phone sequence, $X$, associated with a MED score below a specified threshold, let $\mathcal{P}_X$ represent the set of individual occurrences of $X$, as stored in the index. For each $P \in \mathcal{P}_X$, the score for the occurrence is formed by linearly fusing the MED score with an estimated acoustic log likelihood ratio score, ALLR $(P)$, as follows;

$$\text{Score}\left(P,Y\right) = \text{MED}\left(X,Y\right) - \alpha \cdot \text{ALLR}\left(P\right), \tag{4}$$

where $\alpha$ is tuned empirically. Occurrences with more negative scores represent more confident matches. The incorporation of ALLR $(P)$ allows for differentiation between occurrences with equal MED scores, and promotes occurrences with higher acoustic probability.

Because the index database contains sequences of a fixed length, when searching for a term longer than $N = 11$ phones, the term must first be split (at syllable boundaries) into several smaller, overlapping sub-sequences. Each of these subsequences may then be searched for individually. The subsequence occurrences are then merged by emitting a putative occurrence of the target term where each of the term's subsequences are found overlapping in correct order. The score for each complete occurrence is approximated by a linear combination of scores from the sub-sequence occurrences.

Because the MED score is not directly comparable between terms of different phone lengths, a final verification stage is required. Longer terms have a higher expected MED score as there are more phones which may have been potentially misrecognised. Using a neural network (single hidden layer, four hidden nodes), Score $(P, Y)$ is fused with the number of phones, Phones $(Y)$, and number of vowels, Vowels $(Y)$, in the term, to produce a final detection confidence score for each putative term occurrence.

## 3. Evaluation procedure

### 3.1. Performance metrics

The Receiver Operating Characteristic (ROC) plot is commonly used to describe STD performance by plotting the detection rate against the number of false alarms per keyword per hour (fa/kw-hr). To reduce this representation to a single value, the Figure

of Merit is often used, which is equivalent to the average value of the ROC curve over the range 0 to 10 fa/kw-hr.

The NIST evaluation instead used a cost/value application model to derive the Term-Weighted Value (TWV), defined at an operating point given by the confidence score threshold, $\theta$;

$$\begin{aligned} \text{TWV}\,(\theta) =& 1 - \underset{term}{average}\,\{S_{term}\,(\theta)\} \\ S_{term}\,(\theta) =& P_{miss}\,(term,\theta) + \beta \cdot P_{FA}\,(term,\theta)\,, \end{aligned} \tag{5}$$

where $\beta \approx 10^3$, $P_{miss}\,(term,\theta) = 1 - \frac{N_{correct}(term,\theta)}{N_{true}(term)}$, and $P_{FA}\,(term,\theta) = \frac{N_{spurious}(term,\theta)}{N_{NT}(term)}$. In STD, the number of non-target trials, $N_{NT}\,(term)$, must be defined in order to calculate false alarm probability. In this case, the number of non-target trials was defined to be proportional to the number of seconds of speech in the test data, $T_{speech}$;

$$N_{NT}\,(term) = T_{speech} - N_{true}\,(term)\,, \tag{6}$$

where $N_{true}\,(term)$ was the number of true occurrences of $term$. Selecting an operating point using the confidence score threshold, $\theta$, allowed for the creation of Detection Error Trade-off (DET) plots and calculation of a Maximum TWV. In addition, a binary "Yes/No" decision output by the system for each putative occurrence was used to calculate Actual TWV.

Possible TWV values included 1 for a perfect system, a value of 0 for no output, and negative values for systems which output many false alarms. Further details may be found in [11]. The TWV metric and DET plots require the calculation of miss rate, and therefore any terms with no occurrences in the reference (i.e. $N_{true} = 0$) were excluded from the evaluation.

### 3.2. Evaluation data

The English evaluation data consisted of around 3 hours of American English broadcast news (BNews), 3 hours of conversational telephone speech (CTS) and 2 hours of conference room meetings. The results of the conference room meeting data will not be discussed, as training resources were not available for that particular domain.

The evaluation term list included 898 terms for the BNews data, and 411 for the CTS data. Each term consisted of between one and four words, with a varying number of syllables (Fig. 2). Although the actual term list was not released to participants, for the BNews data, there were, on average, 2.5 occurrences per term per hour in the evaluation reference, and for the CTS data there were 4.8 occurrences per term per hour.

## 4. Results and discussion

### 4.1. Training data

Two sets of tied-state 16 mixture tri-phone HMM's (one for BNews and one for CTS) were trained for speech recognition using the DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus and CSR-II (WSJ1) corpus, then adapted using 1997 English Broadcast News (HUB4) for BNews and Switchboard-1 Release 2 for the CTS models. Phone bi-gram and 4-gram language models, and phonetic confusion statistics were trained using the same data. Overall, around 120 hours of speech were used for the BNews models, and around 160 for the CTS models.

Letter-to-sound rules were generated from The Carnegie Mellon University Pronouncing Dictionary (CMUDICT 0.4).

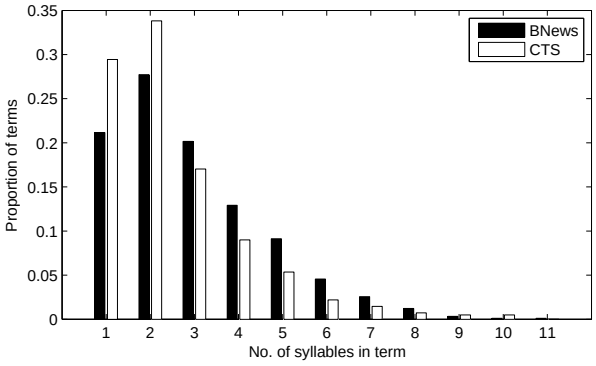


Figure 2: *Histogram of search term length in syllables*

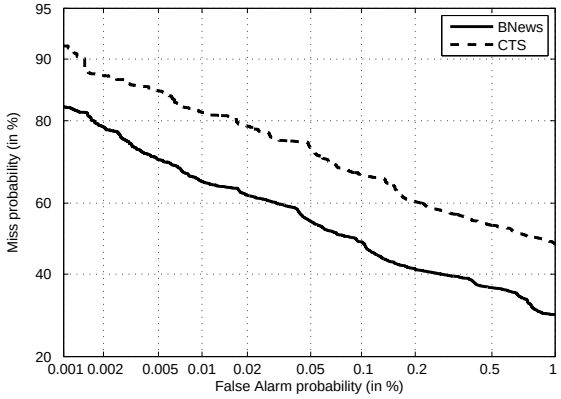


Figure 3: *DET Plot for all terms in BNews and CTS audio*

Tuning of system parameters was performed on separate, held out data from the same sources as the evaluation. Neural network training examples were generated by searching for 1965 development terms and using the resulting $(\text{Score}\,(P,Y)\,, \text{Phones}\,(Y)\,, \text{Vowels}\,(Y)\,, y\,(P,Y))$ tuples, where $y$ represented the class label, which was set to 1 for true occurrences and 0 for false alarms.

### 4.2. Overall results

The overall performance of the system for BNews and CTS data is shown in Fig. 3. Table 1 lists the Actual and Maximum TWV for each source type, along with the 1-best phone error rate (PER) of the phonetic decoder on development data similar to that used in the evaluation. Clearly, better performance is achieved in the BNews domain, probably due to the improved phonetic decoding due to the higher quality and clearer speech. The Actual TWV is reasonably close to the Maximum TWV, demonstrating that the operating point determined during development generalised well to unseen terms and audio.

| Source Type | BNews | CTS |
|---|---|---|
| 1-best PER | 24% | 45% |
| Actual TWV | 0.2265 | 0.0873 |
| Maximum TWV | 0.2459 | 0.1044 |

Table 1: *Term Weighted Value achieved for each source type*

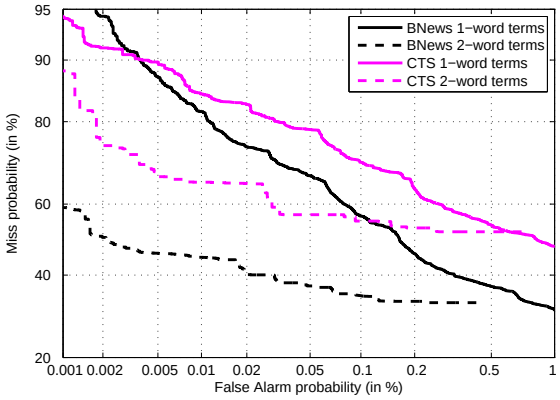


Figure 4: *Performance for single word and 2-word terms*

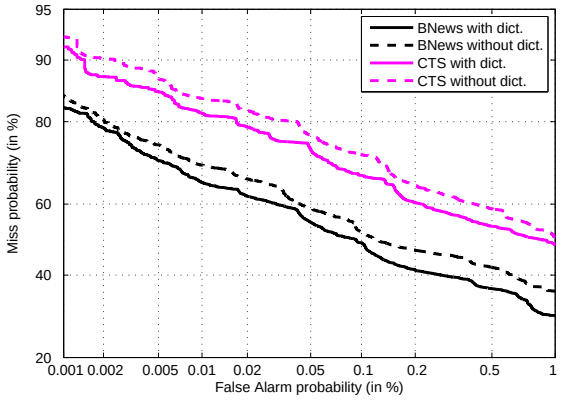


Figure 5: *Comparison of systems with & without pronunciation dictionary*

### 4.3. Effect of term length

One of the difficulties commonly associated with phonetic search is the large number of false alarms generated when searching for short terms. Short phonetic sequences are difficult to detect, as they are more likely to occur as parts of other words and detections must be made based on limited information. The findings displayed in Fig. 4 confirm this, and show much better performance for two-word terms compared to single word terms. The longer terms do however have a higher minimum miss rate, presumably because the likelihood of finding a complete close-matching sequence is lower for long terms.

### 4.4. Processing efficiency

Although the system described has not been optimised for indexing efficiency, some measurements are given in Table 2 to provide a context for the performance results. In addition to the overall search speed, the speed of search excluding the unoptimised merging stage is provided for comparison.

| | |
|---|---|
| Index size | 558 MB/speech hr |
| Indexing time | 18 processing hrs/speech hr |
| Search speed | 5 speech hrs/CPU-sec |
| Search speed (excluding merging) | 8 speech hrs/CPU-sec |

Table 2: *Processing efficiency measurements*

### 4.5. Use of letter-to-sound rules

The system described uses very little word-level information to perform decoding, indexing and search. In fact, in the absence of the pronunciation dictionary, no word-level information is directly used at all. Fig. 5 demonstrates such a system. It can be seen that the degradation in performance is not critical, and therefore demonstrates that without any word-level information, useful performance can still be obtained.

## 5. Conclusions

A phonetic search system has been presented which can successfully detect occurrences of search terms of various lengths, with search speeds in excess of 10000 $\times$ real-time. The system presented demonstrates that phonetic search can lead to useful spoken term detection performance. However, further performance improvements to verification and confidence scoring, particularly for short search terms, are required to compete with systems which incorporate an LVCSR engine.

The system allows for completely open-vocabulary search, avoiding the critical out-of-vocabulary problem associated with word-level approaches. The feasibility of using phonetic search for languages with limited training data, or for large-scale data mining applications, are also promising areas of further research.

## 6. References

[1] National Institute of Standards and Technology, "Spoken Term Detection evaluation web site," December 2006. [Online]. Available: http://www.nist.gov/speech/tests/std/

[2] M. Weintraub, "Keyword-spotting using SRI's DECIPHER large-vocabulary speech-recognition system," in *ICASSP '93*, vol. 2, 1993, pp. 463–466 vol.2.

[3] J.-M. Van Thong, P. Moreno, B. Logan, B. Fidler, K. Maffey, and M. Moores, "Speechbot: an experimental speech-based search engine for multimedia content on the web," *IEEE Trans. Multimedia*, vol. 4, no. 1, pp. 88–96, 2002.

[4] J. Picone, "Information retrieval from voice: The importance of flexibility and efficiency," in *NIST Spoken Term Detection Evaluation Workshop*, Gaithersburg, Maryland, USA, December 2006.

[5] S. Dharanipragada and S. Roukos, "A multistage algorithm for spotting new words in speech," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 8, pp. 542–550, 2002.

[6] K. Thambiratnam and S. Sridharan, "Rapid yet accurate speech indexing using dynamic match lattice spotting," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 1, pp. 346–357, 2007.

[7] M. Clements, P. Cardillo, and M. Miller, "Phonetic searching vs. LVCSR: How to find what you really want in audio archives," in *20th Annual AVIOS Conference*, 2001.

[8] A. J. K. Thambiratnam, "Acoustic keyword spotting in speech with applications to data mining," Ph.D. dissertation, Queensland University of Technology, 2005.

[9] P. Yu and F. Seide, "A hybrid word / phoneme-based approach for improved vocabulary-independent search in spontaneous speech," *Proc. ICLSP'04*, 2004.

[10] A. Amir, A. Efrat, and S. Srinivasan, "Advances in phonetic word spotting," in *CIKM '01*. New York, NY, USA: ACM Press, 2001, pp. 580–582.

[11] National Institute of Standards and Technology, "The Spoken Term Detection (STD) 2006 evaluation plan," September 2006. [Online]. Available: http://www.nist.gov/speech/tests/std/docs/std06-evalplan-v10.pdf