Modeling of Output Probability Distributions to Improve Small Vocabulary Speech Recognition in Adverse Environments

D. Thambiratnam* and S. Sridharan #
Speech Research Laboratory
School of Electrical and Electronic Systems Engineering
Queensland University of Technology
GPO Box 2434, Brisbane, Australia 4001

*dp.thambiratnam@qut.edu.au #s.sridharan@qut.edu.au

ABSTRACT : *This paper presents a solution to the adverse environment, open microphone problem, by using the information stored in HMM output probability distributions to obtain a confidence measure of the results. This information can also be used to perform a secondary classification and improve recognition results. The system was tested on data from the TI46 database that had been corrupted by noise from the NOISEX-92 database, as well as on real-world data, and shows promising results.*

1 Introduction

The performance of modern small vocabulary ASR systems is exceptional, particularly when testing conditions approximate training conditions. However there are still real-life problems that are yet to be resolved in a practical way. Among these problems are adaptation to unmatched acoustical conditions (the classic environmental noise problem), and the detection of out of vocabulary (OOV) words. [1]

In many situations, particularly where there is an open microphone, the ASR system must deal with input that isn't part of the vocabulary (OOV input). A common solution is to use some method to determine a confidence measure, thus providing a measure of accuracy of the recognised utterance. [2]

The advantages of confidence measures are twofold: they can be used to verify the speech recogniser's output for valid input (verification), and they are useful in filtering out OOV input (validation). Most of the existing techniques for hypotheses testing have three major drawbacks [3]:

- They are based on the use of garbage models and alternative recognition networks that are difficult to design and train
- Although techniques have shown to provide improvements in overall system performance, their computational cost is high
- Only acoustic data is used in the verification process, even though for a wide range of systems, valuable information from application-dependent knowledge is available and should be utilized.

This paper presents a novel solution that addresses the first two problems by using application dependant information. Using the information stored in output probability distributions (OPDs), it is possible to obtain a confidence measure. This doesn't require any extra garbage modeling, and has a very low computational cost.

2 Output Probability Distributions

The term Output Probability Distribution refers to the distribution of log probabilities from a set of Hidden Markov Models (HMMs). A given utterance is passed to each HMM in the set and the log probabilities from each model are concatenated to form a feature set, that is:

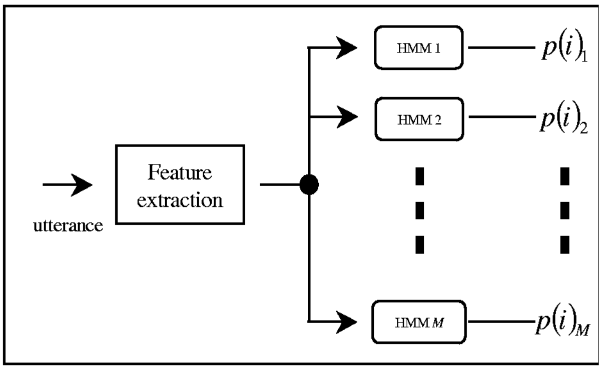


Figure 1: Obtaining an OPD

$$OPD(i) = [p(i)_1\, p(i)_2\, p(i)_3 \ldots\ldots p(i)_M]$$

where $p(i)_j$ indicates the log probability of utterance i from model $j, j = 1..M$ (M is the number of words in the system). Figure 1 illustrates this procedure.

Given a constant environment, each token is recognised in a consistent fashion. For example, in a simple digit-based ASR system, if the input word is 'eleven', the word 'seven' might be a good competitor (that is, its' model will give a score comparable to the score of the model for 'eleven'), and hence will consistently have high scores in the OPD. Likewise, the digit 'six' might consistently have a low score. This trend will be reflected in the OPD for the word 'eleven'. As a result, given a adequately constant environment, we can expect that the OPD is

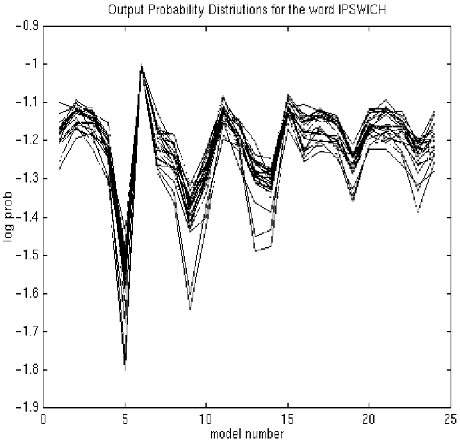


Figure 2: OPD for word 'IPSWICH'

constant. Figure 2 shows the normalised OPDs (scaled so that results have a maximum value of 1) for the word ‘IPSWICH’ over 400 utterance examples and 24 models. As expected, the OPD is approximately constant.

As the acoustic noise levels increase, the distributions change. Often strong competitors now become strong enough to override the correct model and the normalised OPD curve becomes flatter, as all of the models start to perform equally badly. Figure 3 shows the OPD for the word ‘0’ over different SNRs, using 400 utterance examples and 10 models. As the SNR decreases, the OPD becomes flatter, and some of the HMM models start to override the correct model. As the noise level increases, the output probabilities seem to cluster around the -1 value.

These distributions suggest that if a secondary classifier were to be trained on the OPDs it might be possible to obtain a confidence measure for a given background noise level. This information is directly related to the probability of the system output being correct.

3 OPD Models

A set of HMMs is trained for each token and the OPD feature vectors are produced using the training. These are used to train OPD models.

OPD models represent the OPDs for different tokens in a given environment and are used to perform the secondary classification of the OPD scores. This is further discussed in Section 3.2.

3.1 OPD Parameterisation

To adequately represent the OPDs, it is necessary to obtain a good feature set. It is obvious that the use of the OPD itself would provide a good representation. However, we found that other features could improve the performance significantly. These features are obtained from an OPD template, which is discussed below.

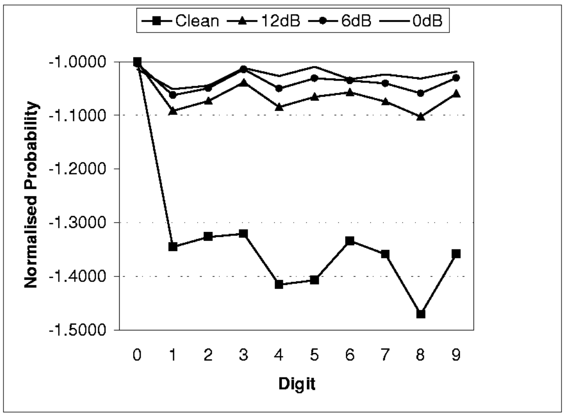


Figure 3: OPD Distributions for varying SNRs

OPD templates are produced for each HMM model. The template T_m is obtained by the following:

$$T_m = \frac{1}{N}\sum_{i=1}^{N} OPD(i)$$

where T_m indicates the template model for word m, N is the number of training utterances for a word that is associated with model m and $OPD(i)$ is the output probability distribution for utterance i.

The feature vector $\vec{X}$ is now obtained for each test utterance. To obtain $\vec{X}$ we first define the function $D_i(x)$:

$$A_i(x) = sort_{ascending}(OPD(i))$$

$$D_i(x) = A_i(x+1) - A_i(x)$$

$D_i(x)$ represents the difference between successive peaks in the OPD. We also define the term $D_t(x)$, which is obtained similarly to $D_i(x)$ except that T_m is used instead of $OPD(i)$.

Our results indicated that the value of $D_i(x)$ was approximately constant for each digit. As such, this vector has use as part of the feature vector set described below.

Feature vectors were selected by the following criteria:

1. They must be able to represent the OPD accurately.
2. They must be invariant to different noise levels.

Based on these criteria, the following terms were used in the feature vector:

Feature	Representation
$\lvert\mu_t - \mu_i\rvert$	the difference in means between T_m and $OPD(i)$. This represents the deviation between the two vectors.
$\frac{\sigma_t}{\sigma_i}$	The ratio of σ_t, the standard deviation of T_m, and σ_i, the standard deviation of $OPD(i)$. This represents the difference in spread between the two vectors.
$D_i(0)$	The difference between the highest and second highest peaks in $OPD(i)$. This is an indicator of how significant the ‘winning’ result is.
$\lvert\Delta\mu_t - \Delta\mu_i\rvert$	The difference between $\Delta\mu_t$, the mean of $D_t(x)$, and $\Delta\mu_i$, the mean of $D_i(x)$. This term represents the deviation between the two vectors $D_t(x)$ and $D_i(x)$.
$\frac{\Delta\sigma_t}{\Delta\sigma_i}$	The ratio of $\Delta\sigma_t$, the standard deviation of T_m, and $\Delta\sigma_i$, the standard deviation of $OPD(i)$. This represents the difference in spread between the two vectors $D_t(x)$ and $D_i(x)$.
$OPD(i)$	The values of the OPD

3.2 OPD Model Training

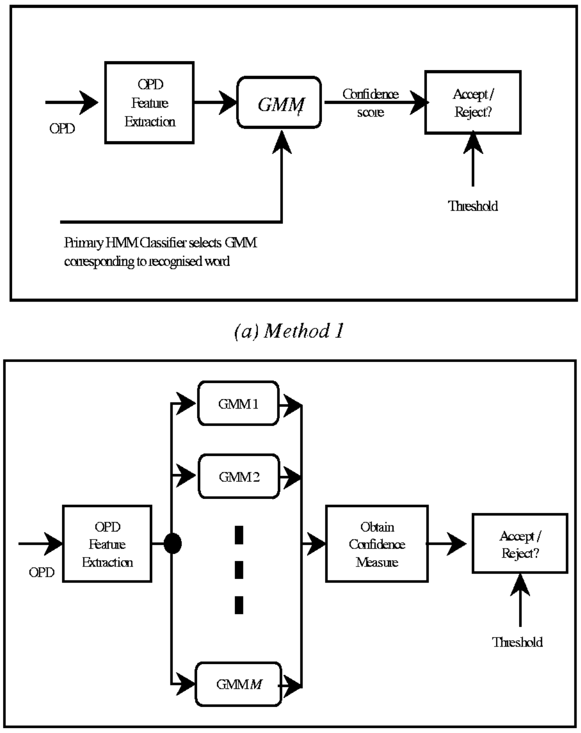
We chose to use Gaussian Mixture Models (GMMs) to model the OPD feature vectors. GMMs have the advantage of being able to represent any given distribution very accurately as they consist of several mixtures, each of which has a Gaussian distribution (hence the name Gaussian Mixture Model).

For each word in the vocabulary, one secondary classifier must be trained using the OPD feature set. We define the feature vector as

$$\vec{X}(i)=\begin{bmatrix} D_i(0),\ |\mu_t-\mu_i|,\ \dfrac{\sigma_t}{\sigma_i},\ |\Delta\mu_t-\Delta\mu_i|, \\ \dfrac{\Delta\sigma_t}{\Delta\sigma_i}, OPD(i) \end{bmatrix}$$

for $i=1..N$ where N is the number of training utterances available for each word in the vocabulary of the recogniser. The models are trained using standard GMM training procedures. [10]

To optimise the performance of the system, it is beneficial to train the system with both clean speech and noisy speech with varying degrees of SNR.


OPD
OPD Feature Extraction
GMM
Confidence score
Accept / Reject?
Threshold
Primary HMM Classifier selects GMM corresponding to recognised word
GMM1
GMM2
GMM M
Obtain Confidence Measure

(a) Method 1

(b) Method 2

Figure 4: Different methods of obtaining a confidence score

4 Application of OPD models

Since the performance of OPD models depends on the distribution, it is obvious that as the number of words in the vocabulary increases, it becomes harder to differentiate between different OPDs. As such, OPD models are best used for applications where the number of words in the vocabulary is small, for example, in voice dialling applications or industrial applications where voice control of a machine is implemented with a small vocabulary and the environment is very noisy.

OPD models can be used to provide a confidence measure for a ASR system. This confidence measure can be used to give a degree of accuracy of the system, and also is very useful in filtering out OOV input.

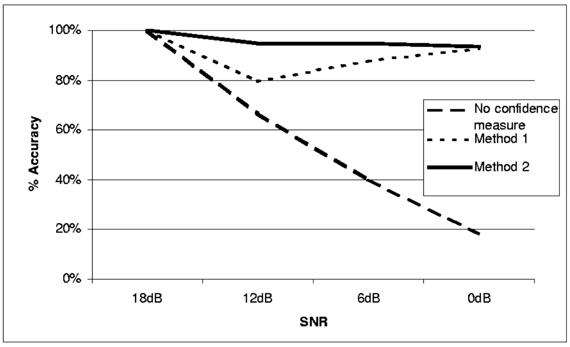


Figure 5: % Accuracy of System at varying SNRs

The confidence measures are based on scores obtained from the GMMs when test speech is applied to the recognition system. The score for a given test speech utterance is obtained from the i^{th} GMM using the log likelihood.

$$S(i)=\log P(\vec{X}|GMM_i) \quad i=1..M$$

where M equals the number of GMM models.

We used two methods to obtain the confidence measures based on these scores. Method 1 used the output score from the GMM in $\vec{X}_s$ corresponding to the winning model from the primary classifier. Method 2 uses the scores from all of the OPD GMMs, and calculates the confidence score as the quotient of the highest GMM score and the second highest GMM score. (Figure 4)

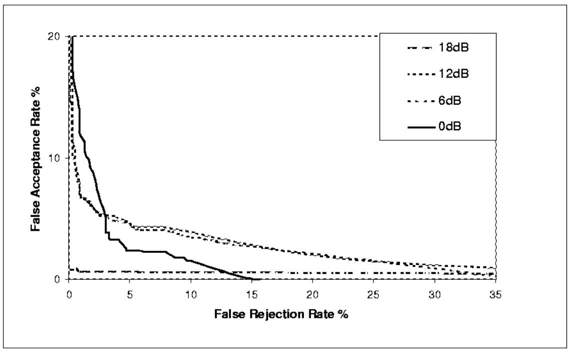
In both cases, the confidence score is compared against a threshold, and a binary accept / reject decision is made.

5 Results

5.1 Database Testing

We used the TI46 database for clean speech, and corrupted it with varying degrees of noise from the NOISEX-92 database (factory room noise and car noise) for noisy speech. The primary HMM classifier was trained from the training set of the TI46 database, using 5 states and 8 mixtures per model. GMMs for the secondary classifier were trained from both the clean and noisy speech, and used 3 mixtures per model.

Figure 5 shows results obtained from applying both methods 1 and 2 to speech data that had been corrupted by varying degrees



SNR	% Accuracy	% Rejection
Clean	99.24	0.38
12dB	92.44	30.73
6dB	92.32	53.78
0dB	93.32	84.26

Figure 6: Performance of Method 2 confidence measures over varying SNRs

of noise. Method 2 yields the best results, achieving an accuracy of 93.3% at 0dB, which means that the system correctly classified input speech and rejected incorrectly classified speech 93.3% of the time. The rejection rate was 84.2%, indicating how much input data is rejected by the confidence system, including both false and true rejections.

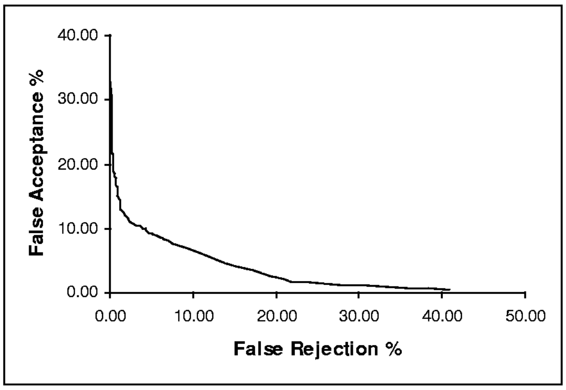


Figure 7: Performance of Method 2 confidence measures with 20% OOV data

The receiver operating characteristics (ROC) curve in Figure 6 clearly indicates that this system also has a very low false acceptance and false rejection rate, leading to its good performance. Method 2 was used to produce this figure.

We added 20% more OOV data into the existing database and ran the tests again. The results appear in Figures 7. Note that testing was not done for different SNRs, so there is only curve. The system achieved an accuracy of 86.5% with a rejection rate of 45.5%.

5.2 Real-world testing

The same system was tested in a factory environment. We developed an application to sort parcels using speech recognition. The system used a vocabulary of 24 destinations, and operated in SNRs ranging from 15dB to 2.3dB. Because the noise in the factory varied randomly, it isn't possible to quote results at different levels of SNR. The results were very good, with performance increasing from 82.4% to 97.4% using OPD confidence measures. The rejection rate was 13.5%.

6 Conclusion

The use of OPDs has been shown to have advantage in limited vocabulary applications. It also shows a good resistance to noise, particularly when the OPD models are trained with both clean and noisy speech. The use of OPD based classifiers help to reduce performance errors due to OOV input.

Further research needs to be done to in the following areas:

- Determine the performance in larger vocabulary systems - the largest vocabulary tested was 24.
- Determine an optimum OPD feature set.

7 Bibliography

[1] Roxane Lacouture, Yves Normandin, "Detection of Ambiguous Portions of Signal Corresponding to OOV words or Misrecognised Portions of Input," *ICSLP 1996*

[2] M. Siu, H. Gish, F. Richardson, "Improved Estimation, Evaluation and Applications of Confidence Measures for Speech Recognition", ICLSP 96

[3] J. Caminero, C. de la Torre, L. Villarrubia, C. Martin, L. Hernandez, "On-line Garbage Modeling with Discriminant Analysis for Utterance Verification", *ICSLP 1996*

[4] W.W. Peterson, t.G. Birdsall, W.C. Fox, "The theory of signal detectability, " *IRE Trans. Info. Theory,* PGIT-4, pp 171-212, Sept 1954.

[5] J.G.Wilpon, C.H. Lee, L.R. Rabiner, "Application of Hidden Markov models for recognition of a limited set of words in unconstrained speech," *ICASSP 1989,* pp. 254-257.

[6] E. Bernstein, W.R. Evans, "OOV Utterance Detection based on the Recogniser Response Function"

[7] M.P. Cooke, A. Morris, P.D. Green, "Recognizing Occluded Speech", Proceedings of the ESCA Tutorial and Research Workshop on The Auditory Basis of Speech Perception, Keele University, UK, 15-19, July 1996, 297-300, San Mateo, 1996.

[8] E. Chang, R. Lippmann, "Improving Wordspotting Performance with Artificially Generated Data", Proc. ICASSP 1996, 526-529

[9] T. Vaich, A. Cohen, "Robust Isolated Word Recognition using WSP-PMC Combination", Proc. Eurospeech 1997

[10] D. A. Reynolds, R.C. Rose, “Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models”, IEEE Trans. Speech and Audio Proc. , Vol 3, No 1, January 1995.