DISCRIMINATIVE UTTERANCE VERIFICATION USING MULTIPLE CONFIDENCE MEASURES

Piyush Modi and Mazin Rahim

AT&T Labs
180 Park Avenue, Florham Park, New Jersey 07932-0971, USA
Email: piyush@research.att.com, mazin@research.att.com

ABSTRACT

This paper proposes an utterance verification system for hidden Markov model (HMM) based automatic speech recognition systems. A verification objective function, based on a multi-layer-perceptron (MLP), is adopted which combines confidence measures from both the recognition and verification models. Discriminative minimum verification error training is applied for optimizing the parameters of the MLP and the verification models. Our proposed system provides a framework for combining different knowledge sources for utterance verification using an objective function that is consistently applied during both training and testing. Experimental results on telephone-based connected digits are presented.

1. INTRODUCTION

When deploying automatic speech recognition (ASR) services on a wide scale, where any user can access a service at any time and from anywhere, it is essential to accommodate for the wide range of acoustic and language variabilities that could severely degrade recognition performance. One means of improving an ASR system performance when dealing with naturally spoken utterances is through utterance verification (UV). The objectives in UV are to reject out-of-vocabulary events, to detect incorrectly recognized keyword events and to determine which part of an input utterance is reliably detected.

Utterance verification can be considered as a statistical hypothesis problem, where the aim is to test the *null* hypothesis (H_0) - which assumes that a given keyword exists in a segment, $\mathbf{O}$, of an utterance against the *alternative* hypothesis (H_1) - which assumes that the keyword does not exist, or is incorrectly recognized, within that utterance segment $\mathbf{O}$. Based on the Neyman-Pearson Lemma [1], the optimal test that maximizes the *power* of the test can be constructed using the likelihood ratio statistics, such that a keyword hypothesis, i, in a segment of speech $\mathbf{O}$ is rejected if its likelihood ratio

$$\mathcal{LR}(\mathbf{O}) = \frac{p_i(\mathbf{O}|H_0)}{p_i(\mathbf{O}|H_1)} \qquad (1)$$

falls below a verification threshold τ_i, where $p_i(\mathbf{O}|H_0)$ and $p_i(\mathbf{O}|H_1)$ are the probability density functions of the null and the alternative hypotheses, respectively.

When dealing with HMM-based recognizers, where neither $p_i(\mathbf{O}|H_0)$ nor $p_i(\mathbf{O}|H_1)$ are known exactly, the Neyman-Pearson Lemma is no longer guaranteed to be optimal nor does it ensure maximum separation of the null and the alternative hypotheses. In [6], it was shown that discriminative training the recognition models using minimum classification error (MCE), can yield improvement in both recognition and verification performance. Further, it was demonstrated in [6, 2, 8] that improved verification performance can be achieved by designing a confidence measure (CM) based on a separate set of HMMs, referred to as verification models, using minimum verification error (MVE) training. Improved verification performance on connected digits recognition was also reported by Setlur and Sukkar [7] when combining multiple CMs using a linear Fisher discriminator.

In this paper, we propose a system for integrating multiple confidence measures (MCM) in UV. The so called, UV-MCM, adopts a MLP for integrating different confidence measures. Two CMs are adopted in this study based on the likelihood ratio statistics of the recognition and verification models. Both measures are computed at the utterance level and are combined using an objective function that is consistently applied in both training and testing. Our system provides a framework for training the parameters of the MLP and the verification models using a discriminative measure that aims to minimize verification error rate. The application of the UV-MCM system on a telephone-based connected digits recognition task is reported in this paper.

2. UV USING MULTIPLE CONFIDENCE MEASURES (UV-MCM)

Figure 1 shows a block diagram of the UV-MCM system. The system is essentially a two-pass process that involves recognition followed by verification. During both recognition and verification, multiple CMs are computed, normalized to accommodate for the wide dynamic range, and finally integrated. The output of the integrator determines whether or not to accept the recognized string.

In general, let $\{CM_1, \ldots, CM_M\}$ be a sequence of confidence measures corresponding to the parameter set $\Lambda = \{\Lambda^{(j)}\}_{j=1,\ldots,M}$, Given a class C_i, the objective

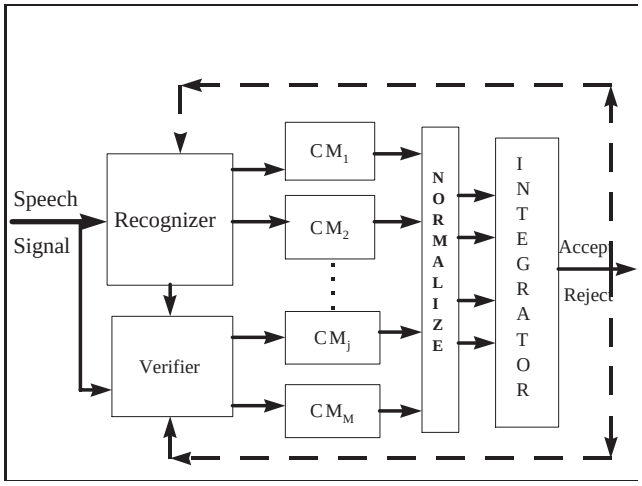

Figure 1: A block diagram of the UV-MCM system.

is to set the output of the integrator function so that

$$d_i^{(MCM)} = J_\Psi\{\overline{d}_i^{(j)}(\mathbf{O};\Lambda^{(j)})_{j=1\ldots M}\} \begin{cases} >= \tau_i & \text{If } \mathbf{O} \in C_i \\ < \tau_i & \text{otherwise,} \end{cases} \quad (2)$$

where Ψ are the parameters associated with integrator function $J_\Psi\{.\}$ and $\overline{d}_i^{(j)}(\mathbf{O};\Lambda^{(j)})_{j=1\ldots M}$ are the normalized CMs computed from the recognition and verification models.

In this study, we have investigated two acoustic-based CMs that are derived at the utterance level. The first is commonly applied in MCE training and is used for designing the recognition models [3, 4]. The so called *misclassification* measure is defined as:

$$d_i^{(MCE)} = d_i^{(1)}(\mathbf{O};\Lambda^{(1)}) = -g_i(\mathbf{O};\Lambda^{(1)}) + G_i(\mathbf{O};\Lambda^{(1)}), \quad (3)$$

where $g_i(\mathbf{O};\Lambda^{(1)})$ is the normalized log likelihood for string class C_i, $G_i(\mathbf{O};\Lambda^{(1)})$ is the normalized log likelihood for the competing classes to C_i and $\Lambda^{(1)}$ are the parameters of the recognition HMMs.

The second CM is applied in MVE training and is used for designing the verification models [6]. The so called *misverification* measure is defined as:

$$d_i^{(MVE)} = d_i^{(2)}(\mathbf{O};\Lambda^{(2)}) = -s_i(\mathbf{O};\Lambda^{(2)}) + S_i(\mathbf{O};\Lambda^{(2)}), \quad (4)$$

where

$$s_i(\mathbf{O};\Lambda^{(2)}) = \log\Big[\frac{1}{N(i)}\sum_{q=1}^{N(i)} \exp\{\kappa \cdot \mathcal{LR}_{i(q)}(\mathbf{O}_q;\Lambda^{(2)})\}\Big]^{\frac{1}{\kappa}} \quad (5)$$

and $S_i(\mathbf{O};\Lambda^{(2)})$ is a confidence measure for the competing classes to C_i. $N(i)$ is the number of keywords for C_i, κ is a negative constant, $\mathbf{O}_q$ is the speech segment for the q^{th} word, and finally $\mathcal{LR}(.)$ is a likelihood ratio computed from the verification models $\Lambda^{(2)}$, namely, keywords, anti-keywords and filler.

When combining different CMs based on the criterion set in Eqn. 2, there is always a question of what integrator to use that best minimizes the misverification error rate. Figure 2 shows a scatter diagram of the two CMs defined by Eqns. 3 and 4. In this figure, the 'x' represents correct recognition and 'o' represents recognition errors. It is clear from the figure that the two classes are not linearly separable and adopting, for example, a Fisher discriminator, as proposed in [7], is unsuitable for our purpose. To accommodate for non-linear decision boundaries, a MLP is adopted in this study which acts as the integrator function in Eqn. 2.

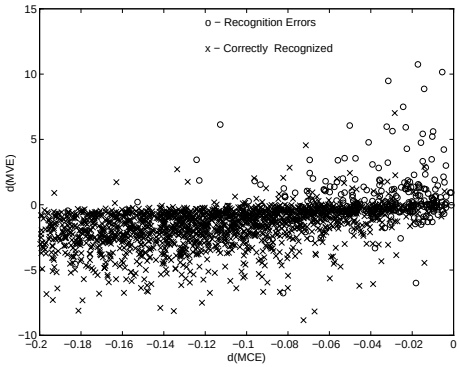

Figure 2: A scatter diagram for $d^{(MCE)}$ vs. $d^{(MVE)}$

The MLP is designed to have two inputs, one for the misclassification distance in Eqn. 3 and one for the misverification distance in Eqn. 4. A single node is used in the output layer to determine whether to accept or to reject the recognized string. The procedure for training the MLP and the verification model parameters is discussed next.

3. UV-MCM TRAINING

Clearly the main objective when training the UV system is to minimize the verification error rate. This includes reducing both false rejection and false acceptance. In our framework, one method to achieve this is to minimize an objective function involving $J_\Psi\{.\}$ over all classes in the training set, such that

$$I = \sum_i [J_\Psi\{\overline{d}_i^{(j)}(\mathbf{O};\Lambda^{(j)})_{j=1\ldots M}\} - T_i]^2, \quad (6)$$

where,

$$T_i = \begin{cases} 1 & \text{If } \mathbf{O} \in C_i \\ 0 & \text{otherwise.} \end{cases}$$

The objective function I is essentially a mean square error (MSE) distance which is compatible with the method for training MLPs. Minimizing I can, in theory, be achieved by applying gradient descent to the parameters of the MLP as well as the recognition and verification models. So that at the n^{th} iteration of the training procedure:

$$, _{n+1} = , _n - \epsilon_n \frac{\partial I}{\partial , } \Big|_{\Gamma=\Gamma_n}, \quad (7)$$

where $, = \{\Lambda, \Psi\}$, ϵ_n is a positive learning rate and $\partial I/\partial ,$ is the gradient of I with respect to the parameters $, .$

To update the parameters of Ψ, the standard back-propagation training described in [5] is used. The update rule for Λ, on the other hand, is somewhat similar to the MVE framework of [6], with the exception of using a MLP as an integrator as opposed to a sigmoid activation function. Therefore, we can form a chain rule for computing $\partial I/\partial \Lambda^{(j)}$, such that

$$\frac{\partial I}{\partial \Lambda^{(j)}} = \sum_i \frac{\partial I}{\partial \overline{d}_i^{(j)}} \cdot \frac{\partial \overline{d}_i^{(j)}}{\partial \Lambda^{(j)}}, \qquad (8)$$

where $\partial I/\partial \overline{d}_i^{(j)}$ is the gradient of the objective function I with respect to the inputs to the MLP, which is a straightforward extension to the back-propagation algorithm. The gradient $\partial \overline{d}_i^{(j)}(.)/\partial \Lambda^{(j)}$ is given in [4] and [6] for the MCE and MVE measures, respectively.

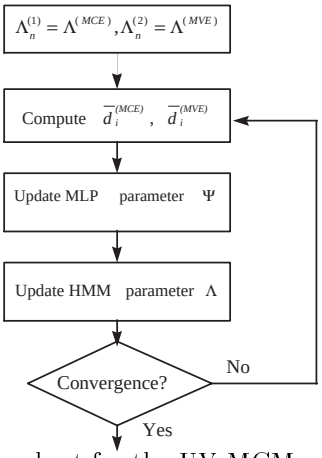

Figure 3: Flow chart for the UV-MCM system.

In principle, we should be able to update Ψ and Λ simultaneously. However, since this implies updating the verification features as well as the verification decision boundaries at the same time, it commonly causes convergence problems. Therefore, an iterative scheme is adopted for updating Λ and Ψ as shown in Fig. 3. Beginning with a set of boot-strap models for recognition and verification, we compute the misclassification distance $d^{(MCE)}$, in Eqn. 3, and the misverification distance $d^{(MVE)}$, in Eqn. 4. Upon normalizing those distances based on their first order statistics, they are passed to the MLP and back-propagation training is performed. The final step includes updating the verification model parameters using the updated MLP to minimize the objective function of Eqn. 6. This entire process can be repeated for few iterations until satisfying some convergence criterion.

In the current study, the MCE trained recognition HMMs and the MVE trained verification HMMs were used as the seed HMMs in the training procedure described above.

4. EXPERIMENTS

The purpose of the experiments presented in this section is to demonstrate the utility of the proposed framework that allows one to integrate multiple confidence measures in a consistent training and testing framework. The objective of these experiments is to identify and reject incorrectly recognized valid-digit strings. Clearly this is a much tougher problem than rejecting invalid spoken input.

A speaker-independent telephone-based connected digit database was used in this study. It consisted of 16-digit credit card numbers that were recorded from a variety of environmental conditions and telephone handsets. 2639 utterances were assigned for training and 713 utterances were assigned for testing. Feature analysis included computing 12 cepstral coefficients plus energy along with their first and second order time derivatives.

The recognition HMMs included 274 context-dependent subword units with 3-4 states per model, and 4 mixture components per state. The verification HMMs included 69 context-independent subword units (34 keywords, 34 anti-keywords and 1 background/filler). Both the recognition and verification HMMs were initialized through MCE and MVE training respectively. The 2-layer MLP used to integrate, the MCE and MVE distances in Eqns. 3 and 4, respectively, consisted of 2 input nodes, 4 hidden layer nodes and 1 output node.

Iteration	% Verification Error	Avg MSE
1	6.45	0.053
2	4.78	0.046
3	4.13	0.041
4	4.06	0.039

Table 1: Verification performance of the UV-MCM on the training database.

The first set of results presented in Table 1 shows the verification error rate (i.e., false acceptance plus false rejection) and the average MSE for the first four iterations on the training data. As one would expect, minimizing the MSE rate leads to a reduction in the verification error rate. Table 2 presents the

System	EER %	MER %
Baseline	24.01	43.75
$d^{(MVE)}(\mathbf{O};\Lambda^{(2)})$	16.20	31.72
$d^{(MCE)}(\mathbf{O};\Lambda^{(1)})$	13.29	23.56
$d^{(MLP)}(\mathbf{O};\Lambda)$	12.28	20.97
$d^{(MCM)}(\mathbf{O};\Lambda)$	9.63	18.40

Table 2: UV performance in terms of EER and MER for various systems.

verification performance in terms of equal error rate (EER) and minimum error rate (MER), when evalu-

ating the fourth iteration model on the test data. The baseline results refer to the performance of the verification system when using ML estimation for training the verification models and the measure in Eqn. 4 for testing (see [6]). Applying the MVE measure in Eqn. 4 consistently in both training and testing leads to the results shown for $d^{(MVE)}$. This amounts to 33% and 28% reductions in the EER and MER, respectively, over the baseline system.

When using the recognition models alone to provide the verification score, as pointed out in Eqn. 3, we achieved 45% and 46% reductions in EER and MER, respectively, over the baseline system (see Table 2, $d^{(MCE)}$). Integrating the two confidence measures, namely, $d^{(MCE)}$ and $d^{(MVE)}$ by simply training a MLP led to a minor improvement in verification performance (see $d^{(MLP)}$). Finally, we adopted the training procedure we outlined in Section 3, in which both the MLP and the verification models were updated by minimizing the objective function in Eqn. 6. The results, shown under $d^{(MCM)}$, demonstrate 60% and 58% reductions in the EER and MER, respectively, over the baseline performance. This is equivalent to a reduction of 41% and 42%, respectively, as compared to the $d^{(MVE)}$ system, and a reduction of 28% and 22%, respectively, as compared to the $d^{(MCE)}$ system. We should point out that the difference in the performance between $d^{(MLP)}$ and $d^{(MCM)}$ attribute to the consistent training and testing strategy outlined in Section 3.

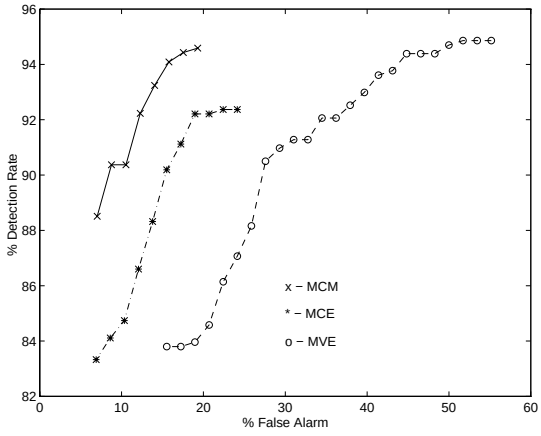

Figure 4: ROC curves.

Figure 4 shows the receiver operating characteristics (ROC) curves (false alarm rate versus detection rate) for $d^{(MVE)}$, $d^{(MCE)}$ and $d^{(MVE)}$. A plot of the rejection rate versus string error rate for these three measures is also shown in Fig. 5. From these plots, one can conclude that our proposed system provides an additional benefit in verification performance over using either MVE or MCE alone. This improvement is rather substantial considering that we are testing on valid strings only.

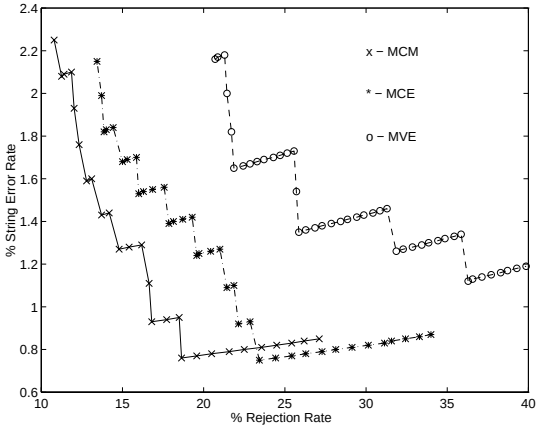

Figure 5: Rejection rate vs. string error rate

5. SUMMARY

This paper described an utterance verification system for integrating multiple confidence measures. The UV-MCM system adopts a MLP for integrating confidence measures that are computed from both the recognition and verification models. It is designed using an objective function that is consistently applied in both training and testing. Discriminative training is performed on the parameters of the MLP and the verification models with the aim of minimizing the verification error rate over the entire training data. Our results on connected digits recognition are very promising and show improvement in verification performance when combining multiple CMs as opposed to using each CM alone. Although more experiments need to be conducted on larger tasks and data, we believe that our system provides a general framework for integrating multiple knowledge sources in UV, such as bands of signal, language information or even the visual cues derived from lip reading.

6. REFERENCES

1. P. Bickel and K. Doksum. *Mathematical Statistics.* Prentice-Hall, Englewood Cliffs, 1977.
2. Rose R. C., Juang B. H., and Lee C. H. A training procedure for verifying string hypothesis in continuous speech recognition. In *Proc. ICASSP*, 1995.
3. W. Chou, B.-H. Juang, and C.-H. Lee. Segmental GPD training of HMM based speech recognizer. In *Proc. ICASSP*, pages 473–476, 1992.
4. B. H. Junag, W. Chou, and C.-H. Lee. Minimum classification error rate methods for speech recognition. In *IEEE Trans. Speech and Audio Processing, Vol. 5, No. 3*, pages 257–265, May 1997.
5. Lippman R. P. An introduction to computing with neural nets. In *IEEE Trans. Acous., Speech, & Sig. Processing Magazine*, pages 4–22, April 1987.
6. M. Rahim, C.-H. Lee, and B-H. Juang. Discriminative utterance verification for connected digits recognition. In *Proc. European Conf. on Speech Communication and Technology*, 1995.
7. A. Setlur, A. Sukkar, and J. Jacob. Correcting recognition errors via discriminative utterance verification. In *Proc. ICSLP*, pages 602–605, 1996.
8. R. Sukkar, A. Setlur, M. Rahim, and C.-H. Lee. Utterance verification of keyword strings using word-based minimum verification error (wb-mve) training. In *Proc. ICASSP*, pages 451–454, 1993.