CONFIDENCE METRICS FOR SPEAKER IDENTIFICATION

Mark C. Huggins\textsuperscript{1} and John J. Grieco\textsuperscript{2}

\textsuperscript{1} ACS Defense, Inc., Rome, NY USA
\textsuperscript{2} Air Force Research Laboratory, Rome, NY USA
Mark.Huggins@rl.af.mil
John.Grieco@rl.af.mil

ABSTRACT

This paper presents a concept and experimental results for a novel confidence metric for speaker identification (SID). The concept is based on three major audio components: duration, signal-to-noise ratio, and model quality. This concept has been successfully applied to two benchmark closed-set SID systems which demonstrates the broad applicability of the technique. A subset of the TIMIT database was used for this investigation. Derivation and validation of the confidence performance are reported in this paper. We also investigate a fusion technique for the composite multi-component confidence metric. Reliability of the confidence metric has been determined to be 94.6% accurate on average.

1. INTRODUCTION

Many years of research have been spent in the area of speech processing and in particular speaker recognition. The primary focus of that research has been on raw performance. As real-world data sets became available, researchers focused on many techniques to make feature sets and classifiers more robust for speaker recognition. That research concentrated on improving raw performance, with varied success. Often overlooked was a measure of trust in the final result. In many commercial, law enforcement, and military applications, the lack of reliable confidence metrics has limited the broad appeal of speaker recognition technology. There are, however a variety of scenarios where less than perfect performance is acceptable when an automated system can gauge its expected performance on a given token of speech. To this end, little research has concentrated on confidence metrics. Ong et. al. [1] studied confidence as a function of population size and Gish and Schmidt [2] applied Bayes rule to the distribution of correct and incorrect decisions. Both of these papers concentrated on confidence from a single dimension. This paper will present confidence as a multi-dimensional function based on the following factors:

- Mismatch conditions between training and testing
- Amount of testing and training
- Magnitude of dissimilarity measurement between models and test samples
- Population size or model overlap and separation

The first factor is the degree of matched conditions of which there are two aspects—data quality and channel mismatch. To measure data quality, we use signal-to-noise ratio (SNR). SNR significantly impacts the confidence measure from two perspectives, the absolute magnitude of the noise as well as the mismatch between testing and training. The other condition affecting confidence is channel. If training and testing audio are processed through different channels, then speaker dependent spectral characteristics can be affected. Hence, we should expect a lower confidence measure. As it currently stands, the channel mismatch is the weakest link in our confidence measure function because there are infinitely many locations in which spectral holes or tones can be placed.

The second factor, the amount of training and testing data, is typically measured in seconds. Intuitively, the greater the training or test duration from a representative class, the more likely that class dependent characteristics will be included in that data set. Consequently, we should expect a higher measure of confidence with respect to duration.

The third factor affecting confidence is the magnitude of the dissimilarity measurement between models and test feature vectors. One should expect an enrolled speaker’s test audio to produce feature vectors that are close to his/her model; consequently, a small dissimilarity magnitude is expected.

The last confidence factor is population size or model separation. If speaker models overlap, which can happen if speakers have similar vocal characteristics, there will be more confusion between those speakers’ models. Hence, a method for measuring model overlap can provide a prediction of classifier performance with respect to model separation.

Of the factors affecting speaker identification performance, only duration, SNR, and model quality are studied in this paper.

2. SPEAKER RECOGNITION SYSTEMS AND DATASET

Two closed set speaker recognition systems were used for evaluation. Each system has training and testing phases. In the training phase, a speaker’s speech is used to derive his/her model. This training speech was varied in duration and SNR. Likewise during the recognition phase, varied test tokens were compared to trained speaker models. The two systems are briefly described below and appropriate references are given from which detailed descriptions can be obtained.

2.1. Multi-feature-Vector Quantization (MF-VQ) Speaker Recognizer

The MF-VQ speaker identification system utilized in this research is based upon the multi-feature classifier fusion technique described by Ricart et. al. [3]. During training, 14-dimensional LPC cepstra, delta-cepstra and Hamming liftered cepstral feature vectors were derived from each 32ms speech frame with 50% frame overlap. These vector streams were...
input to the Linde-Buzo-Gray VQ algorithm in order to create three separate 128-word codebooks per speaker. During testing, the classification was based on the L2 mean-square error (mse) between codebooks and unknown feature vectors. Each feature's mse score for each speaker model was normalized by that feature's minimum mse. The three normalized feature scores were linearly fused, and the speaker having the lowest fused score was declared the closed-set winner.

2.2. Gaussian Mixture Model (GMM) and Universal Background Model (UBM) Speaker Recognition

The Gaussian Mixture Model (GMM) and Universal Background Model (UBM) approach, developed by Reynolds [4], was also used in this research. Front-end feature processing consisted of mel-weighted and delta fft-cepstra generated from a frame size of 20ms with 50% overlap. In the UBM version of this system, a single universal GMM is trained from a large number of non-target speakers. Models for individual speakers are created by adapting a 1024-mixture UBM. During recognition, the likelihood of the test speech is computed for each of the GMMs produced during training. In the implementation used in this paper only 5 mixtures are used for the calculation of the likelihood of a particular speakers’ GMM model. The five mixtures are chosen from the most probable mixtures in the UBM. For closed-set recognition, the speaker corresponding to the most likely GMM is hypothesized as the speaker of the test utterance.

2.3. Dataset

The data used was a subset of the TIMIT database, comprised of speakers from the New England (dr1), Northern (dr2), and the North Midland (dr3) dialect regions. This subset was further partitioned into 3 non-overlapping sets, one for deriving the confidence metrics (CM), one for UBM training, and one for CM validation. All data was 8kHz and silence removed. The CM derivation set and the validation sets each contained 40 speakers with a 1.6:1 male/female ratio. The UBM contained 117 speakers. Speaker identification training and testing was comprised of silence removed speech utterances of 0.5 sec to 5 sec, in 0.5 sec steps, and from 6 to 8 seconds in 1 sec steps. These utterances were mixed with pink noise to levels of 24 to 6dB SNR in 3dB steps. Ninety one separate model spaces were created, one for each SNR-duration pair. This breakdown facilitated the derivation of the confidence metrics for durational and noise mismatch conditions.

3. CONFIDENCE METRICS

3.1. SNR Metric

Alluded to in the introduction, expected performance is treated as our measure of confidence. In this context, expected performance is how often correct speaker decisions are expected. For SNR, expected performance is derived by conducting experiments varying train-test SNRs and then measuring performance. From these data points, least-squares regression is performed to derive the SNR confidence metric. For example, Figure 3.1 is the regression surface for VQ performance versus SNR when train-test durations are fixed at 8secs. The actual surface’s parabolic shape prompted the investigation of second degree functions for fitting the surface, finally settling on

\[ C(L_t, L_s) = a_0 + a_1 L_t + a_2 L_s + a_3 L_t^2 + a_4 L_s^2 + a_5 L_t L_s \]

where \( L_t \) is the train dB level and \( L_s \) the test dB level. Regression analysis for Figure 3.1’s data points yielded the following values for coefficients:

<table>
<thead>
<tr>
<th>SNR</th>
<th>( a_1 )</th>
<th>( a_2 )</th>
<th>( a_3 )</th>
<th>( a_4 )</th>
<th>( a_5 )</th>
<th>( a_6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tr&lt;Ts</td>
<td>.0212</td>
<td>.1471</td>
<td>-.0033</td>
<td>-.0303</td>
<td>.0005</td>
<td>-.0006</td>
</tr>
<tr>
<td>Tr&gt;=Ts</td>
<td>.2989</td>
<td>-.0913</td>
<td>.0022</td>
<td>.1589</td>
<td>-.0035</td>
<td>-.0004</td>
</tr>
</tbody>
</table>

Table 1 Regression Coefficients

A subtle but interesting characteristic of this surface is that for any two given dB levels \( L_t \) and \( L_s \) such that \( L_s < L_t \), the performance is better if the model space was derived from audio rated at \( L_s \) dB and testing against \( L_t \) dB audio, as opposed to when train dB is greater than test dB. We have taken advantage of this fact by doing piecewise regression, splitting the data surface into the two pieces determined by the sets \( \{ L_s, L_t, p(L_t, L_s) \mid L_s < L_t \} \) and \( \{ L_s, L_t, p(L_t, L_s) \mid L_s \geq L_t \} \) where \( p(L_t, L_s) \) is the percent correct when training with \( L_s \) dB and testing on \( L_t \) dB speech.

Figure 3.1 Performance versus SNR

3.2. Durational Metric

For incorporating confidence with respect to duration, the train-test file lengths are varied and the resulting performances used for deriving confidence; however, an explicit duration confidence formula will not be created. Instead, duration confidence will be implicit in the SNR confidence regression process. This process will become clear in Section 3.4.

3.3. Model Confidence Metric

Unlike the SNR or duration confidence metrics, confidence with respect to model overlap is not derived by regression. For the VQ classifier, we use the volume of intersecting hyper-cubes centered about speaker codewords to estimate model overlap and therefore model performance. For the GMM classifier, we employ the technique for composite classes outlined in [5] to predict model performance.

3.3.1. Vector Quantization Models

Since speaker models generated by a vector quantizer occupy Euclidean space, a natural geometric way to measure the
overlap between models is to use the intersecting volume between hyper-geometric objects centered about each codeword such that those geometric shapes approximate the distribution of that codeword’s training vector assignments. Unfortunately, the limited amount of data available for model generation usually means that a reliable distribution about each codeword cannot be obtained. To overcome this practical problem, the codebook sizes for measuring model overlap are smaller than the codebook size for training.

It is obvious that determining the intersection of arbitrarily shaped objects in an n-dimensional space is not trivial. However, the intersection of hypercubes is relatively easy to compute. Consequently, hypercubes which approximate the codewords’ distributions are used.

Another problem faced was controlling the factorial-like explosion in computation when computing the intersecting volume between codewords of different classes. Instead, for each codeword, only the intersection between its hypercube and the hypercube of its n-closest cohorts is considered. These intersecting volume centroids are used to predict the model space’s classification performance. The sum of the intersecting volumes divided by the total volume is proportional to the probability of error, PE. Consequently, 1 – PE is proportional to the identification rate. So using cohorts to keep computational complexity down, for a speaker set of size M, speaker i has N(i) codewords, PE is estimated by:

\[ PE = PE = \frac{1}{M} \sum_{j=1}^{N(i)} p_{ij} \]  

with

\[ p_{ij} = \frac{1}{N(i)} \sum_{k=1}^{N(i)} \frac{vol(H(i,j) \cap K(i,j))}{vol(H(i,j))} \]  

where H(i,j) is codeword c(i,j)’s hypercube and K(i,j) is the union of the hypercubes of H(i,j)’s n-closest cohorts contained in other codebooks.

### 3.3.2. Gaussian Mixture Models

Incorporating GMM model confidence into the Speaker ID global confidence is a straightforward implementation of Garber and Djuaidi’s iterative technique for computing upper and lower bounds on the Bayes risk for composite classes [5]. Before their technique is outlined, the necessary notation is introduced. Suppose there are M GMMs, \( \omega_1, \ldots, \omega_M \), each GMM i having N > 1 mixtures, \( \omega_{i1}, \ldots, \omega_{iN} \). Let \( p_{i}^{(n)} \) denote the mixture weight (i.e. marginal probability) of \( \omega_{in} \). Let \( m_{ij} \) denote the mean vector of \( \omega_{ij} \). Let \( \sigma_{ij} \) denote the cross-covariance matrix between classes i and j. Suppose \( \rho \) takes on \( L \) values from the set \( \rho = \{k, \ell | k, \ell \in [1, \ldots, N]\}, \ L \leq 2N \). Then the upper bound on the Bayes risk \( B(k, \ell, \rho) \) for mixtures k and \( \ell \) of GMMs i and j, respectively, is given by,

\[ B(k, \ell, \rho) \leq \frac{P_{i}^{(n)}P_{j}^{(n)}}{L} e^{\frac{-1}{2} \sum_{m=1}^{L} \left( m_{ij} - m_{ij} \right)^{2}}, L = 2, \ldots, 2N \]  

where

\[ A = \left\| m_{ij} \right\| + \left\| m_{ij} \right\| - \frac{1}{L} \sum_{m=1}^{L} \left\| m_{ij} + m_{ij} - \frac{1}{L} \sum_{m=1}^{L} m_{ij} \right\| \]  

Applying (5) gives the pairwise bounds for all mixture pairs of classes i and j. Adding the minimum values of (5) gives us the upper bound for Bayes risk, \( BR_{ij} \), for GMMs i and j:

\[ BR_{ij} \leq \sum_{k=1}^{N(i)} \sum_{\ell=1}^{N(j)} \min \{B(k, \ell, \rho_{ij})\} = B_{ij} \]  

Lastly, an upper bound for the Bayes risk, \( BR \), for M GMMs with equal prior probabilities is given by

\[ BR \leq \frac{2}{M} \sum_{i=1}^{M} \sum_{i=1}^{M} B_{ij} \]  

Thus, \( 1 - BR \) gives a lower bound on the average correct classification for M GMMs.

### 3.4. Composite Confidence Metric

The fusion of the confidence metrics is inherent in the regression procedure that is used. For each train-test duration pair, we regress on SNR for that pair—giving us 169 regression formulae (13 different train and test durations). Consequently, confidence with respect to duration is incorporated into the SNR metric by selecting the SNR regression formula indexed by the appropriate duration pair. To see how confidence is computed for a train-test duration pair that does not coincide with any pair used for regression, let us introduce some notation. Let \( C_{(\omega,)} (L_x, L_y) = c \) denote the confidence computed from the regression formula derived from regressing on SNR when training and test duration are fixed at \( x \) and \( y \) respectively. Then if \( (L_x, L_y, x', y') \) is a train-test SNR and duration ordered 4-tuple such that \( (x', y') \) is not included in the original 169 experimental pairs, confidence could be computed by using regression formulae \( C_{(\omega,)} (L_x, L_y) \) and \( C_{(\omega,)} (L_x, L_y, x, y, c) \) where \( x, y, y, y, y, y \) are less than \( x, y, y, y, y, y \), and interpolating between points \( (L_x, L_y, x, y, c) \) and \( (L_x, L_y, x', y', c) \) such that

\[ C_{(\omega,)} (L_x, L_y) = c \]  

Hereafter, the confidence metric derived using the fusion technique will be referred to as SNR-Duration CM.

What is left now is fusing model confidence into the process. From empirical observations, the VQ model confidence exhibits large error between actual and predicted performance for low train SNRs or short train durations; consequently, fusing model confidence with SNR and duration confidence for these low train SNRs or short train durations will only lessen the utility of measuring confidence. Hence, only for those train-test SNRs and durations for which expected performance was observed to be above 90% will model confidence be fused with the SNR-Duration CM. The GMM model confidence metric will be fused under the same conditions as for the VQ models.
For those cases warranting model confidence fusion as defined above, fusion is performed through an additional regression layer by computing the regression coefficients $b_i$ for

$$b_i + b_i C_{i,L_x}(L_x, L_y) + b_i C_{i,y}(L_x, L_y) = p$$ (10)

4. RESULTS/VALIDATION

The results were derived from the validation dataset described in Section 2.3. The error in predicted performance (a.k.a. confidence) vs. actual performance is presented next for both VQ and GMM classifiers. A cross section of conditions, SNR and duration, were used to provide this error. From this error the reliability of the proposed confidence metric is revealed. These errors are low enough to justify using predicted performance as a measure of confidence.

The results were derived from 40 speakers with a total of 3240 trials. These trials were comprised of training and testing combinations having SNR values of 6, 15, and 24dB with duration combinations including 8, 3, and .5 seconds for training and 7, 3, and .5 seconds for testing.

The average error in the confidence vs. actual validation results are displayed in Table 2. These results are based only on the SNR-Duration confidence.

<table>
<thead>
<tr>
<th></th>
<th>Error</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>VQ</td>
<td>5.4%</td>
<td>5.1</td>
</tr>
<tr>
<td>GMM</td>
<td>7.2%</td>
<td>5.7</td>
</tr>
</tbody>
</table>

Table 2 Validation Results for SNR and Duration

An interesting observation in the results is noted in Table 3 below. Here we find that for both systems the predicted error in performance increases as the mismatch between training and test SNR is decreased.

<table>
<thead>
<tr>
<th>SNR Mismatch</th>
<th>VQ</th>
<th>GMM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Error</td>
<td>St. Dev.</td>
</tr>
<tr>
<td>18 dB</td>
<td>3.6%</td>
<td>2.7</td>
</tr>
<tr>
<td>9 dB</td>
<td>5.3%</td>
<td>4.9</td>
</tr>
<tr>
<td>0 dB</td>
<td>6.7%</td>
<td>6.2</td>
</tr>
</tbody>
</table>

Table 3 Validation Performance wrt SNR Mismatch

Based on the regression curves, model confidence was derived and incorporated based on a 90% correct rule. This yielded the following region: Train dB $\geq 12$ dB, Test dB $\geq 12$ dB, Train Duration $\geq$ 6 sec., and Test Duration $\geq$ 3 sec. Consequently, the confidence for the validation experiments which fell into this region are displayed in Figure 4.1 with the resulting error in confidence given in Table 4. In Table 4, the addition of the model confidence reduces the error in the predicted performance by 2.8%.

<table>
<thead>
<tr>
<th>SNR/Duration</th>
<th>Error</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Without Model Conf.</td>
<td>7.0%</td>
<td>5.1</td>
</tr>
<tr>
<td>With Model Conf.</td>
<td>4.2%</td>
<td>3.3</td>
</tr>
</tbody>
</table>

Table 4 Results Incorporating Model Confidence

5. DISCUSSION

In this study a technique was introduced and shown to be effective for the prediction of speaker identification performance. This prediction, based on SNR, duration, and model overlap creates a confidence metric which is applicable to both VQ and GMM systems. Deriving confidence from SNR and duration for a VQ based system yielded an average error of 5.4%. Providing confidence at this level is very usable in many real-world situations. With the incorporation of model overlap the error in confidence can be further reduced.

It is interesting that the confidence measure has some degradation in performance with respect to SNR mismatch between training and testing. As the mismatch becomes less, the error in the confidence measure increases. This is probably due to the fact that the regression analysis has heavily favored the highest mismatch conditions and therefore better modeled these conditions. If a greater balance of data points was used for regression this bias should be minimized. Future research endeavors should include the derivation and validation of this method for non-homogeneous datasets (i.e. speakers whose training data is varied in SNR and duration).

6. REFERENCES