Phonetic Refraction for Speaker Recognition

Mary A. Kohler\textsuperscript{1}, Walter D. Andrews\textsuperscript{1}, Joseph P. Campbell\textsuperscript{2}, Jaime Hernández-Cordero\textsuperscript{1}

\textsuperscript{1}Department of Defense
\textsuperscript{2}M.I.T. Lincoln Laboratory

\textcolor{blue}{m.a.kohler@ieee.org, waltandrews@ieee.org, j.campbell@ieee.org, jhernandez@ieee.org}

Abstract

This paper describes a newly realized high-performance speaker recognition system and examines methods for its improvement. Innovative experiments early this year showed that phone strings are exceptional features for speaker recognition. The original system produced equal error rates less than 11.5\% on Switchboard-I audio files. Subsequent research indicates that the equal error rate can be nearly halved by improving the feature extraction and score fusion methods. Pre-processing the speech files to remove cross-talk, improved techniques for combining scores, and gender-specific phone models each reduce the error rates significantly.

1 Introduction

Pronunciation is an elemental factor for human recognition of speakers. Converting the process by which humans recognize speakers to repeatable machine techniques is a challenging task that has not been successfully attempted, until now. By capturing phone sequences and using them to examine the acoustic phonetic details of different speakers, we can detect and exploit differences in pronunciation.

We develop a speaker-recognition system based only on phonetic sequences instead of the traditional acoustic feature vectors. Although the phones are generated based on the acoustic feature vectors, the recognition is performed strictly from the phonetic sequences created by the phone recognizer(s).

Our phonetic speaker recognition approach relies on phonetic recognizers in several languages to capture phone sequences, which are then used for modeling and recognizing speakers. By processing the speech files with phone recognizers of different languages, we produce refracted phonetic sequences that provide complementary information. Combining phone sequences from several languages not only provides improved performance and robustness, but also provides a degree of language independence similar to that of acoustic approaches.

2 NIST Extended Data Task

All the experiments described in this paper use data from the NIST 2001 Speaker Recognition Evaluation Extended Data Task. NIST’s purpose in creating this task was to promote the exploration and development of new approaches to the speaker recognition challenge, such as the idiolectal characteristics reported in [3] that require larger amounts of training data than provided in previous evaluations.

For the 2001 evaluation, the entire Switchboard-I corpus was prepared for the Extended Data Task. Along with the audio data, NIST provided both Dragon System’s automatic speech recognition transcriptions, and manual transcripts from the Institute for Signal and Information Processing. Both sets of transcripts were available for the entire corpus. All forms of data were permitted for training speaker models either alone or in combination.

The speaker model training data consisted of one, two, four, eight, and sixteen conversations. NIST employed a jackknife approach to rotate through the training and testing conversations to insure an adequate number of tests. Table I provides a breakdown, based on the number of training conversations, of the NIST Extended Data Task.

<table>
<thead>
<tr>
<th>Number of Training Conversations</th>
<th>Number of Unique Speakers</th>
<th>Number of Test Conversations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>483</td>
<td>16429</td>
</tr>
<tr>
<td>2</td>
<td>442</td>
<td>15363</td>
</tr>
<tr>
<td>4</td>
<td>385</td>
<td>13777</td>
</tr>
<tr>
<td>8</td>
<td>273</td>
<td>10377</td>
</tr>
<tr>
<td>16</td>
<td>57</td>
<td>2696</td>
</tr>
<tr>
<td>Total</td>
<td>483</td>
<td>58642</td>
</tr>
</tbody>
</table>

For testing, the same options were available as in training. The recognition feature could be computed from either the acoustic data, the transcriptions or a combination of both. The number of test conversations for each set of training conversations is provided in Table I. The test set contains matched handset and mismatched handset conditions as well as a few cross-gender trials.

NIST provided data in two-channel sphere-formatted audio files. Analysis of the individual conversation sides revealed a considerable amount of cross-talk, which could potentially inhibit successful speaker recognition. Unlike other NIST data, the Switchboard-I files were not processed to remove echo. In [7] we processed the Switchboard-I data through NIST’s echo-canceling software prior to

\textcolor{blue}{Paper presented at the RTO IST Workshop on “Multilingual Speech and Language Processing”, held in Aalborg, Denmark, 8 September 2001, and published in RTO-MP-066.}
speaker recognition. All experiments in this paper use
the echo-cancelled Switchboard-I files. This paper
includes experiments for removing cross-talk in the
phone sequences to potentially improve speaker
recognition performance.

3 Algorithm Description

Phonetic speaker recognition is performed in four
steps. First, a phone recognizer, in the given language,
processes the test speech utterance to produce phone
sequences. Then a test speaker model is generated
using phone n-gram (n-phone) frequency counts. Next,
the test speaker model is compared to the hypothesized
speaker models and the Universal Background Phone
Model (UBPM). Finally, the scores from the
hypothesized speaker models and the UBPM are
combined to form a single recognition score.

The single-language system is generalized to
accommodate multiple-languages by incorporating
phone recognizers trained on several languages
resulting in a matrix of hypothesized speaker models.
The system described in this paper used $M$ speakers,
$P$ phone recognizers and $P$ UBPMs, one UBPM
corresponding to each phone recognizer. Figure 1
shows this multi-language phonetic speaker-
recognition system. The following sections provide
more details for the modeling and recognition process.

3.1 Phone Recognition

The phone recognition process takes advantage of a
phone recognition algorithm that Zissman created for
Parallel Phone Recognition with Language Modeling
(PPRLM) [4]. We chose this recognizer since it was
created solely for phone recognition with no language
model constraints. This algorithm calculates twelve
cepstral ($c_1^\prime$····$c_i^\prime$) and thirteen delta-cestral ($c_1''$····$c_i''$)
features on 20 ms speech frames with 10 ms updates.
The cepstra and delta-cepstra are sent as two
independent streams to fully connected, three-state,
null-grammar HMMs.

The HMMs were trained on phonetically marked
speech from the Oregon Graduate Institute (OGI)
multi-language corpus in six languages: English (EG),
German (GE), Hindi (HI), Japanese (JA), Mandarin
(MA), and Spanish (SP). The corpus was hand-marked
by native speakers in each language using OGI
symbols for two of the languages and Worldbet
symbols for the remainder. The number of phonetic
symbols differs for each language from 27 for
Japanese to 51 for Hindi, and includes one symbol to
represent silence. More information on the corpus and
phone symbols can be found in [8] and [9].

The phone recognizer employs a Viterbi HMM
decoder implemented with a modified version of the
HMM Toolkit. The output probability densities for
each observation stream (cepstra and delta-cepstra) in
each state are modeled as six univariate Gaussian
densities. The output from the HMM recognizer for
each language provides four estimates: the symbol for
the recognized phone, its start time, its stop time, and
its log-likelihood score. For this paper we only used
the recognized phone, although future plans include
exploiting the other estimates.

3.1.1 Gender-specific Phone Recognition

Zissman also created gender-specific phone models in
five languages (EG, GE, JA, MA, SP) using the OGI
multi-language phone-marked speech corpus. The
phone models are identical in format to those
described previously, but the training speech was
constrained by gender. We conducted some
preliminary experiments using gender-specific phone
models to create gender-dependent phone sequences
for speaker recognition.

3.2 Cross-talk Removal

As described previously, the original Switchboard-I
audio files contained excessive cross-talk. Since this
interference was potentially deleterious to speaker
recognition performance, we elected to process the
audio files with software created by MIT Lincoln
Laboratory. Their xtalk tool performs energy based
cross talk and silence detection, producing separate
files containing speech marks and speech.

We separated the conversation sides from the raw
erosito Switchboard-I files prior to xtalk processing. In
a time-saving effort, we chose not to run the time-
intensive phone-recognition software on the xtalk-
processed speech. Instead, we converted the existing
phone files (created from echo-cancelled Switchboard-
I files) using the two-channel speech activity detection

![Figure 1. Multilanguage Phonetic Speaker-
Recognition system](image-url)
(SAD) marks to determine whether the phone should exist. Figure 2 shows this procedure in more detail.

**Figure 2. Cross-talk Elimination Process**

We experimented with several thresholds to determine when a phone should be included. We found that the best speaker recognition performance was achieved by including all phones occupying any portion of a valid speech segment. The converted phone files were processed by the back-end for speaker recognition as described below.

### 3.3 Utterance Delineation

Previous work [6], [7], showed that processing the phone files to include start and stop tags around speech phrases improved speaker recognition performance. The previous algorithm inserted start and stop labels between phrases based on pairs of silence phone labels, i.e., all phones between two silence phone labels were considered an utterance. For example, if the recognized phone sequence was

```plaintext
... sil S oU m i: D & m A n i: sil ...
```

the utterance-tagged speech became

```plaintext
...<start> S oU m i: D & m A n i: <end> <start>...
```

regardless of the length of the silence phones.

In this paper we analyzed the distribution of silence phone durations and experimented with more sophisticated methods to determine where to place utterance breaks, as described later.

### 3.4 Hypothesized Speaker Model

As noted in section 2, a jackknife scheme determined the amount of training and testing data for the extended training task. NIST provided a control file listing hypothesized and test speakers, along with a training and testing conversation list [5]. The list provided training information for one, two, four, eight, and sixteen conversations. As a result, a particular hypothesized speaker will have multiple models for a given test set.

Speaker dependent language models, $H$, are generated using a simple n-phone frequency count for each language and consist of all the unique n-phones with the corresponding frequency counts for a given speaker. Unlike the state-of-the-art GMM-UBM systems, the speaker models are not adapted from the UBPM.

### 3.5 Universal Background Phone Model

The UBPM, $U$, is generated using files determined from the NIST control file (specified in [5]), which provides a list of hypothesized and test speakers for exclusion from the UBPM. All of the conversations for the remaining speakers were used to build the UBPM using n-phone frequency counts. For this paper, each of the six phoneme recognizers has a corresponding independent UBPM.

### 3.6 Test Speaker Model

A test set is specified in the NIST control file for all hypothesized speaker models. The test set contains true speaker trials, impostor trials, matched handset, mismatched handset, and a few cross-gender trials. Once the speech utterance to be tested is processed by the phone recognizer(s), a test speaker model, $T$, is generated using n-phone frequency counts.

Doddington, [3] improved performance by ignoring infrequent word n-grams, i.e. ignoring n-grams occurring less than $c_{\text{max}}$ times. This is also the case with the phonetic approach.

### 3.7 Scoring

Producing a speaker recognition score for a speech file requires not only the calculation of a likelihood score for each of the $P$ phone recognizers, but also the combination of each of the $P$ scores into a single score for comparison with other hypothesized speaker model scores. The following sections describe how the individual model score is calculated and fusion of the $P$ model scores.

#### 3.7.1 Single-language Scoring

For a single-language phonetic speaker-recognition system, the scores from the hypothesized speaker models and the UBPM are combined to form the recognition score $\eta_i$ using a conventional log-likelihood ratio given by
\[
\eta_i = \frac{\sum_{n} \left( w(n) \left[ S_i(n) - B(n) \right] \right)}{\sum_{n} w(n)}
\]

where \( n \) is an n-phone type corresponding to the test speaker model, \( T \), and the sums run over all of the n-phone types in the test segment, \( T \). \( S_i(n) \) represents the log-likelihood score from the \( i^{th} \) hypothesized speaker model, \( H_i \), and \( B(n) \) is the log-likelihood score from the UBPM, \( U \), for the n-phone type, \( n \). The log-likelihood scores \( S_i \) and \( B \) are defined by

\[
S_i(n) = \log \left[ \frac{H_i(n)}{N_H} \right] \quad \text{and} \quad B(n) = \log \left[ \frac{U(n)}{N_U} \right],
\]

where \( N_H \) and \( N_U \) represent the total number of unique n-phone types in the \( i^{th} \) hypothesized speaker model and UBPM, respectively. \( H_i(n) \) and \( U(n) \) represent the number of occurrences of a particular n-phone type, \( n \), in the hypothesized speaker model and UBPM, respectively.

The weighting function \( w(n) \) is based on the n-phone token count, \( c(n) \), and the discounting factor, \( d \). The n-phone token count, \( c(n) \), corresponds to the number of occurrences of a particular n-phone type \( n \) in the test speaker model, \( T \). The weighting function, which could be made language dependent, is given by

\[
w(n) = c(n)^{1-d}.
\]

The discounting factor, \( d \), has permissible values between 0 and 1. When \( d = 1 \) a complete discounting occurs, resulting in \( w(n) = 1 \). This gives all n-phone types the same weight regardless of the number of occurrences in the test speaker model, \( T \). When \( d = 0 \), all n-phone types are weighted by their corresponding token count in the test speaker model, \( T \).

3.7.2 Multiple-language Scoring

In [7], the scores from each of the single-language phonetic speaker-recognition systems were fused by a simple linear combination. Subsequent experiments revealed that more sophisticated techniques for combining the individual language scores improved phonetic speaker recognition performance. We used the LNKnet tool developed by MIT’s Lincoln Laboratory to experiment with several different classification techniques using vectors of the individual language scores as features.

The Gaussian mixture classifier showed the most promising results using either expectation-maximization binary split or K-means for clustering. Both clustering algorithms used eight mixtures, a grand/class full covariance matrix, and tied mixture components.

4 Results

The preceding sections described several experiments intended to improve the speaker recognition performance described in [7]. The results from these experiments are presented below. All detection estimation tradeoff (DET) curves shown are for systems trained on eight conversations with triphone models \((n = 3)\), complete discounting \((d = 1)\), and ignoring n-phones that occur less than 1,000 times, \((c_{\min} = 1000)\). Unless noted otherwise, individual scores from six languages \((P = 6)\) were linearly combined with equal weights to calculate the final phonetic speaker recognition score.

4.1 Utterance Delineation

Analysis of the distribution of silence phone lengths using duration information calculated from the phone recognizer output showed that most of the silences were of short duration (less than 200 ms). The existing approach, which used silences of any duration to mark an utterance, seemed inefficient. Since the goal of utterance delineation was to accurately separate speech phrases, the best approach seemed to use only long silences as utterance separators.

We experimented with several thresholds for minimum silence duration to denote utterances, from 300 ms to 1.2 s. When we compared the speaker recognition performance of the more discriminatory techniques with the simple, naïve approach, we found that speaker recognition performance did not improve with the more complex methods.

4.2 Two-channel Speech Activity Detection

We performed two speech activity detection experiments to determine the optimal method for removing cross-talk. In one experiment, the two-channel SAD processing was performed on the original Switchboard-I files. In the other experiment, two-channel SAD processing was performed on the echo-cancelled Switchboard-I files. For both experiments, we converted the existing phone sequences created from echo-cancelled Switchboard-I files, as shown in Figure 2.

Figure 3 and Figure 4 show DET curves of speaker recognition performance for phones processed with two-channel SAD experiments and for phones processed only on echo-cancelled files. Figure 3 shows performance using only an English phone recognizer, and Figure 4 shows performance using six phone recognizers. On both figures, the solid line marks performance for the experiment in which the original Switchboard-I files were used to create two-channel
SAD marks. The dotted line shows performance for the experiment in which the echo-cancelled Switchboard-I files were used to create two-channel SAD marks, and the dash-dot line indicates performance without two-channel SAD processing. The boxes on each plot demarcate the 90% confidence interval around the equal error rate (EER) based on the number of target and non-target trials.

Figure 3. Comparison of Three Audio Processing Techniques, English

The data in Figure 3 and Figure 4 demonstrates that cross-talk removal improves the EER for speaker recognition performance by nearly 2.5% in the English-only case and more than 3.5% in the combined language case. It also shows that using the unprocessed speech files to determine the regions of speech is superior to using echo-cancelled speech to determine these regions. We plan additional audio processing experiments to discover future recognition improvements. The following experiments use this cross-talk removal process to modify the phone sequences.

4.3 Language Fusion

NIST provided six splits of the speech data each containing unique speakers. We used vectors of the P language scores from two of these splits to train the classifiers through LNKnet and two splits to test the classifiers. Figure 5 shows a comparison of the linearly combined phone scores with the Gaussian mixture classifiers. The dotted line and solid line show performance for K-means clustering and expectation-maximization clustering of the Gaussian mixture models, respectively. The dash-dot line shows performance for a linear combination of the P language scores, and the dashed line shows performance using only an English phone recognizer. The boxes on each plot demarcate the 90% confidence interval around the equal error rate based on the number of target and non-target trials.

Figure 5. Comparison of Two Gaussian Mixture Models for Fusing Language Scores

As Figure 5 shows, both Gaussian mixture classification techniques are superior to linear fusion, but are nearly identical in performance to each other. Additional experiments to determine improved methods for combining the individual language scores are planned for the future.

4.4 Gender-Specific Phone Modeling

NIST provided a table with the Switchboard-I files containing information about each of the audio files including the speaker’s gender. We processed the audio files with the five gender-specific phone recognizers described previously. Figure 6 contains a comparison for speaker recognition performance using
only an English phone recognizer with and without gender-dependent phone models. The solid line shows performance for phone recognition using separate phone models for males and females. The dotted line shows speaker recognition performance using the same phone model regardless of gender. The boxes on each plot demarcate the 90% confidence interval around the equal error rate based on the number of target and non-target trials.

**Figure 6. Comparison of Gender-Dependent and Gender-Independent Phone Recognition for English**

As shown by the data in Figure 6 speaker recognition performance can be improved significantly in the one-language case using phone models to match the speaker’s gender. Further experimental results using phones from five language will be reported in future publications.

### 5 Conclusions

This paper described an innovative technique for speaker recognition using phonetic sequences to capture a speaker’s pronunciation. Four improvements to the basic model were described, most of them exploiting front-end signal processing.

Sophisticated methods for separating speech phrases did not outperform the simple, original method. This was an unexpected result, but its simplicity saves computation.

Elimination of cross-talk provides a significant improvement, especially when the unprocessed speech files are used to determine the areas containing speech. The equal error rate decreased by nearly 4% when the phone strings taken from the echo-cancelled data were post-processed to remove cross-talk.

Gaussian mixture classification for fusing individual language scores is an improvement over a linear combination of the scores. K-means and expectation-maximization binary split clustering perform essentially identically. They each provide over 1.5% improvement in equal error rate.

Gender-specific phone models are superior to using a combined phone model for English phone strings. The equal error rate for the phonetic speaker recognition system decreases by almost 2% when the gender of the speaker is matched to the gender of the phone model.

Additional experiments to improve performance are underway. Improved front-end and back-end processing as well as investigation of other feature sets will be reported as we collect results.

### 6 Acknowledgements

The authors thank George Doddington and John Godfrey for their helpful discussions.

### 7 References