Maximum Likelihood and Maximum A Posteriori Adaptation for Distributed Speaker Recognition Systems

Chin-Hung Sit[1], Man-Wai Mak[1], and Sun-Yuan Kung[2] ⋆

[1] Center for Multimedia Signal Processing
Dept. of Electronic and Information Engineering
The Hong Kong Polytechnic University, China
[2] Dept. of Electrical Engineering, Princeton University, USA

Abstract. We apply the ETSI's DSR standard to speaker verification over telephone networks and investigate the effect of extracting spectral features from different stages of the ETSI's front-end on speaker verification performance. We also evaluate two approaches to creating speaker models, namely maximum likelihood (ML) and maximum a posteriori (MAP), in the context of distributed speaker verification. In the former, random vectors with variances depending on the distance between unquantized training vectors and their closest code vector are added to the vector-quantized feature vectors extracted from client speech. The resulting vectors are then used for creating speaker-dependent GMMs based on ML techniques. For the latter, vector quantized vectors extracted from client speech are used for adapting a universal background model to speaker-dependent GMMs. Experimental results based on 145 speakers from the SPIDRE corpus show that quantized feature vectors extracted from the server side can be directly used for MAP adaptation. Results also show that the best performing system is based on the ML approach. However, the ML approach is sensitive to the number of input dimensions of the training data.

1 Introduction

The use of mobile and hand-held devices has become increasingly popular in recent years. However, inputting text and data to these devices is very time consuming and difficult. While speech input is an ideal alternative for this task, mobile phone users tend to use their phones in noisy environment, making robust speech and speaker recognition a challenging task.

Traditionally, speech signals are encoded at the client-side and coded speech is transmitted to the server. Recognition is then performed at the server-side after the reconstruction and parameterization of the decoded speech. However, it has been found that channel- and codec- distortion can degrade recognition

⋆ This work was supported by The Hong Kong Polytechnic University, Grant No. APE44 and the RGC of HKSAR Grant No. PolyU5129/01E.

performance significantly [1] [2]. To address this problem, the European Telecommunications Standard Institute (ETSI) has recently published a front-end processing standard in which feature vectors are extracted at the client-side and the vector quantised features are transmitted to the server-side for recognition [3] [4]. Since data in the data channel contain the recognition parameters only, codec distortion can be avoided. The technology is commonly referred to as distributed speech recognition (DSR) in the literature.

Research has shown that systems based on the DSR front-end achieve a significantly better performance than those based on recognizing the transcoded speech [5]. In this paper, we investigate the performance of the maximum likelihood (ML) and the maximum a postoriori (MAP) approaches to creating speaker models in the context of distributed speaker verification. We also compare the performance of using 12 MFCCs per speech frame against that of using 12 MFCCs plus 12 delta MFCCs per speech frame.

2 Perturbation of Quantized Vectors

Fig. 1 illustrates the feature-extraction and feature-processing stages of an ETSI-compliance DSR system. Since the feature vectors at the server-side are vector quantized, the distribution of the quantized vectors is discrete. As a result, it is difficult to use the maximum-likelihood approach (based on the EM algorithm [6]) to train a Gaussian mixture model whose output represents a continuous density function to fit the quantized data. To overcome this problem, we propose to add zero-mean, random vectors to the quantized MFCCs to produce the training vectors. Specifically, the training feature vectors $\mathbf{u}_t$'s are obtained by

$$\mathbf{u}_t = Q(\mathbf{x}_t) + \boldsymbol{\eta}_t$$

where $Q(\cdot)$ and $\mathbf{x}_t$ represent the quantization operation and the unquantized vectors respectively, $\boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma})$ are zero mean Gaussian vectors, $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_D]^T$ represents the standard deviation of $\boldsymbol{\eta}_t$, and D is the dimension of the feature vectors.

The values of σ_j's are estimated as follows. An index table $\{m_{kt}\}$ is built using the unquantized feature vectors $\mathbf{x}_t^{(b)}$ extracted from background speakers during the training phase. More specification, we have

$$m_{kt} = \arg\min_{i \in \{1,2,\ldots,S\}} \|\mathbf{x}_t^{(b)} - \mathbf{v}_i\|, \quad k = 1, 2, \ldots, N \text{ and } t = 1, 2, \ldots, T_k \quad (1)$$

where S is the codebook size (= 64 in the ETSI standard) with code vectors $\{\mathbf{v}_i; i = 1, \ldots, S\}$, T_k is the total number of training vectors from background speaker k, and N is the total number of background speakers. Let us denote $\boldsymbol{\sigma}_k^{'} = \left[\sigma_{k,1}^{'}, \sigma_{k,2}^{'}, \ldots, \sigma_{k,D}^{'}\right]^T$ as the standard deviation vector corresponding to speaker k. The components of $\boldsymbol{\sigma}_k^{'}$ are found by:

$$\sigma_{k,j}^{'} = \sqrt{\frac{1}{T_k} \sum_{t=1}^{T_k} \left(v_{m_{kt},j} - x_{t,j}^{(b)}\right)^2}, \quad j = 1, 2, \ldots, D \text{ and } k = 1, 2, \ldots, N \quad (2)$$

where $v_{m_{kt},j}$ and $x_{t,j}^{(b)}$ are respectively the j-th component of $\mathbf{v}_{m_{kt}}$ and $\mathbf{x}_t^{(b)}$ and D is the dimension of the feature vectors $\mathbf{x}_t$. Finally, $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_D]^T$ is calculated by:

$$\boldsymbol{\sigma} = \frac{\alpha}{N} \sum_{k=1}^{N} \boldsymbol{\sigma}_k^{'} \quad k = 1, 2, \ldots, N \tag{3}$$

where α is a scaling factor which is determined using enrollment data. Note that we can get access to the unquantized feature vectors derived from the background speakers but not from the client speakers. This is because for client speakers, the server can only extract quantized vectors $Q(\mathbf{x}_t)$ from the DSR bit-stream. As background speakers' speech can be obtained from pre-recorded speech copora, we are always able to implement the DSR front-end in software and obtain the unquantized spectral vectors.

3 Experiments

3.1 Corpus and Features

The SPIDRE corpus, which is a subset of the Switchboard corpus, was used to evaluate the speaker features extracted from different stages of the DSR front-end. SPIDRE consists of 45 target speakers (23 males and 22 females) and 100 non-target speakers. Each utterance contains about 5 minutes speech. There are 4 sessions (conversations) for each target speaker with a total of 3 different telephone numbers (handsets). Each Speaker uses the same handsets in Session 1 and Session 2 while they use a different handset in Session 3 and Session 4. This arrangement allows us to investigate the speaker verification performance under handset matched and handset mismatch conditions.

A collection of all target speakers in the speaker set was used to train a 128-center GMM background model. For each speaker, we trained a personalized GMM to model the characteristics of his/her own voice using the utterances in Session 1. Instead of using the silence detection facility in the ETSI standard, silence was removed by applying the word transcription files provided by SPIDRE.

Two sets of feature vectors, an *unquantized* set and a *quantized* set, were extracted from different stages of the front-end processor of the ETSI standard (see Fig. 1). More specifically, the unquantized set was extracted before feature compression in the terminal front-end while the quantized set was extracted just after server feature processing. To have a better comparison, speaker recognition was performed under three conditions shown in Table 1.

3.2 Speaker Models

The speaker models were created by two different approaches. In one set of experiments, a maximum likelihood (ML) approach based on the EM algorithm [6] was applied to train the speaker models, while in another set, maximum a posteriori (MAP) adaptation [7] was applied. We have also investigated the performance of using 12 MFCCs (c_1 to c_{12}) and 12 MFCCs plus their first-order derivatives as features. Note that the ETSI standard specifies a total of 39 coefficients ($c_1, c_2, \ldots, c_{12}, \{\ln E \& c_0\}$, and their first- and second-derivatives) per

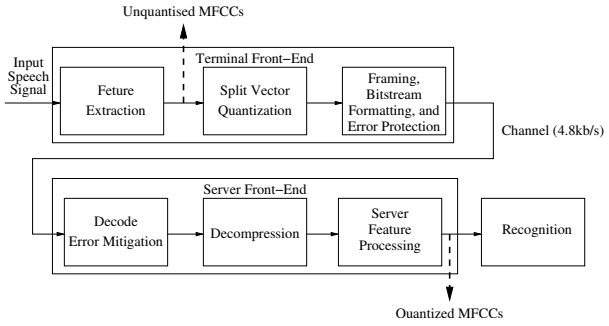

Fig. 1. Feature extraction and processing in an ETSI-compliance DSR system.

	Training	Verification
Condition A	Unquantized	Unquantized
Condition B	Unquantized	Quantised
Condition C	Quantised	Quantised

Table 1. Feature sets for training the speaker models and for verification. "Unquantized" denotes the MFCCs before vector quantization in the client, and "quantized" denotes the vector quantized MFCCs extracted from the bit-stream in the server-side (see Fig. 1).

speech frame. As energy is not speaker-dependent in text-independent speaker recognition, we did not use the energy and its derivatives as features.

For the ML-based speaker models, a personalized 128-center GMM speaker model was trained for each speaker using 12-dimensional MFCCs feature vectors. Due to the curse of dimensionality, however, a 64-center GMM model was trained when 24-dimensional MFCCs feature vectors (MFCCs + delta MFCCs) were used. For the training phase of Condition C shown in Table 1, since the feature vectors were vector quantized, random noise were added to the quantized MFCCs in order to train the Gaussian mixture speaker models. These noise vectors were estimated using the method described in Section 2. The value of α was determined empirically using training data. It was found that $\alpha = 0.26$ gives the lowest EER on the training data.

Fig. 2 shows the projection of the unquantized, quantized, and noise-added feature vectors on the c1-c2 plane. Evidently, the noise vectors are able to make the distribution of the quantized vectors similar to that of the unquantized vectors, which facilitates the ML training of speaker models using the EM algorithm.

For the MAP-adapted speaker models, we adapt a 128-center GMM background model to form a personalized, 128-center GMM speaker model for each speaker in order to fit the quantized feature vectors of that speaker.

3.3 Verification

As each conversation in SPIDRE contains about 5 minutes of speech (2.5 minutes when silence was excluded), we divided the conversation into non-overlapping

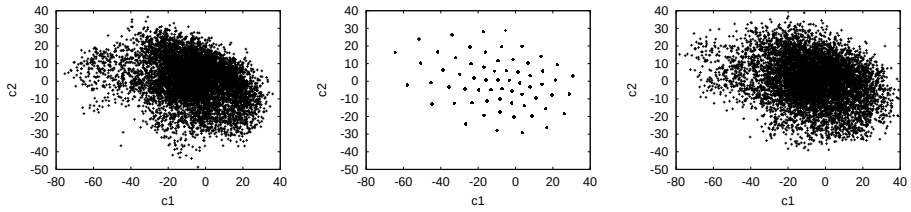

Fig. 2. (a) Projection of unquantized feature vectors on the c1-c2 plane under Conditions A and B. (b) Projection of the quantized feature vectors on the c1-c2 plane before random vectors were added. (c) Projection of quantized training vectors on the c1-c2 plane after adding the random vectors to the quantized vectors in (a).

segments, with each segment contains 200 consecutive feature vectors $\mathbf{Y}$'s. During verification, a vector sequence $\mathbf{Y}$ derived from a claimant's segment was fed to a GMM speaker model ($\mathcal{M}_s$) to obtain a score ($\log p(\mathbf{Y}|\mathcal{M}_s)$), which was then normalized according to

$$S(\mathbf{Y}) = \log p(\mathbf{Y}|\mathcal{M}_s) - \log p(\mathbf{Y}|\mathcal{M}_b) \tag{4}$$

where $\mathcal{M}_b$ represents the 128-center background model. The normalized score $S(\mathbf{Y})$ was compared with a threshold to make a verification decision. Therefore, each verification decision is based on 200 feature vectors, i.e., 2.8 seconds of speech. For ease of comparison, we collect the scores of 45 speakers, each being impersonated by 100 impostors, to compute the speaker-independent equal error rate (EER) and to produce a speaker detection performance curve [8]. Therefore, speaker-independent decision thresholds were used, and for each session in an experimental setting, there were roughly 3,375 client speaker trials (45 client speakers $\times$ around 75 segments per conversation) and 337,500 impostor attempts (100 impostors per client speaker $\times$ 45 client speakers $\times$ 75 segments per conversation).

4 Results

The experimental results are summarized in Table 2 and the DET curves corresponding to Condition C are plotted in Fig 3. All error rates are based on the scores of 45 genuine speakers and 100 impostors. The conversations in Session 2 of SPIDRE were used to produce the results for the handset matched cases, whereas those in Sessions 3 and 4 were used to produce the results for the handset mismatched cases.

For 12-dimensional MFCCs, it is evident from Table 2 that using the technique of maximum likelihood for enrollment generally gives better performance as compared to using MAP adaptation. This is especially obvious in Condition C, as shown in the DET curve (Fig. 3), where both training and verification parameters were extracted from the server side. Compared to MAP adaptation,

	12 MFCCs				12 MFCCs + 12 ΔMFCCs			
	ML		MAP		ML		MAP	
Condition	handset matched	handset mismatched	handset matched	handset mismatched	handset matched	handset mismatched	handset matched	handset mismatched
A	15.63	20.95	15.96	22.83	22.15	20.87	15.47	18.98
B	15.42	21.10	15.39	22.73	22.41	20.98	15.78	19.29
C	**12.15**	18.61	15.20	22.43	34.07	32.78	16.15	19.09

Table 2. Equal error rates (in %) under 3 different training and verification conditions using 2 training approaches with 12-dimensional or 12-D MFCCs plus 12-D ΔMFCCs as features. Refer to Table 1 for the definition of Conditions A, B and C.

the maximum-likelihood approach achieves a 20.07% improvement in the handset matched case and 17.03% improvement in the handset mismatched case.

When both 12-dimensional MFCCs and 12-dimensional delta MFCCs were used as features, 64 centers were found to be just right for maximum likelihood training. This means that setting the number of centers to 64 gives the minimum EER with trainable speaker models. However, as shown in Table 2, when compared to the MAP approach with 12-dimensional MFCCs plus its 12 first-order derivatives, the overall performance of the maximum likelihood approach is poorer. The EER is even the worst in Condition C. This may be due to the curse of dimensionality problem. As a result, the MAP approach is better than the maximum likelihood approach when 12-dimensional MFCCs plus 12-dimensional delta MFCCs were used. It is of interest to see whether the delta MFCCs contain

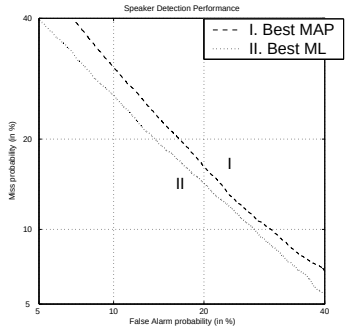

Fig. 3. DET curves for maximum likelihood (ML) and maximum a posteriori (MAP) adaptation under Condition C. Curve I represents result using ML with 12-D MFCCs for all sessions (the best performing ML) and curve II represents result using MAP with 12-D MFCCs plus 12-D Delta MFCCs for all sessions (the best performing MAP).

speaker information and help improve the performance of speaker verification. We may observe this by comparing the performance achieved by the MAP approach using 12-dimensional MFCCs plus 12-dimensional delta MFCCs against the one achieved by the same approach but using 12-dimensional MFCCs only. Table 2 (Columns 4-5 and Columns 8-9) clearly shows that the extra information

provided by the delta MFCCs can lower the EER under handset mismatched conditions and maintain the error rates under handset matched conditions. However, it is surprising to see that the maximum likelihood approach cannot fully utilize the speaker information provided by the delta MFCCs, as evident by Column 7 of Table 2. This may be due to the curse of dimensionality problem as the EM algorithm fails to find a solution in the maximum likelihood approach when the number of Gaussian is larger than 64.

Readers should bear in mind that only Condition C reflects the practical situation in DSR systems. Fig. 3 shows the detection error tradeoff curves under this practical situation. In the figure, the scores of all sessions (Sessions 2 to 4) were aggregated. Therefore, this DET plot shows the average performance under both handset matched and handset mismatched conditions. Evidently, the maximum likelihood approach using 12-dimensional MFCCs is better than the MAP approach using 12-dimensional MFCCs plus 12-dimensional MFCCs.

5 Conclusions

Features extracted from different stages of the ETSI-DSR front-end in the context of distributed speaker verification have been evaluated. Both maximum likelihood and maximum a posteriori adaptation have been applied to create speaker-dependent models. Results show that maximum a posteriori adaptation using 12-dimensional MFCCs plus their first-order derivatives can generally reduce the error rates. However, the maximum likelihood approach favors using 12 MFCC only because of the curse of dimensionality problem.

References

1. S. Euler and J. Zinke, "The influence of speech coding algorithms on automatic speech recognition," in *Proc. ICASSP*, 621-624, p. 1994.
2. B. T. Lilly and K. K. Paliwal, "Effect of speech coders on speech recognition performance," in *Proc. ICSLP*, Oct 1996, vol. 4, pp. 2344–2347.
3. D. Pearce, "Enabling new speech driven services for mobile devices: An overview of the ETSI standards activities for distributed speech recognition front-ends," in *AVIOS 2000: The Speech Application Conference*, 2000.
4. ETSI ES 202 050 V1.1.1 (2002-10), *Speech Processing, Transmission and Quality Aspects (STQ); Distributed Speech Recognition; Advanced Front-end Feature Extraction Algorithm; Compression Algorithms*, Oct 2002.
5. H. Kelleher, D. Pearce, D. Ealey, and L. Mauuary, "Speech recognition performance comparison between DSR and AMR transcoded speech," in *Proc. ICSLP'02*, 2002, pp. 1873–1876.
6. A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. of Royal Statistical Soc., Ser. B.*, vol. 39, no. 1, pp. 1–38, 1977.
7. D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.
8. A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of detection task performance," in *Proc. Eurospeech '97*, Rhodes, Greece, 1997, pp. 1895–1898.