TELEPHONY TEXT-PROMPTED SPEAKER VERIFICATION USING I-VECTOR REPRESENTATION

Hossein Zeinali[1], Elaheh Kalantari[2], Hossein Sameti[1], Hossein Hadian[1]

Department of Computer Engineering, Department of Electrical Engineering
Sharif University of Technology
zeinali@ce.sharif.edu, e_kalantari@kish.sharif.edu, sameti@sharif.edu, hadian@ce.sharif.edu

ABSTRACT

I-vectors have proved to be the most effective features for text-independent speaker verification in recent researches. In this article a new scheme is proposed to utilize i-vectors in text-prompted speaker verification in a simple while effective manner. In order to examine this scheme empirically, a telephony dataset of Persian month names is introduced. Experiments show that the proposed scheme reduces the EER by 31% compared to the state-of-the-art State-GMM-MAP method. Furthermore it is shown that using HMM instead of GMM for universal background modeling leads to 15% reduction in EER.

Index Terms— I-Vector, Text-Prompted, Speaker Verification, Telephony Dataset, GMM, HMM

1. INTRODUCTION

Over recent years, significant improvement has been achieved for text-independent speaker verification. JFA, proposed as a method to compensate channel variability has proved more effective than other conventional GMM-UBM based methods. The more recent method of using i-vector in total variability space together with Probabilistic Linear Discriminant Analysis (PLDA) improved scoring significantly and is currently the state-of-the-art method. In the last few years, many of newly proposed methods for text-independent speaker verification have been adopted in text-dependent speaker verification too.

In [1], in addition to introducing the RSR2015 text-dependent dataset, several methods are evaluated. It notes that lack of sufficient data makes training of the i-vector extractor impossible. Therefore, they use the extractor trained using the NIST telephony dataset. In all cases, the i-vector based methods result in a lower accuracy than the baseline method.

In [2], Kenny et al. address the text dependent speaker verification problem using Joint Factor Analysis (JFA). They train the Universal Background Model (UBM) using background part of RSR2015 dataset and demonstrate that adapting this model for each phrase achieves the best results. In contrast the authors in [3] use phrase-dependent PLDA transforms instead of phrase-dependent UBMs and show empirically that this technique increases performance compared to method proposed in [4].

It is shown in [5] that text-independent background models can be employed in text dependent tasks to give results close to the case where a UBM is trained on text dependent data, by adapting the models using text dependent data.

Most of the work in the literature is devoted to text-dependent conditions and less attention has been paid to text-prompted case. The method proposed in [6] is the state-of-the-art in this field. In this method, for each possible 4-tuple sequence a mean-supervector is extracted to which a Nuisance Attribute Projection (NAP) projection is applied to remove the channel effect. Finally the test score is computed using SVM. It is then shown empirically that this method improves over previous methods and therefore is implemented and evaluated here.

In this article, a new scheme of using i-vectors in text-prompted speaker verification is proposed. It is suggested in this scheme that a separate i-vector extractor be trained for each word (i.e. month name) and the scores be combined at the end. Besides, using Hidden Markov Model (HMM) instead of Gaussian Mixture Models (GMM) as the UBM is examined. This scheme is evaluated on a dataset of month names described in the following sections.

The rest of this paper is organized as follows: Section 2 describes the dataset used for evaluations. The baseline and the State-GMM-MAP methods are briefly described in Section 3. In Section 4, i-vector extractor and LDA are briefly explained. The proposed scheme is explained in Section 5 and evaluations are presented in Section 6. Finally the conclusions are derived in Section 7.

2. DATASET

The dataset used in this research was collected for the purpose of text-prompted speaker verification over telephony channels. This dataset was recorded in uncontrolled environments such as home, office, streets and in other public areas. The recordings were in two channels of landline and cellphone. In order to assure integrity of the recorded data, each file was double checked. Speakers

consisted of both genders and included various accents and ages to cover the Persian language completely. Unlike existing datasets which often use digits as prompt texts [1], Persian month names were used. The main advantage is that the month names are longer than digits. Similar to English, Persian month names are approximately twice as long as digits (in terms of phones and utterance duration) and usually the month names contain more vowels than digits which results in better discrimination of speakers.

This dataset is divided to three standard sets. The first is the development set consisting of 164 speakers, which are distinct from the speakers in train and test sets. This set is used for background modeling. Each speaker in this set has repeated month names successively for several times, each of which is saved in a separate file. The second is the train set which includes 26 speakers. Each speaker repeats the sequence of month names 4 times which are all in either cellphone or landline channel. The last is the test set consisting of a total of 44 speakers including the speakers from the train set. To investigate the effect of aging on speaker verification performance, the test set was recorded more than one year after the train set was recorded. In addition, the recording channel was changed for certain speakers in the test set to examine the channel effects. It should be pointed that segmentation of this dataset was done automatically by Viterbi alignment.

Since the dataset is used for commercial purposes, it won't be accessible publicly. However, the features extracted from the whole segmented dataset are freely available for everyone to use and can be downloaded [7].

3. BASELINE SYSTEMS

3.1. HMM based (Baseline)

In the baseline method, for each month, an HMM is trained using the development set and is used as $UBM_m\{m = 1 \dots 12\}$. Then for each of the speakers 12 models are built from the UBMs by MAP adaptation of the Gaussian means using the training data for that speaker. Finally, the test score for s'th speaker is computed using the log-likelihood ratio as follows:

$$score_s(\boldsymbol{X}_t) = \sum_{j=1}^{M} \alpha_{m_j} \left(\log\, p\left(\mathbf{X}_{\mathrm{t_j}} \middle| \lambda_{s_{m_j}}\right) - \log\, p\left(\mathbf{X}_{\mathrm{t_j}} \middle| \lambda_{UBM_{m_j}}\right)\right), \quad (1)$$

where, M shows the number of months in a test case, m_j shows the j'th month in the sequence, $\mathbf{X}_{\mathrm{t_j}}$ shows the feature vectors for j'th month, and alpha is the scaling factor to combine the scores from different months. In this equation for each model, log-likelihood is computed using the Viterbi method and is normalized using ZT-norm separately.

3.2. State-GMM-MAP-Supervector

Two methods are proposed in [6] based on mean-supervectors and SVM for text-prompted speaker verification. In the first method, the mean-supervector is computed for each month using MAP adaptation while in the second method, the JFA adaptation is used. According to the reported results, the first method performs better in the channel mismatch condition, and therefore this method was chosen for comparison.

In this method, as in the baseline method, after segmenting all the utterances, a GMM is trained for each month using the development set. Then for each speaker and for every repetition of the months a mean-supervector is calculated. Then for each possible 4-tuple sequence of digits a subsystem is built. This is done by concatenating supervectors of sorted digits in succession to make a larger supervector. To compensate channel effects the supervectors are projected to a new space using NAP. Then using these vectors and the supervectors of the development set as imposters, a linear SVM is trained for each speaker. Finally, the score for a test utterance is defined as the distance of the extracted supervector from that utterance to the corresponding SVM hyperplane. The scores are normalized using ZT-norm.

4. I-VECTOR

4.1. i-vector extractor

An i-vector extractor is a system which converts a speech utterance with arbitrary duration to a fixed length vector [8]. For this purpose, Baum-Welch statistics must be extracted from a UBM which can be a GMM or HMM. In this system, mean supervector for an utterance can be modeled as follows:

$$\boldsymbol{M} = \boldsymbol{m} + \boldsymbol{T}\boldsymbol{x} \quad (2)$$

where, $\boldsymbol{M}$ is the speaker dependent supervector, $\boldsymbol{m}$ is the UBM supervector, $\boldsymbol{T}$ is factor loading low-rank matrix and $\boldsymbol{x}$ is a latent variable with standard normal distribution. For each utterance, i-vector $\boldsymbol{w}$ is the MAP point estimate of the latent variable $\boldsymbol{x}$.

4.2. Linear Discriminant Analysis (LDA)

LDA is one of the techniques widely used for reducing the channel effects in speaker verification [8]. LDA aims at reducing intra-class variance while increasing the discrimination between classes. The objective function for LDA is as follows:

$$J(\boldsymbol{v}) = \frac{\boldsymbol{v}^t \boldsymbol{S}_b \boldsymbol{v}}{\boldsymbol{v}^t \boldsymbol{S}_w \boldsymbol{v}}, \quad (3)$$

where, $\boldsymbol{S}_b$ shows between-class variance and $\boldsymbol{S}_w$ shows within-class variance which are calculated using following relations:

$$\boldsymbol{S}_b = \sum_{s=1}^{S} (\overline{\boldsymbol{w}_s} - \overline{\boldsymbol{w}})\,(\overline{\boldsymbol{w}_s} - \overline{\boldsymbol{w}})^t\,, \tag{4}$$

$$\boldsymbol{S}_w = \sum_{s=1}^{S} \frac{1}{N_s} \sum_{n=1}^{N_s} (\boldsymbol{w}_s^n - \overline{\boldsymbol{w}_s})\,(\boldsymbol{w}_s^n - \overline{\boldsymbol{w}_s})^t\,, \tag{5}$$

where $\overline{\boldsymbol{w}_s} = \frac{1}{N_s}\sum_{n=1}^{N_s} \boldsymbol{w}_s^n$ is the mean of all samples from speaker s, S is the number of speakers, N_s is the number of samples for speaker s and $\overline{\boldsymbol{w}}$ is overall samples mean (in case of i-vector this mean is zero).

5. PROPOSED SCHEME

In [6] a subsystem is built for each possible M-tuple to make the scoring of test sequences possible. This is inefficient both in terms of memory and computation costs. This problem can be solved by scoring each month after aligning segmented months in test and train utterances. In fact, a subsystem is trained for each month separately. For evaluation, the segmented utterance is scored using the corresponding subsystems and then the M resulting scores are combined linearly. This will reduce the number of models from 220 (combinations of 12 by 3) down to 12. Thus in this paper, the State-GMM-MAP method is implemented in this manner (12 models).

In the proposed scheme, similar to previous methods, first a GMM is trained for each month as the UBM. Then a separate matrix T is trained for each month using (2). After that, using the trained models and T matrices, an i-vector is extracted for each utterance of month for all speakers (i.e. a total of 4*12 i-vectors for each speaker). Finally all i-vectors of a single month are averaged to give a single i-vector. For evaluation, after segmenting the test utterance, i-vectors are extracted for each test month and the score is computed as follows:

$$score_s(\boldsymbol{X}_t) = \sum_{j=1}^{M} \alpha_{m_j} CD\left(\boldsymbol{w}_{s,m_j}, \boldsymbol{w}_{t,j}\right), \tag{6}$$

$$CD\left(\boldsymbol{w}_{target}, \boldsymbol{w}_{test}\right) = \frac{\langle \boldsymbol{w}_{target}, \boldsymbol{w}_{test} \rangle}{\|\boldsymbol{w}_{target}\| \|\boldsymbol{w}_{test}\|}, \tag{7}$$

where, $\boldsymbol{w}_{s,m_j}$ is the i-vector for m_j'th month of the s'th speaker, and $\boldsymbol{w}_{t,j}$ is the i-vector for j-th test month. Alpha factors are the same as the baseline method. The score of each month is normalized separately using the ZT-norm prior to combining the scores using (6). This method of combining separate scores linearly is analogous to improving accuracy by boosting weak learners [9].

Since the temporal order of speech frames is important in text-dependent speaker verification, it is suggested that HMM be used as the UBM and the Baum-Welch statistics be extracted using it. Both GMM and HMM are implemented and evaluated in this article. All implemented source codes are available online [7].

6. EXPERIMENTS

6.1. Experimental Settings

As explained in Section 2, first step in preparation of the dataset is the segmentation which was done using HMMs. Silent parts were omitted from utterances by using a separate model for silence, eliminating the need for a VAD. After segmenting utterances, 19 Rasta-PLP [10] coefficients together with log energy were extracted from 25ms windows every 15ms. CMVN was applied and finally by adding delta coefficients, feature vectors of length 40 were generated.

In the baseline method, for each month an HMM with 8 states and 8 components in each state was used. To adapt the speaker models from UBMs, a single iteration of MAP adaptation with relevance factor of 19 was performed. A 64-component GMM was used in the State-GMM-MAP method and mean-supervectors were obtained using a single iteration of the MAP.

In the proposed scheme for each month a 64-component GMM was used. The dimension of the i-vectors was 175 for all subsystems (Modified version of MSR toolbox [11] was used). After a set of pre-evaluations, dimension of 150 was decided for LDA transform. HMM models used as UBMs were the same as the baseline method.

In all methods, to normalize the scores of each speaker using the ZT-norm, 50 most similar speakers from the development set were used. Since all scores were normalized before combining, alpha factors were assumed equal in all methods.

6.2. Results

Table I shows the results obtained from three different methods. This table reports Equal Error Rate (EER) and MinDCF based on NIST evaluation 2008.

Table I: Comparison results between baseline, State-GMM-MAP and proposed methods.

	EER	**DCF**
Baseline	4.76 %	0.0197
State-GMM-MAP	4.01 %	0.0195
Proposed-HMM-LDA150	**2.76 %**	**0.0173**

According to Table I the proposed scheme outperformed the State-GMM-MAP method and reduced the EER by 31%. Furthermore this table shows that the DCF

was decreased significantly by using this scheme. It should be noted that the dimensionality of the i-vectors in the proposed scheme is 150 versus the mean-supervectors' dimensionality of 2560 which is a remarkable reduction in computation and memory costs.

Figure 1 compares the DET curves between baseline, State-GMM-Map and proposed methods.

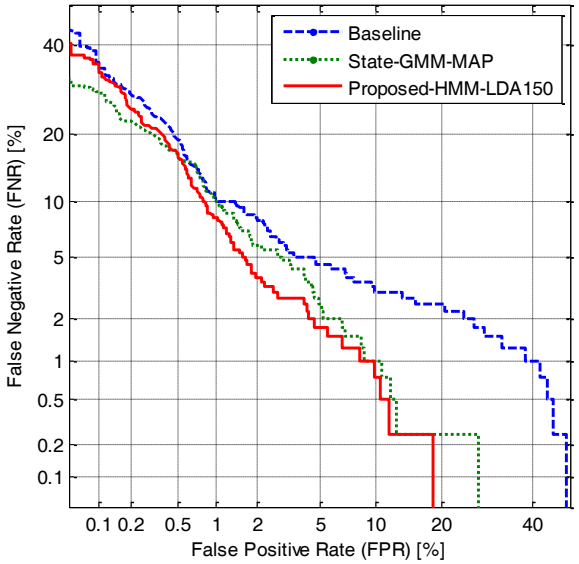

Figure 1: DET curves comparison between baseline, State-GMM-MAP and proposed methods.

As mentioned in section 2, both GMM and HMM were used as UBMs. Table II compares the results for them.

Table II: Comparison results between using GMM or HMM as the UBM.

	EER	**DCF**
Proposed-GMM	3.79 %	0.0227
Proposed-GMM-LDA150	3.26 %	0.0201
Proposed-HMM	3.51 %	**0.0171**
Proposed-HMM-LDA150	**2.76 %**	0.0173

The order of frames is important in speech processing and contains useful information not only for speech recognition but also on how words are uttered by speakers which makes it valuable for speaker verification. Thus, it was anticipated that using HMM would improve the results. In fact, Table II supports this justification and shows a 15% relative reduction in EER by using HMM compared to GMM.

By comparing results of GMM with the baseline results, it can be seen that although this method has improved EER, it has resulted in higher DFC values. On the other hand, the HMM method has improved both EER and DCF which is another advantage of HMM.

7. CONCLUSION

In this work, a dataset of telephony utterances of Persian month names that previously collected has been introduced. In addition, a new scheme for using i-vectors in text-prompted speaker verification has been proposed. According to this scheme, instead of using a single universal extractor, a separate extractor is used for each word (month). This scheme has resulted in a 31% relative reduction in EER in comparison to State-GMM-MAP method. Furthermore it has been shown that using HMM rather than GMM leads to a 15% reduction in EER. It should be emphasized that in practical systems, speed is of great concern; the proposed scheme has managed to increase speed remarkably by reducing computations in test time.

8. REFERENCES

[1] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Text-dependent speaker verification: Classifiers, databases and RSR2015," *Speech Communication,* vol. 60, pp. 56-77, 2014.

[2] P. Kenny, T. Stafylakis, J. Alam, P. Ouellet, and M. Kockmann, "Joint Factor Analysis for Text-Dependent Speaker Verification," in *Proc. Odyssey Speaker and Language Recognition Workshop, Joensuu, Finland*, 2014.

[3] T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. Kockmann, and P. Dumouchel, "I-Vector/PLDA Variants for Text-Dependent Speaker Recognition,". http://www.crim.ca/perso/patrick.kenny.

[4] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Phonetically-constrained PLDA modeling for text-dependent speaker verification with multiple short utterances," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013, pp. 7673-7677.

[5] P. Kenny, T. Stafylakis, J. Alam, P. Ouellet, and M. Kockmann, "In-Domain versus Out-of-Domain Training for Text-Dependent JFA," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[6] S. Novoselov, T. Pekhovsky, A. Shulipa, and A. Sholokhov, "Text-dependent GMM-JFA system for password based speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 729-737.

[7] http://ce.sharif.edu/~zeinali.

[8] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on,* vol. 19, pp. 788-798, 2011.

[9] A. Subramanya, Z. Zhang, A. C. Surendran, P. Nguyen, M. Narasimhan, and A. Acero, "A generative-discriminative framework using ensemble methods for text-dependent speaker verification," in *Acoustics, Speech and Signal Processing, 2007.*

ICASSP 2007. IEEE International Conference on, 2007, pp. IV-225-IV-228.

[10] H. Hermansky and N. Morgan, "RASTA processing of speech," *Speech and Audio Processing, IEEE Transactions on,* vol. 2, pp. 578-589, 1994.

[11] S. O. Sadjadi, M. Slaney, and L. Heck, "MSR Identity Toolbox v1. 0: A MATLAB Toolbox for Speaker Recognition Research," *Speech and Language Processing Technical Committee Newsletter,* 2013.