Interspeech 2018
2-6 September 2018, Hyderabad

Analysis of Language Dependent Front-End for Speaker Recognition

Srikanth Madikeri[1], *Subhadeep Dey*[1,2], *and Petr Motlicek*[1]

[1]Idiap Research Institute, Martigny, Switzerland
[2]Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
srikanth.madikeri@idiap.ch, subhadeep.dey@idiap.ch, petr.motlicek@idiap.ch

Abstract

In Deep Neural Network (DNN) i-vector based speaker recognition systems, acoustic models trained for Automatic Speech Recognition are employed to estimate sufficient statistics for i-vector modeling. The DNN based acoustic model is typically trained on a well-resourced language like English. In evaluation conditions where enrollment and test data are not in English, as in the NIST SRE 2016 dataset, a DNN acoustic model generalizes poorly. In such conditions, a conventional Universal Background Model/Gaussian Mixture Model (UBM/GMM) based i-vector extractor performs better than the DNN based i-vector system. In this paper, we address the scenario in which one can develop a Automatic Speech Recognizer with limited resources for a language present in the evaluation condition, thus enabling the use of a DNN acoustic model instead of UBM/GMM. Experiments are performed on the Tagalog subset of the NIST SRE 2016 dataset assuming an open training condition. With a DNN i-vector system trained for Tagalog, a relative improvement of 12.1% is obtained over a baseline system trained for English.

Index Terms: i-vector, speaker recognition, deep neural networks

1. Introduction

State-of-the-art speaker recognition systems employ the i-vector Probabilistic Linear Discriminant Analysis (PLDA) framework [1]. A conventional implementation uses a Universal Background Model/Gaussian Mixture Model (UBM/GMM) to compute sufficient statistics in order to estimate the speaker model, also known as the identity vector (i-vector). A successful extension of this framework replaces the UBM/GMM with a Deep Neural Network (DNN) based acoustic model (AM) trained for Automatic Speech Recognition (ASR) [2, 3, 4, 5, 6]. Two common techniques exist under this extension. In the first technique, termed DNN i-vector, the AM has acoustically well-defined targets (typically, senones) that replace the components of the UBM/GMM. During AM training, the targets are bootstrapped with alignments from a Hidden Markov Model/Gaussian Mixture Model (HMM/GMM) system. In the other common technique using DNNs in i-vector systems, a Stacked Bottleneck Network (SBN) is trained for acoustic modeling. Bottleneck Features (BNF) are obtained from the SBN, which are then combined with conventional short-term acoustic features such as the Mel Frequency Filterbank Coefficients (MFCC) [7].

In both the above mentioned techniques a labeled corpus is required to train the ASR system. In order to develop systems for benchmark datasets that contain speech only in English, such as the NIST SRE 2010 and 2012, Fisher and Switchboard datasets can be utilized [2, 8, 9]. These datasets contain several thousand hours of transcribed speech data. In matched language conditions, the DNN i-vector performs significantly better than the UBM/GMM i-vector system. However, on datasets such as the NIST SRE 2016 dataset that contains two unseen languages in the evaluation condition, a DNN i-vector system with the AM trained for the English language performs worse than a UBM/GMM i-vector system [10, 11]. This suggests that the UBM/GMM generalizes better than the DNN i-vector. The degradation in the performance of DNN i-vectors may be attributed to phonetic, acoustic and duration mismatch. In [12], a multilingual bottleneck (MLB) system is trained with 14 languages from the BABEL program. Neither of the two evaluation languages, Tagalog (TGL) and Cantonese (YUE), were present in those 14 languages. Once again, the results obtained were not better than the GMM i-vector system.

In this paper, we study the effect of the phonetic mismatch arising due to training a language-dependent DNN to extract posteriors for i-vector modeling. In particular, we address the scenario in which one has access to data to train a low-resource ASR system for a language to be seen during evaluation. We hypothesize that a DNN i-vector system trained on target language (i.e. language that will be seen in the evaluation condition) improves the performance of the DNN i-vector system. The analysis is conducted through text-independent speaker verification experiments on the TGL subset of the NIST SRE 2016 dataset. Tagalog BABEL corpus is used to train the DNN AM. We note that a similar hypothesis was considered in [13]. However, no consistent improvements were shown when using language dependent systems. Two important distinctions in our work are: (1) the use of dataset variability compensation and (2) adaptation of the backend with unlabeled data.

There have been several attempts to improve speaker recognition systems using DNNs. In particular, speaker embeddings proposed in [14] deserves a mention as it

10.21437/Interspeech.2018-2071

targets the same evaluation scenario (NIST SRE 2016). Large amounts of data are generated to train the network for speaker discrimination. However, a DNN i-vector system offers the potential to exploit content information in a speech recording. Thus, our understanding can also be extended to text-dependent speaker verification where DNN i-vectors have been show to be useful [5].

The rest of the paper is organized as follows: in Section 2 the i-vector framework for speaker recognition is introduced. This is followed by a description of the DNN i-vector system and its adaptation to TGL in Section 3. In Section 4, the results of experiments on the NIST SRE 2016 and 2010 datasets are presented.

2. I-vector system

The i-vector extractor projects Gaussian mean supervectors on a low-dimensional subspace called *total variability space* (TVS) [1]. The underlying variability model used for i-vector extraction is

$$\mathbf{s} = \mathbf{m} + \mathbf{T}\mathbf{w}, \qquad (1)$$

where $\mathbf{s}$ is the supervector adapted with respect to a UBM-GMM from a speech recording. The vector $\mathbf{m}$ is the mean of the supervectors usually obtained from the UBM-GMM, $\mathbf{T}$ is the matrix with its columns spanning the total variability subspace and $\mathbf{w}$ is the low-dimensional i-vector representation. In the above model, the i-vector is assumed to have Gaussian distribution with zero mean and unit variance as prior distribution.

Given a sequence of MFCC feature vectors $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t\}$, the first-order statistics ($\mathbf{f}$) are estimated to obtain the i-vector representation. The subvector $\mathbf{f}_c$ of $\mathbf{f}$ is given by

$$\mathbf{f}_c = \Sigma_c^{-\frac{1}{2}} \left(\sum_n \gamma_{n,c} \mathbf{x}_n - \mu_c \right), \qquad (2)$$

where $\mathbf{f} = [\mathbf{f}_1^t, \mathbf{f}_2^t, \ldots, \mathbf{f}_C^t]^t$, C is the number of mixtures in the UBM/GMM, $\boldsymbol{\mu}_c$, $\boldsymbol{\Sigma}_c$ are the mean and covariance matrix of the c^{th} mixture, and $\gamma_{n,c}$ is the posterior for the n^{th} frame of speech with respect to the c^{th} mixture component.

Given the first order statistics, the i-vector is estimated as follows

$$\mathbf{w} = \left(\mathbf{I} + \sum_{c=1}^{C} N_c \mathbf{T}_c^t \boldsymbol{\Sigma}_{\mathbf{c}}^{-\frac{1}{2}} \mathbf{T}_c \right)^{-1} \mathbf{T}^t \boldsymbol{\Sigma}^{-1} \mathbf{f}, \qquad (3)$$

where $\mathbf{T}_c$ is the submatrix of $\mathbf{T}$ for the c^{th} mixture, $\boldsymbol{\Sigma}$ is a block diagonal matrix with each block given by $\boldsymbol{\Sigma}_c$ for $c = 1, 2, \ldots C$ and

$$N_c = \sum_n \gamma_{n,c} \qquad (4)$$

is the effective number of feature vectors assigned to the cluster c. The i-vector estimation equation (Equation 3) is the Maximum a Posteriori estimate of $\mathbf{w}$ assuming Gaussian distribution.

Table 1: *Results of blind evaluation on female subset of the NIST SRE 2016 dataset comparing UBM/GMM and DNN i-vector systems. Results are reported in terms of Equal Error Rate (EER). TGL: Tagalog, YUE: Cantonese.*

System	Language	EER (%)
UBM/GMM i-vector	TGL	14.9
DNN i-vector	TGL	15.7
UBM/GMM i-vector	YUE	5.9
DNN i-vector	YUE	7.8

In [2], it was shown that a DNN trained for ASR can replace the traditional UBM/GMM to obtain γ required for i-vector estimation. The posteriors obtained at the output of the DNN forward pass process are used to compute N_c and $\mathbf{f}$. This technique resulted in large performance gains for speaker verification systems as better alignments are obtained with respect to the UBM components. The results showed that replacing unsupervised training of the UBM components with well-defined acoustic classes can have a significant impact on verification performance.

3. Language dependent DNN

As mentioned in Section 1, the performance of the DNN i-vector system degrades when the language of the DNN and that in the evaluation are mismatched. We present results on NIST 2016 SRE to demonstrate it. Table 1 compares the UBM/GMM and the DNN/i-vector system on the female subset of the NIST SRE 2016 evaluation set. The UBM/GMM i-vector is trained with Fisher English Part I and II, Switchboard Cellular Parts I and II, NIST SRE 2004, 2005, 2006 and 2008. The UBM, LDA and PLDA are all trained on the same data. The PLDA is adapted with unlabeled development data in the NIST SRE 2016 data. Details of features and voice activity detection are given in Section 4. The DNN i-vector system was trained on Fisher English Parts I and II and had 1520 targets (senones).

The results on the evaluation set for female speakers presented in Table 1 demonstrate the degradation in performance, which is contrary to the results observed in matched language conditions. The UBM/GMM system performs 7.6% relatively better than the DNN i-vector on the TGL subset and 24.3% better on the YUE subset.

In this paper, we focus on improving the TGL subsystem by assuming that we have access to a limited amount of labeled data for the language. The Babel Tagalog dataset contains approximately 84 hours of transcribed conversational speech (excluding silence) and thus is not as well-resourced as English. Out of the 84 hours of speech, approximately 48 hours are from female speakers. A DNN based acoustic model is trained with this limited amount of data. In order to understand the advantages of using a language dependent DNN we propose replacing the ENG-DNN (i.e. the DNN trained with Fisher English) with this DNN trained for TGL (TGL-DNN). The data used to train the back-end remains unchanged.

4. Experiments

Speaker verification systems were evaluated on the female subset of the NIST 2016 SRE dataset (SRE2016) and the NIST SRE 2010 (SRE2010) dataset [15]. Only telephone-telephone condition (det5) of the core evaluation are presented for the SRE2010 data. Similarly, only female speakers from the TGL subset of evaluation are considered for scoring the SRE2016 data.

4.1. Feature extraction

The front-end used 20 MFCC features with delta and acceleration parameters, extracted every 10 ms using a window of 30 ms (as used by systems such as [7, 8]). They were further processed through a short term Gaussianization module ([16]) with a context of 300 frames. A DNN based voice activity detector is used, which classifies each frame of audio as either speech or non-speech. The frame-level decisions are then smoothed over 300 ms. All systems presented in this paper use the same feature configuration.

4.2. I-vector baseline

The UBM/GMM i-vector baseline was trained using the following datasets: The NIST datasets - SRE 2004, 2005, 2006, 2008 and 2008 extended, Switchboard Part II and Part III, and Switchboard Cellular Part I and II. A GMM with 2048 components was trained. The i-vector dimension was 500. LDA and PLDA were trained with only the NIST datasets. The setup for LDA and PLDA is consistent for all systems presented in the paper. After LDA, the dimension of the i-vector was reduced to 350.

For the DNN i-vector system, Fisher English Parts I and II were used to train the DNN with 1’520 output states. We term this system ENG-DNN. We use a standard DNN architecture with 6 hidden layers and a final softmax layer. Each hidden layer had 1’024 units with sigmoid activation function. Although it is common for ASR systems to use only 13 MFCC dimensions with delta and acceleration, we preserved the same MFCC configuration for both ASR and i-vector systems. It was observed the Word Error Rate (WER) of ASR systems dropped by $\approx 2\%$ absolute (from 40% to 42%) with the increased number of co-efficients.

In order to exploit unlabeled domain-dependent data for the evaluation set, the PLDA was adapted in an unsupervised fashion using Kaldi [17]. The unsupervised adaptation updates the covariance estimates of the PLDA resulting in domain-dependent back-ends.

All i-vector systems were trained with the implementation in [18] (following [19, 20]). The i-vectors from SRE2016 are forcibly zero-centered prior to evaluation to offset dataset mismatch. This mean is estimated from the unlabeled development data in SRE2016. This data will be referred to as SRE16U.

4.3. Tagalog ASR system

The BABEL Tagalog language pack contains approximately 80 hours of conversational speech to train an ASR system in Tagalog. This training set will be referred to as BTGL. Instead of training an ASR system from scratch, the ENG-DNN trained with the Fisher dataset is adapted to TGL. The final linear layer followed by the softmax layer is retrained using mini-batch Stochastic Gradient Descent. Initially, the targets are bootstrapped by training a HMM/GMM system using triphones. The DNN has 1’530 output units with a WER of 53% on the development set. The DNN/i-vector system based on this acoustic model is termed TGL-DNN. The system was trained with the same MFCC-based feature mentioned previously. While a more common ASR setup for TGL uses Perceptual Linear Prediction (PLP) and pitch based features instead of MFCC, the difference in WER on the development set was only 3% (from 50% to 53%). Therefore, to maintain a homogeneous setup the MFCC-based TGL-DNN was used.

As the amount of data in BTGL is significantly limited compared to the Fisher English corpus, an ASR system with the same amount of data as BTGL is also trained for English. Henceforth, this system will be termed ENG40-DNN signifying that only 40 hours of speech data was used (closely matching 48 hours of data for female speakers in TGL). This helps us observe the effects of the amount of training data (and hence the accuracy of the recognizer) for the DNN.

As an extension, we also compare the results of the three DNNs on the SRE2010 dataset, which contains speech in only one language (English). We demonstrate that the trend of the results obtained on TGL are consistent for English as well.

4.4. Results on SRE 2016

In Table 2, the results with the baseline systems are compared to the DNN i-vector systems using TGL-DNN and ENG40-DNN. Using TGL-DNN clearly benefits speaker verification performance. Replacing ENG-DNN by TGL-DNN and training the i-vector extractor ($\mathbf{T}$) with only BTGL data results in a reduction of EER from 15.7% EER to 13.8% giving a relative improvement of 12.1%. Thus, using language dependent DNNs can certainly bring benefits. Note that only the back-end (LDA and PLDA) was trained with NIST data. The results imply that the phonetic variability is better captured in front-end than in the back-end. PLDA adaptation with SRE16U does not improve the performance further. The TGL-DNN also improves over the UBM/GMM i-vector baseline by $\approx 8\%$ relative.

Table 2: *Comparison of performances of ENG-DNN, TGL-DNN and ENG40-DNN on the TGL subset of the NIST SRE 2016 dataset (SRE2016). ENG-DNN is trained with the entire Fisher dataset, ENG40-DNN is trained with only 40 hours of data, and TGL-DNN is trained with TGL dataset. SRE16U refers to the unlabeled development set in SRE2016.*

System	T-matrix data	PLDA Adaptation	EER
UBM/GMM i-vector	NIST	SRE16U	14.9
DNN i-vector (ENG-DNN)	NIST	SRE16U	15.7
DNN i-vector (TGL-DNN)	BTGL	-	**13.8**
DNN i-vector (TGL DNN)	BTGL	SRE16U	**13.8**
DNN i-vector (TGL-DNN)	NIST + BTGL	-	15.3
DNN i-vector (TGL-DNN)	NIST + BTGL	SRE16U	**13.7**
DNN i-vector (ENG40-DNN)	BTGL	-	16.9
DNN i-vector (ENG40 DNN)	BTGL	SRE16U	17.0
DNN i-vector (ENG40-DNN)	NIST + BTGL	-	17.6
DNN i-vector (ENG40-DNN)	NIST + BTGL	SRE16U	16.6

When the i-vector extractor is trained with NIST and BTGL data, significant improvements are achieved only when adapting the PLDA with SRE16U. One reason is that the two datasets, NIST and BTGL, are imbalanced. Without PLDA adaptation the EER reduces from 15.7% to 15.3%. However, the UBM/GMM baseline is still better by 0.4% absolute. After PLDA adaptation the EER reduces to 13.7% with a relative improvement of 12.7% with respect to the ENG-DNN baseline and 8% relative improvement with respect to the UBM/GMM baseline.

The results for ENG40-DNN system are, as expected, worse than ENG-DNN and TGL-DNN systems. Once again, training with both NIST and BTGL data provides better performance than training with only BTGL and adapting the PLDA with SRE16U. In such a case, an EER of 16.6% is obtained, which is 2.3% relatively better than training the i-vector extractor with only BTGL. The best TGL-DNN system is 17.4% relatively better than the best ENG40-DNN system.

4.5. Results on SRE 2010

The three systems were also evaluated on the SRE2010 dataset, which contains speech samples only in English. All results presented are without PLDA adaptation as the training data is predominantly English. The results on the telephone-telephone subset (det5) of the core conditions are presented in Table 3. The UBM/GMM system has an EER of 2.2%. The DNN i-vector system using ENG-DNN improves relatively by over 54.4%. However, when using ENG40-DNN the EER degrades to 1.7%, which is still relatively 19% better than the UBM/GMM system. As expected, the TGL-DNN performs worse and has an EER of 3.1%. When compared with the UBM/GMM system, the TGL-DNN system is 0.9% worse (absolute). This trend is consistent with the results presented in Table 2 showing that a well-trained language-dependent DNN based front-end can certainly provide consistent improvements over a UBM/GMM i-vector system.

Table 3: *Comparison of ENG-DNN, ENG40-DNN and TGL-DNN based i-vector systems on det5. The results are presented in terms of Equal Error Rate (EER). The i-vector extractors are trained on the TGL+NIST set.*

System	EER (%)
UBM/GMM i-vector	2.2
DNN i-vector (ENG-DNN)	1.0
DNN i-vector (ENG40-DNN)	1.7
DNN i-vector (TGL-DNN)	3.1

5. Conclusions

The effectiveness of language dependent acoustic models for DNN i-vector systems was studied. An acoustic model for Tagalog was trained and speaker verification experiments on NIST SRE 2016 with the resulting DNN i-vector system demonstrated a relative improvement of 12.1% when compared with an acoustic model trained on English, which is a well-resourced language. The improvements in error rates obtained demonstrated the effect of phonetic variability mismatch on the performance of the DNN i-vector system.

6. Acknowledgements

This work was supported by the EU FP7 project ”SPEAKER IDENTIFICATION INTEGRATED PROJECT (SIIP)”.

7. References

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speaker verification,” *IEEE Tran. on Audio, Speech and Language Processing*, pp. 788–798, 2011.

[2] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, “A novel scheme for speaker recognition using a phonetically-aware deep neural network,” in *Acoustics, Speech and Sig-*

nal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, 2014, pp. 1695–1699.

[3] F. Richardson, D. Reynolds, and N. Dehak, "Deep neural network approaches to speaker and language recognition," *Signal Processing Letters, IEEE*, vol. 22, no. 10, pp. 1671–1675, 2015.

[4] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep neural networks for extracting baum-welch statistics for speaker recognition," in *Proc. Odyssey*, 2014.

[5] S. Dey, S. Madikeri, M. Ferras, and P. Motlicek, "Deep neural network based posteriors for text-dependent speaker verification," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5050–5054.

[6] S. Dey, P. Motlicek, S. Madikeri, and M. Ferras, "Exploiting sequence information for text-dependent speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. Ieee, 2017, pp. 5370–5374.

[7] P. Matejka, O. Glembek, O. Novotny, O. Plchot, F. Grezl, L. Burget, and J. Cernocky, "Analysis of dnn approaches to speaker identification," *Proc. IEEE ICASSP, Shanghai, China*, pp. 5100–5104, 2016.

[8] P. Motlicek *et al.*, "Employment of subspace gaussian mixture models in speaker recognition," in *To Appear In Proc. of ICASSP 2015*, 2015.

[9] S. O. Sadjadi, S. Ganapathy, and J. Pelecanos, "The ibm 2016 speaker recognition system," in *Odyssey 2016*, 2016, pp. 174–180.

[10] K. A. Lee, V. Hautamäki, T. Kinnunen, A. Larcher, C. Zhang, A. Nautsch, T. Stafylakis, G. Liu, M. Rouvier, W. Rao *et al.*, "The i4u mega fusion and collaboration for nist speaker recognition evaluation 2016," 2017.

[11] S. Madikeri, S. Dey, M. Ferras, P. Motlicek, and I. Himawan, "Idiap submission to the nist sre 2016 speaker recognition evaluation," Idiap, Tech. Rep., 2016.

[12] P. A. Torres-Carrasquillo, F. Richardson, S. Nercessian, D. Sturim, W. Campbell, Y. Gwon, S. Vattam, N. Dehak, H. Mallidi, P. S. Nidadavolu *et al.*, "The mit-ll, jhu and lrde nist 2016 speaker recognition evaluation system," *Proc. Interspeech 2017*, pp. 1333–1337, 2017.

[13] O. Novotnỳ, P. Matějka, O. Glembek, O. Plchot, F. Grézl, L. Burget *et al.*, "Analysis of the dnn-based sre systems in multi-language conditions," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, 2016, pp. 199–204.

[14] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," 2017.

[15] "The NIST Year 2010 Speaker Recognition Evaluation Plan, http://www.itl.nist.gov/iad/mig/tests/sre/2010/index.html."

[16] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," 2001.

[17] D. Povey, A. Ghoshal *et al.*, "The kaldi speech recognition toolkit," in *In Proc. of ASRU 2011*, December 2011.

[18] S. Madikeri, S. Dey, P. Motlicek, and M. Ferras, "Implementation of the standard i-vector system for the kaldi speech recognition toolkit," No. EPFL-REPORT-223041 Idiap, Tech. Rep., 2016.

[19] O. Glembek *et al.*, "Simplification and optimization of i-vector extraction." In Proc. of ICASSP, 2011, pp. 4516–4519.

[20] S. Madikeri, "A fast and scalable hybrid FA/PPCA-based framework for speaker recognition," *Digital Signal Processing*, vol. 32, pp. 137–145, 2014.