Migrating i-vectors Between Speaker Recognition Systems Using Regression Neural Networks

Ondřej Glembek[1], *Pavel Matějka*[12], *Oldřich Plchot*[1], *Jan Pešán*[1], *Lukáš Burget*[1], *and Petr Schwarz*[12]

[1]Brno University of Technology, Speech@FIT group and IT4I Centre of Excellence, Czech Republic
[2]Phonexia s.r.o., Brno, Czech Republic
{glembek,matejkap,iplchot,ipesan,burget,schwarzp}@fit.vutbr.cz

Abstract

This paper studies the scenario of migrating from one i-vector-based speaker recognition system (SRE) to another, i.e. comparing the i-vectors produced by one system with those produced by another system. System migration would typically be motivated by deploying a system with improved recognition accuracy, e.g. because of technological upgrade, or because of the necessity of processing new kind of data, etc. Unfortunately, such migration is very likely to result in the incompatibility between the new and the original i-vectors and, therefore, in the inability of comparing the two. This work studies various topologies of Regression Neural Networks for transforming i-vectors from three different systems so that—with slight loss in the accuracy—they are compatible with the reference system. We present the results on the NIST SRE 2010 telephone condition.

Index Terms: speaker recognition, i-vector transformation, Regression Neural Networks, system migration

1. Introduction

Ever since their introduction in Speaker Recognition, i-vectors have been widely used in multiple fields of speech processing, such as Language Recognition [1], Age Estimation [2, 3], Emotion Detection [4], and even in Speech Recognition [5, 6]. The so-called i-vector is an information-rich low-dimensional fixed-length vector extracted from the feature sequence representing a speech segment (see Section 2 for details on i-vector extraction).

Due to these properties, the i-vectors are often referred to as audio *voice-prints*. Let us note that the term voice-print should be taken with care—as has been thoroughly discussed in [7] and [8]—and is only used in this work to denote a possible representation of an utterance. As such, the i-vectors can be used for audio indexing purposes, information exchange (e.g. forensic or intelligence agencies), speaker search, etc. Such usage, however, assumes that the i-vector extraction method (including the parameters of the method) is kept fixed, so that all i-vectors are compatible, and that their direct comparison is feasible.

I-vector extraction is a complex process which depends on many sub-tasks, each of which is a subject to continuous research aiming at increasing recognition performance. It is very likely, that with every such improvement or change, the i-vector interpretation changes, therefore making it impossible to perform any direct i-vector comparison. Using a deployed i-vector extraction system—let us refer to it as the *reference* system—for comparing scoring i-vectors from an alternative or *alien* system would therefore require re-extracting the i-vectors for every utterance from the source audio.

Let us study an example scenario of a company having a database of i-vectors. For legal, capacity, or other reasons, the company cannot store the corresponding audio files. At a certain point, the company decides to upgrade its i-vector extraction to a newer system (now the "reference") but would still like to be able to use its existing database of i-vectors (now the "alien-system" generated i-vectors). Another example could be the need of inter-agency "voice-print" exchange; if two agencies use different i-vector extraction methods and want to exchange their i-vectors, there has to be a technique of mapping the alien i-vectors to the reference i-vectors.

In this work, we present a technique of computing the migration transformation of the alien i-vectors to the reference i-vectors, provided that, there is a training set of i-vectors generated by both the reference and the alien systems. We study several topologies of Artificial Regression Neural Networks (NN)—with one and two hidden layers, as well as with no hidden layer, downgrading it to mere linear regression—to transform the i-vectors produced by an alien system to be compatible with the reference system.

2. Theoretical Background

Let us first take a look at the anatomy of our system. We will then describe the techniques used to transform the i-vectors to fit the reference system.

2.1. Feature extraction

In our systems, we used two different core feature extraction—the MFCCs and the Perseus features [9], both described below. Both techniques produce a 20-dimensional feature vector calculated every 10ms. This 20-dimensional feature vector was subjected to short time mean- and variance-normalization using a 3 s sliding window. Delta and double delta coefficients were then calculated using a five-frame window giving a 60-dimensional feature vector.

Speech/silence segmentation was performed by the BUT Czech phoneme recognizer [10], where all phoneme classes are linked to the *speech* class. The recognizer was trained on the Czech CTS data, but we have added noise with varying SNR to the 30% of the database.

2.1.1. MFCC

In our experiments, we used cepstral features, extracted using a 25 ms Hamming window. We used 24 Mel-filter banks and we limited the bandwidth to the 125–3800Hz range. 19 Mel frequency cepstral coefficients together with zero-*th* coefficient were calculated every 10 ms.

Variants of these features are de-facto standard in the SRE community and our reference system is based on these features.

2.1.2. Perseus Features

In [11], MMeDuSa features were proposed as noise robust features for speaker identification. On channel and noise degraded RATS corpus [12, 13], the MMedusa features were shown to provide performance superior to conventional MFCC features. The disadvantage of MMeDuSa features is their high computation complexity of their extraction, which can be the most computationally demanding step of the whole processing chain in a speaker recognition system. Therefore, Perseus features [9] were designed to mimic the MMeDuSa features by modifying MFCC extraction in the following way:

With frame rate 10 ms, power spectrum is calculated for 50 ms speech frames weighted by Hamming window. Like for MFCCs, filter bank output is calculated by integrating regions of spectra using weighting functions. However, magnitude of frequency responses of filters from Gammatone filter bank are used as the weighting functions instead of MFCC-like triangular windows. The 15th root compression is applied to the filter bank output instead of MFCC-like log compression. The resulting coefficients are de-correlated using DCT as in the case of MFCCs. The resulting feature vector is augmented with 3 additional coefficients encoding evolution of energy inside of each frame. These 3 coefficients are calculated as follows: 1) absolute value of frame samples is taken, 2) the resulting signal is projected into 11 DCT bases (skipping the zero-th constant basis), 3) power and 15th-root of these coefficients is taken, 4) the resulting vector is projected into 3 DCT bases.

We have observed that our Perseus features were indeed very similar to our implementation of MMeDuSa in terms of both similarity of feature vectors and the speaker recognition performance.

2.2. i-vectors

The i-vectors provide an elegant way of reducing large-dimensional input data to a small-dimensional feature vector while retaining most of the relevant information. The technique was originally inspired by Joint Factor Analysis (JFA) framework introduced in [14, 15].

The main idea is that the utterance-dependent Gaussian Mixture Model (GMM) supervector of concatenated GMM mean vectors $\mathbf{s}$ can be modeled as:

$$\mathbf{s} = \mathbf{m} + \mathbf{T}\mathbf{w} \tag{1}$$

where $\mathbf{m}$ is the Universal Background Model (UBM) GMM mean supervector, $\mathbf{T}$ is a low-rank matrix representing M bases spanning subspace with important variability in the mean supervector space, and $\mathbf{w}$ is a latent variable of size M with standard normal distribution.

For each observation $\mathcal{X}$, the aim is to compute the parameters of the posterior probability of $\mathbf{w}$:

$$\mathrm{p}(\mathbf{w}|\mathcal{X}) = \mathcal{N}(\mathbf{w}; \mathbf{w}_{\mathcal{X}}, \mathbf{L}_{\mathcal{X}}^{-1}) \tag{2}$$

The i-vector ϕ is the Maximum a Posteriori (MAP) point estimate of the variable $\mathbf{w}$, i.e., the mean $\mathbf{w}_{\mathcal{X}}$ of the posterior distribution $\mathrm{p}(\mathbf{w}|\mathcal{X})$. It maps most of the relevant information from a variable-length observation $\mathcal{X}$ to a fixed- (small-) dimensional vector. $\mathbf{L}_{\mathcal{X}}$ is the precision of the posterior distribution.

2.3. Scoring

The comparison of i-vectors is facilitated via Probabilistic Linear Discriminant Analysis (PLDA) model [16, 17]. Given a pair of i-vectors, i.e. the *trial*, PLDA allows to compute the log-likelihood for the same-speaker hypothesis and for the different-speaker hypothesis.

The pre-processing of i-vectors consists of applying LDA to reduce the dimensionality to 200. Such processed i-vectors are then followed by global mean and variance normalization, followed by length-normalization [18, 19].

2.4. i-vector Transformation

As discussed in the introduction section, in order to allow for PLDA to meaningfully compute a score for a trial, both i-vectors of the trial must be generated using the same i-vector extractor. However, if one or both sides of the trial are based on an i-vector generated by an alien system, PLDA miss-interprets the i-vectors and the comparison fails, as will be demonstrated in the experimental section.

In this work we used Regression Artificial Neural Networks to map the alien i-vectors to the reference. The objective was to minimize the Mean Square Error and we used Stochastic Gradient Descent (SGD) method to train the parameters of the NN. We used random initialization of the NN parameters and sigmoid activation function on the hidden layers.

We experimented with zero-, one-, and two-hidden-layers topologies. Note that the zero-hidden-layer is formally a linear regression, however, we used a cross-validation set and SGD to estimate its parameters.

3. Experiments

3.1. Datasets and Test Protocol

Unless otherwise stated, we used the PRISM [20] training dataset definition to train all parts of our models, including the i-vector transformation. This set comprises the Fisher 1 and 2, Switchboard phase 2 and 3 and Switchboard cellphone phases 1 and 2, along with a set of Mixer speakers. This includes the 66 held out speakers from SRE10 (see Section III-B5), and 965, 980, 485 and 310 speakers from SRE08, SRE06, SRE05 and SRE04, respectively. A total of 13,916 speakers are available in Fisher data and 1,991 in Switchboard data.

We evaluated our experiments on the female portion of the NIST SRE 2010 telephone condition [21]. The recognition performance is evaluated in terms of the equal error rate (EER), the normalized minimum detection cost functions (DCF) as defined in both the NIST 2010 SRE task ($\mathrm{DCF}^{\mathrm{min}}_{\mathrm{new}}$) and the previous SRE 2005, 2006, 2008 evaluations ($\mathrm{DCF}^{\mathrm{min}}_{\mathrm{old}}$), and their actual variants $\mathrm{DCF}^{\mathrm{act}}_{\mathrm{new}}$and $\mathrm{DCF}^{\mathrm{act}}_{\mathrm{old}}$, respectively.

3.2. System Setup and Test Protocol

There were four systems involved in our set of experiments—one reference and three alien systems:

reference — This is the reference system to which all following alien systems are adapted. It is based on the MFCC features, 2048-component GMM, 600-dimensional i-vectors.

512/400 — this system is derived from the reference, but the size of the UBM has been limited to 512 Gaussian components, and the dimensionality of the i-vector is set to 400.

Table 1: *Comparing different NN topologies on the Perseus system. The numbers show the size of the NN layers. The "600-600" indicates no hidden layer in the topology. The asterisk (*) denotes the hybrid test.*

System	$\mathrm{DCF}^{\mathrm{min}}_{\mathrm{new}}$	$\mathrm{DCF}^{\mathrm{act}}_{\mathrm{new}}$	$\mathrm{DCF}^{\mathrm{min}}_{\mathrm{old}}$	$\mathrm{DCF}^{\mathrm{act}}_{\mathrm{old}}$	eer
reference	0.3834	0.3940	0.1089	0.2124	2.13
Perseus on reference	1.0000	102.6261	0.7834	1.7379	23.12
Perseus baseline	0.4924	0.6876	0.1494	0.2078	2.86
600-600	0.4662	0.4836	0.1522	0.2949	2.85
600-600*	0.4490	0.4650	0.1360	0.3207	2.64
600-600-600	0.5596	0.5853	0.1799	0.3463	3.48
600-600-600*	0.5039	0.5108	0.1526	0.3517	2.96
600-1200-600	0.5794	0.6727	0.1732	0.3131	3.56
600-1200-600*	0.4834	0.4962	0.1467	0.3166	2.93
600-600-600-600	0.5845	0.6136	0.1898	0.3642	3.66
600-600-600-600*	0.5045	0.5295	0.1587	0.3549	3.09

Table 2: *Results of linear regression for all systems. The "baseline" numbers show the results of the evaluation carried out on the corresponding systems. The asterisk (*) denotes the hybrid test.*

System		$\mathrm{DCF}^{\mathrm{min}}_{\mathrm{new}}$	$\mathrm{DCF}^{\mathrm{act}}_{\mathrm{new}}$	$\mathrm{DCF}^{\mathrm{min}}_{\mathrm{old}}$	$\mathrm{DCF}^{\mathrm{act}}_{\mathrm{old}}$	eer
reference	baseline	0.3834	0.3940	0.1089	0.2124	2.13
512/400	baseline	0.5711	1.0846	0.1742	0.2192	3.78
	400-600	0.5011	0.5160	0.1548	0.3151	3.12
	400-600*	0.4555	0.4685	0.1387	0.3012	2.76
Red-Ref	baseline	0.4475	0.4581	0.1283	0.2372	2.64
	600-600	0.4392	0.4595	0.1299	0.2580	2.73
	600-600*	0.4224	0.4363	0.1213	0.2514	2.53
Perseus	baseline	0.4924	0.6876	0.1494	0.2078	2.86
	600-600	0.4662	0.4836	0.1522	0.2949	2.85
	600-600*	0.4490	0.4650	0.1360	0.3207	2.64

Red-Ref — this system is essentially the same as the reference system, except the training portion of the training data was reduced by excluding the Fisher and Switchboard portion. However, both of these data-sets were kept for training the i-vector transformation.

Perseus - this system differs from the reference system by substituting the features by the Perseus, as described in Sec. 2.1.2. We have included a system based on these features as they provide complementary information to the MFCC's and—although they are outperformed by the MFCC on the NIST task—they proved to outperform cepstral features on the RATS task [13], which deals with heavily distorted radio recordings.

For each case, we built the whole recognition system to test how each system performs on its own. We mark these as the "baseline" systems in the results section. We have included these numbers to show, how well we would perform the recognition using such system.

Then, for each alien system, we trained the i-vector mapping NN using the PRISM dataset and forwarded the test i-vectors through this transform. These i-vectors were then scored using the reference system PLDA in two scenarios: i) the *matched test*—both the enroll and the test i-vectors are transformed alien i-vectors, and ii) the *hybrid test*, where the enroll i-vectors were the original reference i-vectors, and the test i-vectors were the transformed alien i-vectors. Since the enroll and test sets are disjoint, we repeated this test with the two sides swapped and we averaged the results.

3.3. Results

Tab. 1 shows the performance of the various modifications of the Neural Networks on the Perseus system. The "reference" refers to the case, when reference system i-vectors were evaluated natively using the reference backend. "Perseus on reference" only demonstrates that evaluating Perseus i-vectors using the reference backend without transforming them breaks the performance. "Perseus baseline" shows the performance of the Perseus i-vectors evaluated natively using the Perseus backend, i.e. what the best performance that the Perseus i-vectors can produce is. The non-asterisk labels denote a matched test, and the asterisk (*) marks a hybrid test. The numbers in the description denote the dimensionality of each layer.

First thing to note is that the hybrid test always does better than the matched test. This suggests that the loss of speaker information is happening not at the stage of i-vector transformation, but rather at the stage of i-vector extraction.

The second thing to note is that overall, the linear regression generally performs better than the hidden-layer NNs. We tried to expand the hidden layer to 1200 and to even add another hidden layer, but in general, the more parameters we use, the worse result. Larger systems probably get over-trained. We performed these experiments using the other alien systems, seeing similar trends.

The third thing to note is that—on most operating points—the alien vectors perform better in their linear regression transformed version than in the baseline experiment. Our hypothesis for this is that PLDA was trained robustly using the reference system, where the i-vectors' speaker- and channel- subspaces have cleaner definition.

Tab. 2 shows the results of linear regression for all alien systems. The "reference baseline" is the target system as described above. Not only the systems are comparable to the alien baseline versions, but—as was discussed in the previous paragraph—on many operating points, the transformed alien i-vectors outperform the alien baseline results.

4. Conclusions

We have shown that a linear transformation can be used to transform alien i-vectors to the reference i-vectors as the input to the reference PLDA system. Not only the performance of the transformed i-vectors is comparable to the pure alien-system, but in many cases, the transformed i-vectors outperform the original alien system. It was also shown that the reference PLDA performs better if one side of the trial comes from the reference system. These facts indicate that the loss of information is happening at the level of i-vector extraction rather than at the level of i-vector transformation.

5. References

[1] David González Martínez, Oldřich Plchot, Lukáš Burget, Ondřej Glembek, and Pavel Matějka, "Language recognition in ivectors space," in *Proceedings of Interspeech 2011*. 2011, vol. 2011, pp. 861–864, International Speech Communication Association.

[2] Mohamad Hasan Bahari, Mitchell McLaren, Hugo Van hamme, and David A. van Leeuwen, "Speaker age estimation using i-vectors," *Eng. Appl. of AI*, vol. 34, pp. 99–108, 2014.

[3] Anna Fedorova, Ondrej Glembek, Pavel Matejka, and Tomi Kinnunen, "Exploring ANN back-ends for i-vector based speaker age estimation," in *Submitted to Interspeech, 2015*, 2015.

[4] Marcel Kockmann, Lukáš Burget, and Jan Černocký, "Application of speaker- and language identification state-of-the-art techniques for emotion recognition," *Speech Communication*, vol. 53, no. 9, pp. 1172–1185, 2011.

[5] Martin Karafiát, Lukáš Burget, Pavel Matějka, Ondřej Glembek, and Jan Černocký, "ivector-based discriminative adaptation for automatic speech recognition," in *Proceedings of ASRU 2011*. 2011, pp. 152–157, IEEE Signal Processing Society.

[6] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, Dec 2013, pp. 55–59.

[7] J.P. Campbell, W. Shen, W.M. Campbell, R. Schwartz, J.-F. Bonastre, and D. Matrouf, "Forensic speaker recognition," *Signal Processing Magazine, IEEE*, vol. 26, no. 2, pp. 95–103, March 2009.

[8] Jean-François Bonastre, Louis-Jean Bimbot, Frédéric an Boë, Joseph P. Campbell, Douglas A. Reynolds, and Ivan Magrin-Chagnolleau, "Person authentication by voice: a need for caution.," in *INTERSPEECH*. 2003, ISCA.

[9] Lukáš Burget, "The perseus features," BUT Technical Report. Online: http://www.fit.vutbr.cz/∼burget, Mar. 2015.

[10] Pavel Matějka, Lukáš Burget, Petr Schwarz, and Jan Černocký, "Brno university of technology system for NIST 2005 language recognition evaluation," in *Proceedings of Odyssey 2006: The Speaker and Language Recognition Workshop*, 2006, pp. 57–64.

[11] Vikramjit Mitra, Mitchell McLaren, Horacio Franco, Martin Graciarena, and Nicolas Scheffer, "Modulation features for noise robust speaker identification," in *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25-29, 2013*, 2013, pp. 3703–3707.

[12] Oldrich Plchot, Spyros Matsoukas, Pavel Matejka, Najim Dehak, Jeff Ma, S. Cumani, O. Glembek, H. Hermansky, S.H. Mallidi, N. Mesgarani, R. Schwartz, M. Soufifar, Z.H. Tan, S. Thomas, B. Zhang, and X. Zhou, "Developing a speaker identification system for the DARPA RATS project," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013, pp. 6768–6772.

[13] K. Walker and S. Strassel, "The RATS radio traffic collection system," in *ISCA Speaker Odyssey*, 2012.

[14] P. Kenny, "Joint factor analysis of speaker and session variability : Theory and algorithms - technical report CRIM-06/08-13. Montreal, CRIM, 2005," 2005.

[15] P. Kenny, G. Boulianne, P. Oullet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 7, pp. 2072–2084, 2007.

[16] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *11th International Conference on Computer Vision*, 2007, pp. 1–8.

[17] Patrick Kenny, "Bayesian speaker verification with heavy–tailed priors," in *Proc. of Odyssey 2010*, Brno, Czech Republic, June 2010, http://www.crim.ca/perso/patrick.kenny, keynote presentation.

[18] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. PP, no. 99, pp. 1 –1, 2010.

[19] Daniel Garcia-Romero, "Analysis of i-vector length normalization in Gaussian-PLDA speaker recognition systems," in *Proc. of the International Conference on Spoken Language Processing (ICSLP)*, 2011.

[20] Luciana Ferrer, Harry Bratt, Lukas Burget, Honza Cernockyy, Ondrej Glembeky, Martin Graciarena, Aaron Lawson, Yun Lei, Pavel Matejkay, Olda Plchoty, and Nicolas Scheffer, "Promoting robustness for speaker modeling in the community: the prism evaluation set," 2011.

[21] "National institute of standards and technology," http://www.nist.gov/speech/tests/spk/index.htm.