NEAREST NEIGHBOR BASED I-VECTOR NORMALIZATION FOR ROBUST SPEAKER RECOGNITION UNDER UNSEEN CHANNEL CONDITIONS

Weizhong Zhu, Seyed Omid Sadjadi, Jason W. Pelecanos

IBM Research, Watson Group
Yorktown Heights, NY 10598, USA
{zhuwe,sadjadi,jwpeleca}@us.ibm.com

ABSTRACT

Many state-of-the-art speaker recognition engines use i-vectors to represent variable-length acoustic signals in a fixed low-dimensional total variability subspace. While such systems perform well under seen channel conditions, their performance greatly degrades under unseen channel scenarios. Accordingly, rapid adaptation of i-vector systems to unseen conditions has recently attracted significant research effort from the community. To mitigate this mismatch, in this paper we propose nearest neighbor based i-vector mean normalization (NN-IMN) and i-vector smoothing (IS) for unsupervised adaptation to unseen channel conditions within a state-of-the-art i-vector/PLDA speaker verification framework. A major advantage of the approach is its ability to handle multiple unseen channels without explicit retraining or clustering. Our observations on the DARPA Robust Automatic Transcription of Speech (RATS) speaker recognition task suggest that part of the distortion caused by an unseen channel may be modeled as an offset in the i-vector space. Hence, the proposed nearest neighbor based normalization technique is formulated to compensate for such a shift. Experimental results with the NN based normalized i-vectors indicate that, on average, we can recover 46% of the total performance degradation due to unseen channel conditions.

Index Terms— i-vector, nearest neighbor, PLDA, speaker recognition, unsupervised adaptation

1. INTRODUCTION

The i-vector representation of acoustic signals has become a mainstream front-end in state-of-the-art speaker and language recognition systems [1]. Similar to the supervector representation for a Gaussian mixture model (GMM) framework [2], an i-vector is a fixed-length representation of a speech recording, but with much lower dimensionality. Probabilistic linear discriminant analysis (PLDA) [3] can then be applied to model the distribution of the i-vectors. PLDA provides a powerful mechanism to jointly model the signal (i.e., speaker) and noise (i.e., channel, session, etc) subspaces. In order for PLDA to provide the best performance, the acoustic conditions (e.g., background noise, communication channel, room reverberation, etc) should be similar across the evaluation and development data. Nevertheless, in real world applications, it can be impractical to access large quantities of training data from many speakers and for every possible acoustic environment. Rather, it would be more realistic to assume access to some unlabeled data for unseen conditions and to attempt to adapt to the new environment.

There are several articles that discuss approaches for domain adaptation in order to improve robustness and generalization of i-vector/PLDA speaker recognition systems [4], [5], [6]. These approaches typically involve using some adaptation data to learn and remove/suppress the nuisance channel directions. For instance, nuisance attribute projection (NAP) was proposed, originally in the GMM supervector framework, to remove directions related to channel or session variabilities [7]. Similar to NAP, more recently, an inter-dataset variability compensation (IDVC) method was presented in [5] to directly compensate for dataset shift in the i-vector domain. Another method assumes that development data may originate from several different sources [8] and without speaker labels. The source normalization (SN) method estimates the between speaker covariance based on different sources and the within-speaker covariance is obtained by subtracting the estimated between speaker covariance from the total covariance.

Intra-speaker variability can also be decomposed as between-dataset and inter-session variabilities. In [4] one assumption is that the inter-session covariance is shared across datasets. To adapt a system to a new channel/domain, a Within-class Covariance Correction (WCC) method was proposed to correct the intra-speaker covariance by deemphasizing the direction related to the mean shift of the new data i-vectors from the development data i-vectors. Consequently, speaker labels are not required with this method.

In this paper, we propose a nearest neighbor based i-vector normalization technique to compensate for the shift due to the unseen channel characteristics. Unlike previous methods, the advantage of this approach is its ability to handle multiple unseen channel recordings without retraining existing models or re-clustering unseen channel data. It can also be operated in an online mode as new data is captured.

The rest of the paper is organized as follows: Section 2 presents some properties of i-vectors for unseen channel data and introduces the nearest neighbor based i-vector mean normalization (NN-IMN) and i-vector smoothing (IS) techniques. Section 3 describes the experimental setup and results and Section 4 follows with the conclusions.

2. NEAREST NEIGHBOR I-VECTOR NORMALIZATION

In order to detail the NN i-vector normalization technique, we begin with describing our baseline system [9].

This work was supported in part by Contract No. D11PC20192 DOI/NBC under the RATS program. The views, opinions, findings and recommendations contained in this article are those of the authors and do not necessarily reflect the official policy or position of the DOI/NBC.

2.1. Baseline System

Here we identify a generic i-vector/PLDA speaker recognition system as three main components (see Fig. 1):

1) **I-vector Extraction:** For each recording, the Baum-Welch sufficient statistics are extracted using the UBM to form a high dimensional space (i.e., mean supervector space), and then a factor analysis (FA) framework [10] is used to estimate a lower dimensional total variability subspace [**?**] within which i-vectors represent the coordinates for each observation.
2) **Intersession variability post-processing** is performed by linear discriminant analysis (LDA), followed by within class covariance normalization (WCCN) [11].
3) **Probabilistic linear discriminant analysis (PLDA)** is used to model i-vectors using between and within speaker subspaces [3]. It can also be used to score the verification trials as a log likelihood ratio between same vs. different speaker hypotheses.

It is worth noting that both the LDA and PLDA models assume Gaussian distributions. We have found that performing unit-length normalization before both the LDA and the PLDA stages results in improved speaker recognition performance.

2.2. Unseen Channel Effects in i-vectors

In order to simulate an unseen channel scenario, we use the speech material from the RATS program which provides noisy and channel degraded audio recordings from a total of 8 HF radio channels (labeled A through H) [12]. We hold one channel out at a time and assume that the system is blind to that specific channel (i.e., apart from the GMM-UBM, none of the major components of the baseline system in Fig. 1 have seen data from that specific channel during training). For example, assuming channel G as the unseen channel, we only use data from channels A through to F, and H to train the system components.

As an example, Fig. 2 shows the scatter plot of two (of the highest variance) dimensions of i-vectors extracted from the same set of source recordings transmitted over channels A and G. We note that the i-vector extraction process is trained on all channels except for channel G. (Here, G is considered an unseen channel.) The blue circles represent samples from channel A while the red crosses correspond to recordings from channel G. One observation that can be made from Fig. 2 is that the red crosses (i.e., samples from channel G) are shifted away from the origin. Similar phenomena were also observed when data from other channels were excluded from training. It may be possible to improve performance for unseen channels if such a shift in i-vector space is compensated for. Here we propose the nearest neighbor based i-vector mean normalization (NN-IMN) and i-vector smoothing (NN-IS) technique to correct the shifts.

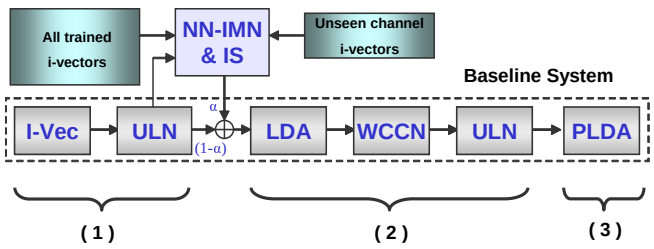

Fig. 1. Block diagram of a baseline i-vector/PLDA speaker recognition system.

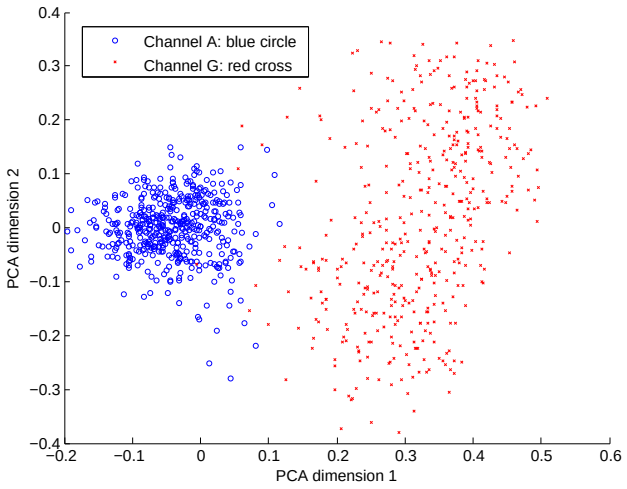

Fig. 2. Scatter plot of the original i-vectors with Channel G excluded from system training.

2.3. Nearest Neighbor I-vector Normalization

In order to perform nearest neighbor based i-vector mean normalization (NN-IMN) and i-vector smoothing (IS), we add a new functional block to the baseline system as shown in Fig. 1. This new block requires two i-vector datasets as inputs. The first dataset (shown on the left) contains all unit-length normalized i-vectors used in the training, while the second dataset (shown on the right) contains all unit-length normalized i-vectors extracted from the unseen channel data. It is worth noting here that our proposed technique performs unsupervised adaptation, and the components in Fig. 1 and their associated models remain unchanged(i.e. no retraining).

The procedure for performing the nearest neighbor based i-vector normalization follows.

2.3.1. Nearest Neighbor I-vector Mean Normalization (NN-IMN)

Given an i-vector to be compensated, we first compare the cosine distance of this i-vector with i-vectors from the (seen and unseen data) library. Specifically, the cosine distance between the i-vector to be compensated, $\mathbf{w}$, and a single library vector, $\mathbf{x}_j$, is given as,

$$d_j = \frac{\mathbf{w} \cdot \mathbf{x}_j}{\|\mathbf{w}\| \cdot \|\mathbf{x}_j\|}. \quad (1)$$

By rank ordering these distances, the K nearest neighbors are determined, and the kth nearest neighbor is given as $NN_k(\mathbf{w})$. The compensated, mean-normalized i-vector is given as,

$$\mathbf{w}_c = \mathbf{w} - \frac{1}{K}\sum_{k=1}^{K} NN_k(\mathbf{w}). \quad (2)$$

Using this formulation, our pilot experiments on the RATS speaker recognition task indicate the effectiveness of the NN-IMN technique on verification trials involving samples from unseen channels. However, a slight performance degradation on the seen channel recordings is observed. Given this observation, a smoothing stage may help in the estimation process of the normalized i-vectors. One such i-vector smoothing method is introduced next to alleviate this issue.

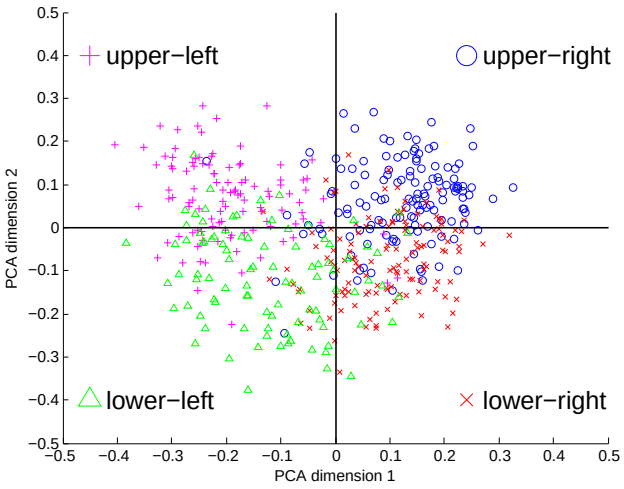

Fig. 3. Scatter plot of i-vectors from Channel G compensated by NN-IMN and IS when Channel G is excluded from system training.

2.3.2. *I-vector Smoothing (IS)*

As noted previously, smoothing (*maximum-a-posteriori* based or otherwise) may provide more robust estimates of compensated i-vectors across a multitude of conditions. More specifically, IS is given as,

$$\begin{aligned} \widetilde{\mathbf{w}}_c &= \alpha \mathbf{w}_c + (1-\alpha)\mathbf{w} \\ &= \mathbf{w} - \frac{\alpha}{K}\sum_{k=1}^{K} NN_k(\mathbf{w}), \end{aligned} \tag{3}$$

where α is a smoothing constant $(0 \leq \alpha \leq 1)$ that controls the contribution of each component in the final estimate. Taken together, for the i-vector, $\mathbf{w}$, the sample average of its K-nearest neighbors is first computed, and then the smoothing parameter, α, determines the degree to which we compensate for the shift due to mismatched conditions.

We now attempt to visualize the collective effect of the nearest neighbor compensation with smoothing ($\alpha = 0.5$) on unseen channel data. Fig. 3 shows the scatter plot of the resulting i-vectors (after LDA and WCCN are applied) for unseen channel G recordings with compensation applied. The channel G samples are colored (and assigned a shape) according to which quadrant the corresponding seen channel A recordings came from. The leakage across the quadrant boundaries indicates some of the variation that could not be normalized. However, the majority of samples in one quadrant for channel A, remain in the same quadrant for channel G after compensation.

Given this, processing i-vectors with the proposed nearest neighbor based normalization technique may help speaker recognition performance. This is verified in the following section.

3. EXPERIMENTS

3.1. Data and System

In this section we describe the experimental setup used in our evaluations. We begin by discussing the data used and then provide a brief overview of the i-vector extraction process.

Speech material used in our evaluations is sourced from data distributed by the Linguistic Data Consortium (LDC) for the DARPA RATS program [12]. It comprises conversational telephone speech (CTS) recordings that have been retransmitted (through LDC's Multi Radio-Link Channel Collection System) and captured over 8 degraded HF radio communication channels (labeled A–H). There are various distortion characteristics, some of which can be described as nonlinear (e.g., clipping, amplitude compression as well as frequency shift effects) and the noise is to some extent correlated with speech. The speech data is from five languages: Levantine Arabic, Dari, Farsi, Pashto, and Urdu.

The training set consists of 20k segments each with approximately 30 seconds of speech. This gives roughly 2,500 segments per channel. There are a total of 5,000 speakers in this training set. In this study, we evaluate the trials constructed from six 30-second sessions for enrollment and a single session for test. Our internal evaluation set contains 21k segments from 314 speakers from which we generated a total of 151k trials across the eight channel conditions. The speaker recognition performance is reported in terms of percent equal error rate (EER).

We now describe details of the system (similar to [9]) used for scoring the trials. For the speech parameterization, 19-dimensional MFCCs (32 ms frames every 10 ms using a 24-channel Mel-filterbank) span the frequency range 125–3700 Hz. The first and second temporal derivatives are computed over a 5-frame window and are appended to the static features. This results in 57-dimensional feature vectors. Segment level mean and variance normalization is applied based on speech-only frames determined by a speech activity detection (SAD) component that uses energy, voicing, and spectral divergence parameters. To learn the i-vector extractor, a 1024-component gender-independent GMM is trained. The dimensionality of the total variability subspace is set to 400, which is further reduced to 200 dimensions using LDA followed by WCCN [11]. Unit-length normalization is applied twice, that is once before LDA and before PLDA modeling (see Fig. 1).

3.2. Results and Discussion

In this paper there are four main approaches evaluated:

1. **Full-Train**: The system is trained with all data available, including all channel data (Channels A–H).
2. **Leave-one-out**: This approach involves removing a single channel (for example Channel A) from the system building process. It allows us to observe the performance degradation for channels not seen in system training.
3. **NN-IMN+IS**: Our proposed method whereby each i-vector is compensated by its K nearest neighbors, ($K = 100$ and with a smoothing factor, $\alpha = 0.5$).
4. **WCC, 1-Cluster**: For comparison, we also incorporated the Within-class Covariance Correction (WCC) method (with details in [4]). This method adjusts the estimate of the within class covariance based on including across-dataset covariance information. We also evaluate a 4-cluster version in an experiment where 4 channels are treated as *held out* (or unseen) from system training.

Given the four approaches, the first set of results (Fig. 4) show the test-channel specific performance when the test-channel is held out from system training. The chart plots the equal error rate as a function of the eight different held-out channel scenarios and the

four system approaches. The unseen channel versus the seen channel performance degradation is substantial; after comparing the first and second bars for each channel, we observed up to 3 times the performance degradation. However, we note that for all eight channel cases, the performance gap was reduced by using NN-IMN+IS (the third bar), or WCC (the fourth bar). Overall, the results of the nearest neighbor approach are mostly comparable to the WCC method. On average, across the channels, the proposed nearest neighbor based i-vector normalization method recovers 46% of the total performance degradation.

While Fig. 4 shows that the unseen channel performance can be improved by using unsupervised methods, it is also important to ensure that the combination of seen and unseen channel scores is robust. In Fig. 5, we show the 30 second task performance calculated from the complete set of trials for seen (7 channels) and unseen (1 channel) test-segment channels (averaged across the eight different held out channel scenarios presented earlier). The results are plotted as a function of the number of K-nearest neighbors for the NN-IMN+IS technique and are contrasted with the other identified methods. The best NN-IMS+IS performance is achieved at 400 nearest neighbors and is comparable to the WCC approach. The overall results show that both methods improve the performance for the unseen channel recordings while maintaining compatibility and performance with the seen channel recordings.

An advantage of the proposed nearest neighbor method is its ability to handle multiple unseen channels. To evaluate, we conducted experiments that exclude channels B, D, F and G from training. The results for each channel (A–H) are shown in Fig. 6 for 5 different systems. The first system is the fully trained system as before. The second approach is the same system as the first system except that channels B, D, F and G are excluded from system training. The remaining systems (NN-IMS+IS, 1 and 4 cluster WCC) use the same data in system building as the second system. In comparing the first and second bars, the performance on unseen channels (shown as [B], [D], [F] and [G] in the chart) is significantly degraded. However, the NN-IMS+IS system (third bar) provides significant performance improvement for unseen channels over the unadapted system. For the 1-cluster WCC method, only small gains on unseen channels are observed. In contrast, if we assume that all channel labels of the unseen channel recordings are known and there are 4 clusters modeled, then the WCC method works well. The results suggest that the WCC technique depends on a suitable clustering of the unseen channel data. We note that the nearest neighbor system does not need an explicit retraining or adaptation and can be run in an online mode.

4. CONCLUSIONS

We proposed a nearest neighbor based i-vector normalization technique for unsupervised adaptation to unseen channel conditions. We have shown that on our internal data set for the DARPA RATS speaker recognition task, the proposed technique can recover 46% of the performance degradation due to unseen channel conditions. Some of the benefits of the proposed method are: 1) the ability to handle multiple unseen channels, 2) no need for explicit clustering or retraining, 3) can be operated in online mode, and 4) maintains performance for seen channels. In conclusion, the presented approach provides for a viable solution in many field applications where data mismatch is a concern.

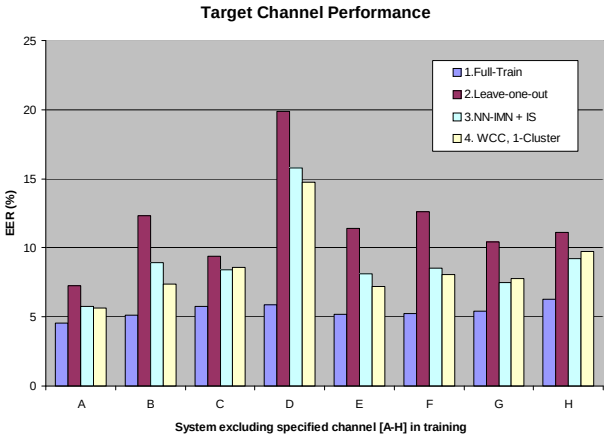

Fig. 4. Target Channel Performance (A–H)

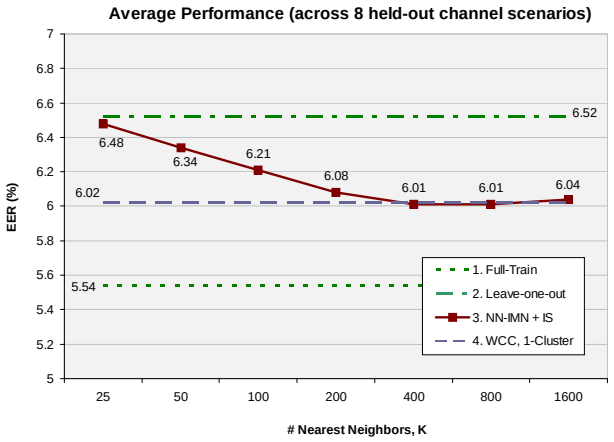

Fig. 5. Comparison of Overall System Performance

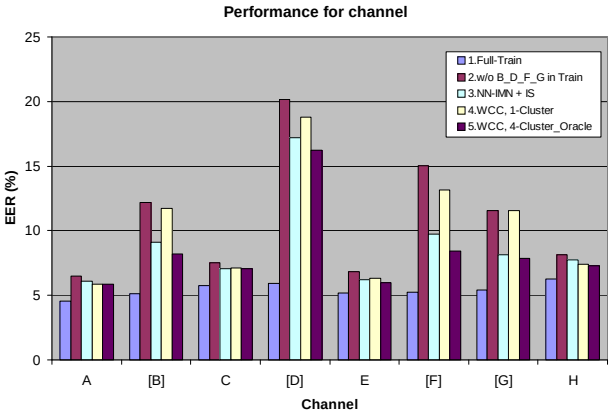

Fig. 6. Results of Excluding B,D,F,G in Training

5. REFERENCES

[1] N. Dehak et al., "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, 2011.

[2] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, 2000.

[3] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *International Conference on Computer Vision*, 2007.

[4] O. Glembek et al., "Domain adaptation via within-class covariance correction in i-vector based speaker recognition systems," in *International Conference on Acoustics, Speech, and Signal Processing*, 2014.

[5] H. Aronowitz, "Inter dataset variability compensation for speaker verification," in *International Conference on Acoustics, Speech, and Signal Processing*, 2014.

[6] D. Garcia-Romero and A. McCree, "Supervised domain adaptation for i-vector based speaker recognition," in *International Conference on Acoustics, Speech, and Signal Processing*, 2014.

[7] W. M. Campbell, "Weighted nuisance attribute projection," in *Odyssey Speaker and Language Recognition Workshop*, 2010.

[8] M. McLaren and D. van Leeuwen, "Source-normalized LDA for robust speaker recognition using i-vector from multiple speech sources," in *IEEE Transactions on Audio, Speech, and Language Processing*, 2012.

[9] W. Zhu, S. Yaman, and J. Pelecanos, "The IBM RATS phase II speaker recognition system: overview and analysis," in *Interspeech*, 2013.

[10] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," in *IEEE Transactions on Audio, Speech, and Language Processing*, 2007.

[11] A. Hatch, S. Kajarekar, and A. Stolcke, "Within-class covariance normalization for SVM-based speaker recognition," in *International Conference on Spoken Language Processing*, 2006.

[12] K. Walker and S. Strassel, "The RATS radio traffic collection system," in *Odyssey Speaker and Language Recognition Workshop*, 2012.