On Leveraging Conversational Data for Building a Text Dependent Speaker Verification System

Hagai Aronowitz[1], *Oren Barkan*[1,2]

[1]IBM Research – Haifa, Haifa, Israel
[2]School of Computer Science, Tel Aviv University, Tel Aviv, Israel
hagaia@il.ibm.com, orenba@il.ibm.com

Abstract

Recently we have investigated the use of state-of-the-art text-independent and text-dependent speaker verification algorithms for a text-dependent user authentication task and obtained satisfactory results mainly by using a fair amount of text-dependent development data. In our study, best results were obtained using the NAP framework rather than using the more advanced JFA and i-vector-based frameworks. In this work we investigate the ability to build high accuracy i-vector-based systems by leveraging widely available conversational data. We explore various techniques for transforming conversational sessions in such a way that attributes which are more relevant to the text-dependent task are enhanced. Using these techniques we managed to reduce verification error significantly.

Index Terms: speaker verification, text-dependent, transfer learning

1. Introduction

With the rapid growth of mobile internet and smart phones, security shortcomings of mobile software and mobile data communication have shifted the focus to strong authentication. Recent advances in voice biometrics offer great potential for strong authentication in mobile environments using voice. This is of particular interest in the financial and health industries, where financial and medical institutes are looking for ways to offer mobile users flexible and easy authentication while maintaining security and significantly reducing fraudulent usage.

In 2010, a work [1] has been done at IBM within the framework of a proof of technology (POT) which was performed on data collected by the Wells Fargo (WF) bank. The focus of the POT was mainly the evaluation of three text-dependent authentication scenarios. For the most accurate authentication scenario an Equal Error Rate (EER) of 0.6% was obtained (using a global 10-digit string) for the channel matched condition. This result was achieved by fusing four systems: HMM (Hidden Markov Models)-NAP (Nuisance Attribute Projection), GMM (Gaussian Mixture Models)-NAP, JFA (Joint Factor Analysis) and an i-vector based system.

However, for many other potential customer engagements the WF POT development dataset is unrealistically large (200 speakers with 4 sessions per speaker). Recently [2], in a follow-up study a more realistic development set was specified consisting of publicly available conversational (NIST) data and a reduced text dependent development set of 100 speakers from the WF-POT corpus with only a single session per speaker. Overall the use of the reduced development set resulted in a modest 20% relative increase in EER compared to the results obtained using the full development set. It was found out that the HMM-NAP and GMM-NAP based systems built on the reduced development set only (100 sessions in total) managed to outperform the JFA and i-vector systems built on the NIST data. This was done using a technique named *common speaker subspace* (*CSS*) compensation [7] which removes a low dimensional total variability subspace using the NAP method trained on the development set.

In this paper we present our efforts for building more accurate factor analysis based systems (i-vector-based in specific) for text-dependent speaker verification, focusing on the global pass-phrase authentication condition. We evaluate our methods using varying subsets of the WF development set ranging from 100 speakers with one session each to 550 speakers with four-sessions each. We explore various techniques for transforming the NIST conversational development data in such a way that attributes which are more relevant to the text-dependent data are enhanced. Using these techniques we managed to reduce verification error significantly.

The remainder of this paper is organized as follows: Section 2 describes the datasets. Section 3 describes our speaker verification baseline systems. Section 4 describes our approach for building the i-vector based system. Section 5 presents the results. Finally, Section 6 concludes.

2. Datasets

2.1. The WF corpus

The WF corpus consists of 750 speakers which are partitioned into a development dataset (200 speakers) and an evaluation dataset (550 speakers). Each speaker has 2 sessions using a landline phone and 2 sessions using a cellular phone. The data collection was accomplished over a period of 4 weeks.

Four authentication conditions were defined and collected, and experimental results were reported for them in [1-3]. In this work we limit ourselves to the global condition in which a global pass phrase is used for both enrollment and verification. The global condition has the advantage of potentially having development data with the same common pass phrase. For the global condition the WF dataset consists of several common pass phrases. In this work we report results on the single common 10-digit pass phrase 0-1-2-3-4-5-6-7-8-9 which we name ZN.

In the WF dataset each session contains 3 repetition of ZN. For each enrollment session we use all 3 repetitions for enrollment, and for each verification session we use only a single repetition.

In order to evaluate the sensitivity of our proposed methods to the size of the text dependent development set, we created additional partitions. The official POT partition is denoted by WF_200_4 (200 speakers, 4 sessions each) for development and WF_550_4 (550 speakers, 4 sessions each) for evaluation. We also switched the roles of the development and evaluation data to see the effect of a very large development set, and also created partition WF_100_2 and WF_100_1 (100 speakers, 2 and 1 sessions each, correspondingly) for development (using the original WF_550_4 evaluation set) to see the effect of smaller development sets.

2.2. Standard telephony development set

We use the following standard conversational telephony datasets: Switchboard-II, NIST 2004, 2005, 2006 and 2008 speaker recognition evaluations (SREs). We denote this development set by NIST.

3. Speaker verification baseline systems

In this section we describe the baseline speaker verification systems we use in this work.

3.1. I-vector-based system

Our baseline i-vector-based system [4] is inspired by the work described in [5]. We use standard i-vector extraction with length normalization followed by LDA (Linear Discriminant Analysis) and WCCN (Within Class Covariance Normalization). We use cosine-based similarity scoring and normalize using ZT-norm which we found to be slightly superior to s-norm in our setup. Our i-vector-based system was built using 12,711 sessions from Switchboard-II, NIST 2004 SRE and NIST 2006 SRE. The only use we made of the WF POT development data is for ZT-score normalization.

3.2. GMM-NAP-based system

Our GMM-NAP system inspired by [6] is described in detail in [1]. Our GMM-NAP system deviates from the standard [10] by the following modifications.

3.2.1. Two-wire NAP

In [7, 8] it was discovered that under the NAP framework, removing dominant components of the inter-speaker variability subspace in addition to removing the intra-speaker inter-session variability subspace improves speaker recognition accuracy not only for 2-wire data (for which this method was originally designed) but also for regular 4-wire data. This variant named 2-wire-NAP is therefore part of our baseline GMM-NAP system and leads to a relative reduction of 6% in EER on the WF POT.

3.2.2. Text dependent UBM & NAP projection

Contrary to the JFA and i-vector frameworks, NAP requires smaller quantities of development data to properly estimate the hyper-parameters (UBM and NAP projection). In [2] it was found that estimating text-dependent UBM and NAP from the WF-POT development set led to a relative reduction of roughly 50% in EER compared to estimating them from the NIST data.

3.2.3. Geometric mean comparison kernel

We use the kernel introduced in [9] for scoring a pair of sessions:

$$C_{GM}(E,T) = m_E^t\left(\lambda_E^{1/2} \otimes I_n\right)\Sigma^{-1}\left(\lambda_T^{1/2} \otimes I_n\right)m_T \qquad (1)$$

where E and T stand for the enrollment and test sessions, m_E and m_T are the corresponding concatenated GMM means, λ_E and λ_T are the corresponding concatenated GMM weights, $\sum$ is a block matrix with covariance matrices from the UBM on the diagonal, n is the feature vector dimension, and $\otimes$ is the Kronecker product.

4. Building an i-vector system for text dependent speaker verification

Although the i-vector based algorithm is considered to be superior to the NAP based algorithm and generally outperforms on NIST evaluations, our past experience on the WF task was the opposite, especially for the global condition [1]. This is due to the fact that our NAP based systems (GMM and HMM) were trained from scratch on the WF development data, while the i-vector based system was not. In this section we propose several methods for training an i-vector based system to better perform on the global condition.

4.1. Building some of the components of the i-vector system on the text-dependent (WF) data

The i-vector based system is composed of the following components: UBM, I-vector extraction, LDA+WCCN and score normalization. For each component we can choose whether it is trained on the text-dependent development set or trained on the NIST development set. The decision may be based on the amount of text-dependent data available. For instance, LDA training requires multiple sessions from many speakers, which some development sets lack. The results for various experiments using different configurations and using different WF partitions are reported in subsection 5.1.

4.2. Transferring the text-independent resources (NIST) to better match the text-dependent (WF) task

Our aim is to find techniques to reduce the mismatch between the text-independent (NIST) and text-dependent (WF) data by transforming the text-independent data to better match the text-dependent task. The techniques we propose are based on the assumption that a well-matched UBM is trained on the WF data. We propose two techniques which both modify the way sufficient statistics are extracted from the NIST data. Following is a description of the proposed techniques.

4.2.1. Motivation

Sufficient statistics extraction is the process of estimating the zero and first order statistics of the GMM for a given session. These sufficient statistics are then used to extract an i-vector using factor analysis.

A well-matched UBM trained on the text-dependent data supposedly partitions the feature space in a way that different speech units (possibly phoneme or phoneme parts) that are relevant to the task are separated into distinct Gaussians. Once out-of-domain data such as text-independent sessions (NIST) are used to extract sufficient statistics using the said

UBM, other speech units which are unrelated to the task may smooth the more accurate information that is encapsulated in the task-dependent speech units (which probably do exist to some extent in the text independent data).

Therefore, a method that can detect and remove at least some of the unrelated speech units may result in estimated sufficient statistics which better match the characteristics of the text dependent task.

4.2.2. Estimating the sufficient statistics of each Gaussian using its top-N scoring frames

Following the motivation presented above, we estimate the sufficient statistics of each Gaussian (count and sum) by selecting a set of frames with the highest posterior probabilities. In our study the size of the set (N) is set to be equal to 5.

4.2.3. Estimating the sufficient statistics of the whole session using the top-P percentage of scoring frames

The choice of selecting a constant number of top scoring frames for each Gaussian (in the method above) may be suboptimal as the optimal number of top-scoring framing should probably be set adaptively. Alternatively we propose the following method: For each session we select only a percentage (P) of the frames to be used for estimation of the sufficient statistics. The hope is that those frames are more related to the speech units used in the text dependent task. Again, the criterion for selection of the frames is according to the posterior probability, this time the posterior is calculated with respect to the whole UBM.

5. Results

The results reported are on pooled same gender and cross gender trials. Both channel matched (landline only or cell-phone only) and channel mismatched (landline vs. cell-phone) trials are pooled. The standard evaluation set (WF_550_4) consists of 17994 target trials and 81720 imposter trials. For reference, our GMM-NAP system obtains an EER of 1.6% on the standard WF partition (using no NIST data).

5.1. Building some of the components of the i-vector system on the text-dependent (WF) data

Tables 1 and 2 present results for building some of the components of the i-vector system from the WF development sets. Table 1 reports results for the official partition, and table 2 reports results for the partition using 550 speakers for development and 200 for evaluation. As can be seen in both Tables, the best results are achieved when the whole training is done on the WF development set. Interestingly, just training the UBM on the WF data seems to give a large improvement compared to baseline NIST-based build. This is mostly beneficial for small development sets which cannot be used to train the i-vector extractor and the LDA+WCCN components (such as the small development set with 100 speakers with 1 session each we evaluated in [1] and is claimed to be a representative of many real-life datasets obtained from clients).

Table 1. A comparison of methods for using WF and NIST data for training an i-vector system using the standard partition. WF_200_4 is used for score normalization. Results are in EER (in %).

UBM	I-vector extractor	LDA+WCCN	EER
NIST	NIST	NIST	2.63
NIST	NIST	WF_200_4	2.73
NIST	WF_200_4	WF_200_4	2.83
WF_200_4	NIST	NIST	2.10
WF_200_4	WF_200_4	WF_200_4	1.97

Table 2. A comparison of methods for using WF and NIST data for training an i-vector system using the WF_500_4 partition. WF_500_4 is used for score normalization. Results are in EER (in %).

UBM	I-vector extractor	LDA+WCCN	EER
NIST	NIST	NIST	2.82
NIST	NIST	WF_550_4	2.54
NIST	WF_550_4	WF_550_4	1.98
WF_550_4	NIST	NIST	2.17
WF_550_4	WF_550_4	WF_550_4	1.29

Table 3. Performance of WF-only-training of an i-vector system on different sizes of development sets. Results are in EER (in %) and in relative improvement (in %) compared to the baseline system trained on NIST data.

Dev set	Eval set	EER	Rel. imp. compared to a NIST-based system
WF_550_4	WF_200_4	1.29	60
WF_200_4	WF_550_4	1.97	25
WF_100_2	WF_550_4	2.04	22

Table 3 shows a comparison of the performance of the approach of training the whole system on the text-dependent development set as a function of development set size. Note that for the i-vector extractor trained on the WF_100_2 development set we use 200 factors only.

5.2. Transferring the text-independent resources (NIST) to better match the text-dependent (WF) task

Tables 4 presents the results for the methods proposed in subsection 4.2. Comparing the EERs reported in Table 4 (1.92% and 1.9%) to the corresponding EER obtained without the selection process (2.1% in Table 1 for a UBM trained on WF and the i-vector extractor and LDA+WCCN trained on NIST) shows a 10% improvement relative. This improvement is statistically significant with a p-value less than 0.0001. The p-value was calculated by counting the number of unique errors that both the baseline and a proposed system made, and testing the null hypothesis that the two counts were obtained using flips of a balanced coin. It can be seen that the proposed methods perform better than the best

approach found on the previous subsection (which was training the whole system on the WF development set). The DET curves of selected systems are shown in Figure 1.

Moreover, the proposed approaches make a weaker use of the WF data as they use the data for UBM training only. Therefore, the proposed methods can be used even for smaller development sets such as WF_100_1. Table 5 reports the corresponding results for the WF_100_1 development set.

Table 4. Results on the standard WF partition for using the proposed methods for estimating the sufficient statistics for the NIST data. WF_200_4 is used for score normalization. Results are in EER (in %).

UBM	I-vector extractor	LDA+WCCN	EER
WF_200_4	NIST Top-5 frames per Gaussian	NIST Top-5 frames per Gaussian	1.92
	NIST Top-25% frames	NIST Top-25% frames	1.90

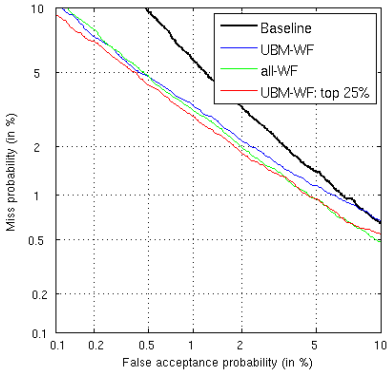

Figure 1: DET curves for the standard WF partition for the baseline system (trained on NIST), the system with the UBM trained on WF, the system trained entirely on WF, and the proposed system described in subsection 4.2 (top-25% selection).

Table 5. A comparison of methods for using WF and NIST data for training an i-vector system using the WF_100_1 partition. WF_100_1 is used for score normalization. Results are in EER (in %).

UBM	I-vector extractor	LDA+WCCN	EER
NIST	NIST	NIST	2.90
WF_100_1	NIST	NIST	2.33
WF_100_1	NIST Top-25% frames	NIST Top-25% frames	2.11

6. Conclusions

In this work we have explored the possibility of building our text-dependent speaker verification i-vector system using both text-dependent data (WF) and text-independent data (NIST).

We have three contributions in this work. First, we have shown that an i-vector-based system may be successfully trained from scratch using relatively small text-dependent datasets (100 speakers with 2 sessions each) and outperform the NIST-based baseline.

Second, we have shown that training the UBM from the text-dependent datasets (while the rest is trained on NIST) gives a significant improvement compared to the NIST-based baseline. This is mostly suitable for very small development datasets or unlabeled development datasets.

Third, a first step towards adapting the NIST data to the text-dependent task was proposed by modifying the way sufficient statistics are extracted from the NIST data.

The following recipe is suggested: when the size of the text-dependent development data is large enough, good performance may be obtained by training the i-vector system from scratch (UBM, i-vector extractor, LDA+WCCN, score normalization). When only small text-dependent development sets are available, they should be used for UBM training and score normalization, and the text-independent data should be used to train the i-vector extractor and LDA+WCCN. Improved performance may be obtained by filtering the text-independent data to better match the text-dependent task.

Overall, using the described techniques the EER was reduced by 27% on the standard WF setup compared to the baseline NIST-based training, and the i-vector-based EER (1.9%) is now closer to the EER we achieve using the GMM-NAP system (1.6%). For the large development set (WF_550_4), EER was reduced by 54% compared to the baseline. For the small development set (WF_100_1), the proposed method (EER=2.11%) outperforms the GMM-NAP system (EER=2.25%).

7. Acknowledgements

This work was part of the SMART EU project, partly funded by the European Commission in the scope of the 7th ICT framework.

The author wishes to thank Wells Fargo for collecting and providing the data for the feasibility study.

8. References

[1] H. Aronowitz, R. Hoory, J. Pelecanos, D. Nahamoo, "New Developments in Voice Biometrics for User Authentication", in Proc. Interspeech, 2011.

[2] H. Aronowitz, "Text Dependent Speaker Verification Using a Small Development Set", in Proc. *Speaker Odyssey*, 2012.

[3] H. Aronowitz, O. Barkan, "New Developments in Joint Factor Analysis for Speaker Recognition", in Proc. *Interspeech*, 2011.

[4] H. Aronowitz. O. Barkan, "Efficient Approximated I-Vector Extraction", in Proc. *ICASSP*, 2012.

[5] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis For Speaker Verification," IEEE TSALP, vol. 19, no. 4, pp. 788 - 798, 2010.

[6] H. Aronowitz, D. Irony, F. Burshtein, "Modeling Intra-Speaker Variability for Speaker Recognition", in Proc. Interspeech, 2005.

[7] H. Aronowitz, "Speaker Recognition using Kernel-PCA and Intersession Variability Modeling", in Proc. *Interspeech*, 2007.

[8] Y. A. Solewicz, H. Aronowitz, "Two-Wire Nuisance Attribute Projection", in Proc. *Interspeech*, 2009.

[9] W. Campbell, Z. Karam, "Simple and Efficient Speaker Comparison using Approximate KL Divergence", in Proc. *Interspeech*, 2010.

[10] W. M. Campbell, et al., "SVM based Speaker Verification using GMM Supervector Kernel and NAP Variability Compensation", in Proc. *ICASSP*, 2006.

[11] W. M. Campbell, D. E. Sturim, P. A. Torres-Carrasquillo, D. A. Reynolds, "A Comparison of Subspace Feature-Domain Methods for Language Recognition", in Proc. Interspeech, 2008.