TEXT-DEPENDENT GMM-JFA SYSTEM FOR PASSWORD BASED SPEAKER VERIFICATION

Sergey Novoselov[1], *Timur Pekhovsky*[1,2], *Andrey Shulipa*[1], *Alexey Sholokhov*[1,2]

[1]Speech Technology Center Ltd., St. Petersburg, Russia
[2]ITMO University, Russia
{novoselov, tim, shulipa, sholohov}@speechpro.com

ABSTRACT

We propose a new State-GMM-supervector extractor for solving the problem of text-dependent speaker recognition. The proposed scheme for supervector extraction makes it easy to implement a text-dependent JFA system for passphrase verification. We examine the conditions of both a global and a text-prompted passphrase. The experiments conducted on the Wells Fargo Bank speech database show that the proposed method makes it possible to create more accurate statistical models of speech signals and to achieve a 44% relative reduction of EER compared to the best state-of-the-art systems of text-dependent verification for a text-prompted passphrase.

Index Terms – speaker recognition, NAP, SVM, JFA, UBM, GMM, HMM, supervector.

1. INTODUCTION

As demonstrated by recent publications, substantial success of state-of-the-art text-dependent verification systems is mostly due to the progress of text-independent speaker recognition.

For example, [1-3] use such widely known paradigms as GMM-UBM (*Gaussian Mixture Model-Universal Background Model*), GMM mean supervector and its MAP (*Maximum A Posteriori*) adaptation to the speaker model [4]. The idea of hybrid GMM/SVM (*Support Vector Machine*) [5] systems is also efficiently adapted. These systems use WCCN (*Within-Class Covariance Normalization*), NAP (*Nuisance Attribute Projection*) or LDA (*Linear Discriminant Analysis*) projections of GMM mean supervectors for compensation of channel effects.

The JFA (*Joint Factor Analysis*) method [6-8], is presented in [2] as an attempt to directly apply a traditional text-independent JFA system, trained on large NIST SRE speaker databases, to a text-dependent task.

The results in [3] lead us to conclude that if the task of text-dependent verification has to be performed under the conditions of a matched training set, for instance, the Wells Fargo Bank database, then the most successful of all the above-mentioned systems using traditional GMM mean supervectors is the *Hidden Markov Model* (HMM) – NAP / SVM system.

Moreover, as shown in [9, 10], under the conditions of a matched training set this system outperforms the currently most promising PLDA (*Probabilistic Linear Discriminant Analysis*) systems for text-dependent verification using *i*-vectors.

This work was partially financially supported by the Government of the Russian Federation, Grant 074-U01

In [1-3] the authors do not focus on the causes of the superiority of their HMM-NAP/SVM system to the rest of the systems examined. However, we see two main causes for that.

The first one explains the advantage of a HMM-NAP/SVM system over a text-independent JFA system [2]. The reason is that all experiments in [1-3] were conducted on the matched dataset of the Wells Fargo Bank, where the recording conditions in the test set closely match the recording conditions in the training set [9]. Under these conditions a system with a small number of parameters trained on this training set can outperform a large text-independent JFA system, which has a large number of parameters but many of them are uninformative because they were trained on NIST SRE datasets.

The second one explains the advantage of a HMM-NAP/SVM system over a GMM-NAP/SVM system. In the case when both systems were trained on the same Wells Fargo Bank training dataset, there was a weak overtraining of the HMM-NAP/SVM system because it had more parameters than the GMM-NAP/SVM system.

In this paper we find potential for further improvement of systems that use GMM mean supervectors [1-3] and are trained on the Wells Fargo Bank datasets. We propose a new scheme for supervector extraction that makes it easy to implement the idea of a text-dependent JFA whose parameters depend on the states of the passphrase, in contrast to the text-independent variant [2]. We argue that such strengthening of the model will be especially useful under the above-mentioned conditions of the dataset.

Section 2 provides a description of state-of-the-art text-dependent verification systems. Section 3 contains the description of the proposed systems. Section 4 describes the system parameters that we use, as well as the Wells Fargo Bank databases. In Section 5 we present comparative experiments using the text-dependent protocol of the Wells Fargo Bank and discuss the results. Section 6 concludes the paper.

2. BASELINE SYSTEMS

In this section we present a brief description of the best two state-of-the-art systems [1], [3] for text-dependent speaker verification, which we will further refer to as baseline systems.

2.1. GMM-supervector

In this paper the baseline GMM system was implemented according to [1]. In this case the GMM mean supervector *m* of the passphrase was obtained using relevant MAP adaptation [4] of the speaker-independent UBM of this passphrase. Using the results of [3], we chose the best realization of the baseline GMM system, and ML-trained the UBM of the passphrase on the development set of the Wells Fargo Bank database, rather than use a text-independent UBM trained on large NIST SRE speaker databases.

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

2.2. HMM-supervector

The baseline HMM system was implemented according to [3]. In this system the mean supervector *m* for the passphrase is found using relevant MAP adaptation of a SI-HMM (*Speaker-Independent HMM)*:

$$m_{i,k} = \frac{\tau}{N_{i,k} + \tau}\mu_{i,k} + \frac{F_{i,k}}{N_{i,k} + \tau}, \quad (1)$$

where τ is the relevant factor, μ is the SI-HMM mean supervector, and the 0- and 1-order statistics $N_{i,k}$, $F_{i,k}$ for the i^{th} digit and the k^{th} component of the Gaussian are calculated based on the parameters of the discrete and the continuous parts of the HMM.

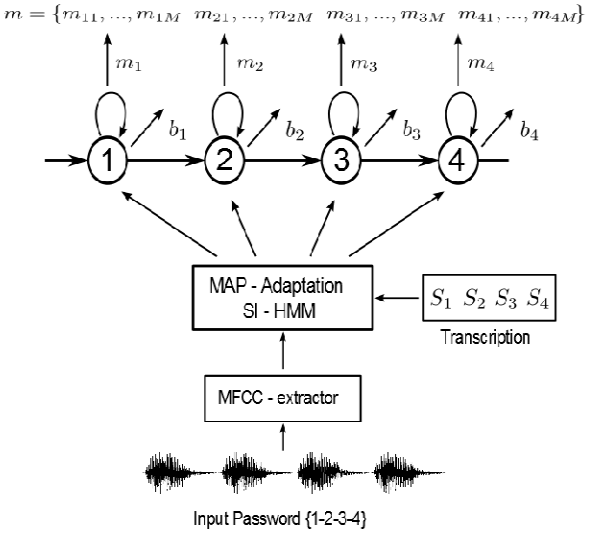

Figure 1. The block diagram of the HMM supervector extraction

Figure 1 shows a diagram of such an HMM extractor [11], where the input is a passphrase consisting of four numbers "1-2-3-4" and its transcription $\{s_1, s_2, s_3, s_4\}$, which is known beforehand. The output is the speaker-dependent SD-HMM mean supervector *m*.

2.3. SVM-kernel, NAP-compensation

In this paper we will consistently follow [1, 3] in using SVM as a classifier in the *m* supervectors space.

In contrast to [12], where NAP projections of GMM mean supervectors *m* are first made for channel effects compensation, and only afterwards used for building KL (*Kullback-Leibler*)-kernel SVM, we will depart from the traditional scheme. Following [3], we will first normalize GMM mean supervectors *m* by multiplying them by mixture weights $\sqrt{\alpha_m}$ and dividing them by the standard deviation σ_m taken from the UBM model, and only then use their NAP projections for a simple linear SVM kernel.

As in [1, 3], SVM scores were normalized using the ZT norm.

3. PROPOSED SYSTEMS

In this section we present our two proposed systems for text-dependent speaker verification.

3.1. DTW-segmentation

The DTW method makes it possible to transfer time labels from a segmented reference speech signal to an unsegmented speech signal with the same text. In order to perform segmentation of the passphrase, the reference signal can be composed of separate words from the full lexicon of passphrase words.

We prepared a full lexicon of states (ten numbers) using manual segmentation for the utterance "0-1-2-3-4-5-6-7-8-9" of one of the speakers from the Wells Fargo dataset. Then the whole Wells Fargo dataset was segmented into these states using the DTW (Dynamic Time Warping) method.

In our experiments we used DTW segmentation only for training and testing our proposed State-GMM systems described below.

3.2. State-GMM-MAP(JFA)-supervector

In contrast to [3], we used extraction of the *State-GMM* mean supervector, demonstrated in Figure 2. Basically, we propose to use its own GMM mean supervector extractor from Section 2.1 for each state.

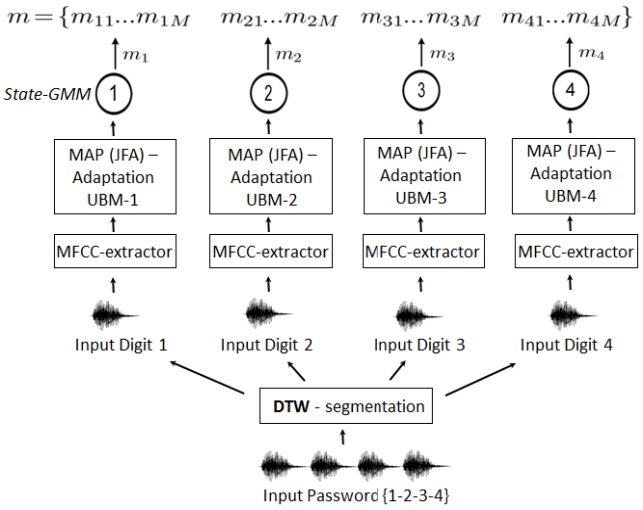

Figure 2. The block diagram of the State-GMM-supervector extraction

Now, for each state, its own speaker-independent UBM must be built from the training set of the Wells Fargo database. The speaker-dependent means of this state are obtained by MAP adaptation of the mean UBMs from the data of a particular speaker. By concatenating the means from all states that were obtained in this way, we have a speaker-dependent GMM mean supervector *m*. In this case the verification system will be called a *State*-GMM-MAP-system. Such a system, split into states, is easy to generalize in case of JFA adaptation. This means that in our other proposed *State*-GMM-JFA-*system* we simply apply the JFA method for each state when extracting the GMM mean supervector. In our

experiments we limited the JFA by the model of eigenvoices, so the JFA-adapted means for the i^{th} state are calculated as follows:

$$m_i = \mu_i + V_i y_i \quad (2)$$

where μ_i is the GMM mean supervector of the UBM of the i^{th} state, y_i is the estimate in the MAP point of the hidden vector of the speaker subspace, V_i is the matrix of the state eigenvoices that was trained on the same dataset as the UBM for the states.

We tried to take into account in JFA the standard member that takes into account the channel effects, but as subsequent experiments showed, this only led to degradation of the *State*-GMM-JFA system.

3.3. S-norm of SVM scores

In this paper we used the S-norm [13] along with the ZT-normalization of scores used in the baseline systems [1, 3].

In our case, if the enrollment is represented by R mean supervectors, the S-normalized estimate is calculated using the Z-normalized distance obtained beforehand $D^{Z-norm}(m_{test}, \Omega_{enroll}^{(R)})$ from the test supervector m_{test} to the multi-session dividing SVM-hyperplane $\Omega_{enroll}^{(R)}$ and the average distance along Z-normalized distances $D^{Z-norm}(m_{enroll}^{(i)}, \Omega_{test})$ from the supervector of the i^{th} enrollment $m_{enroll}^{(i)}$ to the test SVM-hyperplane Ω_{test} :

$$score = \frac{1}{2}\left[D^{Z-norm}(m_{test}, \Omega_{enroll}^{(R)}) + \frac{1}{R}\sum_{i=1}^{R} D^{Z-norm}(m_{enroll}^{(i)}, \Omega_{test}) \right]$$

4. WELLS FARGO DATASET AND EXPERIMENTAL SETTINGS

In our experiments we explored global and text-prompted [1],[3] verification conditions using passphrases consisting of English numbers.

All experiments were conducted on the male part of the speech database collected by Wells Fargo Bank. It consists of recordings of 300 speakers.

We divided it into two parts:

- Development set (100 speakers)
- Evaluation set (200 speakers)

Every speaker has four sessions, two of which correspond to a cellphone recording channel and the other two to a landline recording channel. The dataset collection was accomplished over a period of 4 weeks. For the global passphrase, each session consists of three repetitions of the number sequence "0-1-2-3-4-5-6-7-8-9". For the text-prompted passphrase, each session consists of a one-time utterance of two sequences of four numbers: "2-5-7-4" and "3-5-8-0". In our experiments the total number of target trials was over 10^3, and the total number of impostor trials was over 10^5.

4.1. Front-End Processing

The front-end computes Mel-frequency cepstral coefficients (MFCC), as well as the first derivatives, to yield a 26 dimensional vector per frame. Framing is done every 8 ms using a 16 ms window. An GMM based voice activity detector is used to locate and remove non-speech frames. We also applied a cepstral mean subtraction (CMS) and did not apply Feature Warping for the cepstral coefficients.

4.2. Common experimental settings

For all systems in this paper, matrix dimensionality for the NAP projection in both cases was chosen as 50. For the State-GMM-JFA, 300 eigenvoices were used. In the baseline GMM system we used a GMM order of 64, and in case of the baseline HMM we used GMMs order of 16 as models of emission distributions of probability. We used both State-GMM systems order of 16.

4.3. Global condition systems

Development. To train the baseline GMM and baseline HMM verification systems for the global passphrase condition, we trained UBM and SI-HMM respectively on the development set. The supervectors were obtained using the procedures described in 2.1 and 2.2. To train State-GMM verification systems for the global passphrase condition, we segmented the development set into states (separate numbers), and trained State-UBMs and eigenvoice matrices for each state. In this case the supervectors corresponding to the passphrase "0-1-2-3-4-5-6-7-8-9" were obtained as described in 3.2.

The supervector set of the development data was used as impostors for training the SVM classifiers and calculating the parameters of score normalization.

Enrollment. In case of the global condition, our experiments involved registering the user in the system using three repetitions of the global passphrase from one session. For each repetition in the enroll session of the speaker from the evaluation set, a supervector was extracted (2.1, 2.2, 3.2) and compensation of intra-speaker inter-session variability was performed using a NAP projection (3.3). The three supervectors of the speaker session were used to train the SVM classifier; the parameters of the Z or S normalization were calculated.

Verification. In case of the global condition, verification was performed using one passphrase utterance from the test dataset. The test SVM hyperplane was built using the extracted supervector. S-norm (according to 3.3) or T–norm parameters were calculated.

4.4. Text-prompted condition systems

Development. Training the baseline GMM verification system with a text-prompted passphrase was not different from the case of the global passphrase. To train the State-GMM verification system for the text-prompted passphrase condition, we segmented the development set into states and trained State-UBMs for each state. In this case, prompted mini passphrases consisting of four non-repeating numbers were used for verification. These four numbers were sorted in ascending order, which made it possible to train 210 (the number of combinations of 10 by 4) mini extractor systems for all unique mini passphrases. For each mini system, its own NAP matrix and its own set of vectors of SVM impostors were obtained.

Enrollment. Under the text-prompted condition, user registration in the system was also conducted using three repetitions of the global passphrase from one session. In case of the baseline GMM system, we used a registration procedure as described in 4.3. In case of the State-GMM system, each utterance was segmented into separate words (states). For all repetitions, 210

supervectors were formed using mini extractor systems, according to the possible mini passphrases of four numbers sorted in ascending order. For each of the 210 four-digit mini passphrases, an SVM classifier was trained on three corresponding supervectors (from three repetitions), Z or S normalization parameters were calculated.

Verification. For verification we used two mini-passphrases, the first was "2-5-7-4" and the second was "3-5-8-0". In case of the baseline GMM system, two such consecutive mini passphrases were regarded as one consisting of eight numbers, and verification was conducted as described in 4.3. In case of State-GMM systems, after obtaining the recording of the first mini passphrase, it was segmented and the states were reordered in ascending order according to the numeric sequence, in our case "2-4-5-7". The extractor corresponding to the mini passphrase "2-4-5-7" was chosen for obtaining the supervector. The verification score for the first and the second mini passphrase was calculated according to 4.3. The final score was the result of averaging the estimates for these two mini passphrases.

5. EXPERIMENTS AND DISCUSSION

The goal of the experiments was to compare the baseline systems and the proposed systems on the Wells Fargo dataset. In this paper we limit ourselves to examining two verification conditions: a global passphrase and a text-prompted passphrase.

Tables 1, 2 demonstrate the Equal Error Rate (EER) for the baseline systems, and Tables 3, 4 for the proposed systems.

The following conclusions can be drawn from these tables.

First, the efficiency of the baseline systems that we implemented is close to the results given in [1, 3]. As in [1, 3], in our implementation the baseline HMM system outperforms the baseline GMM. Moreover, our results are close to [1, 3] for another comparison as well: EER in our experiment is approximately two times lower for the matched channel than for the mismatched channel.

Table 1. EER [%] for Baseline GMM-System [1, 3].

GMM-MAP-NAP/SVM	matched channel		mismatched channel	
	ZT-norm	S-norm	ZT-norm	S-norm
Global	1.10	**1.00**	2.00	**1.92**
Text-prompted	**5.03**	5.04	11.61	**11.25**

Table 2. EER [%] for Baseline HMM-System [1, 3].

HMM-MAP-NAP/SVM	matched channel		mismatched channel	
	ZT-norm	S-norm	ZT-norm	S-norm
Global	0.79	0.80	2.03	2.04
Text-prompted	-	-	-	-

Table 3. EER [%] for State-GMM-MAP Proposed System.

State-GMM-MAP-NAP/SVM	matched channel		mismatched channel	
	ZT-norm	S-norm	ZT-norm	S-norm
Global	**0.63**	0.65	**1.44**	**1.44**
Text-prompted	**2.92**	3.15	**6.70**	6.86

Table 4. EER [%] for State-GMM-JFA Proposed System.

State-GMM-JFA-NAP/SVM	matched channel		mismatched channel	
	ZT-norm	S-norm	ZT-norm	S-norm
Global	**0.61**	0.63	**1.55**	1.58
Text-prompted	**2.80**	2.95	**8.00**	8.35

Second, the results show that both our proposed State-GMM models outperform the baseline models [1, 3]. Their superiority is especially obvious under the text-prompted condition, where our State-GMM-JFA system achieves a 44% relative EER reduction compared to the baseline GMM. We tried to use 128 gaussian components in the GMM-baseline, but this led to system degradation. Under the global condition, the reduction is 23% when comparing the State-GMM-JFA with the best baseline HMM system.

It can also be seen that for the proposed systems, using ZT normalization of scores is preferable to the S-norm.

These EER improvements can be explained by a deeper training structure in our scheme (see Figure 2), in which its own UBM and its own JFA model is trained for each state, in contrast to the baseline GMM system where training for a state is performed using MAP adaptation of a speaker-independent SI-GMM passphrase (see Figure 1).

Increasing the number of parameters in our case of the State-GMM-JFA system, compared to the baseline systems, led to a certain extent of overtraining of the State-GMM-JFA. It is especially obvious when switching from the matched channel condition to the mismatched channel condition. After this transition the results of the State-GMM-JFA system become three times worse while those of the baseline GMM system are only two times worse.

6. CONCLUSIONS

The paper proposes a new scheme for extracting State-GMM mean supervectors.

Based on this extractor, we implemented two new systems for text-dependent verification, a State-GMM-MAP and a State-GMM-JFA system, which demonstrated the advantage of the proposed approach compared to baseline systems. Our State-GMM-JFA system achieves a 44% relative EER reduction compared to the best state-of-the-art system for a text-prompted passphrase. The prompted condition has the advantage of robustness to spoofing attacks with recorded speech.

7. ACKNOWLEDGEMENTS

The authors would like to thank Wells Fargo Bank for collecting and providing the database for the 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, O. Barkan, "New Developments in Joint Factor Analysis for Speaker Verification", in Proc. *Interspeech*, 2011.

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

[4] Douglas A. Reynolds, T. F. Quatieri, and R. Dunn, "Speaker verification using adapted Gaussian mixture models", Digital Signal Processing 10, 19–41 (2000)

[5] Belykh, I. N., Kapustin, A. I., Kozlov, A. V., Lohanova, A. I., Matveev, Y. N., Pekhovsky, T. S., Simonchik, K. K., Shulipa, A. K. (2012). Speaker identification system for the NIST SRE 2010. *Informatika i Ee Primeneniya [Informatics and its Applications]*, *6*(1), 91-98.

[6] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A Study of Interspeaker Variability in Speaker Verification," IEEE Transaction on Audio, Speech and Language, vol. 16, no. 5, pp. 980–988, july 2008.

[7] P. Kenny, "Joint factor analysis of speaker and session variability: theory and algorithms", technical report CRIM- 06/08-14, 2006.

[8] Kozlov, A., Kudashev, O., Matveev, Y., Pekhovsky, T., Simonchik, K., Shulipa, A. (2013). SVID Speaker Recognition System for NIST SRE 2012. In *Speech and Computer* (pp. 278-285). Springer International Publishing.

[9] T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. Kockmann and P. Dumouchel, "Text-dependent speaker recognition using PLDA with uncertainty propagation" in Proc. *Interspeech*, 2013.

[10] H. Aronowitz, O. Barkan, "On Leveraging Conversational Data for Building a Text Dependent Speaker Verification System", in Proc. *Interspeech*, 2013.

[11] C. Dong , Y. Dong, J. Li and H. Wang, "Support Vector Machines Based Text Dependent Speaker Verification Using HMM Supervectors", in Proc. *Speaker Odyssey*, 2008.

[12] W. Campbell, D. Sturim, D. Reynolds, A. Solomonoff, "SVM Based Speaker Verification using a GMM Supervector Kernel and NAP Variability Compensation", in Proc. *ICASSP*, 2006

[13] P. Kenny, "Bayesian Speaker Verification with Heavy-Tailed Priors" in Proc. Speaker Odyssey, 2010.