Investigation of Frame Alignments for GMM-based Digit-prompted Speaker Verification

Yi Liu*, Liang He*, Wei-Qiang Zhang*, Jia Liu*, Michael T. Johnson†

* Tsinghua National Laboratory for Information Science and Technology,
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
E-mail: liu-yi15@mails.tsinghua.edu.cn, {heliang, wqzhang, liuj}@tsinghua.edu.cn
† Department of Electrical and Computer Engineering, University of Kentucky,
453 F. Paul Anderson Tower, Lexington, KY 40506-0046, USA
E-mail: mike.johnson@uky.edu

Abstract—**Frame alignments can be computed by different methods in GMM-based speaker verification. By incorporating a phonetic Gaussian mixture model (PGMM), we are able to compare the performance using alignments extracted from the deep neural networks (DNN) and the conventional hidden Markov model (HMM) in digit-prompted speaker verification. Based on the different characteristics of these two alignments, we present a novel content verification method to improve the system security without much computational overhead. Our experiments on the RSR2015 Part-3 digit-prompted task show that, the DNN-based alignment performs on par with the HMM alignment. The results also demonstrate the effectiveness of the proposed Kullback-Leibler (KL) divergence based scoring to reject speech with incorrect pass-phrases.**

I. INTRODUCTION

Automatic speaker verification (ASV) has developed rapidly in the last decade. ASV can be broadly categorized into text-independent and text-dependent applications. In text-independent ASV, the system verifies the speaker's identity using spontaneous speech, which is valuable for military and forensic tasks. Text-dependent ASV, on the other hand, requires users to utter a specific pass-phrase, and is commonly used in commercial applications.

Although text-independent speaker verification is much more flexible, text-dependent ASV is more suitable for applications which require higher security. The pass-phrases in text-dependent speaker verification can be fixed or prompted during enrollment and test, with text-prompted the more popular approach in recent years [1]. Text-prompted systems requires the user to both speak the correct text and be validated as the target speaker. For instance, in digit-prompted ASV, users are required to speak different digit strings. By verifying the text contents, text-prompted speaker verification provides users an additional protection and is more robust to replay spoofing attacks [2]. In this paper, we focus on the digit-prompted case and investigate both the speaker and content verification.

By virtue of a constrained vocabulary, digit-prompted speaker verification usually delivers better performance for short utterances. Many modeling methods have been developed for digit-prompted speaker verification. The most commonly used approach is based on Gaussian mixture models (GMMs), motivated by text-independent speaker verification.

Universal background model (UBM) based maximum a posteriori (MAP) [3], joint factor analysis [4], i-vector with within class covariance normalization (WCCN) [5] and probabilistic linear discriminant analysis (PLDA) [6] have all been applied to this field.

The calculation of the frame posterior (often referred as *frame alignment*) plays an essential role in these GMM-based models. The frame alignment $P(s|\boldsymbol{x}_t)$ is the posterior probability that speech frame $\boldsymbol{x}_t$ belongs to a phonetic unit s. Depending on the alignment methods, the phonetic units can either be the GMM components generated by unsupervised clustering or be assigned to the classes used in speech recognition (e.g., monophones, monophone states or senones). With more accurate frame alignment, the speaker verification system can better model the phonetic units, and effectively compare the features belonging to the same feature subspace. In text-independent speaker verification, researchers have used senones predicted by phonetically-aware deep neural networks (DNNs) to generate frame alignments and improved the performance significantly [7].

In digit-prompted speaker verification, if we assume all users speak the correct digit strings, the transcriptions of the utterances are known beforehand. As in speech recognition, Viterbi forced alignment based on hidden Markov models (HMMs) is a natural choice to obtain the alignments [3]. Using the speech content, the HMM alignment provides accurate results even in adverse acoustic environments. However, the user may incidentally misread the prompted text. This mismatch will severely impact the quality of the HMM alignment.

As an alternative to the HMM alignment, conventional DNN alignment can also be applied to extract the posteriors in digit-prompted speaker verification. It has been found that i-vector modeling using DNN alignment achieves good performance on RSR2015 Part-3 evaluation [8]. In [9], [10], the authors showed that DNN alignments outperformed HMMs on both RSR2015 and RedDots [11]. Note that the quality of the DNN alignment is influenced by the acoustic environment rather than the transcriptions.

In this paper, we first compare the performance of HMM and DNN alignments, using both GMM-MAP and i-vector modeling, on the RSR2015 Part-3 digit-prompted task. To

better understand the difference between these two alignments, the HMM and DNN are trained on the same dataset and based on the same phonetic units. Due to the limited number of phonetic units in the digit-prompted verification, we use a phonetic GMM (PGMM) approach [12] for both GMM-MAP and i-vector modeling, keeping roughly the same number of Gaussian mixtures across the different systems.

The assumption that all the input utterances contain the correct contents is unrealistic. Attackers can record the speech of a target user, and malicious replay attacks often present the speech with wrong text content. In fixed-phrase text-dependent ASV, techniques like keyword spotting or wakeup word detection [13] can be applied to verify the content. While on the digit-prompted condition, speech recognition is the most commonly used method, which introduces extra complexity in the computation and the system deployment. Since the HMM and DNN alignments exhibit complementary properties, the sequence information in both alignments can be considered together. In this paper, we propose a novel Kullback-Leibler (KL) divergence based scoring to verify the text content without speech recognition decoding. This algorithm involves low computational overhead and is especially suitable for the embedded devices. The effectiveness of this method is validated by our experiments.

The organization of this paper is as follows. Modeling methods based on the HMM alignment are briefly introduced in Section 2. Section 3 describes PGMM using the DNN alignment. Then, based on these alignment approaches, a fast and efficient algorithm to verify the utterance content is proposed in Section 4. Experimental setup and results are presented in Section 5 and 6. Finally, Section 7 concludes the paper.

II. The Role of HMM

In GMM-based speaker verification, a frame alignment is first generated and then used to calculate the Baum-Welch statistics. In digit-prompted speaker verification, the Viterbi or forward-backward (FB) algorithm is a common choice to align speech frames to the HMM phonetic units. On the digit-prompted condition, each digit is treated as a whole-word model. In this paper, we use N-state HMMs to represent ten digits plus silence.

Given a transcription, a directed graph of HMM states is first compiled. The Viterbi or FB algorithm uses this graph to find a state path that optimally fits the feature sequence. In Viterbi forced alignment, frame $\boldsymbol{x}_t$ is aligned to the most likely state q_t, i.e. $P(s|\boldsymbol{x}_t) = 1$ if and only if $s = q_t$. The FB alignment, which can be seen as a *soft* version of the Viterbi algorithm, computes the posterior from forward and backward probabilities [14]. In [15], it was shown that these two types of alignments result in similar performance. To better compare the alignments generated by HMM and DNN, the soft FB alignment is used in our experiments.

In GMM-HMM, the state s is modeled by GMM $\lambda_s = \{w_{s,c}, \boldsymbol{\mu}_{s,c}, \boldsymbol{\Sigma}_{s,c}\}$, which is treated as the universal background model (UBM), as in the text-independent speaker

verification. Frames are aligned to different Gaussian mixtures of states using HMM alignment $P(s|\boldsymbol{x}_t, h)$

$$\gamma_{s,c,t}^{\text{HMM}} = P(s|\boldsymbol{x}_t, h)P(c|\boldsymbol{x}_t, \lambda_s) \tag{1}$$

where $\gamma_{s,c,t}^{\text{HMM}}$ is the posterior of $\boldsymbol{x}_t$ occupying the c-th mixture of state s, h represents the HMM, $P(c|\boldsymbol{x}_t, \lambda_s)$ is the Gaussian posterior of component c in GMM λ_s.

The normalized Baum-Welch statistics using the GMM-HMM alignment are

$$N_{s,c} = \sum_t \gamma_{s,c,t}^{\text{HMM}} \tag{2}$$

$$\bar{\boldsymbol{F}}_{s,c} = \sum_t \gamma_{s,c,t}^{\text{HMM}}(\boldsymbol{x}_t - \boldsymbol{\mu}_{s,c}) \tag{3}$$

$$\bar{\boldsymbol{\Sigma}}_{s,c} = \sum_t \gamma_{s,c,t}^{\text{HMM}}(\boldsymbol{x}_t - \boldsymbol{\mu}_{s,c})(\boldsymbol{x}_t - \boldsymbol{\mu}_{s,c})^T \tag{4}$$

To adapt the HMM to a speaker model, GMM-MAP estimation is applied as

$$\hat{\boldsymbol{\mu}}_{s,c} = \alpha_{s,c}\bar{\boldsymbol{F}}_{s,c} + \boldsymbol{\mu}_{s,c} \tag{5}$$

where $\hat{\boldsymbol{\mu}}_{s,c}$ is the mean of the adapted speaker model and $\alpha_{s,c} = 1/(N_{s,c} + r)$, r is the relevance factor [16]. During test, the verification score is the log-likelihood ratio computed against the speaker model and UBM.

In the i-vector modeling, the statistics extracted by (2-4) are used for the total variability space training and i-vector extraction [17]. The flowchart of GMM-MAP and i-vector using the GMM-HMM alignment is illustrated in Fig. 1a. In the figure, speaker and speech features indicate the features more suitable for speaker and speech recognition, respectively.

In DNN-HMM, each state is modeled by a DNN output. The state posterior $P(s|\boldsymbol{x}_t, h)$ is calculated based on the log-likelihoods computed by a DNN rather than several GMMs. In this case, the within-state posterior $P(c|\boldsymbol{x}_t, \lambda_s)$ cannot be calculated directly. It should be estimated using the method introduced in the next section.

III. PGMM with DNN-based Alignment

In conventional text-independent speaker verification, the outputs of the DNN are usually senones modeled by a single Gaussian [7]. However, in digit-prompted applications, the number of states is much smaller than for the text-independent case due to the limited vocabulary [3], [8]. To increase the modeling capability of the DNN alignment, a DNN-based phonetic GMM (PGMM) is used in this paper. Each state in the PGMM is represented by a GMM rather than a single Gaussian. The idea of PGMM was first proposed in [18], and was once applied to DNN/i-vector framework in [19].

To initialize the model, each feature is first hard aligned to one states with the maximum posteriors. The initial state GMMs are then trained individually. Let τ denote the DNN model, and $\gamma_{s,c,t}^{\text{DNN}}$ is the posterior of $\boldsymbol{x}_t$ occupying the c-th mixture of DNN state s. Using the DNN alignment these GMMs can be trained by EM algorithm and the sufficient statistics are accumulated by (6-10).

$$\gamma_{s,c,t}^{\text{DNN}} = P(s|\boldsymbol{x}_t, \tau)P(c|\boldsymbol{x}_t, \lambda_s) \tag{6}$$
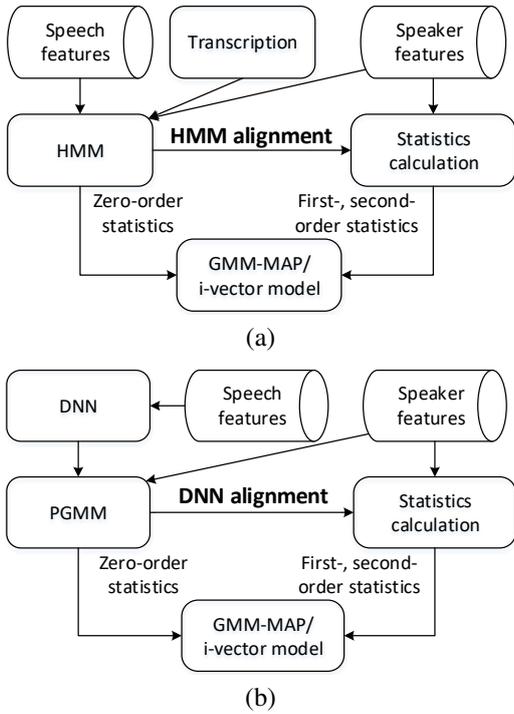
Fig. 1: The flow diagrams of digit-prompted speaker verification systems based on (a) HMM alignment and (b) DNN alignment.

$$N_{s,c} = \sum_t \gamma_{s,c,t}^{\text{DNN}} \tag{7}$$

$$w_{s,c} = N_{s,c} / \sum_c N_{s,c} \tag{8}$$

$$\boldsymbol{\mu}_{s,c} = \frac{1}{N_{s,c}} \sum_t \gamma_{s,c,t}^{\text{DNN}} \boldsymbol{x}_t \tag{9}$$

$$\boldsymbol{\Sigma}_{s,c} = \frac{1}{N_{s,c}} \sum_t \gamma_{s,c,t}^{\text{DNN}} (\boldsymbol{x}_t - \boldsymbol{\mu}_{s,c})(\boldsymbol{x}_t - \boldsymbol{\mu}_{s,c})^T \tag{10}$$

The parameters of PGMM are updated iteratively. The PGMM algorithm can also be used in DNN-HMM to calculate the within state posterior, by replacing the DNN posteriors $P(s|\boldsymbol{x}_t, \tau)$ with the alignment posteriors $P(s|\boldsymbol{x}_t, h)$.

As with the HMM alignment, the posteriors $\gamma_{s,c,t}^{\text{DNN}}$ calculated from (6) are used to extract the statistics. GMM-MAP and i-vector modeling are applied in the same manner as described in Section 2, expect for the different alignment sources. The procedure is demonstrated in Fig. 1b.

IV. FAST CONTENT MATCHING

In real-world applications, speaker verification systems face many kinds of attacks [2]. Replay is a form of low-cost spoofing attack where an adversary claims to be a target speaker using a recorded speech sample. If the text is prompted in the runtime, we can prevent this replay attack effectively, since these samples are often captured surreptitiously and the attacker cannot carefully control the content. Hence the content

matching is important to enhance the security in commercial applications.

However, most published papers about text-prompted speaker verification do not focus on content verification, often treating this as a separated speech recognition task. When pass-phrases are fixed during enrollment and test, some techniques, e.g., query-by-example (QbE) keyword spotting [13] or dynamic time wrapping (DTW) [20], [21], can be used to compare the contents. However, these techniques cannot be easily applied to the digit-prompted case considering that the pass-phrases in the enrollment and test differ from each other. Another option to do the content verification is to insert a speech recognition system [3]. The disadvantage of this method is that the decoding procedure costs significant time and increases the model complexity.

In the PGMM framework, the DNN is trained based on the HMM alignment. Although they share the same set of phonetic units, they generate the alignments from different perspectives. The HMM is capable to find the globally optimal alignment given the transcription. If the transcription is consistent with the speech content (which means the user speaks the correct prompt), the HMM generates the alignment with good quality. If the content is inconsistent, the alignment becomes incorrect respectively. In contrast, the DNN aligns the frames only based on the local acoustic features and will not be impacted by any transcriptions. Considering these different characteristics of the two alignments, we use the DNN alignment as a reference and propose a fast and efficient scoring method to verify the text content without any decoder.

The alignments are posteriors from the HMM and the DNN. The deviation between these posteriors can be defined by Kullback-Leibler (KL) divergence:

$$\text{KL} = \frac{1}{T} \sum_{t=1}^{T} \sum_p \gamma_{p,t}^{\text{HMM}} \log \left(\gamma_{p,t}^{\text{HMM}} / \gamma_{p,t}^{\text{DNN}} \right) \tag{11}$$

where T is the total number of frames and p denotes the phonetic class which will be explained later.

The KL divergence in (11) should be small when the speech is correctly uttered, since the HMM and DNN alignments both exhibit the genuine phonetic sequence. Biases will be introduced into the HMM alignment when the transcription is uttered incorrectly, making the alignment sequence deviate from that of the DNN. The mismatch leads to a increase in the KL divergence. Thus, the deviation between these two posteriors becomes a metric to detect speech with wrong text.

The straightforward definition of a phonetic class p in digit-prompted speaker verification is a single state s of a digit, but we find it is always beneficial to involve a broader class, i.e., $p = \{s | s \in d_p\}$, d_p is a digit. The rational is easy to understand. Since the DNN alignment is noisy and can be harassed by the adverse acoustic environment, the broader definition improves the robustness to some alignment errors.

Given the definition, the posterior of a phonetic class p at

time t is expressed as:

$$\gamma_{p,t}^{\{\cdot\}} = \sum_{s \in p} \sum_c \gamma_{s,c,t}^{\{\cdot\}} \qquad (12)$$

where $\{\cdot\}$ is HMM or DNN respectively.

Note that the KL divergence is defined only if $\gamma_{p,t}^{\text{DNN}} = 0$ implies $\gamma_{p,t}^{\text{HMM}} = 0$. The HMM and DNN alignments are sparse, and may violate this condition. To alleviate this problem, we adjust the both posteriors by

$$\gamma'_{p,t} = \frac{\gamma_{p,t} + \epsilon}{\sum_p (\gamma_{p,t} + \epsilon)} \qquad (13)$$

where $\epsilon = 10^{-5}$ is a small constant.

By this method, the content verification becomes simply a byproduct of the speaker verification. An additional HMM alignment is the only extra step to generate a content verification score, after the DNN posteriors computation. Alignment algorithm based on Viterbi or FB is much faster and easier to implement than a speech recognition decoder [22]. Unlike QbE or DTW, which only works for the fixed-phrase condition, KL divergence scoring can be applied to content verification with arbitrary text. This light-weight algorithm is suitable for embedded devices whose power and computing capability is limited.

V. EXPERIMENTAL SETUP

A. Data

The experiments are carried out on RSR2015 Part-3, which is a digit-prompted task. Digit strings are prompted during enrollment and test. The enrollment data for a speaker contains three ten-digit utterances, and the test utterance is a five-digit sequence. The background and development sets (about 22 hours) are used for the gender-independent UBM, DNN, PGMM and i-vector training. The evaluation set contains 57 male and 49 female speakers. The detailed statistics of this dataset can be found in [1]. The trials can be partitioned into four categories: (1) target speaker produces correct content (TC), (2) target speaker produces wrong content (TW), (3) imposter speaker produces correct content (IC), and (4) imposter speaker produces wrong content (IW). Only the first TC combination is accepted by the digit-prompted system, while the other three categories are all non-target trials. Equal error rate (EER) and minimum decision cost function (minDCF) in SRE08 [23] and SRE10 [24] are used for evaluation. For speaker verification, we investigate the performance of TC-IC (i.e. compute the metrics among TC and IC trials), and the performance of TC-TW is taken into account when we check the effectiveness of the proposed fast content verification method. We do not report TC-IW, because it is the easiest task that all systems achieve relatively good performance.

B. Models

- **DNN**: The DNN in this paper consists of 4 fully connected layers with 512 nodes per layer. The input is a vector of 120-dimensional FBank (40 filter-bank energies plus delta/delta-delta) features with symmetric 5-frame

expansion, resulting in 11 frames in total. The number of output nodes is 33, which equals to the state number in our HMM. The training data is generated by the HMM-based Viterbi alignments. This DNN is also used in the following DNN-HMM model.

- **HMM**: The GMM-HMM and DNN-HMM used in our experiment is trained using HTK [22]. Each word (ten digits plus silence) is modeled by a 3-state HMM. In GMM-HMM, each state is represented by a 16-component GMM. 60-dimensional MFCC features (20 static + delta/delta-delta) with CMVN are used to train the GMM-HMM. The MFCC features are also used in all models other than the DNN.

- **Conventional GMM-MAP/i-vector**: A gender-independent UBM with 512 mixtures is trained. The relevance factor is 5.0 in the GMM-MAP system. For i-vector modeling, the rank of the i-vector subspace matrix is 400. LDA, length normalization and PLDA are applied to score the trials.

- **PGMM**: For DNN and DNN-HMM alignments, the PGMM model is used to calculate the statistics. States corresponding to silence are ignored. Each state is represented by a 16-component GMM. The total number of mixtures in the PGMM is 480. All other configurations are the same as the conventional GMM-MAP/i-vector.

VI. RESULTS

A. Comparison of different alignments

The speaker verification performance of GMM-MAP and i-vector on TC-IC trials are presented at the top and bottom parts of Table I. Within all the alignments, we find that GMM-MAP consistently outperforms i-vector. Since i-vector is only the point estimation of the latent factor in the total variability subspace, it becomes less reliable due to the large variance when the duration is too short and it cannot achieve good performance in this task. Also as expected, frame alignments calculated on the unsupervised GMM perform the worst in both modeling methods.

TABLE I: Performance (EER(%)/minDCF08/minDCF10) of GMM-MAP and i-vector using different frame alignments. Only TC-IC is reported.

Models	male	female
	TC-IC	TC-IC
GMM-MAP	3.38 / 0.0165 / 0.6607	3.36 / 0.0161 / 0.4951
DNN/GMM-MAP	2.08 / 0.0115 / 0.5307	**2.68 / 0.0129 / 0.4525**
HMM/GMM-MAP	2.20 / 0.0116 / 0.5128	3.16 / 0.0140 / 0.4676
DNN-HMM/GMM-MAP	**2.06 / 0.0112 / 0.5173**	2.95 / 0.0137 / 0.4565
i-vector	3.49 / 0.0174 / 0.5686	3.50 / 0.0176 / 0.5467
DNN/i-vector	**2.74 / 0.0148 / 0.5357**	**3.10 / 0.0169 / 0.5740**
HMM/i-vector	2.77 / 0.0162 / 0.6583	3.10 / 0.0170 / 0.6154
DNN-HMM/i-vector	2.71 / 0.0162 / 0.6045	3.34 / 0.0169 / 0.5988

In Table I, it is interesting to find that the DNN-based alignment achieves results on par with the HMM-based alignments. Actually, it outperforms GMM-HMM on many conditions. Even though DNN-HMM can achieve a better alignment quality than GMM-HMM theoretically, it fails to beat the DNN

alignment as well. The results are consistent with [10]. Unlike [10], we use the same dataset to train the HMM and DNN and the basic phonetic units are also the same. The comparison should be fair in this case.

This experiment indicates that the extra text information adds little to speaker verification results on RSR2015 Part-3. We hypothesize the reason is that, the local information used in the DNN aligns the frames to the phonetic units well, while the HMM alignment are likely to be disturbed. For example, users may mispronounce the phones in the training data which leads to poor HMM alignments.

B. Reject wrong text

Although the HMM alignment does not boost the performance of speaker verification, the prompted pass-phrases can still help to improve the security. The KL divergence in (11) is used to reject non-matched content. GMM-HMM and DNN-HMM are two sources of HMM alignments while the DNN provides the reference. As a comparison, a speech recognition decoder is implemented by HTK and the method in [3] is used as the baseline. For speech recognition decoder, the acoustic model is the GMM-HMM used in the last experiment, and the language model is a word-loop constructed by ten digits plus silence. This decoding-based text verification is also shown in [3].

As described in Section 4, two different phonetic classes are investigated. The *state-level* in Table II denotes that the KL divergence is computed between states, while the states belonging to the same digit are clustered as one phonetic class at the *digit-level* systems. Results on the TC-TW trials are shown in Table II.

TABLE II: Performance (EER(%)/minDCF08/minDCF10) of different content verification methods. Results are reported on TC-TW trial.

Methods	male	female
	TC-TW	TC-TW
Decoding [3]	0.25 / 0.0010 / 0.0704	0.05 / 0.0004 / 0.0394
DNN+GMM-HMM (state)	0.74 / 0.0041 / 0.1721	0.37 / 0.0027 / 0.1303
DNN+DNN-HMM (state)	0.40 / 0.0016 / 0.0699	0.03 / 0.0003 / 0.0312
DNN+GMM-HMM (digit)	0.30 / 0.0016 / 0.0566	0.03 / 0.0002 / 0.0250
DNN+DNN-HMM (digit)	**0.25 / 0.0012 / 0.0587**	**0.02 / 0.0002 / 0.0210**

As shown in Table II, the decoding-based method performs well in this task. Meanwhile, the proposed KL divergence based scoring is also an efficient way to verify the text content. Compared to the decoding-based method, the KL divergence achieves competitive performance on both male and female trials. Among all the results, scoring between DNN and DNN-HMM performs better. The reason is that when aligning frames using DNN-HMM, we use the posteriors generated by the DNN, making their posteriors more consistent with each other. Thus the deviation for matched speech can be minimized.

Different phonetic classes are also investigated. In practice, we observed that the DNN alignment can differ from the HMM alignment due to the adverse acoustic environment and the mispronunciation. A broader phonetic class is more robust to this disturbance. It is presented in Table II that, compared to the state-level class, using digits as the phonetic classes illustrates better performance. Overall, the digit-level KL divergence scoring using DNN and DNN-HMM alignments achieves the best performance.

VII. CONCLUSIONS

In this paper, we first compare several speaker verification methods implemented on the RSR2015 Part-3 digit-prompted task. The experiments show that GMM-MAP outperforms vanilla i-vector on this task. In both modeling methods, posteriors calculated on DNN and HMM achieve better performance than unsupervised GMM. Alignments based on DNN generally exhibit better performance than GMM-HMM alignments. The use of DNN-HMM, which is a common practice in speech recognition, does not lead to further improvements. The results demonstrate the effectiveness of DNN alignment in digit-prompted speaker verification. In contrast, the use of HMM seems to be affected by inaccurate pronunciation and mismatched text and do not outperform the DNN alignment.

Considering the characteristics of these different alignments, we then propose a novel algorithm to further verify the text content. The KL divergence between the DNN- and HMM-aligned posteriors efficiently conveys the content matching result, especially when DNN and DNN-HMM is used. The digit-level KL divergence scoring with broader phonetic classes is applied to further improve the robustness. This method is useful under both fixed- and prompted-phrase conditions. Although the speech recognition decoder can also be used, we note that this KL divergence scoring is much faster and easier to implement.

In the future, we will explore new methods to fully utilize the known transcriptions in the text-prompted task. Also, we will develop the KL divergence based method to verify trials with a larger vocabulary.

REFERENCES

[1] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Text-dependent speaker verification: Classifiers, databases and RSR2015," *Speech Communication*, vol. 60, pp. 56 – 77, 2014.

[2] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: A survey," *Speech Communication*, vol. 66, pp. 130 – 153, 2015.

[3] G. Wang, K. A. Lee, T. H. Nguyen, H. Sun, and B. Ma, "Joint speaker and lexical modeling for short-term characterization of speaker," in *Interspeech*, 2016, pp. 415–419.

[4] T. Stafylakis, P. Kenny, M. J. Alam, and M. Kockmann, "Speaker and channel factors in text-dependent speaker recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 1, pp. 65–78, 2016.

[5] H. Zeinali, H. Sameti, and L. Burget, "HMM-based phrase-independent i-vector extractor for text-dependent speaker verification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 7, pp. 1421–1435, 2017.

[6] T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. Kockmann, and P. Dumouchel, "Text-dependent speaker recognition using PLDA with uncertainty propagation," in *Interspeech*, 2013, pp. 3684–3688.

[7] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Interspeech*, 2014, pp. 1714–1718.

[8] J. Zhong, W. Hu, F. Soong, and H. Meng, "DNN i-vector speaker verification with short, text-constrained test utterances," in *Interspeech*, 2017, pp. 1507–1511.

[9] H. Zeinali, L. Burget, H. Sameti, O. Glembek, and O. Plchot, "Deep neural networks and hidden markov models in i-vector-based text-dependent speaker verification," in *Odyssey-The Speaker and Language Recognition Workshop*, 2016, pp. 24–30.

[10] H. Zeinali, H. Sameti, L. Burget, and J. H. Cernocky, "Text-dependent speaker verification based on i-vectors, neural networks and hidden markov models," *Computer Speech and Language*, vol. 46, no. "", pp. 53 – 71, 2017, "".

[11] K. A. Lee, A. Larcher, G. Wang, P. Kenny, N. Brummer, D. Leeuwen, H. Aronowitz, M. Kockmann, C. Vaquero, B. Ma, H. Li, T. Stafylakis, J. Alam, A. Swart, and J. Perez, "The reddots data collection for speaker recognition," in *Interspeech*, 2015, pp. 2996–3000.

[12] Y. Tian, L. He, M. Cai, W. Q. Zhang, and J. Liu, "Deep neural networks based speaker modeling at different levels of phonetic granularity," in *ICASSP*, March 2017, pp. 5440–5444.

[13] Guoguo Chen, Carolina Parada, and Tara N Sainath, "Query-by-example keyword spotting using long short-term memory networks," in *ICASSP*. IEEE, 2015, pp. 5236–5240.

[14] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, Feb 1989.

[15] Y. Liu, L. He, Y. Tian, Z. Chen, J. Liu, and M. T. Johnson, "Comparison of multiple features and modeling methods for text-dependent speaker verification," in *ASRU*, 2017, (in press).

[16] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1, pp. 19 – 41, 2000, "".

[17] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, May 2011.

[18] A. Park and T. J. Hazen, "Asr dependent techniques for speaker recognition," in *ICSLP*. 2002, pp. 1337–1340, Massachusetts Institute of Technology.

[19] Y. Tian, L. He, M. Cai, W. Zhang, and J. Liu, "Deep neural networks based speaker modeling at different levels of phonetic granularity," in *ICASSP*, 2017, pp. 5440–5444.

[20] S. Dey, S. Madikeri, M. Ferras, and P. Motlicek, "Deep neural network based posteriors for text-dependent speaker verification," in *ICASSP*, 2016, pp. 5050–5054.

[21] S. Dey, P. Motlicek, S. Madikeri, and M. Ferras, "Exploiting sequence information for text-dependent speaker verification," in *ICASSP*, March 2017, pp. 5370–5374.

[22] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, and Others, "The HTK book," *Cambridge university engineering department*, 2002.

[23] A. F. Martin and C. S. Greenberg, "NIST 2008 speaker recognition evaluation: Performance across telephone and room microphone channels," in *Interspeech*, 2009, pp. 2579–2582.

[24] A. F. Martin and C. S. Greenberg, "The NIST 2010 speaker recognition evaluation," in *Interspeech*, 2010, pp. 2726–2729.