Exploring Robustness of DNN/RNN for Extracting Speaker Baum-Welch Statistics in Mismatched Conditions

Hao Zheng, Shanshan Zhang, Wenju Liu

National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, P.R.China

hzheng@nlpr.ia.ac.cn, shanshan.zhang@ia.ac.cn, lwj@nlpr.ia.ac.cn

Abstract

This work explores the use of DNN/RNN for extracting Baum-Welch sufficient statistics in place of the conventional GMM-UBM in speaker recognition. In this framework, the DNN/RNN is trained for automatic speech recognition (ASR) and each of the output unit corresponds to a component of GMM-UBM. Then the outputs of network are combined with acoustic features to calculate sufficient statistics for speaker recognition. We evaluate and analyze the performance of networks with different configurations and training corpuses in this paper. Experimental results on text-independent SRE NIST 2008 and text-dependent RSR2015 speaker verification tasks show the robustness of DNN/RNN for extracting statistics in mismatched evaluation conditions compared with GMM-UBM system. Particularly, Long Short-Term Memory (LSTM) RNN realized in this work outperforms traditional DNN and GMM-UBM in most mismatched conditions.

Index Terms: DNN, RNN, speaker recognition, mismatched condition

1. Introduction

The Gaussian mixture model (GMM) universal background model (UBM) has become the dominant approach for front-end modeling in speaker recognition applications for over ten years [1]. In the GMM-UBM framework, the UBM is a large GMM to represent the speaker-independent distribution of features trained with speech samples from a large number of speakers, without any phonetic information provided. For a given utterance, the posteriors of GMM components are combined with acoustic features (e.g., mel-frequency cepstral coefficients (MFCC)) to calculate the Baum-Welch statistics for standard vector-based speaker recognition methods [2, 3, 4].

In recent years, deep neural network (DNN) has become the state-of-the-art architecture in acoustic modeling for automatic speech recognition (ASR) instead of conventional GMM, with an about 30% relative improvement in word error rate (WER) [5]. In the case of speaker recognition, initial approaches of applying the DNN and restricted Boltzmann machines (RBM) have been reported in [6, 7]. Work in [6] utilized the DNN in order to build an alternative i-vector extractor and work in [7] applied the RBM as a back-end classifier for i-vectors instead of probabilistic linear discriminant analysis (PLDA) model. The recently proposed DNN/i-vector framework in which a DNN trained for ASR was used to extract Baum-Welch statistics achieved competitive or even better performance to modern approaches in text-independent speaker recognition tasks [8, 9].

This research was supported in part by the China National Nature Science Foundation (No.91120303, No.61273267, No.61403370 and No.90820011).

This approach takes advantage of the outstanding ability of DNNs for frame assignment to facilitate speaker information representation. Advanced works based on this approach can be found in [10, 11] with applications of convolutional neural networks (CNNs) and systems developed for microphone speech.

As it is usually difficult to have sufficient in-domain translated ASR training data that matches speaker recognition task in practice, we explore the performance of the aforementioned approach in mismatched conditions, including channel mismatched conditions, spoken language mismatched conditions, etc. We also extend the framework by first applying the Long Short-Term Memory (LSTM) recurrent neural network (RNN) in place of the DNN to achieve advanced improvements in performance. LSTM RNNs have recently been demonstrated to outperform the state-of-the-art DNN systems for acoustic modeling in large vocabulary ASR [12]. Recurrent connections and special network units called memory blocks in the recurrent hidden layer of LSTM RNN make it more powerful to model sequence data than feed forward neural networks [13]. Our motivation is that the effectiveness of LSTM RNNs in making use of the sequence information from longer duration can help frame assignment. Experiments reported in this paper demonstrate their outstanding representation ability in speaker recognition.

The rest of the paper is organized as follows. Section 2 is devoted to present the LSTM RNN architecture. An overview of Baum-Welch statistics extraction from the GMM-UBM and DNN/RNN approaches is provided in section 3. Experiments and discussions are shown in Section 4. Finally, we give the conclusions and future work in Section 5.

2. Long Short-Term Memory RNNs

2.1. Recurrent neural networks

A standard recurrent neural network computes a mapping from an input sequence $\mathbf{x} = (x_1, ..., x_T)$ to an output sequence $\mathbf{h} = (h_1, ..., h_T)$ by calculating the network unit activations using the following equations iteratively from $t = 1$ to T:

$$h_t = \Phi(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \tag{1}$$

$$y_t = W_{hy}h_t + b_y \tag{2}$$

where $\mathbf{h} = (h_1, ..., h_T)$ is the hidden output vector, the W terms are weight matrices (e.g., W_{xh} is the weight matrix connect input and hidden layers), the b terms are bias vectors (e.g., b_h is the bias vector of hidden layer) and Φ denotes the hidden layer activation function.

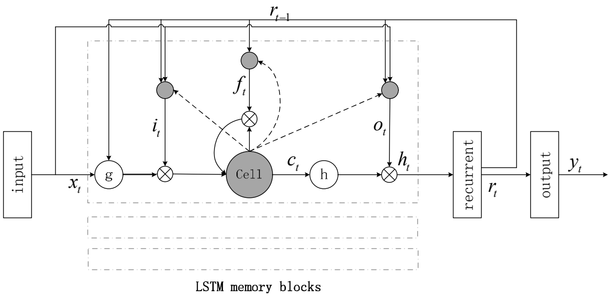

Figure 1: Long Short-Term Memory recurrent neural network architecture. A single memory block is shown for clarity.

2.2. Long Short-Term Memory

In this paper we realize the LSTM RNN described in [12], where a recurrent projection layer is applied to address the computational complexity of learning LSTM models. Each LSTM block contains an input gate, an output gate, a forget gate and a memory cell as shown in Figure 1. It can be implemented by replacing the function Φ in (1) by the following composite functions:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{3}$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{4}$$
$$c_t = f_t c_{t-1} + i_t \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right) \tag{5}$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \tag{6}$$
$$h_t = o_t \tanh(c_t) \tag{7}$$
$$r_t = W_{rh}h_t \tag{8}$$
$$y_t = \phi(W_{yr}r_t + b_y) \tag{9}$$

where σ is the logistic sigmoid function, i_t, f_t, o_t, c_t and r_t are respectively the outputs of the input gate, forget gate, output gate, memory cell and projection layer at time step t, ϕ is the network output activation function, softmax in this paper. The weight matrices from the cell to gate vectors, including W_{ci}, W_{cf}, W_{hc} and W_{co}, are diagonal, in which case the gate vector only receives cell input from its own memory.

3. Baum-Welch statistics extraction

3.1. Roles of the GMM-UBM

In the GMM-UBM framework, each utterance is represented by its zero- and first-order Baum-Welch statistics extracted with the UBM. Typically, given a UBM Ω that composed of C mixture components, the zero- and first-order Baum-Walch statistics of the utterance are obtained by

$$N_c = \sum_t P(c|x_t, \Omega) \tag{10}$$

$$F_c = \sum_t P(c|x_t, \Omega)x_t \tag{11}$$

where $c = 1, \ldots, C$ is the Gaussian component index and $P(c|x_t, \Omega)$ denotes the posterior probability of feature vector x_t generated by the component c.

For the early GMM supervector (GSV) approach [2], a speaker-specific GMM is calculated by adapting the UBM on the speaker's data using maximum a posteriori (MAP) adaptation. The means of the adapted GMM are concatenated to form a supervector that can used in standard classification approaches—inner-product based, SVMs, etc. For the more recent i-vector approach, using a total variability space representation $\boldsymbol{m} = \boldsymbol{m}_0 + T\boldsymbol{w}$ and a Gaussian assumption, factor analysis models the features from the GMM supervector space to the low-dimensional total variability space [4]. The resulting factors are then length-normalized, and used as inputs to PLDA, inner product methods, or other back-end classifiers.

3.2. Replacing GMM-UBM with DNN/RNN

Work reported in [8, 9] shows that how a DNN takes the place of a GMM-UBM in extracting Baum-Welch statistics for text-independent speaker recognition. We extend this work by applying an LSTM RNN described in Section 2 in place of the DNN to the framework. The difference of DNN/RNN approach from GMM-UBM is that the output posteriors are not generated by a GMM but a DNN/RNN trained for ASR. The assumption is that each output unit (tied tri-phone state in general) of DNN/RNN can be accurately modeled by a single Gaussian. For a neural network with parameters Θ, the zero- and first-order Baum-Welch statistics of the given utterance are obtained by

$$N_k = \sum_t p(k|x_t, \Theta) \tag{12}$$

$$F_k = \frac{\sum_t p(k|x_t, \Theta)y_t}{\sum_{k,t} p(k|x_t, \Theta)} \tag{13}$$

where the DNN/RNN is used to compute the posterior $p(k|x_t, \Theta)$ for each class k of each frame, y_t are acoustic features used for speaker recognition, which can be different from the features x_t.

4. Experiments

4.1. Training set

In order to obtain a fair comparison between the baseline system and DNN/RNN systems, both the GMM-UBM and DNN/RNN-HMM ASR models are trained on about 1300 hours of clean English telephone speech from Fisher and Switchboard data sets. The forced alignments for DNN/RNN-HMM training are provided by a GMM-HMM system trained on the same data. A DNN-HMM ASR model trained on about 1000 hours of Mandarin telephone data and GMM-UBM models trained with data matching evaluation data sets are also developed for more comparisons, which will be described in the following subsections. For training the i-vector extractors and LDA projection matrices, we use the telephone speech of NIST 2004, 2005 and 2006 speaker recognition evaluation (SRE) data.

4.2. Configurations

4.2.1. Features

- Features for speaker front-end: A 40-dimensional feature which is formed by 20-dimensional MFCC appended with the first order derivatives computed over a 25ms window every 10ms.
- Features for DNN and RNN models: A 42-dimensional feature which is formed by 13-dimensional perceptual linear prediction (PLP) coefficients and pitch appended with the first and second order derivatives computed over a 25ms window every 10ms.

4.2.2. Models

- GMM-UBM: A gender-independent diagonal UBM with 2048 Gaussian components.

Table 1: *Performance of DNNs, LSTM RNN and GMM-UBMs based on i-vector system on the* 8 *conditions of core test of NIST 2008 (female), in terms of equal error rate and minimum DCF (EER %/minDCF).*

	C1	C2	C3	C4	C5	C6	C7	C8
GMM-UBM(2048)	11.72/0.055	1.50/0.006	12.13/0.058	10.81/0.049	11.90/0.045	7.04/0.033	4.45/0.020	4.68/0.022
GMM-UBM$_{\mathrm{SRE}}$(2048)	11.40/0.054	1.72/0.007	11.82/0.056	10.50/0.051	12.83/0.047	**6.98**/0.033	4.05/0.020	3.95/0.023
GMM-UBM(4096)	12.04/0.056	1.20/0.004	12.56/0.058	9.45/0.049	11.78/0.043	7.04/**0.032**	4.31/0.021	4.22/0.021
DNN-3hid(2992)	8.46/0.040	0.52/**0.002**	8.75/0.042	**8.41**/0.043	10.06/0.039	7.93/0.039	4.05/0.019	4.22/0.022
DNN-3hid$_{\mathrm{Ma}}$(3065)	10.58/0.049	1.20/0.006	10.94/0.051	12.01/0.054	14.55/0.048	8.20/0.042	4.80/0.025	5.54/0.027
DNN-7hid(2992)	8.12/0.036	0.60/0.003	8.42/0.038	9.75/0.043	11.54/0.042	8.09/0.038	3.92/0.020	4.52/0.022
LSTM RNN(2992)	**6.09**/**0.029**	**0.30**/0.003	**6.31**/**0.030**	8.99/**0.042**	**8.77**/**0.037**	7.77/0.037	**3.55**/**0.018**	**3.69**/**0.019**

- DNN: Concatenations of 11 frames (462 dimension in total) are used as input of the network which contains several hidden layers with 2048 units and an output layer with 2992 units for English corpus and 3065 units for Mandarin corpus.
- LSTM RNN: Different to DNNs, the input to the network is just 42 dimensional PLP features calculated at a given time step with no stacking of acoustic frames. The LSTM RNN contains 2 hidden layers with 800 memory cells and 512 recurrent projection units each layer and an output layer with 2992 units for English corpus.
- I-vector model: A gender-dependent i-vector extractor of dimension 600, LDA matrix with speaker factor of dimension 200.

Finally, no score normalization technique is applied to any of the systems. System performance are evaluated in terms of Equal Error Rate (EER) and Minimum Detection Cost Function (MinDCF) of NIST 2008 evaluation plan [14] .

4.3. Experiments on NIST 2008

4.3.1. Evaluation set

As all of the traditional DNNs and LSTM RNNs are trained on telephone speech data, we experiment on all conditions of core test on NIST 2008 SRE list [14], which contains both telephone and interview/microphone speeches spoken with multiple languages, to evaluate the robustness of networks on text-independent speaker verification task. We focus on female data only, where the state-of-the-art performance is worse than that on male data. The classical i-vector/LDA system is applied subsequently, and cosine distance is used for scoring in all experimental systems.

The audio data included in the core test of NIST 2008 are specified to 8 common evaluation conditions by [14]:

C1: interview speech.

C2: interview speech from the same microphone type in training and test.

C3: interview speech from different microphone types in training and test.

C4: interview training speech and telephone test speech.

C5: telephone training speech and non-interview microphone test speech.

C6: telephone speech.

C7: English telephone speech.

C8: English telephone speech spoken by a native U.S. English speaker.

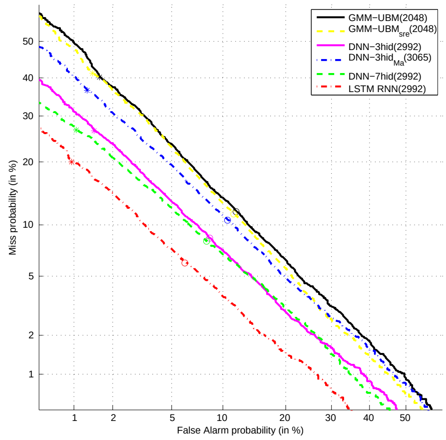

Figure 2: DET curves of DNNs, LSTM RNN and GMM-UBMs based on i-vector system on the interview speech (C1) of core test of NIST 2008, female part.

4.3.2. Results

Experimental results are demonstrated in Table 1. GMM-UBM(2048) denotes the baseline 2048-component GMM system trained with Fisher and Switchboard data. As the DNN/RNN systems effectively use about three thousand classes, another GMM system with 4096 components is trained for comparison. Results show that the performance loss is tiny between GMMs with 2048 and 4096 components. Thus we use 2048-component GMMs only in other experiments for computational reasons. Another GMM system trained on the NIST SRE data (SRE04∼06), named GMM-UBM$_{\mathrm{SRE}}$(2048), is also realized as a system developed in full matched condition for telephone training and evaluating speech. For DNN models, DNN-3hid(2992) denotes the DNN with 3 hidden layers and 2992 output units, and so on.

It can be seen that DNN/RNN systems work better than the baseline systems in most conditions except C6, among them the LSTM RNN achieves best performance. In channel mismatched conditions (i.e., C1∼C3, speech recorded in interview/microphone condition), DNNs and LSTM RNN outperform GMMs significantly, even the model DNN-3hid$_{\mathrm{Ma}}$(3065) which is trained on Mandarin telephone speech data. The detection error tradeoff (DET) curves (derived with the Bosaris toolkit, [15]) of different systems on the the representative test condition C1 are shown in Figure 2 (we do not draw DET curves for all conditions due to space limitations in this paper), from which we clearly observe the superiority of DNN systems, especially the LSTM RNN system. In channel matched conditions (i.e., C6∼C8, speech recorded in telephone condition), the spoken language of data effects system performance. Performance of DNN-3hid$_{\mathrm{Ma}}$(3065)

Table 2: *Performance of DNNs, LSTM RNN and GMM-UBMs based on GSV system on the development set of RSR2015 Part I for different definitions of target and non-target trials, in terms of equal error rate and minimum DCF (EER %/minDCF* $\times 100$*).*

	Male			Female		
	T0-T1	T0-T2	T0-T3	T0-T1	T0-T2	T0-T3
GMM-UBM(2048)-ivec	2.37/1.132	4.23/1.90	0.69/0.315	2.05/0.941	5.44/2.52	0.88/0.451
GMM-UBM(2048)	1.39/0.655	2.93/1.33	0.31/0.110	0.70/0.302	3.30/1.38	0.18/0.067
GMM-UBM$_{\mathrm{RSR}}$(2048)	1.26/0.854	**1.76**/**0.92**	0.10/0.041	0.48/0.212	**2.06**/**0.93**	**0.11**/0.038
DNN-3hid(2992)	0.67/0.369	2.70/1.27	0.15/0.051	0.35/0.138	3.16/1.34	0.15/0.040
DNN-3hid$_{\mathrm{Ma}}$(3065)	0.72/0.350	2.98/1.31	0.15/0.058	0.31/0.136	3.72/1.61	0.15/0.043
DNN-7hid(2992)	0.59/0.322	2.52/1.21	0.11/0.043	0.36/0.134	3.34/1.41	**0.11**/**0.035**
LSTM RNN(2992)	**0.52**/**0.274**	2.71/1.32	**0.098**/**0.031**	**0.28**/**0.129**	3.13/1.30	0.12/0.040

degenerates seriously as the language is mismatched between training and evaluating data. Other DNN/RNN systems trained with English speech are still superior to the baseline GMM-UBM in C7 and C8 (English speech only). While for the multilingual condition C6, DNN/RNN systems yield inferior performance.

The results can be summarized as follows. Firstly, DNN/RNN systems outperform baseline GMM-UBM significantly in channel mismatched conditions. A reasonable explanation is that each of the DNN/RNN output posteriors corresponds to a particular tied tri-phone state and as a result, the DNN/RNN provides a more accurate classification results than that of the unsupervised GMM-UBM in channel mismatched conditions. In other words, the classification ability of data-driven GMM-UBM degenerates more seriously than the DNN/RNN models in channel mismatched conditions. Secondly, in channel matched conditions, the DNN/RNN systems are inferior to GMM-UBM in dealing with unmatched language data. As trained with a particular language, the classification ability of the DNN for speech of other languages is weaker than that of the unsupervised GMM-UBM. Finally, due to the higher frame accuracy rate (an about 30% relative improvement than the DNN) in ASR task, the LSTM RNN outperforms traditional DNNs and the GMM-UBM in most conditions.

4.4. Experiments on RSR2015

4.4.1. Evaluation set

We evaluate the performance of DNN/RNN systems on text-dependent speaker verification task by demonstrating results on RSR2015 [16]. Experiments are developed on the Part I test set of RSR2015 which consists of recordings from 300 speakers in 9 sessions recorded with multiple handphones and tablets, 30 different phrases taken from TIMIT [17]. During the enrollment, one model is trained for each of the 30 sentences of a target speaker. 3 utterances (same content) from the same handset are used to enroll per model, while the other 6 utterances with same content and from different handsets are used for test. During the test, the evaluation task contains 4 types of trials defined as Table 3. We report results on the evaluation set of Part I and train a GMM-UBM system with the background and development sets for comparison purpose.

4.4.2. Results

As our baseline, both the earlier GSV approach and more recent i-vector approach are realized in GMM-UBM systems. Interestingly, system performance is degraded obviously on i-vector approach (also reported in [18]), so we use GSV approach for the following comparisons.

The results are given in Table 2. System performance varies in different combinations of trial types. Although the channel condition of RSR2015 data is very different from the Fisher and SwitchBoard data, the performance of DNN/RNN systems is superior to the baseline GMM-UBM(2048) and GMM-UBM$_{\mathrm{RSR}}$(2048) (trained on the RSR2015 data) in the condition of T0-T1. In this condition, the test utterance of each non-target trial contains the target speaker but wrong lexical content and the problem turns into content verification, thus the DNN/RNN trained for ASR shows the advantage over the GMM-UBM. However, in condition T0-T2, where the problem becomes to verify the speaker with speaking the same content, the performance of the best DNN based systems is slightly better to the GMM-UBM(2048) but inferior to the GMM-UBM$_{\mathrm{RSR}}$(2048). It indicates that channel factor affects the performance significantly under this condition. In the most naive test condition, T0-T3, where both the speaker and content are wrong in non-target test utterances, DNN/RNN systems still outperform the baseline systems. As a comparison of DNN/RNN systems, DNN-3hid$_{\mathrm{Ma}}$(3065) performs slightly inferior to other DNN systems for the reason of unmatched language of training data and LSTM RNN achieves the best performance in most conditions except the female part of condition T0-T3.

Table 3: *The different types of trials defined for text-dependent speaker verification.*

	speaker	lexical content
T0	*target*	*correct*
T1	*target*	*wrong*
T2	*impostor*	*correct*
T3	*impostor*	*wrong*

5. Conclusions and future work

In this paper, we evaluate and analyze the performance of DNN/RNN (from an ASR framework) for extracting Baum-Welch statistics for speaker verification in variant conditions. Experiments on NIST 2008 and RSR2015 show that this approach have a significant superiority compared with conventional GMM-UBM in the data mismatched conditions and still attain comparable performance in matched condition. Especially, we find that the LSTM RNN implemented in this work achieves a further improvement in performance over the traditional DNNs. In the future work we plan to combine the current DNN/RNN framework with other frame-level features in place of the raw acoustic features, such as deep features extracted from neural networks trained with different optimization criterias, and explore the complementary of different systems.

6. References

[1] 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.

[2] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "Svm based speaker verification using a gmm supervector kernel and nap variability compensation," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1. IEEE, 2006, pp. I–I.

[3] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of interspeaker variability in speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 5, pp. 980–988, 2008.

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

[5] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[6] V. Vasilakakis, S. Cumani, and P. Laface, "Speaker recognition by means of deep belief networks," *Proc. Biometric Technologies in Forensic Science*, 2013.

[7] T. Stafylakis, P. Kenny, M. Senoussaoui, and P. Dumouchel, "Preliminary investigation of boltzmann machine classifiers for speaker recognition," in *Proceedings Odyssey Speaker and Language Recognition Workshop*, 2012.

[8] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1695–1699.

[9] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep neural networks for extracting baum-welch statistics for speaker recognition," in *Proc. Odyssey*, 2014.

[10] M. McLaren, Y. Lei, N. Scheffer, and L. Ferrer, "Application of convolutional neural networks to speaker recognition in noisy conditions," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[11] Y. Lei, L. Ferrer, M. McLaren, and N. Scheffer, "A deep neural network speaker verification system targeting microphone speech," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[12] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proceedings of the Annual Conference of International Speech Communication Association (INTERSPEECH)*, 2014.

[13] A. Graves, *Supervised sequence labelling with recurrent neural networks*. Springer, 2012, vol. 385.

[14] N.-S. 2008, "The NIST year 2008 speaker recognition evaluation plan," http://www.itl.nist.gov/iad/mig/tests/sre/2008/index.html, 2008.

[15] N. Brümmer and E. de Villiers, "The bosaris toolkit user guide: Theory, algorithms and code for binary classifier score processing," *Documentation of BOSARIS toolkit*, 2011.

[16] 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.

[17] J. S. Garofolo *et al.*, "Getting started with the darpa timit cd-rom: An acoustic phonetic continuous speech database," *National Institute of Standards and Technology (NIST), Gaithersburgh, MD*, vol. 107, 1988.

[18] T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. Kockmann, and P. Dumouchel, "Text-dependent speaker recognition using plda with uncertainty propagation," *matrix*, vol. 500, p. 1, 2013.