INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Deep Neural Networks for i-Vector Language Identification of Short Utterances in Cars

Omid Ghahabi, Antonio Bonafonte, Javier Hernando, Asunción Moreno

TALP Research Center, Department of Signal Theory and Communications
Universitat Politècnica de Catalunya - BarcelonaTech, Spain
(omid.ghahabi|antonio.bonafonte|javier.hernando|asuncion.moreno)@upc.edu

Abstract

This paper is focused on the application of the Language Identification (LID) technology for intelligent vehicles. We cope with short sentences or words spoken in moving cars in four languages: English, Spanish, German, and Finnish. As the response time of the LID system is crucial for user acceptance in this particular task, speech signals of different durations with total average of 3.8s are analyzed. In this paper, the authors propose the use of Deep Neural Networks (DNN) to model effectively the i-vector space of languages. Both raw i-vectors and session variability compensated i-vectors are evaluated as input vectors to DNNs. The performance of the proposed DNN architecture is compared with both conventional GMM-UBM and i-vector/LDA systems considering the effect of durations of signals. It is shown that the signals with durations between 2 and 3s meet the requirements of this application, i.e., high accuracy and fast decision, in which the proposed DNN architecture outperforms GMM-UBM and i-vector/LDA systems by 37% and 28%, respectively.

Index Terms: Language Identification, Speech Technology in Cars, i-Vector, Deep Neural Network

1. Introduction

Language Identification (LID) is the automatic process of identifying a language spoken in a speech utterance. LID systems use typically one of these two levels of information: acoustic-phonetic or phonotactic [1, 2]. The acoustic-phonetic level statistically represents the characteristic phonemes of each language by a set of acoustic parameters, while the lexical-phonological rules of each language are taken into account in the phonotactic level to connect phonemes and form words.

Recent successful techniques in both acoustic-phonetic and phonotactic levels are typically based on i-vectors [3, 4]. An i-vector is a compact representation of characteristics of a speech signal, which was originally developed for speaker recognition [5] and has also shown promising performance for LID (e.g., [6, 7]). Some post-processing techniques are usually required to compensate undesired session variabilities in the i-vector space. Linear Discriminant Analysis (LDA), Within-Class Covariance Normalization (WCCN), and Probabilistic Linear Discriminant Analysis (PLDA) are the most commonly used techniques in speaker recognition [5, 8, 9]. However, some of these techniques may not be such effective for LID due to the limited number of language classes [7, 10].

On the other hand, the success of Deep Neural Networks (DNN) in speech processing, specifically in speech recognition (e.g., [11, 12]), has motivated the community to make use of DNNs in LID as well (e.g., [13, 14, 3]). A possible way of using DNNs in LID is to combine them with the state-of-the-art i-vector approach. Two kinds of combination have been considered. DNNs have been used in the i-vector extraction process (e.g., [15, 16]) or applied after i-vector computation as classifiers [17, 18, 3]. In [18, 14, 19, 15] DNN bottleneck features are used in the conventional i-vector extraction process, and in [16, 3] in addition to bottleneck features, DNNs are employed for acoustic modeling to extract Baum-Welch statistics. The highest gains are reported when DNN bottleneck features are used with conventional UBM for i-vector extraction [16, 3].

In this paper, the authors have focused on the application of the LID technology in intelligent vehicles. In this scenario, LID systems are evaluated using words or short sentences recorded in cars in four languages. As the use of DNNs in the i-vector extraction process is computationally expensive for both acoustic modeling and bottleneck feature extraction, we will use the conventional i-vectors in this task in which the computational time is important. Instead, we will explore the use of DNNs only for i-vector language classification. Unlike [17, 18, 3] in which neural networks with only one hidden layer are used for this purpose, we will explore DNNs with different architectures. Additionally, both raw i-vectors and channel-compensated i-vectors are considered as inputs to DNNs. In order to have the highest accuracy with the minimum response time of the system, signals with different durations from less than 2s to higher than 3s with the average duration of 3.8s are analyzed. The performances of the proposed DNN architectures are compared with both frame-based GMM-UBM and i-vector baseline systems.

The rest of the paper is organized as follows. Section 2 describes the scenario used for experiments. Section 3 gives a brief overview of the background. Section 4 presents the proposed DNN architecture for language i-vector classification. Section 5 discusses the experimental results, and section 6 concludes the paper.

2. Scenario

This paper is focused on the application of the LID technology in vehicles where the response time of the system is crucial for user acceptance. Four languages were chosen for the experiments: English, Finnish, German, and Spanish. The databases were recorded within the scope of the EU project SpeechDat-Car (LE4-8334) [20]. Each database comprises recordings from 300 speakers recorded in 600 different sessions. Half of the speakers were male and half female. They were equally distributed in three groups of ages between 18 and 65 years old

This work has been supported by the project TEC2015-69266-P (MINECO/FEDER, UE).

http://dx.doi.org/10.21437/Interspeech.2016-1045

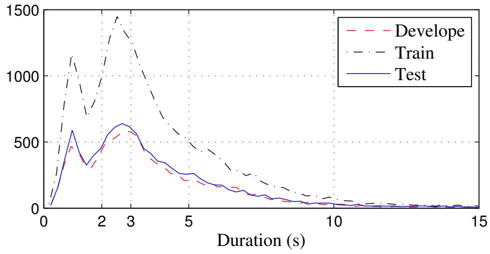

Figure 1: Histogram of signal duration.

and for each language, the speakers were chosen from five dialectal regions.

Four high quality audio channels were recorded simultaneously. For this project, the close-talk microphone is selected. Signals were stored as sequences of 16bit, 16 kHz uncompressed.

Several recording conditions were defined: car stopped by motor running, car in town traffic, car moving at a low speed with rough road conditions, and car moving at a high speed with good road conditions. For the experiments, the signals were taken from all these conditions with windows closed, roof window closed and radio off.

For each language, three sets of speakers were defined by selecting randomly from gender, age group and recording conditions: 150 speakers in the training set, 75 speakers in the development set and 75 speakers in the test set.

Three kinds of data have been selected: spontaneous sentences spoken as answers to specific questions, phonetically rich sentences which are read sentences from a phonetically balanced corpus, and phonetically rich words which are a set of words used to enrich the phonetic balance of the corpus.

Table 1 shows the number of utterances and the total signal duration in hours for each set. The training and development data are around 11 hours for Finnish and German and 6 hours for English and Spanish. Figure 1 shows the histograms of the duration of the utterances. It can be observed that there is a maximum around the first second, corresponding to the phonetically rich words. There is another maximum in the third second, related to phonetically rich sentences. Finally, the tail of the histogram is due to the longer spontaneous utterances.

Table 1: Number of utterances and total duration for train, test and development sets.

Language	Training Utterances	Development Utterances	Test Utterances
German	6,954 (7h18m)	3,491 (3h43m)	3,490 (3h44m)
Spanish	4,755 (3h58m)	2,302 (1h56m)	2,188 (1h46m)
English	4,781 (3h57m)	2,393 (1h58m)	2,324 (1h53m)
Finnish	4,884 (7h18m)	2,489 (3h44m)	2,416 (3h32m)
Total	21,374 (22h32m)	10,675 (11h22m)	10,418 (10h55m)

3. Background

Typical spectral features for LID are Shifted-Delta Cepstral (SDC) features [21] which capture the speech dynamics over a wider range of speech frames than the first- and second-order derivatives of Mel-Frequency Cepstral Coefficients (MFCC). The recent so-called DNN bottleneck features have shown superior performance compared to SDC features (e.g.,[16, 3]), but they are discarded in this task because of the high computational complexity. SDC features can be modeled with different techniques. The i-vector representation is one of the successful techniques which will be described briefly in the following. Nevertheless, since we cope with very short utterances in this task, we will also consider the conventional GMM-UBM as a potential approach which is well-known to perform better than i-vectors dealing with very short signals. At the end, DNNs can be used to model discriminatively the i-vector space of languages which is one of the goals of this paper.

3.1. i-Vector Baseline

An i-vector [5] is a low rank vector representing the characteristics of a speech signal. Feature vectors of a speech signal can be modeled by a set of Gaussian Mixtures (GMM) adapted from a Universal Background Model (UBM). The mean vectors of the adapted GMM are stacked to build the supervector $\mathbf{m}$. The supervector can be further modeled as $\mathbf{m} = \mathbf{m}_u + \mathbf{T}\boldsymbol{\omega}$, where $\mathbf{m}_u$ is the speaker- and session-independent mean supervector from UBM, $\mathbf{T}$ is the total variability matrix, and $\boldsymbol{\omega}$ is a vector of hidden variables. The posterior distribution of $\boldsymbol{\omega}$ is conditioned on the Baum-Welch statistics of the given speech utterance. The mean of the posterior distribution is referred to as i-vector. Recently, it is shown that if the Gaussian UBM is replaced by a DNN, which is typically trained for acoustic modeling in speech recognition, will improve the quality of i-vectors [16, 3].

In the baseline system proposed by the National Institute of Standards and Technology (NIST) [22], all the language i-vectors are centered and whitened. Afterwards, for each language the average of the training i-vectors is considered as the average-language i-vector. Then the cosine distance between the average-language i-vector and test i-vectors is computed as final scores. If a language-labeled background dataset is available, LDA has been shown to be effective for session variability compensation before scoring [6, 23].

3.2. Deep Neural Networks

DNNs are feed-forward neural networks with multiple hidden layers. They are trained using a discriminative backpropagation algorithm given class labels of input vectors. The training algorithm tries to minimize a loss function between the class labels and the outputs. For classification tasks, cross entropy is often used as the loss function and the softmax is commonly used as the activation function at the output layer [24]. Typically, the parameters of DNNs are initialized with small random numbers. However, it has been shown that there are more efficient techniques for parameter initialization, well-known as pre-training, like unsupervised auto-encoders or Restricted Boltzmann Machines (RBM) and supervised layer by layer training [25]. The effectiveness of the pre-training technique depends typically on the amount of available data. It is possible to update the parameters of the network after processing each training example, but it is often more efficient to divide the whole input data (batch) into smaller size batches (minibatch) and to update the parameters by averaging the gradients over each minibatch. The parameter updating procedure is repeated when the whole available input data are processed. Each iteration is called an epoch.

4. DNNs for i-Vector Based LID

The successful use of DNNs for discriminating between target and impostor i-vectors in speaker verification [26, 27], motivated the authors to make use of DNNs for the LID multi classification task. As few i-vectors are available for each target class in speaker recognition and, therefore, the amount of target and impostor i-vectors are highly unbalanced, DNNs need some tricks for training to be efficient. In this application, however, we do not have this problem.

Figure 2 shows the architecture of DNNs we have proposed in this work. The inputs are i-vectors and the outputs are the language class posteriors. The softmax and sigmoid are used as the activation functions of the internal and the output layers, respectively. In order to gaussianize the output posterior distributions, we have proposed to compute the output scores in Log Likelihood Ratio (LLR) forms as,

$$\Lambda(\mathcal{C}_i|\boldsymbol{\omega}) = \log P(\mathcal{C}_i|\boldsymbol{\omega}) - \log \sum_{j \neq i} P\left(\mathcal{C}_j|\boldsymbol{\omega}\right) \qquad (1)$$

where $P(\mathcal{C}_i|\boldsymbol{\omega})$ is the posterior probability of ith language class $\mathcal{C}_i$ given the test i-vector $\boldsymbol{\omega}$. The Gaussian distribution of the output scores is important for being compatible with other LID systems for score fusion.

As the response time of the LID system is important in the car, the computational complexity of the classifier should also be taken into account. Therefore, we have proposed to choose the size of the first hidden layer as the lowest power of 2 greater than the input layer size. From the second hidden layer towards the output, the size of each layer will be half of the previous layer. For example, the configuration of a 3-hidden-layer DNN will be as 400-512-256-128-4, where 400 is the size of the input i-vectors and 4 is the number of language classes. It will be shown in section 5 that, in this way, we can decrease the computational complexity to a great extent while keeping the performance.

Unlike [26, 27] where the pre-training step was helpful, neither RBM nor discriminative pre-training have been effective for this task. This is not only for the the proposed architecture (Fig. 2), but also for other DNNs with hidden layers of the same size in our experiments. Therefore, no pre-training will be employed in this application.

Two forms of i-vectors are considered as inputs to DNNs, raw i-vectors and session-compensated i-vectors. LDA and WCCN are two commonly used techniques for session variability compensation among i-vectors. Although LDA performs better than WCCN for the LID application when cosine scoring is used, we will use only WCCN session-compensated i-vectors as the inputs to DNNs. This is because the number of the language classes is very few in this application and, therefore, the maximum number of meaningful eigenvectors will be also few (number of classes minus one). We implemented different DNN architectures with LDA-projected i-vectors as inputs but no gain was observed. The use of raw i-vectors is advantageous as no language-labeled background data is required.

5. Experimental Results

5.1. Setup and Baseline

Speech signals are pre-emphasized with $\rho = 0.97$. MFCC features are extracted every 10 ms using a 25 ms Hamming window. Each feature vector consists of 8 MFCC coefficients obtained from a Mel filter bank of 24 filters. Before feature extraction, speech signals are subject to an energy-based silence removal process. SDC coefficients are then created by a 8-1-3-5 configuration, spanning a duration of roughly 170 ms.

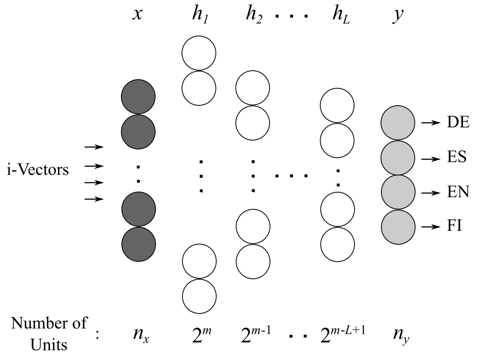

Figure 2: Proposed DNN architecture used for i-vector language identification (L denotes the number of hidden layers and $m = \lceil \log_2^{n_x} \rceil$).

The size of i-vectors are set to 400 and $\mathbf{T}$ matrix is trained with 20 iterations using the same development dataset we have used for training UBM. The gender-independent UBM is represented as a diagonal covariance, 512-component GMM. The i-vector extraction process is carried out using the ALIZE open source software [28]. Since the speech signals are short in this application, both frame-based GMM-UBM and i-vector systems are used as the baselines in this work. The i-vector baseline system is the same as that proposed in [22] and mentioned in section 3.1. Additionally, results with WCCN and LDA session-compensated i-vectors are also reported. As the number of the language classes is 4 in this application, the rank of LDA is set to 3.

For DNN experiments, the proposed architecture in section 4 is implemented with both raw and WCCN session-compensated input i-vectors. In both cases no normalization is applied on i-vectors prior to feeding to DNNs. The Proposed DNN architectures are trained with the learning rates of 0.07 and 0.04 and the number of epochs of 500 and 200 for raw and WCCN compensated i-vectors, respectively. Momentum and weight decay are set, respectively, to 0.9 and 0.001 for all DNNs.

LID systems are evaluated based on the total language identification error rate (LER) defined as,

$$LER = \frac{1}{N} \sum_{i=1}^{N} P_{miss}(L_i) \qquad (2)$$

where $P_{miss}(L_i)$ is the probability that an utterance spoken with the target language L_i is misclassified, and N is the total number of languages.

5.2. Results

Table 2 summarizes the results for all the techniques in four categories based on the test signal durations: less than 2s, between 2 and 3s, more than 3s, and all durations. The first two categories are more interesting because the decision should be made fast in this application. The DNN results are reported based on the proposed architecture of Fig. 2 with 3 hidden layers (400-512-256-128-4). The network is trained with training signals of all durations. As it can be seen in this table, among i-vector baseline systems, i-vector + LDA outperforms the two others

Table 2: Comparison of LID systems for short signals recorded in car. Performance values are reported based on LER (%).

Duration of Test Signals (in s) **Number of Samples**	$t < 2$ 2,472	$2 \leqslant t < 3$ 2,355	$t \geqslant 3$ 5,591	All 10,418
[1] GMM-UBM	**9.98**	4.56	4.70	6.02
[2] i-Vector + Cosine	17.28	6.58	5.00	8.09
[3] i-Vector + WCCN + Cosine	14.50	5.03	3.42	6.31
[4] i-Vector + LDA + Cosine	12.41	3.96	2.32	5.03
[5] i-Vector + WCCN + DNN	12.06	3.30	**2.30**	4.60
[6] i-Vector + DNN	11.01	**2.87**	2.58	**4.54**
Fusion [6] & [4]	11.63	3.41	**1.95**	4.48
Fusion [6] & [1]	10.20	3.04	2.49	4.41
Fusion [6] & [4] & [1]	11.12	3.37	**1.96**	**4.39**

Table 3: Comparison of the proposed DNN architecture with some other architectures.

DNN Architecture	**# Params**	**Duration of Test Signals** $t < 2$	$2 \leqslant t < 3$	$t \geqslant 3$	All
400-512-4	202k	11.74	3.51	2.63	4.79
400-512-512-4	458k	11.47	2.96	**2.39**	4.57
400-512-512-512-4	714k	11.11	3.42	2.62	4.74
400-512-256-4	329k	11.28	3.31	2.51	4.69
400-512-256-128-4	361k	**11.01**	**2.87**	2.58	**4.54**

with a big difference in all categories. Both i-vector-DNN systems show superior performance compared to i-vector + LDA baseline system. However, except for the test signals with longer duration than 3s, DNNs with raw i-vectors perform better than with WCCN session-compensated i-vectors. This is an advantage where no language-labeled background data is available, e.g., [22]. The frame-based GMM-UBM baseline system works better than other systems only for test signals shorter than 2s. However, the accuracy is still high in comparison to other categories.

Furthermore, Table 2 implies that the shortest test signals for which all the techniques achieve adequate performance are between 2 and 3s. For these test signals, the proposed DNN architecture with raw input i-vectors achieves 37% and 27% relative improvements compared to GMM-UBM and i-vector + LDA baseline systems, respectively. In fact, the combination of i-vectors and the proposed DNN architecture meets the goal of this application, that is high accuracy and fast decision. For test signals longer than 3s, i-vector+WCCN+DNN system works the best and for signals with any duration, the i-vector+DNN system outperforms all other individual systems with 25% and 10% relative improvements comparing to GMM-UBM and i-vector+LDA baseline systems, respectively. Finally, the combination of the LID systems in the score level shows that a 16% relative improvement can be achieved for the signals with longer duration than 3s when the i-vector + LDA baseline and the i-vector + DNN systems are fused. Additionally, a slightly improvement is observed for the test signals with all durations when the i-vector + DNN system is combined with both GMM-UBM and i-vector + LDA baseline systems. For score fusion, the scores of different systems are simply summed.

Based on the results reported on Table 2, we can recommend the following LID systems for test signals of different durations: GMM-UBM for shorter than 2s, i-vector + DNN for longer than 2s and shorter than 3s, fusion of i-vector + LDA and i-Vector + DNN for longer than 3s, and fusion of i-vector + LDA, i-vector + DNN, and GMM-UBM for all durations.

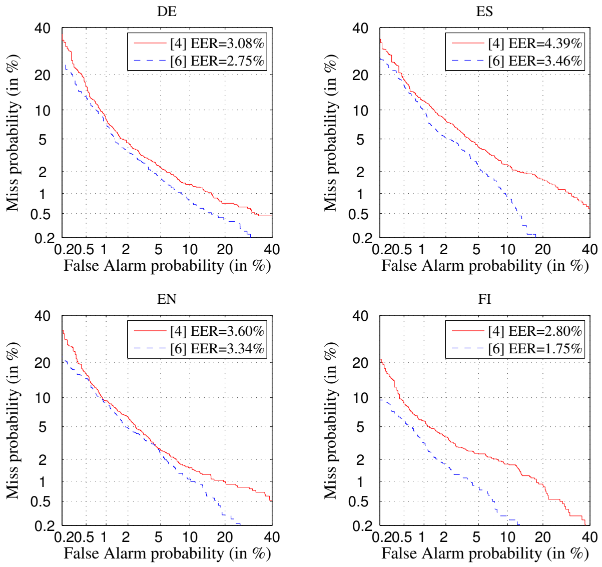

Figure 3: Comparison of the proposed DNN system with the i-vector/LDA baseline system. DET curves are obtained for each language versus all other languages.

Table 3 compares the performance and the size of the proposed DNN architecture with some other DNN architectures. As it can be seen, the proposed architecture with 3 hidden layers achieves the best accuracy in the first two categories. Among these DNN architectures, the 2-hidden-layer DNN with hidden layer size of 512 works slightly better than the proposed architecture for signals longer than 3s, but with the cost of bigger size and, consequently, higher computational complexity.

Figure 3 compares the proposed i-vector + DNN system with the i-vector + LDA baseline system in terms of Detection Error Tradeoff (DET) curves for test signals of all durations. DET curves are obtained for each language versus all other languages. In other words, each language is considered as the target class and all other languages as the non-target one. Each DET curve shows how well the target and non-target languages are distinguished by the LID system. As it can be seen in this figure, the proposed i-vector + DNN system outperforms the i-vector + LDA baseline system for all languages in all operating points, resulting in a 7-38% relative improvements in terms of Equal Error Rate (EER).

6. Conclusions

A Deep Neural Network (DNN) architecture has been proposed in this paper for i-vector language identification (LID) of short utterances recorded in cars. The computational complexity and the response time of the LID system is important in this application. In order to have the highest accuracy with the minimum response time of the system, signals with different durations from less than 2s to higher than 3s with the average duration of 3.8s have been analyzed. It has been shown that for test signals with durations between 2 and 3s the proposed DNN architecture with raw i-vectors as inputs outperforms GMM-UBM and i-vector/LDA baseline systems by 37% and 28%, respectively.

7. References

[1] H. Li, B. Ma, and K. Lee, "Spoken language recognition: from fundamentals to practice," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1136–1159, 2013.

[2] E. Ambikairajah, H. Li, L. Wang, B. Yin, and V. Sethu, "Language identification: A tutorial," *Circuits and Systems Magazine, IEEE*, vol. 11, no. 2, pp. 82–108, 2011.

[3] L. Ferrer, Y. Lei, M. McLaren, and N. Scheffer, "Study of Senone-Based Deep Neural Network Approaches for Spoken Language Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 1, pp. 105–116, Jan. 2016.

[4] A. McCree and D. Garcia-Romero, "DNN senone MAP multinomial i-vectors for phonotactic language recognition," in *INTERSPEECH*, 2015.

[5] N. Dehak, P. 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.

[6] N. Dehak, P. Torres-Carrasquillo, D. Reynolds, and R. Dehak, "Language recognition via i-vectors and dimensionality reduction," in *INTERSPEECH*, 2011, pp. 857–860.

[7] D. González Martínez, O. Plchot, L. Burget, O. Glembek, and P. Matějka, "Language recognition in ivectors space," in *INTERSPEECH*, Firenze Fiera, Florence, Aug. 2011, pp. 861–864.

[8] S. Prince and J. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *IEEE 11th International Conference on Computer Vision, 2007. ICCV 2007*, 2007.

[9] P. Kenny, "Bayesian speaker verification with heavy tailed priors," in *IEEE Odyssey*, 2010.

[10] E. Singer, P. Torres-Carrasquillo, D. Reynolds, A. McCree, F. Richardson, N. Dehak, and D. Sturim, "The MITLL NIST LRE 2011 language recognition system," in *Speaker and Language Recognition Workshop (IEEE Odyssey)*, 2011, pp. 209–215.

[11] G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, Jan. 2012.

[12] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, and T. Sainath, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, 2012.

[13] I. Lopez-Moreno, J. Gonzalez-Dominguez, O. Plchot, D. Martinez, J. Gonzalez-Rodriguez, and P. Moreno, "Automatic language identification using deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 5337–5341.

[14] Y. Song, R. Cui, X. Hong, I. McLoughlin, J. Shi, and L. Dai, "Improved language identification using deep bottleneck network," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015, pp. 4200–4204.

[15] Y. Song, X. Hong, B. Jiang, R. Cui, I. McLoughlin, and L. Dai, "Deep bottleneck network based i-vector representation for language identification," in *INTERSPEECH*, 2015.

[16] F. Richardson, D. Reynolds, and N. Dehak, "A unified deep neural network for speaker and language recognition," in *INTERSPEECH*, 2015.

[17] P. Matjka, O. Plchot, M. Soufar, O. Glembek, L. F. D'haro Enrquez, K. Vesel, F. Grzl, J. Ma, S. Matsoukas, and N. Dehak, "Patrol team language identification system for DARPA RATS P1 evaluation," in *INTERSPEECH*, Portland, Oregon, 2012, pp. 1–4.

[18] P. Matejka, L. Zhang, T. Ng, H. S. Mallidi, O. Glembek, J. Ma, and B. Zhang, "Neural network bottleneck features for language identification," *Proc. of IEEE Odyssey*, pp. 299–304, 2014.

[19] R. Fr, P. Matjka, F. Grzl, O. Plchot, and J. ernock, "Multilingual bottleneck features for language recognition," in *INTERSPEECH*, 2015, pp. 389–393.

[20] A. Moreno, B. Lindberg, C. Draxler, G. Richard, K. Choukri, S. Euler, and J. Allen, "Speechdat-car. a large speech database for automotive environments." in *LREC*, 2000.

[21] P. A. Torres-carrasquillo, E. Singer, M. A. Kohler, and J. R. Deller, "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *Proc. ICSLP 2002*, 2002, pp. 89–92.

[22] "The NIST language recognition i-vector machine learning challenge," 2015, [Online]. Available: http://www.nist.gov/itl/iad/mig/upload/lre-ivectorchallenge-rel-v2.pdf.

[23] E. Singer, P. Torres-Carrasquillo, D. Reynolds, A. McCree, F. Richardson, N. Dehak, and D. Sturim, "The MITLL NIST LRE 2011 language recognition system," in *Proc. Odyssey*, 2012, pp. 209–215.

[24] Z. Ling, S. Kang, H. Zen, A. Senior, M. Schuster, X. Qian, H. Meng, and L. Deng, "Deep Learning for Acoustic Modeling in Parametric Speech Generation: A systematic review of existing techniques and future trends," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 35–52, May 2015.

[25] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning ERRATUM," *APSIPA Transactions on Signal and Information Processing*, vol. 3, 2014.

[26] O. Ghahabi and J. Hernando, "Deep belief networks for i-vector based speaker recognition," in *Proc. ICASSP*, May 2014, pp. 1700–1704.

[27] O. Ghahabi and J. Hernando, "i-vector modeling with deep belief networks for multi-session speaker recognition," in *Proc. Odyssey*, 2014, pp. 305–310.

[28] A. Larcher, J.-F. Bonastre, B. Fauve, K. Lee, C. Lvy, H. Li, J. Mason, and J.-Y. Parfait, "ALIZE 3.0 open source toolkit for state-of-the-art speaker recognition," in *Proc. Interspeech*, 2013, pp. 2768–2771.