Odyssey 2020 The Speaker and Language Recognition Workshop
1-5 November 2020, Tokyo, Japan

Compensation on x-vector for short utterance spoken language identification

Peng Shen, Xugang Lu, Komei Sugiura, Sheng Li, Hisashi Kawai

NICT, Japan

peng.shen@nict.go.jp

Abstract

Feature representation based on x-vector has been successfully applied in spoken language identification tasks. However, the performance on short utterances is severely degraded. The degradation is mainly due to the large variation of the x-vector representation for short utterances which results in large model confusion. One of the solutions is to regularize the representations of short utterances with reference to representations of their corresponding long utterances in x-vector space. Different from previous work, in which both mean and variance statistic components in the x-vector are normalized for speaker recognition task, we argue that variance component in the x-vector encodes discriminative information of languages which should not be normalized for short utterances. Based on this consideration, we proposed an x-vector extraction model for short utterance with adding compensation constraint only for the mean component in the x-vector. Experiments on NIST LRE07 dataset were carried out and showed significant improvement on short utterance LID tasks.

1. Introduction

Spoken language identification (LID) techniques are typically used as a pre-processing stage of multilingual speech recognition and translation systems [1]. For real-time speech processing systems, short utterance LID tasks are important because it can help to reduce the real-time factor and latency of the whole system.

I-vector-based method is one of the effectiveness approaches for LID tasks. The i-vector is a fix-dimensional compact utterance-level representation, therefore, it is easy to be used for building classifier for classification tasks. I-vectors obtained state-of-the-art performance in many LID tasks, especially on relatively longer utterance tasks [2, 3, 4, 5, 6, 7]. However, for short utterance LID tasks, the performance of the i-vector-based approaches often degrades dramatically. One of the main reasons is that the i-vector representation for short utterances has a large distribution variation.

Neural network-based end-to-end approaches, such as, deep neural networks (DNN), recurrent neural networks (RNN), convolutional neural networks (CNN) and attention-based neural networks, have been investigated and demonstrated impressive performance for short utterance LID tasks [8, 9, 10, 11]. Different from conventional i-vector-based approaches, the end-to-end models are easy to be optimized because they do not include many hand-crafted algorithmic components. Recently, an DNN-based speaker embedding approach, i.e., x-vector, was proposed for speaker recognition tasks [12]. In the x-vector approach, by using a frame-level-based DNN network, fix-dimensional embedding representations can be extracted with variable length inputs. The x-vector technique was widely used and obtained state-of-the-art performances for speaker recognition and event detection tasks. The performance of x-vector can be further improved on speaker recognition tasks by using data augmentation techniques [13].

For LID tasks, x-vector method was already investigated [14]. However, as the testing utterances become shorter, the performance also decreases dramatically as that of the i-vector based method. The degradation is mainly because of large pattern confusion caused by the large variation of representations on short utterances in x-vector space. To reduce the variation, one solution is to normalize the representation of short utterances with reference to the representation of their corresponding long utterances in x-vector space. In speaker recognition field, a feature compensation algorithm has been proposed to compensate the data duration mismatch in x-vector space [15]. In the method, both components, i.e., mean and variance statistic components are used. The basic assumption for this compensation algorithm is that x-vectors for long utterances are more stable for speaker characterization which are content independent with more balanced phonemes. However, for language identification on short utterances, both high-level abstract language information and local phonetic information are important cues [1]. We assume that the variance component in x-vector can encode language discriminative information related to local phonetic information. Therefore, different from previous work, both mean and variance statistic components in the x-vector are normalized to reduce the variation of x-vectors [15], we propose an x-vector extraction approach with adding compensation constraint only for the mean component in the x-vector space. In the proposed vector, the mean component is expected to represent high-level abstract language information while retaining variance component to encode frame-based local phonetic information for short utterances.

Using long or full length utterances to improve the performance for short utterances has been already investigated, for example, similar ideas were already used to

10.21437/Odyssey.2020-7

improve i-vector techniques [16, 17], long-short variation transformation was investigated in x-vector space to improve PLDA for speaker recognition task [15], and work on improving CNN-based feature extraction for LID tasks [18]. Different from previous works, we focus on improving x-vector extractor for short utterance LID tasks, and propose a method to add compensation constraint only for mean component in x-vector extraction. We evaluate the proposed method on NIST LRE07 dataset. To the best of our knowledge, the proposed method obtain single system state-of-the-art performance on short utterance task of NIST LRE07.

2. DNN-based embedding: x-vector

DNN-based embedding representation approaches have been successfully applied to many tasks, such as speaker recognition, event/scene detection and language identification [12, 19, 20, 21]. The conventional extracting of the embedding representation, for example x-vector, consists of three modules: a frame-level feature extractor, a statistics pooling layer and utterance-level representation layers. The frame-level feature extractor module outputs frame-level features $\mathbf{h}_t (t = 1, ..., T)$ with inputs of a sequence of acoustic features $\mathbf{x}_t$, where T represents the number of frames. $\mathbf{h}_t$ can be calculated with neural networks, e.g., a time-delay neural network (TDNN) [12] or convolutional neural network [19][20].

Then, a statistics pooling layer converts the frame-level features $\mathbf{h}_t$ into a fixed-dimensional vector by concatenating the mean $\boldsymbol{\mu}$ and standard deviation $\boldsymbol{\sigma}$ of them. The fixed-dimensional vector $\mathbf{V}$ can be described as $[\boldsymbol{\mu}, \boldsymbol{\sigma}]$. The mean and standard deviation are described as

$$\boldsymbol{\mu} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{h}_t, \tag{1}$$

$$\boldsymbol{\sigma} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(\mathbf{h}_t - \boldsymbol{\mu})^2}. \tag{2}$$

Finally, fully-connected hidden layers are used to process the utterance-level representations and a softmax layer is used as the output with each of its output nodes corresponds to one speaker or language ID. In most previous work, the pre-activations of the first fully-connected layer after statistics pooling are extracted as x-vectors. In this work, the feature compensation is applied on $\mathbf{V}$ to improve x-vector extraction.

3. Feature compensation learning for x-vector extractor

3.1. Mean and variance-based compensation learning

The difficulty of LID for short utterances is mainly because short utterances include less phonetic information that makes the representations have large variation. To reduce the representation variation of short utterances, normalization methods using the corresponding long utterances were already investigated for i-vectors and DNN-based approaches [16, 17, 18]. It is natural that we can also apply similar idea on the x-vector extractor.

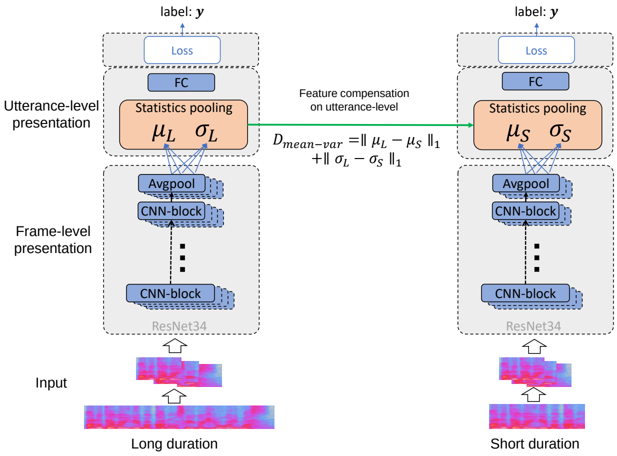

Figure 1: Mean and variance-based compensation learning .

Mathematically, for a short utterance $\mathbf{x}_S$, and its corresponding long utterance $\mathbf{x}_L$, where the short utterance is a part of the long utterance, we define the corresponding x-vectors of short and long utterances as $\mathbf{V}_S$ and $\mathbf{V}_L$. Then, we can use a distance metric to measure the distance of long and short x-vectors as

$$D_{mean-var} = \|\mathbf{V}_L - \mathbf{V}_S\|_1, \tag{3}$$

where $\| \cdot \|_1$ is the L1-norm. $D_{mean-var}$ is used as a regularization term to optimize the network to extract $\mathbf{V}_S$ for short utterances with cross-entropy loss function L_{ce}. The objective function is described as

$$L = (1 - \lambda)L_{ce} + \lambda D_{mean-var}, \tag{4}$$

where λ is a weight coefficient, and $\mathbf{V}_L$ is obtained with a pre-trained long utterance-based x-vector extractor. In this paper, we call this method as mean and variance-based compensation learning that is illustrated in Fig. 1. Similar idea has been successfully used for improving PLDA in x-vector space for speaker recognition task [15].

3.2. Mean-based compensation learning

As x-vector is composed of mean and variance components, Eq.3 is further cast to

$$\begin{aligned} D_{mean-var} =& \|[\boldsymbol{\mu}_L, \boldsymbol{\sigma}_L] - [\boldsymbol{\mu}_S, \boldsymbol{\sigma}_S]\|_1, \\ =& \|\boldsymbol{\mu}_L - \boldsymbol{\mu}_S\|_1 + \|\boldsymbol{\sigma}_L - \boldsymbol{\sigma}_S\|_1. \end{aligned} \tag{5}$$

With $D_{mean-var}$ both mean and variance components are regularized with the corresponding long utterances.

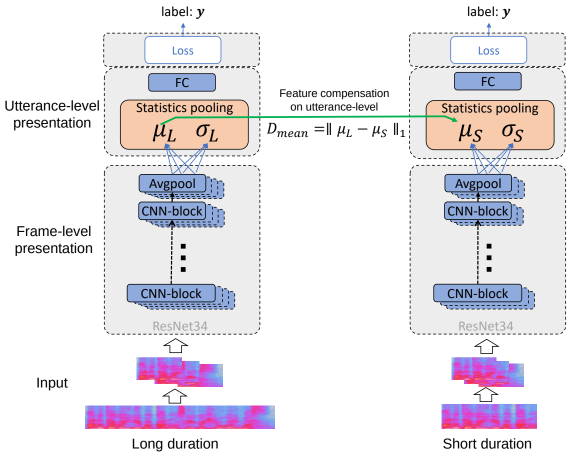

Figure 2: Mean-based compensation learning with mean from long-utterance-based x-vector (Proposed method 1).

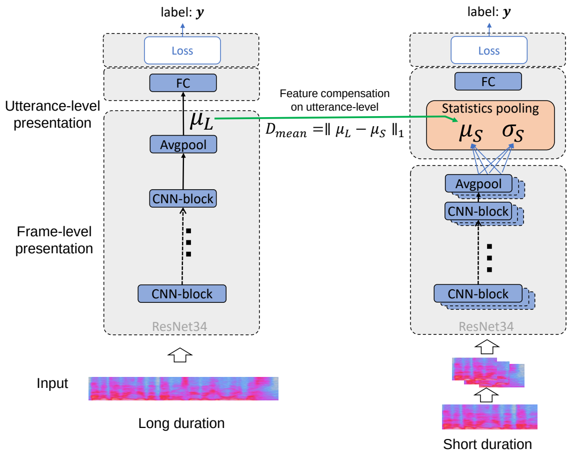

Figure 3: Mean-based compensation learning with mean from a long-utterance-based ResNet network (Proposed method 2).

Table 1: Configuration of ResNet-based x-vector system.

Layer	downsample	channels	output size	blocks
Conv1	False	16	$60 \times L_{in}$	-
res1	True	16	$30 \times L_{in}$	3
res2	True	32	$15 \times L_{in}$	4
res3	True	64	$8 \times L_{in}$	6
res4	True	128	$4 \times L_{in}$	3
avgpool	-	128	128	-
Statistic pooling	-	-	256	-
FC		-	14	-

However, for LID tasks, we argue that regularizing the variance component of x-vector with the representation of long utterances makes x-vectors loss discriminative information of phonetics for short utterances. That will be detrimental to the language identification of short utterances. Based on this consideration, we propose to regularize the mean component of x-vector with representations of long utterances while retaining variance components to encode frame-level phonetic information for providing discriminative features for languages. Then, Eq. 3 is modified to

$$D_{mean} = \|\boldsymbol{\mu}_L - \boldsymbol{\mu}_S\|_1, \tag{6}$$

where $\boldsymbol{\mu}_L$ is obtained from a pre-trained x-vector extractor. Eq.6 can be further cast to

$$D_{mean} = \|\boldsymbol{\mu}_L - \frac{1}{T}\sum_{t-1}^{T}\mathbf{h}_t\|_1. \tag{7}$$

From Eq. 7, we can see that the proposed method reduces the variation of the average of frame-level representations. The standard deviation is calculated with the normalized frame-level representations that is expected to encode the variations related to phonetic information.

4. EXPERIMENTS

Experiments on NIST language recognition evaluation 2007 (LRE07) dataset were conducted to evaluate the effectiveness of the proposed method. The training corpus includes Callfriend datasets, LRE 2003, LRE 2005, development data for LRE07 and LRE 2009 (Only telephone data). The original training data includes 47,843 utterances. Considering the unbalance of the training data for each language. We split the utterances into 30-second based utterances and pick up maximum 10k utterances for each language. 500 utterances for each language were selected as a validation dataset. The total duration of the final training dataset is about 1,100 hours.

The evaluation was done on the closed-set language detection task of NIST LRE07. There are totally 14 target languages in testing dataset, which included 6474 utterances split among three nominal durations: 30, 10 and 3 seconds. We also prepared shorter testing dataset by picking up fixed-length utterances with 1.0, 1.5 and 2.0 seconds from the official 3 seconds testing dataset. Equal error rate (EER) and average cost performance for LRE 2009 challenge (Cavg) were used as the evaluation criteria.

4.1. Implementation of baseline methods

One of the baseline systems is a ResNet-based system similar to [22]. The ResNet systems used a standard 34-layer-ResNet [23] for feature extraction and followed by a global average pooling (GAP) layer and one fully-connected layer with outputs of language IDs. For x-vector systems, the same 34-layer-ResNet was used for the frame-level feature extraction. Statistic pooling of mean and variance components were calculated based on the outputs of the frame-based representations. The network for x-vector is illustrated in table 1. During training, the training samples were prepared with input length between a and b seconds. We compared baseline systems with $[a, b]$ was set to $[1, 10]$ and $[5, 10]$. For x-vector systems, the context width is 50 for each frame. The stochastic gradient descent (SGD) with momentum 0.9 was used for model training. The initial learning rate was set to 0.1 and learning rate was divided 2 every 10 epochs. The training epoch was fixed to 100 and the optimal model was selected using the validation dataset.

In the testing stage, all the duration data was tested on the same trained model. Because the duration length is arbitrary, we feed the testing speech utterance to the trained neural network one by one. Both the training data and testing data were processed with a power energy-based VAD to detect the speech, then randomly length chunks were cut with a shift of 100 frames. 60-dimensional log mel-filterbank features were used. Finally, chunk-based mean and variance normalization was applied on the features.

4.2. Implementation of the proposed method

We focused on building short utterance-based systems, i.e., using utterances 1 to 10 seconds as inputs. The corresponding representation of long utterances was obtained with a pre-trained model with utterances 5 to 10 seconds duration. Similar to the baseline systems, the input length of long and short models are both randomly selected, and we keep the input length of the long always longer than the short. The models were optimized with Eq. 4 and λ was evaluated with values of 0.1, 0.3, 0.5, 0.7 and 0.9. Other settings, such as, optimizer, mini-batch and learning rate, were same to those of the baseline systems.

4.3. Results of the baseline systems

Table 2 shows the results of the baseline systems. For comparison, we also listed latest results reported by other researchers. From the results we can see our baseline systems, i.e., "ResNet-GAP (short)" and "x-vector (short)", obtained comparable results with the state-of-the-art single system results on EER. Compared with short input-based systems, i.e., "ResNet-GAP (short)" and "x-vector (short)", the long input-based systems, i.e., "ResNet-GAP (long)" and "x-vector (long)", performed well on long utterance-based test dataset, for example 10s and 30s. However the performance degraded on short utterances because of the duration mismatch, i.e., models trained with long utterance samples perform well on long utterance test data, bad on short test dataset. Compared with ResNet baseline system, the x-vector systems performed well for long utterance LID tasks, i.e., 10s and 30s.

4.4. Results of the mean and variance-based method

Investigations were done with the mean and variance component-based learning with reference to the representation of the corresponding long utterances. Fig. 1 illustrates the network configuration and the optimization of this investigation. The long utterance-based representations were obtained with the "x-vector (long)" model. The optimization was done with Eq. 4. The investigation results were listed in Table 3. From the results, we can see that compared with the baseline system the mean and variance-based method improved the performance for all the test data.

4.5. Results of the proposed methods

In the proposed method, we regularized mean component of x-vector with the corresponding long utterance representations. We evaluated two models (proposed method 1 and 2) to obtain the long utterance representations, the proposed method 1 used the "x-vector (long)" model and the proposed method 2 used the "ResNet-GAP (long)" model.

The experimental results showed that the proposed method 1, i.e., with long utterance representations of the "x-vector (long)" model, obtained significant improvement on all the durations of the test data. From the results of the mean-variance-based and mean-based methods, we can see that: For long utterance test data, e.g., 10s, and 30s, both the two methods could significantly improve the performance comparing to the baseline x-vector system, i.e., "x-vector (short)", with comparable performances. However, for short utterance test data, e.g., 1.0s, 1.5s, 2.0s and 3s, the proposed mean-base method significantly outperformed the mean-variance-based method on both EER and Cavg.

In proposed method 2, we evaluated the mean-base feature compensation learning with long representations obtained from a long utterance-based ResNet model, i.e., "ResNet-GAP (long)". Experimental results showed that the proposed method 2 also outperformed both the baseline and mean-variance-based method for all the duration test dataset. This demonstrates the possibility that the proposed method can also benefit from advanced utterance-level representation methods.

For a easy comparison, we compared the baseline, mean-variance-based method and the two proposed methods. For each method, we selected the best model by comparing the results of different λ. The comparison is illustrated in Fig. 4. Compared with baseline system, both the mean-variance-based method and the proposed methods obtained better results. For the 30s test data, the mean-variance-based method performed better than the proposed methods. For short utterances, the proposed mean-based methods significantly outperformed the mean-variance-based method. As we have discussed in Section 3.2, variance component of x-vector encodes frame-based discriminative phonetic information that is an important cue for short utterance LID tasks. Directly regularizing the variance component of short utterances with reference to that of long utterances is detrimental to the language identification for short utterances. The proposed methods regularized the mean component of x-vector to capture high-level abstract language information while retaining the variance component to encode

Table 2: Investigation results of baseline systems on NIST LRE07 dataset.

Methods	Training duration	λ	1.0s EER	1.0s Cavg	1.5s EER	1.5s Cavg	2.0s EER	2.0s Cavg	3s EER	3s Cavg	10s EER	10s Cavg	30s EER	30s Cavg
RNN-D&C [24]	-	-	-	-	-	-	-	-	15.57	22.67	6.81	9.45	3.25	3.28
ResNet-GAP [22]	2-8s	-	-	-	-	-	-	-	11.28	9.98	5.76	3.24	3.96	1.73
CNN-BLSTM SAP [22]	2-8s	-	-	-	-	-	-	-	9.50	9.22	3.48	2.54	1.77	0.97
ResNet-GAP (short)	1-10s	-	**20.16**	**28.57**	15.57	**23.00**	12.84	**20.24**	9.18	15.82	5.14	10.66	3.71	8.70
ResNet-GAP (long)	5-10s	-	23.68	31.16	17.70	25.36	14.92	22.38	9.78	**14.74**	3.38	6.22	1.81	3.59
X-vector (short)	1-10s	-	21.50	29.92	**15.43**	23.61	**12.51**	20.83	**8.94**	15.36	4.12	8.34	2.73	5.90
X-vector (long)	5-10s	-	27.15	34.60	20.53	27.55	16.40	23.62	10.89	17.30	**3.29**	**6.18**	**1.25**	**2.64**

Table 3: Experimental results of the mean-variance-based method and the proposed methods on NIST LRE07 dataset.

Methods	Training duration	λ	1.0s EER	1.0s Cavg	1.5s EER	1.5s Cavg	2.0s EER	2.0s Cavg	3s EER	3s Cavg	10s EER	10s Cavg	30s EER	30s Cavg
X-vector (Baseline)	1-10s	-	21.50	29.92	15.43	23.61	12.51	20.83	8.94	15.36	4.12	8.34	2.73	5.90
Mean-var-based learning	1-10s	0.9	20.85	28.96	**14.64**	**22.38**	11.91	**18.41**	**7.92**	**13.38**	**2.87**	**6.05**	**1.62**	**3.82**
Mean-var-based learning	1-10s	0.7	20.67	**28.46**	**14.64**	22.44	**11.77**	19.02	8.02	14.27	3.75	7.51	2.18	4.90
Mean-var-based learning	1-10s	0.5	**20.34**	29.29	15.11	22.52	12.56	20.16	8.62	15.60	3.61	8.64	2.46	5.46
Proposed 1 (mean from x-vector)	1-10s	0.9	18.95	27.13	13.67	21.01	10.94	17.76	**6.91**	13.29	2.97	6.31	1.85	3.99
Proposed 1 (mean from x-vector)	1-10s	0.7	19.32	27.13	13.76	**20.08**	11.08	17.47	7.00	12.63	2.97	6.65	1.99	4.31
Proposed 1 (mean from x-vector)	1-10s	0.5	**18.58**	**26.65**	**13.16**	20.86	**10.57**	**17.33**	7.00	**12.43**	**2.73**	**6.20**	**1.81**	**3.83**
Proposed 2 (mean from ResNet)	1-10s	0.9	**18.67**	**26.26**	**13.48**	**20.30**	10.89	17.02	**6.86**	**12.51**	**3.01**	**6.60**	**2.04**	**4.66**
Proposed 2 (mean from ResNet)	1-10s	0.7	18.72	26.48	13.53	20.41	**10.84**	**16.92**	7.18	12.58	3.29	6.82	2.13	5.05
Proposed 2 (mean from ResNet)	1-10s	0.5	19.05	27.78	14.13	22.11	11.31	18.99	8.16	14.98	4.17	8.67	3.06	6.30

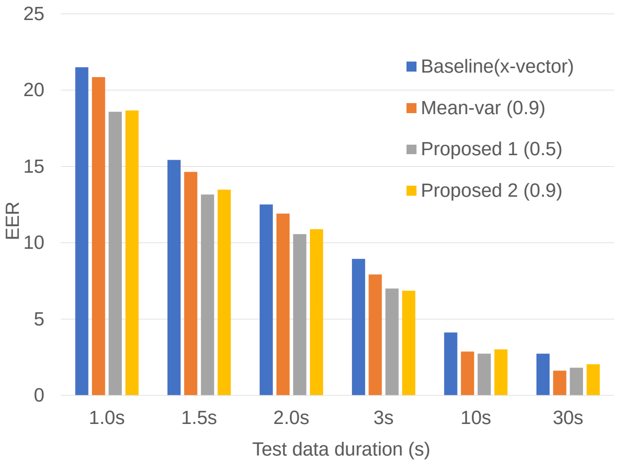

Figure 4: Comparison of baseline, mean-variance-based method and proposed methods.

discriminative phonetic information based on the frame-level representations. The results showed that the proposed methods improved the performance not only on the short but also the long utterance test data.

Our experiment results showed that the proposed method is an effective method for LID tasks, especially for short utterance test data. Compared with the original x-vector method, the proposed method obtained significant improvement for both long and short utterance test data. For the short utterance dataset, i.e., 3s, of NIST LRE07, the proposed method obtained single system state-of-the-art performance.

5. CONCLUSIONS

In this paper, we investigated DNN-based embedding techniques, i.e., x-vector, for LID tasks and proposed a mean component-based feature compensation learning to improve the language identification performance on short utterances. The proposed mean-based feature compensation leaning is expected to capture high-level abstract language information while retaining variance components to encode discriminative phonetic information for shout utterances. Our experimental results showed that the proposed method is an effective method and obtained significant improvement for both long and short utterance LID tasks.

6. ACKNOWLEDGMENTS

The work is partially supported by JSPS KAKENHI No. 19K12035.

7. References

[1] H. Li, B. Ma and K. A. Lee, "Spoken language recognition: From fundamentals to practice," in Proc. of *The IEEE*, vol. 101, no. 5, pp. 1136-1159, 2013.

[2] N. Dehak, P. Torres-Carrasquillo, D. Reynolds and R. Dehak, "Language recognition via ivectors and dimensionality reduction," in Proc. of *Interspeech*, 2011.

[3] S. Novoselov, T. Pekhovsky and K. Simonchik, "STC speaker recognition system for the NIST i-vector challenge," in Proc. of *Odyssey*, 2014.

[4] S. O. Sadjadi, J. W. Pelecanos and S. Ganapathy, "Nearest neighbor discriminant analysis for language recognition," in Proc. of *ICASSP*, 2015.

[5] Y. Song, B. Jiang, Y. Bao, S. Wei and L.-R. Dai, "I-vector representation based on bottleneck features for language identification," in *Electronics Letters*, vol. 49, no. 24, pp. 1569-1570, 2013.

[6] P. Shen, X. Lu, L. Liu and H. Kawai, "Local Fisher discrimiant analysis for spoken language identification," in Proc. of *ICASSP*, 2016.

[7] M. Najafian, S. Safavi, P. Weber and M. Russell, "Augmented Data Training of Joint Acoustic/Phonotactic DNN i-vectors for NIST LRE15,", in Proc. of *Odyssey*, 2016.

[8] I. Lopez-Moreno, J. Gonzalez-Dominguez, D. Martinez, O. Plchot, J. Gonzalez-Rodriguez and P. J. Moreno, "On the use of deep feedforward neural networks for automatic language identification," *Computer Speech & Language*, Vol.40, pp.46-59, 2016.

[9] A. Lozano-Diez, R. Zazo Candil, J. G. Dominguez, D. T. Toledano and J. G. Rodriguez, "An end-to-end approach to language identification in short utterances using convolutional neural networks," in Proc. of *Interspeech*, 2015.

[10] S. Fernando, V. Sethu, E. Ambikairajah and J. Epps, "Bidirectional Modelling for Short Duration Language Identification," in Proc. of *Interspeech*, 2017.

[11] W. Geng, W. Wang, Y. Zhao, X. Cai and B. Xu, "End-to-End Language Identification Using Attention-Based Recurrent Neural Networks," in Proc. of *Interspeech*, 2016.

[12] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in Proc. of *Interspeech*, 2017.

[13] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in Proc. of *ICASSP*, 2018.

[14] D. Snyder, D. Garcia-Romero, A. McCree, G. Sell, D. Povey, and S. Khudanpur, "Spoken language recognition using x-vectors," in Proc. of *Odyssey*, 2018.

[15] A. Kanagasundaram, S. Sridharan, G. Sriram, S. Prachi, and C. Fookes, "A Study of X-vector Based Speaker Recognition on Short Utterances," in Proc. of *Interspeech*, 2019.

[16] A. Kanagasundaram, D. Dean, J. Gonzalez-Dominguezy, S. Sridharan, D. Ramosy, J. Gonzalez-Rodriguezy, "Improving short utterance based i-vector speaker recognition using source and utterance-duration normalization techniques," in Proc. of *Interspeech*, 2013.

[17] W. B. Kheder, D. Matrouf, M. Ajili and J-F Bonastre, "Probabilistic approach using joint long and short session i-vectors modeling to deal with short utterances for speaker recognition," in Proc. of *Interspeech*, 2016.

[18] P. Shen, X. Lu, S. Li and H. Kawai, "Feature Representation of Short Utterances Based on Knowledge Distillation for Spoken Language Identification," in Proc. of *Interspeech*, 2018.

[19] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to end neural speaker embedding system," arXiv preprint arXiv:1705.02304, 2017.

[20] A. Nagrani, J. Chung, and A. Zisserman, "VoxCeleb: A large-scale speaker identification dataset," in Proc. of *Interspeech*, 2017.

[21] H. Zeinali, L. Burget, and J. Cernocky, "Convolutional neural networks and x-vector embedding for DCASE2018 acoustic scene classification challenge," arXivpreprint arXiv:1810.04273, 2018.

[22] W. Cai, D. Cai, S. Huang, M. Li, "Utterance-level End-to-end Language Identification Using Attention-based CNN-BLSTM," in Proc. of *ICASSP*, 2019.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. of *CVPR*, 2016.

[24] G. Gelly, J. L. Gauvain, V. B. Le, and A. Messaoudi, "A Divide-and-Conquer Approach for Language Identification based on Recurrent Neural Networks," in Proc. of *Interspeech*, 2016.