PARALLEL PHONETICALLY AWARE DNNS AND LSTM-RNNS FOR FRAME-BY-FRAME DISCRIMINATIVE MODELING OF SPOKEN LANGUAGE IDENTIFICATION

Ryo Masumura, Taichi Asami, Hirokazu Masataki, Yushi Aono

NTT Media Intelligence Laboratories, NTT Corporation, Japan

{masumura.ryo, asami.taichi, masataki.hirokazu, aono.yushi}@lab.ntt.co.jp

ABSTRACT

Parallel phonetically aware deep neural networks (PPA-DNNs) and long short-term memory recurrent neural networks (PPA-LSTM-RNNs) to enhance frame-by-frame discriminative modeling of spoken language identification are proposed. This idea is inspired by traditional systems based on parallel phoneme recognition followed by language modeling (PPRLM). The proposed methods utilize multiple senone bottleneck features individually extracted from language-dependent senone-based DNNs in a frame-by-frame manner. The multiple senone bottleneck features can yield phonetic awareness to frame-by-frame DNNs and LSTM-RNNs without losing compatibility to real time applications. In experiments, three senone-based DNNs are introduced in order to extract senone bottleneck features, and both single use and parallel use of them are examined. Furthermore, we also examine a combination of PPA-DNNs and PPA-LSTM-RNNs. The proposed method's effectiveness is investigated by comparison with a simple speech aware modeling and traditional systems based on PPRLM.

Index Terms— Spoken language identification, DNNs, LSTM-RNNs, phonetic awareness, senone-DNNs

1. INTRODUCTION

Spoken language identification (LID) is a technology that determines the language label of a speech utterance [1, 2]. LID can be widely used for multilingual speech applications such as speech translation. Nowadays, LID based on deep learning technologies is a subject of great interest [3].

One common approach is to utilize senone-based deep neural networks (DNNs) trained for automatic speech recognition (ASR). The senones are defined as states within context-dependent phones [4]. In fact, senone-based DNNs can capture phonetic awareness. Phonetic awareness is considered to be important for LID, so traditional systems have tried to capture phoneme information using a phoneme recognizer [5]. In recent studies, senone-based DNNs are being used for extracting statistics in i-vectors [6, 7] or extracting features (senone posteriorgrams [8], bottleneck features [9, 10, 11]) for i-vector based systems.

Another approach is to construct discriminative models to directly predict a language label from a speech utterance. In particular, frame-by-frame discriminative modeling based on DNNs [12, 13] or long short-term memory recurrent neural networks (LSTM-RNNs) [14, 15, 16] are attractive because they are compatible with real time applications and work well for short utterances. In fact, they can offer early determination of the language label at any time, unlike the i-vector based systems, because they can perform in a frame-by-frame manner. In addition, utterance-level classification using the frame-by-frame discriminative models can be effectively enhanced by back-end modeling such as sequence summarizing modeling or generative modeling of posteriorgrams [17, 18].

Our aim with this work is to enhance the LID performance of the frame-by-frame discriminative modeling. To this end, we focus on the fact that there is no mechanism for phonetic awareness in conventional frame-by-frame discriminative modeling. As described above, phonetic awareness can be extracted using senone-based DNNs as bottleneck features. Actually, the extraction is also performed in a frame-by-frame manner, so phoneme awareness can be easily transferred to the frame-by-frame discriminative modeling. Furthermore, in consideration of the success of a system based on parallel phoneme recognition followed by language modeling (PPRLM) [5], parallel use of multiple language dependent senone-based DNNs will yield further phonetic awareness.

This paper proposes a modeling that combines frame-by-frame DNNs and LSTM-RNNs with multiple language-dependent senone-based DNNs. Thus, the proposed methods can be regarded as a system that combines the two main approaches mentioned above. In the proposed method, bottleneck features are individually extracted from language-dependent senone-based DNNs in a frame-by-frame manner and then used for frame-by-frame DNNs and LSTM-RNNs. We call the modeling with single senone-based DNNs phonetically aware DNNs (PA-DNNs) and PA-LSTM-RNNs, and the modeling with multiple senone-based DNNs parallel PA-DNNs (PPA-DNNs) and PPA-LSTM-RNNs. An advantage of the proposed methods is that discriminative performance can be improved without losing compatibility to real time applications.

The proposed methods are related to the feature augmentation of DNNs. In ASR fields, speaker aware features or noise aware features are introduced into DNNs or LSTM-RNNs [19, 20, 21]. In addition, a similar idea to our proposed methods was introduced into DNNs for speech activity detection; however, only single senone-based DNN was used for extracting bottleneck features [22]. To the best of our knowledge, this paper is the first work on DNNs and LSTM-RNNs that utilize parallel bottleneck feature extraction based on multiple language dependent senone-based DNNs.

Main contributions are summarized as follows.

- This paper introduces three language-dependent senone-based DNNs in order to extract bottleneck features for constructing PPA-DNNs and PPA-LSTM-RNNs. In addition, we examine a combination of PPA-DNNs and PPA-LSTM-RNNs.
- This paper also presents speech aware DNNs (SA-DNNs) and SA-LSTM-RNNs in which bottleneck features extracted from DNN for speech activity detection (speech/non-speech DNN) are utilized [23]. We investigate relationships between phonetic awareness and speech awareness.
- This paper shows results of a conventional parallel phonet-

ically aware method based on a PPRLM system. In our PPRLM system, RNNLMs trained from senone sequences are used [24].

In Section 2 of this paper, we describe an LID system based on frame-by-frame DNNs and LSTM-RNNs. The proposed methods based on PPA-DNNs and PPA-LSTM-RNNs are detailed in Section 3. Section 4 describes our experiments using the GlobalPhone database [25]. We conclude in Section 5 with a brief summary.

2. LID SYSTEM BASED ON FRAME-BY-FRAME DNNS AND LSTM-RNNS

LID is defined as the problem of determining a language label $\hat{l}$ from an input utterance $\boldsymbol{X} = \boldsymbol{x}_1, \cdots, \boldsymbol{x}_T$, where $\boldsymbol{x}_t$ means an acoustic feature vector of the t-th frame. LID based on frame-by-frame discriminative modeling is performed by simply averaging the frame-level prediction score:

$$\hat{l} = \arg\max_{l \in \mathcal{L}} \frac{1}{T} \sum_{t=1}^{T} \log P(l|\boldsymbol{X}, t, \boldsymbol{\Theta}), \tag{1}$$

where $\mathcal{L}$ represents a set of target languages and $\boldsymbol{\Theta}$ is a model parameter of a discriminative model. $P(l|\boldsymbol{X}, t, \boldsymbol{\Theta})$ represents a posterior probability of label l in the t-th frame. This determination can be conducted in an online manner. Thus, it supports determination even before reaching the end of the utterance.

When using frame-by-frame DNNs, the input layer is composed by stacking a currently-being-processed frame and its left-right contexts. The DNN-based frame-level posterior probability is calculated as

$$P(l|\boldsymbol{X}, t, \boldsymbol{\Theta}) = P(l|\boldsymbol{i}_t, \boldsymbol{\Theta}_{\texttt{DNN}}), \tag{2}$$

$$\boldsymbol{i}_t = [\boldsymbol{x}_{t-M}^\top, \cdots, \boldsymbol{x}_t^\top, \cdots, \boldsymbol{x}_{t+M}^\top]^\top, \tag{3}$$

where M denotes context size in the input layer.

Unidirectional LSTM-RNNs can automatically store previous long-range information in hidden layers without stacking previous frames. The LSTM-RNN-based frame-level discriminative probability is calculated as

$$P(l|\boldsymbol{X}, t, \boldsymbol{\Theta}) = P(l|\boldsymbol{x}_t, \boldsymbol{h}_{t-1}, \boldsymbol{\Theta}_{\texttt{LSTM}}), \tag{4}$$

where $\boldsymbol{h}_{t-1}$ represents outputs of the previous hidden layers. In addition, unidirectional LSTM-RNNs can be fused with DNNs by averaging each log probability.

3. PARALLEL PHONETICALLY AWARE DNNS AND LSTM-RNNS

This paper proposes PPA-DNNs and PPA-LSTM-RNNs based on parallel senone bottleneck feature extraction. Figure 1 shows the detailed procedure of the proposed method. First, a parallel bottleneck feature is extracted from multiple language-dependent senone-based DNNs. Next, the parallel senone bottleneck feature is input into both DNNs and LSTM-RNNs. As shown in Fig. 1, the proposed method also performs in a frame-by-frame manner.

A senone bottleneck feature can be extracted from a senone-based DNN with a bottleneck layer. The bottleneck feature means the output of the bottleneck layer. For the senone-based DNN, an input layer is composed by stacking a currently-being-processed frame and its left-right contexts. A senone bottleneck feature of the t-th frame extracted from a senone-based DNN for language s is defined as

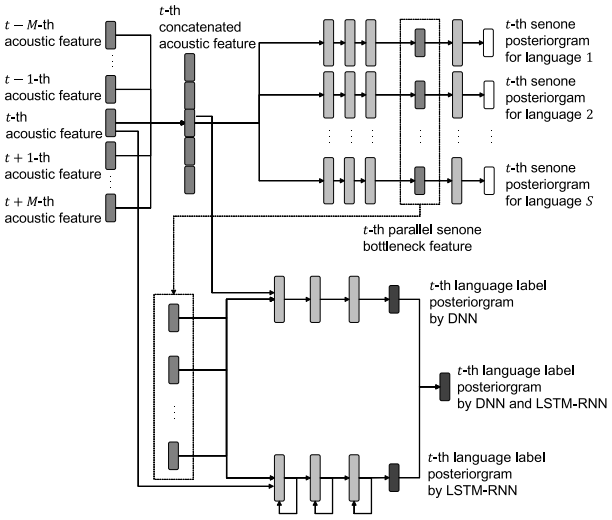

Fig. 1. *PPA-DNNs and PPA-LSTM-RNNs based on parallel senone bottleneck feature extraction.*

$$\boldsymbol{z}_t^{(s)} = f(\boldsymbol{i}_t; \boldsymbol{\Lambda}^{(s)}), \tag{5}$$

where $\boldsymbol{\Lambda}^{(s)}$ is a model parameter of a senone-based DNN for language s. $f()$ means a function of the bottleneck feature extraction.

A parallel senone bottleneck feature is composed by multiple senone bottleneck features individually extracted from language-dependent senone-DNNs. A parallel senone bottleneck feature of the t-th frame is defined as

$$\boldsymbol{b}_t = [\boldsymbol{z}_t^{(1)\top}, \cdots, \boldsymbol{z}_t^{(S)\top}]^\top \tag{6}$$

where S is the number of language-dependent senone-based DNNs. When a single senone-based DNN is used, the parallel bottleneck feature corresponds to one senone bottleneck feature.

PPA-DNNs and PPA-LSTM-RNNs use the parallel senone bottleneck features for feature augmentation. A frame-level posterior probability based on PPA-DNN is calculated as

$$P(l|\boldsymbol{X}, t, \boldsymbol{\Theta}) = P(l|\bar{\boldsymbol{i}}_t, \boldsymbol{\Theta}_{\texttt{PPADNN}}), \tag{7}$$

$$\bar{\boldsymbol{i}}_t = [\boldsymbol{i}_t^\top, \boldsymbol{b}_t^\top]^\top, \tag{8}$$

where $\boldsymbol{\Theta}_{\texttt{PPADNN}}$ is a model parameter of PPA-DNN.

In addition, a frame-level discriminative probability based on PPA-LSTM-RNNs is calculated as

$$P(l|\boldsymbol{X}, t, \boldsymbol{\Theta}) = P(l|\bar{\boldsymbol{x}}_t, \boldsymbol{h}_{t-1}, \boldsymbol{\Theta}_{\texttt{PPALSTM}}), \tag{9}$$

$$\bar{\boldsymbol{x}}_t = [\boldsymbol{x}_t^\top, \boldsymbol{b}_t^\top]^\top, \tag{10}$$

where $\boldsymbol{\Theta}_{\texttt{PPALSTM}}$ is a model parameter of PPA-LSTM-RNN.

PA-DNN and PA-LSTM-RNN correspond to PPA-DNN and PPA-LSTM-RNN with a single senone bottleneck feature. In addition, SA-DNN and SA-LSTM-RNN can be constructed using bottleneck feature extracted from speech/non-speech DNN as well as PA-DNNs and PA-LSTM-RNN.

Table 1. *Data sets for LID evaluation.*

	Train		Valid	Test
French	9,862	(25.3 hours)	308	308
German	9,496	(17.1 hours)	284	303
Korean	7,794	(20.1 hours)	153	160
Mandarin	9,608	(29.3 hours)	203	273
Portuguese	9,568	(24.5 hours)	315	256
Russian	11,549	(24.8 hours)	234	269
Shanghai	2,179	(7.9 hours)	137	227
Spanish	6,500	(20.8 hours)	131	202
Swedish	11,168	(20.4 hours)	154	381
Thai	13,739	(27.3 hours)	100	150
Turkish	6,489	(15.9 hours)	121	281
Vietnamese	18,089	(18.6 hours)	231	371
ALL	116,041	(252.0 hours)	2,371	3,181

4. EXPERIMENTS

4.1. Data sets

For LID evaluation, we used GlobalPhone, a multilingual data corpus [25]. GlobalPhone includes spoken utterances read by native speakers in several languages. The average utterance duration is about 7 seconds. We used 12 languages and split them into a training set (Train), validation set (Valid), and test set (Test). Details on the number of utterances and the data size are given in Table 1.

In addition, we prepared several data sets to train senone-based DNNs and speech/non-speech DNNs. Note that these data sets are not included in the data sets for LID evaluation. Details are given in Table 2, where the number of labels means the number of senones in senone-based DNNs.

4.2. Setups

We used 38 dimensional MFCC coefficients (12MFCC, 12ΔMFCC, 12$\Delta\Delta$MFCC, Δpower and $\Delta\Delta$power) as an acoustic feature that is extracted using 20-msec-long windows shifted by 10 msec.

In addition, we constructed senone-based DNNs and speech/non-speech DNN from individual data sets. Each DNN had five hidden layers. The fourth hidden layer was a bottleneck layer and its unit size was set to 64. Other hidden layers had 1024 units. Output layer size corresponds to the number of labels shown in Table 2. In order to train individual DNNs, we used discriminative pre-training to construct an initial network [26] and then fine-tuned it using mini-batch stochastic gradient descent (MB-SGD). The validation set was used for early stopping.

For LID evaluation, the following systems were constructed.

- *Phoneme recognition-based systems*:
  PRLM was a system based on phoneme recognition followed by language modeling. We used RNNLM in order to model senone sequences decoded by individual senone-based DNN. **PPRLM** was a combination method of three PRLM systems.
- *Frame-by-frame DNN-based systems*:
  All models are DNNs with five hidden layers and 1024 sigmoid units. In **DNN**, the input was 418 dimensional acoustic features formed by stacking the current processed frame and its ± 5 left-right context. In **SA-DNN**, 64 dimensional bottleneck feature extracted from speech/non-speech DNN was added to the input. In **PA-DNN**, 64 dimensional senone bottleneck feature extracted from language-dependent senone-based DNN was added to the input. In **PPA-DNN**, 192 dimensional parallel senone bottleneck feature composed by 3 senone bottleneck features was added to the input. For training, we used discriminative pre-training to construct an initial network and then fine-tuned it using MB-SGD. The validation set was used for early stopping.
- *Frame-by-frame LSTM-RNN-based systems*:
  All models are left-to-right unidirectional LSTM-RNNs with three hidden layers and 512 units. In **LSTM-RNN**, the input was just 38 dimensional acoustic feature of the target frame without stacking other frames. In **SA-LSTM-RNN**, 64 dimensional bottleneck feature extracted from speech/non-speech DNNs was added to the input. In **PA-LSTM-RNN**, 64 dimensional senone bottleneck feature extracted from language-dependent senone-based DNN was added to the input. In **PPA-LSTM-RNN**, 192 dimensional parallel senone bottleneck feature composed by three senone bottleneck features was added to the input. For training, we used discriminative pre-training to construct an initial network and then fine-tuned it using MB-SGD and back propagation through time algorithm. The validation set was used for early stopping.
- *Combination systems*:
  Combination systems of DNN and unidirectional LSTM-RNN were used. Their classification is conducted by averaging log probabilities individually calculated from DNN and LSTM-RNN.

Table 2. *Data sets for speech/non-speech and senone-based DNNs.*

	Size	# of labels
Speech/Non-Speech	85 hours	2
Japanese (Ja)	253 hours	3,072
English (En)	268 hours	2,601
Mandarin (Ma)	374 hours	2,882

4.3. Results

First, we investigated the frame-level LID performance of frame-by-frame DNNs and LSTM-RNNs since the utterance-level performance is affected by the frame-level performance in both the conventional and proposed methods. Table 3 shows the frame-level error rate (FER) for the validation set and test set.

The results show that FER in both frame-by-frame DNNs and LSTM-RNNs was improved by introducing individual senone bottleneck features. On the other hand, bottleneck feature extracted from speech/non-speech DNNs was not so effective compared to senone bottleneck features. This indicates that speech awareness is insufficient to improve LID performance compared to phonetic awareness. The highest performance was attained by using parallel senone bottleneck feature in both DNNs and LSTM-RNNs. This suggests that parallel use of senone bottleneck features can enhance phonetic awareness.

Next, we investigated utterance-level LID performance using an identification task that evaluates hard decision accuracy by selecting the top scored language. Utterance-level error rate (UER) was used as the evaluation metric. In addition, for evaluation of early determination performance, we examined the results achieved after 1 sec

Table 4. *Utterance-level LID performance: UER (%)*

	System	Senone-based DNNs	Valid			Test		
			1 sec	3 sec	Whole	1 sec	3 sec	Whole
(1).	PRLM	Ja	38.01	12.78	8.04	40.24	16.41	9.78
(2).	PRLM	En	39.78	15.53	9.16	42.92	20.02	12.74
(3).	PRLM	Ma	40.79	15.40	7.98	43.58	21.88	12.01
(4).	PPRLM	Ja, En, Ma	24.97	7.85	5.33	28.17	9.84	6.80
(5).	DNN	-	7.77	1.14	0.43	11.29	4.25	3.12
(6).	SA-DNN	Speech/Non-Speech	7.34	0.64	0.34	9.97	4.00	2.30
(7).	PA-DNN	Ja	3.97	0.51	0.34	7.08	1.58	0.73
(8).	PA-DNN	En	6.12	0.98	0.43	7.86	2.39	1.17
(9).	PA-DNN	Ma	5.57	0.47	0.26	8.21	2.11	0.99
(10).	PPA-DNN	Ja, En, Ma	4.35	**0.34**	0.25	6.01	0.85	0.55
(11).	LSTM-RNN	-	14.34	2.45	0.91	16.54	5.22	2.55
(12).	SA-LSTM-RNN	Speech/Non-Speech	13.88	2.54	0.68	16.01	4.53	1.83
(13).	PA-LSTM-RNN	Ja	10.72	1.19	0.34	11.83	2.14	0.73
(14).	PA-LSTM-RNN	En	12.22	2.13	0.76	14.84	3.21	1.07
(15).	PA-LSTM-RNN	Ma	11.77	1.86	0.68	12.61	3.05	0.98
(16).	PPA-LSTM-RNN	Ja, En, Ma	9.33	1.23	0.33	10.56	2.30	0.51
(17).	DNN+LSTM-RNN	-	5.32	0.89	0.38	9.25	3.65	2.17
(18).	PPA-DNN+PPA-LSTM-RNN	Ja, En, Ma	**2.07**	0.38	**0.22**	**4.15**	**0.82**	**0.48**

Table 3. *Frame-level LID performance: FER (%).*

System	Senone-based DNNs	Valid	Test
DNN	-	42.34	46.29
SA-DNN	Speech/Non-Speech	41.45	46.88
PA-DNN	Ja	35.64	40.25
PA-DNN	En	36.76	42.23
PA-DNN	Ma	37.60	42.04
PPA-DNN	Ja, En, Ma	**34,06**	**38.84**
LSTM-RNN	-	16.41	22.36
SA-LSTM-RNN	Speech/Non-Speech	16.58	20.40
PA-LSTM-RNN	Ja	13.32	16.18
PA-LSTM-RNN	En	14.05	17.20
PA-LSTM-RNN	Ma	14.01	16.99
PPA-LSTM-RNN	Ja, En, Ma	**11.76**	**14.56**

and 3 sec. The number of utterances in early determination is the same as in whole-utterance classification.

The results for the phoneme recognition-based systems are shown on lines (1) to (4) of Table 4. The PPRLM system that combined each PRLM system could achieve a performance improvement. This means that parallel phonetic awareness was effectively performed for the conventional systems.

The results for frame-by-frame DNNs are shown on lines (5) to (10) and those for frame-by-frame LSTM-RNNs are shown on lines (11) to (16) in Table 4. Frame-by-frame DNNs and LSTM-RNNs were superior to traditional PRLM systems. LSTM-RNN outperformed DNN in classifying whole utterances while LSTM-RNN was inferior to DNN in the early determination task. PA-DNNs and PA-LSTM-RNNs, which introduced the senone bottleneck feature, outperformed standard DNN and LSTM-RNN, respectively. These results confirm that phonetic awareness performed well for both frame-by-frame DNNs and LSTM-RNNs in the LID task. On the other hand, as well as the frame-level results, SA-DNN and SA-LSTM-RNN were inferior to PA-DNNs and PA-LSTM-RNN. In addition, PPA-DNN and PPA-LSTM-RNN were superior to individual PA-DNNs and PA-LSTM in most conditions. This suggests that parallel phonetic awareness was also effective for state-of-the-art systems using frame-by-frame DNNs or LSTM-RNNs as well as the conventional PPRLM system. This means that each senone bottleneck feature had different properties enhancing LID performance. We conclude that parallel use of multiple senone bottleneck features is an effective solution in order to utilize individual properties.

The results for combination of DNNs and LSTM-RNNs are shown on lines (17) and (18). In most conditions, the best result was attained by PPA-DNN+PPA-LSTM-RNN. This suggests that the combination of DNN and LSTM-RNN was effective when the parallel senone bottleneck feature was used.

5. CONCLUSIONS

This paper presented PPA-DNNs and PPA-LSTM-RNNs to utilize multiple senone bottleneck features individually extracted from language-dependent senone-based DNNs for LID tasks. The proposed method can introduce strong phonetic awareness to DNNs and LSTM-RNNs without losing online processing. Experimental results showed PPA-DNNs and PPA-LSTM-RNNs that utilized three senone bottleneck features outperformed conventional DNN and LSTM-RNN in all conditions. Also, our investigation revealed that multiple senone bottleneck features were more effective than single use of them, and speech awareness was insufficient compared to phonetic awareness. Furthermore, a combination of PPA-DNN and PPA-LSTM-RNN that can also perform in a frame-by-frame manner exhibited the best performance.

In future work, we will confirm the proposed method's effectiveness in a verification task with different data sets. In addition, we will combine the proposed method with back-end modeling to enhancing utterance-level classification performance [18]. We will also compare parallel senone bottleneck features with language-independent bottleneck features that can be extracted from multilingual senone-based DNNs [27, 28, 29].

6. REFERENCES

[1] Eliathamby Ambikairajah, Haizhou Li, Liang Wang, Bo Yin, and Vidhyasaharan Sethu, "Language identification: A tutorial," *Circuits and Systems Magazine, IEEE*, vol. 11, pp. 82–108, 2011.

[2] Haizhou Li, Bin Ma, and Kong Aik Lee, "Spoken language recognition: From fundamentals to practice," *Proceedings of the IEEE*, vol. 101, pp. 1136–1159, 2013.

[3] Fred Richardson, Douglas Reynolds, and Najim Dehak, "Deep neural network approaches to speaker and language recognition," *IEEE Signal Processing Letters*, vol. 22, pp. 1671–1675, 2015.

[4] Luciana Ferrer, Yun Lei, Mitchell McLaren, and Nicolas Scheffer, "Study of senone-based deep neural network approaches for spoken language recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 105–116, 2016.

[5] Marc A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 1, pp. 31–44, 1996.

[6] Yun Lei, Luciana Ferrer, Aaron Lawson, Mitchell McLaren, and Nicolas Scheffer, "Application of convolutional neural networks to language identification in noisy conditions," *In Proc. Odyssey*, pp. 287–292, 2014.

[7] Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," *In Proc. ICASSP*, pp. 1695–1699, 2014.

[8] Luciana Ferrer, Yun Lei, Mitchell McLaren, and Nicolas Scheffer, "Spoken language recognition based on senone posteriors," *In Proc. INTERSPEECH*, pp. 2150–2154, 2014.

[9] Yan Song, Bing Jiang, YeBo Bao, Si Wei, and Li-Rong Dai, "i-vector representation based on bottleneck features for language identification," *Electronics Letters*, vol. 49, no. 24, pp. 1569–1570, 2013.

[10] Bing Jiang, Yan Song, Si Wei, Jun-Hua Liu, Ian Vince McLoughlin, and Li-Rong Dai, "Deep bottleneck features for spoken language identification," *PloS one*, vol. 9, no. 7, pp. 1–11, 2014.

[11] Pavel Matejka, Le Zhang, , Tim Ng, Sri Harish Mallidi, Ondrej Glembek, Jeff Ma, and Bing Zhang, "Neural network bottleneck features for language identification," *In Proc. Oddyssey*, pp. 299–304, 2014.

[12] Ignacio Lopez-Moreno, Javier Gonzalez-Dominguez, and Oldrich Plchot, "Automatic language identification using deep neural networks," *In Proc. ICASSP*, pp. 5337–5341, 2014.

[13] Javier Gonzalez-Dominguez, Ignacio Lopez-Moreno, Pedro J. Moreno, and Joaquin Gonzalez-Rodriguez, "Frame-by-frame language identification in short utterances using deep neural networks," *Neural Networks*, vol. 64, pp. 49–58, 2015.

[14] Javier Gonzalez-Dominguez, Ignacio Lopez-Moreno, Hasim Sak, Joaquin Gonzalez-Rodriguez, and Pedro J. Moreno, "Automatic language identification using long short-term memory recurrent neural networks," *In Proc. INTERSPEECH*, pp. 2155–2159, 2014.

[15] Ruben Zazo, Alicia Lozano-Diez, Javier Gonzalez-Dominguez, Doroteo T. Toledano, and Joaquin Gonzalez-Rodriguez, "Language identification in short utterances using long short-term memory (LSTM) recurrent neural networks," *PloS one*, vol. 11, pp. 1–17, 2016.

[16] Wang Geng, Wenfu Wang, Yuanyuan Zhao, Xinyuan Cai, and Bo Xu, "End-to-end language identification using attention-based recurrent neural networks," *In Proc. INTERSPEECH*, pp. 2944–2948, 2016.

[17] Jan Pesan, Lukas Burget, and Jan Honza Cernocky, "Sequence summarizing neural networks for spoken language recognition," *In Proc. INTERSPEECH*, pp. 3285–3288, 2016.

[18] Ryo Masumura, Taichi Asami, Hirokazu Masataki, Yushi Aono, and Sumitaka Sakauchi, "Language identification based on generative modeling of posteriorgram sequences extracted from frame-by-frame DNNs and LSTM-RNNs," *In Proc. INTERSPEECH*, pp. 3275–3279, 2016.

[19] Mike Seltzer, Dong Yu, and Yongqiang Wang, "An investigation of deep neural networks for noise robust speech recognition," *In Proc. ICASSP*, pp. 7398–7402, 2013.

[20] George Saon, Hagen Soltau, David Nahamoo, and Michael Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," *In Proc. ASRU*, pp. 55–59, 2013.

[21] Tian Tan, Yanmin Qian, Dong Yu, Souvik Kundu, Liang Lu, Khe Chai SIM, Xiong Xiao, and Yu Zhang, "Speaker-aware training of LSTM-RNNs for acoustic modeling," *In Proc. ICASSP*, pp. 5280–5284, 2016.

[22] Luciana Ferrer, Martin Graciarena, and Vikramjit Mitra, "A phonetically aware system for speech activity detection," *In Proc. ICASSP*, pp. 5710–5714, 2016.

[23] Xiao-Lei Zhang and Ji Wu, "Deep belief networks based voice activity detection," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 697–710, 2013.

[24] Tomas Mikolov, Martin Karafiat, Lukas Burget, Jan Cernock y, and Sanjeev Khudanpur, "Recurrent neural network based language model," *In Proc. INTERSPEECH*, pp. 1045–1048, 2010.

[25] Tanja Schultz, Ngoc Thang Vu, and Tim Schlippe, "GlobalPhone: A multilingual text and speech database in 20 languages," *In Proc. ICASSP*, pp. 8126–8130, 2013.

[26] Frank Seide, Gang Li, Xie Chen, , and Dong Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," *In Proc. ASRU*, pp. 24–29, 2011.

[27] Karel Vesely, Martin Karafat, Frantisek Grezl, Milos Janda, and Ekaterina Egorova, "The language-independent bottleneck features," *In Proc. SLT*, pp. 336–341, 2012.

[28] Radek Fer, Pavel Matejka, Frantisek Grezl, Oldrich Plchot, and Jan Honza Cernocky, "Multilingual bottleneck features for language recognition," *In Proc. INTERSPEECH*, pp. 389–393, 2015.

[29] Wang Geng, Jie Li, Shanshan Zhang, Xinyuan Cai, and Bo Xu, "Multilingual tandem bottleneck feature for language identification," *In Proc. INTERSPEECH*, pp. 413–417, 2015.