ADVERSARIAL MULTILINGUAL TRAINING FOR LOW-RESOURCE SPEECH RECOGNITION

Jiangyan Yi[1,2], *Jianhua Tao*[1,2,3], *Zhengqi Wen*[1], *Ye Bai*[1,2]

[1]National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, China
[3]CAS Center for Excellence in Brain Science and Intelligence Technology,
Institute of Automation, Chinese Academy of Sciences, China

ABSTRACT

This paper proposes an adversarial multilingual training to train bottleneck (BN) networks for the target language. A parallel shared-exclusive model is also proposed to train the BN network. Adversarial training is used to ensure that the shared layers can learn language-invariant features. Experiments are conducted on IARPA Babel datasets. The results show that the proposed adversarial multilingual BN model outperforms the baseline BN model by up to 8.9% relative word error rate (WER) reduction. The results also show that the proposed parallel shared-exclusive model achieves up to 1.7% relative WER reduction when compared with the stacked share-exclusive model.

Index Terms— Speech recognition, low-resource, deep neural networks, bottleneck features, adversarial multilingual training

1. INTRODUCTION

Multilingual training is an effective approach to improve the performance of automatic speech recognition (ASR) systems for low-resource languages [1, 2, 3].

Previously, deep neural network (DNN) based acoustic models trained jointly on several languages are used to train bottleneck (BN) networks [4, 5]. Sercu et al. [6] propose to utilize deep convolutional neural networks (CNN) to train multilingual BN networks. More recently, Hartmann et al. [7] use very deep CNNs and bi-directional long-short term memory networks (BLSTM) to train BN feature extractors. The BN extractors have shared and exclusive layers. The shared layers are used to learn language-invariant features. While the exclusive layers are used to capture language-dependent features. The BN features are extracted from the shared layers. Previous studies [8, 9, 10] have shown that acoustic models trained using BN features outperform models trained only on the target language data, especially when the amount of labelled data from the target language is limited. However, the BN shared features may contain some unnecessary language-specific information.

Inspired by the success of adversarial training on domain adaptation [11], this paper proposes an adversarial multilingual training to alleviate this problem. A parallel shared-exclusive model is also proposed to train the BN network using multitask learning [12]. Adversarial training [13] is used to ensure that the shared layers can learn language-invariant features.

Adversarial learning of DNNs is one of the hottest topics in many tasks recently. Ganin et al. [11] proposed to use adversarial strategy for domain adaptation in image tasks. More recently, Chen et al. [14] use adversarial multi-criteria learning for Chinese word segmentation in text tasks. Shinohara [15] and Saon et al. [16] utilizes adversarial multi-task learning for noise robustness and speaker adaptation respectively. These methods use adversarial multi-task learning to improve the performance of the primary task. The results show that they achieve state-of-the-art performance. However, this paper uses the adversarial learning to train multilingual BN networks. The BN networks are used to extract features for the target languages. There has been no work, to the best of our knowledge, that uses adversarial multilingual learning for lower-resource speech recognition.

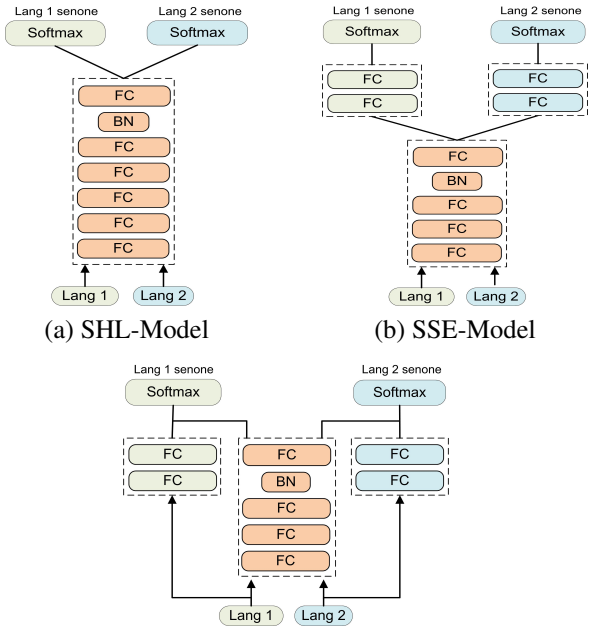
The main contributions of this paper are as follows. 1) A parallel shared-exclusive BN model is proposed to extract features for the target language. 2) An adversarial training is used to force the shared layers to learn language-invariant features. Experiments are conducted on IARPA Babel datasets. The results show that the proposed adversarial multilingual BN model outperforms the baseline BN model by up to 8.9% relative word error rate (WER) reduction. The results also show that the proposed parallel shared-exclusive model achieves up to 1.7% relative WER reduction when compared with the stacked share-exclusive model.

The rest of this paper is organized as follows. Section 2 introduces multilingual bottleneck models. Section 3 describes adversarial training for shared layers. Section 4

presents the experiments. The results are discussed in Section 5. This paper is concluded in Section 6.

2. MULTILINGUAL BOTTLENECK MODELS

Two conventional DNN based BN models are introduced at first. Then the proposed BN model is presented. The three BN models are shown in Fig.1. The BN models are used to extract BN features for the target languages.



(a) SHL-Model (b) SSE-Model

(c) PSE-Model

Fig. 1. Architectures of DNN based BN models. SHL-Model and SSE-Model are the conventional BN models. PSE-Model is the proposed parallel BN model. BN denotes the bottleneck layer used to extract features. FC denotes the full connected layer. The labels of the output layer are language-specific senones.

SHL-Model (shared hidden layers model): The architecture of SHL-Model is widely used for low-resource speech recognition [17, 18]. The shared layers are hidden layers. While the exclusive layers are the output layers.

SSE-Model (stacked shared-exclusive model): There are a few of studies which attempt to extract BN features using SSE-Model [6]. The shared and exclusive layers of this model are stacked. They are both hidden layers. The outputs of the shared layers are the inputs of the exclusive layers.

PSE-Model (parallel shared-exclusive model): None of the existing studies utilize the PSE-Model to train BN feature extractors. The shared and exclusive layers of this model are parallel. The outputs of the shared and exclusive layers are concatenated as the inputs of the output layers.

Given a dataset with N_m training samples $\{x_i^{(m)}, y_i^{(m)}\}_{i=1}^{N_m}$ for the m-th language, where $\{x_i^{(m)}, y_i^{(m)}\}$ is the training samples (frame-level), $x_i^{(m)} \in R^d$ is a feature vector, e.g. filterbank coefficients, d is the dimension of the feature vector, $y_i^{(m)} \in \{1, ..., C_y^{(m)}\}$ is the senone label, $C_y^{(m)}$ is the number of senone labels. The multilingual BN model is trained to minimize the cross-entropy on all the languages. The loss function of multilingual training can be defined as:

$$L_{Mul}(\theta^s, \theta^m) = -\sum_{m=1}^{M}\sum_{i=1}^{N_m} logP(y_i^{(m)}|x_i^{(m)}; \theta^s, \theta^m) \quad (1)$$

where θ^s denotes the parameters of the shared layers, θ^m denotes the parameters of the exclusive layers for the m-th language, M is the number of all the languages.

3. ADVERSARIAL TRAINING FOR SHARED LAYERS

In order to learn language-invariant features, the adversarial training is used to optimize the shared layers of SSE-Model (Adv-SSE-Model) and PSE-Model (Adv-PSE-Model) as shown in Fig.2. Thus the shared layers are prevented from learning the language-specific features.

In adversarial training procedure, a language discriminator is used to recognize the language label using the shared features. An additional language label is given for each training sample $\{x_i^{(m)}, y_i^{(m)}, m\}$, where $m \in \{1, ..., M\}$ denotes the language label for each frame, and M is the number of language labels. The language discriminator loss function $L_{Adv}(\theta^s, \theta^a)$ is defined as:

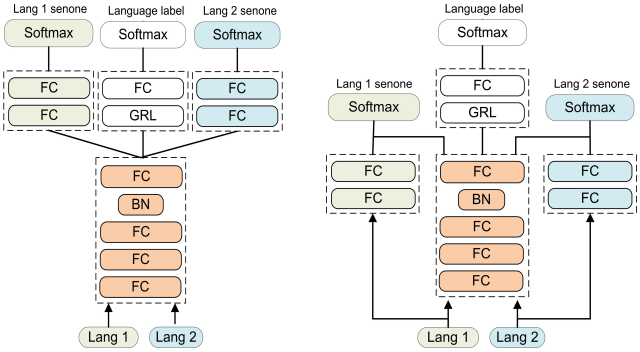
$$L_{Adv}(\theta^s, \theta^a) = -\sum_{m=1}^{M}\sum_{i=1}^{N_m} logP(m|x_i^{(m)}; \theta^s, \theta^a) \quad (2)$$

where θ^a denotes the parameters of the top sub-network of the language discriminator.

The gradient reversal layer (GRL) [11, 19] is introduced to ensure the feature distributions over all the languages are as indistinguishable as possible for the language discriminator. Thus the shared layers can learn language-invariant features. At the feed-forward stage, the GRL acts as an identity transformation. During the back-propagation, however, the GRL takes the gradient from the subsequent level and changes its sign, i.e., multiplying by -1. The GRL has no parameters associated with it.

Thus, the adversarial multilingual training is to optimize the above mentioned two loss functions: $L_{Mul}(\theta^s, \theta^m)$ and $L_{Adv}(\theta^s, \theta^a)$.

The gradient w.r.t. the parameters are computed via back-



(a) Adv-SSE-Model (b) Adv-PSE-Model

Fig. 2. Adversarial training for shared layers. Adv-SSE-Model and Adv-PSE-Model denote SSE-Model and PSE-Model with adversarial training respectively.

propagation, and the parameters are updated as:

$$\theta^m \leftarrow \theta^m - \alpha \frac{\partial L_{Mul}}{\partial \theta_m} \tag{3}$$

$$\theta^a \leftarrow \theta^a - \alpha\lambda \frac{\partial L_{Adv}}{\partial \theta_a} \tag{4}$$

$$\theta^s \leftarrow \theta^s - \alpha(\frac{\partial L_{Mul}}{\partial \theta_s} - \lambda \frac{\partial L_{Adv}}{\partial \theta_s}) \tag{5}$$

where $\alpha \in R$ is the learning rate, $\lambda \in R$ is the loss weight, λ is gradually increased from 0 to 1 as epoch increases so that the model is stably trained [11].

4. EXPERIMENTS

4.1. Datasets

Our experiments are conducted on IARPA Babel datasets. The Babel datasets consist of conversational telephone speech for 25 languages collected across a variety of environments. The total amount of transcribed audio data varies depending on the language and condition. We select 4 languages from the datasets as the source languages: Assamese, Bengali,Kurmanji and Lithuanian. The source languages are the full language pack (FLP), which are only used to train the multilingual BN networks. We also select 3 languages from the datasets as the target languages: Pashto, Turkish, and Vietnamese. The target languages have the FLP and the limited language pack (LLP). All results are reported in terms of word error rate (WER) on 10-hours *dev* sets for the three target languages. Table 1 describes data statistics.

4.2. Experimental setup

Our experiments are conducted using Kaldi speech recognition toolkit [20] and TensorFlow [21]. We follow the officially released Kaldi recipe to build a Gaussian mixture model hidden Markov model (GMM-HMM) at first. The features are extracted with a 25-ms sliding window with a 10-ms shift. Input features for the GMM-HMM model consist of 3-dimensional pitch features and 13-dimensional MFCC and their delta and delta-delta. We use the GMM-HMM models to generate frame-level state alignments for DNN models. All the DNN models use a sliding context window of 11 consecutive speech frames as inputs. Each frame is represented by 3-dimensional pitch features and 40-dimensional log mel-filter bank (Fbank) features plus their delta and delta-delta.

Table 1. Overall experimental data distributions (hours).

	Language (Id)	Dataset	Training	Dev
Source	Assamese (102)	FLP	61	10
	Bengali (103)	FLP	62	10
	Kurmanji (205)	FLP	41	10
	Lithuanian (304)	FLP	42	10
Target	Pashto (104)	FLP	78	10
		LLP	10	10
	Turkish (105)	FLP	77	10
		LLP	10	10
	Vietnamese (107)	FLP	88	10
		LLP	11	10

The four source languages are only used to train BN models. The size of the BN layer is 40, which is set inspired by [7]. The BN features are extracted from the BN models for three target languages respectively. The BN features are concatenated with Fbank and pitch features to train the DNN models for the target languages.

The three target languages are utilized to train DNN based monolingual models. For LLP systems, the DNN models have 5 hidden layers with 2048 nodes in each layers. For FLP systems, the DNN models have 6 hidden layers with 2048 nodes in each layer. The 3-gram language model (LM) is trained using the transcriptions of the training data for each language. We use the officially released vocabulary from IARPA Babel datasets. At the decoding stage, decoding is performed using fully composed 3-gram weighted finite state transducers.

4.3. Baseline model

At first, we train two models only using Fbank with or without pitch features. Then we use four source languages to train two BN models: SHL-Model-5L and SHL-Model-7L. They denote SHL-Model has 5 hidden layers and 7 hidden layers respectively. Each layer has 2048 nodes. The results on the LLP and FLP datasets are listed in Table 2 and Table 3 respectively.

The results show that the models with pitch features outperform the models without pitch features, especially for

Table 2. WERs (%) results on *dev* data for LLP models.

Features	Pashto	Turkish	Vietnamese
Fbank	59.6	58.5	61.9
Fbank+Pitch	59.1	57.9	59.7
SHL-Model-5L	55.6	55.7	59.1
SHL-Model-7L	**55.2**	**55.4**	**58.9**

Table 3. WERs (%) results on *dev* data for FLP models.

Features	Pashto	Turkish	Vietnamese
Fbank	51.1	47.8	53.1
Fbank+Pitch	50.7	47.3	51.4
SHL-Model-5L	48.8	46.5	51.2
SHL-Model-7L	**48.1**	**46.1**	**51.1**

the Vietnamese language. This is because the Vietnamese is a tonal language. The results also show that SHL-Model-7L achieves the best performance. Therefore, SHL-Model-7L is selected as our baseline BN model.

4.4. Adversarial multilingual BN models

In this group of experiments, we use four source languages to train two shared-exclusive multilingual BN models and their adversarial models. The SSE-Model and PSE-Model both have 5 shared hidden layers and 2 exclusive hidden layers. Each hidden layer has 2048 nodes. The network configurations of the Adv-SSE-Model and Adv-PSE-Model are similar to SSE-Model and PSE-Model respectively. The only difference is that the adversarial models add the language discriminator. The results of the models using BN features concatenated with Fbank and pitch features on the LLP and FLP datasets are shown in Table 4 and Table 5 respectively.

The results show that all the models with the adversarial BN features perform better than the models with BN features. Adv-PSE-Model achieves up to 5.5% relative WER reduction when compared with PSE-Model on the LLP. PSE-Model outperforms SSE-Model by up to 1.7% relative WER reduction. Adv-PSE-Model perform better than Adv-SSE-Model.

Table 4. WERs (%) results on *dev* data for LLP models trained using BN features. *Baseline* is SHL-Model-7L.

BN models	Pashto	Turkish	Vietnamese
Baseline	55.2	55.4	58.9
SSE-Model	54.1	54.7	58.1
PSE-Model	53.2	53.6	57.4
Adv-SSE-Model	52.6	52.7	57.1
Adv-PSE-Model	**50.3**	**51.5**	**55.8**

Table 5. WERs (%) results on *dev* data for FLP models trained using BN features. *Baseline* is SHL-Model-7L.

BN models	Pashto	Turkish	Vietnamese
Baseline	48.1	46.1	51.1
SSE-Model	47.7	45.8	50.8
PSE-Model	47.2	45.3	50.2
Adv-SSE-Model	46.8	45.0	50.1
Adv-PSE-Model	**46.1**	**44.4**	**49.5**

5. DISCUSSIONS

The above experimental results show that the proposed adversarial multilingual training is effective. Some interesting observations are made as follows.

The stacked and parallel shared-exclusive models both outperform the shared hidden layers models. The main reason may be that the shared BN features contained mixed language-specific information when the model only has shared hidden layers.

The proposed parallel shared-exclusive models outperform the stacked share-exclusive models for all the target languages. The possible reason is that the shared features contain less language-dependent information when the shared and exclusive layers are parallel.

The proposed adversarial multilingual BN models perform better than multilingual BN models. This is because the adversarial training makes the shared layers to prevent from learning the language-specific features. Thus the shared layers can learn more language-invariant features.

6. CONCLUSIONS

This paper proposes an adversarial multilingual training to train BN feature extractors for the target languages. A parallel shared-exclusive model is also proposed to train the BN network. Adversarial training is used to ensure that the shared layers can extract language-invariant features. Experiments are conducted on IARPA Babel datasets. The results show that the proposed adversarial multilingual BN model outperforms the baseline BN model by up to 8.9% relative WER reduction. The results also show that the proposed parallel shared-exclusive model achieves up to 1.7% relative WER reduction when compared with the stacked share-exclusive model. In future work, we plan to train CNN or BLSTM based adversarial multilingual BN models.

7. ACKNOWLEDGEMENTS

This work is supported by the National High-Tech Research and Development Program of China (863 Program) (No.2015AA016305), the National Natural Science Foundation of China (NSFC) (No.61425017, No.61403386).

8. REFERENCES

[1] J. Cui, B. Kingsbury, B. Ramabhadran, A. Sethy, K. Audhkhasi, X. Cui, E. Kislal, L. Mangu, M. Nussbaum-Thom, and M. Picheny, "Multilingual representations for low resource speech recognition and keyword search," in *Automatic Speech Recognition and Understanding*, 2015, pp. 259–266.

[2] T. Alumae, S. Tsakalidis, and R. Schwartz, "Improved multilingual training of stacked neural network acoustic models for low resource languages," in *INTERSPEECH*, 2016, pp. 3883–3887.

[3] J. Trmal, M. Wiesner, V. Peddinti, X. Zhang, P. Ghahremani, Y. Wang, V. Manohar, H. Xu, D. Povey, and S. Khudanpur, "The kaldi openkws system: Improving low resource keyword search," in *INTERSPEECH*, 2017, pp. 3597–3601.

[4] A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual training of deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7319–7323.

[5] Z. Tuske, D. Nolden, R. Schluter, and H. Ney, "Multilingual mrasta features for low-resource keyword search and speech recognition systems," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 7854–7858.

[6] T. Sercu, C. Puhrsch, B. Kingsbury, and Y. Lecun, "Very deep multilingual convolutional neural networks for lvcsr," 2016.

[7] W. Hartmann, R. Hsiao, and S. Tsakalidis, "Alternative networks for monolingual bottleneck features," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 5290–5294.

[8] H. Xu, H. Su, C. Ni, X. Xiao, H. Huang, E.S. Chng, and H. Li, "Semi-supervised and cross-lingual knowledge transfer learnings for dnn hybrid acoustic models under low-resource conditions," in *INTERSPEECH*, 2016, pp. 1315–1319.

[9] M. Karafit, M.K. Baskar, P. Matjka, K. Vesely, F. Grzl, L. Burget, and J. Aernocky, "2016 but babel system: Multilingual blstm acoustic model with i-vector based adaptation," in *INTERSPEECH*, 2017, pp. 719–723.

[10] T. Sercu, G. Saon, J. Cui, X. Cui, B. Ramabhadran, B. Kingsbury, and A. Sethy, "Network architectures for multilingual speech representation learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 5295–5299.

[11] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Mario Marchand, and Victor Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[12] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997.

[13] I. J Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Advances in Neural Information Processing Systems*, vol. 3, pp. 2672–2680, 2014.

[14] X. Chen, Z. Shi, X. Qiu, and X. Huang, "Adversarial multi-criteria learning for chinese word segmentation," in *ACL*, 2017, pp. 1193–1203.

[15] Y. Shinohara, "Adversarial multi-task learning of deep neural networks for robust speech recognition," in *INTERSPEECH*, 2016, pp. 2369–2372.

[16] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, and et al., "English conversational telephone speech recognition by humans and machines," in *INTERSPEECH*, 2017, pp. 132–136.

[17] K. Vesely, M. Karafiat, F. Grezl, M. Janda, and E. Egorova, "The language-independent bottleneck features," in *Spoken Language Technology*, 2012, pp. 336–341.

[18] G. Heigold, V. Vanhoucke, A. Senior, P. Nguyen, M. Ranzato, M. Devin, and J. Dean, "Multilingual acoustic models using distributed deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8619–8623.

[19] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *International Conference on International Conference on Machine Learning*, 2015, pp. 1180–1189.

[20] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlłaek, Y. Qian, and P. Schwarz, "The kaldi speech recognition toolkit," in *Automatic Speech Recognition and Understanding*, 2011.

[21] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, and M. Isard, "Tensorflow: a system for large-scale machine learning," 2016.