INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Constrained Output Embeddings for End-to-End Code-Switching Speech Recognition with Only Monolingual Data

Yerbolat Khassanov[1], *Haihua Xu*[2], *Van Tung Pham*[1,2], *Zhiping Zeng*[2], *Eng Siong Chng*[1,2], *Chongjia Ni*[3] *and Bin Ma*[3]

[1]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[2]Temasek Laboratories, Nanyang Technological University, Singapore
[3]Machine Intelligence Technology, Alibaba Group

{yerbolat002,haihuaxu,vantung001,zengzp,aseschng}@ntu.edu.sg,
{ni.chongjia,b.ma}@alibaba-inc.com

Abstract

The lack of code-switch training data is one of the major concerns in the development of end-to-end code-switching automatic speech recognition (ASR) models. In this work, we propose a method to train an improved end-to-end code-switching ASR using only monolingual data. Our method encourages the distributions of output token embeddings of monolingual languages to be similar, and hence, promotes the ASR model to easily code-switch between languages. Specifically, we propose to use Jensen-Shannon divergence and cosine distance based constraints. The former will enforce output embeddings of monolingual languages to possess similar distributions, while the later simply brings the centroids of two distributions to be close to each other. Experimental results demonstrate high effectiveness of the proposed method, yielding up to 4.5% absolute mixed error rate improvement on Mandarin-English code-switching ASR task.

Index Terms: code-switching, embeddings, Jensen-Shannon divergence, cosine distance, speech recognition, end-to-end

1. Introduction

The code-switching (CS) is a practice of using more than one language within a single discourse which poses a serious problem to many speech and language processing applications. Recently, the end-to-end code-switching automatic speech recognition (E2E-CS-ASR) gained increasing interest where impressive improvements have been reported [1, 2, 3, 4]. The improvements are mainly achieved for CS languages where sufficient amount of transcribed CS data is available such as Mandarin-English [5]. Unfortunately, for the vast majority of other CS languages the CS data remains too small or even non-existent.

Several attempts have been made to alleviate the CS data scarcity problem. Notably, [6, 7] used semi-supervised approaches to utilize untranscribed CS speech data. On the other hand, [2, 3, 4] employed transfer learning techniques where additional monolingual speech corpora are either used for pre-training or joint-training. On the account of increased training data, these approaches achieved significant improvements. However, all these approaches rely on the cross-lingual signal imposed by some CS data or other linguistic resources such as a word-aligned parallel corpus.

In this work, we aim to build an E2E-CS-ASR using only monolingual data without any form of cross-lingual resource. The only assumption we make is an availability of monolingual speech corpus for each of the CS languages. This setup is important and common to many low-resource CS languages, but has not received much research attention. Besides, it will serve as a strong baseline performance that any system trained on CS data should reach.

However, due to the absence of CS train data, the E2E-CS-ASR model will fail to learn cross-lingual relations between monolingual languages. Consequently, the output token embeddings of monolingual languages will diverge from each other, and hence, prevent the E2E-CS-ASR model from switching between languages. Indeed, we examined the shared output token embedding space learned by E2E-CS-ASR and observed that output token embeddings of two monolingual languages are differently distributed and located apart from each other (see Figure 3a). We hypothesize that the difference between output token embedding distributions restricts the E2E-CS-ASR model from correctly recognizing CS utterances.

To address this problem, we propose to impose additional constraints which will encourage output token embeddings of monolingual languages to be similar. Specifically, we propose to use Jensen-Shannon divergence and cosine distance based constraints. The former will enforce output token embeddings of monolingual languages to possess similar distributions, while the later simply brings the centroids of two distributions to be close to each other. In addition, the imposed constraints will act as a regularization term to prevent overfitting. Our method is inspired by [8, 9] where intermediate feature representations of text and speech are forced to be close to each other. We evaluated our method on Mandarin-English CS language pair from the SEAME [5] corpus where we removed all CS utterances from the training data. Experimental results demonstrate high effectiveness of the proposed method, yielding up to 4.5% absolute mixed error rate improvement.

The rest of the paper is organized as follows. In Section 2, we review related works addressing the CS data scarcity problem. In Section 3, we briefly describe the baseline E2E-CS-ASR model. In Section 4, we present the constrained output embeddings method. Section 5 describes the experiment setup and discusses the obtained results. Lastly, Section 6 concludes the paper.

2. Related works

An early approach to build CS-ASR using only monolingual data is so-called "multi-pass" system [10]. The multi-pass system is based on traditional ASR and consists of three main steps. First, the CS utterances are split into monolingual speech segments using the language boundary detection system. Next, obtained segments are labeled into specific languages using the

http://dx.doi.org/10.21437/Interspeech.2019-1867

language identification system. Lastly, labeled segments are decoded using the corresponding monolingual ASR system. However, this approach is prone to error-propagation between different steps. Moreover, the language boundary detection and language identification tasks are considered difficult.

More recently, the semi-supervised approaches have been explored to circumvent the CS data scarcity problem. For instance, [6] used their best CS-ASR to transcribe a raw CS speech, the transcribed speech is then used to re-train the CS-ASR. In a similar manner, [7] employed their best CS-ASR to re-transcribe the poorly transcribed portion of the training set and then use it to re-train the model. Although the semi-supervised approaches are promising, they still require CS data as well as other systems such as language identification.

In the context of end-to-end ASR models, the transfer learning techniques are widely used to alleviate the CS data scarcity problem. For example, [2, 3] used monolingual data to pre-train the model followed by the fine-tuning with CS data. On the other hand, [4] used both CS and monolingual data for pre-training followed by the standard fine-tuning with the CS only data. While being effective, the transfer learning based techniques highly rely on the CS data.

Generating synthesized CS data using only monolingual data has been also explored in [11, 12, 13, 14], however, they only address the textual data scarcity problem.

3. Baseline E2E-CS-ASR

Figure 1 illustrates the baseline E2E-CS-ASR model based on hybrid CTC/Attention architecture [15] which incorporates the advantages of both Connectionist Temporal Classification (CTC) model [16] and attention-based encoder-decoder model [17]. Specifically, the CTC and attention-based decoder modules share a common encoder network and are jointly trained.

Encoder. The shared encoder network takes a sequence of T-length speech features $\boldsymbol{x} = (x_1, \ldots, x_T)$ and transforms them into L-length high level representations $\boldsymbol{h} = (h_1, \ldots, h_L)$ where $L < T$. The encoder is modeled as a deep convolutional neural network (CNN) based on the VGG network [18] followed by several bidirectional long short-term memory (BLSTM) layers.

$$\boldsymbol{h} = \text{BLSTM}(\text{CNN}(\boldsymbol{x})) \tag{1}$$

CTC module. The CTC sits on top of the encoder and computes the posterior distribution $P_{\text{CTC}}(\boldsymbol{y}|\boldsymbol{x})$ of N-length output token sequence $\boldsymbol{y} = (y_1, \ldots, y_N)$. The CTC loss is defined as a negative log-likelihood of the ground truth sequences $\boldsymbol{y}^*$:

$$\mathcal{L}_{\text{CTC}} = -\log P_{\text{CTC}}(\boldsymbol{y}^*|\boldsymbol{x}) \tag{2}$$

Attention-based decoder module. The attention-based decoder computes the probability distribution $P_{\text{ATT}}(\boldsymbol{y}|\boldsymbol{x})$ over the output token sequence $\boldsymbol{y}$ given the previously emitted tokens $y_{<n}$ and input feature sequence $\boldsymbol{x}$ using the chain rule:

$$\alpha_n = \text{Attention}(s_{n-1}, \alpha_{n-1}, \boldsymbol{h}) \tag{3}$$

$$c_n = \sum_{j=1}^{L} \alpha_{n,j} h_j \tag{4}$$

$$s_n = \text{LSTM}(s_{n-1}, c_n, \text{InputProj}(y_{n-1})) \tag{5}$$

$$P(y_n|y_{<n}, \boldsymbol{x}) = \text{Softmax}(\text{OutputProj}(s_n)) \tag{6}$$

$$P_{\text{ATT}}(\boldsymbol{y}|\boldsymbol{x}) = \prod_n P(y_n|y_{<n}, \boldsymbol{x}) \tag{7}$$

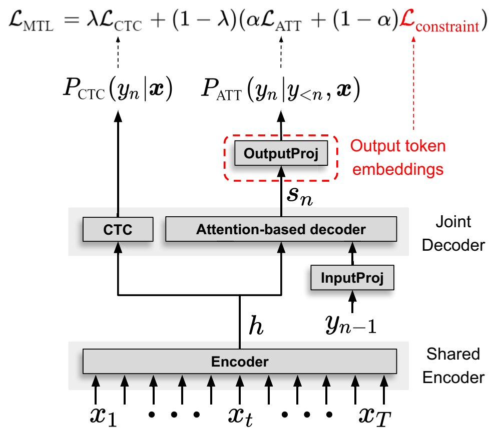


Figure 1: *Hybrid CTC/Attention end-to-end ASR architecture with constrained output token embeddings. The output token embeddings are learned by the parametric matrix of linear output projection layer (OutputProj).*

where α_n is an attention weight vector produced by Attention($\cdot$) module, c_n is a context vector which encapsulates the information in the input speech features required to generate the next token, s_n is a hidden state produced by unidirectional long short-term memory (LSTM). InputProj($\cdot$) and OutputProj($\cdot$) are input and output linear projection layers with learnable matrix parameters, respectively. The input and output learnable matrices hold input and output embedding representations of tokens, respectively. The loss function of attention-based decoder module is computed using Eq. (7) as:

$$\mathcal{L}_{\text{ATT}} = -\log P_{\text{ATT}}(\boldsymbol{y}^*|\boldsymbol{x}) \tag{8}$$

Finally, the CTC and attention-based decoder modules are jointly trained within multi-task learning (MTL) framework as follows:

$$\mathcal{L}_{\text{MTL}} = \lambda\mathcal{L}_{\text{CTC}} + (1-\lambda)\mathcal{L}_{\text{ATT}} \tag{9}$$

where λ controls the contribution of the losses.

Our proposed method will append additional constraint into the MTL framework which will mainly impact the learnable matrix parameter of OutputProj($\cdot$) layer in Eq. (6) as will be explained in the following section.

4. Constrained output embeddings

In this work, we aim to build E2E-CS-ASR using only monolingual data. This setup is essential for the vast majority of CS languages for which CS data is non-existent. However, an E2E-CS-ASR model trained on monolingual data will fail to learn language switch-points, and hence, will perform sub-optimally on input CS speech. We investigated the E2E-CS-ASR model and found that the output token representations of monolingual languages, modeled by linear projection layer OutputProj($\cdot$), to be different and apart from each other (see Figure 3a). We hypothesize that the difference between output token distributions of monolingual languages restricts the E2E-CS-ASR model from switching between languages.

To reduce the discrepancy between these distributions, we propose to constrain output token embeddings using Jensen-Shannon divergence (JSD) and cosine distance (CD). These constraints will typically act as a cross-lingual signal source which will force output token embedding representations of

Table 1: *SEAME dataset statistics after removing the CS utterances from the train set. 'Man' and 'Eng' refer to Mandarin and English languages, respectively*

	train		test_{man}	test_{eng}
	Man	Eng		
# tokens	∼216k	∼109k	∼96k	∼54k
# utterances	21,476	17,925	6,531	5,321
(# CS utterances)	(0)	(0)	(4,418)	(2,652)
Duration	15.8 hr	11.8 hr	7.5 hr	3.9 hr

monolingual languages to be similar. Specifically, JSD will enforce the output token embeddings of monolingual languages to possess similar distributions. On the other hand, CD will enforce the centroids of two distributions to be close to each other.

Jensen-Shannon divergence. First, we assume that learned output token embeddings of monolingual language pair L_1 and L_2 follow a z-dimensional multivariate Gaussian distribution:

$$L_1 \sim Normal(\mu_1, \Sigma_1) \tag{10}$$

$$L_2 \sim Normal(\mu_2, \Sigma_2) \tag{11}$$

The JSD between these distributions is then computed as:

$$\begin{aligned}\mathcal{L}_{\text{JSD}} =& tr(\Sigma_1^{-1}\Sigma_2 + \Sigma_1\Sigma_2^{-1}) \\ &+ (\mu_1 - \mu_2)^T(\Sigma_1^{-1} + \Sigma_2^{-1})(\mu_1 - \mu_2) - 2z \end{aligned}\tag{12}$$

Lastly, we fuse the JSD constraint with the loss function of E2E-CS-ASR using Eq. (9) as follows:

$$\mathcal{L}_{\text{MTL}} = \lambda\mathcal{L}_{\text{CTC}} + (1-\lambda)(\alpha\mathcal{L}_{\text{ATT}} + (1-\alpha)\mathcal{L}_{\text{JSD}}) \tag{13}$$

where $\alpha \in [0, 1]$ controls the importance of the constraint.

Cosine distance. We first compute the centroid vectors C_1 and C_2 obtained by taking the mean of all output token embeddings of monolingual language pair L_1 and L_2, respectively. The cosine distance between two centroids is then computed as follows:

$$\mathcal{L}_{\text{CD}} = 1 - \frac{C_1 \cdot C_2}{\|C_1\|\,\|C_2\|} \tag{14}$$

The CD constraint is integrated into the loss function in a similar way as Eq. (13).

5. Experiment

5.1. Dataset

We evaluate our method on Mandarin-English CS language pair from the SEAME [5] corpus (see Table 1). We used standard data splitting[1] on par with previous works [1, 7] which consists of 3 sets: train, test_{man} and test_{eng}. To match the no CS data scenario, where we assume that we only possess monolingual data, we removed all CS utterances from the train set. The test_{man} and test_{eng} sets were used for evaluation. Both evaluation sets are gender balanced and consist of 10 speakers, but the matrix[2] language of speakers is different, i.e. Mandarin for test_{man} and English for test_{eng}.

[1] https://github.com/zengzp0912/SEAME-dev-set

[2] The dominant language into which elements from the embedded language are inserted.

5.2. E2E-CS-ASR model configuration

We used ESPnet toolkit [19] to train our baseline E2E-CS-ASR model. The encoder module consists of VGG network followed by 6 BLSTM layers each with 512 units. The attention-based decoder module consists of a single LSTM layer with 512 units and employs multi-headed hybrid attention mechanism [20] with 4 heads. The CTC module consists of a single linear layer with 512 units and its weight in Eq. (9) is set to 0.2. The network was optimized using Adadelta with gradient clipping. During the decoding stage, the beam size was set to 30. The baseline model achieves 34.3% and 46.3% mixed error rates (MER)[3] on test_{man} and test_{sge} respectively, when trained on entire SEAME train set including the CS utterances.

5.3. Results and analysis

The experiment results are shown in Table 2. We split the test sets into monolingual and CS utterances to analyze the impact of the proposed method on each of them. We first report the MER performance of a conventional ASR model built using Kaldi toolkit [21] (row 1), the model specifications can be found in [7]. The MER performance of the baseline E2E-CS-ASR model is shown in the second row. We followed the recent trends [1, 2, 4] to obtain a much stronger baseline model. Specifically, we applied speed perturbation (SP) based data augmentation technique [22] and used byte pair encoding (BPE) based subword units [23] to balance Mandarin and English tokens (rows 3 and 4). We tried different vocabulary sizes for BPE and found $4k$ units to work best in our case.

Table 2: *The MER (%) performance of different ASR models built using monolingual data. The test sets are further split into monolingual (mono) and code-switching (CS) utterances*

No.	Model	test_{man}			test_{eng}		
		mono utts.	CS utts.	all	mono utts.	CS utts.	all
1	Kaldi	-	-	39.1	-	-	45.2
2	Baseline	57.7	73.3	70.6	73.7	80.6	78.3
3	+ SP	39.4	56.0	53.2	54.2	65.9	62.2
4	+ BPE	38.1	51.8	49.5	52.9	61.4	58.9
5	+ CD	34.4	49.0	46.3	**47.2**	58.5	55.1
6	+ JSD	34.9	48.8	46.3	47.8	57.6	54.6
7	+ CD	**34.0**	**48.1**	**45.6**	**47.2**	**57.4**	**54.4**

The performance of models employing proposed CD and JSD constraints are shown in rows 5 and 6, the interpolation weights for CD and JSD are set to 0.9 and 0.97, respectively. Both constraints gain considerable MER improvements. Notably, we found that CD constraint is more effective on monolingual utterances, whereas JSD constraint is more effective on CS utterances. To complement the advantages of both constraints, we combined them as follows:

$$\begin{aligned}\mathcal{L}_{\text{MTL}} =& \lambda\mathcal{L}_{\text{CTC}} + (1-\lambda)(\alpha\mathcal{L}_{\text{ATT}} + \\ &(1-\alpha)(\beta\mathcal{L}_{\text{JSD}} + (1-\beta)\mathcal{L}_{\text{CD}}))\end{aligned}\tag{15}$$

where α and β are set to 0.05 and 0.9, respectively. The combination of two constraints significantly improves the MER over the strong baseline model by 3.9% and 4.5% on test_{man} and test_{eng}, respectively (row 7). These results suggest that the proposed method of constraining the output token embeddings is effective.

[3] The term "mixed" refers to different token units used for English (words) and Mandarin (characters).

5.3.1. *Changing the interpolation weight*

We repeat the experiment with different interpolation weights for CD and JSD constraints (hyperparameter α in Eq. (13)) to investigate its effect on MER performance. Figure 2 shows that the proposed method consistently improves the MER over the strong baseline model with SP and BPE. The best results are achieved for interpolation weights in range 0.8-0.99.

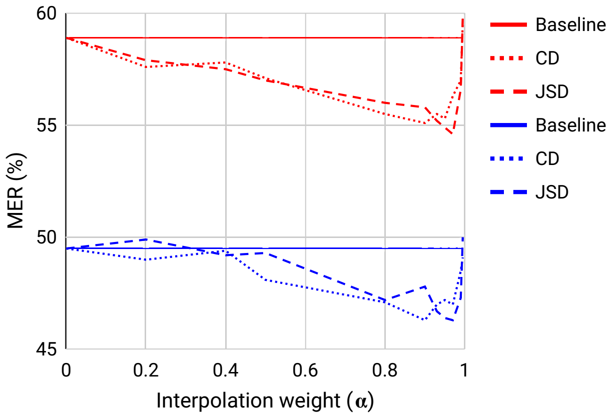
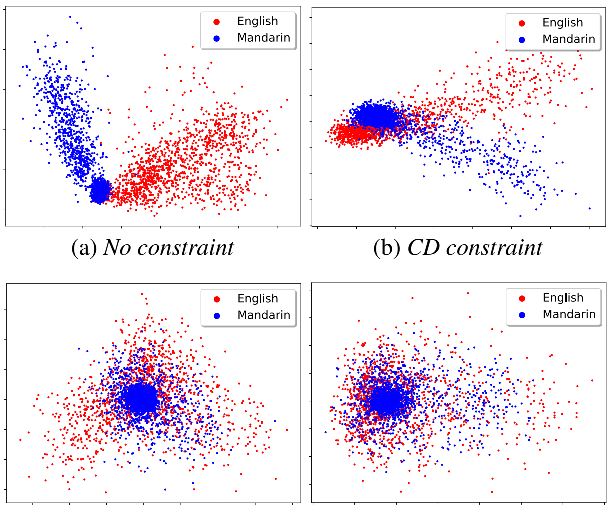


Figure 2: *The impact of CD and JSD constraint interpolation weights on MER performance for test$_{eng}$ (red/top) and test$_{man}$ (blue/bottom) sets.*



(a) *No constraint* (b) *CD constraint*

(c) *JSD constraint* (d) *CD & JSD constraints*

Figure 3: *PCA visualization of shared output token embedding space without (a) and with (b,c,d) proposed constraints.*

5.3.2. *Visualization of shared output token embedding space*

To gain insights from the effects of the proposed method on the shared output embedding space, we visualize it using dimensionality reduction technique based on the principal component analysis (PCA). Figure 3 shows the shared output embedding space without (3a) and with (3b, 3c, 3d) proposed constraints. Note that the learned output token embeddings of monolingual languages strongly diverge from each other when proposed constraints are not employed. Visualization of the shared output embedding space confirms that our method is effective at binding the output token embeddings of monolingual languages.

5.3.3. *Applying language model*

The state-of-the-art results in ASR are usually obtained by employing a language model (LM). To examine whether proposed constraints are complementary with LM, we employed LM during the decoding stage. In this experiment, we tried different LM interpolation weights changed with a step size of 0.025 and report the best results (see Table 3). The LM was trained on the entire SEAME train set, including CS utterances, as a single layer LSTM with 512 units and was integrated using shallow fusion technique [24]. Obtained MER improvements show that proposed constraints and LM complement each other. Moreover, the proposed method benefits from the LM more than the strong baseline model.

Table 3: *The MER performance after applying the language model during the decoding stages*

Model	Decoder LM	MER (%)	
		test$_{man}$	test$_{eng}$
Baseline	No	49.5	58.9
	Yes	49.0	58.6
Baseline + CD & JSD	No	45.6	54.4
	Yes	**45.0**	**53.7**

6. Conclusions

In this work, we proposed a method to train improved E2E-CS-ASR model using only monolingual data. Specifically, our method constrains the output token embeddings of monolingual languages to force them to be similar, and hence, enable E2E-CS-ASR to easily switch between languages. We examined Jensen-Shannon divergence and cosine distance based constraints which are incorporated into the objective function of the E2E-CS-ASR. We evaluated the proposed method on Mandarin-English CS language pair from the SEAME corpus where CS utterances were removed from the train set. The proposed method outperforms the strong baseline model by a large margin, i.e. absolute 3.9% and 4.5% MER improvements on test$_{man}$ and test$_{eng}$, respectively. The visualization of the shared output embedding space confirms the effectiveness of the proposed method. In addition, our method is complementary with the language model where further MER improvement is achieved. Importantly, all these improvements are achieved without using any additional linguistic resources such as word-aligned parallel corpus or language identification system. We believe that the proposed method can be easily adapted to other scenarios and benefit other CS language pairs.

For the future work, we plan to test the proposed method on scenarios with a larger amount of monolingual data and examine its effectiveness on E2E-CS-ASR models trained using CS data. We also plan to study the effects of the proposed method in transfer learning approach where it will be used to pre-train the model with external monolingual data.

7. Acknowledgements

This work is supported by the project of Alibaba-NTU Singapore Joint Research Institute.

8. References

[1] Z. Zeng, Y. Khassanov, V. T. Pham, H. Xu, E. S. Chng, and H. Li, "On the end-to-end solution to mandarin-english code-switching speech recognition," in *20th Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2019, in-press.

[2] N. Luo, D. Jiang, S. Zhao, C. Gong, W. Zou, and X. Li, "Towards end-to-end code-switching speech recognition," *arXiv preprint arXiv:1810.13091*, 2018.

[3] K. Li, J. Li, G. Ye, R. Zhao, and Y. Gong, "Towards code-switching asr for end-to-end ctc models," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, 2019, pp. 6076–6080.

[4] C. Shan, C. Weng, G. Wang, D. Su, M. Luo, D. Yu, and L. Xie, "Investigating end-to-end speech recognition for mandarin-english code-switching," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, 2019, pp. 6056–6060.

[5] D. Lyu, T. P. Tan, E. Chng, and H. Li, "SEAME: a mandarin-english code-switching speech corpus in south-east asia," in *11th Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2010, pp. 1986–1989.

[6] E. Yilmaz, M. McLaren, H. van den Heuvel, and D. A. van Leeuwen, "Semi-supervised acoustic model training for speech with code-switching," *Speech Communication*, vol. 105, pp. 12–22, 2018.

[7] P. Guo, H. Xu, L. Xie, and E. S. Chng, "Study of semi-supervised approaches to improving english-mandarin code-switching speech recognition," in *19th Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2018, pp. 1928–1932.

[8] S. Karita, S. Watanabe, T. Iwata, A. Ogawa, and M. Delcroix, "Semi-supervised end-to-end speech recognition," in *19th Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2018, pp. 2–6.

[9] J. Drexler and J. Glass, "Combining end-to-end and adversarial training for low-resource speech recognition," in *IEEE Spoken Language Technology Workshop, SLT*, 2018, pp. 361–368.

[10] D. Lyu, R. Lyu, Y. Chiang, and C. Hsu, "Speech recognition on code-switching among the chinese dialects," in *IEEE International Conference on Acoustics Speech and Signal Processing, ICASSP*, 2006, pp. 1105–1108.

[11] A. Pratapa, G. Bhat, M. Choudhury, S. Sitaram, S. Dandapat, and K. Bali, "Language modeling for code-mixing: The role of linguistic theory based synthetic data," in *56th Annual Meeting of the Association for Computational Linguistics, ACL, Volume 1: Long Papers*, 2018, pp. 1543–1553.

[12] C. Chang, S. Chuang, and H. Lee, "Code-switching sentence generation by generative adversarial networks and its application to data augmentation," *CoRR*, vol. abs/1811.02356, 2018.

[13] G. I. Winata, A. Madotto, C. Wu, and P. Fung, "Learn to code-switch: Data augmentation using copy mechanism on language modeling," *CoRR*, vol. abs/1810.10254, 2018.

[14] E. Yilmaz, H. van den Heuvel, and D. A. van Leeuwen, "Acoustic and textual data augmentation for improved ASR of code-switching speech," in *19th Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2018, pp. 1933–1937.

[15] T. Hori, S. Watanabe, Y. Zhang, and W. Chan, "Advances in joint CTC-Attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM," in *18th Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2017, pp. 949–953.

[16] A. Graves, S. Fernández, F. J. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *23rd International Conference on Machine Learning, ICML*, 2006, pp. 369–376.

[17] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, 2016, pp. 4945–4949.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[19] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "Espnet: End-to-end speech processing toolkit," in *19th Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2018, pp. 2207–2211.

[20] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems, NIPS*, 2015, pp. 577–585.

[21] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," IEEE Signal Processing Society, Tech. Rep., 2011.

[22] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *16th Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2015, pp. 3586–3589.

[23] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *54th Annual Meeting of the Association for Computational Linguistics, ACL, Volume 1: Long Papers*, 2016.

[24] S. Toshniwal, A. Kannan, C. Chiu, Y. Wu, T. N. Sainath, and K. Livescu, "A comparison of techniques for language model integration in encoder-decoder speech recognition," in *IEEE Spoken Language Technology Workshop, SLT*, 2018, pp. 369–375.