



Language-specific Characteristic Assistance for Code-switching Speech Recognition

Tongtong Song¹, Qiang Xu¹, Meng Ge^{1,2}, Longbiao Wang^{1*}, Hao Shi^{3*},
Yongjie Lv¹, Yuqin Lin¹, Jianwu Dang^{1,4}

¹Tianjin Key Laboratory of Cognitive Computing and Application,

College of Intelligence and Computing, Tianjin University, Tianjin, China

²Department of Electrical and Computer Engineering, National University of Singapore, Singapore

³Graduate School of Informatics, Kyoto University, Sakyo-ku, Kyoto, Japan

⁴Japan Advanced Institute of Science and Technology, Ishikawa, Japan

{songtongtong, longbiao-wang}@tju.edu.cn, shi@sap.ist.i.kyoto-u.ac.jp

Abstract

Dual-encoder structure successfully utilizes two language-specific encoders (LSEs) for code-switching speech recognition. Because LSEs are initialized by two pre-trained language-specific models (LSMs), the dual-encoder structure can exploit sufficient monolingual data and capture the individual language attributes. However, existing methods have no language constraints on LSEs and underutilize language-specific knowledge of LSMs. In this paper, we propose a language-specific characteristic assistance (LSCA) method to mitigate the above problems. Specifically, during training, we introduce two language-specific losses as language constraints and generate corresponding language-specific targets for them. During decoding, we take the decoding abilities of LSMs into account by combining the output probabilities of two LSMs and the mixture model to obtain the final predictions. Experiments show that either the training or decoding method of LSCA can improve the model's performance. Furthermore, the best result can obtain up to 15.4% relative error reduction on the code-switching test set by combining the training and decoding methods of LSCA. Moreover, the system can process code-switching speech recognition tasks well without extra shared parameters or even retraining based on two pre-trained LSMs by using our method.

Index Terms: language-specific characteristic assistance, dual-encoder, code-switching, speech recognition

1. Introduction

Language mixing, especially bilingual mixing, has become a common phenomenon in international communication. Bilingual mixing, also called code-switching (CS), includes inter-sentential code-switching and intra-sentential code-switching. With the development of automatic speech recognition (ASR) [1, 2, 3, 4, 5, 6], CS ASR is urgently needed and attracts more attention [7, 8, 9, 10, 11, 12].

Compared to monolingual ASR with plenty of monolingual data, CS ASR is limited by hard-to-collect speech and transcriptions, especially in the era of deep learning. Therefore, reducing the demand for CS data and making full use of rich resources monolingual data have become research hotspots [13, 14, 15, 16, 17, 18]. Dual-encoder structure is an effective way to make full use of two monolingual data [15, 16, 17, 18]. Under this structure, two language-specific encoders (LSEs) are initiated by corresponding pre-trained language-specific models

(LSMs). Then the language-specific features can be extracted by LSEs and fused by the following shared layers. The high-level representations have strong language discrimination, and it is beneficial for bilingual situation. Therefore, the mixture model achieves comparable performance in two monolingual test sets compared to the well-trained LSMs [15, 16]. Furthermore, the mixture model can recognize CS sentences without language identity information and CS training data [16].

However, existing methods based on dual-encoder structure have some problems in the training and decoding stages. In the training stage, only the loss of the mixture model, which is called mixture loss, is used to optimize the entire model. On the one hand, there are no language constraints on the corresponding LSEs. Then the mixture loss will bias the language-specific features to another language and weaken the strong language discrimination of language-specific features during training. It may decrease the performance of the model in bilingual scenario. On the other hand, projection layers of two pre-trained LSMs are discarded in existing methods, which does not fully use the language-specific knowledge of two pre-trained LSMs. In the decoding stage, we find that LSMs tend to recognize speech in unseen language as in-vocabulary tokens with low confidence or as out-of-vocabulary (OOV) token "unk", but have higher confidence when recognizing the speech in the corresponding language. This will be discussed in Section 4.5. Therefore, the outputs of LSMs have rich language-specific knowledge. However, only the outputs of the mixture model are used for decoding in existing methods.

In this work, we propose a language-specific characteristic assistance (LSCA) method to mitigate the above problems in the training and decoding stages. During training, we introduce language-specific losses as language constraints for LSEs and interpolate them with the mixture loss. Moreover, Since the language-specific losses will face some sentences with the tokens in another language, we generate the language-specific targets for them. During decoding, we take the decoding abilities of two LSMs into account and combine the output probabilities of two LSMs and the mixture model to obtain the final predictions.

Moreover, we explore the impacts of different weights of the language-specific losses in the whole loss and the output probabilities of two LSMs in the final output probabilities for model performance during training and decoding, respectively. In addition, we also explore completing CS ASR tasks by adding no extra shared layers or even without retraining based

*Corresponding author.

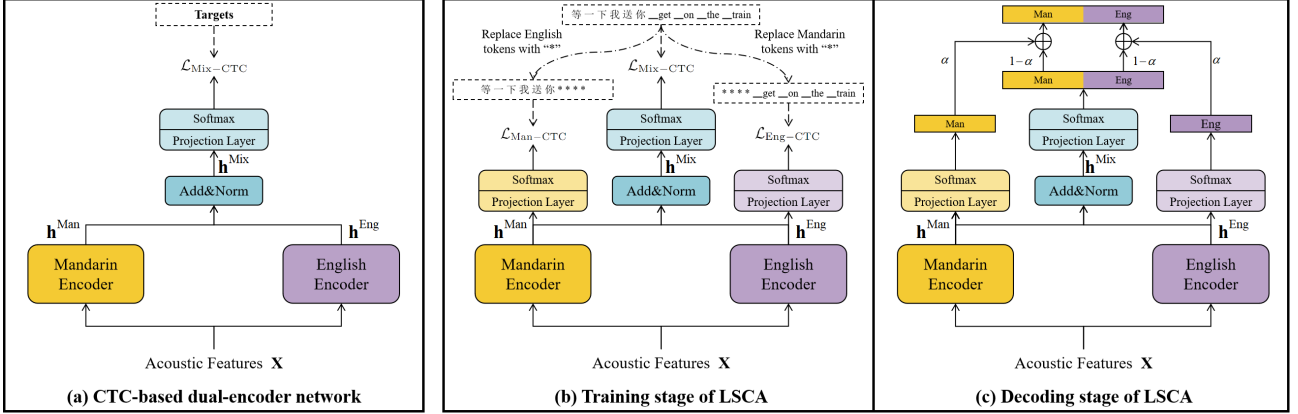


Figure 1: The CTC-based dual-encoder network and the proposed LSCA method. The LSCA method includes the training and decoding stages. In (b), “*” denotes OOV token “unk”. In (c), we only show the situation of the modeling units belonging to Chinese characters or English BPE.

on two pre-trained LSMs.

The rest of this paper is organized as follows. Section 2 describes the CTC-based dual-encoder network for CS ASR as our baseline system. In Section 3, we propose the LSCA method improves the CTC-based dual-encoder network in the training and decoding stages. The experiment details are presented, and the proposed method is evaluated in Section 4. Finally, we conclude this paper and discuss the future work.

2. CTC-based dual-encoder network for CS ASR

Transformer [19] has been introduced into ASR and received comparable performance to the conventional hybrid and other end-to-end approaches [6, 20]. It is also widely used in CS ASR tasks [15, 16, 17, 18, 21, 22, 23]. Because the encoder of the Transformer already has strong language modeling capability, it can achieve comparable performance by only using Connectionist Temporal Classification (CTC) [24] without the decoder during decoding [25, 26]. Since the dual-encoder structure can better use large amounts of monolingual data and capture the individual language attributes, it can perform better than the single-encoder structure in bilingual scenario [15, 16, 17, 18].

In this work, we use a CTC-based dual-encoder network to leverage rich resources monolingual data. The encoder of the Transformer is used as the feature extractor, and CNN is used to do time-scale down-sampling for the input acoustic features before each encoder, and CTC [24] loss is the objective function.

As shown in Figure 1 (a), two LSEs named Mandarin Encoder and English Encoder are initialized by two pre-trained LSMs, respectively. This two pre-trained LSMs are trained with large scale monolingual data, they have learned language-specific knowledge. Therefore, language-specific features \mathbf{h}^{Man} and \mathbf{h}^{Eng} can be extracted from the input acoustic features \mathbf{x} , and they have strong language discrimination. Then, they are added and through a layer-normalized affine transformation to get the mixture features \mathbf{h}^{Mix} :

$$\mathbf{h}^{\text{Man}} = \text{MandarinEncoder}(\mathbf{x}) \quad (1)$$

$$\mathbf{h}^{\text{Eng}} = \text{EnglishEncoder}(\mathbf{x}) \quad (2)$$

$$\mathbf{h}^{\text{Mix}} = \text{LayerNorm}(\mathbf{h}^{\text{Man}} + \mathbf{h}^{\text{Eng}}) \quad (3)$$

The mixture features \mathbf{h}^{Mix} are input to the mixture projection and softmax layer, and then calculate the mixture loss $\mathcal{L}_{\text{Mix-CTC}}$ with original targets. The entire model is finally optimized using a small amount of CS data with $\mathcal{L}_{\text{Mix-CTC}}$.

For modeling units, we use Chinese characters for Mandarin and BPE [27] for English represented by $\phi_{\text{character}}$ and ϕ_{BPE} , respectively. Furthermore, they are combined to generate the Character-BPE modeling units for the mixture model.

3. Language-specific characteristic assistance

This section describes the proposed LSCA method that improves the CTC-based dual-encoder network. The proposed method is shown in the right part of Figure 1. It includes the training stage in Figure 1 (b) and the decoding stage in Figure 1 (c). They can be used alone or in combination.

3.1. The training stage of the LSCA method

In the training stage, in order to add language constraints to the LSEs, we take two language-specific CTC losses (LSCs) into account and interpolate them with the mixture loss $\mathcal{L}_{\text{Mix-CTC}}$ as the new objective function:

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_{\text{Mix-CTC}} + \lambda\mathcal{L}_{\text{LS-CTC}} \quad (4)$$

$$\mathcal{L}_{\text{LS-CTC}} = \frac{\mathcal{L}_{\text{Man-CTC}} + \mathcal{L}_{\text{Eng-CTC}}}{2} \quad (5)$$

where $\mathcal{L}_{\text{LS-CTC}}$ is the combination of two LSCs $\mathcal{L}_{\text{Man-CTC}}$ and $\mathcal{L}_{\text{Eng-CTC}}$, represent the CTC loss for Mandarin encoder and English encoder, respectively. $\lambda \in [0, 1]$ is the weight of $\mathcal{L}_{\text{LS-CTC}}$ in the whole loss. Specially, $\lambda = 0$ and $\lambda = 1$ means only training with $\mathcal{L}_{\text{Mix-CTC}}$ and $\mathcal{L}_{\text{LS-CTC}}$, respectively. When $\lambda = 0$, the system is the same as the baseline. When $\lambda = 1$, there are no extra parameters based on two pre-trained LSMs.

Since we reuse the LSCs of two pre-trained LSMs, the corresponding projection layers have learned language-specific knowledge. However, in the training stage, each LSC will face some sentences with the tokens in another language, and these tokens need to be shielded to prevent LSMs from learning knowledge of another language. For this purpose, we need to

generate language-specific targets for LSCs. As shown in Figure 1 (b), the language-specific targets are generated by mapping the tokens in another language to the OOV token “unk” based on the original targets. Then the LSMs will focus on the parts in the corresponding language and ignore the other parts of the speech just like human beings. $\mathcal{L}_{\text{Mix-CTC}}$ uses the mixture vocabulary while LSCs use the corresponding vocabulary.

3.2. The decoding stage of the LSCA method

In the decoding stage, the LSMs have rich language-specific knowledge, especially after the training stage of our method. In order to make full use of the language-specific knowledge for better decoding, we take the decoding abilities of LSMs into account by combining the output probabilities of LSMs and the mixture model to obtain the final predictions.

However, the output dimensions of two LSMs and the mixture model are not the same. We can not directly add the output probabilities of these three parts. In addition, we found that the probabilities of several modeling units with relatively high probability account for a large proportion. Therefore, it is reasonable to add the corresponding parts of the modeling units proportionally.

When the modeling unit belongs to $\phi_{\text{character}}$ or ϕ_{BPE} , the final probability will be obtained by interpolating the outputs of two parts, the corresponding LSMs and the corresponding part of the mixture model like Figure 1 (c). Since two LSMs and the mixture model can accurately predict the token “blank”, the final probability of “blank” is obtained by interpolating the outputs of these three parts. Because we use “unk” as unseen language token in LSMs, so the token “unk” for the mixture model is true OOV token, and it has little impact on the system performance:

$$P_u = \begin{cases} (1 - \alpha)P_u^{\text{Mix}} + \alpha P_u^{\text{Man}} & u \in \phi_{\text{character}} \\ (1 - \alpha)P_u^{\text{Mix}} + \alpha P_u^{\text{Eng}} & u \in \phi_{\text{BPE}} \\ (1 - \alpha)P_u^{\text{Mix}} + \alpha \frac{P_u^{\text{Eng}} + P_u^{\text{Man}}}{2} & u = \text{blank} \end{cases} \quad (6)$$

where u represents a modeling unit, P_u^{Mix} , P_u^{Man} and P_u^{Eng} represent the probabilities of u in the output probabilities of the mixture model and two LSMs, respectively. $\alpha \in [0, 1]$ controls the weight of the output probabilities of LSMs in the final output probabilities. Moreover, $\alpha = 0$ means that only the mixture model are used for decoding. $\alpha = 1$ means that only the LSMs are used for decoding.

4. Experiments

4.1. Data

Our experiments have two monolingual corpora and one Mandarin-English CS corpus:

Monolingual corpora: (1) AISHELL-2 [28] with 1K hours monolingual Mandarin read speech. (2) Librispeech [29] with 960 hours monolingual English read speech.

CS corpus: The Mandarin-English CS corpus, which is from a contest¹, contains not only Mandarin-English code-switching but also monolingual sentences. It covers daily conversation, tourism, finance, and other fields. The sentences are recorded with a microphone in the silent room or a mobile phone in the office, and in single-channel, 16 kHz, 16-bit PCM format. The information of its training and test sets are described in Table 1.

¹<http://contest.aicubes.cn>

Table 1: *The details of training and test sets of the Mandarin-English CS corpus. M-E means Mandarin-English code-switching.*

Set	Category	Mandarin	English	M-E	SUM
Train	Ratio	26.75%	25.08%	48.17%	100%
	# Utterance	2140	2006	3854	8000
	Duration (h)	1.92	3.50	5.32	10.74
Test	Ratio	20.09%	32.41%	47.50%	100%
	# Utterance	3416	5509	8075	17000
	Duration (h)	3.05	9.63	11.26	23.94

4.2. Experimental setup

We extract 80-dimensional log-Mel filterbanks with a window size of 25 ms and a step size of 10 ms as the acoustic features. SpecAugment [30] is applied with 2 frequency masks ($F = 10$) and 3 time masks ($T = 50$) during all training stages.

Each LSE has a 12-layer Transformer encoder with attention dimension $d_{\text{model}} = 256$ and the feedforward neural network dimension $d_{\text{ffn}} = 1024$. And 4 heads are used for multi-head attention. 2-layer CNN is used to down-sample the time dimension of the input features to one quarter. All models are trained for 50 epochs, and warmup [19] is used for first 250K iterations for pre-training two monolingual models, and 2500 for training with CS data, because of the smaller data scale of CS training data. The dropout is 0.1 to avoid overfitting. The maximum number of frames in one batch is 10K. The last 5 checkpoints are averaged and greedy search is used for decoding.

There are about 3K Chinese characters and 5K English BPE for the modeling units, OOV character or BPE units are mapped to “unk” token. We report a mix error rate (MER) for the CS test set with character error rate (CER) for the Mandarin part and word error rate (WER) for the English part.

Table 2: *Performance (MER%) of different systems on the CS test set. The CTC-based dual-encoder network is our baseline system.*

Type	System	MER
Hybrid	Kaldi(sMBR)	38.71
E2E	CTC-based single-encoder network	29.18
	CTC-based dual-encoder network	27.35

4.3. Evaluation of baseline system

There are three different systems, as shown in Table 2, including one Hybrid and two E2E systems. The Kaldi system [31] is from the official organizers of the contest². The Kaldi system first is trained with about 150 hours Mandarin read speech AISHELL-1 [32] and 100 hours English read speech *train_clean_100* of Librispeech [29] with 3-ways speed perturbation, then the model is trained with the CS training data and sMBR criterion. In addition, We also train a CTC-based single-encoder network, and the other sets are the same as the CTC-based dual-encoder network. The MER of the CTC-based dual-encoder network is 27.35%, which performs better than the other two systems and shows it is a strong baseline system for the experiments.

²<https://github.com/10jqka-aicubes/code-switching-contest>

4.4. Comparative study on the LSCA method

In Table 3, each column shows the impacts of different weights λ of LSCs on model performance in the training stage of LSCA. Each row shows the impacts of different weights of LSMs output probabilities on decoding results in the decoding stage of LSCA. When $\lambda = 0$ and $\alpha = 0$, it is the same as the baseline system.

Table 3: The impacts of different values of λ and α on the model performance (MER%). “-” means the results are unavailable for those sets.

λ	α						
	0	0.1	0.3	0.5	0.7	0.9	1.0
0	27.35	27.02	25.98	25.56	30.30	34.69	36.50
0.1	26.56	26.25	25.34	24.31	24.05	24.32	24.54
0.3	26.14	25.81	24.91	23.91	23.49	23.60	23.68
0.5	26.01	25.67	24.74	23.75	23.29	23.31	23.39
0.7	25.87	25.52	24.63	23.69	23.13	23.16	23.27
0.9	26.34	25.92	24.86	23.77	23.23	23.24	23.31
1.0	-	-	-	-	-	-	23.57

When $\alpha = 0$, only the training stage changes to the LSCA method, the decoding stage is the same as the baseline system. The system can achieve up to 5.4% relative error reduction over the baseline system at $\lambda = 0.7$. This proves that adding language constraints to LSEs can improve the acoustic representation of each language and improves the outputs of the mixture model.

When $\lambda = 0$, the training stage is the same as the baseline system, only the decoding stage changes to the LSCA method. The system can obtain up to a relative 6.5% error reduction at $\alpha = 0.5$. This shows that we better take advantage of language-specific knowledge of LSMs to improve the decoding results.

Furthermore, the system greatly improves when the training and decoding stages change to the LSCA method. We can get the best result at $\lambda = 0.7$ and $\alpha = 0.7$, and the relative error reduction is up to 15.4% compared to the baseline system. This proves that our decoding method can better combined with our training method, and fully demonstrates the effectiveness of our method.

When $\alpha = 1$, only the outputs of LSMs are used for decoding, the system can achieve good performance without using the mixture model for decoding. Therefore, we argue that even without any extra parameters based on two pre-trained LSMs, the system still performs well by using the proposed LSCA method for CS ASR. Then, we let $\lambda = 1$ and $\alpha = 1$, only training the model with LSCs, and decoding with the outputs of LSMs. The relative error reduces up to 13.8% compared to the baseline system. Therefore, we think the system can also perform well for CS ASR, even directly decoding on two pre-trained monolingual models by using our decoding method, and we get 47.84% MER on the CS test, which provides a good solution for directly using large-scale pre-trained monolingual models for CS ASR.

4.5. Analyse the outputs of two LSMs

We visualize the top one probabilities and the decoding results of two LSMs before and after the training stage of LSCA. As shown in the upper part of Figure 2, the monolingual model has higher confidence when the speech is in the corresponding language, like the triangle markers in the solid box, but has lower

confidence when in unseen languages, like the square markers in the solid box. Moreover, monolingual models tend to map the speech in unseen languages to the OOV token, like the Mandarin predictions in the dotted box.

As shown in the under part of Figure 2, after the training stage of the LSCA method, LSMs have higher confidence when the speech is in the corresponding language, like the triangle and square markers. Because we map the tokens in another language to “unk” based on the original targets in the training stage of the LSCA method. Then, LSMs predict the tokens in another language to “unk” with high confidence, like the predictions of Mandarin model and English model. This fully demonstrates the effectiveness of the proposed LSCA method.

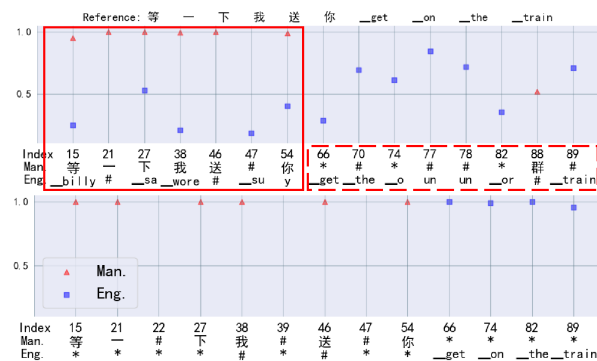


Figure 2: Visualization of the top one probabilities and the decoding results of two LSMs. The upper part belongs to two pre-trained monolingual models. The under part belongs to two LSMs after the training stage of the LSCA method with $\lambda = 0.7$. For better display, we denote “unk” and “blank” tokens as “*” and “#” respectively. We remove those frames which are both “blank” predicted by two models. Furthermore, we do not display the probabilities of “unk”. “Index” means frame index in the outputs of one utterance. “Man.” and “Eng.” mean the predictions of Mandarin and English models, respectively.

5. Conclusion and future work

In this paper, we proposed a LSCA method that improves the training and decoding stages based on the CTC-based dual-encoder network for CS ASR. The proposed training and decoding method can work alone or in combination. Experimental results show that we can obtain up to 15.4% relative error reduction compared to the baseline system by combining the proposed training and decoding method. In addition, the system processes CS ASR tasks well without adding extra parameters or even retraining based on two monolingual pre-trained models by using the proposed method. This provides a good solution for quickly and directly using large-scale monolingual models for CS ASR.

For future work, we will further try our method on larger CS corpus. For model structure, we will introduce our method into the encoder-decoder structure. We will try to apply our method to multilingual speech recognition.

6. Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62176182 and Alibaba Group through Alibaba Innovative Research Program.

7. References

- [1] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International conference on machine learning*. PMLR, 2014, pp. 1764–1772.
- [2] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 6645–6649.
- [3] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.
- [4] S. Kim, T. Hori, and S. Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4835–4839.
- [5] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid ctc/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.
- [6] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5884–5888.
- [7] N. T. Vu, D.-C. Lyu, J. Weiner, D. Telaar, T. Schlippe, F. Blaicher, E.-S. Chng, T. Schultz, and H. Li, "A first speech recognition system for mandarin-english code-switch conversational speech," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 4889–4892.
- [8] S. Sitaram, K. R. Chandu, S. K. Rallabandi, and A. W. Black, "A survey of code-switched speech and language processing," *arXiv preprint arXiv:1904.00784*, 2019.
- [9] 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," *arXiv preprint arXiv:1811.00241*, 2018.
- [10] 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 *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6056–6060.
- [11] K. Li, J. Li, G. Ye, R. Zhao, and Y. Gong, "Towards code-switching asr for end-to-end ctc models," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6076–6080.
- [12] S. Shah, B. Abraham, S. Sitaram, V. Joshi *et al.*, "Learning to recognize code-switched speech without forgetting monolingual speech recognition," *arXiv preprint arXiv:2006.00782*, 2020.
- [13] Y. Khassanov, H. Xu, V. T. Pham, Z. Zeng, E. S. Chng, C. Ni, and B. Ma, "Constrained output embeddings for end-to-end code-switching speech recognition with only monolingual data," *arXiv preprint arXiv:1904.03802*, 2019.
- [14] K. Taneja, S. Guha, P. Jyothi, and B. Abraham, "Exploiting monolingual speech corpora for code-mixed speech recognition," in *INTERSPEECH*, 2019, pp. 2150–2154.
- [15] Y. Lu, M. Huang, H. Li, J. Guo, and Y. Qian, "Bi-encoder transformer network for mandarin-english code-switching speech recognition using mixture of experts," in *INTERSPEECH*, 2020, pp. 4766–4770.
- [16] S. Zhang, Y. Liu, M. Lei, B. Ma, and L. Xie, "Towards language-universal mandarin-english speech recognition," in *INTERSPEECH*, 2019, pp. 2170–2174.
- [17] X. Zhou, E. Yilmaz, Y. Long, Y. Li, and H. Li, "Multi-encoder-decoder transformer for code-switching speech recognition," *arXiv preprint arXiv:2006.10414*, 2020.
- [18] M. S. M. NJ, V. M. Shetty, and S. Umesh, "Investigation of methods to improve the recognition performance of tamil-english code-switched data in transformer framework," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7889–7893.
- [19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [20] T. Nakatani, "Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration," in *INTERSPEECH*, 2019.
- [21] S. Zhang, J. Yi, Z. Tian, Y. Bai, J. Tao, X. Liu, and Z. Wen, "End-to-end spelling correction conditioned on acoustic feature for code-switching speech recognition," *INTERSPEECH*, pp. 266–270, 2021.
- [22] S. Zhang, J. Yi, Z. Tian, Y. Bai, J. Tao *et al.*, "Decoupling pronunciation and language for end-to-end code-switching automatic speech recognition," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6249–6253.
- [23] Z. Huang, P. Wang, J. Wang, H. Miao, J. Xu, and P. Zhang, "Improving transformer based end-to-end code-switching speech recognition using language identification," *Applied Sciences*, vol. 11, no. 19, p. 9106, 2021.
- [24] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.
- [25] B. Zhang, D. Wu, Z. Yao, X. Wang, F. Yu, C. Yang, L. Guo, Y. Hu, L. Xie, and X. Lei, "Unified streaming and non-streaming two-pass end-to-end model for speech recognition," *arXiv preprint arXiv:2012.05481*, 2020.
- [26] Z. Yao, D. Wu, X. Wang, B. Zhang, F. Yu, C. Yang, Z. Peng, X. Chen, L. Xie, and X. Lei, "Wenet: Production oriented streaming and non-streaming end-to-end speech recognition toolkit," *arXiv preprint arXiv:2102.01547*, 2021.
- [27] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909*, 2015.
- [28] J. Du, X. Na, X. Liu, and H. Bu, "Aishell-2: Transforming mandarin asr research into industrial scale," *arXiv preprint arXiv:1808.10583*, 2018.
- [29] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.
- [30] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.
- [31] 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," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF. IEEE Signal Processing Society, 2011.
- [32] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, "Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline," in *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*. IEEE, 2017, pp. 1–5.