TRANSFORMER-TRANSDUCERS FOR CODE-SWITCHED SPEECH RECOGNITION

Siddharth Dalmia

Carnegie Mellon University, USA
sdalmia@cs.cmu.edu

Yuzong Liu, Srikanth Ronanki, Katrin Kirchhoff

Amazon AWS AI, USA
{liuyuzon,ronanks,katrinki}@amazon.com

ABSTRACT

We live in a world where 60% of the population can speak two or more languages fluently. Members of these communities constantly switch between languages when having a conversation. As automatic speech recognition (ASR) systems are being deployed to the real-world, there is a need for practical systems that can handle multiple languages both within an utterance or across utterances. In this paper, we present an end-to-end ASR system using a transformer-transducer model architecture for code-switched speech recognition. We propose three modifications over the vanilla model in order to handle various aspects of code-switching. First, we introduce two auxiliary loss functions to handle the low-resource scenario of code-switching. Second, we propose a novel mask-based training strategy with language ID information to improve the label encoder training towards intra-sentential code-switching. Finally, we propose a multi-label/multi-audio encoder structure to leverage the vast monolingual speech corpora towards code-switching. We demonstrate the efficacy of our proposed approaches on the SEAME dataset, a public Mandarin-English code-switching corpus, achieving a mixed error rate of 18.5% and 26.3% on test$_{\text{man}}$ and test$_{\text{sge}}$ sets respectively.

Index Terms— code-switching, end-to-end, neural transducers

1. INTRODUCTION

Code-switching (CS) refers to the phenomenon of two or more languages used by one speaker in a single conversation. CS widely exists in multilingual communities, which corresponds to around 60% of the world's population [1]. Examples include code-switching between Mandarin and English or between Spanish and English [2, 3]. Code-switching can occur either at an utterance level (extra-sentential CS) or within an utterance (intra-sentential CS).

While there are numerous studies on building multilingual ASR [4, 5, 6, 7], these systems typically assume that the input speech is from native speakers that do not mix different languages. However, this assumption is often impractical as speakers are bi/multilingual and continuously switch between their native language and their language of professional proficiency [3]. Such mixed speech poses a severe challenge to multilingual ASR systems due to different phone sets among languages, the influence of native language in pronunciation, and insufficient CS training data [3, 8, 9]. These effects are compounded in intra-sentential CS, which is the focus of our work.

There are promising approaches for building hybrid ASR systems for CS speech. However, these require cumbersome language-specific handcrafted features like phone merging between languages for acoustic models [9] and linguistic structures like part-of-speech tags and language ID for language modeling [8]. Additionally, the unbalanced language distribution within CS utterances can lead to a poor n-gram language model [8], suggesting the need for handling longer contexts. End-to-end ASR systems [10, 11, 12] are becoming increasingly popular, since they do not require explicit alignments and usually have fewer hyperparameters to tune. Despite their simplistic design, end-to-end ASR systems need larger amounts of training data than the hybrid based models leading to inferior performance on data-sparse tasks like code-switching [13, 14]. This behavior is starting to turn around [15] with data-augmentation techniques like SpecAugment [16] and joint training with alignment-based loss functions like connectionist temporal classification (CTC) loss [17].

In this work, we propose the use of neural transducers for code-switched ASR. Unlike CTC, where each output label is conditionally independent of the others given the input speech, neural transducers condition the output on all the previous labels. Unlike attention-based encoder-decoder models, transducer models learn explicit input-output alignments, making it robust towards long utterances [18]. In particular, we focus on adapting the transformer-transducer (T-T) model to code-switched ASR [19, 20]. The T-T model replaces the recurrent neural networks with non-recurrent multi-head self-attention transformer encoders [21]. Transformers allow superior modeling of long-term temporal dependencies in speech data [22]. As noted earlier, the ability to handle longer contexts is crucial for intra-sentential CS ASR. The language structure of a new phrase might depend on the structure before the language-switch [2].
We summarize the contributions of this paper below:

- We present training strategies and insights towards improving transformer-transducer models in the data-sparse scenario of code-switching by extending the model with two auxiliary loss functions: a language model (LM) loss and a CTC loss (§2.1.1).
- To address intra-sentential CS, we propose language ID (LID) aware masked training for the transformer-transducer (§2.1.2).
- To leverage additional monolingual corpora, we propose a multi-label/multi-audio encoder framework for the T-T model (§2.2).

On the Mandarin-English CS SEAME corpus, our proposed architecture improves over the previous RNN-transducer baseline [14] by around 15% (absolute) without using any additional data and by 17% with only 200 hours of monolingual data in each language (§4).

2. BACKGROUND AND PROPOSED APPROACH

Given an input speech sequence $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T)$, where $\mathbf{x}_t \in \mathcal{R}^d$ is a d dimensional speech feature vector and T is the input sequence length, and target transcription $\mathbf{y} = (y_1, y_2, \ldots, y_L)$, where $y_l \in \mathcal{V}$ is the output label and L is sequence length, the transducer loss [11] models the posterior of the output label sequence as the marginalization over all possible alignments $\mathbf{z} \in \mathcal{Z}(\mathbf{x}, \mathbf{y})$:

$$P(\mathbf{y}|\mathbf{x}) = \sum_{\mathbf{z}\in\mathcal{Z}(\mathbf{x},\mathbf{y})} P(\mathbf{z}|\mathbf{x}) \quad = \sum_{\mathbf{z}\in\mathcal{Z}(\mathbf{x},\mathbf{y})} \prod_{i=1}^{T+L} P(z_i|\mathbf{x}, y_{1:l_{i-1}})$$

Work done while Siddharth was interning at AWS AI

To appear in *Proc. ICASSP 2021, June 6-11, 2021*

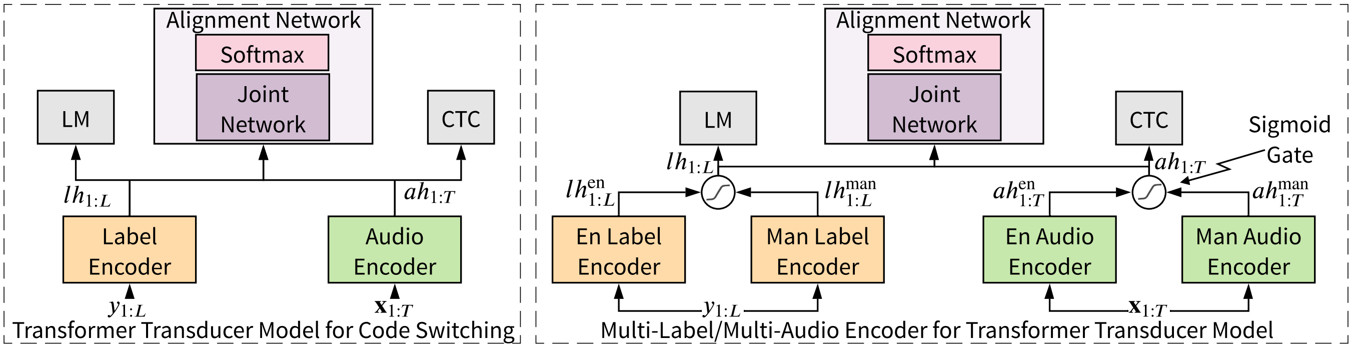

Fig. 1. Proposed Transformer-Transducer Model for Code-Switching

where $\mathcal{Z}(\mathbf{x}, \mathbf{y}) \subset \{\mathcal{V} \cup \phi\}^{T+L}$ corresponds to all possible values y can take in the alignment path $(\mathbf{x}, \mathbf{y})$ following the transducer lattice [23]. A valid alignment path after removing the blank symbol ϕ gives the target sequence $\mathbf{y}$. $y_{1:l_{i-1}}$ corresponds to the non-blank labels chosen in the alignments till z_i. The transducer model consists of three components for parameterizing $P(\mathbf{z}|\mathbf{x})$ – a label encoder, an audio encoder, and an alignment network. The label encoder encodes the output sequence $y_{1:l_{i-1}}$ as lh, the audio encoder encodes the input audio frames $\mathbf{x}_{1:T}$ as ah and the alignment network prepares the output lattice $\mathbf{z}$ given the audio encoder and label encoder outputs using a joint network.

When transducer models were first introduced (RNN-T), the audio and label encoders used Long Short-Term Memory models (LSTMs) [11]. More recently, they have been replaced with transformer encoders [21] in the transformer-transducer model (T-T) [19, 20]. The joint network uses feed-forward layers and a tanh non-linearity to transform the label and audio encodings to lie on orthogonal axes, resulting in an output lattice of the shape $(T, L, \mathcal{V} \cup \phi)$. For simplicity, we ignore the extra ϕ label encoding added to the beginning of every utterance (more details in [11]).

Next, we describe our proposed adaptations of the transformer-transducer model to CS data. First, we present a modification to the training of the T-T model that serves to handle the 'data-sparsity' and 'intra-sentential' aspects of code-switching. Next, we propose multi-label and multi-audio encoders to leverage monolingual corpora. Figure 1 presents the schematics of our proposed architecture.

2.1. Training Transformer-Transducer for Code-Switching

We use the vanilla T-T model as the base model towards training a CS ASR system. For all our models, we use two data augmentation strategies: three-way speed perturbation [24] and SpecAugment [16]

2.1.1. CTC and LM Joint Training

Collecting code-switched data is expensive, making data-sparsity a challenge when training code-switched ASR models. To overcome this issue, we jointly train the audio and label encoders towards auxiliary tasks along with the standard transducer loss ($\mathcal{F}_{\text{Transducer}}$) .

The task of the audio encoder is to learn frame-level audio representations. Audio encoders trained solely with the CTC loss learns such conditionally independent frame-level representations with low amounts of training data [7, 25]. Additionally, previous work has demonstrated the effectiveness of the CTC loss in learning better alignments when pre-training in transducers [26] and joint-training in encoder-decoder models [17]. Taking motivation from these works, we use the CTC loss ($\mathcal{F}_{\text{CTC}}$) as an auxiliary task to provide supervision to the audio encoder.

Similarly, the task of the label encoder is to encode the past context, which is used in predicting the next word during alignment with the transducer loss. Setting aside the audio alignment (which is done using the alignment network), the label encoder's task is very similar to that of the language model, allowing us to use the next word prediction task ($\mathcal{F}_{\text{LM}}$), as an auxiliary task for providing supervision to the label encoder. Using the two auxiliary tasks, our overall objective function is defined as follows:

$$\mathcal{F}_{\text{obj}} = \mathcal{F}_{\text{Transducer}}\left(\mathbf{x}, \mathbf{y}\right) + \lambda_{\text{CTC}}\mathcal{F}_{\text{CTC}}\left(\mathbf{x}, \mathbf{y}\right) + \lambda_{\text{LM}}\mathcal{F}_{\text{LM}}\left(\mathbf{y}\right)$$

where λ_{CTC} and λ_{LM} are tunable weights assigned to the auxiliary tasks. The left side of the Figure 1, presents the overall transducer model being used for our experiments. We use this model for all our experiments when training on code-switched data. The ablations for individual objective functions are discussed in §4.1.

2.1.2. LID aware Masked Training of Label Encoder

Intra-sentential CS can be a big challenge for transducer models as the alignment network and the label encoder have to learn alignments for different languages and learn when to switch between languages for next word prediction. To counter this, we propose two modifications to the training process of the transducer network:

- **Randomly masking target tokens in the label encoder**: The mask tag will help the alignment network to focus on learning audio alignments independent of the target token. Additionally, in conversational speech (our target use case), the mask tags can help make the label encoder robust towards the irregularities in utterances such as disfluency and non-speech sounds. This approach could be thought of as analogous to learning back-off language models or ensembles of language models.
- **Adding an LID tag to the target sequence whenever there is a switch in the language**: The LID tags help the label encoder language switches within an utterance. Additionally, they teach the alignment network which language it is currently working on.

The motivation behind the masking strategy comes from systematic dropout used for language modeling [28, 29]. Like SpecAugment [16], we employ a similar strategy by randomly masking 40% of the text during training. Each masked word is replaced by a `<mask>` token. For the LID tags, we add `<en>` or `<man>` tag for the start of an English segment or Mandarin segment respectively. The ablations for individual techniques will be discussed in §4.2.

Architecture	Model Name	Monolingual Data	dev	test_{man}	test_{sge}
LF-MMI	Zhou et. al. (2020) [15]	✗	-	19.0%	26.6%
Att Enc-Dec	Khassanov et. al. (2019) [13]	✗	-	49.5%	58.9%
Att Enc-Dec	Zeng et. al. (2019) [27]	✗	-	26.4%	36.1%
Att Enc-Dec	Zhou et. al. (2020) [15]	✗	-	18.9%	26.2%
Att Enc-Dec	Our Implementation	✗	20.8%	19.2%	26.9%
Transducer	Zhang et. al. (2020) [14]	✗	-	33.3%	44.9%
Transducer	Our Proposed Model	✗	23.4%	**20.2%**	**27.7%**
Transducer	+Monolingual Data	✓	22.2%	**18.5%**	**26.3%**

Table 1. Results presenting the overall performance (% MER) of our proposed transformer-transducer model. The best performing transducer models are **highlighted**. Results from previous papers and our own implementation of the Att Enc-Dec are shown for comparison.

2.2. Leveraging Monolingual Corpora for Code-Switched ASR

While procuring CS data is difficult, it is relatively easier to find large monolingual corpora of the languages present. However, utilizing these monolingual corpora is challenging [30] owing to factors like pronunciation shift, accent shift and phone influence that occur in code-switched data [9, 31]. Previous work [15, 32] has attempted different strategies to overcome this issue. Inspired by the model in [15], we propose a multi-label/multi-audio encoder for handling monolingual corpora in transformer-transducer, as depicted on the right side of Figure 1. Here, we have individual audio and label encoders for each language present in the CS data. Each encoder accepts monolingual data in its respective language as well as CS data. The information from them are combined using a sigmoid gate α_{enc}, with learnable weights w_{enc} and w_α:

$$\alpha_{\text{enc}} = \text{sigmoid}(w_\alpha(\tanh(w_{\text{enc}}^{\text{man}}(h^{\text{man}}) + w_{\text{enc}}^{\text{en}}(h^{\text{en}}))))$$
$$h = \alpha_{\text{enc}} * h^{\text{man}} + (1 - \alpha_{\text{enc}}) * h^{\text{en}}$$

where h_{man} and h_{en} are outputs of either of the label/audio encoder. We learn the gate only for code-switched data and force the gate to 1 if the input is Mandarin only and 0 for English only. This is done automatically for each mini-batch. This allows us to train on all the data (monolingual and code-switched) jointly end-to-end, without the need for pre-training the individual encoders, as needed in [15]. In order to get improved results over the target dataset, the last few thousand updates are only from the CS corpora. We use this model when leveraging monolingual data and the contribution of the multi-label encoder and multi-audio encoder has been studied in §4.3.

3. DATASET AND EXPERIMENTAL SETUP

All our transformer-transducers models are implemented using the ESPnet library [33]. We follow the standard data prep in [33], where we use global mean-variance normalized 83 log-mel filterbank and pitch features from 16kHz audio. We augment the data with speed perturbation of 0.9 and 1.1. For SpecAugment, we use the SS augmentation policy presented in [16]. For the audio encoder, we subsample the input features by a factor of 4 using convolutions [33], followed by 12 transformer encoder blocks with 1024 feed-forward dim and 512 attention dim with 8 attention heads and a dropout of 0.1. For the label encoder, we use 4-layer transformer encoder blocks with the same dimensions, with an attention dropout of 0.5, a positional-embedding dropout of 0.1, and a dropout of 0.3 for all other components. We mask 40% of the tokens in the label sequence for each utterance during training, and set $\lambda_{\text{CTC}} = 0.5$ and $\lambda_{\text{LM}} = 0.4$. We train our models with an effective mini-batch size of 192 utterances. We use the Adam optimizer with the inverse square root decay learning rate schedule presented in [33] with transformer-lr scale set to 2.0 and 25K warmup steps. We keep a validation loss patience of 5, after which we stop training. For decoding, we use the beam-search algorithm described in [11] with a beam size of 20.

SEAME Corpus: For our experiments, we use the SEAME corpus [34], a conversational Mandarin-English CS corpus collected in Singapore, consisting of around 134 speakers (100 hours). We hold out 6 speakers (4.7 hours) from the training data as our development set to do hyper-parameter tuning and for studying ablations. The SEAME corpus has two official test sets, test_{man} and test_{sge}, each consisting of 10 speakers. The test_{man} is biased towards Mandarin speech and test_{sge} towards English. In the SEAME corpus, we have $\approx$ 2.5K Mandarin characters. We use subword-nmt [35] to convert the English target vocabulary to have a similar target size by doing 2K byte-pair encoding (BPE) merges. We run the BPE merges after splitting English into individual words to avoid merges with Mandarin, which can be present in the context due to code-switching.

Monolingual Corpora: We use AISHELL-1 [36], a 150-hour Mandarin corpus, and TEDLIUMv2 [37], a 211-hour English corpus, as our monolingual speech dataset for the method described in §2.2. We prepare 4K English BPE units to match the $\approx$ 4K character set for Mandarin on the combined monolingual and CS corpora.

Evaluation: We evaluate our models with Mixed Error Rate (MER), which refers to the standard WER metric but computes edits at the character-level for Mandarin and word-level for English. We use the NIST sclite scoring script to score the models and report all numbers without any post-normalization for either languages.

4. RESULTS

Table 1 presents the overall performance of our proposed transformer-transducer model. We improve over the previously published transducer model by 13.1% and 17.2% absolute MER over the two test sets, test_{man} and test_{sge}, without using any monolingual training data. Our base transformer-transducer model also performs considerably better than most attention-based encoder-decoder models on this dataset and comes close to the best model [15] and our implementation of the same. We also see that using monolingual data with the proposed multi-label encoder can improve the model further, giving us an MER of 18.5% and 26.3% on test_{man} and test_{sge} respectively.

4.1. CTC and LM Joint Training

Table 2 shows the improvements in the development set by using CTC and LM loss as auxiliary loss functions along with the trans-

T-T Model	Dev Set
Vanilla Transducer	25.6%
+ CTC Loss	24.9%
+ LM Loss	25.1%
+ MaskedTraining	24.0%
+ LIDMaskedTraining	23.4%

Table 2. Ablation showing the contribution of the individual proposed modifications to the vanilla transformer-transducer model.

Model	Utterance
Reference	its like wah you waste my time
Base T-T	its like wah WE ALWAYS my time
+ LM Loss	its like wah you waste my time
Reference	读 engineering science then 他
Base T-T	TWO engineering LEH 他
+ LM Loss	TO engineering science AND HER

Table 3. Example utterances showing how jointly training with an LM auxiliary loss improves the grammar of the decoded sentence.

ducer loss. We see that CTC loss improves the dev set MER from 25.6% to 24.9%. Although using LM loss causes a drop in performance, it actually helps stabilize the training of these models. With the use of LM loss, we are able to double our learning rate, reducing the convergence time from ≈ 24 hours to ≈ 15 hours. We also see in Table 3 that joint training with LM loss makes the model output grammatically coherent. This is true even though we use the LM loss during training and do not perform any LM rescoring.

4.2. LID aware Masked Training

Table 2 also presents the contribution of masked training with and without LID tags. We see that just the masked training improves the transducer model by around 1% MER, and with the LID tags, it improves further by 0.6% MER. We noticed that by incorporating *only* LID tags, as done in [14], our model has a negligible change in performance, indicating that our vanilla model is already well trained.

4.3. Multi-Label/Multi-Audio Encoder

Table 4 shows the ablation of multi-label and multi-audio encoders for leveraging monolingual training data. We see that the multi-label encoder improves considerably over our best transformer-transducer model. We do not observe significant improvement when using the multi-audio encoder likely because it is difficult to learn language boundaries in the acoustic space. We also noticed that using both multi-audio and multi-label encoder does not cascade the improvements. We believe that using larger corpora, as in [15], could help overcome this issue by making the audio encoders learn better acoustic representations and expose the model to a wider set of speakers, leading to better generalizability on code-switched data.

5. RELATION TO PRIOR WORK

This section discusses previous literature that this work takes inspiration from and explains how our work extends from them.

T-T Model	Dev Set
Proposed Transducer Model	23.4%
+ Multi Audio Encoder	23.1%
+ Multi Label Encoder	22.2%

Table 4. Ablation showing the contribution of multi-label and multi-audio encoder when trying to leverage monolingual training data.

Code-Switching Background The first speech recognizer for code-switched data [38] was trained on the SEAME corpus [34]. They looked at phone-merging techniques to handle the two languages in acoustic modeling, explored further in [9, 31], and generating code-switched text data for language modeling, studied more in [32, 39]. Since then, different approaches have been applied to improve code-switched speech recognition like speech chains [40], transliteration [41], and translation [42]. Authors in [14, 27, 43] focus on tracking the language switch points, similar to our LID aware training. To leverage monolingual data, different techniques have been proposed like constrained output embedding [13], multi-encoder-decoder networks [15] and LID integrated acoustic modeling [30].

Transducer Loss Background Transducer models are widely used for online speech recognition for its streaming capabilities and low memory footprint [44, 45, 46]. The transducer loss is an alignment-based loss that does the task equivalent of cross-attention in encoder-decoder models [47], making it useful in generating closed captioning [26, 45]. With the introduction of self-attention based transformer models [21] and data augmentation techniques like SpecAugment [16], transducers have also seen competitive performance to the attention-based encoder-decoder models [19]. With transducer models, CTC loss has been primarily studied as a pre-training objective [18, 26]. This process requires a fine-tuning phase which can be cumbersome due to the decisions involved in neural-network optimization like learning rate, optimizer, etc. SpecAugment [16] further increases the complexity while pre-training as the audio encoder is already invariant to the noise [48]. The label encoder in transducers behaves like a language model that can benefit from large amounts of text data [11]. Due to the code-switching nature of our task, obtaining text data is difficult; instead, we use the next word prediction task as a joint training objective.

6. CONCLUSION

In this paper, we present a transformer-transducer model for code-switched speech recognition. We show significant improvements over the previous transducer model and perform at par with the best attention-based encoder-decoder and LF-MMI based hybrid models. We propose modifications to improve transformer-transducers training towards the data-sparse and intra-sentential nature of code-switched corpora. Additionally, we propose a multi-label/audio encoder framework to leverage monolingual data to improve recognition performance. In the future, we would like to extend this mechanism to further train with unlabeled audio from either monolingual or code-switched sources in an unsupervised manner.

7. ACKNOWLEDGEMENTS

We are grateful to the AWS Speech Science team, Shruti Rijhwani, Samridhi Choudhary and Brian Yan for their valuable feedback.

8. REFERENCES

[1] "Multilingual People," http://ilanguages.org/bilingual.php, Accessed: 2020-10-15.

[2] P. Muysken, "Code-switching processes: Alternation, insertion, congruent lexicalization," *Language choices: Conditions, constraints, and consequences*, 1997.

[3] S. Sitaram, K. R. Chandu, S. K. Rallabandi, and A. W Black, "A Survey of Code-switched Speech and Language Processing," *arXiv:1904.00784*, 2019.

[4] K. Knill, M. Gales, S. P Rath, P. C Woodland, C. Zhang, and S. Zhang, "Investigation of multilingual deep neural networks for spoken term detection," in *Proc. ASRU*, 2013.

[5] F. Grézl, E. Egorova, and M. Karafiát, "Study of large data resources for multilingual training and system porting," in *Proc. SLTU*, 2016.

[6] J. Cho, M. K. Baskar, R. Li, M. Wiesner, S. H. Mallidi, et al., "Multilingual sequence-to-sequence speech recognition: architecture, transfer learning, and language modeling," in *Proc. SLT*, 2018.

[7] S. Dalmia, R. Sanabria, F. Metze, and A. W. Black, "Sequence-based Multi-lingual Low Resource Speech Recognition," in *Proc. ICASSP*, 2018.

[8] H. Adel, N. T. Vu, and T. Schultz, "Combination of Recurrent Neural Networks and Factored Language Models for Code-Switching Language Modeling," in *Proc. ACL*, 2013.

[9] S. Sivasankaran, B. M. L. Srivastava, S. Sitaram, et al., "Phone Merging For Code-Switched Speech Recognition," in *Workshop on Computational Approaches to Linguistic Code-Switching*, 2018.

[10] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks," in *Proc. ICML*, 2006.

[11] A. Graves, "Sequence Transduction with Recurrent Neural Networks," in *ICML Representation Learning Workshop*, 2012.

[12] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. ICASSP*, 2016.

[13] Y. Khassanov, H. Xu, V. T. Pham, Z. Zeng, E. S. Chng, et al., "Constrained Output Embeddings for End-to-End Code-Switching Speech Recognition with Only Monolingual Data," in *Proc. Interspeech*, 2019.

[14] S. Zhang, J. Yi, Z. Tian, J. Tao, and Y. Bai, "RNN-Transducer with language bias for end-to-end Mandarin-English code-switching speech recognition," *arXiv:2002.08126*, 2020.

[15] X. Zhou, E. Yilmaz, Y. Long, Y. Li, and H. Li, "Multi-Encoder-Decoder Transformer for Code-Switching Speech Recognition," in *Proc. Interspeech*, 2020.

[16] D. S. Park, W. Chan, Y. Zhang, C. Chiu, et al., "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," in *Proc. Interspeech*, 2019.

[17] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multi-task learning," in *Proc. ICASSP*, 2017.

[18] A. Zeyer, A. Merboldt, R. Schlüter, and H. Ney, "A New Training Pipeline for an Improved Neural Transducer," *arXiv:2005.09319*, 2020.

[19] Q. Zhang, H. Lu, H. Sak, A. Tripathi, et al., "Transformer transducer: A streamable speech recognition model with transformer encoders and RNN-T loss," in *Proc. ICASSP*, 2020.

[20] C. Yeh, J. Mahadeokar, K. Kalgaonkar, Y. Wang, D. Le, et al., "Transformer-Transducer: End-to-End Speech Recognition with Self-Attention," *arXiv:1910.12977*, 2019.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, et al., "Attention is All you Need," in *Proc. NeurIPS*, 2017.

[22] L. Dong, S. Xu, and B. Xu, "Speech-Transformer: A No-Recurrence Sequence-to-Sequence Model for Speech Recognition," in *Proc. ICASSP*, 2018.

[23] E. Battenberg, J. Chen, R. Child, A. Coates, et al., "Exploring neural transducers for end-to-end speech recognition," in *Proc. ASRU*, 2017.

[24] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, et al., "The Kaldi Speech Recognition Toolkit," in *Proc. ASRU*, 2011.

[25] S. Tong, P. N. Garner, and H. Bourlard, "Multilingual training and cross-lingual adaptation on CTC-based acoustic model," *arXiv:1711.10025*, 2017.

[26] H. Hu, R. Zhao, J. Li, L. Lu, and Y. Gong, "Exploring Pre-Training with Alignments for RNN Transducer Based End-to-End Speech Recognition," in *Proc. ICASSP*, 2020.

[27] 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 *Proc. Interspeech*, 2019.

[28] Z. Xie, S. I Wang, J. Li, D. Lévy, A. Nie, et al., "Data Noising as Smoothing in Neural Network Language Models," in *ICLR*, 2017.

[29] Y. Gal and Z. Ghahramani, "A Theoretically Grounded Application of Dropout in Recurrent Neural Networks," in *Proc. NeurIPS*, 2016.

[30] K. Li, J. Li, G. Ye, R. Zhao, and Y. Gong, "Towards Code-switching ASR for End-to-end CTC Models," in *Proc. ICASSP*, 2019.

[31] Y. Li, P. Fung, P. Xu, and Y. Liu, "Asymmetric acoustic modeling of mixed language speech," in *Proc. ICASSP*, 2011.

[32] K. Taneja, S. Guha, P. Jyothi, and B. Abraham, "Exploiting Monolingual Speech Corpora for Code-Mixed Speech Recognition," in *Proc. Interspeech*, 2019.

[33] S. Watanabe, T. Hori, S. Karita, T. Hayashi, et al., "ESPnet: End-to-End Speech Processing Toolkit," in *Proc. Interspeech*, 2018.

[34] D. Lyu, T. Tan, C. Siong, and H. Li, "SEAME: a Mandarin-English code-switching speech corpus in south-east asia," in *Proc. Interspeech*, 2010.

[35] R. Sennrich, B. Haddow, and A. Birch, "Neural Machine Translation of Rare Words with Subword Units," in *Proc. ACL*, 2016.

[36] 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 *Proc. O-COCOSDA*, 2017.

[37] A. Rousseau, P. Deléglise, and Y. Estève, "Enhancing the TED-LIUM Corpus with Selected Data for Language Modeling and More TED Talks," in *Proc. LREC*, 2014.

[38] N. T. Vu, D. Lyu, J. Weiner, et al., "A first speech recognition system for Mandarin-English code-switch conversational speech," in *Proc. ICASSP*, 2012.

[39] S. Garg, T. Parekh, and P. Jyothi, "Dual Language Models for Code Switched Speech Recognition," in *Proc. Interspeech*, 2018.

[40] S. Nakayama, A. Tjandra, S. Sakti, and S. Nakamura, "Speech Chain for Semi-Supervised Learning of Japanese-English Code-Switching ASR and TTS," in *Proc. SLT*, 2018.

[41] J. Emond, B. Ramabhadran, B. Roark, et al., "Transliteration Based Approaches to Improve Code-Switched Speech Recognition Performance," in *Proc. SLT*, 2018.

[42] Z. Huang, X. Zhuang, D. Liu, et al., "Exploring Retraining-free Speech Recognition for Intra-sentential Code-switching," in *Proc. ICASSP*, 2019.

[43] C. Shan, C. Weng, G. Wang, et al., "Investigating End-to-end Speech Recognition for Mandarin-english Code-switching," in *Proc. ICASSP*, 2019.

[44] Y. He, T. N. Sainath, R. Prabhavalkar, et al., "Streaming End-to-end Speech Recognition For Mobile Devices," in *Proc. ICASSP*, 2019.

[45] J. Li, R. Zhao, H. Hu, and Y. Gong, "Improving RNN Transducer Modeling for End-to-End Speech Recognition," in *Proc. ASRU*, 2019.

[46] W. Huang, W. Hu, Y. Yeung, and X. Chen, "Conv-Transformer Transducer: Low Latency, Low Frame Rate, Streamable End-to-End Speech Recognition," in *Proc. Interspeech*, 2020.

[47] R. Prabhavalkar, K. Rao, T. N. Sainath, et al., "A Comparison of Sequence-to-Sequence Models for Speech Recognition," in *Proc. Interspeech*, 2017.

[48] S. Dalmia, A. Mohamed, M. Lewis, et al., "Enforcing encoder-decoder modularity in sequence-to-sequence models," *arXiv:1911.03782*, 2019.