BUILDING HIGH-ACCURACY MULTILINGUAL ASR WITH GATED LANGUAGE EXPERTS AND CURRICULUM TRAINING

Eric Sun, Jinyu Li, Yuxuan Hu, Yimeng Zhu, Long Zhou, Jian Xue
Peidong Wang, Linquan Liu, Shujie Liu, Edward Lin, Yifan Gong

ABSTRACT

We propose gated language experts to improve multilingual transformer transducer models without user's language identification (LID) input during inference. We define gating mechanism and LID loss to let transformer experts learn language-dependent information, construct the multilingual transformer block with gated transformer experts and shared transformer layers, and apply linear experts to better regularize joint network. In addition, a curriculum training scheme is proposed to let LID guide gated experts serve their own languages better. Evaluated on English and Spanish bilingual task, our method achieves average 12.5% and 7.3% relative word error reductions over baseline bilingual and monolingual models, obtaining similar results to the upper-bound model trained and inferred with oracle LID. We further explore our method on trilingual, quadrilingual, and pentalingual models, and observe similar advantages as in bilingual models, demonstrating its easy extension to more languages.

***Index Terms*—**Multilingual automatic speech recognition, transformer transducer, language ID, expert

1. INTRODUCTION

While end-to-end (E2E) models have made rapid progress in automatic speech recognition (ASR) [1-8], there are large amount of demands of multilingual ASR models since there are more than 60% people in the world can speak more than 2 languages according to [10]. There have been plenty of efforts to develop E2E multilingual models [11-26], and these models can achieve the comparable or even better ASR performance than monolingual baselines by passing the language identification (LID) information in the form of a one-hot or learnable embedding vector to distinguish different languages. In order to build streaming multilingual ASR systems for lots of practical applications that can perform similarly as the monolingual ones, we should not request users to input any LID information during model inference. One solution is to infer LID as an embedding vector and attach it to the input features [18, 19, 21]. However, this kind of solution either leads to limited improvement due to LID prediction inaccuracy or introduces extra latency for reliable LID prediction [18,19].

In this paper, we propose gated transformer with auxiliary LID loss and linear experts to improve multilingual speech recognition. The gated transformer experts can make compact models and better speech information sharing across different languages. Linear experts can better regularize joint network output, which greatly stabilizes the model training. We further propose a curriculum training strategy for LID input during training to make the experts better learn the corresponding language information. Our model does not need any LID input from users during inference. Our experiments on English and Spanish bilingual models achieve 12.5% relative word error (WER) reduction over the bilingual model baseline without LID as input, similar to the performance of the bilingual model with the oracle LID as input. In addition, our bilingual model can also beat monolingual baselines. We further extend our methods to build trilingual, quadrilingual, and pentalingual models and achieve similar success as bilingual models with only limited model size increase.

2. RELATED WORK

The concept of expert has been applied to ASR in [27, 28]. It has been also explored to solve bilingual code-switching problem as in [29, 30] by using a dedicated encoder as the expert for a specific language. A gate function is defined to combine the output from different experts, and there is no LID loss applied to regularize the experts' outputs. In [31], informed experts based on RNN-transducer with LID input are applied for multilingual ASR. A LSTM model is proposed as a gate to generate scores to combine experts from different languages. In [20], a configurable multilingual model is proposed that is trained once and can be configured as different language combinations. Linear language experts are applied in both encoder and prediction networks.

3. MULTILINGUAL TRANSFORMER TRANSDUCER WITH GATED LANGUAGE EXPERTS

The paper title (on the first page) should begin 1.38 inches (35 mm) from the top edge of the page, centered, completely capitalized, and in Times 14-point, boldface type. The authors' name(s) and affiliation(s) appear below the title in capital and lower case letters. Papers with multiple authors and affiliations may require two or more lines for this information.

3.1 Transformer transducer model

A neural transducer model [4] has three components: an acoustic encoder, a label prediction network, and a joint network. Neural transducer models can use different types of models as encoders such as LSTMs in RNN-T [4] and transformers [7, 8, 9, 17, 20, 21, 22] in transformer transducer (T-T). In this study, we use T-T as the backbone model for the development. Each transformer module in the encoder network is constructed from a multi-head self-attention layer followed by a feed-forward layer. The loss function of neural transducer models is the negative log posterior of output target label $\boldsymbol{y}$ given input acoustic feature $\boldsymbol{x}$ and is defined as

$$L_{rnnt} = -\log P(\boldsymbol{y}|\boldsymbol{x}) \quad (1)$$

which is calculated by the forward-backward algorithm as in [4].

3.2 Gated language transformer experts in encoder

Encoder is the most important component in T-T models. In multilingual speech recognition, if all languages share one encoder, different languages may affect the model performance since they can be confused by each other as discussed in [22]. In this work, we associate each language with its own specific transformer encoder as shown in Figure 1. Different encoders can be combined with a gate g that is defined as

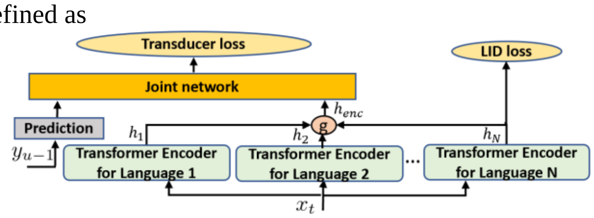

Figure 1: Architecture of multilingual T-T model with separated transformer encoders for different languages

$$O = W_o(\tanh(\textstyle\sum_{i=0}^{N} W_i h_i))) \quad (2)$$

$$g = Softmax(O) \quad (3)$$

where h_i is the encoder embedding from language i, and W_i is a linear matrix that is associated with each language, N is the number of languages. Then the whole encoder network embedding output is

$$h_{enc=} \textstyle\sum_{i=0}^{N} g_i h_i \quad (4)$$

Since gate g combines encoder embedding from different languages, the encoder networks themselves do not realize which language they should serve as the corresponding transformer encoder. In order to make the encoder networks to learn their own corresponding languages, a LID cross entropy (CE) loss is proposed as

$$L_{lid} = CE(O) \quad (5)$$

Therefore, the overall loss is defined as a combination of the original transducer loss and the LID loss as following

$$L = L_{rnnt} + \lambda L_{lid} \quad (6)$$

where λ is the weight to adjust the ratio of these two losses.

One drawback of the above design is each language has its own encoder that makes the model difficult scale up when the number of languages increases. Also, separated encoders for different languages may not be an optimal choice since there are still lots of acoustic conditions, speaker voice characteristics, and even pronunciation similarity that could be shared across different languages. Therefore, we propose another more effective and compact encoder structure for the T-T based multilingual ASR as shown in Figure 2. Instead of building fully separated encoders for each language, different languages can share transformer layers while each language can still have their own corresponding transformer experts that are combined with gate g as defined in Equations (2) and (3) while the LID loss function is defined as following

$$L_{lid} = CE(\textstyle\sum_{l=1}^{M} O_l) \quad (7)$$

where O_l is the logit output as in Equation (2) but from different layer l. Language dependent experts, their gate, and the shared transformer layers can construct a multilingual transformer block as shown in the dotted lines of Figure 2. In a T-T multilingual model, there can be several multilingual transformer blocks included. In addition, we can also apply the shared transformer layers at the bottom of the network since the input filterbank speech features share lots of common characteristics from different languages. With this new structure, the network can share common speech information while learning linguistic knowledge from different languages. Besides, the multilingual model size can be easily controlled by the number of blocks, which is beneficial for scaling up the multilingual models to more languages.

3.3 Gated language linear experts for joint network

Joint network in T-T model combines both acoustic and language information from encoder and prediction networks. Inspired by [20], instead of adding language specific gated linear experts on both encoder and prediction networks, linear experts can be directly applied to the output of the joint network as shown in Figure 2. Let's define $W = \{w_1, w_2, \dots, w_N\}$ is linear matrix combination of w_i (i = 1 to N) that is a linear expert matrix corresponding to a language i. After multiplying this language gated linear matrix, the new joint network output is defined as

$$h'_{joint} = h_{joint}\, W\, G_L \quad (8)$$

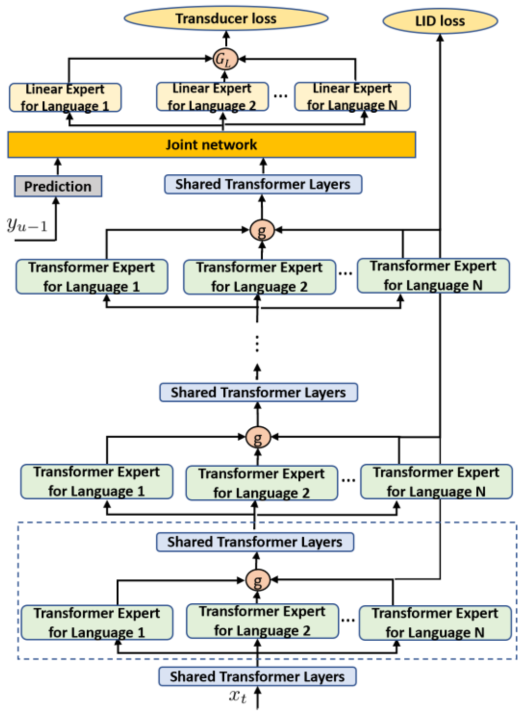

Figure 2: Architecture of multilingual T-T models with shared transformer layers at the bottom, multilingual transformer blocks (one block is defined as in dotted lines) including gated language transformer experts and shared layers, and linear experts for joint network

where h_{joint} is the output of joint network and $G_L = \{g_{L_1}, g_{L_2}, \dots, g_{L_N}\}$ is the gating network that is an affine LID projection referred as a "LID gate" that is controlled by the LID input. In order to get better model performance, we do leverage oracle LID in model training, but during inference, we don't need users to input any LID information, which we elaborate in more detail in section 3.4. In addition, a layer normalization is applied on h'_{joint} to stabilize the training.

3.4 Curriculum training strategy for LID input in training

Even though the LID loss is applied to enforce the transformer experts to learn the corresponding speech information for its own language, the network can still be confused by the languages with the similar word pronunciations and the same writing letters, especially at the early stage of training which could lead to the model performance degradation. In [29], a seed model with explicitly leveraging LID information is pre-trained to relieve this language confusion issue. In our method, a single LID is passed to the LID gate to set its corresponding values to 1 as mentioned in section 3.3 at the early stage of model training to guide the transformer experts to learn their own languages. When training is going on, we also pass multilingual LIDs from all languages to the LID gate while still keeping a portion of passing one single LID to have the gradual transition from one LID training to multiple LID training, which is called as the curriculum training strategy for LID input. At the final stage of model training, only multiple LIDs for all languages are passed to the LID gate. Let's take the bilingual model as an example. At the beginning of training, we only pass LID vectors [0, 1] or [1, 0] to the LID gate for different input languages, and then in the middle stage of training, we pass 1hot vectors [0, 1], [1,0], or 2-hot vector [1,1] to the model training with a probability of p for 1 hot vectors, and 1- p for 2-hot vector. p decreases when training goes on. At the final stage of training, p reduces to 0, and we only pass 2-hot vector [1,1] in model training. Then the Equation (2) is further improved as

$$O_l = W_o(\tanh(\textstyle\sum_{i=0}^{N} W_{l_i} h_{l_i} g_{L_i})) \qquad (9)$$

where l is the layer number of multilingual transformer block. Only one g_{L_i} corresponding to the language i in the language gate G_L is set to 1 at the beginning of training, and all values in G_L are set to 1 for the final stage of training (Note: we omit to draw G_L for the transformer experts in Figure 2 to make the figure less complicated). During inference, the multi-hot LID vector with all its element value 1 is passed to the LID gate by the system for multilingual speech recognition and there is no need for users to input any specific LID information. This training strategy also applies to linear experts of joint network described in Equation (8) in model training.

4. EXPERIMENTAL SETUPS

4.1 Language and data

We develop our multilingual T-T models to support up to five languages which are English (EN), Spanish (ES), German (DE), Italian (IT), and French (FR). For all these languages, both training and test data are transcribed and anonymized with personally identifiable information removed. Test data includes both in-domain data sampled from the same distribution as training, and also out-of-domain data that is different from training. The training and test data amount per language is summarized in Table 1.

4.2 Model structures and training configurations

In our baseline T-T models, 18 basic transformer modules with 320 hidden nodes, 8 attention heads, and 2048 feedforward nodes are used as the encoder; 2 LSTM layers with 1024-dimensional embedding and hidden layer are used in the prediction network. The basic transformer modules are also applied as the transformer experts in multilingual transformer blocks as shown in Figure 2 without any structure change. 80-dimensional log-Mel filterbank are used with 25 milliseconds (ms) windows and 10ms shift. LID vectors are appended to input features in both model training and inference. Two convolutional layers are applied to get features with 40ms sampling rate. The input acoustic feature sequence is segmented into chunks with a chunk size of 4 and chunks are not overlapped. In addition, we also apply 18 left chunks to leverage history acoustic information. An effective mask strategy to truncate history and allow limited future lookahead information has been designed as in [9]. The learning rate warmup strategy is the same as in [32]. Each training mini-batch consists of utterances from all languages, sampled according to their training data distributions. We train BPE models to generate token lists for each language separately, and then merge token lists together as the multilingual model output. For monolingual, bilingual, trilingual, quadrilingual, and pentalingual models, their output tokens are 4k, 7k, 10k, 12k, and 14k, respectively.

Table 1: Train and test data per language (in hours)

Language	Train	Test
EN	23,035	208
ES	3,770	33
DE	2,893	38
IT	3,345	19
FR	3,176	33
Total	36,219	331

Table 2: WERs (%) and parameter numbers (M) for English and Spanish bilingual models

Model	Params	EN	ES	Avg
Monolingual	78*2	13.2	16.2	15.1
B1 Oracle LID	80	12.8	14.9	13.9
B2 baseline without LID	80	14.9	17.1	16.0
B3 fully seperated encoder	133	13.6	16.0	14.8
B4 6 transformer blocks	100	13.1	15.5	14.3
B5 3 transformer blocks	90	13.2	15.8	14.5
B6 + joint linear expert	90.5	13.1	15.6	14.4
B7 + CT for LID input	90.5	13.0	14.9	14.0

5. RESULTS

5.1 English and Spanish bilingual model

We start investigating our methods proposed in Section 3 from English and Spanish bilingual models. We train the baseline English and Spanish bilingual model by simply pooling all data from both languages without feeding any LID information to the model as B2 with parameter size of 80M in Table 2. We also train monolingual models as another baseline. In addition, model parameter information is also provided for different model structures. From Table 2, we can observe that B2 gets an average WER of 16.0% that is worse than the average WER of 15.1% from monolingual models. In addition, we also train the bilingual model with the oracle LID as input to the model during both training and inference to get the upper bound of bilingual model performance as B1 which has 13.1% relative WER reduction over B2. Oracle LID is used as 1-hot vector that is appended to input features as in [16]. We then train the bilingual model with fully separated encoders as shown in Figure 1 as B3. In order to avoid the model divergence, one minor change is that the gating mechanism is not based on the top encoders' output but on the last 3rd layer embeddings from both encoders and the last two layers are used as shared layers. This model obtains the WER of 14.8% that is 7.5% relative WER reduction than B2. However, its model size for encoder is almost doubled and the whole parameter number increases to 133M which makes it not easy to extend to more languages. As proposed in Section 3.2, B4 is trained with 6 multilingual transformer blocks (defined in dotted lines of Figure 2) to avoid the explosion of model size. There are two shared layers in each multilingual transformer block and no shared layers applied at the bottom layer before the multilingual transformer blocks. B4 not only has the more compact structure with 100M model parameters, but also encourages the speech and language information sharing among different languages. B4 achieves 10.6% and 3.4% relative WER reductions over B2 and B3, respectively. In order to further reduce the model size, we change the number of multilingual transformer blocks from 6 to 3 and add 9 shared transformer layers before the multilingual transformer blocks to train model B5 that is only 1.4% relative WER worse but with 10M less model parameters than B4. We also try adding transformer blocks to different locations and model B5's structure to add them near model output is the most effective. Based on B5, we further train B6 model with joint linear experts as proposed in Section 3.3 and get a slightly improved model by reducing the average WER from 14.5% to 14.4% with only 0.5M model parameter increase while getting a much stabler model training recipe, especially when more languages are involved in the multilingual model building. We finally applied the curriculum strategy (CT) for LID input to guide the language dependent transformer and linear experts to learn their own languages and get model B7 that is 12.5% and 7.3% relative WER reductions over B2 baseline and monolingual models, respectively. It can even get the similar average result as the upper model of B1 (14.0% vs. 13.9%). In addition, we also increase the number of parameters for B2 to the same parameter number as B7, and the small parameter increase does not get a significant WER change.

We also measure the model performance on a Spanish/English code-switching test set in which English words are included for entity names based on model B2 and B7. Results in Table 3 show that B7 can achieve 9.3% relative WER reduction over B2 baseline that further verifies the effectiveness of our methods for multilingual ASR.

5.2 Extension to more languages

Based on the success of developing English and Spanish bilingual models, we extend our methods to build

Table 4: WERs (%) and parameter numbers (M) for trilingual, quadrilingual, and pentalingual models

Languages	Params	EN	ES	DE	IT	FR	Avg
Monolingual	78*n (n=3,4,5)	13.2	16.2	15.7	13.2	16.5	15.1
T1 Trilingual Oracle LID	82	12.8	14.5	15.5	-	-	14.3
T2 Trilingual without LID	82	14.9	17.2	16.0	-	-	16.0
T3 Gated Expert Trilingual	100.5	12.9	14.6	15.4	-	-	14.3
Q1 Quadrilingual Oracle LID	84	12.9	14.6	15.2	12.0	-	13.7
Q2 Quadrilingual without LID	84	15.0	17.6	16.2	14.7	-	15.9
Q3 Gated Expert Quadrilingual	110.5	13.0	14.8	15.4	12.2	-	13.9
P1 Pentalingual Oracle LID	86	12.9	14.5	15.3	12.0	15.2	14.0
P2 Pentalingual without LID	86	15.2	18.2	16.5	15.6	16.6	16.4
P3 Gated Expert Pentalingual	120.5	13.2	14.8	15.5	12.1	15.5	14.2

Table 3: Spanish/English code-switching test set results

Model	WER
B2	28.1%
B7	25.5%

multilingual models with more languages such as German, Italian, and French. The model structure is similar to bilingual model B7 but with the addition of the corresponding transformer and linear experts for each new language. More specifically, for each added language, 3 more language-dependent experts are applied in the transformer blocks of model B7 and the number of model parameter increases 10M as shown in Table 4. We still provide monolingual models as a reference to measure the performance of the multilingual models. In addition, we only provide the baseline model results without leveraging any LID information, and also upper bound results from the models trained and inferred with oracle LID to compare with our proposed models as shown in Table 4. When adding more languages, the baseline models' parameter size also slightly increases due to more token labels in the output layer. When adding more languages, our models with gated experts can always get significant improvement as relative 10.6%, 12.6% and 13.4% average WER reductions over the corresponding trilingual, quadrilingual, and pentalingual baseline models without LID, respectively. Compared to the upper bound models with oracle LID, our models can achieve the similar WERs as 14.3% vs 14.3%, 13.7% vs. 13.9%, and 14.0% vs. 14.2% for trilingual, quadrilingual, and pentalingual models, respectively. If we look at individual languages, when adding more languages, the WERs for Spanish without LID are becoming worse and worse as 17.2%, 17.6% and 18.2% for trilingual, quadrilingual, and pentalingual models. The similar patten also applies for German and Italian. However, for our models, adding more languages in multilingual models almost does not make the model performance degrade, which shares the consistent views as the model trained and inferred with the oracle LID information as input. Finally, our gated expert based trilingual, quadrilingual, and pentalingual models can always obtain similar or better WERs over monolingual models.

6. CONCLUSIONS

In this paper, we propose to use gated language experts and auxiliary LID loss to improve the multilingual T-T model performance without any LID input from users during model inference. We construct multilingual transformer blocks including the gated transformer experts and shared layers in encoders to make the model share common speech and acoustic information through shared layers while transformer experts can learn language-dependent knowledge. We also apply linear experts to joint network output to better regularize the joint speech acoustic and token label information and greatly improve the stability of model training. To guide the language dependent experts to learn their corresponding languages even better, we also propose a curriculum training strategy for LID input. On English and Spanish bilingual models, we achieve 12.5% and 7.3% relative WER reductions on average over the bilingual model baseline without leveraging LID information and monolingual models. Our method can even achieve very similar model performance as the model trained and inferred with oracle LID. When extending our method up to five languages, we obtain similar patterns as we get from bilingual models.

7. REFERENCES

[1] J. Li, “Recent Advances in End-to-End Automatic Speech Recognition,” APSIPA Transactions on Signal and Information Processing, vol. 11, no. 1, 2022.

[2] S. Watanabe , et al. "Hybrid CTC/attention architecture for end-to-end speech recognition." IEEE Journal of Selected Topics in Signal Processing 11.8 (2017): 1240-1253.

[3] 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, pp. 4960–4964.

[4] Alex Graves, “Sequence Transduction with Recurrent Neural Networks,” In Proc. of the 29th ICML, 2012.

[5] Y. He, T.N. Sainath, R. Prabhavalkar, et al. “Streaming End-to-End Speech Recognition for Mobile Devices,” in Proc. ICASSP, 2019.

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

[7] Q. Zhang, H. Lu, H. Sak, A. Tripathi, E. McDermott, S. Koo, and S. Kumar, “Transformer Transducer: A Streamable Speech Recognition Model with Transformer Encoders and Rnn-t Loss,” in Proc. ICASSP 2020.

[8] C.-F. Yeh, J. Mahadeokar, K. Kalgaonkar, Y. Wang, D. Le, M. Jain, K. Schubert, C. Fuegen, and M. L. Seltzer, “Transformer-Transducer: End-to-End Speech Recognition with Self-Attention,” arxiv.org/abs/1910.12977, 2019.

[9] X. Chen, Y. Wu, Z. Wang, S. Liu, and J. Li, “Developing Real-Time Streaming Transformer Transducer for Speech Recognition on Large-scale Dataset,” in Proc. ICASSP 2021.

[10] “Multilingual People,” http://ilanguages.org/bilingual.php, Accessed: 2021-10-16.

[11] Watanabe, S., Hori, T., and Hershey, J. R. “Language Independent End-to-End Architecture for Joint Language Identification and Speech Recognition,” in Proc. ASRU, pp. 265-271, 2017.

[12] S. Toshniwal, T. Sainath, R. Weiss, B. Li, P. Moreno, E. Weinstein, and K. Rao, “Multilingual Speech Recognition with a Single End-to-End Model,” in Proc. ICASSP, 2018.

[13] B. Li, Y. Zhang, T. Sainath, Y. Wu, and W. Chan, “Bytes Are All You Need: End-to-End Multilingual Speech Recognition and Synthesis with Bytes,” in Proc. ICASSP, 2018.

[14] V. Pratap, A. Sriram, P. Tomasello, A. Hannun, V. Liptchinsky, Synnaeve, and R. Collobert, “Massively Multilingual ASR: 50 Languages, 1 Model, 1 Billion Parameters,” in Proc. INTERSPEECH, 2020.

[15] O. Adams, M. Wiesner, S. Watanabe, and D. Yarowsky, “Massively Multilingual Adversarial Speech Recognition,” arXiv:1904.02210, 2019.

[16] A. Kannan, A. Datta, T. N. Sainath, E. Weinstein, B. Ramabhadran, Y. Wu, A. Bapna, Z. Chen, and S. Lee, “Large-Scale Multilingual Speech Recognition with a Streaming End-to-End Model,” in Proc. INTERSPEECH, 2019.

[17] Y. Zhu, P. Haghani, A. Tripathi, B. Ramabhadran, B. Farris, H. Xu, H. Lu, H. Sak, I. Leal, N. Gaur, P. J. Moreno, Q. Zhang, “Multilingual Speech Recognition with Self-Attention Structured Parameterization,” in Proc. INTERSPEECH, 2020.

[18] A. Waters, N. Gaur, P. Haghani, P. Moreno, Z. Qu, “Leveraging Language ID in Multilingual End-to-End Speech Recognition,” in Proc. ASRU, 2019.

[19] S. Punjabi, H. Arsikere, Z. Raeesy, C. Chandak, N. Bhave, A. Bansal, M. Muller, S. Murillo, A. Rastrow, S. Garimella, et al., “Streaming End-to-End Bilingual ASR Systems with Joint Language Identification,” arXiv preprint arXiv:2007.03900, 2020.

[20] L. Zhou, J. Li, E. Sun, and S. Liu, “A Configurable Multilingual Model is All You Need to Recognize All Languages,” in Proc. ICASSP 2022.

[21] C. Zhang, B. Li, T.N. Sainath, T. Strohman, S. Mavandadi, S. Chang, P. Haghani, “Streaming End-to-End Multilingual Speech Recognition with Joint Language Identification,” in Proc. INTERSPEECH 2022.

[22] E. Sun, J. Li, Z. Meng, Y. Wu, J. Xue, S. Liu, and Y. Gong, “Improving Multilingual Transformer Transducer Models by Reducing Language Confusions,” in Proc. INTERSPEECH, 2021.

[23] C. Zhang, B. Li, T.N. Sainath, T. Strohman, S. Mavandadi, S. Chang, P. Haghani, “Streaming End-to-End Multilingual Speech Recognition with Joint Language Identification,” in Proc. INTERSPEECH, 2022.

[24] B. Houston and K. Kirchhoff, “Exploration of Language-specific Self-attention Parameters for Multilingual End-to-End Speech Recognition,” in Proc. SLT, 2022.

[25] C. Zhang, B. Li, T.N. Sainath, T. Strohman, and S. Chang, “UML: A Universal Monolingual Output Layer for Multilingual ASR,” arxiv.org/abs/2302.11186, 2023.

[26] C. H. Yang, B. Li, Y. Zhang, N. Chen, R. Prabhavalkar, and T.N. Sainath, and T. Strohman, ” From, English to More Languages: Parameter-efficient Model Reprogramming for Cross-lingual Speech Recognition,” arxiv.org/abs/2301.07851, 2023.

[27] S. Papi, E. Trentin, R. Gretter, M. Matassoni, and D. Falavigna, “Mixtures of Deep Neural Experts for Automated Speech Scoring,” in Proc. INTERSPEECH, 2020, pp. 3845–3849.

[28] A. Jain, V. P. Singh, and S. P. Rath, “A Multi-Accent Acoustic Model Using Mixture of Experts for Speech Recognition,” in Proc. INTERSPEECH, 2019, pp. 779–783.

[29] Y. Lu, M. Huang, H. Li, J., Y. Qian, “Bi-encoder Transformer

Network for Mandarin-English Code-Switching Speech Recognition using Mixture of Experts," in Proc. INTERSPEECH, 2020.

[30] S. Dalmia, Y. Liu, S. Ronanki, and K. Kirchhoff, "Transformer-Transducers for Code-Switched Speech Recognition," in Proc. ICASSP, 2021.

[31] N. Gaur, B. Farris, P. Haghani, I Leal, P. J. Moreno, et al., "Mixture of Informed Experts for Multilingual Speech Recognition," in Proc. ICASSP, 2021.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All You Need," in Advances in neural information processing systems, pages 5998–6008, 2017.