INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Adapt-and-Adjust: Overcoming the Long-Tail Problem of Multilingual Speech Recognition

Genta Indra Winata[1⋆], *Guangsen Wang*[2⋆], *Caiming Xiong*[3], *Steven Hoi*[2]

[1]The Hong Kong University of Science and Technology, Hong Kong SAR, China
[2]Salesforce Research, Singapore
[3]Salesforce Research, USA

giwinata@connect.ust.hk, guangsen.wang@salesforce.com

Abstract

One crucial challenge of real-world multilingual speech recognition is the long-tailed distribution problem, where some resource-rich languages like English have abundant training data, but a long tail of low-resource languages have varying amounts of limited training data. To overcome the long-tail problem, in this paper, we propose Adapt-and-Adjust (A2), a transformer-based multi-task learning framework for end-to-end multilingual speech recognition. The A2 framework overcomes the long-tail problem via three techniques: (1) exploiting a pretrained multilingual language model to improve the performance of low-resource languages; (2) proposing dual adapters consisting of both language-specific and language-agnostic adaptation with minimal additional parameters; and (3) overcoming the class imbalance, either by imposing class priors in the loss during training or adjusting the logits of the softmax output during inference. Extensive experiments on the CommonVoice corpus show that A2 significantly outperforms conventional approaches.

Index Terms: speech recognition, multilingual, transfer learning, long-tail, logits adjustment

1. Introduction

Deploying a single Automatic Speech Recognition (ASR) model to recognize multiple languages is highly desired but very challenging for real-world multilingual ASR scenarios due to the well-known long-tailed distribution challenge, namely, that some resource-rich languages like English have abundant training data, while the majority low-resource languages have varying amounts of training data. The recent popular end-to-end (E2E) monolingual ASR architecture [1, 2, 3] with multi-task training is promising to be used as a single model for recognizing multiple languages. However, the imbalanced or long-tailed data distribution problem makes building an end-to-end [1, 2, 3] multilingual ASR notoriously challenging. These challenges stem from two aspects. First, very limited audio samples are available for low-resource languages, such as Kyrgyz, Swedish, while simultaneously, vast amounts of data exist from high-resource languages, such as English. Second, graphemes or subwords of a language follow a long-tailed distribution in ASR since some units appear significantly more frequently. Furthermore, a multilingual system may include languages with writing scripts other than the Latin alphabet, such as Chinese or Cyrillic, further worsening the skewness.

While a standard end-to-end multilingual training approach can improve overall performance compared with monolingual end-to-end approaches, it does not address the long-tail problem explicitly. A common practice is "balanced sampling" [4, 5] to sample same number of training examples for each language within a batch. However, such an ad-hoc approach often hurts the performance of resource-rich languages. This is not desirable considering the resource-rich languages also have the most users. Furthermore, another critical issue with balanced sampling is that it does not address the long-tailed distribution caused by the multilingual training labels. Another risk of balanced sampling is over-fitting for the languages with limited data due to up-sampling. Therefore, balanced sampling only leads to marginal overall improvement over the baseline. In this paper, we formulate these challenges as a twofold "long-tail problem": 1) the long-tailed class distribution arising from the skewed multilingual label distribution 2) the skewed distribution of the multilingual training data, *i.e.,* robust modeling of languages with limited training data (a.k.a the tail languages).

To this end, we propose the Adapt-and-Adjust (A2) framework for multilingual speech recognition using a speech transformer to address the twofold long-tail problem. Firstly, for better language modeling, a distilled mBERT [6] is converted to an autoregressive transformer decoder to jointly explore the multilingual acoustic and text space to improve the performance of low-resource languages. Secondly, to adapt the multilingual network to specific languages with minimal additional parameters, both language-specific and language-agnostic adapters are used to augment each encoder and decoder layer. Lastly, to increase the relative margin between logits of rare versus dominant language labels, we perform class imbalance adjustments during multilingual model training or inference by revisiting the classic idea of logit adjustment [7]. Class imbalance adjustment [8, 9, 10] is applied by adjusting the logits of the softmax input with the class priors. We conduct experiments and establish a benchmark from the CommonVoice corpus with a realistic long-tailed distribution of different languages. The extensive experiments show that A2 significantly outperforms conventional approaches for end-to-end multilingual ASR.

2. Adapt-and-Adjust Framework

2.1. Overview

Figure 1 gives an overview of the proposed A2 framework for end-to-end multilingual ASR. A2 is built on a transformer-based sequence-to-sequence model with three key novel contributions: (1) an mBERT-based decoder, (2) language adapters, and (3) class-imbalance adjustments. Firstly, the vanilla transformer decoder is replaced with mBERT for better language modeling, particularly for low-resource languages. Secondly, the common and language-specific adapters are added to each encoder and decoder layer to learn both the shared and

[⋆]Equal contributions.

http://dx.doi.org/10.21437/Interspeech.2021-1390

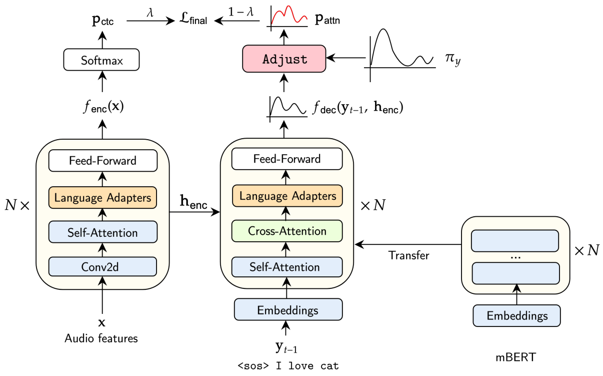

Figure 1: *Overview of the Adapt-and-Adjust framework. The layer norm is omitted to save space.* $\mathbf{p}_{ctc}$ *is the CTC output,* $\mathbf{p}_{attn}$ *is the decoder output,* $\mathbf{y}_{t-1}$ *is the previous token.*

language-specific information for better acoustic modeling. Finally, we perform class imbalance adjustments during training or inference, where the logits are adjusted with the class priors estimated from the training data.

2.2. Hybrid CTC-Attention Speech Transformer

A sequence-to-sequence speech transformer model [11, 12, 13] based on the hybrid CTC-Attention network is used for acoustic modeling to predict sentence piece tokens. We adapted the implementation in ESPNET [14] for A2. Multi-task loss $\mathcal{L}_{\text{MTL}}$ is computed as an interpolation of the attention loss $\mathcal{L}_{\text{ATTN}}$ and the CTC [15] loss with a hyper-parameter λ ($0 \le \lambda \le 1$)

$$\mathcal{L}_{\text{MTL}} = \lambda \log \mathbf{p}_{\text{ctc}}(\mathbf{y}|\mathbf{h}_{\text{enc}}) + (1-\lambda)\mathcal{L}_{\text{ATTN}}, \tag{1}$$

$\mathbf{h}_{\text{enc}}$ is the output generated by the encoder. Kullback-Leibler divergence (KL) loss [16] is used for $\mathcal{L}_{\text{ATTN}}$. Beam search is used to predict the sentence pieces. The decoding score is computed as a weighted sum of both the CTC and attention probabilities using β to balance them [17]:

$$\hat{\mathbf{y}} = \underset{\mathbf{y}\in\mathcal{Y}^*}{\arg\max}\{\beta p_{\text{ctc}}(\mathbf{y}|\mathbf{h}_{\text{enc}}) + (1-\beta)p_{\text{attn}}(\mathbf{y}|\mathbf{h}_{\text{enc}}, \mathbf{y}')\}, \tag{2}$$

where $\mathbf{y}'$ is the decoded sequence so far.

2.3. Multilingual BERT as Autoregressive Decoder

For better language modeling, especially for low-resource languages, mBERT is used as the transformer decoder. Since mBERT is pretrained on text data, it is essential to augment a cross-attention layer to the encoder output for each mBERT layer. The cross-attention and its self-attention layers are learned to "align" the acoustic and text spaces for the speech recognition. This is because the text space may diverge significantly from the acoustic space of the encoder output. We copy the embeddings and self-attention parameters of mBERT into the decoder layers. Let t denote the current decoding step. The autoregressive decoder takes the current input token y_t to predict the next token y_{t+1}. The mBERT embedding layer converts the input token to a vector representation. Subsequently, the cross-attention layer takes the encoder output $\mathbf{h}_{\text{enc}}$ and computes the attention output. To the best of our knowledge, this work is the first to adapt a pretrained multilingual language model for multilingual ASR.

2.4. Dual Language Adapters

Similar to [4], lightweight residual language adapters are used for better acoustic modeling with minimal language-specific parameters to increase the model robustness to languages with limited resources. As shown in Figure 2, in addition to the language-specific adapter for capturing the language-intrinsic knowledge, a common adapter is also trained to learn language-agnostic information in the multilingual data; we call these **Dual-Adapters**. The language-specific and common adapters are denoted as A_{lang} and A_{com}, respectively. Each adapter of layer l consists of a down-projection layer $\mathbf{W}_d^l$, followed by a ReLU activation function, and an up-projection layer $\mathbf{W}_u^l$. The adapters take $\mathbf{h}^l$ as the input, where $\mathbf{h}^l$ is the self attention output of layer l. We compute the output of $\text{A}(\mathbf{h}^l)$ as follows for both the language-specific and common adapters:

$$\text{A}(\mathbf{h}^l) = \mathbf{W}_u^l(\text{ReLU}(\mathbf{W}_d^l(\text{LN}(\mathbf{h}^l)))) + \mathbf{h}^l, \tag{3}$$

where LN is the layer norm. The final adapter output is computed as $\mathbf{o}^l = \mathbf{o}^l_{\text{lang}} + \mathbf{o}^l_{\text{com}}$. $\mathbf{o}^l$ is then used as the input to the next encoder or decoder layer.

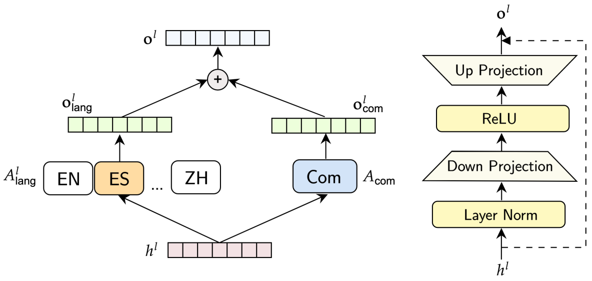

Figure 2: *Dual-Adapters.*

2.5. Sentence Piece Class Imbalance Adjustments

The sentence piece class imbalance problem is addressed by incorporating the class priors during training or inference via logit adjustments. Derived from a Bayesian point of view in [10] for computer vision tasks, the softmax classifier with adjusted logits as input minimizes the balanced error across all classes. A natural adjustment is to scale the logits $f_y(x)$ by the inverse of the corresponding class prior π_y. In log domain, the adjustment can be performed as follows:

$$f_y^{\text{adj}}(x) = f_y(x) - \tau \cdot \log \pi_y, \tag{4}$$

where $\tau > 0$ is a hyper-parameter. The adjustment can be viewed as applying a class-dependent offset to re-weight each logit according to its class prior.

Class priors: The class priors are the natural frequencies of the sentence piece tokens estimated from the multilingual training data. To form a valid prior distribution, smoothing is applied to the raw counts according to Equation 5 for zero occurrence tokens:

$$\pi_y = \begin{cases} \frac{C_i}{C} - \frac{1}{(N-n_0)\times C}, & c_i > 0 \\ \frac{1}{n_0 \times C}, & \text{otherwise}, \end{cases} \tag{5}$$

where C is the total number of counts for all labels, n_0 is the number of labels with zero occurrences, N is the number of classes and c_i is the raw count of class i.

Training phase adjustments: To incorporate the priors during training, the logits $f_{y_t}^{\text{dec}}$ of the last decoder layer are adjusted before softmax as the following:

$$f_{y_t}^{\text{dec}} = w_y^T \cdot \text{Decoder}(\mathbf{h}_{\text{enc}}, \text{Emb}(y_{t-1})) \tag{6}$$

$$f_{y_t}^{\text{adj}} = f_{y_t}^{\text{dec}} - \tau \cdot \log \pi_{y_t}, \tag{7}$$

$$p_{y_t}^{\text{adj}} = \frac{\exp(f_{y_t}^{\text{adj}})}{\sum_{y'_t \in [N]} \exp(f_{y'_t}^{\text{adj}})}. \tag{8}$$

The adjusted softmax output vector $\mathbf{p}_y^{\text{adj}}$ of the sequence is used to compute the KL loss and perform the backward propagation to update the model. y_{t-1} is the previous label available only during training. During later training iterations, instead of using the ground truth label y_{t-1} for computing the logits, y'_{t-1} is chosen from the maximum prediction output of the current model to simulate the inference:

$$y'_{t-1} = \underset{y}{\text{argmax}}\, \mathbf{p}_{y_{t-1}}^{\text{adj}}. \tag{9}$$

If the scheduled sampling is used, the adjusted logits at step t will have influence over all of the following tokens in the current sequence. This is a crucial difference from the image classification task in [10]. If τ is set to be 1, the training phase logit adjustment becomes similar to the widely used label smoothing technique [18]. However, in conventional label smoothing, the prior π_y is usually a uniform distribution that is independent of the data. The logit adjustment applies a class-specific "smoothing" based on the class prior, and has been shown to be superior to the standard label smoothing baseline.

Inference phase adjustments: Alternatively, the class priors can be applied during inference via logit adjustments.

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y} \in \mathcal{Y}^*} \{\beta \mathbf{p}_{\text{ctc}}(\mathbf{y}|\mathbf{h}_{\text{enc}}) + (1-\beta)\mathbf{p}_{\mathbf{y}}^{\text{adj}}\}. \tag{10}$$

During beam search, the attention decoding scores $p_{\mathbf{y}}^{\text{adj}}$ are computed in the same way as the scheduled sampling from the adjusted logits.

3. Experiments

3.1. Experimental Setup

Dataset: CommonVoice [19] is a multilingual corpus collected by Mozilla. Similar to [20], we use 11 languages: *English (en), Spanish (es), French (fr), Italian (it), Kyrgyz (ky), Dutch (nl), Russian (ru), Swedish (sv), Turkish (tr), Tatar (tt),* and *Chinese (zh)*. The transcriptions are tokenized using the SentencePiece model with the unigram algorithm resulting in 5237 multilingual tokens. We then add special tokens, such as <unk>, <sos>, <eos>, and a blank token for the CTC objective.

Network configurations: We use six transformer encoder layers with a hidden size of 2048 units and eight attention heads, each with an attention dimension of 256. For the decoder, distilmBERT [21] and XLM-R [20] are used. We train our model with a batch size of 32 and accumulate the gradient in two steps using a GPU NVIDIA V100 16GB. The models are trained with the Adam optimizer with a warm-up step of 25000. For balanced sampling, we take six samples for each language and construct a balanced batch by accumulating the gradient 11 times.

Training and Evaluation: We evaluate our model using beam-search with a beam width of 10, $\lambda = 0.3$, and $\beta = 0.5$. The hyper-parameter τ is set to 0.3 for both the training and inference phase class imbalance adjustments. The multilingual models are trained with 150K iterations. We compute the average over the last ten checkpoints as the decoding model. For the monolingual setting, we stop after 100 epochs of training. Models are evaluated using the character error rate (CER) to simplify the evaluation for all languages.

Baselines: We consider the following baselines: **Monolingual**: we train monolingual models; **SMT** (Standard Multilingual Training), we randomly sample the batch from the data distribution; **BS** (Balanced Sampling), we sample the same number of utterances for each language in a batch so that they have roughly equal contributions to the training; **LAN-Specific Adapters**: language-specific adapters by [4]; and **LID**: (language ID) conditioning with one-hot language vectors proposed by [22].

3.2. Experimental Results with Different A2 Components

In Table 1, we present the test results with different model configurations. Compared to the monolingual models, the SMT models improve the performance of the low-resource languages significantly. We conjecture that this may be because the multilingual models can capture common sub-phonetic articulatory features [23, 24, 25, 26] that are shared by different languages and are beneficial for low-resource languages recognition.

Balanced Sampling: We observe the same trend as in [4]: compared to the SMT, the tail language performance is significantly boosted. However, the performance of the head languages suffers due to fewer occurrences during training. The model is over-fitted to the tail languages due to up-sampling, for example, the CERs on the training set of "ky" and "sv" are significantly lower than the test data (3.4% and 4.2% training vs. 13.4% and 22.8% test).

Language Adapters: We then compare the language adaptation techniques including the LAN-Specific Adapters [4], the one-hot language vector [22], and the Dual-Adapters. Note that all adapters are based on BS + mBERT, which has better performance than the BS-only model. Adding the language-specific adapters [4] without common adapters significantly outperforms the BS baseline, with a 0.9% absolute performance gain. The LID performs similarly to the language-specific adapters. The ablation study of our dual-adapters is given in Table 2. The Dual-Adapters outperform the language-specific adapters significantly, by a 0.5% absolute CER reduction, indicating knowledge transfer with the common adapter is effective. To reduce the parameter size and encourage fine-grained language transfer, we apply parameter sharing by dividing languages into groups and languages in the same group share the same dual-adapters trained with all language members. According to the written scripts, we divide the 11 languages into language groups, *e.g.,* Latin, Chinese characters and Cyrillic scripts. They can also be grouped according to language families, *e.g.,* Romance, Chinese, Turkic, Germanic, focusing more on the similarities in lexica, grammars, and pronunciations, which are usually subsumed under the end-to-end multilingual architectures. Results in Table 2 show that the individual adapters are quite robust. In addition, grouping by language families is better than grouping by written scripts since they are more consistent with the acoustic space adaptation.

Sentence Piece Class Imbalanced Adjustment: Further performance boost can be achieved by the training or inference phase adjustments, with 0.5% CER reduction over the best language adapters. The gains are mostly due to the improved performance of the head languages, although tail languages also benefit from the logit adjustments. More importantly, the gap

Table 1: *Test results in terms of CER (%) on the CommonVoice dataset.*

	high-resource			intermediate					low-resource			
Model	**en**	**fr**	**es**	**it**	**ru**	**zh**	**tt**	**nl**	**tr**	**ky**	**sv**	**avg**
Training hours	80	50	40	20	15	15	15	15	10	9	4	
Testing hours	10	10	10	8.3	7.3	4.3	2.5	2.5	1.3	1.1	0.5	
Monolingual	22.6	20.1	17.3	20.7	23.9	37.6	20.7	38.1	30.3	31.6	57.7	29.1
BS	25.2	20.3	14.5	12.7	13.2	32.6	11.5	18.0	12.9	13.4	22.8	17.9
SMT	**20.1**	**17.4**	**13.0**	12.5	13.7	34.1	12.2	18.7	13.9	14.4	26.4	17.9
LAN-Specific Adapters [4]	24.2	19.4	13.9	12.3	**11.8**	32.0	**10.7**	16.9	**11.8**	12.7	**21.7**	**17.0**
LID [22]	25.7	19.2	13.7	**12.0**	12.0	**31.6**	10.8	**16.4**	12.0	**12.5**	21.8	17.1
BS + mBERT	25.0	20.0	15.4	12.6	13.2	32.9	11.1	17.3	12.6	12.8	22.8	17.8
BS + Dual-Adapters	23.5	18.9	13.5	12.1	12.3	31.0	10.9	16.5	12.0	12.9	21.6	16.8
BS + Dual-Adapters + mBERT	23.4	18.6	13.4	11.8	**11.7**	**30.8**	**10.8**	**16.2**	**11.6**	**12.4**	**21.5**	**16.5**
SMT + mBERT	20.4	17.9	13.4	13.0	14.0	34.8	12.4	18.9	14.0	14.3	26.2	18.1
SMT + Adjust-Train	20.2	16.9	12.5	11.9	13.1	32.9	11.3	18.5	13.5	13.9	25.3	17.3
SMT + Adjust-Train + mBERT	**19.4**	**16.5**	**12.1**	**11.7**	12.2	31.1	11.0	17.5	12.7	13.2	24.6	**16.5**
A2 (Adjust-Inference) w/ mBERT	22.6	17.9	12.7	**11.2**	11.2	30.1	10.2	**15.8**	11.4	12.2	**21.1**	**16.0**
A2 (Adjust-Train) w/ mBERT	**22.0**	17.7	**12.5**	11.3	**11.1**	30.4	**10.0**	15.9	**11.3**	**12.1**	21.3	**16.0**
A2 (Adjust-Train) w/ XLM-R	22.1	**17.6**	**12.5**	11.4	11.4	**29.6**	10.3	15.9	11.5	**12.1**	21.4	**16.0**

Table 2: *Ablations on language adapters.*

	#Params	**avg**
LAN-Specific Adapters	84M	17.0
Dual-Adapters (individual)	84M	**16.5**
Dual-Adapters (grouped by written scripts)	78M	16.8
Dual-Adapters (grouped by language families)	78M	**16.5**

between the monolingual and multilingual performance for the head languages is greatly reduced, leading to a better "balanced error" performance. This strongly justifies the importance of class imbalance adjustments under the long-tail setting.

Pretrained Language Models: Table 1 shows that adding mBERT helps the performance, especially with good quality acoustic models like "BS+Dual-Adapters". This may be due to that with better acoustic models like A2, the text space of the vanilla mBERT is better aligned with the acoustic space, leading to improved performance across all languages, especially for low-resource ones. To investigate whether further improvement can be obtained, a larger XLM-R model is used. Although XLM-R has a better multilingual language generation capability than mBERT, it does not translate to the final performance gain for the multilingual ASR task. We believe this is because it becomes more difficult to align the text and acoustic space with the increased language model complexities.

Table 3: *Transcriptions of a traditional Chinese utterance.*

Reference	困難與挑戰是激發我們的原動力	
Pinyin	Kun4 Nan2 Yu3 Tiao3 Zhan4 Shi4 Ji1 Fa1 Wo3 Men2 De0 Yuan2 Dong4 Li4	
Translation	Obstacles and challenges are the sources of power that motivate us.	
Model	**Hypothesis**	**CER**
BS + mBERT	困難與挑戰是資料我們的員動力	21.4
A2 (Adjust-Train)	困難與挑戰是機發我們的員動力	**14.3**
Monolingual	負能一票佔是機發我的能員動力	64.3
SMT	可能與調站是機發我們的員動力	42.9

Transcription Example: As shown in Table 3, not only did the monolingual model miss an important character 們 for distinguishing "my" and "our", the pinyin (Fu4 Neng2 Yi2 Piao4 Zhan4 Shi4 Wo3 De0 Yuan2 Dong4 Li4) is also quite different from the reference. SMT improves the monolingual model by recovering 與 (Yu3) from 一 (Yi2). In addition, the output 調站 (Tiao2 Zhan4) sounds almost the same as the reference 挑戰 (Tiao3 Zhan4), indicating multilingual training helps improve the acoustic modeling over the monolingual training with limited training data. The mBERT model further helps by correcting the grammar from 可能與調站 to 困難與挑戰 ("obstacles and challenges"). However, mBERT also replaces 機發 (Ji1 Fa1) to 資料 (Zi1 Liao4) since 資料 ("documents") is more common than 機發 (not grammatically correct). Lastly, the adapters and logits adjustment successfully correct 資料 to 機發 . The remaining errors including the wrong characters in 機發 and 員動力 can be easily corrected by an external language model during decoding.

4. Related Work

Conventional approaches to addressing the long-tail problem are focused on data sampling [27, 28]. The problem has regained interest and was studied in [10] and the methods compared included weight normalization [29], adaptive margin [30], and equalized loss [31]. Adapters were proposed to learn efficient domain-specific representations [32]. They were adopted for NLP tasks to avoid fine-tuning a new model by training an adapter module [33, 34]. E2E architectures like LAS [35] and the RNN-Transducer [4] have been used for building a multilingual ASR for Indian languages. Language adapters are used to tackle the data imbalance problem [4]. Acoustic vector quantization is also used by [20] on multilingual ASR. A massive multilingual ASR study with more than 50 languages [5].

5. Conclusion

In this paper, we introduce Adapt-and-Adjust (A2), an end-to-end transformer-based framework to address the crucial challenge of long-tail data distribution issues in multilingual speech recognition. A2 consists of three major components, namely, language adapters, class imbalance adjustments, and a pretrained multilingual language model. Extensive experiments on the CommonVoice corpus show that A2 significantly outperforms conventional approaches for end-to-end multilingual speech recognition due to 1) the better acoustic modeling with adapters and class imbalance adjustments; and 2) the better language modeling with pretrained multilingual BERT.

6. References

[1] A. Graves, A. Mohamed, and G. E. Hinton, "Speech recognition with deep recurrent neural networks," *CoRR*, vol. abs/1303.5778, 2013. [Online]. Available: http://arxiv.org/abs/1303.5778

[2] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell," *CoRR*, vol. abs/1508.01211, 2015. [Online]. Available: http://arxiv.org/abs/1508.01211

[3] 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*, 2017, pp. 5998–6008.

[4] 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," *Proc. Interspeech 2019*, pp. 2130–2134, 2019.

[5] V. Pratap, A. Sriram, P. Tomasello, A. Hannun, V. Liptchinsky, G. Synnaeve, and R. Collobert, "Massively multilingual asr: 50 languages, 1 model, 1 billion parameters," 2020.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[7] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Trans. on Knowl. and Data Eng.*, vol. 18, no. 1, p. 63–77, Jan. 2006.

[8] G. Collell, D. Prelec, and K. Patil, "Reviving threshold-moving: a simple plug-in bagging ensemble for binary and multiclass imbalanced data," *arXiv preprint arXiv:1606.08698*, 2016.

[9] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9268–9277.

[10] A. K. Menon, S. Jayasumana, A. S. Rawat, H. Jain, A. Veit, and S. Kumar, "Long-tail learning via logit adjustment," *arXiv preprint arXiv:2007.07314*, 2020.

[11] 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)*, 2018, pp. 5884–5888.

[12] S. Kim, T. Hori, and S. Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," *CoRR*, vol. abs/1609.06773, 2016. [Online]. Available: http://arxiv.org/abs/1609.06773

[13] S. Karita, N. E. Y. Soplin, S. Watanabe, M. Delcroix, A. Ogawa, and T. Nakatani, "Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration," *Proc. Interspeech 2019*, pp. 1408–1412, 2019.

[14] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen *et al.*, "Espnet: End-to-end speech processing toolkit," *arXiv preprint arXiv:1804.00015*, 2018.

[15] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006, pp. 369–376.

[16] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[17] S. Karita, N. E. Y. Soplin, S. Watanabe, M. Delcroix, A. Ogawa, and T. Nakatani, "Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration," in *Proc. Interspeech 2019*, 2019, pp. 1408–1412.

[18] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" in *Advances in Neural Information Processing Systems*, 2019, pp. 4694–4703.

[19] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 4218–4222.

[20] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised cross-lingual representation learning for speech recognition," *arXiv preprint arXiv:2006.13979*, 2020.

[21] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[22] B. Li, T. N. Sainath, K. C. Sim, M. Bacchiani, E. Weinstein, P. Nguyen, Z. Chen, Y. Wu, and K. Rao, "Multi-dialect speech recognition with a single sequence-to-sequence model," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 4749–4753.

[23] K. Kirchhoff, G. A. Fink, and G. Sagerer, "Combining acoustic and articulatory feature information for robust speech recognition," *Speech Communication*, vol. 37, no. 3, pp. 303 – 319, 2002.

[24] F. Metze, "Articulatory features for conversational speech recognition," 2005.

[25] K. Livescu, O. Cetin, M. Hasegawa-Johnson, S. King, C. Bartels, N. Borges, A. Kantor, P. Lal, L. Yung, A. Bezman, S. Dawson-Haggerty, B. Woods, J. Frankel, M. Magimai-Doss, and K. Saenko, "Articulatory feature-based methods for acoustic and audio-visual speech recognition: Summary from the 2006 jhu summer workshop," in *2007 IEEE ICASSP*, vol. 4, 2007, pp. IV–621–IV–624.

[26] G. Wang and K. C. Sim, "Regression-based context-dependent modeling of deep neural networks for speech recognition," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 22, no. 11, p. 1660–1669, Nov. 2014.

[27] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-sided selection," in *ICML*, 1997.

[28] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[29] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, and Y. Kalantidis, "Decoupling representation and classifier for long-tailed recognition," in *ICLR*, 2019.

[30] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *Advances in Neural Information Processing Systems*, 2019, pp. 1567–1578.

[31] J. Tan, C. Wang, B. Li, Q. Li, W. Ouyang, C. Yin, and J. Yan, "Equalization loss for long-tailed object recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 662–11 671.

[32] S.-A. Rebuffi, H. Bilen, and A. Vedaldi, "Learning multiple visual domains with residual adapters," in *Advances in Neural Information Processing Systems*, 2017, pp. 506–516.

[33] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for nlp," in *International Conference on Machine Learning*, 2019, pp. 2790–2799.

[34] J. Pfeiffer, I. Vulić, I. Gurevych, and S. Ruder, "Mad-x: An adapter-based framework for multi-task cross-lingual transfer," *arXiv preprint arXiv:2005.00052*, 2020.

[35] S. Toshniwal, T. N. Sainath, R. J. Weiss, B. Li, P. Moreno, E. Weinstein, and K. Rao, "Multilingual speech recognition with a single end-to-end model," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr 2018.