Interspeech 2022
18-22 September 2022, Incheon, Korea

ScoutWav: Two-Step Fine-Tuning on Self-Supervised Automatic Speech Recognition for Low-Resource Environments

Kavan Fatehi[1], *Mercedes Torres Torres*[2], *Ayse Kucukyilmaz*[1]

[1]School of Computer Science, University of Nottingham
[2]B-hive Innovations

kavan.fatehi@nottingham.ac.uk, mtorrestorres@b-hiveinnovations.co.uk,
ayse.kucukyilmaz@nottingham.ac.uk

Abstract

Recent improvements in Automatic Speech Recognition (ASR) systems obtain extraordinary results. However, there are specific domains where training data can be either limited or not representative enough, which are known as Low-Resource Environments (LRE). In this paper, we present ScoutWav, a network that integrates context-based word boundaries with self-supervised learning, wav2vec 2.0, to present a low-resource ASR model. First, we pre-train a model on High-Resource Environment (HRE) datasets and then fine-tune with the LRE datasets to obtain context-based word boundaries. The resulting word boundaries are used for fine-tuning with a pre-trained and iteratively refined wav2vec 2.0 to learn appropriate representations for the downstream ASR task. Our refinement strategy for wav2vec 2.0 comes determined by using canonical correlation analysis (CCA) to detect which layers need updating. This dynamic refinement allows wav2vec 2.0 to learn more descriptive LRE-based representations. Finally, the learned representations in the two-step fine-tuned wav2vec 2.0 framework are fed back to the Scout Network for the downstream task. We carried out experiments with two different LRE datasets: I-CUBE and UASpeech. Our experiments demonstrate that using the target domain word boundary after pre-training and automatic layer analysis, ScoutWav shows up to 12% relative WER reduction on the LR data.

Index Terms: Automatic Speech Recognition, Self-Supervised Learning, Fine-tuning, Low-resource Environment

1. Introduction

Recently, there have been remarkable improvements in end-to-end (E2E) automatic speech recognition (ASR) systems, which need large amount of labeled speech to perform well, which may be impossible for all applications. A low-resource environment (LRE) is an environment in which training data and labels are insufficient and difficult to collect [1], such as a new language, such as Kyrgyz [2], or for a specific group of speakers with different accents [1]. These highly-accurate performances coupled with such large amount of labeled data proposes a need to use unlabeled data in developing the ASR model for LREs.

Self-supervised learning (SSL) has been proposed to acquire meaningful representations from unlabeled data, with promising results in low and high resource ASR settings [3, 4]. In SSL, a large amount of unlabeled data is used to extract representations, which are used as input to a final model for a downstream task [3]. Recently, wav2vec 2.0 [5] was proposed as a layer-based SSL model based on the Transformer [6]. SSL models can obtain high-quality representations for the ASR systems and achieve better performance by fine-tuning a small amount of annotated in-domain data. However, one-step fine-tuning is unable to adapt the pre-trained model to the downstream task, and there is still a performance gap in this area.

Recently, Wang et al. [7] have proposed a new low-latency end-to-end (E2E) model, called the scout network (SN), which showed state-of-the-art results in ASR systems. They hypothesize that the speech segment which relates to the word is the most valuable contextual information to provide an output token. Therefore, they proposed two different neural components, the SN to detect the word boundary in the speech segments, while the recognition network (RN) detects the sub-word by considering the context from all frames before boundary prediction. However, the lack of global context information in this model decreases the performance of the ASR model.

In this paper, we demonstrate the use of out-of-domain large-scale corpora to boost the performance of low-resource (LR) ASR tasks. To address the training data bottleneck, we propose a novel model, called ScoutWav that integrates SSL with context-based word boundaries to obtain a high-performance ASR model for LREs. ScoutWav utilizes an enhanced SN that involves a context vector embedding mechanism to capture both local acoustic features and global context attributes to obtain high-quality word boundary data for two-stage fine-tuning. Initially, we pre-train a wav2vec 2.0 model with a high-resource (HR) dataset and then fine-tune with LR data to adapt the model for the downstream task. Since different layers in a Transformer architecture can capture different ranges of linguistic information[3], we apply a wav2vec 2.0 layer analysis to detect poor layers, which do not sufficiently capture acoustic-linguistic features. These poor layers are then improved through a second fine-tuning step using context-based word boundary data to embed globalization in ScoutWav. We demonstrate the performance of ScoutWav model on two LRE datasets.

2. Related Work

SSL has been known as a paradigm for learning general data representations from unlabeled examples, then fine-tuning the model on labeled data [5]. wav2vec [8] used the Contrastive Predictive Coding (CPC) loss function for pre-training speech representations by predicting the near future frames in the acoustic sequence. The vq-wav2vec [9] model integrated wav2vec approach with Bidirectional Encoder Representations from Transformers (BERT) [10] to obtain BERT-like speech representations through two-stage training. wav2vec 2.0 [5] enhanced vq-wav2vec approach through a single-stage training by masking the input speech data into the latent space and then solving a contrastive task which is defined over a quantization of the latent representations. TERA [11] is a self-supervised pre-training method that utilizes alteration along time, frequency,

10.21437/Interspeech.2022-10270

and magnitude to pre-train Transformer Encoders on a large amount of unlabeled speech. Hidden unit BERT (HuBERT) [6] is an SSL method, which uses an offline clustering step to provide noisy labels for a BERT-like prediction loss. However, Transfer learning techniques for LR ASR has emerged as a paradigm to transfer knowledge from HR languages to LR languages and have been extensively studied. Some recent models have attempted to transfer acoustic models with a shared phone layer [12] or separate phoneme layers [13].

A considerable amount of literature has been published on the processing of end-to-end attention-based ASR. Stream-based ASR models, which are considered in the scope of this work, can be classified into look-ahead-based [14] and chunk-based models [15]. Look-ahead models apply a window for each frame to obtain the crucial context information, while in chunk-based approach the input audio segment is divided into several fixed-length chunks. Most studies in these categories use windows shifting, e.g. MoCha [16], and parametric Gaussian attention [17].

In addition, researchers have shown an increased interest in applying Transformer for online processing [18] in which local context can play an important role in addressing the issue of in acoustic modeling. In [19] a new masking technique has been proposed to improve the efficient training of the Transformer model. In [7] a low-latency streaming approach is presented for Transformer-based models, which consists of two separate networks, the scout network (SN) and the recognition network (RN). SN detects where a word starts and ends, and finally any end-to-end ASR model can be used as the RN to predict the next sub-word by utilizing all the information of the previous frames. This model has suffered from a lack of using global context information, which led to degrade performance of the ASR model in general. In this paper, we explore contextual information by adding context vectors to the Scout network approach to obtain global channel, speaker and linguistic to provide context-based word boundary to resolve this issue to prevent performance degradation in ASR model.

3. Proposed Approach

ScoutWav is an end-to-end ASR model which integrates context-based word boundary with a layer analysis module to efficiently adapt a wav2vec 2.0 pre-trained model to a target downstream ASR task in a low-resource environment. The overall ScoutWav training procedure is shown in Figure 1. Obtaining context-based representations is the main aim of the ScoutWav approach to increase the performance of the high-resource ASR model in low-resource environments. The proposed model consists of two modules: a) building context-based word boundaries and b) layer analysis-based fine-tuning. In the first module, we pre-train a SN on high-resource data and then fine-tune the model with low-resource (LR) data to achieve context-based word boundaries for the target task. In the second module, we pre-train wav2vec 2.0 with the high-resource (HR) dataset and fine-tune with the LR dataset to adapt the model for the target LR task. After fine-tuning wav2vec 2.0, we apply a layer analysis to detect the poor layers. These poor layers are then improved by a second stage of fine-tuning using the context-based word boundary data to enhance and adapt those layers to the low-resource target ASR task.

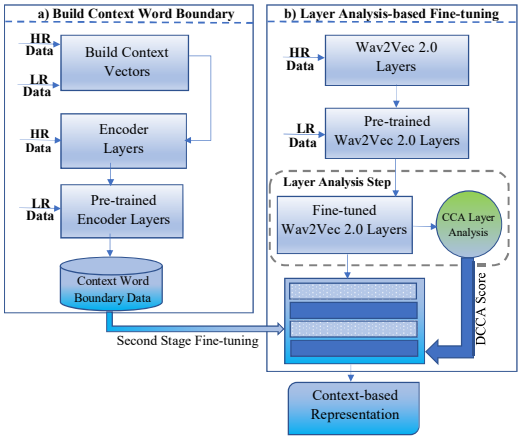

Figure 1: *ScoutWav structure and training procedure*

3.1. Build Context-Based Word Boundary

In this section, we summarize how a high-resource ASR model can be adapted for a LR task by capturing the most valuable global and local contextual information. The most valuable contextual information for preparing the annotated output text can be obtained from the speech segment that is related to the target word [7]. Therefore, a look-ahead-based SN model is used to detect the word boundary in the speech segment to identify where a word starts and ends. SN consists of CNN layers for pre-processing of the input sequence followed by N_s self-attention layers. Then, a combination of the linear layer and a sigmoid layer is used to detect the probability of the boundary p_i. The output of the current frame depends on the previous one. To train the model, the following cross-entropy loss is minimized to optimize the model for the word-boundary structure:

$$Loss = \sum b_i log(p_i) = \sum b_i log(Sigmoid(W h_i^s)) \quad ,$$

where $b_i \in 0, 1$, h_i, and W are the ground truth of the word boundary, the output of the hidden sequence, and the trainable matrices, respectively.

In LREs, the context information at each boundary should be adapted to the LR task to have reasonable performance in the target environment. An SN does not capture global contextual information when detecting the word boundary, reducing the overall performance of the ASR model in both high- and low-resource settings. In contrast, ScoutWav utilizes two sets of context vectors in each self-attention layer, that are calculated through all previous frames to capture not only local acoustic information, but also global context features. This allows ScoutWav to adapt reasonably well to the LR downstream task. The first set of the context vectors are calculated in each layer of each block and fed into the upper layer of the current layers. The second vector is obtained by concatenating all vectors in the current layer to share the global characteristics, speaker, and linguistic between the layers to enhance the adaptation procedure of the model into the LRE. We calculate the multihead self-attention as follows:

$$MHD(Q^n, K^n, V^n) = Concat(head_1, \cdots, head_m) W_O^n$$

$$head_i = Attention(Q^n W_{Q,i}^n, K^n W_{K,i}^n, V^n W_{V,i}^n) \quad ,$$

where W are trainable matrices. In the first layer, Q^1, K^1, and V^1 are represented as a feature matrix which include block input and context vector. For context vector initialization, the

positional encoding is adapted with rearranging for each layer, and only the output of each encoder layer is utilized. In the following layers, Q, K, and V are enhanced with two sets of context information vectors that are calculated from the previous layer, a context vector from each encoder of the previous layer, and a summarized context vector from all encoders in the current layer. By adding context information vector to SN, we generate an improved scout model which is able to detect more accurate word boundaries.

In order to adapt the SN model to the LR task, we pre-train the improved SN with high-resource data and then fine-tune it with a LR dataset. Our context vector mechanism allows the model to capture the context-based word boundaries.

3.2. Layer Analysis-Based Fine-tuning

In this section, we summarise how we adapt the wav2vec 2.0 approach by integrating layer analysis of the model and two-step fine-tuning mechanism to achieve a higher performance for LREs. The wav2vec 2.0 framework maps the raw audio sequence into a high-level contextual representation through a set of convolutional layers followed by self-attention layers, which are trained with a contrastive objective. Investigating the Transformer layers of the BERT model in natural language processing indicated that different blocks behave differently and capture different levels of information; the earlier blocks represent syntactic information, while the high-level ones present high-level semantic information [2]. Therefore, such a layer analysis over wav2vec 2.0 helps to have a better insight of layers behavior to enhance and fit the model for the low-resource ASR setting. To get a better understanding of layer behavior, we use Canonical Correlation Analysis (CCA) [20] inspired by [3] over different layers of wav2vec 2.0 and detect poor layers, which may not be well suited for the LR target ASR task. Then the context-based word boundaries obtained from the previous section are used for the second stage of the model fine-tuning to improve the performance of the poor layers.

We use Canonical Correlation Analysis (CCA) [20] as a measure to detect which layer of the wav2vec 2.0 model may not well suited for the target low-resource ASR task. CCA is a statistical approach to represent the maximum correlations between linear combinations of two continuous value vectors. Therefore, CCA can be used to calculate the similarity between the representations of the layers and the acoustic feature vector to evaluate how the different layers of the model are adapted to the downstream target task. CCA takes n pairs of vectors $(x_1, y_1), \cdots, (x_n, y_n)$ as input and return a correlation score as a measure of similarity between two vectors. In ScoutWav, we use Deep CCA (DCCA) [21] to dig the complex relationship between to view if data by passing into a deep network and then the output of the network fed into CCA to measure the similarity. The DCCA solution can be defined as follows:

$$\begin{aligned} &\arg\max_{W_1,W_2} \rho = tr(W_1' f_1(X^1) f_2(X^2)' W_2) \\ &s.t. \begin{cases} W_1'(f_1(X^1)f_1(X^1)' + r_1 I)W_1 = I \\ W_2'(f_2(X^2)f_2(X^2)' + r_2 I)W_2 = I \end{cases} \end{aligned} \tag{1}$$

where f_1 and f_2 are two DNN networks, $f_1(X^1)$ and $f_2(X^2)$ are DNN outputs which are interpreted by CCA to calculate the similarity score. The tr calculates the total correlation; W_1 and W_2 are corresponding weight matrix embedded; r_1 and r_2 are regularization constants. The similarity score is between 0 and 1, where 1 is the maximum similarity.

In this stage, the wav2vec 2.0 is pre-trained with HR data and then fine-tuned with the LR target data. Then the layer analysis procedure is applied through each layer with a word embedding vector to detect the poor layers. Finally, the context-based word boundaries are used as second-stage fine-tuning to fit the poor layers to the target task and improve the performance of the model.

4. Experiments

4.1. Datasets

We examine the performance of ScoutWav in the low-resource environments with two LR datasets. We use the Industrial Cobots Understanding Behavior (I-CUBE) dataset as one of our LR datasets. I-CUBE is a Human-Robot Collaboration dataset, where participants were asked to interact with an actor posing as a robot (following the Wizard of Oz protocol) using natural language [22]. They had to instruct and ultimately teach this robot how to sort different garments into four baskets as if they were sorting their own laundry. During the experiments, the robot would also respond to the participant's actions with its own actions or speech. Video recordings of each session were collected, resulting in a total of 42 videos, which represents 300 minutes of transcribed audio.

The second LRE dataset we use is the UASpeech dataset [23], which is the largest corpus of dysarthric speech in American English. It is a collection of 541 read speech recordings from 19 individuals with cerebral palsy. Furthermore, we use LibriSpeech (LS) [24], Wall Street Journal (WSJ) [25], TED-LIUM v3 (TL) [26], and Mozilla Common Voice (CV) [27] as our HR datasets.

4.2. Experiment Setup

For the WSJ, the models were trained on the SI-284 set and evaluated on the eval92 set. We trained the models with LibriSpeech, by using 960 hours of training data, and evaluated with data from both clean and contaminated testsets. Finally, for TED and CV datasets, we used 10-fold cross-validation and reported average and standard deviation WER across all folds. The input acoustic features were 80-dimensional filterbanks, extracted with a hop size of 10 ms and a window size of 25 ms, which were normalized with the mean and variance. For the WSJ setup, the number of output classes was 52, including the 26 letters of the alphabet, space, noise, symbols such as period, an unknown marker. To predict the probability distribution of all characters in the alphabet, we use the CTC loss function and use AdamW optimizer [28] as a hyperparameter setting with an initial learning rate of 0.001. The text is tokenized using SentencePiece [29] and we set the vocabulary size to 500. We run the second-stage fine-tuning stage for 20 epochs. We also use beam width $K = 10$, boundary decision threshold $\sigma = 0.0005$, language model weight $\alpha = 0.5$ and length penalty $\beta = 2.0$. We use Montreal forced aligner [30] to define phone and word segment. Finally, we pre-train and fine-tune wav2vec 2.0 in two different settings; Base setting with 12 layers and Large setting with 24 layers of the encoder.

4.3. Results

We carried out a word error rate (WER) comparison on different datasets to evaluate our proposed context-based word boundary detection model in ScoutWav with SN and a chunk-based model. In ScoutWav, we pre-train each model with HR data

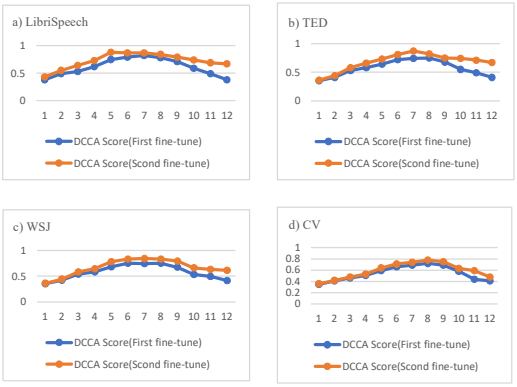

Figure 2: *Layer analysis of wav2vec 2.0 with different pre-training and fine-tuning with I-CUBE data.*

and then fine-tune with the target LR in-domain data. The results are summarized in Table 1, which shows that ScoutWav outperforms other models for both the I-CUBE and UASpeech datasets. The best performance is achieved after pre-training with LibriSpeech. This indicates a correlation between the model performance and the amount of pre-training data. In summary, adding contextual information to obtain local and global features enables ScoutWav to detect more accurate word boundaries compared to SN and chunk-based method.

Table 1: *Word error rate (WER) for detecting context-based word boundary on different datasets with different models.*

		High-Resource Data			
Model	**LR Data**	**Libri**	**WSJ**	**TED**	**CV**
SN	**I-CUBE**	16.41	18.83	17.39	20.17
	UASpeech	28.87	30.12	29.73	33.48
Chunk-Based	**I-CUBE**	19.81	21.35	20.93	22.87
	UASpeech	31.18	33.98	32.35	34.11
ScoutWav	**I-CUBE**	**14.29**	16.37	17.28	19.87
	UASpeech	**25.93**	28.17	26.53	30.13

In the second stage of our experiments, we evaluated how different layers of the pre-trained and fine-tuned wav2vec 2.0 model represents different attributes of the input acoustic. Figure 2 compares the results obtained from wav2vec 2.0 pre-training on four high-resource datasets and then fine-tuning with I-CUBE data and finally applying the second stage of fine-tuning with context-based word boundaries. Figure 2 shows DCCA scores to denote layer-to-input similarity in all pre-training datasets for the Base setting (using 12 layers). The DCCA trends are similar for the Large setting. The figure demonstrates that the first layers (1-3) and last layers (9-12) deviate from input, hence are classed as poor layers; while middle layers (5-8) are more similar to the input data and we can identify these as more suited representations for the final target task. After applying the second step fine-tuning on poor layers with word boundaries, the results show a improvement in the last layers. An interesting finding is the correlation between the accuracy of the context-based word boundary and the layer improvement: The context-based word boundary accuracy was the lowest in the CV dataset, which is mirrored by the inferior improvement rate for CV.

The analysis of the wav2vec 2.0 layers with the UASpeech dataset are shown in Figure 3. Similar to the results achieved with I-CUBE, the second step fine-tuning with obtained word boundaries helps the model to extract more contextual information from the first and last layers of the model that lead to improve the performance of the ASR model in the LRE.

Figure 2 and Figure 3 indicate that the last layers of the model have the largest improvement after the second fine-tuning

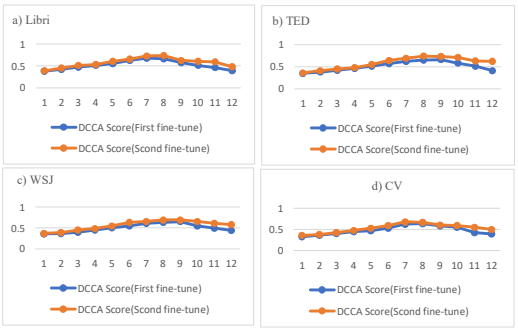

Figure 3: *Layer analysis of wav2vec 2.0 with different pre-training and fine-tuning with UASpeech data.*

Table 2: *Word error rate (WER) results for different methods in two LREs. Best performing models are highlighted.*

LRE	**Method**	**Libri**	**WSJ**	**TED**	**CV**
I-CUBE	**ScoutWav Base**	15.32	16.73	15.21	17.89
	ScoutWav Large	**10.14**	**13.98**	**12.57**	17.78
	wav2vec 2.0 Base	17.38	16.61	15.45	18.42
	wav2vec 2.0 Large	11.61	14.73	13.64	**17.22**
	QuartzNet	26.51	29.75	28.39	31.53
UASpeech	**ScoutWav Base**	18.46	22.21	19.38	24.55
	ScoutWav Large	**13.32**	**15.29**	**14.93**	**18.35**
	wav2vec 2.0 Base	19.07	23.94	21.31	25.18
	wav2vec 2.0 Large	14.28	16.23	15.19	18.87
	QuartzNet	29.15	34.93	31.79	36.79

step, which indicates that that the pre-trained and fine-tuned model is significantly improved by the context-based word boundary fine-tuning to embed task-specific information.

Finally, Table 2 presents results for the LR setup, where the second fine-tuning step was performed on the pre-trained and fine-tuned ScoutWav model and compared with wav2vec 2.0 and QuartzNet[31]. In the LR setup, the Large ScoutWav model can achieve a WER of 10.14% on I-CUBE and 13.32% on UASpeech, which are respectively 12% and 6.7% relative improvement on the than next best score of the Large wav2vec 2.0 model. The superiority of ScoutWav persists across most of settings on different datasets, where Base ScoutWav is 0.7% and 7.7% higher than Base wav2vec 2.0 on I-CUBE and UASpeech, respectively. In addition, ScoutWav also outperforms QuartzNet by a large margin in all setups.

5. Conclusions

This paper presents ScoutWav, an end-to-end LR ASR model that relies on two fine-tuning steps to adapt the HR ASR model for the task in the target domain of LR. We propose a context-based word boundary mechanism to capture both global and local acoustic properties, which enable ScoutWav to detect accurate word boundary in LREs. A layer analysis module is used to detect poor performance layers in our model. By performing the second fine-tuning step with context-based word boundary, ScoutWav shows a significant improvement in performance over two well-known ASR models.

6. Acknowledgement

This work was supported by the UKRI Engineering and Physical Sciences Research Council (EPSRC) ["Trustworthy Autonomous Systems Hub", EP/V00784X/1].

7. References

[1] J. Meyer, "Multi-task and transfer learning in low-resource speech recognition," Ph.D. dissertation, The University of Arizona, 2019.

[2] A. Baevski, W.-N. Hsu, A. Conneau, and M. Auli, "Unsupervised speech recognition," *arXiv preprint arXiv:2105.11084*, 2021.

[3] A. Pasad, J.-C. Chou, and K. Livescu, "Layer-wise analysis of a self-supervised speech representation model," *arXiv preprint arXiv:2107.04734*, 2021.

[4] W.-N. Hsu, A. Sriram, A. Baevski, T. Likhomanenko, Q. Xu, V. Pratap, J. Kahn, A. Lee, R. Collobert, G. Synnaeve *et al.*, "Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training," *arXiv preprint arXiv:2104.01027*, 2021.

[5] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *arXiv preprint arXiv:2006.11477*, 2020.

[6] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[7] C. Wang, Y. Wu, S. Liu, J. Li, L. Lu, G. Ye, and M. Zhou, "Low latency end-to-end streaming speech recognition with a scout network," *arXiv preprint arXiv:2003.10369*, 2020.

[8] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," *arXiv preprint arXiv:1904.05862*, 2019.

[9] A. Baevski, S. Schneider, and M. Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," *arXiv preprint arXiv:1910.05453*, 2019.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[11] A. T. Liu, S.-W. Li, and H.-y. Lee, "Tera: Self-supervised learning of transformer encoder representation for speech," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2351–2366, 2021.

[12] X. Li, S. Dalmia, J. Li, M. Lee, P. Littell, J. Yao, A. Anastasopoulos, D. R. Mortensen, G. Neubig, A. W. Black *et al.*, "Universal phone recognition with a multilingual allophone system," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 8249–8253.

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

[14] N. Moritz, T. Hori, and J. Le, "Streaming automatic speech recognition with the transformer model," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6074–6078.

[15] E. Tsunoo, Y. Kashiwagi, T. Kumakura, and S. Watanabe, "Towards online end-to-end transformer automatic speech recognition," *arXiv preprint arXiv:1910.11871*, 2019.

[16] C.-C. Chiu and C. Raffel, "Monotonic chunkwise attention," *arXiv preprint arXiv:1712.05382*, 2017.

[17] J. Hou, S. Zhang, and L.-R. Dai, "Gaussian prediction based attention for online end-to-end speech recognition." in *Interspeech*, 2017, pp. 3692–3696.

[18] M. Sperber, J. Niehues, G. Neubig, S. Stüker, and A. Waibel, "Self-attentional acoustic models," *arXiv preprint arXiv:1803.09519*, 2018.

[19] E. Tsunoo, Y. Kashiwagi, T. Kumakura, and S. Watanabe, "Transformer asr with contextual block processing," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 427–433.

[20] H. Hotelling, "Relations between two sets of variates," in *Breakthroughs in statistics*. Springer, 1992, pp. 162–190.

[21] X. Yang, W. Liu, W. Liu, and D. Tao, "A survey on canonical correlation analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 6, pp. 2349–2368, 2019.

[22] N. Dahlbäck, A. Jönsson, and L. Ahrenberg, "Wizard of oz studies—why and how," *Knowledge-based systems*, vol. 6, no. 4, pp. 258–266, 1993.

[23] H. Kim, M. Hasegawa-Johnson, A. Perlman, J. Gunderson, T. S. Huang, K. Watkin, and S. Frame, "Dysarthric speech database for universal access research," in *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[24] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[25] D. B. Paul and J. Baker, "The design for the wall street journal-based csr corpus," in *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992.

[26] F. Hernandez, V. Nguyen, S. Ghannay, N. Tomashenko, and Y. Esteve, "Ted-lium 3: twice as much data and corpus repartition for experiments on speaker adaptation," in *International conference on speech and computer*. Springer, 2018, pp. 198–208.

[27] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.

[28] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[29] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," *arXiv preprint arXiv:1804.10959*, 2018.

[30] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal forced aligner: Trainable text-speech alignment using kaldi." in *Interspeech*, vol. 2017, 2017, pp. 498–502.

[31] S. Kriman, S. Beliaev, B. Ginsburg, J. Huang, O. Kuchaiev, V. Lavrukhin, R. Leary, J. Li, and Y. Zhang, "Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6124–6128.