INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Improving Transformer-based Speech Recognition with Unsupervised Pre-training and Multi-task Semantic Knowledge Learning

Song Li[1], *Lin Li*[1], *Qingyang Hong*[2], *Lingling Liu*[1]

[1] School of Electronic Science and Engineering, Xiamen University, China
[2]School of Informatics, Xiamen University, China

lilin@xmu.edu.cn, qyhong@xmu.edu.cn

Abstract

Recently, the Transformer-based end-to-end speech recognition system has become a state-of-the-art technology. However, one prominent problem with current end-to-end speech recognition systems is that an extensive amount of paired data are required to achieve better recognition performance. In order to grapple with such an issue, we propose two unsupervised pre-training strategies for the encoder and the decoder of Transformer respectively, which make full use of unpaired data for training. In addition, we propose a new semi-supervised fine-tuning method named multi-task semantic knowledge learning to strengthen the Transformer's ability to learn about semantic knowledge, thereby improving the system performance. We achieve the best CER with our proposed methods on AISHELL-1 test set: 5.9%, which exceeds the best end-to-end model by 10.6% relative CER. Moreover, relative CER reduction of 20.3% and 17.8% are obtained for low-resource Mandarin and English data sets, respectively.

Index Terms: unsupervised pre-training, speech recognition, Transformer, multi-task learning, semi-supervised learning

1. Introduction

Sequence-to-sequence attention-based models have recently shown very promising results for automatic speech recognition (ASR) tasks [1–5]. Compared with the traditional speech recognition systems based on hidden Markov model (HMM), they directly learn speech-to-text mapping with the pure neural networks, without a special phoneme dictionary to convert words into phonemes. Transformer [6] is one of the state-of-the-art sequence-to-sequence architectures, which has performed promisingly in building end-to-end speech recognition systems [7–10]. Compared with RNN, Transformer introduces multi-head attention to study features from multiple subspaces and directions, so that the model can extract more representative features. In addition, it calculates in parallel, which is faster than RNN. However, end-to-end speech recognition usually requires a great deal of paired data to train, in order to achieve better recognition results. This is unfriendly to some low-resource applications. Relative to supervised data (including both speech data and the corresponding text data), unpaired data are much easier to collect. Therefore, how to use a large amount of unpaired speech and text data from real life scenario to strengthen the training performance of speech recognition systems has become one of the hot topics for researchers.

In order to make full use of unpaired data, researchers had proposed two main strategies: unsupervised pre-training and semi-supervised learning. Unsupervised pre-training strategies [11–13], like bidirectional encoder representations from Transformers (BERT) [11] and generative pre-training (GPT) [13] in the natural language processing field, aim to learn a general feature representation by unsupervised learning the data itself. Through fine-tuning stage, the learned general feature representation knowledge allows the target function for numerous downstream tasks to be set at a better start position for further optimizations, which accelerates the model convergence and improves the accuracy. Semi-supervised learning strategies [14–16] usually enhance feature extraction from the paired data through reconstructing the unpaired data with an Autoencoder [17]. These two strategies have proven efficient to use large amounts of unpaired data to enhance supervised learning.

In the speech recognition field, researchers also proposed some unsupervised pre-training strategies, such as contrastive predictive coding (CPC) [18], autoregressive predictive coding (APC) [19], and masked predictive coding (MPC) [20]. The key idea of neural network based autoregressive language model or masked language model (MLM) [11] pre-training objective is utilized in these strategies to extract representations by predicting future information or masked information. However, the limitation of these strategies is that only the extracted acoustic semantic knowledge is explored in the unpaired speech samples. In addition, [21] proposed using a trained TTS (text to speech) model to convert unpaired text into speech and [22] employed a trained ASR model to convert unpaired speech into text. The above two methods aim to make labels for unpaired data, but they are more complicated to operate than unsupervised pre-training, and inaccurate labels may cause erroneous back propagation.

Inspired by RNN-T [23] and BERT [11], we introduce unsupervised pre-training into Transformer. For Transformer's encoder and decoder, we propose two unsupervised pre-training strategies, speech predictive coding (SPC) and text predictive coding (TPC), respectively. The SPC strategy employs a large amount of unpaired speech data with an MLM-like [11] objective to obtain general feature representations for speech, such as acoustic semantic features. And the TPC strategy uses a large amount of unpaired text data with an autoregression language model objective to get general feature representations of the text, such as linguistic semantic features. In the global view, joint utilization of acoustic semantic knowledge and linguistic semantic knowledge in speech recognition systems improves the performance. In order to prevent the model from forgetting the semantic knowledge during the fine-tuning stage, we propose a new semi-supervised fine-tuning method, named multi-task semantic knowledge learning (MTSL), which further strengthens the model's learning ability of semantic knowledge. With the proposed unsupervised pre-training strategies and fine-tuning method, a large amount of unpaired data can be used to improve the performance of speech recognition systems.

The rest of the paper is organized as follows. In Section 2, the unsupervised pre-training strategies and fine-tuning method

http://dx.doi.org/10.21437/Interspeech.2020-2007

are described. In Section 3, we introduce specific experimental details. Experimental results are presented in Section 4. Finally, the paper is concluded in Section 5.

2. Our proposed methods

2.1. System overview

In this paper, we investigate our proposed unsupervised pre-training strategies and fine-tuning method on the Transformer architecture, which includes three significant components: the encoder, attention, and the decoder. We replace the position-wise fully-connected feed-forward network layers in the standard Transformer architecture with one-dimensional convolution (Conv1D) layers, which introduce more non-linear characteristics to speech recognition systems. To explore the efficient contributions of sufficient amounts of unpaired data, we pretrain the encoder and the decoder models with our proposed unsupervised pre-training strategies, respectively. The SPC pre-training aims to integrate useful acoustic semantic information contained in speech into Transformer's encoder by predicting some masked features in the speech feature sequence. The TPC pre-training provides Transformer's decoder with rich linguistic semantic information by an autoregressive language model objective. Fig.1. illustrates the details of our Transformer block and unsupervised pre-training strategies.

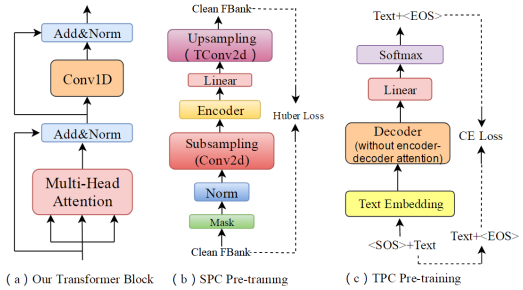

Figure 1: *A schematic representation of our unsupervised pre-training strategies and Transformer block.*

2.2. SPC model

In order to obtain acoustic semantic knowledge, SPC takes an MLM-like objective to obtain general feature representations of speech. To pre-train the SPC model, time masking is first applied to the acoustic features, so that a series of t consecutive time steps [t_0, t_0+t] are masked. The parameter t is chosen randomly from a uniform distribution from 0 to the time mask parameter W, and t_0 is chosen randomly from [0, T-t], where T is the time step of acoustic features and W is set to 30. Similar to time mask, frequency mask is used so that the frequency in range of [h_0, h_0+h] are masked. In addition, in order to make SPC model training easier, we subsample and normalize the FBank coefficients with the normalization layers and convolution layers. The down-sampled FBank coefficients are further extracted by an encoder, which is composed of our Transformer block, to obtain high-level acoustic feature representations. Finally, the clean FBank coefficients are reconstructed through a linear layer and transposed convolution layers. The detailed structure for the SPC is shown in Fig.1b. Instead of calculating the loss of all feature frames, we only calculate the loss of the position corresponding to the masked FBank coefficients. The loss function of SPC model is defined as follows:

$$L_{spc} = \frac{1}{B \cdot T} \sum_{b=1}^{B} \sum_{i=1}^{T} L_{Huber}(x_{bi}, x_{bi}{}') \tag{1}$$

$$L_{Huber}(y - y') = \begin{cases} \frac{1}{2}(y - y')^2, & \left|y - y'\right| \leq \delta \\ \delta(\left|y - y'\right| - \frac{1}{2}\delta), & otherwise \end{cases} \tag{2}$$

where $\{x_{bi}{}', i = 1, 2, ..., n\}$ are the output of SPC, $\{x_{bi}, i = 1, 2, ..., n\}$ are the original unmasked FBank coefficients, B is the batch size, and T is the time step of the FBank coefficients. L_{Huber} is the Huber loss function with $\delta = 0.5$.

The SPC model utilizes the reconstruction loss to predict the clean acoustic features from the masked acoustic features, which enables the SPC model to learn the inter-relationships within speech frames, such as the relationships between the middle speech frames and the front or the back frames, and the relationships between the local speech frames and the overall speech frames. These relationships usually represent rich semantic information from speech, which is very helpful for the downstream task (ASR). After pre-training with SPC, we remove the linear layer and upsampling layers of the SPC model and initialize the encoder of Transformer with the SPC model's weights, so that the Transformer assimilates rich acoustic semantic knowledge.

2.3. TPC model

The linguistic semantic information contained in the text is very rich. In order to obtain this information, the TPC model uses an autoregression language model objective to obtain general feature representations of the text. As shown in Fig.1c, TPC consists of a word embedding layer, a linear layer, and a decoder. The decoder is composed of our Transformer block with a masked multi-head self-attention. We train a TPC model using a large amount of unpaired text. The loss function of the TPC model is defined as follows:

$$L_{TPC} = \frac{1}{B \cdot N} \sum_{b=1}^{B} \sum_{i=1}^{N} y_{bi} \log p(y_{bi}{}' | x_{b1}, x_{b2}, ..., x_{b(i-1)}) \tag{3}$$

where B is the training batch size, and N is the index for text tokens. $\{x_{bi}, i = 1, 2, ..., n\}$ are the input text sequences for TPC, which consist of a sentence with a starting symbol $< sos >$. $\{y_{bi}{}', i = 2, ..., n\}$ are the autoregressive output sequences of TPC. $\{y_{bi}, i = 1, 2, ..., n\}$ are the target sequences of TPC, which consist of a sentence with a terminator symbol $< eos >$.

TPC model calculates the conditional probability of the next word based on previous words in an autoregressive manner, which enables the TPC model to learn about the relationships between the former and the latter words in a sentence. These relationships usually represent rich linguistic semantic knowledge from the text. The Transformer-based speech recognition system is also an autoregressive model. If we incorporate linguistic information in each decoding step, the performance of speech recognition systems improves, which is exactly the benefit of the TPC model. After pre-training, we use the TPC model's weights to initialize Transformer's decoder, except for the encoder-decoder attention part.

2.4. Multi-task semantic knowledge learning

In order to prevent Transformer from forgetting acoustic and linguistic semantic knowledge during the fine-tuning process, we propose a multi-task semantic knowledge learning (MTSL) method, which is shown in Fig.2. Specifically, an auxiliary task is introduced at the encoder output of Transformer, which reconstructs the clean acoustic features with the same loss func-

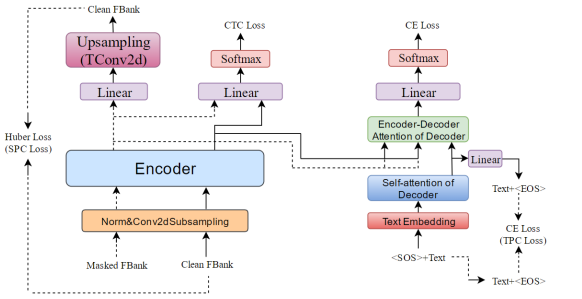

Figure 2: *The structure of fine-tuning with multi-task semantic knowledge Learning.*

tion of the SPC model. Note that reconstruction loss is back-propagated only when acoustic features are masked. In this case, the masked acoustic features are seen as one kind of data augmentations of speech. We add another auxiliary task after the self-attention of the decoder, which uses the same loss function as the TPC model for language modeling. Through multi-task semantic knowledge learning, Transformer continues to learn about acoustic and linguistic semantic knowledge during fine-tuning. The loss function of the fine-tuning process with multi-task semantic knowledge learning is defined as follows:

$$L_{MTSL} = \alpha L_{CTC} + (1-\alpha) L_{TRA} + \lambda_1 L_{SPC} + \lambda_2 L_{TPC} \quad (4)$$

where L_{CTC} is the CTC loss, L_{TRA} is the cross entropy loss of Transformer, and L_{SPC} and L_{TPC} are the SPC loss and the TPC loss respectively. α and $\lambda_{1,2}$ are hyper parameters used to balance every losses. We set $\alpha = 0.3$, $\lambda_1 = 0.2$, and $\lambda_2 = 0.1$ in this paper.

3. Experimental setup

3.1. Data sets

In our experiments, for the sake of universality, both Mandarin and English applications are considered. Mandarin datasets are used including AISHELL-1 [24], AISHELL-2 [25], THCHS30 [26] and the Datatang[1]dataset: Chinese500. LibriSpeech [27] and Tedlium2 [28] are used as the English datasets. For the AISHELL-2 datset, only 681 hours of transcription are used. AISHELL-1 training set includes about 151 hours of speech for SPC pre-training and 151 hours of transcription for TPC pre-training respectively, while AISHELL-1 fine-tuning dataset consists of 151 hours paired data (both speech and transcription). Our test experiments are executed on AISHELL-1 test set (27 hours) and Tedlium2 test set(3 hours), the other datasets are used only for unsupervised pre-training. Detailed information of unsupervised pre-training datasets are provided in Table 1. All experiments are conducted using 80-dimensional FBank coefficients, computed with a 25ms window and shifted every 10ms. The features are normalized via mean subtraction on the speaker basis. In addition, the speed perturbations of 0.9, 1.0, and 1.1 are used on the training data.

3.2. Implementation details

Our experiments were conducted on a Transformer-CTC hybrid speech recognition model. As suggested in [9], we used the configuration named *big model* for Transformer, in which d^{att}=256, d^{conv1d}=2048, d^{head}=4 , e=12, and d=6, except the one-dimensional convolution with a 1*1 convolution kernel was used to replace the position-wise fully-connected feed-forward network. In addition, a convolutional front-end was used to subsample the acoustic features by a factor of 4.

[1]A Chinese data provider (https://www.datatang.com/)

Table 1: *Detailed Unsupervised Pre-training Dataset Information*

Dataset	Speech (hours)	Text transcription (hours)
AISHELL-1	151	151
THCHS30	30	30
Chinese500	500	500
AISHELL-2	–	681
LibriSpeech	960	960

For the unsupervised pre-training stage, we used the Adam [29] optimizer with a square root learning rate scheduling [8] (25000 warmup steps, 64 mini-batch size) to train the SPC model, and used the SGD [30] optimizer with 128 mini-batch size to train the TPC model, respectively.

For the fine-tuning stage, inspired by discriminative fine-tuning [31], we set different learning rates for different layers. From the bottom to the upper layer, the learning rate of the Transformer's encoder and decoder decreases in turn, and the Adam optimizer with $weightdecay$ =1e-3 was used. We also applied three regularization methods: 10% dropout on every attention matrix and weight in Conv1D, layer normalization before every multi-head attention and Conv1D, and label smoothing with a penalty of 0.1.

4. Results

4.1. Unsupervised pre-training

Our baseline system is the Tranformer-CTC hybrid structure used in [7], except that the position-wise fully-connected feed-forward network layers are replaced with Conv1D layers, which has been proven to achieve state-of-the-art performance in all current end-to-end speech recognition systems. In addition, we conducted two representative benchmarks: the first one is a TDNN-HMM model optimized with the lattice free maximum mutual information (LF-MMI) objective, the second one is a LSTM-attention model with CTC and CE objectives. We also conducted two unsupervised pre-training methods named MPC [20] and APC [19] and fine-tuned on the same fine-tuning datasets.

We first conducted experiments with SPC and TPC using only AISHELL-1 training set as pre-training data. The results listed in Table 2 show that, compared to the baseline, 0.8% relative CER reduction is obtained with only SPC pre-training or both SPC and TPC pre-training, 0.5% with TPC pre-training . This indicates that the our proposed pre-training strategies are useful even without any additional data.

To further verify the effects of various amounts of pre-training data size on the fine-tuning results, we first merged THCHS30 and AISHELL-1 training set as a new pre-training dataset. The results listed in Table 2 show that using SPC alone achieves 6.51% CER, while using TPC alone further reduces CER to 6.41%. So, for a small amount of additional pre-training data, TPC is more useful than SPC. We further increased the pre-training data by combining AISHELL-1 training set, THCHS30, and Chinese500 to create a new pre-training dataset called $Combine$. The results show that CER decreased over the baseline relatively by 5.9% using only TPC, 7.0% using only SPC, and 7.3% using both SPC and TPC. We find that when there is more pre-training data, SPC plays a more important role.

Table 2: *The test set CER(%) of AISHELL-1 with different pre-training strategies and pre-training data size*

Model	Strategy	Unsupervised Pre-training Data Size	CER(%)
TDNN-hybrid [32]	–	–	7.51
LSTM enc + LSTM dec [5]	–	–	8.10
LSTM enc + LSTM dec [5]	APC	Combine(681 hours)	7.70
Transformer-FNN [20]	MPC	Combine(681 hours)	6.25
Transformer-FNN [7]	–	–	6.70
Transformer-Conv1d	–	–	6.60
Transformer-Conv1d-MTSL	–	–	6.50
Transformer-Conv1d	SPC	AISHELL-1(151 hours)	6.55
Transformer-Conv1d	TPC	AISHELL-1(151 hours)	6.57
Transformer-Conv1d	SPC+TPC	AISHELL-1(151 hours)	6.55
Transformer-Conv1d-MTSL	SPC+TPC	AISHELL-1(151 hours)	6.40
Transformer-Conv1d	SPC	AISHELL-1+THCHS30(181 hours)	6.51
Transformer-Conv1d	TPC	AISHELL-1+THCHS30(181 hours)	6.41
Transformer-Conv1d	SPC+TPC	AISHELL-1+THCHS30(181 hours)	6.41
Transformer-Conv1d-MTSL	SPC+TPC	AISHELL-1+THCHS30(181 hours)	6.33
Transformer-Conv1d	SPC	Combine(681 hours)	6.14
Transformer-Conv1d	TPC	Combine(681 hours)	6.21
Transformer-Conv1d	SPC+TPC	Combine(681 hours)	6.12
Transformer-Conv1d-MTSL	SPC+TPC	Combine(681 hours)	5.90
Transformer-Conv1d-MTSL	SPC+TPC	Combine(681 hours)+AISHELL-2(681 hour text)	5.91

4.2. Effects of multi-task semantic knowledge learning

The results listed in Table 2 indicate that the proposed multi-task semantic knowledge learning is beneficial for CER reduction even without unsupervised pre-training. We obtained the best CER for AISHELL-1 test set: 5.9% when we used our proposed unsupervised pre-training strategies with the *Combine* dataset as the pre-training data and with the integration of MTSL, which achieved 10.6% reduction for CER. We conducted APC and MPC with the same unsupervised pre-training data set, and the CER of AISHELL-1 only reduced by 5.3% and 4.9% respectively, which indicates that our methods are better than MPC and APC using the same unsupervised pre-training data size. We owe the effects of multi-task semantic knowledge learning to the fact that time and frequency masking has a similar effect to dropout [33], which prevents overfitting in the training set. In addition, the mask strategies of SPC for input features is similar to SpecAugment [34] , which can achieve the effect of data augmentation and improve the generalization of speech recognition systems. Finally, since the two mask strategies are equivalent to introduce some noise to input features, the reconstruction loss strengthens the anti-noise performance of the Transformer's encoder.

4.3. Randomness of unpaired data

In order to prove the effectiveness of our strategies are not due to some inherent connections in the pre-training data(originally paired), we used the speech data of *Combine* dataset to pre-train SPC and used the same-scale text of AISHELL-2, which is not paired with the Combine dataset, to pre-train TPC. The results show that we can still achieve the similar results as our best results, which indicates that our strategies are useful for arbitrary unpaired data.

4.4. Low-resource case

To prove that our proposed unsupervised pre-training strategies and fine-tuning method are still effective in low-resource case, we used *Combine* and LibriSpeech datasets to perform unsupervised pre-training using TPC and SPC strategies, and then fine-tuned on the 50-hour AISHELL-1 dataset and Tedlium2 dataset with MTSL, respectively. As shown in Table 3, the CERs of AISHELL-1 and Tedlium2 have been reduced relatively by 20.3% and 17.8%, which indicates that our proposed methods are able to learn useful semantic information from a large amount of unpaired speech and text data so as to improve the performance of low-resource speech recognition systems.

Table 3: *The test set CER(%) of AISHELL-1 and Tedlium2 Dataset in low-resource case*

Fine-tuning Dataset	Pre-training Data Size (hours)	CER(%)
AISHELL-1(50 hour)	–	15.3
AISHELL-1(50 hour)	Combine(681 hours)	12.2
AISHELL-1(50 hour)	LibriSpeech(960 hours)	14.6
Tedlium2(50 hours)	–	20.8
Tedlium2(50 hours)	LibriSpeech(960 hours)	17.1

4.5. Cross-lingual case

For some low-resource languages, we may not have enough data for pre-training. Inspired by [35], we assume that SPC pre-training in other languages is also helpful for fine-tuning on Mandarin dataset, so we used LibriSpeech dataset to pre-train the SPC model and then fine-tuned on the low resource AISHELL-1 set. As we can see from Table 3 that CER decreased relatively by 4.6% over the baseline, which indicates that there are some commonalities between speech in different languages. We attribute this improvement to some phonemic features shared between different languages at a certain level.

4.6. Loss and ACC

We observed the effectiveness of the proposed pre-training strategies for model convergence and accuracy improvement. The loss and the accuracy rate (ACC) curves are shown in Fig.3. The curve ending with *_std* represents the baseline without pre-training, while the curve ending with *_Com* represents the case where the pre-training strategies are used with the *Combine* data set. We can find that the pre-training strategies proposed in this paper provide a better initial position for model training, so that the model converges faster, and the accuracy increases faster. It's obviously that rich acoustic and linguistic knowledge obtained from pre-trained SPC and TPC models benefits downstream automatic speech recognition (ASR) tasks.

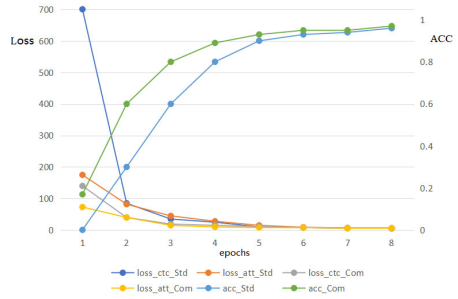

Figure 3: *Loss and ACC curves with and without our proposed methods.*

5. Conclusions

In this paper, we propose two unsupervised pre-training strategies, named speech predictive coding (SPC) and text predictive coding (TPC). These strategies use a large amount of unpaired speech and text data for pre-training, and provide rich acoustic and linguistic semantic information for downstream tasks. We also propose a new semi-supervised fine-tuning method, named multi-task semantic knowledge learning, which helps Transformer to strengthen the learning capability of semantic knowledge during the fine-tuning process. Through our unsupervised pre-training strategies and the fine-tuning method, the performance of Transformer-based speech recognition system is improved, which is suitable for the low-resource and cross-lingual speech recognition applications.

6. Acknowledgements

This work is supported by the National Natural Science Foundation of China (Grant No.61876160).

7. References

[1] A. Mohamed, D. Okhonko, and L. Zettlemoyer, "Transformers with convolutional context for ASR," *arXiv preprint arXiv:1904.11660*, 2019.

[2] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4774–4778.

[3] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 4945–4949.

[4] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell," *arXiv preprint arXiv:1508.01211*, 2015.

[5] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in neural information processing systems*, 2015, pp. 577–585.

[6] 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.

[7] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang *et al.*, "A comparative study on Transformer vs RNN in speech applications," *arXiv preprint arXiv:1909.06317*, 2019.

[8] T. Nakatani, "Improving Transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration," 2019.

[9] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5884–5888.

[10] S. Zhou, L. Dong, S. Xu, and B. Xu, "A comparison of modeling units in sequence-to-sequence speech recognition with the Transformer on Mandarin Chinese," in *International Conference on Neural Information Processing*. Springer, 2018, pp. 210–220.

[11] 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.

[12] A. Raganato, J. Tiedemann *et al.*, "An analysis of encoder representations in Transformer-based machine translation," in *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics, 2018.

[13] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training." *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/language-unsupervised/language_understanding_paper. pdf*, 2016.

[14] S. Karita, S. Watanabe, T. Iwata, M. Delcroix, A. Ogawa, and T. Nakatani, "Semi-supervised end-to-end speech recognition using text-to-speech and autoencoders," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6166–6170.

[15] B. Li, T. N. Sainath, R. Pang, and Z. Wu, "Semi-supervised training for end-to-end models via weak distillation," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 2837–2841.

[16] Y. Ren, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Almost unsupervised text to speech and automatic speech recognition," *arXiv preprint arXiv:1905.06791*, 2019.

[17] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 1096–1103.

[18] A. V. D. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[19] Y.-A. Chung, W.-N. Hsu, H. Tang, and J. Glass, "An unsupervised autoregressive model for speech representation learning," *arXiv preprint arXiv:1904.03240*, 2019.

[20] D. Jiang, X. Lei, W. Li, N. Luo, Y. Hu, W. Zou, and X. Li, "Improving Transformer-based speech recognition using unsupervised pre-training," *arXiv e-prints*, p. arXiv:1910.09932, Oct 2019.

[21] M. Mimura, S. Ueno, H. Inaguma, S. Sakai, and T. Kawahara, "Leveraging sequence-to-sequence speech synthesis for enhancing acoustic-to-word speech recognition," in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 477–484.

[22] T. Hayashi, S. Watanabe, Y. Zhang, T. Toda, T. Hori, R. Astudillo, and K. Takeda, "Back-translation-style data augmentation for end-to-end ASR," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 426–433.

[23] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6381–6385.

[24] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, "AISHELL-1: An open-source Mandarin speech corpus and a speech recognition baseline," in *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*. IEEE, 2017, pp. 1–5.

[25] J. Du, X. Na, X. Liu, and H. Bu, "AISHELL-2: Transforming Mandarin ASR research into industrial scale," *arXiv preprint arXiv:1808.10583*, 2018.

[26] D. Wang and X. Zhang, "THCHS-30: A free chinese speech corpus," *arXiv preprint arXiv:1512.01882*, 2015.

[27] 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.

[28] A. Rousseau, P. Deléglise, and Y. Esteve, "Enhancing the ted-lium corpus with selected data for language modeling and more ted talks." in *LREC*, 2014, pp. 3935–3939.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[30] N. S. Keskar and R. Socher, "Improving generalization performance by switching from Adam to SGD," *arXiv preprint arXiv:1712.07628*, 2017.

[31] J. Howard and S. Ruder, "Fine-tuned language models for text classification," *arXiv preprint arXiv:1801.06146*, 2018.

[32] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI." in *Interspeech*, 2016, pp. 2751–2755.

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[34] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.

[35] J. Cho, M. K. Baskar, R. Li, M. Wiesner, S. H. Mallidi, N. Yalta, M. Karafiat, S. Watanabe, and T. Hori, "Multilingual sequence-to-sequence speech recognition: architecture, transfer learning, and language modeling," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 521–527.