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

Phonetically Motivated Self-Supervised Speech Representation Learning

Xianghu Yue, Haizhou Li

Department of Electrical and Computer Engineering, National University of Singapore, Singapore
xianghu.yue@u.nus.edu, haizhou.li@nus.edu.sg

Abstract

Self-supervised representation learning has seen remarkable success in encoding high-level semantic information from unlabelled speech data. The studies have been focused on exploring new pretext tasks to improve the learned speech representation and various masking schemes with reference to speech frames. We consider effective latent speech representation should be phonetically informed. In this work, we propose a novel phonetically motivated masking scheme. Specifically, we select the masked speech frames according to the phonetic segmentation in an utterance. The phonetically motivated self-supervised representation learns the speech representation that benefits downstream speech processing tasks. We evaluate the proposed learning algorithm on phoneme classification, speech recognition, and speaker recognition, and show that it consistently outperforms competitive baselines.

Index Terms: self-supervised learning, representation learning, pre-training, speech recognition

1. Introduction

Speech signal contains acoustic and linguistic information at multiple levels, such as short-term frame, phoneme, lexical word, phrase, and sentence [1, 2]. Recently, self-supervised speech representation learning [2, 3, 4, 5, 6, 7, 8, 9] has seen remarkable success in encoding high-level information from a large amount of unlabeled speech data, that benefits downstream speech processing tasks, such as automatic speech recognition (ASR), speaker verification (SV), and speech translation (ST).

Self-supervised learning leverages unsupervised pre-training strategy to discover useful representations from unlabeled data. Such speech representations are typically more general and robust than those derived from supervised learning, which tend to bias towards downstream applications [10]. Therefore, self-supervised representation learning is usually studied for various downstream applications [11].

The recent self-supervised algorithms for speech representations include Autoregressive Predictive Coding (APC) [2, 3], Mockingjay [4], TERA [5], Speech SimCLR [12], Speech XLNet [13], wav2vec [6, 7] and Problem-agnostic Speech Encoder (PASE) [8, 9], to name a few. Generally, there are two major branches of self-supervised speech representation learning methods: contrastive-based method and reconstruction-based method.

A typical contrastive-based method is Contrastive Predictive Coding (CPC) [11, 14, 6], which is a mutual information maximization method that has been successfully applied to both speech and image processing [15]. The reconstruction-based method includes the autoregressive prediction [2, 3, 16] and BERT-style masked reconstruction algorithms [4, 5, 17, 18]. APC and CPC have a similar methodology conditioning on the past context to predict the future information and learn representations based on autoregressive model. The main difference is that APC attempts to optimize the model via L1 regression while CPC optimizes the model by discriminating the future frames from the negative samples using InfoNCE loss [19].

Inspired by the success of Masked Language Model (MLM) of BERT [20, 21] in language representation learning, recent studies have investigated the BERT-style masked reconstruction to learn speech representations. Mockingjay [4] randomly masks the input speech frames into zero to pre-train the attention-based encoder, where the masking policy is similar to BERT [20] and RoBERTa [21]. Audio ALBERT [22] explores a lite version of the self-supervised speech representation model based on Mockingjay [4]. TERA [5] is another extended version of Mockingjay, where the alterations are done along three dimensions: temporal, channel and magnitude. In [17, 18], they also follow the standard BERT masking policy. In [23], the input features are divided into chunks of four frames, and applied mask on chunks with a probability of 15%. The above masking schemes directly follow those in natural language processing (NLP), and show effectiveness in speech processing by replacing a discrete text symbol with a speech frame. We note that speech and text are different in nature, where speech frames are of continuous value, while text symbols are discrete. Furthermore, as far as speech recognition is concerned, it makes more sense to have a phoneme as the minimum speech unit than a speech frame.

We propose a novel phonetically motivated masking strategy for reconstruction-based self-supervised learning. In other words, we use phonemes instead of speech frames to define the mask. In this way, we expect the model to learn the phonotactic constraint of a spoken language to benefit the downstream tasks. To verify the proposal, we first use the force-alignment to find the phoneme boundary, that is required for the phoneme-based masking. We then apply the learned representations to several downstream speech recognition tasks. We also study the unsupervised phoneme segmentation method, a self-supervised learning technique for phoneme segmentation.

To the best of our knowledge, this is the first study on phoneme-based masking strategy for speech representation learning. We evaluate our method on LibriSpeech [24] and TIMIT [25] datasets. Experiments show that the proposed phoneme-based masking strategy benefits the downstream tasks, and outperforms frame-based masking schemes. The rest of the paper is organized as follows: In Section 2, we formulate the proposed masking scheme. Experiments and analysis on LibriSpeech and TIMIT datasets are reported in Section 3 and Section 4, respectively. Finally, we conclude in Section 5.

2. Phoneme-based Masked Predictive Coding

We propose to use bidirectional Transformer encoder [26], similar to Mockingjay [4], to learn speech representations via masked reconstruction objective as illustrated in Figure 1. Each

http://dx.doi.org/10.21437/Interspeech.2021-905

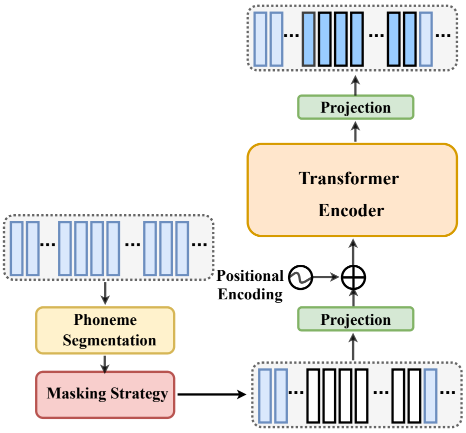

Figure 1: *An overview of the proposed phoneme-based masked predictive coding (pMPC) model architecture.*

encoder layer consists of two sub-layers: (1) a multi-head self-attention module, and (2) a position-wise fully connected feed-forward network. Each sub-layer has a residual connection, followed by layer normalization [27].

2.1. Transformer Encoder Block

Multi-head attention (MHA) is the core module of Transformer encoders, which learns the relationship between queries, keys and values from different representation subspaces at different positions. The basic unit of MHA is self-attention module calculated by the following equation:

$$SelfAttention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

where $Q \in \mathbb{R}^{t_q \times d_q}$, $K \in \mathbb{R}^{t_k \times d_k}$, and $V \in \mathbb{R}^{t_v \times d_v}$ are queries, keys and values, respectively. MHA performs multiple attention functions with d_{model} dimensional queries, keys and values,

$$MHA(Q, K, V) = Concat(head_1, \dots, head_H)W^O$$
$$head_h = SelfAttention(QW_h^Q, KW_h^K, VW_h^V)$$

where $W_h^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_h^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_h^V \in \mathbb{R}^{d_{model} \times d_v}$, and $W_h^O \in \mathbb{R}^{H \times d_v \times d_{model}}$ are parameter matrices, H is the number of self-attention heads, and h is the head index.

Since the Transformer encoder contains no recurrence and no convolution, we use positional encoding to allow the model aware of the relative position information of the input acoustic sequence order. Specifically, we first linearly project the input frames into the dimension of the model d_{model} and then add them with the positional encoding. We use sinusoidal positional encoding instead of learnable positional encoding, because the lengths of the input speech frames are arbitrary with high variance:

$$PE_{(pos,2j)} = sin(pos/10000^{2j/d_{model}})$$
$$PE_{(pos,2j+1)} = cos(pos/10000^{2j/d_{model}})$$

where pos is the position of the input frames and j is the dimension.

2.2. Phoneme-based Masked Predictive Coding

With a frame-based masking scheme, the masked predictive coding (MPC) technique learns speech representations to recover the original masked frames. Hence, the masking scheme plays a vital role in the effectiveness of the self-supervised learning. Typically, a frame-based masking scheme firstly selects 15% of frames, and then 1) masks 80% of the time to zero, 2) replaces 10% of the time with a random frame, 3) leaves 10% of the time unchanged, which simply adopts the BERT masking strategy in NLP by replacing a text symbol with one or more consecutive speech frames.

Although effective, this simple random masking scheme treats speech as a sequence of features, without considering its phoneme-based construct. Addressing this issue, we propose to mask and recover phonemes instead of speech frames, and formulate a phoneme-based masked predictive coding (pMPC) scheme. By doing so, we expect that the model learns the strong phonotactic constraint that is present in the spoken languages, instead of the weak frame constraint, which ultimately benefits phoneme-based speech recognition tasks in downstream.

2.3. Unsupervised Phoneme Segmentation

Now the question is how to obtain the phonetic boundary in a spoken utterance. One can easily think of two ways: 1) force-alignment by a speech recognizer when the phonetic transcription of the utterance is available, 2) unsupervised phoneme segmentation that doesn't require the phonetic transcription. In this paper, we explore the use of a state-of-the-art unsupervised phoneme boundary detection technique [28], where the model is a convolutional neural network that operates directly on raw speech waveform and optimized to discriminate spectral changes of the speech signal using Noise-Contrastive Estimation (NCE) criterion [19].

Specifically, the self-supervised phoneme segmentation model learns an encoding function $f : \mathcal{X} \rightarrow \mathcal{Z}$, from the time-domain of speech signal to the spectral representations. The function f is optimized to discriminate between the natural adjacent frame pairs in the sequence $\mathbf{z}$, and randomly sampled frame pairs, i.e., negative distractor. Hence, the loss for the i^{th} frame $\mathbf{z}_i$ in an utterance is:

$$\mathcal{L}(\mathbf{z}_i, D_K(\mathbf{z}_i)) = -log\frac{e^{sim(\mathbf{z}_i,\mathbf{z}_{i+1})}}{\sum_{\mathbf{z}_j \in \{\mathbf{z}_{i+1}\} \cup D_K(\mathbf{z}_i)} e^{sim(\mathbf{z}_i,\mathbf{z}_j)}}$$

where $sim(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^T\boldsymbol{v}/||\boldsymbol{u}||\,||\boldsymbol{v}||$ is the cosine similarity between two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$, and $D(\mathbf{z}_i)$ is the set of non-adjacent frames to $\mathbf{z}_i$,

$$D(\mathbf{z}_i) = \{\mathbf{z}_j : |i - j| > 1, \mathbf{z}_j \in \mathbf{z}\}$$

$D_K(\mathbf{z}_i)$ contains K randomly selected frames from $D(\mathbf{z}_i)$. Overall, given a training set with N examples $S = \{\mathbf{x}_n\}_{n=1}^N$, the objective function is:

$$\mathcal{L} = \sum_{\mathbf{x}_n \in S} \sum_{\mathbf{z}_i \in f(\mathbf{x})} \mathcal{L}(\mathbf{z}_i, D_K(\mathbf{z}_i))$$

During inference, given a new utterance $\hat{\mathbf{m}}$, we apply the encoding function to obtain $\hat{\mathbf{x}} = f(\hat{\mathbf{m}})$. Then we compute the score for a boundary at time t between the t^{th} frame and the $t + 1^{th}$ for $t = 1, \dots, T - 1$,

$$score(\hat{\mathbf{x}}_t) = -sim(\hat{\mathbf{x}}_t, \hat{\mathbf{x}}_{t+1})$$

$score(\hat{\mathbf{x}}_t)$ represents the confidence about whether $\hat{\mathbf{x}}_t$, and $\hat{\mathbf{x}}_{t+1}$ belong to the same phonetic segment. A high dissimilarity value indicates the transition from one segment to another. We apply a peak detection algorithm over the dissimilarity values, $score(\hat{\mathbf{x}})$ to obtain the final segmentation.

2.4. Transferring to Downstream Tasks

Just like other pre-trained models, one can transfer the pre-trained speech representations to downstream tasks. In general, there are two ways: fine-tuning and representation learning. With fine-tuning, the whole model, including pMPC model and the additional layers for downstream tasks, participates in the task-oriented training with the downstream data. With representation learning, the parameters in the pre-trained pMPC model are fixed. A typical implementation is to use the activations at the final layer as the features for downstream tasks, which is adopted in this work.

3. Experiments

We evaluate the quality of the speech representations extracted from pre-trained pMPC models on several downstream tasks, including phoneme classification, speech recognition and two speaker identification tasks. For different downstream tasks, different downstream models are trained with different hyperparameters. The detailed model architectures and hyperparameters are elaborated in the following subsections.

3.1. Data

We use publicly available LibriSpeech [24] and TIMIT [25] datasets in our experiments. We perform self-supervised pre-training on *train-clean-100*, *train-clean-360*, or the full 960 hours (*train-clean-100* + *train-clean-360* + *train-other-500*) data of LibriSpeech. We also consider TIMIT for ASR to evaluate the transferability of the pretrained pMPC models.

3.2. Experimental setup

We implement the pMPC model similar to Mockingjay [4] architecture for a fair comparison, in which we have the model dimension $d_{model} = 768$, attention heads $H = 12$, feed-forward size of $3,072$, and encoder layer $L = 3$. We use 80-dimensional FBANK features (normalized to zero mean and unit variance per speaker) as inputs using Kaldi toolkit [29]. The total number of training steps is set to 200k, 500k and 1M for 100, 360, and 960 hours, respectively. The batch size is 6 and the Adam optimizer [30] is used, warming up the learning rate for the first 7 % of total training steps to a peak of 4e-4 and then linearly decayed. Two phoneme segmentation methods, namely force-alignment and unsupervised phoneme segmentation, are compared. We implement the force-alignment with a GMM-HMM speech recognizer trained on LibriSpeech [29]. For long phoneme segments, we only mask 12 frames at their centers.

3.2.1. Unsupervised phoneme segmentation

To obtain the phoneme boundary, we use the pre-trained unsupervised phoneme segmentation model from [28], which is trained on the *train-other-500* subset of LibriSpeech. As reported in Table 1, the results suggest that the chosen unsupervised segmentation method by Felix et al. [28], surpasses several unsupervised baselines and reaches state-of-the-art performance. Moreover, this model achieves comparable results to a supervised method based on a Kernel-SVM [31]. Unlike the force-alignment obtained by GMM-HMM, which requires additional ASR and phonetic transcription, this model is totally self-supervised without the need of phonetic transcription.

Table 1: *The comparison of phoneme segmentation models using TIMIT dataset. P means Precision and R means recall, both are calculated with tolerance value of 20 ms.*

Setting	Precision	Recall	F1	R-val
Unsupervised model				
Hoang et al. [32]	-	-	78.20	81.10
Michel et al.[33]	74.80	81.90	78.20	80.10
Felix et al. [28]	**83.89**	**83.55**	**83.71**	**86.02**
Supervised model				
King et al. [31]	87.00	84.80	85.90	87.80
Franke et al. [34]	91.10	88.10	89.6	90.80
Kreuk et al. [35]	94.03	90.46	92.22	92.79

3.2.2. Phoneme classification

To measure the phonetic information in the learned representations, we train a classifier on top of the pMPC model. Following the common setting of previous work, we use a linear classifier to evaluate the linear separability of phonemes on the *train-clean-100* subset of LibriSpeech. For a fair comparison, we use the aligned phoneme labels and train/test split provided in the CPC [11] paper, where there are 41 possible phoneme classes obtained using the Kaldi toolkit [29] and pre-trained models on LibriSpeech [24]. Besides, we also train a two-layer linear classifier since not all the encoded information from pMPC model is linearly accessible.

3.2.3. Speech recognition

We build a hybrid DNN/HMM ASR system on TIMIT to observe the effort of domain mismatch of the learned representations between training data, i.e., LibriSpeech, and downstream task, i.e., TIMIT. The hybrid ASR system is implemented with the PyTorch-Kaldi toolkit [36]. We use the advanced DNN architecture, which is a 5-layer light gated recurrent units (liGRU) followed by 2-layers of fully-connected network, in the hybrid ASR framework. The output of the pMPC model is directly fed to the DNN and the parameters of the pMPC model are frozen during training. Following the conventional settings, the ASR model on TIMIT is based on 48 phoneme classes, while accuracy is measured after mapping the prediction to a smaller set of 39 phoneme classes. Splits of these sets are obtained from the Kaldi TIMIT [29] recipe.

3.2.4. Speaker identification

Here we evaluate the model performance with two tasks, utterance-level and frame-level speaker identification, both are linear classifiers trained on the *train-clean-100* subset of LibriSpeech, which consists of 251 speakers. For utterance-level speaker identification, the representations of each utterance are first averaged over time, then the classifier predicts speaker identity conditioning on the averaged vector. For frame-level speaker identification, the classifier predicts the speaker identity for each input speech frame. We use the same train/test split as provided in the CPC [11] paper.

4. Results and Discussion

Table 2 reports the results of phone classification accuracy using different self-supervised speech representations, including CPC, TERA, and Mockingjay. All self-supervised models are pre-trained on the *train-clean-100* subset of LibriSpeech. We also present the classification accuracy of three surfaces features, MFCC, FBANK, and fMLLR. As expected, all self-supervised speech representations outperform the surface features on both classifiers, indicating the effectiveness of the self-

Table 2: *Phoneme classification results on the train-clean-100 subset of LibriSpeech. Accuracy (%) are obtained using linear classifier and non-linear classifier (1 hidden layer).*

Models	linear	non-linear
Self-supervised representations		
CPC [11]	64.6	72.5
TERA [5]	65.1	77.3
Mockingjay [4]	64.3	76.8
pMPC (unsupervised)	67.3	78.8
pMPC (force-alignment)	**68.5**	**78.9**
Surface features		
MFCC	39.7	59.9
FBANK	42.1	46.9
fMLLR	52.6	68.4

Table 3: *Comparison of different pretraining data size on phoneme classification .*

Models	100 hr	360 hr	960 hr
TERA [5]	65.1	66.4	66.4
Mockingjay [4]	64.3	64.4	67.0
pMPC	**67.3**	**67.8**	**68.6**

Table 4: *Phone Error Rate (PER) of different pre-training approaches on TIMIT.*

Models	pretrain	PER
liGRU + MFCC [37]	None	16.7
liGRU + FBANK [37]	None	15.8
liGRU + fMLLR [37]	None	14.9
wav2vec [6]	80 hr	17.6
wav2vec [6]	960 hr	15.6
liGRU + TERA [5]	100 hr	15.2
liGRU + TERA [5]	360 hr	14.9
liGRU + TERA [5]	960 hr	14.5
pMPC (force-alignment)	100 hr	14.4
pMPC (unsupervised)	100 hr	14.6
pMPC (unsupervised)	360 hr	14.2
pMPC (unsupervised)	960 hr	**14.0**

Table 5: *Speaker identification results on the train-clean-100 subset of LibriSpeech. Testing set accuracy (%) are obtained using frame-level and utterance-level classifier, respectively.*

Models	frame	utterance
CPC [11]	97.4	-
TERA [5]	98.9	99.2
Mockingjay [4]	68.4	96.1
pMPC (unsupervised)	98.5	99.7
pMPC (force-alignment)	**99.5**	**99.9**

supervised learning for capturing high-level phonetic information. Among all self-supervised speech representations, the proposed phoneme masking strategy of pMPC based on force-alignment gives the best results (68.6% and 79.2 %), better than the baseline random masking method of Mockingjay (64.3 % and 76.8 %). The results of our method also outperform the TERA, the enhanced version of Mockingjay, which suggests that our masking strategy can force the model to learn more phonetic information and benefit the downstream tasks. Moreover, because the force-alignment gives more accurate phoneme boundary, the pre-trained model using these boundary information yields better speech representations and performance on phoneme classification.

Table 3 shows the accuracy of the linear phone classifier when we increase the amount of pretraining data. These models are all pre-trained using unsupervised phoneme segmentation. When the proposed pMPC is only trained on 100 hours data, the accuracy is already higher than TERA and Mockingjay trained on full 960 hours data of Librispeech. This indicates incorporating phoneme boundary information when selecting mask will boost the quality of the learned speech representations.

The speech recognition results are presented in Table 4. TERA, wav2vec and the proposed method are pre-trained on LibriSpeech data while the speech recognition systems are performed on TIMIT data, so there is much domain mismatch between the pretraining and downstream ASR. In this experiment, we explore the transferability of the learned speech representations. The results show that pMPC achieves the best PER among the pretraining approaches. pMPC (14.0 %) outperforms the wav2vec (14.7 %) trained on full Librispeech and Wall Street Journal (WSJ) data, as well as the strong supervised baseline (14.9 %). pMPC trained on only 100 hours of speech data even outperforms both wav2vec and TERA trained on 960 hours of speech data.

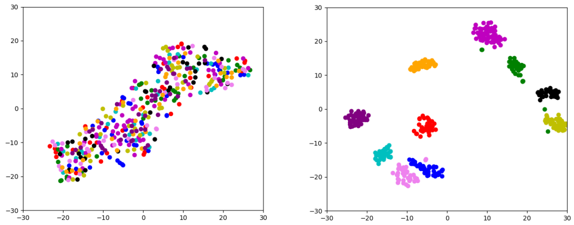

Figure 2: *Visualization of 10 speakers representations via t-SNE, who are color-coded.*

Table 5 shows the frame-level and utterance-level speaker identification results. The results show that our method gives 99.5 % and 99.9 % on frame-level and utterance-level tasks, respectively. Surprisingly, pMPC encodes well the speaker information. We conclude that the model learns the global phonotactic constraints via phoneme-based masking, that contributes to speaker recognition.

In Figure 2, we randomly select ten speakers and use t-SNE to visualize the acoustic speech features and learned speech representations from the MPC model of each speaker. Each point in the figure represents an utterance, which is generated by one average pooling layer, and different colors represent different speakers. From the figure, we observe that the representations of the same speaker from the pMPC model are clustered together and different speakers have different clusters, while the acoustic speech features are just randomly distributed without any cluster, indicating that the pMPC could encode much speaker information.

5. Conclusion

In this paper, we propose a phoneme-based masked predictive coding scheme to learn self-supervised speech representations. We investigate two different phoneme segmentation methods and show that the force-alignment delivers better performance. We evaluate the learned speech representations with pMPC masking scheme on phoneme classification, speech recognition, and speaker identification tasks, and confirm that phoneme-based masking strategy is more effective than frame-based masking.

6. Acknowledgement

This research work is supported by Programmatic Grant No. A1687b0033 from the Singapore Government's Research, Innovation and Enterprise 2020 plan (Advanced Manufacturing and Engineering domain).

7. References

[1] Y.-A. Chung, Y. Belinkov, and J. Glass, "Similarity analysis of self-supervised speech representations," in *ICASSP*, 2021.

[2] Y.-A. Chung, W.-N. Hsu, H. Tang, and J. Glass, "An unsupervised autoregressive model for speech representation learning," in *Interspeech*, 2019, pp. 146–150.

[3] ——, "Generative pre-training for speech with autoregressive predictive coding," in *ICASSP*, 2020, pp. 3497–3501.

[4] A. Liu, S.-W. Yang, P.-H. Chi, P.-C. Hsu, and H.-Y. Lee, "Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders," in *ICASSP*, 2020, pp. 6419–6423.

[5] A. Liu, S.-W. Li, and H.-Y. Lee, "Tera: Self-supervised learning of transformer encoder representation for speech," in *arXiv preprint arXiv:2007.06028*, 2020.

[6] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," in *Interspeech*, 2019, pp. 3465–3469.

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

[8] S. Pascual, M. Ravanelli, J. Serrà, A. Bonafonte, and Y. Bengio, "Learning problem-agnostic speech representations from multiple self-supervised tasks," in *Interspeech*, 2019, pp. 161–165.

[9] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, "Multi-task self-supervised learning for robust speech recognition," in *ICASSP*, 2020, pp. 6989–6993.

[10] J. Chorowski, R. J. Weiss, S. Bengio, and A. van den Oord, "Unsupervised speech representation learning using wavenet autoencoders," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 2041–2053, 2019.

[11] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," in *arXiv preprint arXiv:1807.03748*, 2018.

[12] D. Jiang, W. Li, M. Cao, R. Zhang, W. Zou, K. Han, and X. Li, "Speech simclr: Combining contrastive and reconstruction objective for self-supervised speech representation learning," in *arXiv preprint arXiv:2010.13991*, 2020.

[13] X. Song, G. Wang, Y. Huang, Z. Wu, D. Su, and H. Meng, "Speech-xlnet: Unsupervised acoustic model pretraining for self-attention networks," in *Interspeech*, 2020, pp. 3765–3769.

[14] M. Rivière, A. Joulin, P.-E. Mazaré, and E. Dupoux, "Unsupervised pretraining transfers well across languages," in *ICASSP*, 2020, pp. 7414–7418.

[15] O. J. Henaff, A. Srinivas, J. D. Fauw, A. Razavi, C. Doersch, S. M. A. Eslami, and A. van den Oord, "Data-efficient image recognition with contrastive predictive coding," in *ICML*, 2020, pp. 4182–4192.

[16] Y.-A. Chung and J. Glass, "Improved speech representations with multi-target autoregressive predictive coding," in *ACL*, 2020, pp. 2353–2358.

[17] D. Jiang, X. Lei, W. Li, N. Luo, Y. Hu, W. Zou, and X. Li, "Improving transformer-based speech recognition using unsupervised pre-training," in *arXiv preprint arXiv:1910.09932*, 2019.

[18] L. Liu and Y. Huang, "Masked pre-trained encoder base on joint ctc-transformer," in *arXiv preprint arXiv:2005.11978*, 2020.

[19] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation of unnormalized statistical models with applications to natural image statistics," *Journal of Machine Learning Research*, vol. 13, no. 2012, pp. 605–626.

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL*, 2019, pp. 4171–4186.

[21] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," in *arXiv preprint arXiv:1907.11692*, 2019.

[22] P.-H. Chi, P.-H. Chung, T.-H. Wu, C.-C. Hsieh, S.-W. Li, and H. yi Lee, "Audio albert: A lite bert for self-supervised learning of audio representation," in *SLT*, 2021.

[23] D. Jiang, W. Li, R. Zhang, M. Cao, N. Luo, Y. Han, W. Zou, and X. Li, "A further study of unsupervised pre-training for transformer based speech recognition," in *arXiv preprint arXiv:2005.09862*, 2020.

[24] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *ICASSP*, 2015, pp. 5206–5210.

[25] K.-F. Lee and H.-W. Hon, "Speaker-independent phone recognition using hidden markov models," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 37, no. 11, pp. 1641–1648, 1989.

[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2018, pp. 6000–6010.

[27] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," in *arXiv arXiv:1607.06450*, 2016.

[28] F. Kreuk, J. Keshet, and Y. Adi, "Self-supervised contrastive learning for unsupervised phoneme segmentation," in *Interspeech*, 2020, pp. 3700–3704.

[29] 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.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[31] S. King and M. Hasegawa-Johnson, "Accurate speech segmentation by mimicking human auditory processing," in *ICASSP*, 2013, pp. 8096–8100.

[32] D.-T. Hoang and H.-C. Wang, "Blind phone segmentation based on spectral change detection using legendre polynomial approximation," *The Journal of the Acoustical Society of America*, vol. 137, no. 2, pp. 797–805, 2015.

[33] P. Michel, O. Rasanen, R. Thiollière, and E. Dupoux, "Blind phoneme segmentation with temporal prediction errors," in *ACL*, 2017, pp. 62–68.

[34] J. Franke1, M. Müller1, F. Hamlaoui, S. Stüker, and A. Waibel, "Phoneme boundary detection using deep bidirectional lstms," in *ITG Symposium on Speech Communication*, 2016, pp. 1–5.

[35] F. Kreuk, Y. Sheena, J. Keshet, and Y. Adi, "Phoneme bound- ary detection using learnable segmental features," in *ICASSP*, 2020, pp. 8089–8093.

[36] M. Ravanelli, T. Parcollet, and Y. Bengio, "The pytorch-kaldi speech recognition toolkit," in *ICASSP*, 2019, pp. 6465–6469.

[37] M. Ravanelli, P. Brakel, M. Omologo, and Y. Bengio, "Light gated recurrent units for speech recognition," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, pp. 92–102, 2018.