Interspeech 2022
18-22 September 2022, Incheon, Korea

Similarity and Content-based Phonetic Self Attention for Speech Recognition

Kyuhong Shim and Wonyong Sung

Department of Electrical and Computer Engineering, Seoul National University, Korea
{skhu20, wysung}@snu.ac.kr

Abstract

Transformer-based speech recognition models have achieved great success due to the self-attention (SA) mechanism that utilizes every frame in the feature extraction process. Especially, SA heads in lower layers capture various phonetic characteristics by the query-key dot product, which is designed to compute the pairwise relationship between frames. In this paper, we propose a variant of SA to extract more representative phonetic features. The proposed phonetic self-attention (phSA) is composed of two different types of phonetic attention; one is similarity-based and the other is content-based. In short, similarity-based attention captures the correlation between frames while content-based attention only considers each frame without being affected by other frames. We identify which parts of the original dot product equation are related to two different attention patterns and improve each part with simple modifications. Our experiments on phoneme classification and speech recognition show that replacing SA with phSA for lower layers improves the recognition performance without increasing the latency and the parameter size.

Index Terms: speech recognition, self attention, transformer, phoneme classification, phonetic attention

1. Introduction

End-to-end automatic speech recognition (ASR) has made great progress in line with the advances in deep neural networks (DNNs). Among various architectures, Transformer [1] models have shown state-of-the-art performance [2, 3, 4, 5, 6] in ASR. Most Transformer-based ASR models stack the same layer multiple times without considering the difference between layer positions, although the behaviors are very different [7, 8, 9]. If we can identify the role of each layer, we can improve the model architecture by exploiting domain-specific knowledge, resulting in a more heterogeneous composition of layers. However, because end-to-end DNN performs as a black box, it is difficult to design and apply specific modifications for relevant layers.

Recently, a study suggested that the role of self-attention (SA) in Transformer-based ASR models can be distinguished into two types, phonetic and linguistic localization [10]. Two roles contribute to speech recognition in a row; the ASR system first extracts phonologically meaningful features by reducing the pronunciation variations and then combines such information into textual features to produce natural output sentences. These two-stage processes, which correspond to phonetic and linguistic localization, seem to be natural because ASR is a many-to-one problem in that multiple speeches can be transcribed as the same text. The study discovered that phonetic localization mainly appears in lower layers while linguistic localization happens in upper layers [10], and their attention patterns are also very different. The findings imply that we can identify layers of a certain role, and we may boost the performance by improving such layers to perform their role better.

Among the two types of roles mentioned above, we focus on improving phonetic localization based on a deeper understanding of the behavior. Here, we call SA heads that perform phonetic localization a phonetic (attention) head. From the observation of the attention weights produced by phonetic heads, we can separate two distinct types of attention patterns. The first type is similarity-based phonetic attention that gives a larger attention weight value on similarly pronounced frames. For example, frames corresponding to phoneme class 'S' often show large attention weight for frames corresponding to 'S', 'Z', 'SH', and vice versa [10]. The second type is content-based phonetic attention that attends to certain phonemes regardless of the query. In other words, a certain attention head may be highly optimized for detecting a specific phoneme class. We suggest that each phonetic head can be more specialized from the decomposition of similarity-based and content-based attention mechanisms.

In this paper, we propose phonetic self-attention (phSA), a variant of SA that extracts similarity and content-based phonetic features in phonetic localization. We modify the query-key dot product term inside the SA mechanism to capture similarity and content separately. In particular, we improve the dot product by (1) decomposing the two terms to remove shared parameters and (2) inserting trainable non-linearity functions. We evaluate the proposed phSA using phoneme classification and speech recognition and achieve considerably improved recognition performance on both tasks. In addition, we empirically show that similarity-based and content-based phonetic attention produce relatively concentrated and distributed attention probabilities, respectively.

2. Motivation

2.1. Dot Product in Self Attention

Self-attention (SA) is the key component of Transformer that computes the relationship between every pair of frames. For a sequence of speech frame features $X = \{x_1, x_2, ...x_T\}$ as an input, SA first projects features into three components, namely query (Q), key (K), and value (V). SA utilizes multiple attention heads with different parameters to capture diverse relationships in each layer. Without loss of generality, we explain the behavior of a single attention head below. Q, K, and V are linear projections of input as follows:

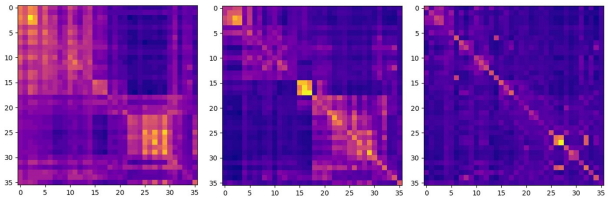
$$Q, K, V = XW_{\{Q,K,V\}} + b_{\{Q,K,V\}} \tag{1}$$

where $X \in \mathbb{R}^{T \times d_h}$, $W \in \mathbb{R}^{d_h \times d_h}$ and $b \in \mathbb{R}^{1 \times d_h}$ are input, weight, and bias, respectively. d_h is the dimension of each attention head. The attention map A is then calculated as:

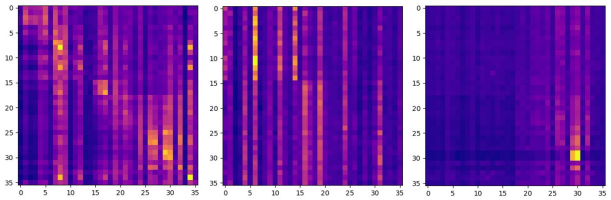
$$A = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_h}}\right) \in \mathbb{R}^{T \times T}. \tag{2}$$

Each element of the attention map represents how much one frame focuses on the other one, which is, in practice, imple-

10.21437/Interspeech.2022-422



(a) Similarity-based PAR



(b) Content-based PAR

Figure 1: *Visualization of PAR from selected attention heads in the baseline model. Two rows show representative examples of similarity-based and content-based phonetic attention, respectively. Brighter points indicate higher attention weight.*

mented as a dot product of the query and key. The dot product equation can be decomposed into four terms:

$$QK^T = XW_Q W_K^T X^T + XW_Q b_K^T + b_Q W_K^T X^T + b_Q b_K^T. \quad (3)$$

The first term ($\in \mathbb{R}^{T\times T}$) calculates the correlation between frames. The second term ($\in \mathbb{R}^{T\times 1}$) adds offset value per row, while the third term ($\in \mathbb{R}^{1\times T}$) adds offset per column. The fourth term is a constant. Because the dot product is followed by the row-wise softmax operation, the second and the fourth terms do not affect the output after softmax. In other words, the bias of K (b_K) can be safely removed, and then the dot-product can be simplified as follows:

$$QK^T = (XW_Q + b_Q)(XW_K)^T \quad (4)$$

$$= (XW_Q)(XW_K)^T + (XW_K b_Q^T)^T. \quad (5)$$

Please note that the second term of Eq. (5) had not been studied much compared to the first term.

2.2. Phonetic Behavior of Self Attention

The behavior of SA in Transformer-based ASR models has been analyzed in several previous works [7, 8, 10]. Recently, a study revealed the reason why SA is especially beneficial for ASR [10]. In a nutshell, SA in lower layers performs phonetic localization that extracts features based on phonological relationships through the whole sequence. This unique behavior is expected to improve the recognition performance by standardizing the various pronunciation of the same phoneme within the utterance. The findings on phonetic localization are supported by the phoneme attention relationship (PAR), a tool that visualizes the phonetic behavior of SA by converting frame-to-frame attention to phoneme class-to-class attention [10]. Specifically, the (i, j)-th element of PAR indicates how much attention weight (in average) is assigned from i-th phoneme class to j-th class. Please refer to the original paper for more details about PAR [10].

We investigate PAR of phonetic heads and find that such heads can be further separated into two groups. Figure 1 visualizes representative PAR examples. The first row focuses on the similarity of frames, characterized by symmetric PAR. For attention heads belonging to this type, the attention weight follows the correlation between phoneme classes of query and key. On the other hand, the second row focuses on the individual frame, represented as vertical lines in PAR. In this case, the attention weight highly depends on the phoneme class of key, and therefore might not be sufficiently represented by the query-key dot product. Note that individual attention heads cannot be clearly separated into two groups; the more accurate interpretation is that one head contains both tendencies with different portions. The original work on PAR also observed various PAR patterns of phonetic heads [10], however, did not much investigate this phenomenon.

3. Phonetic Self-Attention

3.1. Decomposition of Similarity and Content

We distinguish the two important phonetic behaviors by the dependency on other frames. The first one, *similarity-based* attention, focuses on the similarity between two frames. The second one, *content-based* attention, focuses more on the content of each frame. We connect these two different phonetic behaviors to two terms in Eq. (5). The attention weight $A[i, j]$ is determined by both the similarity between i, j-th frames and the content of j-th frame. These behaviors can be simultaneously performed with vanilla SA, where the original formulation does not clearly separate these two.

We first decompose two behaviors by modifying the dot product in SA. Specifically, in Eq. (5), we remove the effect of the first term on the second term by replacing the shared weight W_K with a separate parameter W_C:

$$XW_K b_Q^T \rightarrow \phi(XW_C)c^T, \quad (6)$$

where ϕ is the Swish [11] function and $c \in \mathbb{R}^{1\times d_h}$ is a bias parameter. We insert the non-linearity function ϕ to avoid two parameters (W_C and c^T) collapse.

3.2. Non-linear Activation Function

Next, we apply the PReLU [12] activation function so that the influence of each term can be controlled before adding the two. PReLU contains a single trainable parameter α that controls the tangent of the negative slope.

$$\psi_{s,c}(x) = \begin{cases} x & \text{if} \quad x \geq 0 \\ \alpha_{s,c} \cdot x & \text{otherwise,} \end{cases} \quad (7)$$

where ψ_s and ψ_c represent PReLU for similarity- and content-based terms, respectively. We initialize α to 1 for PReLU to behave like an identity function at the beginning of training.

The proposed *phonetic self-attention (phSA)* is the addition of two terms that correspond to two different phonetic behavior:

$$\psi_s((XW_Q)(XW_K)^T) + \psi_c(\phi(XW_C)c^T)^T. \quad (8)$$

The first and the second terms represent similarity-based and content-based phonetic attention, respectively. The proposed phSA is a direct drop-in replacement to the conventional dot product and is easy to implement.

3.3. Additional Design Choices

3.3.1. Remove Positional Encoding

The relative positional encoding (RPE) has been widely used for Transformer models for ASR [2, 3, 13, 14]. For exam-

Table 1: *Phoneme classification accuracy (%) of different dot product variants evaluated on LibriSpeech dataset. M2 is the dot product of the original self-attention, and M5 is the dot product of the proposed phonetic self-attention.*

Model	Dot-product	*dev-clean*	*dev-other*	*test-clean*	*test-other*
M1	$(XW_Q)(XW_K)^T$	81.92	73.42	81.86	73.63
M2	$(XW_Q)(XW_K)^T + (XW_K b_Q^T)^T$	81.84	73.37	81.79	73.55
M3	$(XW_Q)(XW_K)^T + (Xc^T)^T$	81.93	73.26	81.82	73.52
M4	$(XW_Q)(XW_K)^T + (\phi(XW_C)c^T)^T$	82.40	73.89	82.25	74.20
M5	$\psi_s((XW_Q)(XW_K)^T) + \psi_c(\phi(XW_C)c^T)^T$	**82.66**	**74.20**	**82.53**	**74.48**

ple, Conformer [3] exploits the same RPE implementation as Transformer-XL [15]. Although the previous study suggested that RPE may be unnecessary for large size ASR models [5], RPE helps small to medium size ASR models to better generalize to variable sequence lengths [16]. The downside of RPE is the heavy computation cost caused by additional query-position relationship computation and complex tensor operations to match the relative position. We decide not to use any positional information when using phSA; neither absolute nor relative PE is used. The design is based on the idea that the phonetic behavior of SA would consider each frame's phonetic characteristics, not necessarily the relative distance between frames. As a good side effect, the weight parameter for RPE is removed while W_C is added, so the number of parameters in phSA remains almost the same as in SA. We note that using RPE and phSA together may provide additional gain on performance at the expense of increased resource usage.

3.3.2. Replace in Lower Layers

We only replace the vanilla SA with phSA for the lower layers of the model, where phonetic localization is performed [10]. Because upper layers are known to be responsible for linguistic localization that combines the extracted phonetic information to generate text, we expect phSA may not be useful for those layers. From the experiments, we show that using phSA for the entire layers actually hurts the performance (see Sec. 4.3).

4. Experimental Results

4.1. Setup

We train our ASR models on the LibriSpeech-960 dataset [17]. For both phoneme classification and speech recognition experiments, we employ 80-dimensional log-Mel filterbank features as the input, extracted from a 25ms window with a 10ms stride. We employ 36 phoneme classes (including 'silence') for the phoneme classification as in [10]. For speech recognition, the subword vocabulary size is set to 128, built by SentencePiece [18] on the training data transcripts.

We choose the Conformer-M [3] as our baseline and train the model with CTC [19] loss. The baseline Conformer-M consists of 16 Conformer layers with RPE. We follow the training details from the previous work [10] for ASR. For the phoneme classification task, we stack 4 Conformer layers with the hidden dimension of 256. When replacing SA with phSA, we only modify the self-attention block inside the Conformer layer and preserve other blocks such as convolutional and feed-forward blocks. We set the learning rate to 1.56e-3 and weight decay to 1e-4 for the phoneme classification.

Table 2: *Word error rate (%) of different configurations of phonetic self-attention layers. The baseline performance (without phSA) is presented in the first row. The best results are in bold, and the second best results are underlined.*

#Layers		*dev-*		*test-*	
phSA	SA	*clean*	*other*	*clean*	*other*
0	16	3.10	8.23	3.25	8.21
4	12	**2.87**	8.11	<u>3.19</u>	**7.88**
6	10	<u>3.01</u>	**7.77**	**3.15**	<u>7.93</u>
8	8	3.05	<u>8.06</u>	<u>3.19</u>	8.06
12	4	3.08	8.36	3.30	8.30
16	0	3.58	9.55	3.81	9.51

4.2. Phoneme Classification

To evaluate the phonetic feature extraction performance, we train the models for phoneme classification. Table 1 compares the vanilla SA (M2), phSA (M5), and other variants. M2 is the original dot-product, and M1 is the same version without bias parameter that only focus on similarity-based relationships. M3 is identical to the M2 but differs in the implementation that the parameter W_K is not shared. M1, M2, and M3 show almost similar accuracy with less then 0.1% difference. In contrast, M4 shows a noticeable gain in phoneme classification accuracy compared to M2 and M3. The proposed phSA (M5) achieves the highest accuracy among the dot-product variants. The results verify that our architectural modifications, M2→M4 (Sec. 3.1) and M4→M5 (Sec. 3.2), each contributes to better phonetic feature extraction.

4.3. Speech Recognition

Table 2 shows the end-to-end speech recognition performance with the proposed phSA. Compared to the baseline, replacing the vanilla SA to phSA reduces the word error rate (WER) on every data subset, especially for the challenging LibriSpeech *dev-other* and *test-other* datasets. We empirically show that adopting phSA only for lower layers (under 8-th layer) achieves the best performance. This observation is aligned with previous analysis [10] that the lower layers more focus on phonetic information than upper layers. For example, replacing phSA for lower 6 layers decreases the WER from 8.23% to 7.77% (5.6% relative reduction) and 8.21% to 7.93% (3.4% relative reduction) for *dev-other* and *test-other* datasets, respectively. In contrast, utilizing phSA for 12 layers shows worse performance than the baseline, and using phSA for the entire (16) layers suffers from significant performance degradation.

Table 3: *Effect of similarity-based (S) and content-based (C) attention by removing each component. Entropy (mean ± std) of phSA attention maps and word error rate (%) are reported.*

S	C	Entropy	*dev-other*	*test-other*
✓	✓	1.91 ± 0.12	7.77	7.93
✓		2.02 ± 0.14	8.16 (+0.39)	8.54 (+0.61)
	✓	2.39 ± 0.04	9.20 (+1.43)	9.35 (+1.42)

4.4. Discussion

4.4.1. Speed and Parameter Size

Although we add several new computation steps for phSA, we observe that the training and inference time does not change much. The main reason is that the removal of RPE can compensate for the additional cost of phSA, in both latency and parameter size. For example, the wall-clock training time of the phSA (2^{nd} row in Table 2) is about 5% faster than the baseline (1^{st} row) with almost the same number of parameters.

4.4.2. Comparison of Similarity and Content

To understand the relative importance between similarity-based and content-based attention, we evaluate the recognition performance without each component. Table 3 presents the word error rate of the model using only similarity-related or content-related computation. Specifically, we fix the parameters of the converged model with 6 phSA layers and discard either term of the phSA dot product (Eq. (8)). Removing similarity-based attention (bottom row of Table 3) degrades the performance more than removing content-based one (top row of Table 3), which implies that the phonetic features extracted from similarity are more important than content-based attention; however, both are indispensable for speech recognition.

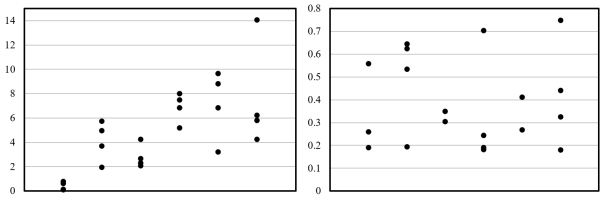
In addition, we calculate the average per-head entropy of attention probability for two settings and observe the meaningful difference. Similarity-based attention probabilities are more concentrated (lower entropy) and content-based attention probabilities are more distributed (higher entropy). In other words, the similarity-based term emphasizes the difference while the content-based term enhances the uniformness. We believe that the proposed phSA encourages two terms to be specialized for different attention patterns.

4.4.3. PReLU Negative Slope

The range of PReLU negative slopes is very different for similarity-based and content-based terms after training. Figure 2 shows the negative slopes $\alpha_{s,c}$ of each attention head. For similarity-based ones, most of α_s parameters are trained to become much larger than 1, implying that the negative correlation values are aggressively ignored and therefore produce a concentrated probability distribution. On the other hand, α_c parameters for content-based ones are trained to be smaller than 1, decreasing the difference between negative results. The combination of large α_s and small α_c is connected to different characteristics discussed in Sec. 4.4.2.

5. Related Work

Architectural modifications for Transformer-based ASR models have been of great interest. Many works focus on reducing the heavy computational cost caused by SA [20, 21, 22, 23]. For example, Efficient Conformer [20] proposed grouped SA and downsampling block to shorten the length of the sequence to be processed. Our work is very distinct from previous works in two points. First, phSA is designed to enhance the quality of intermediate feature representation, therefore improving the recognition performance. Second, only a lower part of the model is changed to phSA so that the model utilizes two different types of self-attention mechanisms together.



(a) Similarity-PReLU slope (b) Content-PReLU slope

Figure 2: *PReLU negative slopes (α_s and α_c) of the phSA after training. Four dots in each layer indicates PReLU parameters in four attention heads. x-axis indicates the layer index. Note that the range of y-axis is very different, $(0 \sim 14)$ for (a) and $(0 \sim 0.8)$ for (b).*

Pretraining-based approaches have been proven effective in improving the ability to capture useful phonetic information for various downstream tasks. For example, Wav2Vec2.0 [24], XLSR [25], TERA [26], and ACPC [27] presented various self-supervised speech pretraining methods and showed that phonologically meaningful features can be captured while learning the general characteristics of speech. However, these models use identical Transformer architecture for every layer without considering the different behaviors of each. Explicit pretraining objectives have also been introduced for learning the useful phonetic features during pretraining. For example, UniSpeech [28] and BERTphone [29] exploited CTC loss using phoneme sequence as label. The drawback of the abovementioned studies is that they require an additional pretraining stage before fine-tuning the model for ASR.

6. Conclusion

In this paper, we proposed a variant of self-attention (SA), named phonetic self-attention (phSA), to improve the ASR performance. Especially, we investigated the phonetic behavior of attention heads and distinguished two different attention patterns, similarity-based and content-based attention. The proposed phSA emphasized the two behaviors by applying simple and effective modifications to the original dot-product in SA. In addition, the effect of each behavior is controlled by additional trainable parameters. From the phoneme classification experiments, we showed that phSA is more suitable than the vanilla SA for phonetic feature extraction. By replacing SA in lower layers with phSA, we improved the speech recognition performance on the end-to-end Transformer-based ASR model.

7. Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by Korea government (MSIT) (No. 2021R1A2C1013513). This work was also supported in part by Samsung Advanced Institute of Technology, Samsung Electronics Co., Ltd.

8. References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[2] Q. Zhang, H. Lu, H. Sak, A. Tripathi, E. McDermott, S. Koo, and S. Kumar, "Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7829–7833.

[3] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. Interspeech 2020*, 2020, pp. 5036–5040.

[4] E. G. Ng, C.-C. Chiu, Y. Zhang, and W. Chan, "Pushing the limits of non-autoregressive speech recognition," *arXiv preprint arXiv:2104.03416*, 2021.

[5] Y. Zhang, J. Qin, D. S. Park, W. Han, C.-C. Chiu, R. Pang, Q. V. Le, and Y. Wu, "Pushing the limits of semi-supervised learning for automatic speech recognition," *arXiv preprint arXiv:2010.10504*, 2020.

[6] W. Chan, D. Park, C. Lee, Y. Zhang, Q. Le, and M. Norouzi, "Speechstew: Simply mix all available speech recognition data to train one large neural network," *arXiv preprint arXiv:2104.02133*, 2021.

[7] S. Yang, A. T. Liu, and H. yi Lee, "Understanding self-attention of self-supervised audio transformers," in *Proc. Interspeech 2020*, 2020, pp. 3785–3789.

[8] S. Zhang, E. Loweimi, P. Bell, and S. Renals, "On the usefulness of self-attention for automatic speech recognition with transformers," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 89–96.

[9] ——, "Stochastic attention head removal: A simple and effective method for improving transformer based asr models," *arXiv preprint arXiv:2011.04004*, 2020.

[10] K. Shim, J. Choi, and W. Sung, "Understanding the role of self attention for efficient speech recognition," in *International Conference on Learning Representations*, 2022.

[11] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Networks*, vol. 107, pp. 3–11, 2018.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[13] N.-Q. Pham, T.-L. Ha, T.-N. Nguyen, T.-S. Nguyen, E. Salesky, S. Stüker, J. Niehues, and A. Waibel, "Relative Positional Encoding for Speech Recognition and Direct Translation," in *Proc. Interspeech 2020*, 2020, pp. 31–35.

[14] T. Likhomanenko, Q. Xu, G. Synnaeve, R. Collobert, and A. Rogozhnikov, "Cape: Encoding relative positions with continuous augmented positional embeddings," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[15] Z. Dai, Z. Yang, Y. Yang, J. G. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 2978–2988.

[16] J. Park, C. Kim, and W. Sung, "Convolution-based attention model with positional encoding for streaming speech recognition on embedded devices," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 30–37.

[17] 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)*, 2022, pp. 5206–5210.

[18] T. Kudo and J. Richardson, "Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2018, pp. 66–71.

[19] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[20] M. Burchi and V. Vielzeuf, "Efficient conformer: Progressive downsampling and grouped attention for automatic speech recognition," *arXiv preprint arXiv:2109.01163*, 2021.

[21] X. Wang, S. Sun, L. Xie, and L. Ma, "Efficient Conformer with Prob-Sparse Attention Mechanism for End-to-End Speech Recognition," in *Proc. Interspeech 2021*, 2021, pp. 4578–4582.

[22] H. Luo, S. Zhang, M. Lei, and L. Xie, "Simplified self-attention for transformer-based end-to-end speech recognition," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 75–81.

[23] M. Xu, S. Li, and X.-L. Zhang, "Transformer-based end-to-end speech recognition with local dense synthesizer attention," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5899–5903.

[24] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020.

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

[26] 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.

[27] J. Chorowski, G. Ciesielski, J. Dzikowski, A. Łańcucki, R. Marxer, M. Opala, P. Pusz, P. Rychlikowski, and M. Stypułkowski, "Aligned Contrastive Predictive Coding," in *Proc. Interspeech 2021*, 2021, pp. 976–980.

[28] C. Wang, Y. Wu, Y. Qian, K. Kumatani, S. Liu, F. Wei, M. Zeng, and X. Huang, "Unispeech: Unified speech representation learning with labeled and unlabeled data," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10 937–10 947.

[29] S. Ling, J. Salazar, Y. Liu, K. Kirchhoff, and A. Amazon, "Bertphone: Phonetically-aware encoder representations for utterance-level speaker and language recognition," in *Proc. Odyssey 2020 the speaker and language recognition workshop*, 2020, pp. 9–16.