INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Exploring Downstream Transfer of Self-Supervised Features for Speech Emotion Recognition

Yuanbo Fang[1], *Xiaofen Xing*[1,*], *Xiangmin Xu*[1,2], *Weibin Zhang*[3]

[1]South China University of Technology, China
[2]Pazhou lab, China
[3]VoiceAI Technologies, Shenzhen, China

eeybfang@mail.scut.edu.cn, {xfxing,xmxu}@scut.edu.cn, eeweibin@gmail.com

Abstract

Huge progress has been made in self-supervised audio representation learning recently, and transformer based downstream model using Multi-head Self-Attention and Feed-Forward Network (MSA-FFN) as the basic block delivered promising transfer performance on downstream speech tasks. However, it is unclear whether the traditional transformer architecture is appropriate for downstream transfer. In this paper, we adopt a block architecture search strategy (BAS) to explore this issue, taking speech emotion recognition as an example. We found that 1) it is crucial to incorporate an FFN-like representation learning module without MSA design in the early stages of the downstream model; 2) with the use of self-supervised features, it is good enough to use a simple FFN for the downstream task. This work can serve as a source of inspiration for all other downstream speech tasks that utilize self-supervised features.

Index Terms: self-supervised features, downstream transfer, speech emotion recognition, block architecture search

1. Introduction

Self-Supervised Learning (SSL) is a form of unsupervised learning that allows the network to learn universal representations from a large amount of unlabeled data. SSL can extract more salient and robust features that is useful in improving the performance of downstream tasks [1, 2]. The generalization ability of representations extracted by using a pre-trained model help decrease the urgency of searching for hand-crafted, engineered features [3]. Thus, many self-supervised pre-trained models have been developed for natural language processing and computer vision, for example, BERT [4] and MAE [5]. Recently, SSL has also achieved impressive success in the field of audio and speech processing[3]. Excellent self-supervised pre-trained models such as Wav2vec [6], Hubert [7] and WavLM [8] have emerged.

In the field of Speech Emotion Recognition (SER), due to the complexity of human emotion, as far as we know, currently there is no such a hand-crafted feature type that is widely accepted as an effective emotion representation. Some researchers generally adopt traditional features such as fundamental frequency [9], MFCC [10] and speech spectrogram [11] for SER. Given the success of self-supervised learning, researchers have also tried to use self-supervised features for SER, which has demonstrated superior performance to traditional features [12, 13]. The Superb [14] shows that various frozen self-supervised encoders achieve high performance on SER, the WavLM large model stood out with an accuracy of 70.62% on the IEMOCAP dataset [15]. Fine-tuning the pre-trained model is a common approach to leverage self-supervised learning for downstream tasks. Wang et al. [16] conducted experiments on fine-tuning the transformer encoder part of a SSL model, comparing entire fine-tuning with partial fine-tuning. The results showed that partial fine-tuning outperformed entire fine-tuning to some extent. The authors suggested that entire fine-tuning may lead to overfitting due to the limited amount of data available for fine-tuning. Although fine-tuning upstream self-supervised models can improve performance on downstream tasks, it requires a large amount of downstream training data and is computationally expensive. Therefore, this paper focuses on exploring how to better perform downstream transfer when the parameters of the self-supervised model are fixed.

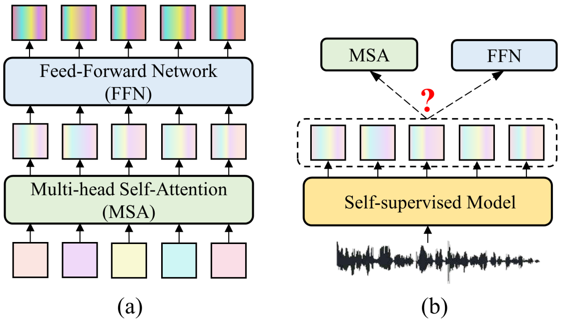

Figure 1: ***The difference between two ways to extract features.*** *(a) represents the principle of feature extraction in traditional transformer. (b) represents the uncertainty of feature extraction using a self-supervised model.*

Downstream transfer via a downstream model is an effective approach to enhance the adaptability of self-supervised features to downstream tasks. Pepino et.al. [12] used pointwise convolutional layer and LSTM as downstream model for SER. Li et al. [17] proved the advantages of using the transformer model as a downstream model for speech, text and multimodal emotion recognition when using self-supervised features. Chen et al. [18] proposed a hierarchical transformer with neighboring attention with self-supervised features and achieved high performance. The components of each basic block of traditional transformer are ordered as Multi-head Self-Attention (MSA) is applied first, followed by Feed-Forward Network (FFN) processing. As shown in Fig. 1(a), MSA can enable each token in the sequence to interact information with tokens in other locations to generate new token vectors. By using FFN after MSA, non-linear transformation can be performed on each token with global interaction information to further extract task-related information. While the self-supervised model can extract general representations with global information through unsupervised pre-training (i.e., as shown in Fig. 1(b), it remains unclear whether the pre-defined MSA-first setting is suitable for

[*] Xiaofen Xing is the corresponding author.

10.21437/Interspeech.2023-653

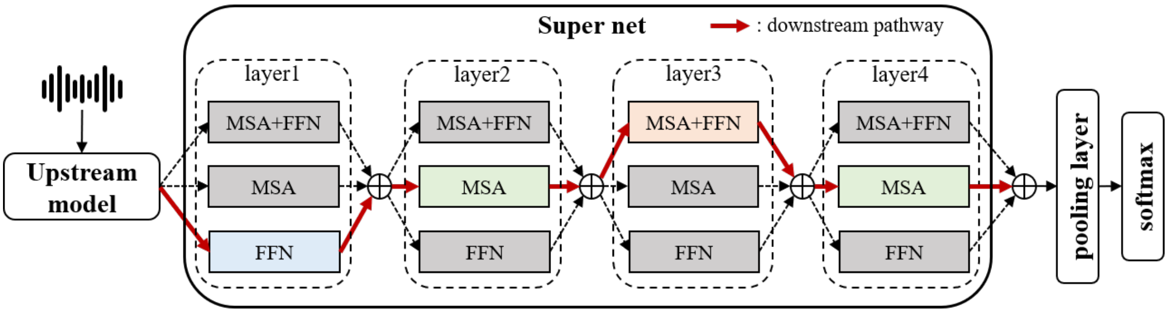

Figure 2: ***An overview of the BAS framework.*** *BAS explores a layer-wise search space that each layer of the super net can choose a different block from three candidate blocks. The network parameters and the sampling parameters are trained together. The downstream model architecture is then sampled from the trained super net. The red arrow flow denotes an example sampled architecture.*

the downstream model. Thus, considering that each token of self-supervised features already contains context information, an important question arises: should each token be first mapped to the task-specific representation space and then exchange task-related information?

To tackle the above problem, we adopt Block Architecture Search (BAS) strategy to explore suitable representation transfer way for downstream task (i.e., SER in this work) within a finite searching space based on transformer encoder. We find that the MSA-first design, which is the case in a traditional transformer model, is not optimal when self-supervised pre-trained features are used. Perhaps representation transfer learning should be performed as early as possible in the downstream model. This study may inspire future research on various downstream tasks based on self-supervised features. The main contributions of this paper can be summarized as follows:

- We demonstrate that the block architecture search strategy is effective to search suitable downstream transfer way based on self-supervised pre-trained features. Though only the SER task is studied in this paper, this strategy can be easily adopted for other tasks.
- It is found that putting a task-specific representation projection at the first place is the key to downstream model. This can also inspire other downstream speech processing tasks.

2. Methodology

To explore a suitable task-specific downstream transfer way based on transformer architecture, the basic building block of the transformer is further decomposed into a MSA sub-module and a FFN sub-module, which serve different functions. We employ the block architecture search strategy to choose the suitable architecture for each layer block in downstream model, providing the flexibility to remove either the MSA or FFN sub-module for downstream transfer. A conceptual overview of the proposed method can be found in Fig. 3. Details about the methodology will be elaborated below.

2.1. Preliminaries

Transformer models have achieved superior speech emotion classification performance. The building block of each layer in a vanilla transformer [19] consists of a MSA sub-module and a FFN sub-module. Formally, for an input speech sequence $\mathcal{X} = [x_1, x_2, ..., x_N]$ where N is the number of speech frames, and $x_i \in \mathbb{R}^D$ where D is the dimension of the features, a single-head self attention is computed as follows:

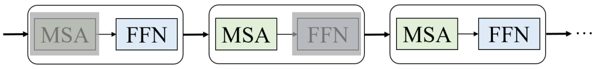

Figure 3: ***A conceptual overview of our method.*** *A sub-module may be completely removed (e.g. shadowed blocks) from the downstream model by using the BAS strategy.*

$$\text{head}_h = \text{Attn}(Q_h, K_h, V_h) = \text{softmax}(\frac{Q_h K_h^{\text{T}}}{\sqrt{d_h}})V \quad (1)$$

Where Q_h, K_h and V_h are usually called the query, key and value respectively and they are achieved by projecting the inputs, i.e., $Q_h = XW^{Q_h}$, $K_h = XW^{K_h}$, and $V_h = XW^{V_h}$, where $W^{Q_h}, W^{K_h}, W^{V_h} \in \mathbb{R}^{D \times d_h}$. The output of MSA is computed by concatenating the outputs of single-head self attentions with different projection parameters:

$$\text{MSA}(X) = \text{concat}(\text{head}_1, ..., \text{head}_H) \quad (2)$$

Then the outputs of a MSA are fed into a FFN, usually with two-layers. Skip connections are used to bypass the MSA or the FFN. In all, each layer in a vanilla transformer performs the following computation:

$$\text{layer}_l = \text{FFN}(\text{MSA}(X) + X) + \text{MSA}(X) + X \quad (3)$$

2.2. Block Architecture Search Strategy

Designing effective network for the downstream SER task is challenging since the design space may be very large. To address this problem, we construct a layer-wise search space with a pre-defined fixed macro-architecture which is similar to [20]. In this paper, BAS explores a layer-wise space that each layer of a transformer can choose a different block. The search process trains the stochastic super net using SGD to optimize the architecture distribution. The downstream network architecture are sampled from the trained distribution under the limited search space. Fig. 2 provides an overview of the whole strategy. Each step will be briefly described below. We refer the reader to [20] for more detailed information.

Search Space. The macro-architecture for the transformer-based downstream model is pre-defined. Each searchable layer can choose a different block from the layer-wisesearch space. As shown in Fig. 2, inspired by the basic building block of a transformer, the layer-wise search space consists of 3 candidate blocks, i.e., an original MSA+FNN, an MSA, and a FFN without residual connection. Since this paper is aiming at exploring the influence of block function on the speech emotion representations transfer based on self-supervised features, the network parameters such as the number of neurons and the number of MSA heads are fixed, and skip connections are not set between layers. In summary, the BAS framework in this paper is based on transformer encoder to explore suitable downstream models. When the number of layers of the supernet is set to 4, each searchable layer can choose from 3 candidate blocks, and it contains $3^4 = 81$ possible architectures.

Search Algorithm. It is computationally expensive to find the suitable block architecture through brute-force enumeration of the search space. As in [20], the whole search space is represented by a stochastic super net which has the same macro-architecture as described, and each searchable layer contains all the three block candidates in parallel as shown in Fig. 2. Only one of the candidate block is sampled and executed during the inference. To make the loss function described in the following subsection differentiable with respect to the network weights, as well as to the sampling parameters, the sampling process is relaxed through the Gumbel softmax trick [20], i.e.

$$m_{l,i} = \frac{\exp[(\theta_{l,i} + g_{l,i})/\tau]}{\sum_i \exp[(\theta_{l,o'} + g_{l,o'})/\tau]} \tag{4}$$

where $m_{l,i}$ represents the probability of sampling the i^{th} block in the l^{th} layer, and $g_{l,i} \sim \text{Gumbel}(0,1)$ is a random noise following the Gumbel distribution, $\theta_{l,i}$ denotes the sampling parameter for i^{th} block at l^{th} layer and τ is a temperature parameter. Gumbel softmax introduces randomness so that each candidate block can be selected with a certain probability. The output of layer-l can be expressed as:

$$x_{l+1} = \sum_i m_{l,i} \cdot f_{l,i}(x_{l-1}) \tag{5}$$

Where $f_{l,i}(\cdot)$ is the block operation corresponding to $m_{l,i}$. In this way, the target loss is directly differentiable to the sampling parameters $\theta_{l,i}$. After the super net training is completed, we can obtain the suitable downstream model architecture by sampling a candidate block operation with the maximum probability from the block distribution of each layer.

2.3. Loss Function

By using BAS, the super net is trained in the same way as a normal neural network and can quickly estimate the performance of all block architectures in the search space. Since we seek to find an suitable emotion recognition network architecture based on self supervised features for downstream transfer, the objective function of BAS aims to improve the classification accuracy. Therefore, cross-entropy loss is used to optimize the super net. Define the ground-truth as y and the final prediction of the super net $F(x)$, then the cross-entropy loss is computed as follows:

$$\text{loss}_{\text{ce}} = -\sum_i y_i \log(F(x_i)) \tag{6}$$

3. Experiments

3.1. Datasets

The following datasets were used in our experiments.

IEMOCAP [15] is one of the most popular dataset for speech emotion recognition. As with other researchers [13, 18, 21], the subset of IEMOCAP, which contains 5531 utterances of angry, happy (the category excited is re-labeled as happy), sad, and neutral was used. Experiments are conducted in a leave-one-session-out cross-validation strategy.

MELD [22] is another dataset used for SER. It consists of 13,708 utterances with seven emotions. The dataset is splitted into the train, validation and test sets, and the scores on the test set are reported.

3.2. Self-supervised Features

Hubert is a SSL approach for speech processing that assigns a masked segment of input speech to a pseudo-label provided by applying K-means clustering. Hubert learns a combined acoustic and language model over the continuous inputs by applying a prediction loss on the masked regions only.

WavLM is another large-scale SSL model trained with 94k hours of audio. It aims to solve full stack speech processing tasks. WavLM jointly learns masked speech prediction and denoising in pre-training, enabling the pre-trained model to perform well on both automatic speech recognition (ASR) and non-ASR tasks.

3.3. Experimental Setup

In our experiments, the self-supervised pre-trained upstream models HuBERT-Large and WavLM-Large were used as feature extractors. To optimize the parameters of the constructed super net in BAS, we trained the network for 100 epochs using the SGD optimizer with the learning rate of 0.005, and the batch size was set to 32. For the evaluation metrics, we chose the widely used weighted accuracy (WA) and unweighted accuracy (UA) for IEMOCAP, and weighted average F1 (WF1) for MELD.

3.4. Experiment Results and Analysis

3.4.1. Block architecture search results

To reduce the influence of the number of downstream model layers on BAS results, we set the number of layers in the supernet to 3 and 4, respectively, and conducted experiments on the IEMOCAP and MELD datasets. The search results using BAS with Hubert-large and WavLM-large are shown in Fig. 4.a and Fig. 4.b, respectively.

It can be observed that using self-supervised features as input of downstream model, the first layer of downstream block architecture found in all conditions is always a FFN module, while other layers always contain the MSA module. According to the experimental results, we believe that since self-supervised audio features already contain global interaction information, and encoding the sequence and capturing interrelationships between sequences prior to the MSA module could be considered redundant. In addition, the MSA module is still included in the later layers, indicating that the MSA module is still indispensable. For SER, we analyze that the self-supervised features undergo initial processing through the FFN module, which enables each token to extract emotion-related information. This process facilitates the transfer of the universal representation space into a emotion-specific representation space. Subsequently, the MSA module can perform global modeling of all tokens in the emotion-specific representation space, thereby avoiding redundancy associated with emotion-independent global modeling in the universal representation space.

3.4.2. Ablation experiments

As the candidates building blocks are decomposed from transformers, we selected traditional transformer models with 4 layers and 3 layers respectively as baselines for downstream transfer. According to the block architecture searching results of downstream model in Fig. 4, we know the importance of the FFN module. It is interesting to see the performance of a simple FFN (i.e. a 2-layer multi-layer perceptron). Thus a simple FFN network is also used for comparison. The baseline and BAS

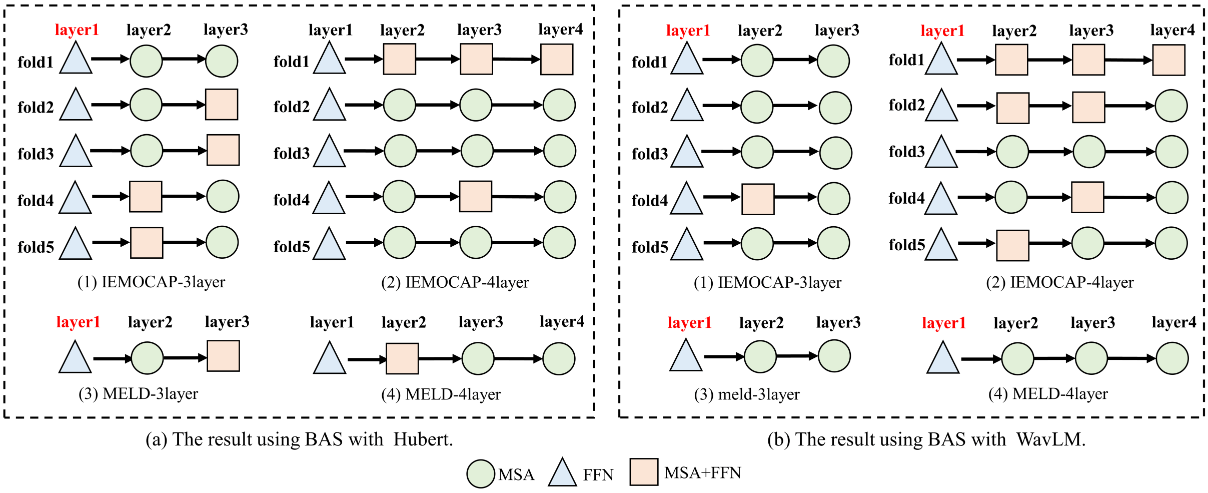

Figure 4: ***The downstream model obtained by using the BAS method.*** *(a) and (b) show the downstream architectures of each layer obtained by using the BAS method with Hubert large model and WavLM large model, respectively.*

Table 1: *The performances of different architectures with self-supervised features on IEMOCAP, MELD.*

Features	Architectures	IEMOCAP		MELD
		WA(%)	UA(%)	WF1(%)
Hubert	3L-baseline	68.95	70.61	39.12
	4L-baseline	68.16	69.61	44.96
	simple-FFN	71.57	72.73	44.80
	3L-BAS	72.24	73.19	45.08
	4L-BAS	**72.35**	**73.35**	**45.24**
WavLM	3L-baseline	70.74	72.09	45.09
	4L-baseline	70.08	71.65	45.75
	simple-FFN	72.53	73.70	46.25
	3L-BAS	73.92	74.22	46.18
	4L-BAS	**74.03**	**74.95**	**47.18**

results with different number of layers settings are presented in Table 1. The results indicate that the FFN-first downstream model uncovered by BAS outperforms the original transformer in terms of downstream transfer learning performance, and the simple FFN as downstream model even overcomes the transformer when evaluated on the IEMOCAP dataset. On MELD, the simple FFN also achieves comparable performance with the transformer. The results illustrate that an FFN-like epresentation transfer module works well for downstream transfer, and the number of parameters of the FFN is much less than that of the transformer and thus the FFN requires much less computation. The 4-layer downstream model obtained by using the BAS achieves 74.03% WA and 74.95% UA on IEMOCAP, and 47.18% WF1 on MELD, which indicate that BAS produces the best results under different settings.

3.4.3. Comparison to previous methods

Table 2 shows the comparison results between the BAS method and previous speech emotion recognition systems on the two datasets. As can be seen, BAS-selected model achieves state-of-the-art performance. Specifically, on IEMOCAP, WA is 4.23% better than previous results and UA improvement is 3.9%. When evaluated on MELD, BAS-selected model also obtains SOTA result of 47.18% WF1. These results demonstrate the effectiveness of the method.

Table 2: *Results of the BAS and previous published models on both datasets.*

Dataset	Methods	WA(%)	UA(%)
IEMOCAP	[Guo et al.,2021] [21]	61.32	60.43
	[Chen et al.,2022] [18]	62.90	64.50
	[Li et al.,2022] [17]	67.99	68.24
	[Zou et al.,2022] [13]	69.80	71.05
	BAS+Hubert	72.35	73.35
	BAS+WavLM	**74.03**	**74.95**

Dataset	Methods	WF1(%)
MELD	[Lian et al.,2021] [23]	38.20
	[Vishal et al.,2022] [24]	39.63
	[Chen et al.,2022] [18]	41.90
	[Hu et al.,2022] [25]	42.72
	BAS+Hubert	45.24
	BAS+WavLM	**47.18**

4. Conclusion

Our exploration has demonstrated that it is important to design FFN-like representation transfer function module projecting the task-independent self-supervised features into a task-specific space at the beginning of the downstream transfer process. In previous work, researchers generally focus on module enhancement, while we explore how to perform suitable downstream transfer for self-supervised features. When evaluated in the field of SER, models constructed by using BAS achieves new SOTA results. We hope this methodology will be helpful for the model design of other downstream tasks such as speaker recognition.

5. Acknowledgement

The work is supported in part by the National Natural Science Foundation of China U1801262; in part by the Fundamental Research Funds for the Central Universities 2022ZYGXZR0075; in part by Key-Area Research and Development Program of Guangdong Province 2022B0101010003; in part by the Guangdong Provincial Key Laboratory of Human Digital Twin 2022B1212010004; in part by the Science and Technology Project of Guangzhou 202103010002; in part by Natural Science Foundation of Guangdong Province 2022A1515011588; in part by Nansha key project 2022ZD011.

6. References

[1] L. Jing and Y. Tian, “Self-supervised visual feature learning with deep neural networks: A survey,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 11, pp. 4037–4058, 2020.

[2] L. Ericsson, H. Gouk, and T. M. Hospedales, “How well do self-supervised models transfer?” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5414–5423.

[3] S. Liu, A. Mallol-Ragolta, E. Parada-Cabaleiro, K. Qian, X. Jing, A. Kathan, B. Hu, and B. W. Schuller, “Audio self-supervised learning: A survey,” *Patterns*, vol. 3, no. 12, p. 100616, 2022.

[4] 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.

[5] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, “Masked autoencoders are scalable vision learners,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 000–16 009.

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

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

[8] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, “Wavlm: Large-scale self-supervised pre-training for full stack speech processing,” *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[9] C. Busso, S. Lee, and S. Narayanan, “Analysis of emotionally salient aspects of fundamental frequency for emotion detection,” *IEEE transactions on audio, speech, and language processing*, vol. 17, no. 4, pp. 582–596, 2009.

[10] J. Ancilin and A. Milton, “Improved speech emotion recognition with mel frequency magnitude coefficient,” *Applied Acoustics*, vol. 179, p. 108046, 2021.

[11] A. Satt, S. Rozenberg, and R. Hoory, “Efficient emotion recognition from speech using deep learning on spectrograms,” in *Interspeech*, 2017, pp. 1089–1093.

[12] L. Pepino, P. Riera, and L. Ferrer, “Emotion Recognition from Speech Using wav2vec 2.0 Embeddings,” in *Interspeech*, 2021, pp. 3400–3404.

[13] H. Zou, Y. Si, C. Chen, D. Rajan, and E. S. Chng, “Speech emotion recognition with co-attention based multi-level acoustic information,” in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7367–7371.

[14] S. wen Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, T.-H. Huang, W.-C. Tseng, K. tik Lee, D.-R. Liu, Z. Huang, S. Dong, S.-W. Li, S. Watanabe, A. Mohamed, and H. yi Lee, “SUPERB: Speech Processing Universal PERformance Benchmark,” in *Interspeech*, 2021, pp. 1194–1198.

[15] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, “IEMOCAP: Interactive emotional dyadic motion capture database,” *Language resources and evaluation*, vol. 42, no. 4, pp. 335–359, 2008.

[16] Y. Wang, A. Boumadane, and A. Heba, “A fine-tuned wav2vec 2.0/hubert benchmark for speech emotion recognition, speaker verification and spoken language understanding,” *arXiv preprint arXiv:2111.02735*, 2021.

[17] J. Li, S. Wang, Y. Chao, X. Liu, and H. Meng, “Context-aware multimodal fusion for emotion recognition,” in *Interspeech*, 2022, pp. 2013–2017.

[18] W. Chen, X. Xing, X. Xu, J. Pang, and L. Du, “Speechformer: A hierarchical efficient framework incorporating the characteristics of speech,” in *Interspeech*, 2022, pp. 346–350.

[19] 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.

[20] B. Wu, X. Dai, P. Zhang, Y. Wang, F. Sun, Y. Wu, Y. Tian, P. Vajda, Y. Jia, and K. Keutzer, “Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 734–10 742.

[21] L. Guo, L. Wang, C. Xu, J. Dang, E. S. Chng, and H. Li, “Representation learning with spectro-temporal-channel attention for speech emotion recognition,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6304–6308.

[22] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, and R. Mihalcea, “Meld: A multimodal multi-party dataset for emotion recognition in conversations,” *arXiv preprint arXiv:1810.02508*, 2018.

[23] Z. Lian, B. Liu, and J. Tao, “Ctnet: Conversational transformer network for emotion recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 985–1000, 2021.

[24] V. Chudasama, P. Kar, A. Gudmalwar, N. Shah, P. Wasnik, and N. Onoe, “M2fnet: Multi-modal fusion network for emotion recognition in conversation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4652–4661.

[25] D. Hu, X. Hou, L. Wei, L. Jiang, and Y. Mo, “Mm-dfn: Multimodal dynamic fusion network for emotion recognition in conversations,” in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7037–7041.