LIGHT GATED MULTI MINI-PATCH EXTRACTOR FOR AUDIO CLASSIFICATION

Bo He[⋆]*, Shiqi Zhang*[⋆]*, Xianrui Wang*[⋆]*, Zheng Qiu*[⋆]*,*
Daiki Takeuchi[†]*, Daisuke Niizumi*[†]*, Noboru Harada*[†]*, Shoji Makino*[⋆]

[⋆] Waseda University, Japan
[†] NTT Corporation, Japan

ABSTRACT

Audio classification, which serves as a fundamental step for acoustic signal processing, has attacked a lot of research interest and numerous audio classification neural networks have been proposed. In these networks, down-sampling blocks which compresses audio features are essential due to the computational capacity. However, compressing the signal will inevitably cause the loss of relevant information. To mitigate this issue, large amount of parameters are used. In this paper, we present a novel down-sampling method called gated multi mini-patch extractor (GMME), in which multiple convolutive layers are used to extract relevant information at different levels, including time frames, pseudo-frequency bins, and global features. And gate mechanism is adopted to retain the correlation with the original features. Several simulations demonstrate that, compared to the baseline, our method can achieve comparable or slightly better performance with significant reduction of number of parameters.

***Index Terms*—** Audio classification, feature extraction, down-sampling blocks, gated multi mini-patch extractor

1. INTRODUCTION

Audio events, to name a few, animal sounds, speech and music, are fundamental parts of our daily lives. As there are various types of audio events, audio classification should be carried out before other tasks, such as enhancement, separation, label indexing, and segmentation. Traditional methods use Gaussian mixture model (GMM) [1] or hidden Markov model (HMM) [2, 3] as classifiers, in which time-frequency representations, e.g., short-time fourier transformation (STFT) spectrogram and Mel spectrogram are used as input features. Limited by computational capacity and variety of datasets, the accuracy of these methods is far from satisfying.

With the booming of deep neural networks (DNNs), numerous audio event classification networks have been proposed. These models, especially those based on the convolutional neural network (CNN) [4, 5], recurrent neural network (RNN) [6], and transformer architectures [7], have achieved remarkable superiority over the aforementioned machine learning (ML) methods. Inspired by the great achievement of computer vision (CV) field, a popular kind of audio classification methods which regard spectrogram of audio signal as an image are proposed [8]. In these methods, various DNNs are implemented to extract features from these spectrograms. And then fully connected layer is carried out for final judgment. It is claimed that the Mel spectrogram is the most suitable time-frequency representation [9, 10] for such networks. The main problem of such methods is using a fixed time-frequency representation which might not be able to extract time-frequency structures of different kinds of audio. Besides, these methods have to deal with the raw audio with lots of data augmentation.

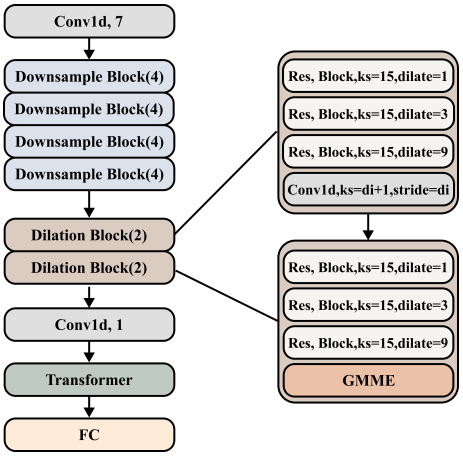

Fig. 1: Structure of EAT and replace for GMME

To addreass this problems, some end-to-end systems have been proposed [11–13]. These systems generally consists of three major parts. Firstly, a CNN-based encoder is implemented to extract time-domain features. This transformation can significantly improve feature extraction accuracy with some specific techniques [14]. The second part entails down-sampling which aims at shortening the duration of time frames and consequently, reducing the dimension of features from the first stage. Finally, entirely connected layers are carried out for classification. Although end-to-end methods are theoretically more flexible and able to achieve better performance, there were still some problems like a large number of parameters or insufficient generalization. To deal with these problems, the end-to-end audio transformer (EAT) [15] which employed extra data augmentation and a well-designed

This work was supported by JSPS KAKENHI Grant Number 23H03423.

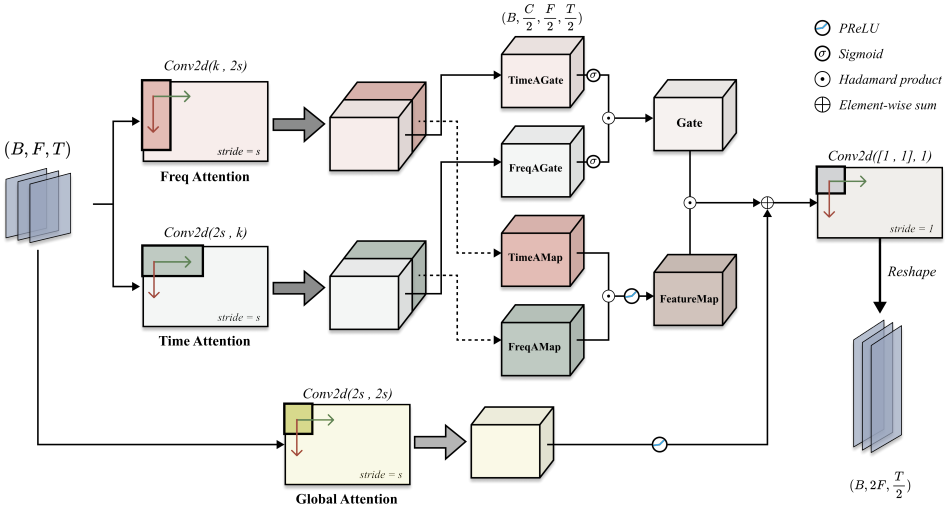

Fig. 2: Detailed structure of Gated Multi Mini-patch Extractor (GMME).

structure has been proposed and become the best-performing model in this field.

For both aforementioned two approaches, a sufficient time domain pooling is essential due to following two reasons. On the one hand, an insufficient pooling will lead to biased statistical estimates due to the natural non-stationarity of audio signals, on the other hand, it cannot capture the periodic features of audio signals, and consequently lose some key information. To mitigate these concerns, we presented a simplistic method named gated multi mini-patch extractor (GMME) for down-sampling time frames that allows for the interdependence of both temporal and frequency domain data with much fewer parameters compared to the state-of-the-art (SOTA) technique. EAT was considered as the backbone network architecture for our study, wherein we trained and assessed our proposed approach using the ESC-50 [16], Speech Command [17] and UrbanSound8K [18] datasets as benchmarks. The mentioned datasets were used to evaluate the efficacy of our proposed method. Simulations validate that the proposed method successfully retains classification features while decreasing over 98% of the parameters compared to the original anti-alias down-sampling block.

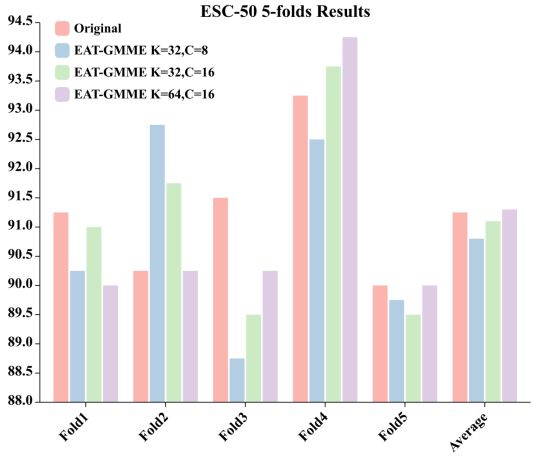

Fig. 3: Result in ESC-50 dataset 5-folds

2. RELATED WORKS

2.1. Transformer

Transformer [7] model has exhibited exceptional proficiency in natural language processing (NLP) [7] and CV [19]. The primary application of this technique involves extracting and converting relevant features with the manipulation of sequential data through a series of multi-head self-attention mechanisms. Owing to its ability to extract long-term attention, transformer has also been implemented in audio classification. The audio spectrogram transformer (AST) [20] is a pure-transformer model based on the vision transformer (ViT) [19] architecture and achieved remarkable performance. Some similar networks such as M2M-AST [21], HTS-AT [22], Beats [23] were also proposed.

Although transformer-based models have achieved promising performance, they still face the limitation of disregarding local priori knowledge. Consequently, these models frequently possess substantial parameters and necessitate prolonged training periods. Furthermore, real-world tasks are constrained by training datasets without the same data distribution, requiring models to be more robust. In order to solve these problems, researchers have leveraged the flexibility and efficacy of CNN and integrated them with transformer [6], and the SOTA work is EAT [15]. Owing to its implementation of efficient data augmentation techniques, the EAT model exhibits superior performance even when data is scarce.

Table 1: Result in ESC-50 and UrbanSound8K, K and C is respective Kernel size and Channel number. #D-Param is the parameter number of Downsample block, while #T-Param is represent Total model's parameter.

Model	Set	#D-Param(D)[K]	#T-Param(T)[M]	Accurancy(%)	
EAT-S (conventional)		1961.1	5.18	ESC-50	Urban8k
				91.25	**84.11**
EAT-GMME (proposed)	$K = 32, C = 8$	**17.4**	**3.23**	90.80	83.64
	$K = 32, C = 16$	**34.9**	**3.25**	91.10	83.64
	$K = 64, C = 16$	**67.8**	**3.28**	**91.30**	83.59

2.2. Anti-alias Downsampling

EAT adopts anti-aliasing (AA) down-sampling [24] which is regarded as an improvement of maxpool. It applies a low-pass filter with a blur kernel after the max operation to mitigate aliasing caused by the pooling. When the temporal receptive field is small, this operation reduces aliasing and enhances translational invariance. However, with the gradual increment of the frequency domain resolution, the parameter size of the 1-d convolution grows and the temporal receptive field expands. The blur kernel might ignore the spectral correlation and alter the distribution of its features, leading to performance degradation.

3. PROPOSED METHOD

In this paper we presented the gated multi mini-patch extractor to improve the preservation of pseudo-frequency and temporal features during the down-sampling process. We replace the anti-alias down-sampling block in EAT with our GMME and the structure of proposed network is illustrated in Fig. 1. GMME takes a tensor $X = [\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_B}] \in \mathbb{R}^{f \times t}$ as input and the corresponding output is $Y = [\mathbf{y_1}, \mathbf{y_2}, \ldots, \mathbf{y_B}] \in \mathbb{R}^{2f \times \frac{t}{s}}$, where B is the batch size and s is the stride and we set it to 2. Along the time frames, Y undergoes down-sampling, while along the pseudo-frequency bins, it represents expansion. In audio convolution operations, it is often beneficial to improve the pseudo-frequency domain resolution for feature learning [25]. The detailed structure of GMME is depecited in the Fig. 2.

The main workflow of GMME can be divided into four setps. Firstly, temporal and pseudo-spectral relationships are encoded by local audio characteristics. In order to focus on neighboring information, we split $\mathbf{x}_i$ into temporal, frequency, and global branches, using rectangular convolution kernels for CNN's locality and anti-aliasing [25, 26]. Specifically, given the input $\mathbf{x_i} \in \mathbb{R}^{f \times t}$, we use kernels with three spatial ranges: $[k, 2s]$, $[2s, k]$, and $[2s, 2s]$, where k denotes the kernel size and s represents the stride, i.e., the rate of the time down-sampling. At the same time the channel was expanded to C. Mathematically, this process can be denoted as

$$\begin{aligned} \mathbf{Z}^f &= PRelu\left[\mathcal{F}_f(x_i)\right], \\ \mathbf{Z}^t &= PRelu(\mathcal{F}_t(x_i)), \\ \mathbf{Z}^g &= PRelu(\mathcal{F}_g(x_i)), \end{aligned} \tag{1}$$

where $\mathcal{F}_f, \mathcal{F}_t, \mathcal{F}_g$ represent the convolutional operations in the frequency, temporal, and global branches, respectively. $PRelu$ is the parametric rectified linear unit (PRelu) [27]. By applying these kernels, we obtain different attention features $\mathbf{Z}^f, \mathbf{Z}^t \in \mathbb{R}^{C \times \frac{f}{s} \times \frac{t}{s}}$, and $\mathbf{Z}^g \in \mathbb{R}^{\frac{C}{2} \times \frac{f}{s} \times \frac{t}{s}}$.

Secondly, to extract the correlation between different channels, we borrow the gate mechanism from gated recurrent unit (GRU) [28]. We divided the input channels into two parts, one half of the channels is defined as gated maps, referred to as $\mathbf{G^t}$ and $\mathbf{G^f}$. The other half is treated as feature maps, denoted as $\mathbf{M^t}$ and $\mathbf{M^f}$. This process can be expressed as (2).

$$\begin{aligned} \mathbf{G}^t &= \mathbf{Z}^t[:, : \frac{1}{2} \times C, :, :] \quad & \mathbf{M}^t &= \mathbf{Z}^t[:, \frac{1}{2} \times C :, :, :] \\ \mathbf{G}^f &= \mathbf{Z}^f[:, : \frac{1}{2} \times C, :, :] \quad & \mathbf{M}^f &= \mathbf{Z}^f[:, \frac{1}{2} \times C :, :, :] \end{aligned} \tag{2}$$

Thirdly, gating is carried out to extract time and frequency interactive information and create the global feature map. The sigmoid function σ is used to convert $\mathbf{G}^t$ and $\mathbf{G}^f$ into weight maps over different domains. As strong probability distributions over different domains will emphasize the global weights and weak probability distributions will lead to underplay global weights, we utilize Hadamard product $\odot$ in both weight maps to get the global gated weights $\mathbf{G}$, that the different domains will interact with each other to get the weights, acting as a mask for removing as much redundant or distracting information as possible and keeping the effective information. Similarly, $\mathbf{G}^t$ and $\mathbf{G}^f$ were also element-wise product to get the global feature $\mathbf{M}$

$$\mathbf{G} = \sigma(\mathbf{G}^t) \odot \sigma(\mathbf{G}^f) \tag{3}$$

$$\mathbf{M} = PRelu(\mathbf{M}^t \odot \mathbf{M}^f) \tag{4}$$

Eventually, Hadamard product of the gated weight $\mathbf{G}$ and the feature map $\mathbf{M}$ is considered to provides an global enhancement and feature fusion. To preserve output in the original representation, we employe a square kernel for global information learning as $\mathbf{Z}^g$. Then element-wise sum with

Table 2: Result in Speech Commands V2. K and C is respective Kernel size and Channel number. #D-Param is the parameter number of Downsample block, while #T-Param is represent Total model's parameter.

Model	Set	#D-Param(K)	#T-Param(M)	Accurancy(%)
EAT-S-AA (conventional)		493.1	1.97	97.76
EAT-S-GMME (proposed)	$K = 32, C = 8$	**17.4**	**1.50**	97.74
	$K = 32, C = 16$	**34.9**	**1.51**	97.71
	$K = 64, C = 16$	**67.8**	**1.54**	**97.88**

the weighted feature map, which works as shortcut is implemented. The output feature is then generated as

$$\mathbf{O} = reshape(\phi_c(\mathbf{Z}^g + (\mathbf{G} \odot \mathbf{M}))), \tag{5}$$

where ϕ_c refers to point-wise convolution kernel. It is used for channel adjustment and structure fitting. Besides, it also promotes fusion between the pseudo-frequency domain and the time domain. Finally, $reshape$ is implemented to adjust the feature dimensions for the rest modules.

4. EXPERIMENT

4.1. Dataset

- **ESC-50 dataset** [16]: it consists of 2000 5-seconds long audio signals of 50 different environmental audio events, and the sampling rate of 44.1 kHz. We resampled them to 22.05 kHz, and trained 3500 epochs.
- **Speech Commands V2** [17]: each audio sample in the dataset has a fixed duration of 1-second and a sampling frequency of 16 kHz. The dataset includes 35 classification tasks, each representing a specific word or command to be recognized. We trained 300 epochs.
- **UrbanSound8K** [18]: this dataset comprises 8732 audio recordings, each belonging to one of 10 predefined classes, representing various common urban sounds. We resampled them to 22.05 kHz, zero-padding the short samples to 4-seconds and trained 1500 epochs.

4.2. Training Configuration

We adopted EAT-S model structure and replaced the original anti-alias down-sampling with our GMME module. We compared the performance and complexity of our network with original ETAs with above three datasets. We also investigate the performance with convolutional kernel sizes and channel widths.

All hyperparameters remain consistent with the open-source code[1] of the original paper. The parameters are optimized using the AdamW optimizer [29]. We use the Onecycle scheduler [30] with a maximum learning rate of $3e^{-4}$ and epsilon of $1e^{-8}$, and employ the exponential moving average (EMA) [31, 32] method with a decay rate of 0.995. The batchsize B is set to be 128. The loss is label-smoothing with a noise parameter set to be 0.1 for single-label classification tasks and binary cross-entropy. We perform data augmentation according to the methods described in the original paper, all experiments are conducted from scratch without pretraining.

[1] https://github.com/Alibaba-MIIL/AudioClassfication

4.3. Result

Tabel 1 presents the results of ESC-50 and UrbanSound8K datasets. For ESC-50, we can observe that the proposed method demonstrates significant advantages. Compared to the original network, it can maximally compresses the parameter size by approximately 98% (from 1969 K to 34.9 K) and achieve slight better performance (91.25 % vs 91.30 %) with 67.8 K model size. The overall model structure parameter size is also compressed by nearly 40% (from 5.18 M to 3.25 M). The results of Speech Commands V2 presented in Tab. 2 again validate the improvement. In UrbanSound8k, due to the existence of large amount of silent pieces, performance of all algorithms degraded as silent pieces do not contain any information, which also indicate it is important to implement audio activity detection before classification. On this dataset, our methods can still achieve comparable performance with EAT-S. We also experimented different settings of GMME, the results manifest that both kernel size K and the number of channels C affect the accuracy. It is clear that bigger kernel size achieves better performance. One possible reason is that bigger kernels can capture more local features. About channel numbers, there is a similar tendency in channel numbers, it may due to the enhancement of the anti-aliasing capability by convolutional layer [25].

5. CONCLUSION

Inspired by GRU, we proposed a down-sampling module for feature extraction, termed as gated multi mini-patch extractor, which integrates the advantages of attention and pooling. GMME extracts time-domain and the frequency-domain information through a mini-path extraction. Then the interrelationship of the time-domain and the frequency-domain information is modeled with the gate mechanism and a square kernel is carried out to extract global feature. Compared to conventional anti-aliasing blocks, GMME significantly reduces the model size. Experiments indicate that GMME could serve as a viable substitute for the process of audio classification.

6. REFERENCES

[1] M. Rajapakse and L. Wyse, "Generic audio classification using a hybrid model based on gmms and hmms," in *Proc. MMM*, 2005, pp. 53–58.

[2] Z. Liu, J. Huang, and Y. Wang, "Classification tv programs based on audio information using hidden markov model," in *Proc. IEEE ICASSP*, 1998, pp. 27–32.

[3] P. Gimeno, I. Viñals, A. Ortega, A. Miguel, and E. Lleida, "Multiclass audio segmentation based on recurrent neural networks for broadcast domain data," *EURASIP J.ASMP*, vol. 2020, no. 1, pp. 1–19, 2020.

[4] Y. Gong, Y.-A. Chung, and J. Glass, "PSLA: Improving audio tagging with pretraining, sampling, labeling, and Aggregation," *IEEE/ACM Trans. ASLP*, vol. 29, pp. 3292–3306, 2021.

[5] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Trans. ASLP*, vol. 28, pp. 2880–2894, 2020.

[6] Q. Kong, Y. Xu, W. Wang, and M. D. Plumbley, "Sound event detection of weakly labelled data with cnn-transformer and automatic threshold optimization," *IEEE/ACM Trans. ASLP*, vol. 28, pp. 2450–2460, 2020.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, and L. Jones, "Attention is all you need," *in Proc. NIPS*, vol. 30, pp. 5998—-6008, 2017.

[8] J. S. Luz, M. C. Oliveira, F. H. Araujo, and D. M. Magalhães, "Ensemble of handcrafted and deep features for urban sound classification," *Applied Acoustics*, vol. 175, p. 107819, 2021.

[9] M. Huzaifah, "Comparison of time-frequency representations for environmental sound classification using convolutional neural networks," *arXiv preprint arXiv:1706.07156*, 2017.

[10] J. K. Das, A. Ghosh, A. K. Pal, S. Dutta, and A. Chakrabarty, "Urban sound classification using convolutional neural network and long short term memory based on multiple features," in *Proc. IEEE ICDS*, 2020, pp. 1–9.

[11] T. Tax, J. L. D. Antich, H. Purwins, and L. Maaløe, "Utilizing domain knowledge in end-to-end audio processing," in *Proc NIPS*, 2017.

[12] S. Verbitskiy, V. Berikov, and V. Vyshegorodtsev, "Eranns: Efficient residual audio neural networks for audio pattern recognition," *PRL*, vol. 161, pp. 38–44, 2022.

[13] S. Venkatesh, D. Moffat, and E. R. Miranda, "You only hear once: a yolo-like algorithm for audio segmentation and sound event detection," *Applied Sciences*, vol. 12, p. 3293, 2022.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE CVPR*, 2016, pp. 770–778.

[15] A. Gazneli, G. Zimerman, T. Ridnik, G. Sharir, and A. Noy, "End-to-end audio strikes back: Boosting augmentations towards an efficient audio classification network," *arXiv preprint arXiv:2204.11479*, 2022.

[16] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proc. ACM MM*, 2015, pp. 1015–1018.

[17] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[18] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proc. ACM MM*, 2014, pp. 1041–1044.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, and X. Zhai, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[20] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio spectrogram transformer," *arXiv preprint arXiv:2104.01778*, 2021.

[21] S. Park, Y. Jeong, and T. Lee, "Many-to-many audio spectrogram tansformer: Transformer for sound event localization and detection." in *Tech. Rep., DCASE2021 Challenge*, 2021.

[22] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, "HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection," in *Proc. ICASSP*, 2022, pp. 646–650.

[23] S. Chen, Y. Wu, C. Wang, S. Liu, and D. Tompkins, "BEATs: Audio pre-training with acoustic tokenizers," *arXiv preprint arXiv:2212.09058*, 2022.

[24] R. Zhang, "Making convolutional networks shift-invariant again," in *Proc. ICML*, 2019, pp. 7324–7334.

[25] A. H. Ribeiro and T. B. Schön, "How convolutional neural networks deal with aliasing," in *Proc. ICASSP*, 2021, pp. 2755–2759.

[26] L. Chi, B. Jiang, and Y. Mu, "Fast fourier convolution," *in Proc. NIPS*, vol. 33, pp. 4479–4488, 2020.

[27] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arXiv:1505.00853*, 2015.

[28] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

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

[30] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay," *arXiv preprint arXiv:1803.09820*, 2018.

[31] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," *in Proc. NIPS*, vol. 30, 2017.

[32] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, and A. G. Wilson, "Averaging weights leads to wider optima and better generalization," *arXiv preprint arXiv:1803.05407*, 2018.