MedFM Solution: Conv-adapter

Tianyi Wang[1*], Mengkang Lu[1*], and Yong Xia[1(✉)]

National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, School of Computer Science and Engineering, Northwestern Polytechnical University, Xi'an 710072, China

1 Method

1.1 Backbone

In this challenge, we choose several benchmark networks as backbones, such as vision Transformer (ViT), Swin Transformer, BEiT, and EVA. Based on the submitted results, we chose EVA as the final backbone.

1.2 Adapter

In this challenge, we choose visual prompt tuning and three types of adapters (Lora, adaptformer and conv-adapter), as shown in Fig. 1. Based on the submitted results, we chose conv-adapter as the final backbone, which contains 2 1×1 conv layers, 2 GELU layers, and a 3×3 conv layer. We add adapters on both MHSA and FFN layers.

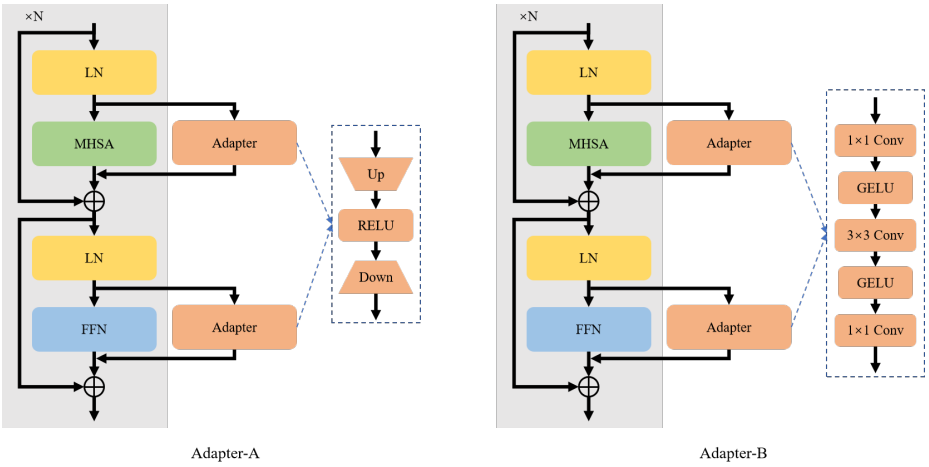

Fig. 1. Diagrams of adapter-A and adapter-B in MedFM challenge.

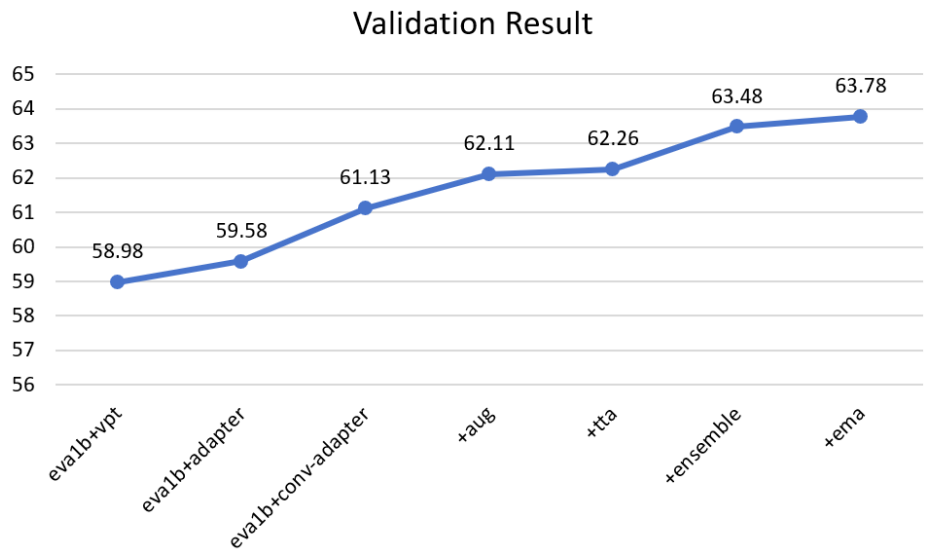

Fig. 2. The validation results of exp1.

2 Experiment Setup

The experiments were conducted with PyTorch using 4 NVIDIA A100 Tensor Core GPUs. We initialized the model with one of the official EVA pre-trained weights. AdamW optimizer was employed with a learning rate of $6e-4$. To schedule the learning rate, we used Cosine Annealing with 20 warmup epochs. Our model was fine-tuned for 50 epochs and the best weight was kept based on the validation set.

To reproduce our result, please follow the README file in our repository on GitHub. If you encounter any issues, feel free to contact us via GitHub issue or email. More details of our submission will be available in the near future.

3 Result

The submitted results are shown in Fig. 2. Finally, we select the EVA-1b with conv-adapter. Besides, strong data augmentation, test time adaptation (TTA), model ensemble, and exponential moving average (EMA) are used to achieve better results.