Bi-Tuning: Efficient Transfer from Pre-Trained Models

Anonymous authors

No Institute Given

A Key Generating Mechanisms

A.1 Momentum Contrast

Momentum Contrast (MoCo) [1] is a general key generating mechanism for using contrastive learning loss. The main idea in MoCo is producing encoded keys on-the-fly via a momentum-updated encoder and maintaining a queue to support sampling operations. Thus, the memory cost in MoCo does not depend on the size of the training set (while a memory bank [3] will store the whole dataset). In all our experiments (Section 5), Bi-tuning chooses the unsupervised MoCo as our default setting.

Formally, denoting the momentum-updated encoder as f_{k} with parameters θ_{k}. Likewise, denoting the backbone encoder as f_{q} with parameters θ_{q}. θ_{k} is updated by:

$$\theta_{\mathrm{k}} \leftarrow m\theta_{\mathrm{k}} + (1-m)\theta_{\mathrm{q}}. \tag{1}$$

Here we set the momentum coefficient $m = 0.999$. To fit the Bi-tuning approach, we reorganize the queues in MoCo for items in each category separately. Moreover, two contrastive mechanisms in Bi-tuning are performed on different instance levels and category levels, respectively, and we maintain two groups of queues correspondingly.

Table 1: Top-1 accuracy (%) of Bi-tuning on CUB with memory bank as key generating mechanism (Backbone: ResNet-50 pretrained via MoCo).

Key Generating Mechanism	Sample Rate			
	25%	50%	75%	100%
MoCo [1]	49.25±0.23	66.88±0.13	74.27±0.05	77.12±0.23
Memory bank [3]	50.01±0.55	66.69±0.26	74.22±0.31	77.62±0.29

A.2 Memory Bank

Bi-tuning is a general approach, which is not bound to any special key generating mechanism (MoCo). The memory bank proposed by [3] generates encoded keys via momentum-updated snapshots of all items in the training set. Keys for each mini-batch are uniformly sampled from the memory bank. Compared to MoCo, maintaining a memory bank is more computation-efficient with more memory required. Similar to Eq. (1), snapshots here are updated by:

$$\mathbf{z}_i^{\mathrm{k}} \leftarrow m\mathbf{z}_i^{\mathrm{k}} + (1-m)\mathbf{z}_i^{\mathrm{q}}, \tag{2}$$

$$\mathbf{h}_i^{\mathrm{k}} \leftarrow m\mathbf{h}_i^{\mathrm{k}} + (1-m)\mathbf{h}_i^{\mathrm{q}}. \tag{3}$$

Notations follow Section 3. Here we set the momentum coefficient $m = 0.5$ [3]. Other hyper-parameters are the same as Section 5. We evaluate Bi-tuning with a memory bank on CUB [2] with the same configurations in Section 4. The results in Table 1 show that the performance is close in both methods. Key generating mechanisms in Bi-tuning only have limited effects on the final performance in the supervised paradigm. These suggest that the key generating mechanism in Bi-tuning can be implemented by some variants with similar performance. MoCo is recommended regarding its scalability and simplicity.

References

1. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR (2020)
2. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-UCSD Birds 200. Tech. Rep. CNS-TR-2010-001, California Institute of Technology (2010)
3. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: CVPR (2018)