Coeff-Tuning: A Graph Filter Subspace View for Tuning Attention-Based Large Models

Supplementary Material

A. More on Analysis

Proposition A.1. *Given N nodes $\{\mathbf{x}_i\}_{i=1}^N$, the set $\mathcal{S}(\mathbf{x}) = \{\sum_{i=1}^N \lambda_i \mathbf{x}_i \mid \lambda_i > 0,\ \sum_i \lambda_i = 1\}$ is a bounded convex set.*

Proof. For each element $\mathbf{x}' = \sum_{i=1}^N \lambda_i \mathbf{x}_i$ of $\mathcal{S}(\mathbf{x})$, we have,

$$\|\mathbf{x}'\| = \left\| \sum_{i=1}^N \lambda_i \mathbf{x}_i \right\| \leq \sum_{i=1}^N \lambda_i \|\mathbf{x}_i\| . \tag{14}$$

With given $\{\mathbf{x}_i\}_{i=1}^N$, every element $\mathbf{x}' = \sum_{i=1}^N \lambda_i \mathbf{x}_i$ is bounded. Therefore, $\mathcal{S}(\mathbf{x})$ is a bounded convex set. □

Proposition A.2. *If $\mathcal{S}^h$ is a bounded convex set, $\forall h = 1, \cdots, H$, the set $\mathcal{S} = \mathcal{S}^1 + \cdots + \mathcal{S}^H = \{\mathbf{x}^1 + \cdots + \mathbf{x}^H \mid \forall \mathbf{x}^h \in \mathcal{S}^h\}$ is also a bounded convex set.*

Proof. For each element $\mathbf{x}' = \sum_{h=1}^H \mathbf{x}^h$ of $\mathcal{S}$, we have,

$$\|\mathbf{x}'\| = \left\|\mathbf{x}^1 + \cdots + \mathbf{x}^H\right\| \leq \left\|\mathbf{x}^1\right\| + \cdots + \left\|\mathbf{x}^H\right\| . \tag{15}$$

Since $\left\|\mathbf{x}^h\right\|, \forall h = 1, \cdots, H$ are bounded, $\|\mathbf{x}'\|$ is also bounded. Therefore, $\mathcal{S} = \mathcal{S}^1 + \cdots + \mathcal{S}^H$ is also a bounded convex set. □

Detailed calculation of parameters. We have provided the analysis and estimation of the parameters in Sec. 4.2 and Sec. 5 in the submission. Our method requires a significantly small number of parameters. In ViT for example, our method only requires an additional $\mathbf{12 \times 12 \times 12 = 0.0017M}$ parameters (12 heads with 12 layers), which is negligible. We provide the number of parameters with more precisions in Table 1.

Complexity Analysis. We provide comparisons of training flops of forward pass & backward pass of a single image in Table 1. Our method utilizes comparable training flops with other baseline methods while achieving consistent performance improvement.

B. Supplementary Experimental Details

B.1. Task and Experiment Backgrounds

More on PEFT. Besides the widely-adapted LoRA [21] and following works [38], [4] propose the filter subspace decomposition for weight matrices. The filter subspace decomposition method [52] has shown effectiveness in continual learning [3, 40, 75], video representation learning [41], graph learning [7], and generative tasks [33, 72–74]. There are some other works on fine-tuning the SVD decomposition of weights[15], Kronecker decomposition [47–49], sparsity [32, 39, 70], non-linearity [83] or fine-tunig bias parameters[76].

Tuning Vision Foundation Models. As there are emergent needs to customize the pre-trained foundation models for downstream tasks, a large corpus of fine-tuning methods has been proposed for both both discriminative and generative tasks. Among them, [16–19, 42, 43, 56, 61, 62, 66, 77] have focused on tuning pre-trained image diffusion models for personalized generation, diversity, compositional generation, and human preference. While [23, 24] propose to tune propose to tune vision transformers for downstream discriminative tasks.

B.2. ViT Fine-tuing

We first describe our selection of high-resolution sub-tasks from the 19 VTAB [81] fine-tuning dataset. Specifically, we select datasets containing images with a resolution equal to or higher than 224, corresponding to the pre-training image in ImageNet-21k [9]. The selected datasets are shown in Table 1.

We now present the training details. For the first setting, i.e., tuning subspace coefficient $\boldsymbol{\alpha}$ only, we only add for each attention layer a subspace coefficient $\boldsymbol{\alpha}$ with the proposed parameterization and tune $\boldsymbol{\alpha}$ together with the linear head for each task. We set the dropout rate $p = 0$. For the second case, we add $\boldsymbol{\alpha}$ in the same way while adding the scaling and shift parameters as in SSF [35], and set the dropout rate $p = 0.1$. For both settings, we adopt the batch size of 64 and train the model on each task with the AdamW optimizer for 100 epochs.

B.3. Concept Customization

In this experiment, we choose 10 concepts from Dreambooth [56] and Custom Diffusion [30]. These concepts include toys, objects, and animals. We generate images with 25 text prompts adapted from Dreambooth [56]. We utilize Adam [26] optimizer with a learning rate of 3×10^{-4} and fine-tune the SDXL for 200 steps. The ranks of LoRA [21] and Dora [38] are $r = 2$. We generate 4 different images with the shape of 1024×1024 for each text prompt.

We provide additional comparison in Figure 5-10. The generated images with *Coeff-Tuning* have higher concept fi-

delity, preserving more characteristics from the input images.

B.4. Image-Text Understanding

We provide the experiment details of the multi-modal tuning with VL-BART [8]. Following the settings in DoRA [38], we utilize the fixed vision tower CLIP-ResNet-101 [54], and tune the BART, a encoder-decoder language model, with *Coeff-Tuning* with multi-task image-text datasets as described in Sec. 5.3. Specifically, we demonstrate the ability of the proposed *Coeff-Tuning* can be integrated with other popular weight-based PEFT methods in a plug-and-play manner. So we first add either DoRA [38] or LoRA [21] with $r = 128$, and then add the subspace coefficient $\boldsymbol{\alpha}$ in each attention layer, including self-attention layers in both the encoder and the decoder, and the cross-attention in the decoder. Specifically, we introduce additional $(12 + 6) * 12 * 12 = 2.6K$ parameters in the BART, which are neglectable compared with the added LoRA or DoRA which takes millions of parameters.

As for the training, we set the batch size to 300, adopt the AdamW optimizer, and train for 20 epochs. For DoRA and *Coeff-Tuning*+DoRA, we adopt the learning rate of 1×10^{-3}, weight decay of 0.01 for DoRA parameters, and learning rate 5×10^{-4}, weight decay of 1×10^{-6}, $p = 0.2$ for tuning $\boldsymbol{\alpha}$. For LoRA and *Coeff-Tuning*+LoRA, we choose a learning rate 5×10^{-4}, weight decay of 0.01 for LoRA parameters, and $2 \times 10^{-4}, 1 \times 10^{-6}$ for coefficient $\boldsymbol{\alpha}$.

Figure 5. Results on Concept Customization.

Figure 6. Results on Concept Customization.

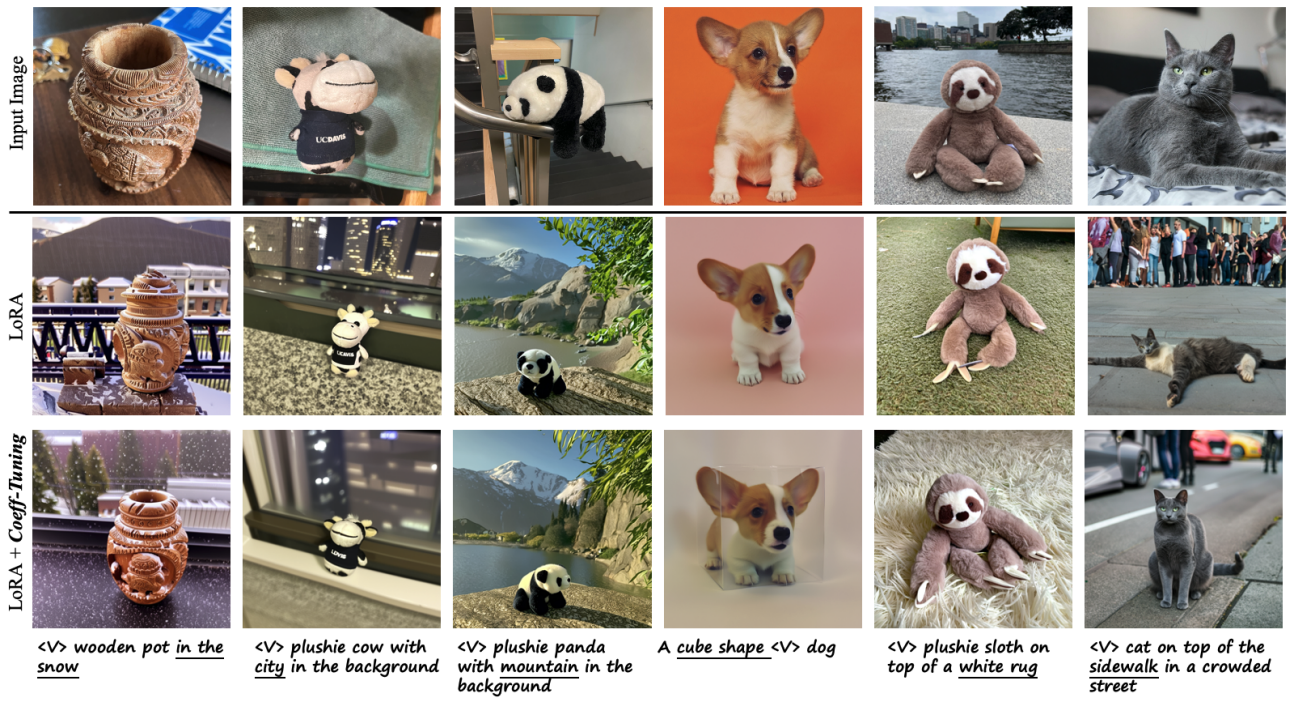

Figure 7. Results on Concept Customization.

Figure 8. Results on Concept Customization.

Figure 9. Results on Concept Customization.

Figure 10. Results on Concept Customization.