Multitask Vision-Language Prompt Tuning

Sheng Shen* Shijia Yang* Tianjun Zhang* Bohan Zhai
Joseph E. Gonzalez Kurt Keutzer Trevor Darrell
University of California, Berkeley
{sheng.s,shijiayang,tianjunz,zhaibohan,jegonzal,keutzer,trevordarrell}@berkeley.edu

Abstract

Prompt Tuning, conditioning on task-specific learned prompt vectors, has emerged as a data-efficient and parameter-efficient method for adapting large pretrained vision-language models to multiple downstream tasks. However, existing approaches usually consider learning prompt vectors for each task independently from scratch, thereby failing to exploit the rich shareable knowledge across different vision-language tasks. In this paper, we propose multitask vision-language prompt tuning (MVLPT), which incorporates cross-task knowledge into prompt knowledge for vision-language models. Specifically, (i) we demonstrate the effectiveness of learning a single transferable prompt from multiple source tasks to initialize the prompt for each target task; (ii) we show many target tasks can benefit each other from sharing prompt vectors and thus can be jointly learned via multitask prompt tuning. We benchmark the proposed MVLPT using three representative prompt tuning methods, namely text prompt tuning, visual prompt tuning, and the unified vision-language prompt tuning. Results in 20 vision tasks demonstrate that the proposed approach outperforms all single-task baseline prompt tuning methods, namely text prompt tuning, visual prompt tuning, and the unified vision-language prompt tuning. Results in 20 vision tasks demonstrate that the proposed approach outperforms all single-task baseline prompt tuning methods, setting the new state-of-the-art on the few-shot ELEVATE benchmarks and cross-task generalization benchmarks. To understand where the cross-task knowledge is most effective, we also conduct a large-scale study on task transferability with 20 vision tasks in 400 combinations for each prompt tuning method. It shows that the most performant MVLPT for each prompt tuning method prefers different task combinations and many tasks can benefit each other, depending on their visual similarity and label similarity.

1. Introduction

Recent large-scale vision-language models, pretrained on a wide variety of images with natural language supervision (i.e., CLIP [67], ALIGN [38] and Florence [96]), have demonstrated strong open-set recognition abilities for image classification in-the-wild [50, 67] and open-vocabulary detection [29]. Despite the impressive zero-shot transfer capabilities, adapting these large-scale vision-language models to downstream tasks presents its own challenges. It is usually prohibitive to fine-tune the entire model due to both huge parameter sizes and well-known overfitting issues for few-shot learning.

Such a trend emerges the essential need to study different adaptation methods [36, 37, 55], where Prompt Tuning [48, 105] has shown to be one of the most effective strategies. Typically, Prompt Tuning tunes only a small number of parameters for each task in a model’s input spaces (prompt vectors) while keeping the pretrained model frozen. It was first introduced in NLP community [48, 55, 61] and has recently demonstrated superior few-shot adaptation performance [39, 105, 106] for vision-language models. CoOp [105] and VPT [39] are two representative vision-language prompt tuning methods, in which the former uses a textual prompt and the latter leverages the visual prompt.

However, on the one hand, most of these vision-language
prompt tuning methods (i.e., CoOp, VPT) focuses on learning a prompt for each downstream task independently, failing to incorporate cross-task knowledge when adapting to various downstream tasks. On the other hand, multitask learning has a rich literature [8, 80, 84, 102] for vision. Applying multitask prompt tuning to language models has also presented impressive few-shot [4, 58] or zero-shot generalization capability [13, 71]. This motivates us to investigate the question: Can vision-language model also benefit from multitask knowledge sharing via prompt tuning during adaptation?

To this end, we propose multitask vision-language prompt tuning (MVLPT), to the best of our knowledge, the first method incorporating the cross-task knowledge into vision-language prompt tuning. MVLPT is a simple yet effective way to enable information sharing between multiple tasks. MVLPT consists of two stages: multitask source prompt initialization and multitask target prompt adaptation. Specifically, multitask prompt initialization first learns shared prompt vectors from various source tasks. Then, the shared prompt can initialize the prompt for target tasks. To adapt to target tasks, multitask prompt adaptation will group relevant tasks together and perform multitask prompt tuning within the selected groups. We remark that we could also perform single-task adaption with setting group size as one. This simple scheme enables passing cross-task knowledge from source tasks to target tasks through multitask prompt initialization, and exploiting shareable knowledge within target tasks via multitask prompt adaptation further.

We conduct extensive evaluations of MVLPT on 20 vision tasks in few-shot ELEVATER [50] in Section 4.2. Comparing to CoOp [105], VPT [39] and UPT (Section 3.1), MVLPT improves the baselines by 0.72%, 1.73% and 0.99% respectively and sets the new state-of-the-art on 20-shot ELEVATER benchmark. We also show the strong generalizability of MVLPT where MVLPT improves CoOp, VPT and UPT by 1.73%, 4.75% and 4.53%, respectively on cross-task generalization benchmark in Section 4.1 and study task transferability with the 20 vision tasks and in 400 combinations for each prompt method in Section 4.3.

In summary, we make the following contributions:

- We propose the multitask vision-language prompt tuning (MVLPT) framework, including multitask prompt initialization and multitask prompt adaptation, and demonstrate the efficacy for each component.

- We rigorously study the task transferability across 20 vision tasks with 400 combinations for each prompt tuning method to understand when MVLPT is most effective.

- We systematically evaluate the proposed MVLPT on the few-shot ELEVATER and cross-task generalization benchmarks, which sets the new state-of-the-art on 20-shot ELEVATER benchmark.

2. Related Work

Vision-Language Models [11, 101] align images and texts into a joint embedding space using image and text encoding, and loss functions for alignment. Traditionally, models are designed and learned independently for images and texts, connected only by a loss module. Images are encoded using hand-crafted descriptors [19, 77] or neural networks [23, 47], while texts can be encoded with pre-trained word vectors [23, 77] or frequency-based features [19, 47]. To align these modalities, metric learning [23], multi-label classification [28, 41], and n-gram language learning [49] are used.

With the rise of large-scale pretraining, vision-language models [24, 44, 51–54, 56, 74, 75, 81, 83, 88, 90, 93, 98] now learn two encoders jointly and use larger neural networks (up to 80B parameters as in [2]) and datasets. As discussed in Zhe et al. [25], recent successes in vision-language models can mainly attribute to developments in Transformers [85],
3. Methodology

We first revisit the CLIP [67], in company with text, visual, and unified prompt tuning approaches for visual recognition in Section 3.1. We then present technical details of our proposed MVLPT learning in Section 3.2.

3.1. Preliminaries

CLIP [67] is a model that trains both an image encoder and a text encoder to create similar embeddings for image-text pairs. It accomplishes this through minimizing a symmetric contrastive loss during pretraining, which predicts a positive sample in a batch of image-text combinations:

\[
p_{i \rightarrow v} = \frac{\exp (\cos (u_i, v_j) / \tau)}{\sum_{j=1}^{N} \exp (\cos (u_i, v_j) / \tau)}
\]  

where \( u = \psi(x) \in \mathbb{R}^d \) indicating the projection of image \( x \) to the final hidden space of dimension \( d \); \( v = \phi(y) \in \mathbb{R}^d \) indicating the projection of text \( y \); \( \cos(\cdot, \cdot) \) denotes the cosine similarity; \( \tau \) is a learnable temperature value. In zero-shot prediction, CLIP takes an image and a set of target classes, constructs a fixed prompt “a photo of a [CLASS]” for each class, and predicts the class with the highest cosine similarity between the encoded image and the set of prompts.

While the definition of a “task” is unclear, we borrow the definition from CLIP. For clarity, we formally distinct different tasks: \( K \)-way classification on dataset \( D \) is a different task than \( M \)-way classification on different dataset \( D' \), where \( K \) and \( M \) are different.

Text Prompt Tuning is a method used for adapting CLIP-like vision-language models to downstream tasks. It is a more efficient approach than finetuning the entire CLIP model. CoOp [105] proposed this method by replacing a prompt’s context words with a learnable vector \( P \in \mathbb{R}^{d \times n} \) of adjustable length \( n \). The text input is modified to:

\[
P = [p_1, p_2, \ldots, p_n, \text{CLASS}].
\]  

This modification allows for freezing the image and text encoders while optimizing only \( P \) with task-specific objective functions.

CoCoOp [106] is a newer method that adds a network to obtain an input-conditional token and achieves better performance than CoOp. However, its limitation in training...
In this stage, we transfer \( \mathbf{A} \) (Section 4.3). We use \( \mathbf{B} \) in Eq. (4.2) to perform this pre-training, \( \mathbf{C} \) and \( \mathbf{D} \) are pretrained jointly. In this stage, the shareable source prompt \( \mathbf{E} \) is later employed as text prompt tuning within the selected groups from the same multitask. Relevant effects. During downstream training, UPT froze both the text and visual encoder \( \phi \) and \( \psi \) and only optimizes the vision-language prompts \( \mathbf{F} \) and the lightweight Transformer layer \( \theta \). In this way, both the dynamic classifiers \( \mathbf{G} \) and visual features \( \mathbf{H} \) in Eq. (1) are effectively tuned for reliable prediction in the downstream task.

3.2. Multitask Vision-Language Prompt Tuning

Our proposed framework MVLPT mainly consists of two stages as shown in Figure 2, multitask source prompt initialization and multitask target prompt adaptation.

Multitask Prompt Initialization. In this stage, the shareable prompts for all source tasks are pretrained jointly through multitask prompt tuning. Note that we only use few-shot training set from source tasks to perform this pre-train versus using the entire set in NLP community [4, 87].

Multitask Prompt Adaptation. In this stage, we transfer the shareable source prompt to target tasks. For single-task target prompt adaptation, we then directly use the learned source prompt to initialize the target prompt and optimize with the regular task loss on each task (i.e., cross-entropy loss). For multitask prompt adaptation, we first group relevant tasks together, then perform multitask prompt tuning within the selected groups from the same multitask-initialized source prompt. The grouping strategies are further discussed in Section 4.3. A theoretical justification of task grouping is provided in Appendix.

4. Experiments

Our approach is mainly evaluated in the following three problem settings: 1) cross-task generalization (Section 4.1) that measures the efficacy of multitask prompt initialization; 2) few-shot ELEVATER (Section 4.2) that shows the effectiveness of multitask prompt adaption; and, 3) zero-shot task transferability (Section 4.3) that is based on the 20 vision
Table 1. Comparison of CoOp, CoCoOp, VPT, UPT, and our MCoCoOp, MCoOp, MVPT, and MUPT in the cross-task generalization setting. The results strongly justify the strong generalizability of multitask prompt initialization. Specifically, each multitask variant learns shared prompt vectors from 11 source tasks before single task adaptation to 12 target tasks. The shots number (1, 5, 20) denotes both the number of shots we use for multitask prompt initialization and single task adaptation. For instance, 1 shot means we use 1 shot from each source task for multitask prompt initialization and adapt that for 1 shot learning to each target task. **Boldface** text denotes the best performance in that setting. Noted that we include the CIFAR-10 in the averaged task table and the CIFAR-10 performance is in Appendix.

(a) Average over 12 tasks.  
(b) CIFAR-100.  
(c) Hateful Memes.

| # shots | 1 | 5 | 20 |
|---------|---|---|----|
| **CoOp** | 50.51 ± 1.8 | 55.50 ± 2.1 | 65.87 ± 0.5 |
| **CoCoOp** | 53.23 ± 1.6 | 57.37 ± 1.7 | 66.34 ± 0.6 |
| **VPT** | 57.06 ± 1.3 | 60.14 ± 1.0 | 69.98 ± 0.7 |
| **UPT** | 56.76 ± 0.7 | 62.16 ± 0.8 | 67.62 ± 0.6 |
| **MCoOp** | 55.85 ± 1.1 | 61.54 ± 1.6 | 67.60 ± 0.5 |
| **MCoCoOp** | 57.61 ± 0.6 | 63.49 ± 0.5 | 70.54 ± 0.4 |
| **MVPT** | 60.98 ± 0.4 | 65.91 ± 0.4 | 86.17 ± 0.3 |
| **MUPT** | 61.66 ± 0.2 | 65.77 ± 0.4 | 72.15 ± 0.4 |

(d) MNIST.  
(e) Resisc-45.  
(f) Country-211.

| # shots | 1 | 5 | 20 |
|---------|---|---|----|
| **CoOp** | 49.98 | 78.31 | 91.79 |
| **CoCoOp** | 51.61 | 79.41 | 92.07 |
| **VPT** | 71.61 | 74.00 | 88.62 |
| **UPT** | 60.44 | 81.64 | 89.88 |
| **MCoOp** | 65.06 | 78.30 | 94.14 |
| **MCoCoOp** | 66.17 | 79.52 | 95.08 |
| **MVPT** | **82.36** | **89.57** | **95.31** |
| **MUPT** | 81.29 | 88.48 | 94.54 |

(g) VOC 2007 Classification.  
(h) Patch-Camelyon.  
(i) Rendered-SST2.

| # shots | 1 | 5 | 20 |
|---------|---|---|----|
| **CoOp** | 55.78 | 63.70 | 77.43 |
| **CoCoOp** | 62.14 | 68.00 | 78.52 |
| **VPT** | 77.54 | 75.91 | 80.59 |
| **UPT** | 79.57 | 76.10 | 78.88 |
| **MCoOp** | 75.84 | 75.46 | 77.60 |
| **MCoCoOp** | 77.97 | 78.73 | 81.39 |
| **MVPT** | 78.39 | 79.19 | 81.67 |
| **MUPT** | **80.18** | **80.51** | **80.92** |

(j) GTSRB.  
(k) FER 2013.  
(l) KITTI Distance.

| # shots | 1 | 5 | 20 |
|---------|---|---|----|
| **CoOp** | 37.55 | 61.71 | 71.52 |
| **CoCoOp** | 41.55 | 64.41 | 72.21 |
| **VPT** | 52.58 | 72.42 | 86.17 |
| **UPT** | **57.67** | 70.72 | 85.34 |
| **MCoOp** | 37.89 | 59.31 | 72.09 |
| **MCoCoOp** | 40.54 | 62.08 | 76.49 |
| **MVPT** | 50.56 | 75.83 | 89.75 |
| **MUPT** | 51.79 | 69.22 | 85.30 |

| # shots | 1 | 5 | 20 |
|---------|---|---|----|
| **CoOp** | 29.34 | 28.25 | 50.71 |
| **CoCoOp** | 34.08 | 31.45 | 51.52 |
| **VPT** | 49.76 | 47.48 | 56.39 |
| **UPT** | 49.76 | 47.85 | 56.77 |
| **MCoOp** | 52.49 | 47.76 | 50.24 |
| **MCoCoOp** | 55.50 | 50.90 | 55.23 |
| **MVPT** | 51.43 | 50.85 | 57.12 |
| **MUPT** | **55.95** | **51.27** | **60.07** |

| # shots | 1 | 5 | 20 |
|---------|---|---|----|
| **CoOp** | 48.40 | 52.60 | 52.40 |
| **CoCoOp** | 49.44 | 53.30 | 52.58 |
| **VPT** | 55.40 | 53.20 | 57.20 |
| **UPT** | 51.80 | 54.93 | 56.60 |
| **MCoOp** | 54.00 | 53.80 | 59.40 |
| **MCoCoOp** | 54.63 | 54.53 | 60.56 |
| **MVPT** | **56.20** | **55.27** | **57.60** |
| **MUPT** | 56.20 | 55.20 | 56.60 |

Tasks in ELEVATER.

**Datasets** For the domain generalization setting, we use the 11 image recognition tasks from [105] as source tasks. In Section 4.1, we use the non-overlapped 12 image recognition tasks in ELEVATER [50] as target tasks, covering a diverse set of recognition tasks. Specifically, the source tasks include ImageNet [15] and Caltech101 [21]
Table 2. Comparison of prompt learning methods on the few-shot ELEVATER. The number of shots is set to be 20 in each case, except for zero-shot CLIP. The results suggest the significant generalizability of multitask prompt initialization. \(^1\) denotes the zero-shot CLIP results from ELEVATER \([50]\). “Source” denotes the prompt initialization source, where “-” stands for random initialization, and “M” stands for using all 20 ELEVATER tasks for prompt initialization. “Adaptation” denotes the target task prompt adaptation method, where “S” stands for single target task prompt adaptation that each target task will be adapted independently, and ‘M’” stands for multitask prompt adaptation that certain tasks (selected based on results in Section 4.3) will be learned together. Clearly, MVLPT demonstrates better transferability than single target task prompt adaptation counterparts. \(\Delta\) denotes the best M-variant’s gain over the respective baseline methods.

| Target          | CLIP\(^1\) | CoOp | VPT | UPT | MCoOp | MVPT | MUPt | CLIP | CoOp | VPT | UPT | MCoOp | MVPT | MUPt | CLIP | CoOp | VPT | UPT | MCoOp | MVPT | MUPt | CLIP | CoOp | VPT | UPT | MCoOp | MVPT | MUPt | CLIP | CoOp | VPT | UPT | MCoOp | MVPT | MUPt |
|-----------------|------------|------|-----|-----|-------|------|------|------|------|------|-----|------|-------|------|------|------|------|-----|------|-------|------|------|------|------|-----|------|-------|------|------|------|------|-----|------|-------|------|------|------|------|-----|------|-------|------|------|
|                 | Source     | Adaptation | Caltech101 | CIFAR10 | CIFAR100 | Country-211 | DTD | EuroSat | FER-2013 | FGVCAircraft | Flowers102 | Food101 | GTSRB | Hateful Memes | KITTI-Distance | MNIST | OxfordPets | PatchCamelyon | Rendered-SST2 | VPT - S | MCoOp - S | MVPT - S | MUPt - S | CLIP - S | CoOp - S | VPT - S | UPT - S | MCoOp - S | MVPT - S | MUPt - S | CLIP - S | CoOp - S | VPT - S | UPT - S | MCoOp - S | MVPT - S | MUPt - S |
|                 | 88.9       | 90.8   | 68.2  | 22.8  | 44.8   | 54.7   | 48.5  | 24.3  | 88.7   | 43.5   | 58.1   | 27.0   | 52.0  | 69.4   | 89.0   | 54.0  | 60.9   | 65.6   | 64.8   | 83.7   | 87.3   | 60.0   | 70.1   | 88.98  | 89.64  | 82.72  | 89.91  | 91.24  | 60.41  | 59.03  | 83.32  | 79.60  | 72.40  |
|                 | ±5.6       | ±3.0   | ±6.9  | ±1.4  | ±0.6    | ±0.5    | ±4.6  | ±0.3    | ±0.45  | ±0.30  | ±0.40  | ±4.28  | ±8.53  | ±3.20  | ±0.80  | ±0.93  | ±5.34  | ±0.56  | ±0.08  | ±3.33  | ±1.81  |

Baselines We compare our approach against the following methods: (i) Zero-shot CLIP \([67]\)^2. This baseline uses does not involve any prompt-learning strategies as mentioned in Section 3.1. (ii) Single Task Prompt Tuning methods, including CoOp \([105]\), VPT \([39]\), UPT \([97]\) for vision, language, and vision-language prompt tuning method.

Training Details Our implementation is based on CoOp.\(^3\) Throughout the experiments, we use CLIP as our vision-language model (i.e., ViT-B/16) for all the experiments except for the scaling ablation). Following CoOp \([105]\) and VPT \([39]\), we use a context length of 16 for both CoOp and VPT throughout the study. We empirically find a shorter context length of 4 leads to better performance for UPT, and we use 4 contexts for UPT only. (This design choice is discussed in more detail in the Appendix). The resulting prompt vectors of CoOp/MCoOp, VPT/MVPT, UPT/MUPT account for 0.01%, 0.11%, 0.45% total parameters of the ViT-B/16 (124M parameters) model. All the prompt vectors for CoOp, VPT or UPT are randomly initialized without using the pretrained word embeddings of “a photo of a” for initialization in \([105]\) for a fair comparison. All the methods are trained with a batch size of 32 for 200 epochs following \([105]\). All the image input size is set to 224 x 224. We use Adam optimizer and cosine learning rate schedule. All the learning rate is set as 2e-3, and the warmup period is set as 1 epoch following \([105]\). All the few-shot experiments are averaged with 3 runs. For each experiment, we select the best prompt checkpoint using the validation set that consists 20% splits from the few-shot sampled training set.

4.1. Cross-task Generalization

We examine the efficacy of the proposed multitask prompt initialization in MVLPT via cross-task generalization. Specifically, we use all the 11 tasks in \([105]\) as source tasks and the non-overlapped 12 tasks in ELEVATER as target tasks. We perform multitask learning on all the source tasks to learn the shared prompt vectors. The resulting shared prompt vectors will be used as the prompt initialization for single-task adaptation on each target task. We evaluate across 1, 5, 20 shots as suggested in the ELEVATER \([50]\).
benchmark. The shot number is adopted for both multitask prompt initialization and single target task adaptation, respectively. It means that for 1 shot, we will sample 1 instance for each image class of all the source tasks for multitask prompt initialization and then adapt the learned prompt initialization to 1-shot learning for each target task. The baseline prompt learning method CoOp, CoCoOp, VPT and UPT are using random initialized prompt as in [39,105] for single target task adaptation. The results are summarized in Table 1, showing that multitask prompt initialization variants MCoOp, MVPT and MUPT mostly outperform the baseline prompt learning counterparts by a significant margin. (averaged over 3 runs). The improvement is also consistent across different numbers of shots. It is also interesting that the most effective task of multitask prompt initialization differs for each prompt learning method. Specifically, MCoOp benefits the task where the class names are distinct the most like Resisc-45, while MVPT/MUPT favors the task where the images are more separable like VOC 2007 Classification. We further analyze this different preference in Section 4.3. Nevertheless, we note that multitask prompt initialization does not always guarantee performance improvement when the number of source task shots is extremely small as 1 and the target task needs extreme fine-grained or specialized classification like 211-way classification in Country-211.

4.2. Few-shot ElevaTER

We measure the effectiveness of the proposed multitask prompt adaptation in MVLPT on all 20 few-shot ElevaTER tasks. We set the number of shots as 20 in each setting. Specifically, versus adapting the learned prompt initialization to each target task independently (single-task prompt adaptation), we group several target tasks as in Figure 2 and perform multitask learning in each group to learn shared prompt vectors during prompt adaptation. We determine which tasks should be grouped for each prompt learning method based on the transferability map shown in Figure 4, which is discussed in more details in Section 4.3. The detailed results are shown in Table 2. It clearly demonstrates that multitask prompt adaptation variants exhibit better transferability than single target task prompt adaptation counterparts. Comparing single-task prompt adaption and multitask prompt adaption, multitask adaption boosts the averaged performance on CoOp, VPT, UPT by 0.86%, 0.75% and 0.81%, respectively. Using 20 ElevaTER tasks as source tasks can further improve the results for MVPT and MUPT. For MCoOp, multitask prompt initialization may make the class name distribution less separable for the task has distinct categories like KITTI Distance, which efiaces the improvement on other tasks. The resulting MVPT achieves 74.13% the new state-of-the-art on 20 shot ElevaTER benchmark for ViT-B/16 model comparing to 64.41% in [57]. We also observe that there exist tasks that are not improved using multitask prompt adaptation. We attribute that to some tasks like FGVCAircraft with distant and specialized categories may not be able to leverage useful cross-task knowledge from other ElevaTER tasks during prompt adaptation.

4.3. Task Transferability

To understand the cross-knowledge [8, 69, 80, 84, 102] in vision-language prompt tuning, we conduct a large-scale study on task transferability with 20 ElevaTER tasks in 400 combinations for each prompt tuning method, following [82]. We use checkpoints from each task in ElevaTER after 20-shot learning on 3 different seeds as the source. Then, we perform zero-shot adaptation to the rest of the tasks. We normalized the scores by dividing the transfer performance with the best one on that task and presented the results in Figure 4. To select groups for multitask adaptation, we select the top 1 and 2 transferability with respect to each target task. We jointly train such group of 2 and 3 tasks and select the best checkpoint based on the the validation performance for each task, respectively.

We also report the performance with different grouping strategies for multitask prompt adaptation on 20-shot ElevaTER in Table 4, where Best M stands for using the aforementioned grouping strategy and Worst M stands for grouping the most dissimilar tasks from the transferability map. It directly suggests that the transferability map could serve as a principal way to group the relevant tasks and failing to do that leads to significant performance degradation.

We additionally try two other task grouping methods, exploring task similarity encoded in learned prompts and unsupervised grouping. We first mimic ATTEMPT [3] and SPoT [87] to calculate cosine similarity between learned prompts. Adapting to our method, we choose tasks with the highest similarity based on the attention map and apply to multitask adaptation. For the second grouping method, we extract the feature of all training set images using CLIP. The features are clustered into 20 groups using K-Means, which gives the task grouping proposals. The results are shown in Table 3. As we adopt the common grouping method in NLP (Prompt Sim), we find that the result is slightly lower, while the cost is similar to ours. The unsupervised grouping (K-Means) does not require training prompts before grouping. It is efficient, but the performances are mostly lower than the single task baselines (Single). An unsupervised method can induce error during task grouping, which hurts the performance in return.

5. Discussion

Source Tasks There is rich literature [10, 30, 31, 33] to use ImageNet1K to pretrained vision backbones for various downstream vision tasks (object detection [27], semantic seg-

---

4We provide detailed task group information in Appendix.
The ablation study on context length is multitask prompt initialization.

**Table 3.** Averaged results of multitask adaptation on ELEVATER with different task grouping methods.

| Method   | CoOp | VPT | UPT |
|----------|------|-----|-----|
| Single   | 71.67±0.2 | 72.32±0.6 | 72.40±0.3 |
| Ours     | 72.39±0.5 | 74.13±0.3 | 73.39±0.6 |
| Prompt Sim | 72.15±0.3 | 73.94±0.2 | 72.99±0.4 |
| K-Means  | 70.98±0.2 | 73.16±0.3 | 72.25±0.3 |

**Table 4.** Ablation of prompt adaptation strategies for MVLPT.

| Model   | Source | Adaptation | Averaged ELEVATER |
|---------|--------|------------|-------------------|
| MCoOp   | M      | S          | 70.93±0.3          |
| MVPT    | M      | S          | 73.16±0.3          |
| MUPT    | M      | S          | 72.25±0.5          |
| MCoOp   | Best M | S          | 72.39±0.5          |
| MVPT    | Best M | S          | 74.13±0.3          |
| MUPT    | Best M | S          | 73.39±0.6          |
| MCoOp   | Worst M| S          | 70.13±0.7          |
| MVPT    | Worst M| S          | 71.81±0.2          |
| MUPT    | Worst M| S          | 69.94±1.0          |

**Table 5.** Ablation of source tasks for MCoOp, MVPT and MVLPT.

| Model   | Source | Adaptation | Averaged 12 target tasks |
|---------|--------|------------|--------------------------|
| CoOp    | ImageNet1K | S      | 66.36±0.5                 |
| VPT     | ImageNet1K | S      | 68.80±0.9                 |
| UPT     | ImageNet1K | S      | 67.45±0.7                 |
| MCoOp   | 10 source tasks | S    | 66.51±0.5                 |
| MVPT    | 10 source tasks | S    | 70.31±1.1                 |
| MUPT    | 10 source tasks | S    | 70.08±0.9                 |
| MCoOp   | 11 source tasks | S    | 67.60±0.5                 |
| MVPT    | 11 source tasks | S    | 71.73±0.6                 |
| MUPT    | 11 source tasks | S    | 72.15±0.7                 |

**Figure 5.** Ablation on context length.

In this paper, we propose multitask vision-language prompt learning (MVLPT). We demonstrate that MVLPT exhibits strong generalizability and few-shot learning performance compared to baseline prompt learning methods. The most performant MVLPT sets the new state-of-the-art performance on the ELEVATER benchmark. We also study task transferability across 20 vision tasks and provide a guideline for multitask prompt learning.

**References**

[1] FER 2013: Kaggle challenges in representation learning facial expression recognition. https://www.kaggle.
[34] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019. 6

[35] Olivier J. Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, S. M. Ali Eslami, and Airon van den Oord. Data-efficient image recognition with contrastive predictive coding. In *ICML*, 2020. 3

[36] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *ICML*, pages 2790–2799. PMLR, 2019. 1

[37] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *NeurIPS*, 2021. 1

[38] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 1, 3

[39] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022. 1, 2, 3, 4, 6, 7

[40] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *ACL*, 2020. 3

[41] Armand Joulin, Laurens Van Der Maaten, Allan Jabri, and Nicolas Vasilache. Learning visual features from large weakly supervised data. In *ECCV*, 2016. 2

[42] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. *arXiv preprint arXiv:2112.04478*, 2021. 3

[43] Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. *NeurIPS*, 2020. 6

[44] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, pages 5583–5594. PMLR, 2021. 2

[45] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV-W*, 2013. 6

[46] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[47] Jimmy Lei Ba, Kevin Swersky, Sanja Fidler, et al. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *ICCV*, 2015. 2

[48] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 1, 3

[49] Ang Li, Allan Jabri, Armand Joulin, and Laurens van der Maaten. Learning visual n-grams from web data. In *ICCV*, 2017. 2

[50] Chunyuan Li, Haotian Liu, Liunian Harold Li, Pengchuan Zhang, Jyoti Aneja, Jianwei Yang, Ping Jin, Yong Jae Lee, Houdong Hu, Zicheng Liu, et al. Elevator: A benchmark and toolkit for evaluating language-augmented visual models. In *NeurIPS*, 2022. 1, 2, 5, 6

[51] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 2

[52] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafig Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*, volume 34, pages 9694–9705, 2021. 2

[53] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Shi, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*, 2020. 2

[54] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL*, 2021. 1, 3

[55] Yangguang Li, Feng Liang, Lichen Zhao, Wanyi Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*, 2021. 2

[56] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroyuki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021. 3

[57] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxia Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *ACL*, pages 61–68, 2022. 3

[58] Xiao Liu, Yanan Zheng, Zhengxiao Du, Zhilin Yang, and Jie Tang. Gpt understands, too. In *ICML*, 2022. 2

[59] Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. In *ICML*, 2021. 1, 3

[60] Haokun Liu, Derek Tam, Mohammed Muqeeeth, Jay Motha, Teng Hao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *NeurIPS*, 2022. 2

[61] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroyuki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021. 3

[62] Yao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhongxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *ACL*, pages 61–68, 2022. 3

[63] Xiaofei Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. In *ICML*, 2021. 1, 3

[64] Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *NeurIPS*, pages 5206–5215, 2022. 3

[65] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 6

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. In ICLR, 2022.

Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Cpt: Colorful prompt tuning for pre-trained vision-language models. arXiv preprint arXiv:2109.11797, 2021.

Jiahui Yu, Ziping Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. arXiv preprint arXiv:2205.01917, 2022.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. arXiv preprint arXiv:2111.11432, 2021.

Yuhang Zang, Wei Li, Kaiyang Zhou, Chen Huang, and Chen Change Loy. Unified vision and language prompt learning. arXiv preprint arXiv:2210.07225, 2022.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In CVPR, 2021.

Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. arXiv preprint arXiv:2112.02413, 2021.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. arXiv preprint arXiv:2205.01068, 2022.

Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. arXiv preprint arXiv:2010.00747, 2020.

Yu Zhang and Qiang Yang. A survey on multi-task learning. TKDE, 2021.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In ICML, pages 12697–12706. PMLR, 2021.

Zexuan Zhong, Dan Friedman, and Danqi Chen. Factual probing is [mask]: Learning vs. learning to recall. In NAACL, 2021.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. arXiv preprint arXiv:2109.01134, 2021.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In CVPR, pages 16816–16825, 2022.