Joint Scheduling of Causal Prompts and Tasks for Multi-Task Learning

Chaoyang Li[1,2], Jianyang Qin[1], Jinhao Cui[1], Zeyu Liu[1], Ning Hu[2], Qing Liao[1,2*]
[1]Harbin Institute of Technology, Shenzhen, China
[2]Peng Cheng Laboratory, Shenzhen, China
{lichy, hun}@pcl.ac.cn, {22b351005, cuijinhao, liuzeyu}@stu.hit.edu.cn, liaoqing@hit.edu.cn

Abstract

Multi-task prompt learning has emerged as a promising technique for fine-tuning pre-trained Vision-Language Models (VLMs) to various downstream tasks. However, existing methods ignore challenges caused by spurious correlations and dynamic task relationships, which may reduce the model performance. To tackle these challenges, we propose JSCPT, a novel approach for Joint Scheduling of Causal Prompts and Tasks to enhance multi-task prompt learning. Specifically, we first design a Multi-Task Vison-Language Prompt (MTVLP) model, which learns task-shared and task-specific vison-language prompts and selects useful prompt features via causal intervention, alleviating spurious correlations. Then, we propose the task-prompt scheduler that models inter-task affinities and assesses the causal effect of prompt features to optimize the multi-task prompt learning process. Finally, we formulate the scheduler and the multi-task prompt learning process as a bi-level optimization problem to optimize prompts and tasks adaptively. In the lower optimization, MTVLP is updated with the scheduled gradient, while in the upper optimization, the scheduler is updated with the implicit gradient. Extensive experiments show the superiority of our proposed JSCPT approach over several baselines in terms of multi-task prompt learning for pre-trained VLMs.

1. Introduction

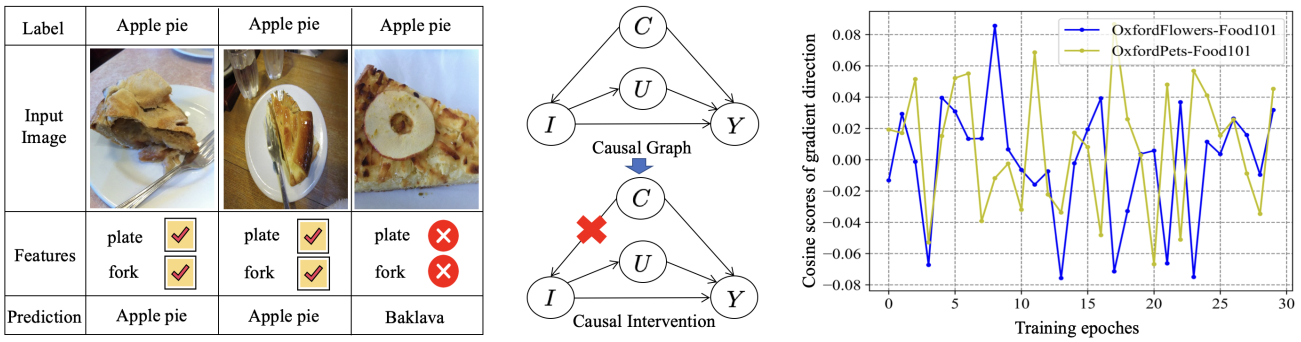
Pre-trained vision-language models (VLMs) [22, 52, 55], such as CLIP [34] and ALIGN [15], significantly enhance image-text matching by aligning their embeddings from vast web data [6, 14], which provide new insights for solving various downstream visual tasks. However, fine-tuning the full VLMs may forget useful knowledge acquired in the pre-training and can pose the risk of overfitting downstream tasks [19]. To address this, prompt tuning [16, 19, 61–63] is proposed to modify embeddings of pre-trained VLMs by learning soft prompts, encouraging the pre-trained VLMs better fitting to downstream tasks.

Early prompt learning methods for VLMs [16, 19, 61–63] can enhance individual visual task performance with limited training data. Recognizing that multi-task learning (MTL) can boost performance and efficiency over single-task models, researchers have explored prompt learning from VLMs to multi-task settings [24, 42, 47]. For example, HiPro [24] builds a hierarchical task prompt tree to group tasks and extract multi-grained task information. MVLPT [42] uses task-shared multi-modal prompts for multi-task transfer, while MmAP [47] trains group-shared and task-specific cross-modal aligned prompts for MTL. However, existing multi-task vision-language prompt-tuning methods may be suboptimal, as they overlook the challenges posed by spurious correlations and dynamic task relationships.

(1) Spurious correlations: Fine-tuning pre-trained VLMs on downstream tasks within specific domains can be challenging due to spurious correlations arising from domain-specific biases, which may cause the model to learn useless features and ultimately degrade performance. [33, 34, 50]. For instance, Fig. 1(a) shows a failed MVLPT [42] test case caused by such spurious correlations. In the training set, plates and forks frequently appear with apple pie, creating spurious correlations between these features (i.e., the plate and the fork) and the label. Consequently, the model misclassifies real apple pie images (without the plate or the fork) as baklava. To identify the causes of spurious correlations and suggest solutions, we construct a causal graph (Fig. 1(b)), where I, U, Y, and C represent input images, useful features, labels, and confounders, respectively. Here, $I \rightarrow Y$ denotes the desired causal effect, enabling the model to predict label Y directly from image I. However, not all features in I are useful (e.g., plates and forks). Confounders C, stemming from data bias, can interfere between I and Y, creating spurious correlations with useless features [20]. To address this, we aim to mitigate confounder interference via causal intervention, extracting useful features U to learn beneficial causal effects (i.e., $I \rightarrow U \rightarrow Y$).

(2) Dynamic task relationships: Although pre-grouping

*Corresponding author



(a) Spurious correlations in Food101.

(b) Causal graph and causal intervention.

(c) Gradient direction similarity.

Figure 1. The observation experiment of MVLPT [42] on a three-task dataset (i.e., Food101 [2], OxforsFlowers [29], and OxfordPets [31], under 20-shots setting) and the causal analysis. (a) shows that spurious correlations between the label and the useless features (i.e., plate and fork) misled the model to incorrectly predict an apple pie (without a plate and fork) as a baklava. (b) depicts the causal graph and intervention, where I, U, Y, and C represent input images, useful features, labels, and confounders, respectively. The confounders C from data bias create spurious correlations between I and Y, and causal intervention is to eliminate the interference from C to I. (c) illustrates the cosine score variation of gradient directions for shared parameters between the Food101 task and the other two tasks, where positive values indicate the positive correlations among tasks, while negative values suggest task conflicts.

tasks can alleviate task conflicts to a certain extent (such as HiPro [47] and MmAP [47]), they ignore the dynamic changes in task relationships, which hinders efficient global convergence of MTL [43]. Studies in MTL show that gradients of shared parameters can reveal task relationships [23, 54]. Positive gradient similarity among tasks indicates that multiple tasks are being solved along similar directions in the shared parameter space, suggesting mutually beneficial relationships, while negative similarity signals task conflicts. Fig. 1(c) shows changing task gradient similarities in MVLPT [42] on a three-task dataset, highlighting the need to perceive dynamic task relationships to mitigate task conflicts adaptively.

Spurious correlations and dynamic task relationships increase the difficulty of multi-task vision-language prompt learning. To address this, we explore a novel joint scheduling framework for causal prompts and tasks, prioritizing prompt features with fewer spurious correlations and tasks with fewer conflicts. Specifically, the *Multi-Task Vision-Language Prompt* (MTVLP) model is designed to learn task-shared and task-specific vision-language prompts, selecting useful prompt features via causal intervention. We then propose a task-prompt scheduler that guides multi-task prompt learning by assessing inter-task affinity and the causal effects of prompt features. Finally, bi-level optimization is to solve the optimal schedule, with the lower level updating MTVLP using scheduled gradients and the upper level updating the scheduler using implicit gradients. The contributions of this paper are summarised as follows:

- We propose JSCPT, a novel bi-level optimization-based scheduling framework for causal prompts and tasks, enhancing multi-task vision-language prompt learning by mitigating spurious correlations and task conflicts.
- We propose the task-prompt scheduler that perceives inter-task affinity and causal effects of prompt features, and adaptively assigns optimal learning weights to improve the multi-task joint training.
- Extensive experiments show the superiority of our proposed JSCPT over several baselines in terms of multi-task vision-language prompt learning.

2. Related Work

2.1. Multi-Task Learning

Multi-task learning (MTL) aims to improve efficiency and generalization performance through knowledge sharing across various tasks [3, 58]. Traditional MTL methods can be divided into hard parameter sharing [3, 48], soft parameter sharing [8, 26], and learnable sharing mechanisms [12, 37], to share knowledge across tasks. However, training different tasks on a unified model may pose task conflicts and negative transfer. Several studies aim to find optimal task groupings that benefit each other during joint training [9, 43, 44]. Furthermore, other methods strive to optimize task gradients to alleviate potential conflicts [7, 23, 28, 54]. Our proposed approach, which steers a frozen VLM to tackle diverse tasks by prompt learning, embodies the efficient multi-task learner.

2.2. Vision-Language Models (VLMs)

Pre-trained VLMs like CLIP [34] and ALIGN [15] have revolutionized the field by learning an aligned embedding space for text and images through contrastive learning on massive text-image pairs. These methods have shown exceptional transferability to downstream tasks, motivating extensive research into pre-training VLMs [1, 22, 27, 49, 52, 53]. Currently, CLIP stands as a prominent publicly accessible vision-language model, which has shown immense

potential in tackling diverse visual tasks by leveraging linguistic priors, including object detection [11, 59], image segmentation [21, 36], and visual recognition [16, 46].

2.3. Prompt Learning

Prompt learning is initially proposed to adapt large pre-trained language models (LMs) for NLP tasks [13, 17, 35]. With the rise of pre-trained VLMs, prompt-based methods have achieved notable success in various visual tasks [11, 16]. For instance, CoOp [62] generates task-specific prompts to enhance vision recognition. CoCoOp [61] explores prompt generalization for unseen classes but falls short of CoOp in in-distribution performance. VPT [16] adds a small number of trainable parameters to the visual input space while keeping the backbone frozen. MaPLe [19] focuses on improving vision-text alignment via prompt learning. To enable multi-task learning with prompts, HiPro [24] organizes tasks into a hierarchical tree based on gradient direction similarity, while MVLPT [42] shares knowledge via multi-modal prompts. MmAP [47] groups tasks based on gradient direction similarity and aligns prompts with task-specific modalities. Although task grouping can help reduce task conflicts [24, 47], it overlooks the challenges of dynamic task relationships and spurious correlations. In contrast, we propose a joint scheduling approach for causal prompts and tasks to adaptively eliminate spurious correlations and resolve task conflicts.

2.4. Causal Learning

Causal learning has gained considerable attention in computer vision for its potential to improve performance by exploring causal relationships instead of relying on statistical correlations. Early work on causal learning [32] introduced structural causal models, using probabilistic inference for causal intervention to mitigate spurious correlations [40]. Another approach involves generating counterfactuals via deigning mask [5], while neural networks can predict counterfactual outcomes [18] and estimate causal effects from observational data [38, 39]. Recent advancements have focused on incorporating counterfactual reasoning into computer vision models [30, 56], using representation learning to estimate cross-domain treatment effects [51].

3. Method

The framework of our proposed JSCPT is shown in Fig. 2. First, the multi-task vison-language prompt model learns multi-task prompts and uses a causal intervention network with learnable masks to select useful prompt features. Second, the task-prompt scheduler can assign better learnable weights for tasks based on the joint impact of prompt features (i.e., causal effect) and task relationships (i.e., inter-task affinity), achieving better multi-task prompt learning. Finally, bi-level optimization is to find optimal solutions in joint scheduling for prompts and tasks.

3.1. Prelimiaries

CLIP. CLIP [34] is a widely used VLM that enhances zero-shot classification performance via training image encoder E_V and text encoder E_T. For zero-shot classification, given an image $I \in \mathbb{R}^{3\times h\times w}$ and its text prompt T (e.g., "a photo of a [CLASS]") incorporating class labels $y \in \{1, 2, ..., C\}$, where C is the number of classes, we can obtain the visual embedding $x = E_V(I)$ and the textual embedding $z_{\hat{y}} = E_T(T)$. The prediction $\hat{y}$ for the image I is determined by the class with the highest cosine similarity score (cosine $(\cdot)$), formulated as follows:

$$p(\hat{y}|x) = \frac{\exp\left(\text{cosine}\left(x, z_{\hat{y}}\right)/\tau\right)}{\sum_{i=1}^{C}\exp\left(\text{cosine}\left(x, z_i\right)/\tau\right)}, \tag{1}$$

where τ is the temperature parameter.

Text Prompt Tuning. Text prompt tuning [62] can adapt pre-trained VLMs to downstream tasks by training a learnable text token vector $P_l = [P_{l,1}, ..., P_{l,n}]$ that replaces context words of text prompt, where $P_l \in \mathbb{R}^{d\times n}$, d and n are the dimensions and number of tokens, respectively. Only P_l and the class token [CLASS] are optimized with task objectives while the image and text encoders are frozen.

Visual Prompt Tuning. Visual prompt tuning [16] finetunes pretrained VLMs by adding a tunable vector $P_v = [P_{v,1}, ..., P_{v,n}]$ to input of each transformer layer in the image encoder, where $P_v \in \mathbb{R}^{d\times n}$, d and n are the dimensions and number of tokens, respectively. The resultal input $q = [\text{CLS}, P_v, V]$ comprises a classification token CLS, the tunable vector P_v and m patchified image tokens $V = [V_1, ..., V_m]$.

Granger-causal Objective. Granger-causal objective [38, 39] aims to quantify the causal effect of an input feature on the performance of model f. Given an input image $I = \{I_i\}_{i=1}^{m}$ with m patch features, the model prediction is $\hat{y} = f(I)$. $I_{\neg i}$ represents the input without the i-th patch feature, and the corresponding prediction is $\hat{y}_{\neg i} = f(I_{\neg i})$. The causal effect of the i-th patch feature is measured by the loss difference when the feature is included or excluded, defined as $\Delta(I_i) = \mathcal{L}(y, \hat{y}_{\neg i}) - \mathcal{L}(y, \hat{y})$, where $\mathcal{L}$ is the loss function and y is the ground-truth label. Here, $\Delta(I_i)$ indicates the causal effect of the i-th feature on the prediction $\hat{y}$ as the reduction in loss when the feature is included.

3.2. Multi-Task Vison-Language Prompt Model

In this subsection, we propose the multi-task vison-language prompt (MTVLP) model to learn vison-language prompts for multiple tasks and select useful features.

Multi-Task Vison-Language Prompt. Given N downstream tasks $\mathcal{T} = \{\mathcal{T}_i\}_{i=1}^{N}$, to share knowledge across different tasks and learn unique task characteristics, we

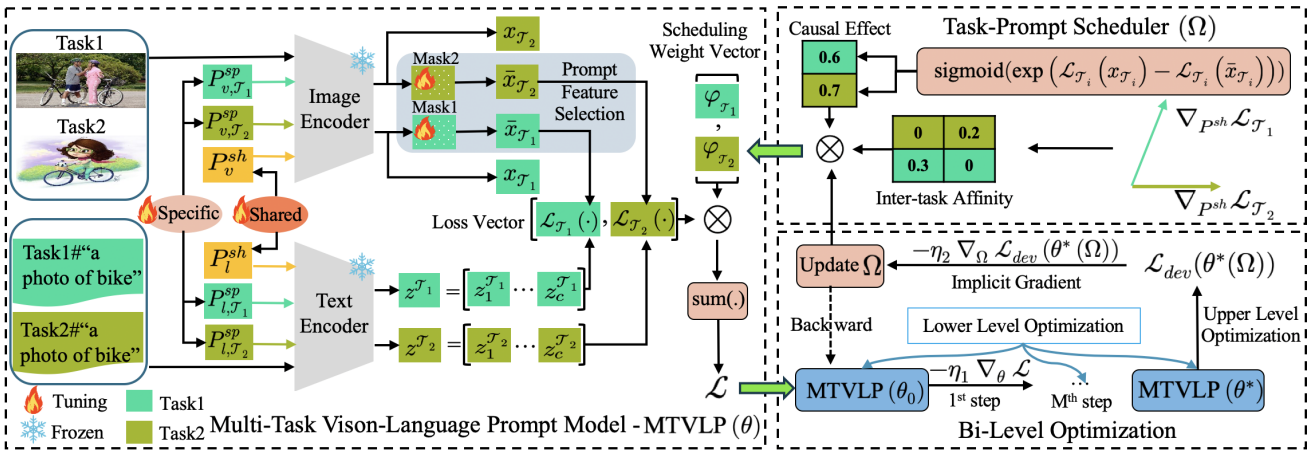


Figure 2. The framework of JSCPT. The MTVLP model trains task-shared and task-specific prompts, using task-specific learnable masks to select useful features. The task-prompt scheduler adaptively arranges learnable task weights via perceiving the causal effect and inter-task affinity. The bi-level optimization solves the optimal scheduling weights.

initialize one task-shared vison-language prompt $P^{sh} = (P_v^{sh}, P_l^{sh})$ and N task-specific vison-language prompts $P^{sp} = \{(P_{v,\mathcal{T}_i}^{sp}, P_{l,\mathcal{T}_i}^{sp})\}_{\mathcal{T}_i \in \mathcal{T}}$. Here, P_l^{sh} and $P_{l,\mathcal{T}_i}^{sp}$ have the same tokens as P_l, representing task-shared and task-specific prompts for text, respectively. P_v^{sh} and $P_{v,\mathcal{T}_i}^{sp}$ have the same tokens as P_v, denoting task-shared and task-specific prompts for the image, respectively.

For task $\mathcal{T}_i$, we take task-shared and task-specific prompts as additional inputs to the text and image encoders. Specifically, for the text encoder, the concatenated text prompt $P_{l,\mathcal{T}_i} = \left[P_l^{sh}, P_{l,\mathcal{T}_i}^{sp}, \text{CLASS}\right]$ is fed into each transformer block. The embedding of the [CLASS] token produced by the transformer stack is denoted as $\mathcal{W}_{[\text{CLASS}]}$, which is then projected into the vision-language latent space by the text projection head (i.e., TextHead) to obtain the final text features $z_{\hat{y}} = \text{TextHead}\left(\mathcal{W}_{[\text{CLASS}]}\right)$. Similarly, for the image encoder, $P_{v,\mathcal{T}_i} = \left[\text{CLS}, P_v^{sh}, P_{v,\mathcal{T}_i}^{sp}, V\right]$ is concatenated by [CLS] token, visual prompts, and image tokens V, which is input into each transformer block. The embedding of the [CLS] token from the complete transformer sequence output is denoted as $\mathcal{V}_{[\text{CLS}]}$, which is then projected into the vision-language latent space by the image projection head (i.e.,ImageHead) to produce the final visual features $x_{\mathcal{T}_i} = \text{ImageHead}\left(\mathcal{V}_{[\text{CLS}]}\right)$.

Prompt Feature Selection. To mitigate the spurious correlations and learn useful visual features, we design a causal intervention network with N task-specific learnable masks $\mathcal{M} = \{\mathcal{M}_i\}_i^N$ with the same size as $x_{\mathcal{T}_i}$. The element values of masks belong to $(0, 1)$, which can filter the visual prompt features by training its learnable parameters and reduce the confounders in the original feature space [5, 57]. For the task $\mathcal{T}_i$, we can obtain its counterfactual visual prompt features $\bar{x}_{\mathcal{T}_i}$ through causal intervention:

$$\bar{x}_{\mathcal{T}_i} = \text{MLP}(\mathcal{M}_i * x_{\mathcal{T}_i} + \text{MLP}\left(\mathcal{M}_i * x_{\mathcal{T}_i}\right)), \tag{2}$$

where MLP is the multi-layer perception.

Given task $\mathcal{T}_i$, the causal intervention network is expected to select useful visual features that help predict correct labels. Hence, we use the contrastive objective as the task loss to associate selected visual features with text labels, as follows:

$$\mathcal{L}_{\mathcal{T}_i}\left(\bar{x}_{\mathcal{T}_i}\right) = -\log \frac{\exp\left(\text{cosine}\left(\bar{x}_{\mathcal{T}_i}, z_{\hat{y}}\right)/\tau\right)}{\sum_{c=1}^{C} \exp\left(\text{cosine}\left(\bar{x}_{\mathcal{T}_i}, z_c\right)/\tau\right)}. \tag{3}$$

Training Eq. (3) enables the model to mine useful visual prompt features and learn beneficial causal effects (i.e., $I \rightarrow U \rightarrow Y$ of the causal graph in the Fig. 1). Finally, we can optimize all the task losses on the selected features to improve multi-task prompt learning.

3.3. Task-Prompt Scheduler

Although we have implemented feature selection, simply summing task losses in multi-task prompt learning, as done in [42, 47], may lead to suboptimal results. This way ignores the challenges of feature selection, i.e., samples with different levels of spurious correlations impact the model differently. Samples with fewer spurious correlations make it easier to select useful features, accelerating model convergence, and vice versa. Additionally, it fails to consider the dynamic task relationships during joint training. To address these issues, we propose a task-prompt scheduler that adaptively weights task losses based on the causal effects of prompt features, and inter-task affinity.

Causal Effect of Prompt. Inspired by the Granger-causal objective [38, 39], we quantify the causal effect of selected prompt features by calculating the loss reduction

from counterfactual visual prompt features. Given the original and counterfactual views of visual prompt features for task $\mathcal{T}_i$ (i.e., $x_{\mathcal{T}_i}$ and $\bar{x}_{\mathcal{T}_i}$, respectively), the causal effect of the selected visual prompt features is:

$$\Delta\left(x_{\mathcal{T}_i}\right)=\sigma(\exp\left(\mathcal{L}_{\mathcal{T}_i}\left(x_{\mathcal{T}_i}\right)-\mathcal{L}_{\mathcal{T}_i}\left(\bar{x}_{\mathcal{T}_i}\right)\right)), \tag{4}$$

where σ is the sigmoid function and $\Delta\left(x_{\mathcal{T}_i}\right) \in (0,1)$. When $\Delta\left(x_{\mathcal{T}_i}\right)>0.5$, it indicates that compared with $x_{\mathcal{T}_i}$, $\bar{x}_{\mathcal{T}_i}$ can generate smaller task loss, depicting that the selected features have a positive impact to correct prediction. When $\Delta\left(x_{\mathcal{T}_i}\right)<0.5$, the selected features have a negative impact to task. As the model is trained, the value of the causal effect is expected to keep increasing (or at least not decline), such that the model can learn more useful features.

Inter-task Affinity. To model task relationships, a feasible approach is to estimate the affinity between task gradients [23, 54]. However, existing multi-task prompt methods [24, 47] focus solely on using gradient direction for relationship modeling, ignoring gradient magnitude information [54]. To address this, we propose a new gradient-based inter-task affinity, denoted as A, to perceive the task relationships by considering both gradient direction and magnitude. Specifically, the inter-task affinity from source task $\mathcal{T}_i$ to target task $\mathcal{T}_j$ can be calculated based on the task losses (i.e., $\mathcal{L}_{\mathcal{T}_i}\left(\bar{x}_{\mathcal{T}_i}\right)$ and $\mathcal{L}_{\mathcal{T}_j}\left(\bar{x}_{\mathcal{T}_j}\right)$) and their gradients with respect to the shared prompt (i.e., $g_{\mathcal{T}_i}^{sh}=\nabla_{P^{sh}}\mathcal{L}_{\mathcal{T}_i}\left(\bar{x}_{\mathcal{T}_i}\right)$ and $g_{\mathcal{T}_j}^{sh}=\nabla_{P^{sh}}\mathcal{L}_{\mathcal{T}_j}\left(\bar{x}_{\mathcal{T}_j}\right)$), as follows,

$$A_{i,j}=\frac{\nabla_{P^{sh}}\mathcal{L}_{\mathcal{T}_i}\left(\bar{x}_{\mathcal{T}_i}\right).\nabla_{P^{sh}}\mathcal{L}_{\mathcal{T}_j}\left(\bar{x}_{\mathcal{T}_j}\right)}{\parallel\nabla_{P^{sh}}\mathcal{L}_{\mathcal{T}_j}\left(\bar{x}_{\mathcal{T}_j}\right)\parallel^2}, \tag{5}$$

where $A_{i,j}\in[-1,1]$, the numerator is the similarity of gradient directions between tasks, and the denominator is the L2 norm of the gradient magnitude of the target task. The larger the positive value of $A_{i,j}$, the greater the positive impact of task $\mathcal{T}_i$ on task $\mathcal{T}_j$, while a negative value indicates a negative impact. Note that A is an asymmetric matrix with zeros on the diagonal, as the affinity $A_{j,i}$ differs from $A_{i,j}$ due to a different denominator.

Joint Scheduling Objective. To jointly schedule the prompt features and tasks, we design the learnable joint scheduling weights $\varphi_{b,\mathcal{T}_i}$ for the task $\mathcal{T}_i$ that considers both inter-task affinity and causal effect of prompt features,

$$\varphi_{b,\mathcal{T}_i}=\sigma(\alpha_{\mathcal{T}_i}\cdot\Delta\left(x_{\mathcal{T}_i}\right))\cdot\sigma\left(\beta_{\mathcal{T}_i}\cdot A_{:,\mathcal{T}_i}\right), \tag{6}$$

where b is the batch index of data set, σ denotes the sigmoid function, $\alpha_{\mathcal{T}_i}$ and $\beta_{\mathcal{T}_i}$ are the learnable parameters. $\varphi_{b,\mathcal{T}_i}$ aim to enable tasks to assign greater learning weights to useful prompt features (with greater causal effects) and positively correlated tasks (with greater inter-task affinity) while assigning smaller learning weights to useless features and negatively correlated tasks. Finally, combining task scheduling weights $\varphi_{b,\mathcal{T}_i}$ with Eq. (3), the joint scheduling objective for MTL can be formulated as follows:

$$\mathcal{L}=\sum_{\mathcal{T}_i\in\mathcal{T}}\sum_{b\in D_{train}}\varphi_{b,\mathcal{T}_i}\cdot\mathcal{L}_{\mathcal{T}_i}\left(\bar{x}_{\mathcal{T}_i,b}\right), \tag{7}$$

where D_{train} denotes the training set of task $\mathcal{T}_i$.

3.4. Bi-level Optimization

The task-prompt scheduler optimizes the learnable parameters $\Omega=\{\alpha_{\mathcal{T}_i},\beta_{\mathcal{T}_i}\}_{\mathcal{T}_i\in\mathcal{T}}$ to achieve better multi-task joint training by minimizing Eq. (7). To achieve this, we introduce a small developing dataset $D_{dev}=\left\{\left(x_b^{dev},y_b^{dev}\right)\right\}_b^B$, which is a small subset sampled from the validation set D_v [4]. The scheduler aims to obtain an optimal multi-task training schedule on D_v, and the loss on D_{dev} is used to update the parameters Ω. Formally, our problem is formulated as a bi-level optimization problem, as follows:

$$\begin{aligned}\Omega^*&=\arg\min_{\Omega}\mathcal{L}_{dev}\left(\theta^*\left(\Omega\right)\right),\\ s.t.\ \theta^*&=\arg\min_{\theta}\mathcal{L}\left(\theta,\Omega\right),\end{aligned} \tag{8}$$

where $\mathcal{L}_{dev}\left(\theta^*\left(\Omega\right)\right)=\sum_{\mathcal{T}_i\in\mathcal{T}}\sum_{b\in D_{dev}}\mathcal{L}_{\mathcal{T}_i}\left(\bar{x}_{\mathcal{T}_i,b}\right)$ and $\mathcal{L}\left(\theta,\Omega\right)$ is the scheduled training loss in Eq. (7). It is noted that $\varphi_{b,\mathcal{T}_i}$ is parameterized by Ω.

In the lower-level optimization, we update the MTVLP parameter θ with the fixed parameter Ω. θ is updated by using the weighted gradient sum of the samples within different tasks as follows,

$$\nabla_\theta\mathcal{L}\left(\theta,\Omega\right)=\sum_{\mathcal{T}_i\in\mathcal{T}}\sum_{b\in D_{train}}\varphi_{b,\mathcal{T}_i}\cdot\nabla_\theta\mathcal{L}_{\mathcal{T}_i}\left(\bar{x}_{\mathcal{T}_i,b}\right). \tag{9}$$

In the upper-level optimization, it is necessary to compute the gradient $\mathcal{L}_{dev}\left(\theta^*(\Omega)\right)$ to Ω. Considering that $\mathcal{L}_{dev}\left(\theta^*(\Omega)\right)$ depends indirectly on Ω through θ, we apply implicit differentiation to obtain this implicit gradient [25]. Drawing inspiration from the **Cauchy-based Implicit Function Theorem** [25] (the derivation details are in Appendix), we can leverage the chain rule to systematically derive the gradient of $\mathcal{L}_{dev}\left(\theta^*(\Omega)\right)$ to Ω,

$$\begin{aligned}&\nabla_\Omega\mathcal{L}_{dev}\left(\theta^*(\Omega)\right)=\nabla_\theta\mathcal{L}_{dev}\cdot\nabla_\Omega\theta^*\\&=-\nabla_\theta\mathcal{L}_{dev}\cdot\left(\nabla_\theta^2\mathcal{L}\right)^{-1}\cdot\nabla_\Omega\nabla_\theta\mathcal{L}|_{(\Omega,\theta^*(\Omega))}.\end{aligned} \tag{10}$$

In Eq. (10), the inverse of the Hessian matrix $\left(\nabla_\theta^2\mathcal{L}\right)^{-1}$ for deep neural network is typically vast and complex, leading to impractical computation complexity. To address this, we adopt the **H-truncated Neumann** series [4] to approximate $\left(\nabla_\theta^2\mathcal{L}\right)^{-1}$ rather than directly computing it. Thus, Eq. (10) can be reformulated as follows,

$$\begin{aligned}&\nabla_\Omega\mathcal{L}_{dev}\left(\theta^*(\Omega)\right)\\&\approx-\nabla_\theta\mathcal{L}_{dev}\cdot\sum_{j=0}^{H}\left(I-\nabla_\theta^2\mathcal{L}\right)^j\cdot\nabla_\Omega\nabla_\theta\mathcal{L}|_{(\Omega,\theta^*(\Omega))},\end{aligned} \tag{11}$$

where I is the identity matrix and approximation details are provided in Appendix.

Algorithm 1 outlines the procedure for optimizing the MTVLP (θ) and scheduler (Ω) using their gradients. During the lower-level optimization, Ω is fixed while θ is updated using the gradient in Eq. (9) with learning rate η_1. Instead of aiming for full convergence of θ, we adopt an efficient M-step strategy [25], updating θ for M iterations before moving to the upper-level optimization for Ω. In the upper-level optimization, we first evaluate $\mathcal{L}_{dev}$ and then update Ω using the implicit gradient with learning rate η_2, as per Eq. (11). Given N tasks, the truncated Neumann series number as H, assuming M lower-level and 1 upper-level optimization iteration, the time complexity for the gradient backward pass of JSCPT is $O(N + (H + N)/M)$, as detailed in the Appendix. In practice, H is set to 3 [25] and M to 10.

Algorithm 1 JSCPT Algorithm

Input: datasets: D_{train}, D_{dev}; hyperparameters: H, M, η_1, η_2
Initialization: θ, Ω
while not converge **do**
  // lower-level optimization (update θ with fixed Ω)
  for $t = 0$ to $M - 1$ **do**
    Calculate the causal effect by Eq. (4)
    Calculate the inter-task affinity by Eq. (5)
    Update θ by $\theta = \theta - \eta_1 \nabla_\theta \mathcal{L}(\theta, \Omega)$
  end for
  // upper-level optimization (update Ω with current θ)
  Obtain the $\mathcal{L}_{dev}(\theta^*(\Omega))$ on D_{dev}
  Update Ω by $\Omega = \Omega - \eta_2 \nabla_\Omega \mathcal{L}_{dev}$
end while
Return θ_{opt} (the optimal θ)

4. Experiment

4.1. Experimental setup

Dataset. We conduct experiments on three multi-task datasets, including Office-Home [45], MiniDomainNet [60], and a large-scale multi-task learning benchmark with 10 visual datasets. Following previous MTL methods [41], we randomly select 10% and 20% samples from Office-Home, and 1% and 2% samples from MiniDomainNet for training. Following the splitting of [42], we sample 1, 5, 10, and 20 training samples of each class from the large-scale multi-task learning benchmark for training. We report the results of each dataset over 3 runs. Due to the limited space, more details of the datasets are in the Appendix.

Baselines. We compare JSCPT with 7 tuning baselines: (1) **Zero-Shot**; (2) **CoOp**; (3) **VPT** [16]; (4) **MaPLe** [19] ; (5) **CLIP-Adapter** [10]; (6) **MVLPT** [42] ;(7) **MmAP** [47]. Except for MVLPT and MmAP, the other methods

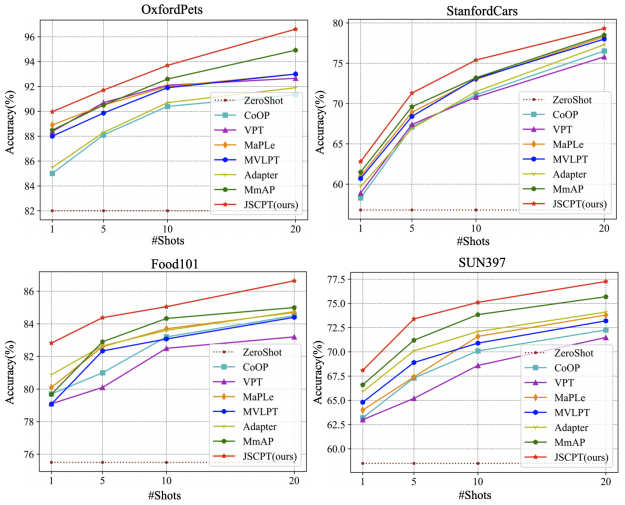


Figure 3. Comparison with accuracy(%) of various methods on the OxfordPets, StanfordCars, Food101, and SUN397 under the few-shot setting.

are mainly for single tasks. We build a multi-task version for the single-task method by training task-shared prompts or adapters. More details are in the Appendix.

Implementation details. All experiments use the PyTorch toolkit on 2 NVIDIA H800 and 4 NVIDIA V100 GPUs, with CLIP (ViT-B/16) as our default model. To guarantee fairness, we standardized the hyperparameter configurations across all methods. Specifically, we utilized a batch size of 16, 4, and 32 respectively for the Office-Home, MiniDomainNet, and Large-Scale MTL Benchmark datasets, and trained for 50 epochs. Additionally, we employ the SGD optimizer with a learning rate fixed at 0.0015 and set the number of prompt tokens to 16. The average number of parameters required to train a single task using our method is approximately 0.203MB, with the total training parameters accumulating as the number of tasks increases in multitask scenarios.

4.2. Main Results

Office-Home. Table 1 presents the results on the Office-Home dataset. Our JSCPT demonstrates obvious improvements across different splits compared to traditional baseline methods. Specifically, when compared with representative multi-task multi-modal prompt learning methods (i.e., MVLPT and MmAP), JSCPT not only achieves optimal results for each task but also outperforms the second-best method MmAP by an average accuracy of 0.8% under 10% and 20% data splits. This indicates the effectiveness of jointly scheduling prompts and tasks.

MiniDomainNet. The results on the MiniDomainNet dataset (Table 2) show that JSCPT continues to excel, achieving an accuracy of 85.1% on the 1% split and 86.4% on the 2% split. On the 1% split, JSCPT outperforms

Table 1. Comparison with accuracy(%) of various methods on Office-Home.

		Single-Task Learning					Multi-Task Learning						
	Method	Zero	CoOP	VPT	MaPLe	Adapter	CoOP	VPT	MaPLe	MVLPT	Adapter	MmAP	Ours
10%	Art	82.7	83.8±0.2	83.7±0.3	84.3±0.5	83.3±0.3	84.3±0.4	84.1±0.3	84.7±0.4	85.3±0.2	84.3±0.2	85.5±0.5	**86.4±0.2**
	Clipart	68.0	72.5±0.3	70.7±0.1	72.9±0.4	69.1±0.3	72.9±0.1	72.5±0.2	73.0±0.1	73.2±0.3	70.3±0.1	74.3±0.2	**75.5±0.3**
	Product	89.1	92.0±0.2	90.5±0.4	92.0±0.2	90.6±0.1	92.2±0.3	91.2±0.4	92.3±0.6	91.8±0.1	90.8±0.1	92.4±0.2	**92.8±0.1**
	Real	89.8	90.1±0.1	89.0±0.5	90.2±0.3	88.9±0.2	90.5±0.3	90.1±0.2	90.6±0.2	90.6±0.1	90.0±0.4	90.5±0.3	**91.1±0.4**
	Avg.	82.4	84.6±0.2	83.5±0.3	84.9±0.4	83.0±0.2	85.0±0.3	84.5±0.3	85.2±0.3	85.2±0.2	83.9±0.2	85.7±0.3	**86.5±0.3**
20%	Art	84.5	85.5±0.3	85.3±0.2	85.7±0.5	85.1±0.4	86.0±0.2	85.9±0.4	86.2±0.3	86.1±0.4	85.7±0.4	86.5±0.1	**87.4±0.3**
	Clipart	68.1	74.2±0.3	71.4±0.3	74.1±0.3	70.6±0.3	73.8±0.4	72.1±0.3	74.1±0.5	75.2±0.2	71.4±0.4	76.8±0.1	**77.9±0.3**
	Product	89.3	92.7±0.5	91.3±0.3	92.7±0.4	91.5±0.3	92.7±0.3	92.0±0.3	92.8±0.1	92.6±0.3	92.1±0.4	93.1±0.4	**93.6±0.4**
	Real	90.5	91.5±0.3	90.8±0.2	91.8±0.3	90.2±0.1	92.1±0.2	91.7±0.5	91.8±0.3	91.3±0.2	90.8±0.2	92.0±0.2	**92.7±0.2**
	Avg.	83.1	86.0±0.4	84.7±0.3	86.1±0.4	84.4±0.3	86.2±0.3	85.2±0.4	86.2±0.3	86.3±0.3	85.0±0.4	87.1±0.2	**87.9±0.3**

Table 2. Comparison with accuracy(%) of various methods on MiniDomainNet.

		Single-Task Learning					Multi-Task Learning						
	Method	Zero	CoOP	VPT	MaPLe	Adapter	CoOP	VPT	MaPLe	MVLPT	Adapter	MmAP	Ours
1%	Clipart	82.4	82.6±0.2	82.1±0.2	82.8±0.2	82.1±0.1	83.1±0.3	83.0±0.1	83.1±0.3	83.2±0.2	82.7±0.3	83.4±0.1	**84.1±0.2**
	Paint	82.3	81.7±0.2	81.4±0.2	82.0±0.1	80.5±0.3	82.0±0.2	81.3±0.4	82.2±0.2	82.4±0.3	80.9±0.2	82.9±0.1	**83.9±0.2**
	Real	91.0	91.7±0.3	91.3±0.2	91.8±0.2	90.8±0.3	90.7±0.3	90.1±0.1	91.1±0.3	91.4±0.2	90.8±0.1	91.7±0.3	**92.4±0.3**
	Sketch	79.7	77.4±0.1	78.2±0.2	78.4±0.3	77.7±0.4	79.1±0.4	78.5±0.3	78.9±0.4	79.3±0.3	78.2±0.4	79.6±0.2	**80.1±0.2**
	Avg.	83.9	83.4±0.2	83.3±0.2	83.8±0.2	82.8±0.3	83.7±0.3	83.2±0.2	83.8±0.3	84.1±0.2	83.2±0.2	84.4±0.2	**85.1±0.2**
2%	Clipart	82.5	83.6±0.3	83.5±0.2	83.7±0.3	83.0±0.3	84.3±0.3	84.2±0.4	84.2±0.3	84.7±0.3	83.3±0.1	85.0±0.4	**85.9±0.3**
	Paint	82.3	82.6±0.2	82.3±0.3	82.5±0.3	82.3±0.5	83.2±0.1	82.4±0.3	83.4±0.5	83.5±0.5	83.1±0.2	84.5±0.5	**85.3±0.1**
	Real	90.9	91.5±0.4	91.2±0.3	91.5±0.4	91.0±0.1	91.4±0.3	90.7±0.2	91.7±0.3	91.8±0.4	91.1±0.4	91.8±0.5	**92.8±0.2**
	Sketch	79.9	79.1±0.5	79.3±0.2	79.7±0.3	79.3±0.4	80.0±0.5	79.1±0.3	80.5±0.3	80.7±0.1	80.1±0.3	80.9±0.4	**81.5±0.5**
	Avg.	83.9	84.2±0.4	84.1±0.2	84.4±0.3	83.9±0.3	84.7±0.3	84.1±0.3	85.0±0.4	85.2±0.3	84.4±0.3	85.6±0.5	**86.4±0.3**

Table 3. Average accuracy(%) of 10 tasks in the Large-Scale MTL benchmark under k-shot setting.

Methods	k=1	k=5	k=10	k=20
ZeroShot	58.0	58.0	58.0	58.0
CoOP	64.1	71.3	75.7	78.8
VPT	66.0	73.3	76.7	79.6
MaPLe	67.9	74.8	78.4	81.7
MVLPT	67.8	74.7	77.6	81.1
Adapter	66.6	73.2	76.9	80.3
MmAP	68.7	75.9	79.3	83.0
Ours	**70.1**	**77.2**	**80.8**	**84.4**

the suboptimal baseline MmAP by 0.7%, and on the 2% split, it outperforms by 0.8%. This indicates that even in a more complex multi-task setting where training data remains scarce, JSCPT can still enhance multi-task transfer performance.

Large-Scale MTL benchmark. Table 3 displays the average accuracy of 10 tasks with varying sample sizes. We can see that JSCPT effectively enhances model performance. JSCPT achieves an accuracy of 70.1% with the 1-shot setting. As the number of shots increases, the effectiveness of JSCPT becomes more pronounced, surpassing the suboptimal method MVLPT by 1.4% under 20 shots. As shown in Fig. 3, JSCPT outperforms other methods on four datasets under few-shot settings (results for the remaining six datasets are provided in the Appendix). This demonstrates that JSCPT can enhance the performance of multi-task vision-language prompt tuning with limited data.

4.3. Ablation Study

We ablate three main components (i.e., feature selection, prompt scheduling, and task scheduling) of JSCPT to show their effectiveness. There are four settings: (1) **w/o Feature Selection**: select visual prompt features with fixed masks with values set to 0.5. (2) **w/o Prompt Scheduling**: do not calculate the causal effect of prompt features nor learn its scheduling weights. (3) **w/o Task Scheduling**: do not calculate the inter-task affinity nor learn its scheduling weights. (4) **MT-CAGrad**: use the representative multi-

task gradient optimization method (i.e. CAGrad [23]) to replace the task scheduling. Table 4 shows the ablation results on the Office-Home. Table 4 indicates that all three components contribute to improving the performance of multi-task prompt learning, with feature selection playing a more crucial role in particular. Meanwhile, our task scheduling is better than multi-task gradient optimization.

Table 4. Ablation results on the Office-Home, using the average accuracy(%) of four tasks.

Setting	10%	20%
w/o Feature Selection	85.4	87.0
w/o Prompt Scheduling	<u>86.1</u>	<u>87.3</u>
w/o Task Scheduling	85.9	87.2
MTL-CAGrad	85.6	87.1
Full Model	**86.5**	**87.9**

Effectiveness of Prompt Scheduling. To further illustrate the effect of prompt scheduling, we experimented by randomly selecting 16 batches of samples from the Office-Home (20%) training set. Fig. 4 shows the causal effects of prompt features and the prompt scheduling weights for the Art task across batches in the 1st and 20th training epochs, where the 16 values represent the causal effects and scheduling weights of prompt features corresponding to training samples in each of the 16 batches. As seen from Fig. 4, the causal effects of most prompt features gradually increase as training, indicating that JSCPT learns more causal features. The prompt features with larger causal effects receive greater scheduling weights, demonstrating that the model prioritizes learning causal features, which reflects the effectiveness of prompt scheduling.

Effectiveness of Task Scheduling. To demonstrate the role of task scheduling in mitigating task conflicts, we show the training losses of JSCPT and the naive multi-task prompt method (i.e., MVLPT) on Office-Home (20%). As shown in Fig. 5, MVLPT is prone to task conflicts, and the conflicts are dynamically changing. For example, in the fifth epoch, the loss of the Clipart increases, while the losses of the Art and Product tasks decrease, indicating that the Clipart task conflicts with the Art and Product tasks. In the tenth epoch, the loss of the Clipart task decreases, while the losses of Product and Real tasks increase, indicating a change in the task conflicts. It underscores the necessity of scheduling tasks via perceiving task conflicts adaptively. Fig. 5 also displays the normalized inter-task affinity in the fifth and tenth epochs, reflecting the actual inter-task relationships. Moreover, compared with MVLPT, JSCPT can alleviate task conflicts.

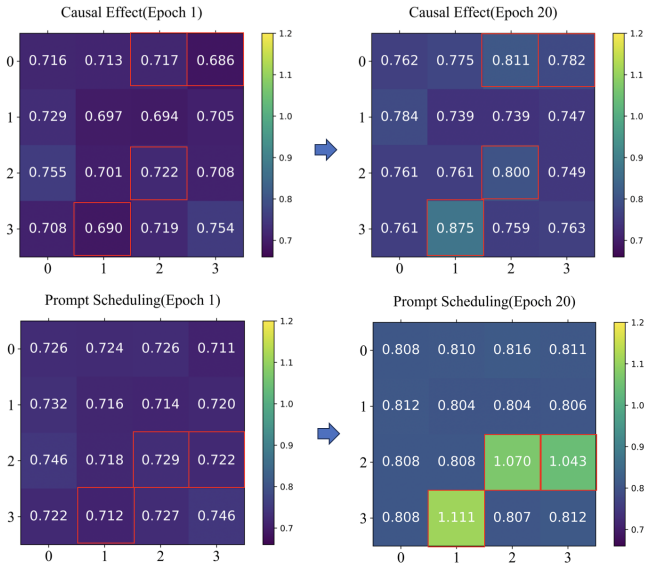


Figure 4. The causal effect and prompt scheduling for the Art task across 16 batch samples in Office-Home (20%), during the 1st and 20th training epochs.

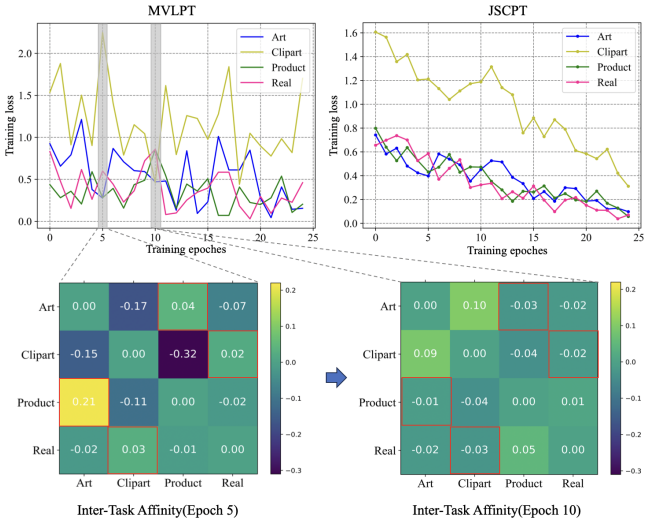


Figure 5. The task training loss and inter-task affinity on Office-Home (20%).

5. Conclusion

In this work, we propose JSCPT, an innovative optimization framework for scheduling causal prompts and tasks. This framework learns useful prompt features and adaptively mitigates task conflicts, enhancing pre-trained VLMs' adaptability to various downstream tasks. The extensive experiments indicate that our proposed JSCPT achieves significant performance improvements compared to the baselines on three large MTL datasets. In the future, we aim to apply causal ideas to prompt learning to improve generalization in task incremental settings.

6. Acknowledgement

This work was supported by the National Natural Science Foundation of China (Grant No. 62227808), the Shenzhen Science and Technology Program (Grant No. ZDSYS20210623091809029), and the Major Key Project of PCL (Grant No. PCL2024A05).

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems*, 2022. 2

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 - mining discriminative components with random forests. In *Computer Vision - ECCV 2014 - 13th European Conference*, pages 446–461. Springer, 2014. 2

[3] Rich Caruana. Multitask learning: A knowledge-based source of inductive bias. In *Machine Learning, Proceedings of the Tenth International Conference*, pages 41–48. Morgan Kaufmann, 1993. 2

[4] Hong Chen, Xin Wang, Chaoyu Guan, Yue Liu, and Wenwu Zhu. Auxiliary learning with joint task and data scheduling. In *International Conference on Machine Learning, ICML*, pages 3634–3647, 2022. 5

[5] Long Chen, Yuhang Zheng, Yulei Niu, Hanwang Zhang, and Jun Xiao. Counterfactual samples synthesizing and training for robust visual question answering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):13218–13234, 2023. 3, 4

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, pages 1597–1607. PMLR, 2020. 1

[7] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proceedings of the 35th International Conference on Machine Learning*, pages 793–802. PMLR, 2018. 2

[8] Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 845–850. The Association for Computer Linguistics, 2015. 2

[9] Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Efficiently identifying task groupings for multi-task learning. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems*, pages 27503–27516, 2021. 2

[10] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *Int. J. Comput. Vis.*, 132(2):581–595, 2024. 6

[11] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *The Tenth International Conference on Learning Representations*, 2022. 3

[12] Pengsheng Guo, Chen-Yu Lee, and Daniel Ulbricht. Learning to branch for multi-task learning. In *Proceedings of the 37th International Conference on Machine Learning*, pages 3854–3863. PMLR, 2020. 2

[13] Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. PTR: prompt tuning with rules for text classification. *AI Open*, 3:182–192, 2022. 3

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 9726–9735. Computer Vision Foundation / IEEE, 2020. 1

[15] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. 1, 2

[16] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge J. Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Computer Vision - ECCV 2022*, pages 709–727. Springer, 2022. 1, 3, 6

[17] Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. How can we know what language models know. *Trans. Assoc. Comput. Linguistics*, 8:423–438, 2020. 3

[18] Fredrik D. Johansson, Uri Shalit, and David Sontag. Learning representations for counterfactual inference. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 3020–3029, 2016. 3

[19] Muhammad Uzair Khattak, Hanoona Abdul Rasheed, Muhammad Maaz, Salman H. Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*, pages 19113–19122. IEEE, 2023. 1, 3, 6

[20] Junho Kim, Byung-Kwan Lee, and Yong Man Ro. Demystifying causal features on adversarial examples and causal inoculation for robust network by adversarial instrumental variable regression. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12032–12042, 2023. 1

[21] Boyi Li, Kilian Q. Weinberger, Serge J. Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *The Tenth International Conference on Learning Representations*, 2022. 3

[22] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. In *The Tenth International Conference on Learning Representations, ICLR 2022*, 2022. 1, 2

[23] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems*, pages 18878–18890, 2021. 2, 5, 8

[24] Yajing Liu, Yuning Lu, Hao Liu, Yaozu An, Zhuoran Xu, Zhuokun Yao, Baofeng Zhang, Zhiwei Xiong, and Chenguang Gui. Hierarchical prompt learning for multi-task learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 10888–10898, 2023. 1, 3, 5

[25] Jonathan Lorraine, Paul Vicol, and David Duvenaud. Optimizing millions of hyperparameters by implicit differentiation. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS*, pages 1540–1552. PMLR, 2020. 5, 6

[26] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3994–4003. IEEE Computer Society, 2016. 2

[27] Norman Mu, Alexander Kirillov, David A. Wagner, and Saining Xie. SLIP: self-supervision meets language-image pre-training. In *Computer Vision - ECCV 2022*, pages 529–544. Springer, 2022. 2

[28] Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In *International Conference on Machine Learning*, pages 16428–16446. PMLR, 2022. 2

[29] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Sixth Indian Conference on Computer Vision, Graphics & Image Processing, ICVGIP*, pages 722–729. IEEE Computer Society, 2008. 2

[30] Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12700–12710, 2021. 3

[31] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3498–3505. IEEE Computer Society, 2012. 2

[32] Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000. 3

[33] Suzanne Petryk, Lisa Dunlap, Keyan Nasseri, Joseph Gonzalez, Trevor Darrell, and Anna Rohrbach. On guiding visual attention with language specification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 18071–18081, 2022. 1

[34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, pages 8748–8763, 2021. 1, 2, 3

[35] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21: 140:1–140:67, 2020. 3

[36] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18061–18070. IEEE, 2022. 3

[37] Dripta S. Raychaudhuri, Yumin Suh, Samuel Schulter, Xiang Yu, Masoud Faraki, Amit K. Roy-Chowdhury, and Manmohan Chandraker. Controllable dynamic multi-task architectures. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10945–10954. IEEE, 2022. 2

[38] Patrick Schwab and Walter Karlen. Cxplain: Causal explanations for model interpretation under uncertainty. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems*, pages 10220–10230, 2019. 3, 4

[39] Patrick Schwab, Djordje Miladinovic, and Walter Karlen. Granger-causal attentive mixtures of experts: Learning important features with neural networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence*, pages 4846–4853, 2019. 3, 4

[40] Uri Shalit, Fredrik D. Johansson, and David Sontag. Estimating individual treatment effect: Generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 3076–3085, 2017. 3

[41] Jiayi Shen, Xiantong Zhen, Marcel Worring, and Ling Shao. Variational multi-task learning with gumbel-softmax priors. pages 21031–21042, 2021. 6

[42] Sheng Shen, Shijia Yang, Tianjun Zhang, Bohan Zhai, Joseph E. Gonzalez, Kurt Keutzer, and Trevor Darrell. Multitask vision-language prompt tuning. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV*, pages 5644–5655. IEEE, 2024. 1, 2, 3, 4, 6

[43] Xiaozhuang Song, Shun Zheng, Wei Cao, James J. Q. Yu, and Jiang Bian. Efficient and effective multi-task grouping via meta learning on task combinations. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems*, 2022. 2

[44] Trevor Standley, Amir Zamir, Dawn Chen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *Proceedings of the 37th International Conference on Machine Learning*, pages 9120–9132. PMLR, 2020. 2

[45] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for

unsupervised domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 5385–5394, 2017. 6

[46] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7949–7961. IEEE, 2022. 3

[47] Yi Xin, Junlong Du, Qiang Wang, Ke Yan, and Shouhong Ding. Mmap: Multi-modal alignment prompt for cross-domain multi-task learning. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024*, pages 16076–16084, 2024. 1, 2, 3, 4, 5, 6

[48] Enneng Yang, Junwei Pan, Ximei Wang, Haibin Yu, Li Shen, Xihua Chen, Lei Xiao, Jie Jiang, and Guibing Guo. Adatask: A task-aware adaptive learning rate approach to multi-task learning. In *Thirty-Seventh AAAI Conference on Artificial Intelligence*, pages 10745–10753, 2023. 2

[49] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Ce Liu, Lu Yuan, and Jianfeng Gao. Unified contrastive learning in image-text-label space. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 19141–19151. IEEE, 2022. 2

[50] Yu Yang, Besmira Nushi, Hamid Palangi, and Baharan Mirzasoleiman. Mitigating spurious correlations in multi-modal models during fine-tuning. In *International Conference on Machine Learning, ICML*, pages 39365–39379, 2023. 1

[51] Liuyi Yao, Sheng Li, Yaliang Li, Mengdi Huai, Jian Gao, and Aidong Zhang. Representation learning for treatment effect estimation from observational data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2638–2648, 2018. 3

[52] 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 *The Tenth International Conference on Learning Representations*, 2022. 1, 2

[53] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *Trans. Mach. Learn. Res.*, 2022. 2

[54] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems*, 2020. 2, 5

[55] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, Ce Liu, Mengchen Liu, Zicheng Liu, Yumao Lu, Yu Shi, Lijuan Wang, Jianfeng Wang, Bin Xiao, Zhen Xiao, Jianwei Yang, Michael Zeng, Luowei Zhou, and Pengchuan Zhang. Florence: A new foundation model for computer vision. *CoRR*, abs/2111.11432, 2021. 1

[56] Zhongqi Yue, Tan Wang, Qianru Sun, Xian-Sheng Hua, and Hanwang Zhang. Counterfactual zero-shot and open-set visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15404–15414, 2021. 3

[57] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS*, 2020. 4

[58] Zheng Zhang, Chuyao Luo, Baoquan Zhang, Hao Jiang, and Bowen Zhang. Multi-task framework of precipitation nowcasting. *CAAI Transactions on Intelligence Technology*, 8 (4):1350–1363, 2023. 2

[59] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, and Jianfeng Gao. Regionclip: Region-based language-image pretraining. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16772–16782. IEEE, 2022. 3

[60] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain adaptive ensemble learning. *IEEE Trans. Image Process.*, 30:8008–8018, 2021. 6

[61] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 16795–16804. IEEE, 2022. 1, 3

[62] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *Int. J. Comput. Vis.*, 130(9):2337–2348, 2022. 3

[63] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. In *IEEE/CVF International Conference on Computer Vision, ICCV*, pages 15613–15623. IEEE, 2023. 1