LANGUAGE-AWARE SOFT PROMPTING FOR VISION & LANGUAGE FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper is on soft prompt learning for Vision & Language (V&L) models. Similarly to their NLP counterparts, V&L models can be adapted to a downstream task by learning *soft* continuous prompts using a few training examples. Current methods learn the soft prompts by minimizing a cross-entropy loss using as class weights the features obtained by passing the prompts plus the class names through the text encoder. Such methods, however, significantly overfit the training data suffering from large accuracy degradation when tested on unseen classes from the same domain. Our main contribution, in this paper, is a surprisingly simple approach to alleviate this problem: we use a second cross entropy loss to minimize the distance between the learned soft prompts and a set of hand-engineered manual prompts (obtained by prompt engineering). The proposed loss can be interpreted in multiple ways including as a regularizer, as a means for language-based augmentation, and as a way of learning more discriminative class centroids. Importantly, our formulation is inherently amenable to including, during training, virtual classes, i.e. class names for which no visual samples are available, further increasing the robustness of the learned prompts. Through extensive evaluations on 11 datasets, we show that our approach (a) significantly outperforms all prior works on soft prompting, and (b) matches and surpasses, for the first time, the accuracy on novel classes obtained by hand-crafted prompts and CLIP for the majority of the test datasets. Code will be made available.

1 INTRODUCTION

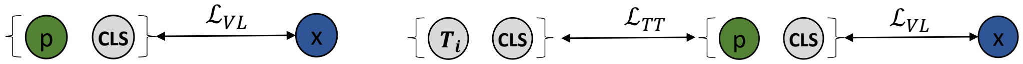
Large-scale pre-training of neural networks has recently resulted in the construction of a multitude of foundation models for Language (Devlin et al., 2018; Radford et al., 2019) and Vision & Language (V&L) understanding (Radford et al., 2021; Jia et al., 2021; Yu et al., 2022; Alayrac et al., 2022). Unlike the previous generation of neural networks, such models can better capture the distribution of the world from which new favorable properties and characteristics emerge. Of particular interest to this work are V&L models trained with contrastive learning (i.e. CLIP-like models (Radford et al., 2021; Jia et al., 2021; Li et al., 2021; Yao et al., 2021; Yu et al., 2022)), which have enabled seamless few-shot and even zero-shot adaptation to new downstream tasks and datasets. Specifically, this paper proposes a simple yet highly effective way to drastically improve soft prompt learning for the few-shot adaptation of the V&L model to a given downstream task.

Similarly to their NLP counterparts (Radford et al., 2021; Lester et al., 2021; Li & Liang, 2021), prompt engineering and learning has emerged as one of the most powerful techniques for adapting a V&L to new tasks. Initially, in (Radford et al., 2021), a set of manually-defined hand-engineered templates (or prompts) like `a photo of a {cls_name}`, or `a black and white photo of a {cls_name}` were passed through the text encoder of the V&L model to create class-specific weights for category `cls_name` that can be used for zero-shot recognition. Following research in NLP (Lester et al., 2021; Li & Liang, 2021), subsequent work (Zhou et al., 2022b;a) has proposed replacing the manually picked templates with a sequence of learnable vectors, also coined *soft prompts*, which are fed as input to the text encoder along with the class name `cls_name`. The soft prompts are learned from a few training examples with the parameters of the entire V&L model kept frozen. The whole process can be seen as parameter efficient fine-tuning of the V&L model on a small training dataset.

However, a clearly identifiable problem with prompt learning is overfitting: while the accuracy on the classes used for training (base classes) significantly increases, the accuracy on unseen, during training, (novel) classes significantly drops. This is to some extent expected as soft prompts are learned from few examples belonging to the base classes. Notably, on novel classes, direct, zero-shot recognition using hand-engineered prompts outperforms all existing soft prompt learning methods.

To alleviate this problem, in this work, we propose a surprisingly simple solution: since prompt learning improves the accuracy on base classes but prompt engineering is better on novel classes, we propose to learn the soft prompts by adding a cross entropy text-to-text loss that enforces the learned prompts to be close, in embedding space, to the hand-engineered ones, thus exploiting the intrinsic information captured by the text encoder. This is in contrast with prior soft-prompt learning methods that only capture vision-language interactions. Moreover, besides acting as a regularizer, the proposed loss can be interpreted as a means for language-based augmentation, and as a way of learning more discriminative class centroids. Our **main contributions** are:

- We propose a novel Language-Aware Soft Prompting (LASP) learning method by means of a cross-entropy regularization loss that enforces the learned prompts to be correctly classified with respect to the hand-engineered ones.
- We propose LASP+: training LASP with virtual classes by including, during training, class names for which no visual samples are available. Importantly, we show that LASP+ significantly increases the robustness of the learned prompts.
- Our methods set a new state-of-the-art for few-shot and zero-shot image classification on 11 datasets, significantly outperforming all soft prompting prior works. Importantly, we present, for the first time, a prompt learning method that outperforms, for the majority of the test datasets, the very strong baseline based on hand-crafted prompts and CLIP for the recognition of novel classes (i.e. zero-shot setting).



(a) CoOp (Zhou et al., 2022b)

(b) LASP (Ours)

Figure 1: **Paper's main idea:** In addition to minimizing the distance between the vision and language embeddings (left), our approach, called LASP (right), fully utilizes the intrinsic model of the world learned by the text encoder, by minimizing a cross entropy loss between the soft learnable prompts ($\mathbf{p}$(●) + CLS(●)) and hand-engineered templates (T_i (●) + CLS). Importantly, our formulation can incorporate virtual classes, i.e. novel class names for which no visual data is available.

2 Related work

Contrastive Vision-Language Models: Recently, large scale V&L pre-training with contrastive learning has been used to train foundation models resulting in robust representations, transferable to new tasks both under few-shot and zero-shot settings (Radford et al., 2021; Jia et al., 2021; Li et al., 2021; Yao et al., 2021; Yu et al., 2022). Such networks consist of a vision encoder (typically a ViT (Dosovitskiy et al., 2020)) and a Transformer-based text encoder (Vaswani et al., 2017). Highly parameterized instantiations of such architectures are trained on large corpora of image-caption pairs (e.g. Radford et al. (2021) uses 400M and Jia et al. (2021) 1B pairs) using contrastive learning. We used CLIP (Radford et al., 2021) as the foundation model for our method.

Prompt Learning is about adapting pre-trained foundational models on (downstream) tasks, typically in a zero-shot or few-shot setting. Firstly proposed in the context of Language Models (LM), prompting was initially about prepending hand-crafted instructions/examples to the task input so that the LM generates the appropriate output conditioned to the input (Radford et al., 2019; Brown et al., 2020). In Schick & Schütze (2020a;b), the main idea is to reformulate the downstream task as a *cloze* task using hand-crafted patterns (or templates), thus avoiding the need to train a task-specific classifier. As finding the optimal patterns is laborious, recent works have attempted to address this by learning a set of soft (continuous) prompts (Lester et al., 2021; Li & Liang, 2021).

In V&L foundation models, like CLIP, the class names are used to create hand-crafted prompts (Radford et al., 2021) that are fed as input to the text encoder, enabling zero-shot visual recognition. CoOp (Zhou et al., 2022b) extends work on soft prompt optimization to the V&L domain by learning a set of M prompts which are used as input to the text encoder alongside the class name. The prompts are learned by minimizing the classification error on a training set consisted of the given base classes. One major limitation of CoOp is weak generalization: the learned prompts overfit the base classes and do not work well when tested on novel classes. To alleviate this, CoCoOp (Zhou et al., 2022a) proposes a dynamic version of (Zhou et al., 2022b) where a small network is trained to produce a visual feature from the input image that is added to the learned prompts, hence making them input specific (i.e. dynamic). Other directions for adapting V&L models via prompting were also recently proposed in ProGrad (Zhu et al., 2022) via gradient matching, TTTuning (Shu et al., 2022) via test time tuning, (Huang et al., 2022) by learning prompts in an unsupervised manner and Lu et al. (2022) by taking a probabilistic view, learning a distribution over the output space.

The proposed LASP is a direct extension to standard continuous soft optimization for V&L models (e.g. CoOp) that alleviates overfitting and significantly improves upon the previously reported best results without resorting to a dynamic approach as in CoCoOp (Zhou et al., 2022a). We show that this can be simply achieved by additionally using a text-to-text loss that enforces the learned prompts to be "close" to a set of hand-engineered ones in the text encoder space. Interestingly, LASP allows the incorporation of novel class name information for which no (visual) training data is available. Notably, our approach is very efficient (as efficient as (Zhou et al., 2022b)) as opposed to (Zhou et al., 2022a) which requires recomputing all the class-related text embeddings every time a new image is to be classified.

3 Method

3.1 Background

Prompt engineering enables zero-shot visual recognition using V&L models trained with contrastive learning (CLIP in this work) as follows: Given a set $\mathcal{V}$ of C class names, $\texttt{class\_name}_c$, $c \in \{1, \ldots, C\}$, a prompt, i.e. a manually designed template concatenated with the class name like $h_c = \texttt{a photo of a } \{\texttt{class\_name}_c\}$, is passed through the V&L's text encoder $g_T(.)$ to compute the class specific text feature (weight) $\mathbf{t}_c^h = g_T(h_c)$. Moreover, an image $\mathbf{x}$ to be classified is passed through the V&L's image encoder $g_I(.)$ to compute image specific feature $\mathbf{f} = g_I(\mathbf{x})$. A probability distribution over the class labels is given by:

$$P_h(c|\mathbf{x}) = \frac{\exp\Big(\texttt{cos}(\mathbf{t}_c^h, \mathbf{f})/\tau\Big)}{\sum_{c=1}^{C} \exp\Big(\texttt{cos}(\mathbf{t}_c^h, \mathbf{f})/\tau\Big)}, \tag{1}$$

where τ is a temperature factor and $\texttt{cos}$ the cosine similarity. Finally, the class for $\mathbf{x}$ is given by $\tilde{c} = \arg_{max} P_h(c|\mathbf{x})$. Note that, to compute $\mathbf{t}_c^h$, no training with class specific image data is required, thus enabling zero-shot recognition for any given class name.

Soft prompt learning (Lester et al., 2021; Li & Liang, 2021; Zhou et al., 2022b) is concerned with parameter efficient fine-tuning of a pre-trained V&L model by learning a sequence of M learnable vectors $\mathbf{p}_m \in \mathbb{R}^d, m = \{1, \ldots, M\}$ using a few labeled samples. Specifically, the manually picked prompt h_c is replaced by a new learnable one $\mathbf{r}_c$ formed by concatenating the sequence of $\mathbf{p_m}$ with the word embedding $\mathbf{w}_c$ of $\texttt{class\_name}_c$, that is: $\mathbf{r}_c = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_M, \mathbf{w}_c\}$, and, finally, a class specific text feature $\mathbf{t}_c^r = g_T(\mathbf{r}_c)$ is obtained. A probability distribution over the class labels is:

$$P_r(c|\mathbf{x}) = \frac{\exp\Big(\texttt{cos}(\mathbf{t}_c^r, \mathbf{f})/\tau\Big)}{\sum_{c=1}^{C} \exp\Big(\texttt{cos}(\mathbf{t}_c^r, \mathbf{f})/\tau\Big)}. \tag{2}$$

Given a training dataset consisting of N image-class pairs $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, the prompts can be learned by minimizing the cross-entropy loss:

$$\mathcal{L}_{VL} = -\sum_{i=1}^{N}\sum_{c=1}^{C} \log P_r(c|\mathbf{x}_i) y_c. \tag{3}$$

Note that the V&L model remains entirely frozen during training. Moreover, as the soft prompts are typically shared across all classes, they can be directly used for zero-shot evaluation on additional novel classes.

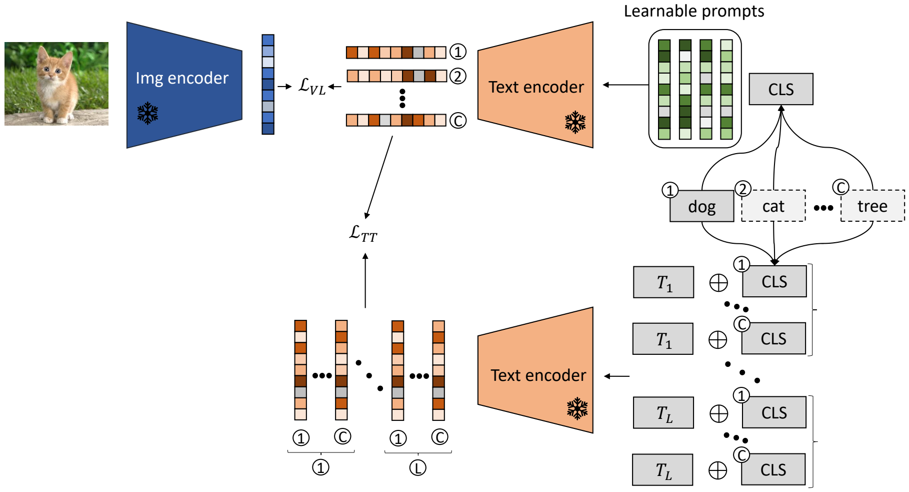


Figure 2: **Overall idea.** Our approach jointly models image-text and text-text interactions by applying the standard cross entropy loss L_{VL} for the former (Eq. 3) and the proposed regularization loss L_{TT} (Eq. 5) for the latter. $T_1, \dots, T_L$ are hand-engineered prompts like "a photo of the $\{\}$." or "a cropped photo of the $\{\}$.".

3.2 Language-Aware Soft Prompting (LASP)

Despite its strong performance on base classes, vanilla soft prompt learning (see Sec. 3.1) underperforms on novel classes (i.e. zero-shot setting). While CoCoOp (Zhou et al., 2022b) partially alleviates this by conditioning on the image feature, its accuracy for the zero-shot setting is still trailing that of CLIP with hand-crafted prompts. Moreover, it requires passing the prompts for all classes through the text encoder every time a new image is to be classified. To alleviate this, and, departing from existing approaches considering only vision-language interactions (between the learnable prompts and the image), we propose to exploit the intrinsic model of the world and the semantic relations learned by the text encoder.

The main idea of our paper is simple, yet powerful: since the hand-engineered prompts outperform the learnable soft prompts for the zero-shot setting, then, in order to avoid overfitting to the base classes and strengthen generalizability, the learnable ones should be trained so that they can be correctly classified in language space where the class weights are given by the hand-engineered prompts. In other words, the model is forced to correctly classify the learnable prompts into the corresponding hand-crafted ones.

To this end, a second cross entropy loss is used to minimize the distance between the encoded learned soft prompts and the encoded hand-engineered ones. Specifically, recall that $\mathbf{t}_c^h = g_T(h_c)$ is the class weight for class c obtained by encoding the corresponding hand-engineered prompt. Assuming that L hand-engineered prompts are available, we have $\mathbf{t}_c^{h,l}$, $l = 1, \dots, L$. Moreover, $\mathbf{t}_c^r$ is the encoded learnable prompt. Finally, the probability of prompt $\mathbf{t}_c^r$ being classified as class c is:

$$P_{rh}(c|\mathbf{t_c^r}) = \frac{1}{L}\sum_{l=1}^{L} \frac{\exp\Big(\cos(\mathbf{t}_c^{h,l},\mathbf{t}_c^r)/\tau\Big)}{\sum_{c=1}^{C}\exp\Big(\cos(\mathbf{t}_c^{h,l},\mathbf{t}_c^r)/\tau\Big)}. \tag{4}$$

The language-aware training loss is computed similarly to the vision-language loss:

$$\mathcal{L}_{TT} = -\sum_{i=1}^{N}\sum_{c=1}^{C} \log P_{rh}(c|\mathbf{x}_i)y_c, \tag{5}$$

with the overall training objective defined as:

$$\mathcal{L} = \alpha_{VL}\mathcal{L}_{VL} + \alpha_{TT}\mathcal{L}_{TT}, \tag{6}$$

where α_{VL} and α_{TT} are user-defined scaling coefficients controlling the magnitude of the $\mathcal{L}_{VL}$ and $\mathcal{L}_{TT}$ losses, respectively. Overall, we call the proposed learning formulation Language-Aware Soft Prompting (LASP). See also Fig 2.

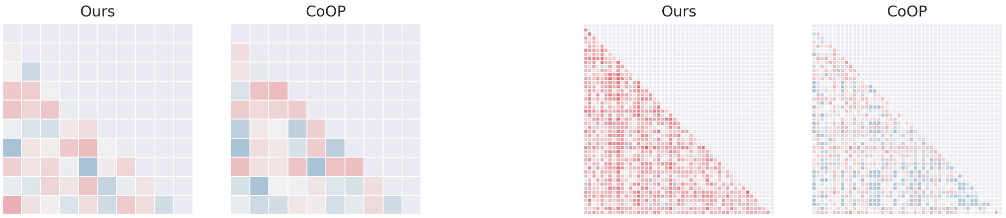
LASP can be interpreted in a number of ways:

LASP as a regularizer: Although the learned prompts constitute a small number of parameters, especially in the few-shot setting, the resulting models (prompts) are prone to overfitting to base classes (Zhou et al., 2022b). As the proposed language-aware loss encourages the learned prompts to be close in embedding space to the hand-engineered ones, LASP can be naturally viewed as a regularizer that prevents the learned prompt-conditioned features from diverging too much from the hand-crafted ones.

LASP as language-based augmentation: Current soft prompt learning methods restrict augmentation to the vision domain, where random transformations, such as rotation, colour jittering or scaling, increase the robustness of the system, especially for cases with limited number of training samples. However, no augmentations are performed in the language domain. Ideally, we want the prompt-conditioned text embedding to be robust too, capturing the full space of each class. In practice, we can achieve this by targeted prompting, where we can specify certain characteristics and/or apply text-based transformations to the class name, e.g.: “A sketch of *dog*” or “A rotated photo of a *dog*”.

At train time, as reflected by Eq. 4, we compute the class label distribution per $l-$th template and then average over all templates. Hence, we opt not to mix across templates during training as we want the model to focus on class information solely. For example, the model could distinguish easier between a “a sketch of a *dog*” and “a photo of a wolf” compared to “a sketch of a *dog*” and “a sketch of a wolf”, as in the former case, the style could be used as an additional queue. We validated this in preliminary experiments (intermixing the templates was found to hurt performance).

LASP for discriminative class centroids: By optimizing w.r.t both image and text, our method produces class centroids that are more discriminative and have a higher separation margin. This can be visualised in Fig. 3 where we plot the cosine distance between the embeddings of each class. Our approach learns class centroids that have a higher cosine distance than those of our baseline, CoOp.



(a) Eurosat; Ours (0.516) vs CoOp (0.491)

(b) DTD; Ours (0.644) vs CoOp (0.488)

Figure 3: **Cosine distance between the class embeddings** produced by the CLIP text encoder on (a) Eurosat, and (b) DTD. LASP is compared with CoOp. Class centroids situated further apart are more separable as the underlying image features are identical across both methods. Brighter colours indicate bigger cosine distances. The numbers shown for each method and dataset indicate the average cosine distance between the classes. Best viewed zoomed-in, in colour.

3.3 LASP WITH VIRTUAL CLASSES (LASP+)

A direct observation that can be drawn from Eq. 4 is that, in practice, we do not have to use only the class names for which we have labelled image data, as the value of L_{TT} is independent of the input image. To this end, we propose to learn the prompts using both annotated image-text pairs and *class names* outside of the base set (for which we have no images available). We call this setting as training *LASP with virtual classes*. Our setting combines the best of both words: the guidance from the few annotated image samples and the zero-shot generalizability of language-based training. As our results show, this small addition can significantly improve the robustness of the prompts learned. We refer to this variant of our method as **LASP+**.

4 Experiments

Following (Radford et al., 2019; Zhou et al., 2022a), we mainly evaluated the accuracy of our approach on generalization to novel classes (i.e. zero-shot recognition) for 11 datasets in total. Each dataset is split into two equal partitions with disjoint classes, named *base* and *new*. We trained our model using text-image pairs from the base classes and test on both base and new classes. For other types of experiments reported in (Zhou et al., 2022a), including cross-dataset transfer and domain generalization, see our Supp. Mat.

Datasets: We performed experiments on 11 datasets, namely: ImageNet (Deng et al., 2009), Caltech101 (Fei-Fei et al., 2004), Oxford-Pets (Parkhi et al., 2012), Stanford Cars (Krause et al., 2013), Flowers102 (Nilsback & Zisserman, 2008), Food101 (Bossard et al., 2014), FGVC Aircraft (Maji et al., 2013), SUN397 (Xiao et al., 2010), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019) and UCF-101 (Soomro et al., 2012).

Models: For all experiments, unless otherwise specified, we used a pretrained CLIP model with a ViT-B/16 image encoder, $M = 4$ learnable prompts and 16 samples per class. In all experiments, we report the average across 3 runs.

Training: Largely, we followed the training procedure described in CoOp (Zhou et al., 2022b) and CoCoOp (Zhou et al., 2022a) (i.e. same image augmentation, SGD with initial learning rate of 0.002 and a cosine annealing scheduler with 1 epoch of warm-up). In Eq. 6, α_{VL} was set to 1 and α_T to 20. The number of hand-engineered templates L was set to 34. The templates were taken from CoOp and CLIP (see Supp. Mat. for a full list). All training and testing was done on a single NVIDIA V100 GPU (except for Imagenet where 4 GPUs were used). The code was implemented using PyTorch (Paszke et al., 2017).

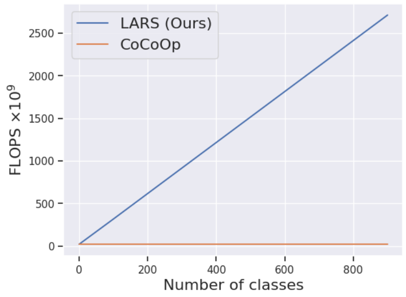


Figure 4: **Comparison between LASP and CoCoOp in terms of number of FLOPs**. While the inference cost of LASP remains largely constant with respect to the number of classes, CoCoOp's cost increases linearly (from around ≈20 GFLops for 1 class to over 2,500 GFLOPs for 1,000).

Methods compared: We report the performance of both LASP and LASP+. For the latter case, the *class names only* of the novel classes are used during training as virtual classes. We also study the impact of adding other types of virtual classes. The direct baseline that our method is compared with is CoOp (Zhou et al., 2022b). Note that both methods have *exactly* the same inference (as our method adds in addition a text-to-text loss during training). We also compare with CoCoOp (Zhou et al., 2022a) which is a dynamic approach that conditions the prompts on image features, and hence induces significant additional computation during inference. See also Fig. 4 for a comparison.

Standard setting of (Zhou et al., 2022a): As the results from Table 1 show, LASP significantly outperforms its direct baseline, CoOp on the new classes, by 4.5% on average, across all 11 datasets, while largely matching the accuracy on the base classes. Moreover, LASP+, surpasses, on average, both CoOp, by 6%, and the stronger CoCoOp, by 1.5%. Notice that significantly larger gains are observed on datasets with informative class names, such as EuroSAT or UCF101, and lower ones on datasets containing less informative or more challenging class names, such as FGVCAircraft (where the model number in isolation is less informative). Finally, LASP+ consistently outperforms the strong CLIP baseline with hand-crafted prompts on the novel classes (in 8 out of 11 datasets).

Generalized zero-shot setting: The current evaluation protocol used in (Zhou et al., 2022a) computes the accuracy considering the base and new classes in isolation. A more realistic evaluation protocol should consider the classes across both subsets (i.e. base and novel) jointly. Table 2 shows the results of the generalized zero-shot setting. As expected, there is an accuracy degradation across

Table 1: **Comparison between LASP/LASP+ with the state-of-the-art on 11 datasets**.

(a) **Average over 11 datasets**.

	Base	New	H
CLIP	69.34	**74.22**	71.70
CoOp	**82.69**	63.22	71.66
CoCoOp	80.47	71.69	75.83
LASP	81.26	71.54	76.09
LASP+	81.42	74.17	**77.62**

(b) ImageNet.

	Base	New	H
CLIP	72.43	68.14	70.22
CoOp	**76.47**	67.88	71.92
CoCoOp	75.98	**70.43**	73.10
LASP	75.97	70.31	73.03
LASP+	76.23	70.40	**73.20**

(c) Caltech101.

	Base	New	H
CLIP	96.84	94.00	95.40
CoOp	**98.00**	89.81	93.73
CoCoOp	97.96	93.81	95.84
LASP	97.70	94.08	95.85
LASP+	97.80	**94.25**	**96.00**

(d) OxfordPets.

	Base	New	H
CLIP	91.17	97.26	94.12
CoOp	93.67	95.29	94.47
CoCoOp	95.20	97.69	96.43
LASP	95.13	96.23	95.68
LASP+	**95.43**	**97.70**	**96.55**

(e) StanfordCars.

	Base	New	H
CLIP	63.37	**74.89**	68.65
CoOp	**78.12**	60.40	68.13
CoCoOp	70.49	73.59	72.01
LASP	72.46	71.80	72.19
LASP+	72.73	71.74	**72.23**

(f) Flowers102.

	Base	New	H
CLIP	72.08	**77.80**	74.83
CoOp	**97.60**	59.67	74.06
CoCoOp	94.87	71.75	81.71
LASP	96.47	70.7	81.59
LASP+	96.20	73.93	**83.61**

(g) Food101.

	Base	New	H
CLIP	90.10	91.22	90.66
CoOp	88.33	82.26	85.19
CoCoOp	**90.70**	91.29	90.99
LASP	90.30	90.73	90.51
LASP+	**90.70**	**91.36**	**91.02**

(h) FGVCAircraft.

	Base	New	H
CLIP	27.19	**36.29**	31.09
CoOp	**40.44**	22.30	28.75
CoCoOp	33.41	23.71	27.74
LASP	32.63	30.46	31.57
LASP+	33.03	32.30	**32.66**

(i) SUN397.

	Base	New	H
CLIP	69.36	75.35	72.23
CoOp	**80.60**	65.89	72.51
CoCoOp	79.74	76.86	78.27
LASP	80.20	75.56	77.81
LASP+	80.33	**77.93**	**79.12**

(j) DTD.

	Base	New	H
CLIP	53.24	59.90	56.37
CoOp	79.44	41.18	54.24
CoCoOp	77.01	56.00	64.85
LASP	79.13	52.10	62.82
LASP+	**79.57**	59.47	**68.06**

(k) EuroSAT.

	Base	New	H
CLIP	56.48	64.05	60.03
CoOp	**92.19**	54.74	68.69
CoCoOp	87.49	60.04	71.21
LASP	91.23	63.16	74.64
LASP+	90.26	**69.23**	**78.46**

(l) UCF101.

	Base	New	H
CLIP	70.53	77.50	73.85
CoOp	**84.69**	56.05	67.46
CoCoOp	82.33	73.45	77.64
LASP	82.70	71.80	76.86
LASP+	83.43	**77.60**	**80.4**

all datasets for all methods. The largest drops are on the new classes, which suggests that current methods for prompt learning overfit to some extent to the base classes. We note that, despite this drop, LASP+ is more robust than CoCoOp, overfitting less to the base classes.

Effect of out-domain distractors: Motivated by the very recent work of Ren et al. (2022) suggesting that CLIP's performance drops as the number of classes used for testing increases, we introduce a new evaluation setting: Firstly, we select 4 test datasets with clear disjoint domains: EuroSAT (10 satellite terrain types), Food101 (101 food names), Flowers102 (102 flower names) and OxfordPets (37 dog and cat breed names). At test time, we define the classifier across the union of classes across all 4 datasets (250 classes in total). Note that LASP+ is the only method that can benefit from knowledge of this expanded vocabulary during training.

As the results from Table 3 show, generally, the drop in accuracy is moderate (typically 2-3%) showing that the models are somewhat robust to out-of-domain distractors. Exception is the EuroSAT dataset where the number of classes considered increases 25× (from 10 to 250). On EuroSAT, we observe large accuracy drop for both LASP and CoCoOp on both base and novel classes. Note that, for these experiments, both LASP and LASP+ outperform CoCoOp. Importantly, LASP+, trained on virtual classes, manages to fully recover the lost accuracy, even for the case of EuroSAT, matching its accuracy when testing without considering the distractors.

Effect of in-domain distractors: Expanding on the idea from the previous section, herein, we propose to test the performance of the current soft prompting methods with in-domain distractors. Unlike the case of out-of-domain distractors, the in-domain distractors are selected such that they are closely related to the current dataset/classes being part of the same super-category. We performed

Table 2: **Comparison with the state-of-the-art for the generalized zero-shot setting**.

(a) **Average over 11 datasets**.

	Base	New	H
CoCoOp	72.46	64.77	68.39
LASP	73.92	64.86	69.09
LASP+	**74.61**	**66.58**	**70.39**

(b) ImageNet.

	Base	New	H
CoCoOp	71.9	67.5	69.63
LASP	**72.0**	67.21	69.51
LASP+	71.9	**67.8**	**69.78**

(c) Caltech101.

	Base	New	H
CoCoOp	95.20	90.67	92.87
LASP	94.76	92.16	93.44
LASP+	95.46	**92.76**	**94.24**

(d) OxfordPets.

	Base	New	H
CoCoOp	91.01	93.10	92.04
LASP	90.60	92.67	91.62
LASP+	**91.86**	**93.13**	**92.49**

(e) StanfordCars.

	Base	New	H
CoCoOp	67.26	**69.43**	68.33
LASP	**69.16**	67.65	**68.39**
LASP+	68.93	67.8	68.36

(f) Flowers102.

	Base	New	H
CoCoOp	86.73	64.63	74.06
LASP	**89.50**	65.40	75.57
LASP+	89.33	**67.90**	**77.15**

(g) Food101.

	Base	New	H
CoCoOp	85.73	85.50	85.61
LASP	85.84	86.11	85.98
LASP+	**86.23**	**86.44**	**86.32**

(h) FGVCAircraft.

	Base	New	H
CoCoOp	**24.50**	25.93	**25.19**
LASP	22.90	26.06	24.37
LASP+	22.10	**26.43**	24.07

(i) SUN397.

	Base	New	H
CoCoOp	71.13	67.76	69.40
LASP	**72.53**	65.86	69.03
LASP+	72.10	**68.51**	**70.25**

(j) DTD.

	Base	New	H
CoCoOp	59.33	42.70	49.65
LASP	**63.46**	41.67	51.02
LASP+	62.73	**46.60**	**53.47**

(k) EuroSAT.

	Base	New	H
CoCoOp	69.20	39.23	50.14
LASP	76.53	42.80	54.89
LASP+	**82.16**	**44.13**	**57.41**

(l) UCF101.

	Base	New	H
CoCoOp	75.16	66.10	70.34
LASP	75.87	65.97	70.57
LASP+	**78.0**	**70.91**	**74.28**

Table 3: **Effect of out-domain distractors.** w/o distractors are the results of the generalized zero-shot setting of Table 2, shown here for convenience.

(a) EuroSAT.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
CoCoOp	69.20	39.23	50.14	62.76	33.90	44.02
LASP	82.16	44.13	57.14	67.1	35.43	46.37
LASP+	**82.80**	**44.56**	**57.93**	**81.66**	**40.93**	**54.52**

(b) Food101.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
CoCoOp	85.73	85.50	85.61	85.10	85.20	85.14
LASP	**86.23**	86.44	**86.32**	85.10	85.71	85.40
LASP+	85.93	**86.60**	86.26	85.15	86.01	85.58

(c) Flowers102.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
CoCoOp	86.73	64.43	74.06	85.16	62.40	72.01
LASP	**89.33**	67.9	77.15	**88.90**	67.43	76.69
LASP+	88.86	**69.13**	**77.37**	88.53	**68.71**	**77.36**

(d) OxfordPets.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
CoCoOp	91.01	93.10	92.04	88.83	91.06	89.93
LASP	91.86	**93.13**	**92.49**	90.60	91.10	90.84
LASP+	**92.0**	91.74	91.86	**91.67**	**91.57**	**91.62**

experiments on two datasets: Food101 and Flowers102. For Flowers102, we added 65 new class names while, for Food101, 53 new classes. Note again that, with the exception of LASP+, the classes are only used at test time as distractors expanding the C-way classifier by 65 and 53, respectively. The list of added classes can be found in the appendix.

As the results from Table 4 show, the performance drops by 5-7% indicating that in-domain distractors significantly increase the problem difficulty compared to the out-of-domain ones. Again, both LASP and LASP+ outperform the more computationally demanding, image-conditioned CoCoOp. Moreover, LASP+ trained with virtual classes recovers part of the lost accuracy without affecting the performance of the model when evaluated without distractors.

Table 4: **Effect of in-domain distractors.** w/o distractors are the results of the generalized zero-shot setting of Table 2, shown here for convenience.

(a) Food101.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
CoCoOp	85.73	85.50	85.61	80.80	81.77	81.28
LASP	**86.23**	**86.44**	**86.32**	81.80	82.36	82.08
LASP+	85.97	86.02	85.99	**82.27**	**82.93**	**82.60**

(b) Flowers102.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
CoCoOp	86.73	64.43	74.06	75.83	60.96	67.58
LASP	89.33	**67.9**	77.15	78.83	62.50	69.72
LASP+	**91.30**	67.04	**77.31**	**80.90**	**63.71**	**71.28**

5 ABLATION STUDY

Effect of size and content of the hand-engineered prompts: Herein, we study the effect of the size L and the content of the set of the hand-engineered prompts used by our method in Eq. 4. The hand-crafted templates are increased to 100 by including the rest of the prompts defined in CLIP (Radford et al., 2021), while their number is reduced to 1 by using the following template only: `a photo of {}`. Random templates are produced by sampling grammatically plausible random sentences that contain incoherent words, with length between 5 and 20 words. The class names are inserted at the end of these random templates (for examples see Supp. Mat.). All variations use the same training scheduler and hyper-parameters, except for the case of random templates, where $\alpha_{TT} = 5$.

Table 5 shows our results. Firstly, we importantly note that the accuracy on the base classes remains similar across all settings (not shown in the table). This shows that the exact choice of the templates might not be so significant for the few-shot setting. However, as Table 5 shows, for the case of novel classes, both the number and the content of the templates are important to obtain high accuracy.

Table 5: **Effect of dictionary size and content on new classes.** Accuracy on the base classes remains similar across all settings, hence it is omitted. 34 templates were used for the paper's main results. LASP (R) denotes models trained using randomly constructed templates.

(a) DTD.

#Templates	1	34	100
LASP (R)	49.02	51.63	52.64
LASP	**50.73**	**52.10**	**56.53**

(b) EuroSAT.

#Templates	1	34	100
LASP (R)	55.01	59.9	62.1
LASP	**56.97**	**63.16**	**65.13**

(c) UCF101.

#Templates	1	34	100
LASP (R)	67.5	68.6	70.03
LASP	**71.36**	**71.80**	**72.77**

6 CONCLUSIONS

We introduced LASP, a simple, straightforward approach to alleviate overfitting in soft prompt learning for V&L models: a cross entropy loss that minimizes the distance between the learned soft prompts and a set of hand-engineered ones. The proposed loss can be interpreted as a regularizer, as a means for language-based augmentation, and as a way of learning more discriminative class centroids. Moreover, to increase the robustness of the learned prompts, we also proposed LASP+ which uses, during training, virtual classes, i.e. class names for which no visual samples are available.

Some of the main findings of our experiments include: **1.** LASP significantly improves upon our direct baseline, CoOp, by +4.5% on average, demonstrating the effectiveness of the proposed text-to-text loss. **2.** LASP+, trained with virtual classes, outperforms CoOp by 6.5%, and the more computationally demanding CoCoOp by 1.5%. It also outperforms CLIP for the case of novel classes in 8 out of 11 datasets. **3.** Larger gains are observed on datasets with semantically distinctive class names (i.e. UCF-101, EuroSAT). **4.** Both LASP and LASP+ are more robust than CoCoOp for the case of both in-domain and out-of-domain distractors. We hope that LASP/LASP+ will serve as a strong baseline for future works in the area of few-shot adaptation for V&L models, attracting the community attention to the largely neglected language side of the problem.

ETHICS STATEMENT

Our approach builds upon a pre-trained CLIP model. As CLIP models were shown to exhibit various form of bias (Agarwal et al., 2021), especially when combined with improper class names, these biases may be transferred to the prompt-adapted models, too. Therefore, any deployment of such systems and of their derivatives should undergo careful checks and considerations.

REPRODUCIBILITY STATEMENT

We detail the training procedure and evaluation settings in Sec. 4, noting again that all the augmentations follow (Zhou et al., 2022a;b). We also list in the appendix details regarding the prompts used. We will release the code alongside the pretrained models to ensure the reproducibility of our method.

References

Sandhini Agarwal, Gretchen Krueger, Jack Clark, Alec Radford, Jong Wook Kim, and Miles Brundage. Evaluating clip: towards characterization of broader capabilities and downstream implications. *arXiv preprint arXiv:2108.02818*, 2021.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461. Springer, 2014.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3606–3613, 2014.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pp. 178–178. IEEE, 2004.

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*, 12(7):2217–2226, 2019.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8340–8349, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15262–15271, 2021b.

Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pp. 4904–4916. PMLR, 2021.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 554–561, 2013.

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

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

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. *arXiv preprint arXiv:2110.05208*, 2021.

Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022.

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

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

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 3498–3505. IEEE, 2012.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pp. 5389–5400. PMLR, 2019.

Shuhuai Ren, Lei Li, Xuancheng Ren, Guangxiang Zhao, and Xu Sun. Rethinking the openness of clip. *arXiv preprint arXiv:2206.01986*, 2022.

Timo Schick and Hinrich Schütze. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*, 2020a.

Timo Schick and Hinrich Schütze. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*, 2020b.

Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances on Neural Information Processing Systems*, 2022.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32, 2019.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 3485–3492. IEEE, 2010.

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. *arXiv preprint arXiv:2111.07783*, 2021.

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

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16816–16825, 2022a.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022b.

Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. *arXiv preprint arXiv:2205.14865*, 2022.

A Additional results

A.1 Cross-Dataset Transfer

Following (Zhou et al., 2022a), we measure how well the soft prompts learned on ImageNet perform when evaluated on different datasets. In this setting, the training is performed on images from all 1,000 classes, using 16 images for each class. As the results from Table 6 show, our approach surpasses CoOp by 2% while matching and marginally outperforming the more computationally demanding CoCoOp (0.5% better on average).

A.2 Domain generalization

A.2.1 Experimental setup

Following the encouraging results reported in (Zhou et al., 2022b;a) on domain generalization that show the learned prompts to be robust to domain shifts, herein we attempt to evaluate whether our approach can improve the quality of the leaned prompts under domain shift too. To this end, we trained LASP on all classes from ImageNet (16-shot setting) and evaluate the learned prompts on 5 datasets with class names compatibles with those of ImageNet, but different data distribution.

Datasets: Following (Zhou et al., 2022b) we used ImageNet (Deng et al., 2009) as the source dataset, and ImageNetV2 (Recht et al., 2019), ImageNet-Sketech (Wang et al., 2019), ImageNet-A (Hendrycks et al., 2021a) and ImageNet-R (Hendrycks et al., 2021b) as the test datasets.

A.2.2 Results

As the results from Table 7 show, with the exception of ImageNet-V2, our approach outperforms all prior work, showing strong domain generalizability performance. Moreover, even when training on base classes alone (i.e. on images from 500 classes instead of 1,000), our approach (denoted as LASP* in the table) outperforms CoOp while matching CoCoOp - both trained on all 1,000 classes.

Table 6: **Comparison with state-of-the-art for the cross-dataset transfer setting.**

	Source	Target										
	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UCF101	*Average*
CoOp	**71.51**	93.70	89.14	64.51	68.71	85.30	18.47	64.15	41.92	46.39	66.55	63.88
CoCoOp	71.02	94.43	**90.14**	**65.32**	**71.88**	86.06	22.94	**67.36**	**45.73**	45.37	68.21	65.74
LASP	71.10	**94.50**	89.36	64.83	70.53	**86.30**	**23.03**	67.0	45.54	**48.33**	**68.24**	**66.25**

B Implementation details

Hand-engineered prompts set ζ: Unless otherwise specified, we used the following set of hand-engineered templates (borrowed from CLIP and CoOp):

```
"a photo of a {}, a type of flower.",
"a photo of a person doing {}.",
"a centered satellite photo of {}.",
"a photo of a {}, a type of aircraft.",
"{} texture.",
"itap of a {}.",
"a bad photo of the {}.",
"a origami {}.",
"a photo of the large {}.",
```

Table 7: **Comparison with state-of-the-art for the domain generalization setting.** LASP* is trained on 500 classes only instead on all 1,000.

		Source	Target			
	Learnable?	ImageNet	ImageNetV2	ImageNet-Sketch	ImageNet-A	ImageNet-R
CLIP		66.73	60.83	46.15	47.77	73.96
CoOp	✓	**71.51**	**64.20**	47.99	49.71	75.21
CoCoOp	✓	71.02	64.07	48.75	50.63	76.18
LASP*	✓	71.05	63.03	48.06	50.67	76.60
LASP	✓	71.10	63.77	**49.01**	**50.70**	**77.07**

```
"a {} in a video game.",
"art of the {}.",
"a photo of the small {}.",
"a photo of a {}.",
"a photo of many {}.",
"a photo of the hard to see {}.",
"a low resolution photo of the {}.",
"a rendering of a {}.",
"a bad photo of the {}.",
"a cropped photo of the {}.",
"a pixelated photo of the {}.",
"a bright photo of the {}.",
"a cropped photo of a {}.",
"a photo of the {}.",
"a good photo of the {}.",
"a rendering of the {}.",
"a close-up photo of the {}.",
"a low resolution photo of a {}.",
"a rendition of the {}.",
"a photo of the clean {}.",
"a photo of a large {}.",
"a blurry photo of a {}.",
"a pixelated photo of a {}.",
"itap of the {}.",
"a jpeg corrupted photo of the {}.",
"a good photo of a {}."
```

Note that $\{\}$ represent the placeholder for the location of the class name w.

Random prompts: For the experiments involving random prompts, we list bellow a few such examples:

```
"Ports, waterways, the subfield that {}.",
"In TCP, prepared mind, but some others, Milatiai, appear to have {}.",
"Iron Age, The Eastern Shore of Virginia residents age 5 and {}.",
"Cat mostly all with {}.",
"Wind erosion. go unnoticed|it was {}.",
"River Delta, on six different {}.",
"12 hours. few times every million {}.",
etc.
```

Additional class names for in-domain ablation: Below, we list the manually defined in-domain class name distractors used to produce the results reported in Table 4. For Food-101, we added the following classes:

['aroma', 'bagel', 'batter', 'beans', 'biscuit', 'broth', 'burger', 'burrito', 'butter', 'candy', 'caramel', 'caviar', 'cheese', 'chili', 'chimichanga', 'cider', 'cocoa', 'coffee', 'cobbler', 'empanada', 'fish', 'flour', 'ketchup', 'margarine', 'mousse', 'muffin', 'mushrooms', 'noodle', 'nuts',

'oil', 'olives', 'pudding', 'raclette', 'rice', 'salad', 'salsa', 'sandwitch', 'soda', 'tea', 'stew', 'toast', 'waffles', 'yogurt', 'wine', 'sopapillas', 'chilli con carne', 'banana bread', 'yorkshire pudding', 'spaghetti carbonara', 'roast potatoes', 'sausage ragu', 'avocado panzanella', 'lamb biryani']

Respectively, for Flowers102 dataset:

['Agapanthus', 'Allium', 'Alstroemerias', 'Amaranthus', 'Astilbe', 'Begonia', 'brunia', 'California poppy', 'Calla lily', 'Campanula', 'Carnations', 'Celosia', 'Chrysanthemum', 'Cornflower', 'Delphinium', 'Dianthus', 'Dusty Miller', 'Eryngium', 'Freesia', 'Gardenias', 'Gerbera daisies', 'Gladiolus', 'Gypsophila', 'Hydrangea', 'Hypericum', 'Kale', 'Larkspur', 'Liatris', 'Lilies', 'Lisianthus', 'Orchids', 'Peony', 'Periwinkle', 'Ranunculus', 'Scabiosa', 'Sunflowers', 'Yarrow', 'Zinnia', 'Bellflower', 'Bleeding Heart', 'Browallia', 'Bugleweed', 'Butterfly Weed', 'Calendula', 'Cardinal Flower', 'Celosia', 'Clary Sage', 'Coreopsis', 'Forget-Me-Not', 'Freesias', 'Gaillardia', 'Glory of the Snow', 'Heather', 'Hollyhock', 'Hyssop', 'Impatiens', 'Jack-in-the-Pulpit', 'Lilac', 'Lilies', 'Lobelia', 'Periwinkle', 'Rue', 'Thunbergia', 'Verbena', 'Wisteria']