LASP: Text-to-Text Optimization for Language-Aware Soft Prompting of Vision & Language Models

Adrian Bulat[1,2], Georgios Tzimiropoulos[1,3]
[1]Samsung AI Cambridge [2]Technical University of Iasi [3]Queen Mary University of London

Abstract

Soft prompt learning has recently emerged as one of the methods of choice for adapting V&L models to a downstream task using a few training examples. However, current methods significantly overfit the training data, suffering from large accuracy degradation when tested on unseen classes from the same domain. To this end, in this paper, we make the following 4 contributions: (1) To alleviate base class overfitting, we propose a novel Language-Aware Soft Prompting (LASP) learning method by means of a text-to-text cross-entropy loss that maximizes the probability of the learned prompts to be correctly classified with respect to pre-defined hand-crafted textual prompts. (2) To increase the representation capacity of the prompts, we propose grouped LASP where each group of prompts is optimized with respect to a separate subset of textual prompts. (3) We identify a visual-language misalignment introduced by prompt learning and LASP, and more importantly, propose a re-calibration mechanism to address it. (4) We show that LASP 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 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 8 out of 11 test datasets. Code will be made available here.

1. Introduction

Large-scale pre-training of neural networks has recently resulted in the construction of a multitude of foundation models for Language [7,25] and Vision & Language (V&L) understanding [1, 13, 24, 34]. 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 [13, 18, 24, 33, 34]), 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 [16, 17, 24], prompt engineering and learning has emerged as one of the most powerful techniques for adapting a V&L to new tasks. Initially, in [24], 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 [16, 17], subsequent work [35, 36] 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 entire V&L model kept frozen. The whole process can be seen as parameter efficient fine-tuning of the model on a small training dataset.

However, a clearly identifiable problem with prompt learning is base class 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.

Key idea: To alleviate base class overfitting, in this work, we propose a solution motivated by the following observation: since prompt learning improves the accuracy on base classes, but prompt engineering is significantly 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 textual ones, thus exploiting the intrinsic information captured by the text encoder. The proposed text-to-text loss enables language-only optimization for V&L model adaption

for the first time. This is in contrast with prior soft-prompt learning methods that only capture V&L interactions.

Key contributions: Based on the above, we propose a novel framework for soft prompt learning which we call Language-Aware Soft Prompting (LASP). Our main contributions within the LASP framework are as follows:

- We propose, for the first time, language-only optimization for V&L model adaption. Specifically, we propose a novel text-to-text cross-entropy loss that maximizes the probability of the learned prompts to be correctly classified with respect to the hand-engineered ones and show its effectiveness in terms of alleviating base-class overfitting.
- To increase the representation capacity of the prompts, and inspired by grouped convolution and multi-head attention, we propose a grouped language-aware prompt representation where *each group* of prompts specializes to a different subset of the pre-defined manual templates.
- We identify a visual-language misalignment introduced by prompt learning and LASP which impacts the generalization. More importantly, we propose a re-calibration mechanism based on (a) Layer Normalization fine-tuning and (b) learning a class-agnostic bias to address it.
- Thanks to our language-only learning framework, we propose training LASP with virtual classes by including, during training, class names for which no visual samples are available. Importantly, we show that this further increases the robustness of the learned prompts.

Main results: 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 (8 out of 11), the very strong baseline based on hand-crafted prompts and CLIP for the recognition of novel classes (i.e. zero-shot setting).

2. Related work

Contrastive V&L 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 [13, 18, 24, 33, 34]. Such networks consist of a vision encoder (typically a ViT [8]) and a Transformer-based text encoder [30]. Highly parameterized instantiations of such architectures are trained on large corpora of image-caption pairs (e.g. [24] uses 400M and [13] 1B pairs) using contrastive learning. We used CLIP [24] 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 [4, 25]. In [27, 28], 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 [16, 17].

In V&L foundation models, like CLIP, the class names are used to create hand-crafted prompts [24] that are fed as input to the text encoder, enabling zero-shot visual recognition. CoOp [36] 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 [35] proposes a dynamic version of [36] 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). ProDA [19] adopts a probabilistic approach by modelling the distribution of the prompts at the output of the text encoder as a multivariate Gaussian distribution. The estimated mean is used during inference. Finally, UPL [12] uses CLIP to generate pseudo-labels on the target dataset and then self-training to learn the soft prompts. Finally, ProGrad [37] aims to adapt the V&L model to each target domain by encouraging it "not to forget" CLIP's zero-shot predictions using a KL visual-text loss between the CLIP's logits and their model's logits (*i.e*. they use visual features). The weights are then updated in the direction perpendicular to CLIP gradients. In contrast, our loss is a pure text-to-text loss, further allowing for the incorporation of virtual classes. Unlike [37], we outperform CLIP on novel classes.

The proposed LASP framework alleviates base class overfitting and significantly improves upon the previously reported best results without resorting to a dynamic approach as in CoCoOp [35]. In its basic version, LASP deploys a text-to-text loss that enforces the learned prompts to be "close" to a set of manually defined textual prompts in the text encoder space. Importantly, the basic LASP can be extended in three important ways: (1) by allowing the incorporation of virtual classes, i.e. novel class name information for which no (visual) training data is available (LASP-V). This is shown to significantly improve the robustness of the learned prompts at no extra cost during inference; (2) by allowing the use of a grouped prompt representation within the proposed language-aware training which is shown to increase the representation capacity of the learned prompts; (3) by performing further optimization of the visual encoder

so that the visual and text embeddings are realigned resulting in significant accuracy gains. Notably, our approach is very efficient (as efficient as [36]) as opposed to [35] 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, $\texttt{class\_name}_c$, $c \in \{1, \dots, C\}$, 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:

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

where τ is a temperature factor and $\cos$ the cosine similarity. Finally, the class for $\mathbf{x}$ is given by $\tilde{y} = \arg_{max}\, P_h(y|\mathbf{x})$. 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 [16, 17, 36] is concerned with parameter efficient fine-tuning of a pre-trained V&L model by learning a sequence of M learnable vectors $\mathbf{p}_m \in \mathbb{R}^d, m = \{1, \dots, M\}$ using a few labelled 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: $\mathbf{r}_c = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_M, \mathbf{w}_c\}$, 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:

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

The prompts can be learned by minimizing the cross-entropy loss:

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

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.

3.2. Language-Aware Soft Prompting (LASP)

Despite its strong performance on base classes, vanilla soft prompt learning (see Sec. 3.1) under-performs on novel classes (i.e. zero-shot setting). While CoCoOp [36] 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.

In this work, we propose, for the first time, language-only optimization for V&L downstream adaption. This is in contrast with prior soft-prompt learning methods that only capture V&L interactions. Specifically, since the hand-engineered textual prompts outperform the learnable soft prompts for the zero-shot setting, then, in order to avoid base-class overfitting and strengthen generalizability, we propose that the learnable ones should be trained so that they can be correctly classified in language space where the class weights are given by the textual 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 textual ones. Specifically, recall that $\mathbf{t}_c^h = g_T(h_c)$ is the class weight for class c obtained by encoding the corresponding textual prompt. Assuming that L manually defined textual prompts are available [1], we have $\mathbf{t}_c^{h,l}$, $l = 1, \dots, L$. Moreover, $\mathbf{t}^r$ is an encoded learnable prompt to be classified in one of the C classes. Finally, the probability of prompt $\mathbf{t}^r$ being classified as class y is:

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

The language-aware training loss is computed similarly to the V&L loss:

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

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 1. **Interpretations:** LASP can be interpreted in a number of ways:

[1] The original CLIP prompts serve as textual prompts without any tweaking or change. Note, that our method can even work with random sentences (see Sec. 4.2).

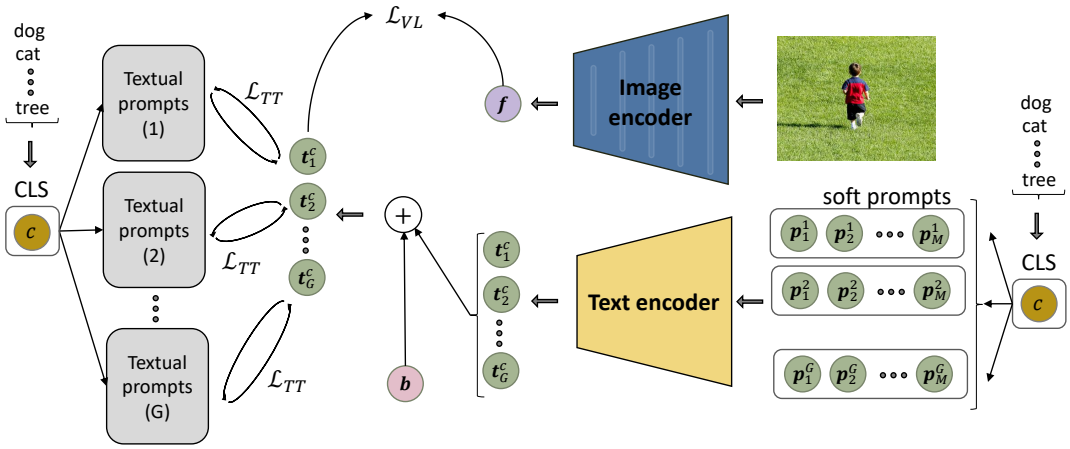


Figure 1. **Overall idea.** While standard prompt learning is based on image-text interactions (L_{VL} loss; Eq. 3), LASP additionally models text-text interactions using the proposed Text-to-Text loss L_{TT} (Eq. 5). There are G groups of learned prompts $\mathbf{p}_i^j$ passed through the text encoder to form G text embeddings $\mathbf{t}_j$ summarizing the input. The L_{TT} loss is then applied over the different groups of the text embeddings and the textual prompts. Moreover, to alleviate data distribution shift and visual-language misalignment, the LN layers of the visual encoder are fine-tuned and the embeddings are "corrected" at the output space by the learnable vector $\mathbf{b}$, shared for all classes. The text encoder remains entirely frozen. Notably, LASP can be trained with virtual classes by including, during training, class names for which no visual samples are available.

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 [36]. As the proposed language-aware loss encourages the learned prompts to be close in embedding space to the textual 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, color 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*".

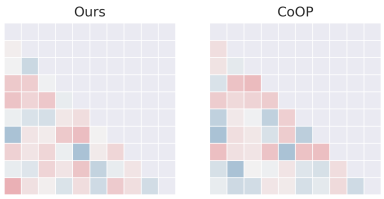
At train time, as reflected by Eq. 4, we compute the class label distribution per $l-$th template and then average over all templates. Hence, we opt not to mix across templates during training as we want the model to focus on class information solely. For example, the model could distinguish easier between 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 by 0.5% on novel classes).

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 visualized in Fig. 2 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.

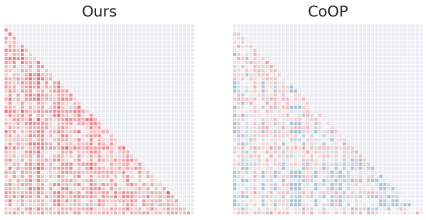
LASP as data-free distillation: Typically, knowledge distillation requires a training set of images, where a teacher network provides a training signal for the student [11]. LASP's text-to-text loss can be also interpreted as a data-free distillation (*i.e.* does not use any image data) where the learnable prompts define the "samples". As CLIP learns a joint V&L space, similar concepts are close together across both domains. Hence, optimizing against a concept or object in the language domain, using the proposed loss, should also help make a step in the visual domain, improving the classification of the images.

3.3. Grouped LASP

Grouped convolutions [15] and multi-head attention [30] have been shown to learn strong representations. The groups or the number of heads, respectively, can be also interpreted as a set of experts that are then combined to produce a strong feature. Drawing inspiration from this, we propose a grouped prompt representation, where each group is optimized with respect to a separate subset of textual prompts. Effectively, the prompts from each group will learn a transformation specialized to its corresponding subset (analogous to the aforementioned techniques that also



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



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

Figure 2. **Cosine distance between the class embeddings** produced by the CLIP text encoder on Eurosat and DTD for LASP and CoOp. Class centroids situated further apart are more separable as the underlying image features are identical across both methods. Brighter colors indicate bigger distances. The numbers shown are the average cosine distance between the classes.

specialize to a part of the signal). In particular, we split the set of L templates into G equally sized sub-sets. Moreover, each sub-set is associated with a sequence of M prompts $\mathbf{r}_c^g = \{\mathbf{p}_1^g, \ldots, \mathbf{p}_M^g, \mathbf{w}_c\}, g = 1, \ldots, G$ each producing a class specific text feature $\mathbf{t}_c^{r,g} = g_T(\mathbf{r}_c^g)$. Finally, our text-to-text loss in Eq. 5 becomes:

$$\mathcal{L}_{TT-G} = -\sum_{g=1}^{G}\sum_{c=1}^{C} \log P_{rh}^g(c|\mathbf{t}^g) y_c, \tag{7}$$

with P_{rh}^g computed for each group similarly to Eq. 4. We note that the splits were created randomly. As the text templates are in general semantically independent, no preferred grouping arises. At test time, the final result is computed by taking the average of the cosine similarity scores between each group and the visual feature $\mathbf{f}$.

3.4. Re-aligning LASP

Combating data distribution shift: for some downstream tasks, it is possible that there is a data distribution shift between the downstream image dataset and the one used by CLIP during training. Hence, we would like this aspect to be captured by the downstream adaptation method. To this end, some optimization of the visual encoder can be performed; nevertheless this can very easily result in base class overfitting if, after the training, the V&L embeddings are pushed away from the joint space learned by CLIP. For example, preliminary results with visual adapters have shown that they hurt zero-shot accuracy. On the contrary, we found that Layer Normalization (LN) [2] fine-tuning is a much more robust way to adapt the visual encoder. Overall, we propose fine-tuning the LN of the CLIP encoder as a way to combat distributional shift.

Combating V&L misalignment: Because after LN fine-tuning the V&L are not guaranteed to continue to be aligned, we also propose to learn a "correction" at the output of the text encoder in the form of a learnable offset (bias) that aims to re-align the two modalities. Let $\mathbf{W}$ be the set of weights of the linear classifier obtained by passing the learned prompts from the text encoder. We propose to learn a vector $\mathbf{b} \in \mathbb{R}^d$ that is simply added to $\mathbf{W}$, that is $\mathbf{W} = \mathbf{W} + \mathbf{b}$. Importantly, the learned offset is shared among all classes, and in this way it can be readily applied for the case of novel classes too.

3.5. LASP with Virtual Classes (LASP-V)

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 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, LASP with virtual classes can significantly improve the robustness of the prompts learned. We refer to this variant of our method as **LASP-V**.

Note that training with virtual classes does not violate the zero-shot setting [31][2]. Moreover, from a practical perspective, if the novel class names are not known during initial training, the model can be simply retrained in a zero-shot manner when they become available.

4. Experiments

Following [25, 35], 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 including cross-dataset transfer and domain generalization, see Supp. Mat.

Datasets: We used 11 in total, namely: ImageNet [6], Caltech101 [9], Oxford-Pets [22], Stanford Cars [14], Flowers102 [21], Food101 [3], FGVC Aircraft [20], SUN397 [32], DTD [5], EuroSAT [10] and UCF-101 [29].

Models: For all experiments, unless otherwise specified, we used a pretrained CLIP model with a ViT-B/16 image

[2] according to [31] "Zero-shot learning aims to recognize objects whose **instances** may not have been seen during training."

encoder, $M = 4$ learnable prompts and 16 samples per class. The number of groups G (when used) is set to 3. In all experiments, we report the average across 3 runs.

Training: Largely, we followed the training procedure described in CoOp [36] and CoCoOp [35] (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 textual 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 [23].

Methods compared: We report the performance of LASP and its improved version trained with virtual classes (LASP-V). For LASP-V, 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 [36], as we add the proposed components on top of it. 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 ProDA [19] and CoCoOp [35] which conditions the prompts on image features and hence induces significant additional computation during inference.

4.1. Comparison with state-of-the-art

Standard setting of [35]: Table 1 compares our approach with the current state-of-the-art. We conclude:

- **Conclusion 1: In terms of harmonic mean, LASP outperforms all methods by large margin.** It outperforms, on average, the second best (ProDA) by $> 2\%$. The improvement on specific datasets is even bigger (e.g. $> 3\%$ on Flowers102, $> 11\%$ on EuroSAT, $> 3\%$ on UCF101).

- **Conclusion 2: On the novel classes, LASP outperforms all methods by large margin.** It is the first reported method outperforming CLIP by 0.68% (but notice that CLIP performs very poorly on the bases classes). It also outperforms ProDA (third best) by $> 2.5\%$. Again, compared to ProDA, the improvement on specific datasets is even bigger (e.g. $> 5\%$ on Flowers102, $> 3\%$ on Food101, $> 11\%$ on EuroSAT, $> 6\%$ on UCF101).

- **Conclusion 3: On new classes, LASP with virtual classes has significant impact for specific datasets**. These include datasets with informative class names like EuroSAT and DTD where the improvement over LASP is $\sim 5.5\%$ and $\sim 4.0\%$, respectively.

Generalized zero-shot setting: The current evaluation protocol used in [35] 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. Detailed results for this setting are provided in the Supp. Mat., but, in general, the same conclusions as above hold.

Table 1. **Comparison with the state-of-the-art on 11 datasets**. We provide the results of LASP and LASP trained with virtual classes (LASP-V). Δ denotes the absolute improvement of our best variant, LASP-V, over the previous best result.

Dataset	Set	CLIP [24, 36]	CoOp [36]	CoCoOp [35]	ProDA [19]	LASP (Ours)	LASP-V (Ours)	Δ
Average	Base	69.34	82.69	80.47	81.56	82.70	**83.18**	**+0.49**
	New	74.22	63.22	71.69	72.30	74.90	**76.11**	**+1.89**
	H	71.70	71.66	75.83	76.65	78.61	**79.48**	**+2.83**
ImageNet	Base	72.43	**76.47**	75.98	75.40	76.20	76.25	**-0.22**
	New	68.14	67.88	70.43	70.23	70.95	**71.17**	**+0.74**
	H	70.22	71.92	73.10	72.72	73.48	**73.62**	**+0.52**
Caltech101	Base	96.84	98.0	97.96	**98.27**	98.10	98.17	**-0.10**
	New	94.0	89.91	93.81	93.23	94.24	**94.33**	**+0.33**
	H	95.40	93.73	95.84	95.86	96.16	**96.21**	**+0.35**
OxfordPets	Base	91.17	93.67	95.20	95.43	**95.90**	95.73	**+0.30**
	New	97.26	95.29	97.69	97.83	**97.93**	97.87	**+0.04**
	H	94.12	94.47	96.43	96.62	**96.90**	96.79	**+0.16**
Stanford Cars	Base	63.37	**78.12**	70.49	74.70	75.17	75.23	**-2.89**
	New	**74.89**	60.40	73.59	71.20	71.60	71.77	**-3.12**
	H	68.85	68.13	72.01	72.91	73.34	**73.46**	**+0.55**
Flowers102	Base	72.08	97.60	94.87	**97.70**	97.0	97.17	**-0.53**
	New	**77.80**	59.67	71.75	68.68	74.0	73.53	**-4.27**
	H	74.83	74.06	81.71	80.66	83.95	**83.71**	**+2.0**
Food101	Base	90.10	88.33	90.70	90.30	91.20	**91.20**	**+0.50**
	New	91.22	82.26	91.29	88.57	91.70	**91.90**	**+0.61**
	H	90.66	85.19	90.99	89.43	91.44	**91.54**	**+0.55**
FGVC Aircraft	Base	27.19	**40.44**	33.41	36.90	34.53	38.05	**-2.39**
	New	**36.29**	22.3	23.71	34.13	30.57	33.20	**-3.09**
	H	31.09	28.75	27.74	**35.46**	32.43	**35.46**	**0.0**
SUN397	Base	69.36	80.6	79.74	78.67	80.70	**80.70**	**+0.10**
	New	75.35	65.89	76.86	76.93	78.60	**79.30**	**+2.37**
	H	72.23	72.51	78.27	77.79	79.63	**80.0**	**+1.73**
DTD	Base	53.24	79.44	77.01	80.67	81.4	**81.10**	**+1.53**
	New	59.9	41.18	56.0	56.48	58.6	**62.57**	**+3.10**
	H	56.37	54.24	64.85	66.44	68.14	**70.64**	**+4.20**
EuroSAT	Base	56.48	92.19	87.49	83.90	94.60	**95.0**	**+2.81**
	New	64.05	54.74	60.04	66.0	77.78	**83.37**	**+17.37**
	H	60.03	68.9	71.21	73.88	85.36	**88.86**	**+14.98**
UCF101	Base	70.53	84.69	82.33	85.23	84.77	**85.53**	**+0.30**
	New	77.50	56.05	73.45	71.97	78.03	**78.20**	**+0.70**
	H	73.85	67.46	77.64	78.04	81.26	**81.70**	**+3.66**

4.2. Ablation studies

Effect of different LASP components: LASP proposes a number of contributions which are evaluated incrementally. The start point is the proposed Text-to-Text loss of Eq. 5. On top of this, we incrementally apply the grouped prompt representation (Eq. 7), and then the re-alignment module (Sec. 3.4). This gives rise to LASP. Finally, we add virtual

Table 2. **Effect of different LASP components.** Text-to-Text is Eq. 5, only. On top of this, we incrementally apply the grouped prompt of Eq. 7, and the re-alignment module of Sec. 3.4. Up to this point, this is equiv. to LASP. Finally, we add virtual classes (equiv. to LASP-V). Baseline is CoOp.

Dataset	Set	Baseline [36]	Text-to-Text	+Grouped	+Align (LASP)	+ Virtual (LASP-V)
Average	Base	82.69	81.26	81.87	82.70	**83.18**
	New	63.22	71.54	73.48	74.90	**76.11**
	H	71.66	76.09	77.44	78.61	**79.48**
ImageNet	Base	**76.47**	75.97	76.20	76.20	76.25
	New	67.88	70.31	70.70	70.95	**71.17**
	H	71.92	73.03	73.34	73.48	**73.62**
Caltech101	Base	98.0	97.70	97.97	98.10	**98.17**
	New	89.91	94.08	94.27	94.24	**94.33**
	H	93.73	95.85	96.08	96.16	**96.21**
OxfordPets	Base	93.67	95.13	95.63	**95.90**	95.73
	New	95.29	96.23	97.87	**97.93**	97.87
	H	94.47	95.68	96.73	**96.90**	96.79
Stanford Cars	Base	**78.12**	72.46	73.50	75.17	75.23
	New	60.40	71.80	**72.1**	71.60	71.77
	H	68.13	72.19	72.93	73.34	**73.46**
Flowers102	Base	**97.60**	96.47	96.80	97.0	97.17
	New	59.67	70.7	**74.0**	**74.0**	73.53
	H	74.06	81.59	83.87	**83.95**	83.71
Food101	Base	88.33	90.30	91.0	**91.20**	**91.20**
	New	82.26	90.73	90.87	91.70	**91.90**
	H	85.19	90.51	90.93	91.44	**91.54**
FGVC Aircraft	Base	**40.44**	32.63	33.05	34.53	38.05
	New	22.3	30.46	31.80	30.57	**33.20**
	H	28.75	31.57	32.41	32.43	**35.46**
SUN397	Base	80.6	80.20	80.55	**80.70**	**80.70**
	New	65.89	75.56	77.11	78.60	**79.30**
	H	72.51	77.81	78.79	79.63	**80.0**
DTD	Base	79.44	79.13	80.5	**81.4**	81.10
	New	41.18	52.1	56.20	58.6	**62.57**
	H	54.24	62.82	66.19	68.14	**70.64**
EuroSAT	Base	92.19	91.23	91.90	94.60	**95.0**
	New	54.74	63.16	66.37	77.78	**83.37**
	H	68.9	74.64	77.07	85.36	**88.86**
UCF101	Base	84.69	82.7	83.47	84.77	**85.53**
	New	56.05	71.80	77.07	78.03	**78.20**
	H	67.46	76.86	80.14	81.26	**81.70**

classes giving rise to LASP-V. Our baseline is CoOp. From the results of Table 2, we conclude:

- **Conclusion 4: Our idea in its plain form (Text-to-Text loss) outperforms its direct baseline (CoOp) by a large margin**. Specifically, it improves upon CoOp by $\sim 4.5\%$ on average, demonstrating its effectiveness.

- **Conclusion 5: All components are needed to obtain high accuracy.**

Effect of size and content of the textual prompts: Herein, we study the effect of the size L and the content of the set of the textual prompts used by our method in Eq. 4. For simplicity, we report results using our Text-to-Text loss (Eq. 5), only. The hand-crafted templates are increased to 100 by including the rest of the prompts defined in CLIP [24], 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 hyperparameters, except for the case of random templates, where $\alpha_{TT} = 5$.

Table 3 shows our results. We importantly note that the accuracy on the base classes remains similar across all settings (not shown in the table). Moreover, we conclude:

- **Conclusion 6: The exact choice of the templates might not be so significant for the few-shot setting.**

- **Conclusion 7: For the case of novel classes, both the number and the content of the templates are important to obtain high accuracy.**

Effect of type of loss: In Table 6, we vary the choice of loss in LASP, *i.e.* we replace the Cross-Entropy (CE) with an L_2 and L_1 loss. Again, for simplicity, we report results using our Text-to-Text loss (Eq. 5), only.

- **Conclusion 8: The proposed CE loss based formulation outperforms other losses for LASP.**

Effect of out-domain distractors: Motivated by the recent work of [26] 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-V is the only method that benefits from knowledge of this expanded vocabulary during training. From Table 4, we can conclude:

- **Conclusion 9: The models are somewhat robust to out-of-domain distractors.** Specifically, the drop in accuracy is moderate (typically 1-2%). The exception is EuroSAT where the number of classes increases $25\times$. Importantly, LASP-V manages to largely recover the lost accuracy.

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 experiments on two datasets: Food101 and Flowers102. For Flowers102,

Table 3. **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. For simplicity, we report results using our Text-to-Text loss (Eq. 5), only. Text-to-Text (R) denotes models trained using randomly constructed templates.

(a) DTD.

#Templates	1	34	100
Text-to-Text (R)	49.02	51.63	52.64
Text-to-Text	**50.73**	**52.10**	**56.53**

(b) EuroSAT.

#Templates	1	34	100
Text-to-Text (R)	55.01	59.9	62.1
Text-to-Text	**56.97**	**63.16**	**65.13**

(c) UCF101.

#Templates	1	34	100
Text-to-Text (R)	67.5	68.6	70.03
Text-to-Text	**71.36**	**71.80**	**72.77**

Table 4. **Effect of out-domain distractors.** w/o distractors are the results on the generalized zero-shot setting.

(a) EuroSAT.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	86.25	64.63	73.89	86.0	55.80	67.68
LASP-V	**90.0**	**65.73**	**75.97**	**90.8**	**59.87**	**72.16**

(b) Food101.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	**87.17**	87.53	87.34	**87.01**	86.90	86.95
LASP-V	**87.17**	**87.63**	**87.39**	86.99	**87.10**	**87.04**

(c) Flowers102.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	90.97	67.8	77.69	90.0	67.1	76.68
LASP-V	**93.20**	**69.93**	**79.9**	**92.05**	**69.08**	**78.92**

(d) OxfordPets.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	**92.53**	**94.20**	**91.52**	91.53	92.60	92.06
LASP-V	92.25	93.97	93.10	**92.23**	**93.17**	**92.69**

Table 5. **Effect of in-domain distractors.** w/o distractors are the results on the generalized zero-shot setting evaluation.

(a) Food101.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	**87.17**	87.53	87.34	82.70	83.47	83.08
LASP-V	**87.17**	**87.63**	**87.39**	**83.11**	**83.95**	**83.52**

(b) Flowers102.

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	90.97	67.8	77.69	80.16	62.50	70.23
LASP-V	**93.20**	**69.93**	**79.9**	**83.95**	**65.31**	**73.47**

we added 65 new class names while, for Food101, 53 new classes. Note again that, except for LASP-V, 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 Supp. Mat. From the results of Table 5, we conclude:

- **Conclusion 10: In-domain distractors significantly increase the problem difficulty.** Specifically, the drop in accuracy is large (4-7%). LASP-V manages to recover part of the lost accuracy.

Table 6. **Effect of type of loss.** For simplicity, we report results using our Text-to-Text loss (Eq. 5), only.

Set	CE	L_1	L_2
Base	81.26	**81.50**	81.47
New	**71.54**	66.01	65.80
H	**76.09**	73.54	72.80

5. Conclusions

In this paper, we introduced LASP - a language aware soft prompting method for V&L adaptation that is shown to outperform prior work by large margin. Specifically, we made the following contributions: *Firstly*, we introduced a novel text-to-text loss that largely alleviates the problem of base-class overfitting. *Secondly*, we proposed a *grouped* language-aware prompting for learning more specialized and stronger prompt representations. *Thirdly*, we identified a visual-language misalignment within LASP and propose a re-calibration mechanism to address it. *Fourthly*, we showed that our approach, unlike prior work, is amenable to, including during training, *virtual classes*, i.e. class names for which no visual samples are available, significantly increasing the robustness of the learned prompts. We hope that LASP/LASP-V will serve as a strong baseline for future works in the area of few-shot adaptation for V&L models.

References

[1] 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. 1

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 5

[3] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer, 2014. 5

[4] 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. 2

[5] 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*, pages 3606–3613, 2014. 5

[6] 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*, pages 248–255. Ieee, 2009. 5

[7] 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. 1

[8] 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. 2

[9] 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*, pages 178–178. IEEE, 2004. 5

[10] 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. 5

[11] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015. 4

[12] Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022. 2

[13] 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*, pages 4904–4916. PMLR, 2021. 1, 2

[14] 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*, pages 554–561, 2013. 5

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 4

[16] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 1, 2, 3

[17] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021. 1, 2, 3

[18] 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. 1, 2

[19] Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 2, 6

[20] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 5

[21] 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*, pages 722–729. IEEE, 2008. 5

[22] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012. 5

[23] 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. 6

[24] 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*, pages 8748–8763. PMLR, 2021. 1, 2, 6, 7

[25] 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. 1, 2, 5

[26] Shuhuai Ren, Lei Li, Xuancheng Ren, Guangxiang Zhao, and Xu Sun. Rethinking the openness of clip. *arXiv preprint arXiv:2206.01986*, 2022. 7

[27] Timo Schick and Hinrich Schütze. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*, 2020. 2

[28] 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*, 2020. 2

[29] 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. 5

[30] 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. 2, 4

[31] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4582–4591, 2017. 5

[32] 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*, pages 3485–3492. IEEE, 2010. 5

[33] 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. 1, 2

[34] 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. 1, 2

[35] 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*, pages 16816–16825, 2022. 1, 2, 3, 5, 6

[36] 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, 2022. 1, 2, 3, 4, 6, 7

[37] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. *arXiv preprint arXiv:2205.14865*, 2022. 2