Can Better Text Semantics in Prompt Tuning Improve VLM Generalization?

Hari Chandana Kuchibhotla*, Sai Srinivas Kancheti*, Abbavaram Gowtham Reddy,
Vineeth N Balasubramanian
Indian Institute of Technology Hyderabad, India
{ai20resch11006, cs21resch01004, cs19resch11002, vineethnb}@iith.ac.in

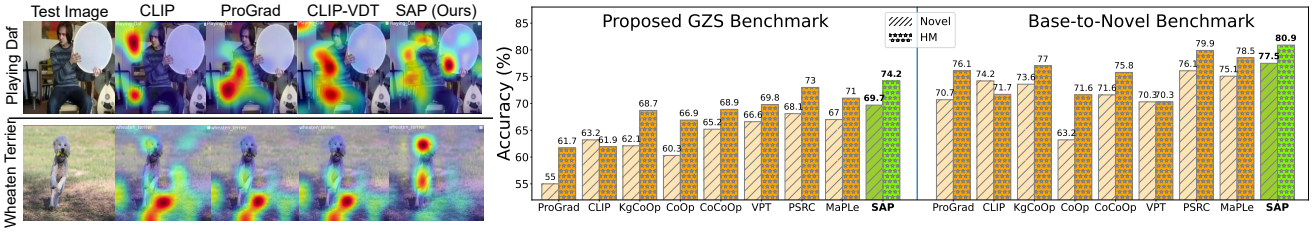

Figure 1. **Left:** Comparison of GradCAM [10] visualizations for our proposed method SAP against other baselines, on novel classes *"Playing Daf"* from UCF101 [12] and *"Wheaten Terrier"* from Oxford Pets [7]. The saliency maps indicate image regions that are most relevant to the descriptions "A photo of a person playing daf, with fingers hitting the drumhead" and "A photo of a wheaten terrier, which is wheat colored and has a square-shaped head" respectively. SAP effectively localizes the text semantics in images compared to baselines. **Right:** SAP surpasses other baselines on Generalized Zero-Shot (GZS) and Base-to-Novel (B2N) benchmarks, showing improvements of **+1.6%** and **+1.2** on Novel Accuracy and Harmonic Mean (HM) for GZS, and **+1.4%** and **+0.9** for B2N compared to best performing baseline.

Abstract

Learnable prompt tuning has emerged as a resource-efficient alternative to fine-tuning vision-language models (VLMs). However, challenges include overfitting of learnt prompts in low-shot scenarios and decreased performance in large class spaces. In this work, we propose a prompt-tuning method that leverages class descriptions from large language models (LLMs) to construct part-level description-guided views of image and text features, which are aligned. Our experiments across 11 benchmark datasets demonstrate substantial improvements over baselines.

1. Introduction

Foundational VLMs like CLIP [8] have displayed remarkable zero-shot and open-vocabulary capabilities in recent times. This has led to VLMs being employed successfully in various vision-only downstream tasks. Recently, learnable prompt-tuning [14] has emerged as a promising parameter-efficient alternative for fine-tuning foundation models. However, some challenges include (i) training such prompts in a low-shot setting, which leads to overfitting and hinders their generalizability, causing sub-optimal performance when applied to newer classes, and (ii) the performance of prompt-tuning methods being highly dependent on the label space in which they operate. If inference time label space is large, performance tends to decrease due to bias towards the classes the model was fine-tuned on. This prods us to ask the question: *can better text semantics in prompt tuning improve VLM generalization?*

Based on our studies in this direction, we propose *SAP*, **S**emantic **A**lignment for **P**rompt-tuning, which utilizes auxiliary text information in the form of easily obtained class descriptions to learn generalizable prompts. More specifically, given a set of class descriptions, we show how to construct *class description-guided views* of both image and text features. We then define semantic alignment as the average of similarities between corresponding views of image and text features. Due to our part-level image-text alignment, SAP also showcases superior localization of visual concepts relevant to a class description, as seen through class activation maps (see Fig. 1). Most existing prompt-tuning methods do not use such additional text semantics; even among the recent few that use such information, our method utilizes class descriptions at a part-level for both image and text. To summarize our contributions – i) We propose a prompt-tuning method that can leverage class descriptions obtained from an LLM, to perform part-level image-text alignment. ii) We propose two new evaluation protocols: GZS evaluation and *classification without class name* to better study the generalizability of prompt-tuning methods for VLMs. iii) We carry out a comprehensive suite of exper-

* Equal contribution

iments with comparisons against state-of-the-art and very recent methods on standard benchmark datasets. We outperform existing baselines with a significant margin on all evaluation protocols.

2. SAP: Methodology

We propose SAP, **S**emantic **A**lignment for **P**rompt-tuning, which utilizes auxiliary information in the form of class descriptions obtained from LLMs to learn more generalizable prompts. Our method constructs *description guided text views* and *description guided image views* that are aligned with each other, as shown in Fig. 2. Conceptually, a class description provides a semantic context, and a *description guided view* is a feature conditioned on this context. This external semantic knowledge, when integrated into the model, transfers to novel classes since the semantics are common concepts shared across multiple classes.

Generating Class Description Features. We use the popular LLM GPT-3.5 to obtain text descriptions for each class in a given dataset. Class descriptions commonly contain visual cues such as shape, texture, and color, as well as narratives of objects commonly correlated with the class. To keep our method cost-efficient, we use descriptions that are class-specific but not image-specific, thus making them reusable for many images. For each class $y \in \mathcal{Y}$, where $\mathcal{Y}$ is the label space under consideration, we denote by A_y the set of generated class descriptions. Let $A = \bigcup_{y\in\mathcal{Y}} A_y$, $N = |A|$ denote the set of descriptions of all classes and the size of the set, respectively. *Class description features* $\mathbf{t}^A \in \mathbb{R}^{N\times d}$ are obtained by passing the descriptions through text-encoder $\mathcal{T}$ (description features for y are $\mathbf{t}^{A_y}$).

Leveraging Class Descriptions for Text Features. The text feature $\mathbf{t}^y$ for a class $y \in \mathcal{Y}$ is generally obtained by encapsulating the class name in a text template and passing it through $\mathcal{T}$. We use aforementioned class descriptions A_y to construct $|A_y|$ description guided views of this text feature. For a class description $a \in A_y$, the description guided text template for y looks like: `'a photo of a [y], which has [a]'` (e.g., `'a photo of a cat, which has whiskers'`). Learnable text prompts are then added to the above token sequence, and are passed through $\mathcal{T}$ to get the prompted, description guided view of class y conditioned on a, $\mathbf{t}_p^{y,a}$. The set of description guided text views of y is denoted as $\mathbf{t}_p^{y,A_y} \in \mathbb{R}^{|A_y|\times d}$. The various text views are averaged to generate a *unified text feature*, $t_{pu}^y = \frac{1}{|A_y|}\sum_{a\in A_y} \frac{\mathbf{t}_p^{y,a}}{\|\mathbf{t}_p^{y,a}\|}$.

Leveraging Class Descriptions for Image Features. An image batch of size B is passed through $\mathcal{I}$ (vision transformer) with output shape $B \times (1 + M + n) \times d'$, where 1 is for $\mathbf{cls}_\mathcal{I}$ token, M for grid tokens, n for prompt tokens, and d' for transformer dimension. The $\mathbf{cls}_\mathcal{I}$ token, passed through final $proj \in \mathbb{R}^{d'\times d}$ of $\mathcal{I}$, yields *global image features* $\mathbf{V}_g \in \mathbb{R}^{B\times d}$, capturing global context but lacking local semantics. We aim to use M grid tokens for local information, obtaining *grid image features* $\mathbf{V}_r \in \mathbb{R}^{B\times M\times d}$ by passing them through $proj$ with a learnable d-dimensional bias. This bias fine-tunes $proj$ for local details. Prompted features are denoted as $\mathbf{V}_{pg}$ and $\mathbf{V}_{pr}$ for global and grid features, respectively.

Constructing Description Guided Image Views using Grid Features. We use class description features $\mathbf{t}^A \in \mathbb{R}^{N\times d}$ together with prompted grid features of a given image $\mathbf{v}_{pr} \in \mathbb{R}^{M\times d}$ to construct N description guided views for the image. To this end, we compute a parameter-free cross-attention with descriptor features as queries and grid features as keys and values. The cross-attention map $\mathbf{x}_{att} \in \mathbb{R}^{N\times M}$ is $\mathbf{x}_{att} = row\_softmax(\mathbf{t}^A \cdot \mathbf{v}_{pr}^\intercal)$. Let $\hat{\mathbf{v}}_{pr} = \mathbf{x}_{att} \cdot \mathbf{v}_{pr} \in \mathbb{R}^{N\times d}$ denote the N *description guided views of the image*. The i^{th} row of $\hat{\mathbf{v}}_{pr}$ is a feature vector that describes the image in terms of the i^{th} description.

Pooling Image Views into Local Image Features. To tackle irrelevant descriptions, we introduce a *description relevance score* $\mathbf{r} \in [0,1]^N$, which quantifies each description's similarity to the image. This is computed as $\mathbf{r} = softmax(\mathbf{v}_{pg} \cdot (\mathbf{t}^A)^\intercal)$. We weigh each image view by its relevance score to compute a *local image feature* as $\mathbf{v}_{pl} = \mathbf{r} \cdot \hat{\mathbf{v}}_{pr}$, $\mathbf{v}_{pl} \in \mathbb{R}^d$.

Fusing Global and Local Image Features. For an image, the global feature $\mathbf{v}_{pg} \in \mathbb{R}^d$ encodes class information pertaining to the image and the local feature $\mathbf{v}_{pl} \in \mathbb{R}^d$ encodes finer visual context. The final image feature is a fusion of the global and local features. We give a higher weight to the *local score* of an image if the descriptions attend strongly to specific grid patches of the image. We define the local score as $l_s = \mathbf{r} \cdot max(\mathbf{x}_{att}, dim = 1)$, $l_s \in [0,1]$. The final *prompted unified image feature* $\mathbf{v}_{pu} \in \mathbb{R}^d$ is an l_s-weighted combination of the global and local features defined as $\mathbf{v}_{pu} = (1 - l_s) \cdot \mathbf{v}_{pg} + l_s \cdot \mathbf{v}_{pl}$.

Aligning Prompted Image and Text Features. Given labeled data from a downstream dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^B$ with label space $\mathcal{Y}$, we obtain the prompted and unprompted unified image features, $\mathbf{V}_{pu} \in \mathbb{R}^{B\times d}$ and $\mathbf{V}_u \in \mathbb{R}^{B\times d}$ respectively. For every class $y \in \mathcal{Y}$, description guided prompted and unprompted text features, $\mathbf{t}_p^{y,A_y} \in \mathbb{R}^{|A_y|\times d}$ and $\mathbf{t}^{y,A_y} \in \mathbb{R}^{|A_y|\times d}$ respectively, are obtained from $\mathcal{T}$. We denote the set of all learnable text and visual prompts by θ. Prompts are trained by minimizing the negative log-likelihood of the data:

$$L_{ce}(\theta) = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp(s\cdot O_\theta(\mathbf{x}_i, y_i))}{\sum_{y\in\mathcal{Y}} \exp(s\cdot O_\theta(\mathbf{x}_i, y))}$$

where $O_\theta(\mathbf{x}_i, y) = \frac{1}{|A_y|}\sum_{a\in A_y} sim(\mathbf{v}_{pu}^{\mathbf{x}_i}(\theta), \mathbf{t}_p^{y,a}(\theta))$, s is the scale (inverse temperature) parameter and logit $O(\mathbf{x}, y)$ is the similarity between image $\mathbf{x}$ and class y. $sim(\mathbf{v}_{pu}^{x_i}, \mathbf{t}_p^{y,a})$ is the similarity between the unified image

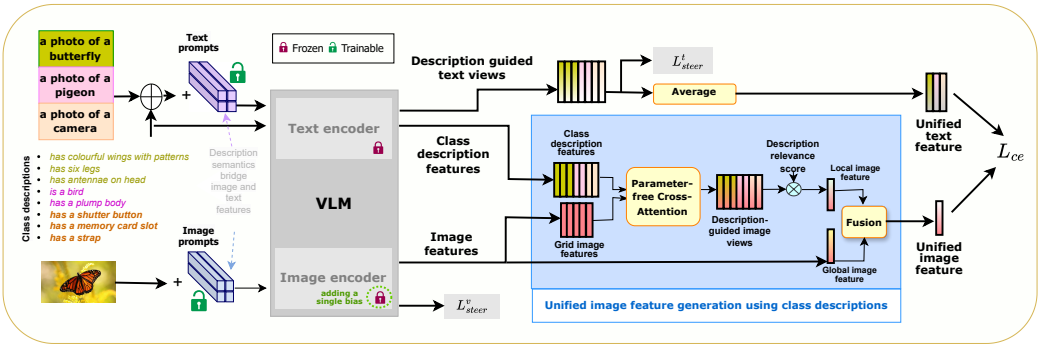

Figure 2. Our proposed workflow, SAP, performs part-based semantic alignment between image and text features. SAP integrates class descriptions into the text template and passed through the text encoder to construct description guided text views, which are then averaged to form a unified text feature. Global and grid image features are obtained from the image encoder. Description guided image views are obtained by performing parameter free cross-attention between class descriptor features and grid features. These image views are pooled into a local image feature, which is then fused with a global image feature to obtain the unified image feature. Unified text and image features contain part-level semantic information, and are aligned using a cross-entropy loss L_{ce}, and two steering losses L^v_{steer}, and L^t_{steer}.

Dataset		CLIP (ICML '21)	CoOp (IJCV '22)	VPT (ECCV '22)	CoCoOp (CVPR '22)	MaPLe (CVPR '23)	KgCoOp (CVPR '23)	ProGrad (ICCV '23)	PSRC (ICCV '23)	CLIP-VDT (ICCVW '23)	SAP (Ours)
Average	gBase	60.81	75.19	73.48	73.13	75.47	76.86	70.15	78.81	63.75	**79.47 (+0.66)**
on 11	gNovel	63.21	60.39	66.62	65.23	67.09	62.12	55.07	68.13	63.89	**69.75 (+1.62)**
datasets	gHM	61.99	66.99	69.89	68.96	71.04	68.71	61.70	73.08	63.82	**74.29 (+1.21)**

Table 1. Comparison on the GZS benchmark. gNovel & gBase indicate the accuracy of the novel classes and base classes respectively under the joint classification label space. gHM is the harmonic mean of gBase and gNovel. SAP outperforms the best performing baseline on average gBase (by $+0.66\%$), gNovel (by $+1.62\%$), and gHM (by $+1.21$) computed across all datasets.

feature and a single description guided text feature. Following [4, 13], we add regularization terms designed to penalize prompted features that deviate significantly from their unprompted counterparts.

$$L^v_{steer}(\theta) = \frac{1}{B}\sum_{i=1}^{B} \|\mathbf{v}^{\mathbf{x}_i}_{pg}(\theta) - \mathbf{v}^{\mathbf{x}_i}_{g}\|_1$$

$$L^t_{steer}(\theta) = \frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}} \|\mathbf{t}^{y,A_y}_{p}(\theta) - \mathbf{t}^{y,A_y}\|_1$$

The final objective is $L_{ce}(\theta) + \lambda_1 L^v_{steer}(\theta) + \lambda_2 L^t_{steer}(\theta)$, where λ_1 and λ_2 are hyperparameters.

3. Experiments and Results

In this section, we comprehensively evaluate the generalization performance of SAP on four benchmark settings. **(i) Generalized Zero-Shot (GZS) Benchmark:** In this setting, the label space of a dataset is equally split into disjoint base and novel classes. Only a small number (16-shot) of labeled samples from the base classes are available as training data. However, during evaluation, the classification label space is the union of base and novel classes. Zero-shot classification performance depends on the label set considered, and introducing base classes into the label space tests the bias of the fine-tuned model towards them. We believe this benchmark is a more realistic measure of the generalization performance of VLM fine-tuning methods. **(ii) Base-to-Novel (B2N) Generalization:** In this setting, followed by all prior work [3, 4, 13, 14], the dataset is split into equal disjoint base and novel classes, and the model is fine-tuned on few-shot (K=16) training split of the base classes. During evaluation, the label space is constrained to the set of classes (base or novel) the test image belongs to. The testing phase for B2N is thus separate for base and novel classes, whereas the GZS benchmark has a unified testing phase.**(iii) Classification without Class Names (CwC):** VLMs requires explicit class names to perform classification. This is a limitation for images whose label lies outside the VLM's vocabulary. This benchmark tests the ability of a VLM to classify truly novel images without explicitly using the class-name. During inference, all class names are replaced with the word "object", and the model is tested on it's ability to classify an image based on descriptions alone. The model is fine-tuned on base classes, and evaluated on base and novel classes separately by replacing all class names. **(iv) Cross-Dataset Generalization:** In this setting, the model is fine-tuned on ImageNet and tested on the remaining datasets. This measures the ability of a VLM fine-tuning method to generalize to novel datasets.

Baselines. We compare our proposed approach, SAP, against state-of-the-art baselines which include very recent prompt-tuning methods – CLIP [8], CoOp [14], VPT [1], CoCoOp [15], ProDA [5], MaPLe [3], KgCoOp [13], ProGrad [16], PSRC [4] and LoGoPrompt [11]. We also compare against contemporary works that use external knowledge, such as KAPT [2], CLIP-VDT [6] and against concurrent work CoPrompt [9].

Datasets. We follow [4] to evaluate our method on 11 image classification datasets ImageNet, Caltech101, Stanford Cars, OxfordPets, Flowers102, Food101, FGVCAircraft.

Dataset		CLIP	CoOp	VPT	CoCoOp	ProDA	MaPLe	KgCoOp	ProGrad	PSRC	L.Prompt	CLIP-VDT	KAPT	SAP (Ours)
Average on 11 datasets	Base	69.34	82.69	80.81	80.47	81.56	82.28	80.73	82.48	84.26	84.47	82.48	81.10	**84.68** (+0.21)
	Novel	74.22	63.22	70.36	71.69	72.30	75.14	73.60	70.75	76.10	74.24	74.50	72.24	**77.51** (+1.41)
	HM	71.70	71.66	70.36	75.83	76.65	78.55	77.00	76.16	79.97	79.03	78.28	76.41	**80.94** (+0.97)

Table 2. Comparison on Base-to-Novel Generalization benchmark. SAP outperforms the best performing baseline on average Base (by $+0.21\%$), Novel (by $+1.41\%$) and HM (by $+0.97$) computed over all datasets.

Dataset	CoOp	CoCoOp	VPT	MaPLe	KgCoOp	ProGrad	PSRC	CLIP-VDT	KAPT	SAP (Ours)
Avg. on 10 Datasets	63.88	65.74	63.42	66.30	65.49	57.36	65.81	53.98	61.50	**66.85** (+0.55)

Table 3. Cross-Dataset Generalization benchmark. Models are trained on Imagenet and tested on the entire label space of new datasets without fine-tuning. SAP outperforms all baselines on average.

SUN397, UCF101, DTD, EuroSAT.

3.1. Empirical Results

Generalized Zero-Shot Benchmark Evaluation. We compare SAP against baselines and report the results in Tab. 1. gHM is the harmonic mean of the generalized base and novel accuracies. We outperform PSRC, achieving better results in 8 out of 11 datasets, with a **+1.21** margin in gHM averaged over all 11 datasets. We don't report the results of ProDA, LoGoPrompt and KAPT due to code unavailability.

Base-to-Novel Generalization. We compare our method with twelve baselines and report the average accuracies in Tab. 2, outperforming all baselines. In 7 out of 11 datasets, our method surpasses the state-of-the-art PSRC, with significant gains in challenging datasets like EuroSAT (**+5.66** HM) and DTD (**+2.92** HM). We also improve on UCF-101 by **+2.49** HM over PSRC. These results show that our approach effectively integrates class-agnostic knowledge from class descriptions to learn generalizable prompts.

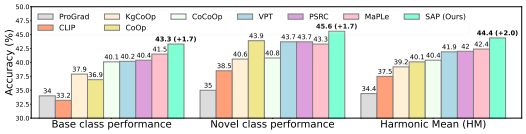

Figure 3. Comparison in the CwC setting. We show average Base, Novel, and HM accuracies over all 11 datasets . SAP outperforms baselines by average Base (by $+1.75\%$), Novel (by $+1.76\%$) and HM (by $+2.04$) computed over all datasets.

Classification without Class-names. In this benchmark, we study the ability of a pretrained VLM to classify images without explicit class-names. We report average accuracies in Fig. 3, where we beat MaPLe by **+2.04** in HM.

Cross-Dataset generalization. We compare our method with nine baselines and outperform all of them as shown in Tab. 3. SAP outperforms PSRC by **+1%** and MaPLe by **+0.5%** on average test accuracy over all datasets, which indicates that our method learns prompts that generalize across datasets.

References

[1] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge J. Belongie, Bharath Hariharan, and Ser Nam Lim. Visual prompt tuning. *ArXiv*, abs/2203.12119, 2022.

[2] Baoshuo Kan, Teng Wang, Wenpeng Lu, Xiantong Zhen, Weili Guan, and Feng Zheng. Knowledge-aware prompt tuning for generalizable vision-language models. *ICCV*, 2023.

[3] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. *2023 CVPR*, pages 19113–19122, 2022.

[4] Muhammad Uzair Khattak, Syed Talal Wasim, Muzammal Naseer, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Self-regulating prompts: Foundational model adaptation without forgetting. In *ICCV*, pages 15190–15200, 2023.

[5] Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. *CVPR*, 2022.

[6] Mayug Maniparambil, Chris Vorster, Derek Molloy, Noel Murphy, Kevin McGuinness, and Noel E. O'Connor. Enhancing clip with gpt-4: Harnessing visual descriptions as prompts. *ICCVW*, 2023.

[7] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. *2012 CVPR*, pages 3498–3505, 2012.

[8] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[9] Shuvendu Roy and Ali Etemad. Consistency-guided prompt learning for vision-language models. In *The Twelfth ICLR*, 2024.

[10] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017.

[11] Cheng Shi and Sibei Yang. Logoprompt: Synthetic text images can be good visual prompts for vision-language models. In *ICCV*, 2023.

[12] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *ArXiv*, abs/1212.0402, 2012.

[13] Hantao Yao, Rui Zhang, and Changsheng Xu. Visual-language prompt tuning with knowledge-guided context optimization. *2023 CVPR*, pages 6757–6767, 2023.

[14] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV 2021*.

[15] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. *2022 CVPR*, pages 16795–16804, 2022.

[16] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. In *ICCV 2023*.

Can Better Text Semantics in Prompt Tuning Improve VLM Generalization?

Supplementary Material

Appendix

In this appendix, we present the following details, which we could not include in the main paper due to space constraints.

- Related work is presented in § A
- Preliminaries and Background is presented in § B
- List of notations used in this paper and their descriptions are presented in § C
- Overall algorithm of SAP is presented in § D
- Additional implementation details are in § E and additional results are presented in § F
- Examples of attribute priors generated using GPT-3.5 are presented in § G

A. Related Work

Vision-Language Models. Vision-language models exhibit significant promise in acquiring generic visual representations. These models aim to harness natural language guidance for image representation learning and concurrently align both the text and image features within a shared embedding space. Conceptually, a vision-language model comprises three components: a text encoder, an image encoder, and a learning methodology that effectively utilizes information from both modalities. Recent research delves into establishing semantic connections between linguistic and visual elements, capitalizing on a vast reservoir of internet-based image-text pairs. For instance, CLIP [8] is the product of contrastive learning from 400 million carefully curated image-text pairs, while ALIGN [1] utilizes 1.8 billion noisy image-text pairs extracted from raw alt-text data. Nonetheless, a substantial challenge persists in transferring these foundational models to downstream tasks while preserving their initial capacity for generalization, which we aim to solve.

Prompt-Tuning. Prompt-tuning introduces task-specific text tokens designed to be learnable to customize the pre-trained VLM for downstream tasks. Context Optimization (CoOp) [14] marks the pioneering effort in replacing manually crafted prompts with adaptable soft prompts, fine-tuned on labeled few-shot samples. Conditional Context Optimization (CoCoOp) [15] builds upon this by generating image-specific contexts for each image and merging them with text-specific contexts for prompt-tuning. In contrast, Visual Prompt Tuning [1] introduces learnable prompts exclusively at the vision branch, resulting in sub-optimal performance for transferable downstream tasks. ProDA [5] focuses on learning the distribution of diverse prompts. KgCoOp [13] introduces regularization to reduce the discrepancy between learnable and handcrafted prompts, enhancing the generalizability of learned prompts to unseen classes. PSRC [4] shares a similar concept with KgCoOp [13] but introduces Gaussian prompt aggregation. ProGrad [16] selectively modifies prompts based on gradient alignment with general knowledge. MaPLe [3] introduces prompts at text and image encoder branches to strengthen coupling. In a distinct approach, LoGoPrompt [11] capitalizes on synthetic text images as effective visual prompts, reformulating the classification problem into a min-max formulation. All the existing methods focus on learning prompts but lack the ability to use class descriptions to capture finer contexts for better generalizability toward newer classes or datasets.

Use of External Knowledge A set of recent works [2, 3] provide evidence that visual recognition can be improved using concepts, and not just class names. However, [2] does not facilitate a way to perform fine-tuning on a downstream dataset. In contrast, [3] is a concept bottleneck model with a fixed label space and thus cannot be used for zero-shot classification. In fine-tuning methods incorporating external knowledge, KAPT [2] introduces complementary prompts to simultaneously capture category and context but lacks semantic alignment of each class description at the part-level of both image and text. On the other hand, CLIP-VDT [6] utilizes semantic-rich class descriptions only in the text modality, without semantic alignment with images. In a concurrent work CoPrompt [9], class descriptions are utilized via a regularizer acting as a consistency constraint to train the text prompts. There is no consideration of explicit semantic alignment with the image modality. Our approach utilizes class descriptions to semantically construct both text and image features, enhancing alignment between the two modalities. It helps unveil the hidden structures and promotes greater generalizability towards newer classes or datasets.

B. Preliminaries and Background

VLMs perform image classification on a downstream dataset by comparing an image representation with text representations of the class names in the dataset's label space. When a small amount of labeled data is available, it has been shown that fine-tuning VLMs substantially boosts downstream performance [14, 15]. However, the fine-tuned model does not generalize to novel classes that were absent during fine-tuning [15]. In this work, we propose **S**emantic **A**lignment for **P**rompt Learning (SAP), that leverages class descriptions to fine-tune VLMs for better generalization to novel classes. Before we describe our methodology, we

briefly discuss the required preliminaries, beginning with CLIP [8], the VLM chosen as our backbone following earlier work [3–5, 13–16].

CLIP Preliminaries. CLIP consists of an image encoder $\mathcal{I}$ and a text encoder $\mathcal{T}$, which are trained contrastively on paired image-text data to learn a common multi-modal representation space. $\mathcal{I}$ is typically a vision transformer [5] (we also show results with a ResNet [6]-based backbone) and $\mathcal{T}$ is a transformer. $\mathcal{I}$ takes an image $\mathbf{x} \in \mathbb{R}^{C\times H\times W}$ as input and returns a d-dimensional feature vector. That is, $\mathbf{x}$ is divided into M patches that are passed through the first layer of $\mathcal{I}$ to obtain patch tokens $\{\mathbf{e}_1, \ldots, \mathbf{e}_M\}$. A cls-token $\mathbf{cls}_{\mathcal{I}}$ is then added at the beginning of $\{\mathbf{e}_1, \ldots, \mathbf{e}_M\}$ to get $\{\mathbf{cls}_{\mathcal{I}}, \mathbf{e}_1, \ldots, \mathbf{e}_M\}$. These tokens are passed through the transformer layers of $\mathcal{I}$. The $\mathbf{cls}_{\mathcal{I}}$ from the last layer undergoes a final projection to get an image feature vector $\mathbf{v}^{\mathbf{x}}$.

$\mathcal{T}$ processes a text string S into a d-dimensional feature vector. The input text S is tokenized into a sequence of q word embeddings $\mathbf{w}_1, \ldots, \mathbf{w}_q$, appended with start and end tokens $\mathbf{w}_s$ and $\mathbf{w}_e$ respectively. The sequence $\mathbf{w}_s, \mathbf{w}_1, \ldots, \mathbf{w}_q, \mathbf{w}_e$ is then encoded by $\mathcal{T}$, producing the text feature vector $\mathbf{t}^S$. CLIP is trained with InfoNCE loss [8] to enhance cosine similarity for matching image-text pairs and to reduce it for non-matching pairs.

Zero-shot Classification using CLIP. CLIP performs zero-shot visual recognition of an image $\mathbf{x}$ by choosing the most similar class name from a set of candidate class names $\mathcal{Y}$, i.e., predicted class $\hat{y} = \underset{y\in\mathcal{Y}}{\arg\max}\, sim(\mathbf{v}^{\mathbf{x}}, \mathbf{t}^y)$, where the similarity measure sim is cosine-similarity. In practice, for a class name y, $\mathbf{t}^y$ is the text representation of a manually crafted prompt encapsulating y such as `'a photo of a [y]'`. Zero-shot classification performance significantly depends on the label set $\mathcal{Y}$ considered, and can also vary with the template of the text prompt [8].

Fine-Tuning CLIP with Learnable Prompts. To perform efficient adaptation under limited supervision, prompt-tuning methods add a small number of learnable tokens to the input token sequence of either modality which are fine-tuned to generate task-specific representations. For instance, CoOp [14] adds n learnable text-prompts $\mathcal{P}_t = \{\mathbf{p}_1^t, \ldots, \mathbf{p}_n^t\}$ to the input text token sequence $\{\mathbf{w}_s, \mathbf{w}_1, \ldots, \mathbf{w}_q, \mathbf{w}_e\}$. The final sequence $\{\mathbf{w}_s, \mathbf{p}_1^t, .., \mathbf{p}_n^t, \mathbf{w}_1, .., \mathbf{w}_q, \mathbf{w}_e\}$ is passed through $\mathcal{T}$ to obtain the *prompted text feature* $\mathbf{t}_p(\mathcal{P}_t)$. We follow IVLP [7], which adds learnable prompt tokens at transformer layers of both image and text encoders. That is, along with text prompts, IVLP appends learnable visual prompts $\mathcal{P}_v$ to image patch tokens, which are passed through $\mathcal{I}$ to yield the *prompted visual feature* $\mathbf{v}_p(\mathcal{P}_v)$[1]. Let $\theta = \{\mathcal{P}_t, \mathcal{P}_v\}$ denote the set of all trainable text and visual prompts. These prompts are trained to maximize the similarity between a prompted image feature and the corresponding prompted text feature of its class name. Given B image-text pairs $\{(\mathbf{x}_i, y_i)\}_{i=1}^B$, where $y_i \in \mathcal{Y}$, the likelihood of $\mathbf{x}_i$ predicting class y_i is given by: $P_\theta(y_i \mid \mathbf{x}_i) = \frac{\exp(s\cdot sim(\mathbf{v}_p^{\mathbf{x}_i}(\theta), \mathbf{t}_p^{y_i}(\theta)))}{\sum_{y\in\mathcal{Y}} \exp(s\cdot sim(\mathbf{v}_p^{\mathbf{x}_i}(\theta), \mathbf{t}_p^{y}(\theta)))}$, where s is the inverse temperature. The negative log-likelihood loss to be optimized is $L_{ce}(\theta) = \frac{-1}{B} \sum_{i\in[B]} \log(P_\theta(y_i \mid \mathbf{x}_i))$.

[1] We add a subscript p to indicate prompted features for images and text

C. Summary of Notations and Terminology

We denote vectors, matrices, and tensors using small-case bold letters. We use capital-case bold letters to denote batched data. E.g., $\mathbf{x}$ denotes a single image, and $\mathbf{X}$ denotes a batch of images. We use $\cdot$ (*dot*) to represent various types of multiplication operations – matrix multiplication, matrix-vector or vector-matrix product, and vector dot-product. Detailed descriptions of notations are presented in Tab. A1.

Notation	Description	Dimension
$\mathcal{I}$	Image Encoder	
$\mathcal{T}$	Text Encoder	
$\mathcal{Y}$	Classification label space	
θ	Set of all learnable text and visual prompts	
B	Batch size	
M	Number of grid tokens	
N	Size of the set of descriptions	
n	Number of the learnable prompt tokens	
d	Dimension of the multimodal space	
A_y	LLM generated descriptions for class y	
A	Union of all descriptions of the classification label space	
$\mathbf{t}^A$	Unprompted text features of all descriptions	$\mathbb{R}^{N\times d}$
$\mathbf{t}^{y,j}$	Unprompted description guided text view for class y and description j	$\mathbb{R}^d$
$\mathbf{t}_p^{y,j}$	Prompted description guided text view for class y and description j	$\mathbb{R}^d$
$\mathbf{v}_g$	Unprompted global image feature	$\mathbb{R}^d$
$\mathbf{v}_{pg}$	Prompted global image feature	$\mathbb{R}^d$
$\mathbf{v}_{pr}$	Prompted grid image feature	$\mathbb{R}^{M\times d}$
$\mathbf{x}_{att}$	Cross-attention map for an image computed using its grid image feature and all descriptions	$\mathbb{R}^{N\times M}$
$\hat{\mathbf{v}}_{pr}$	Prompted description guided views of a single image	$\mathbb{R}^{N\times d}$
$\mathbf{r}_{att}$	Description relevance score for an image	$\mathbb{R}^N$
$\mathbf{v}_{pl}$	Prompted local image feature	$\mathbb{R}^d$
l_s	Local score of an image	$\mathbb{R}$
$\mathbf{v}_{pu}$	Prompted unified image feature	$\mathbb{R}^d$

Table A1. Notations used in this paper and their descriptions. We denote batched features using bold capital letters. E.g., $\mathbf{V}_{pu}$ are the prompted unified image features for a batch of images.

D. Algorithm

Algorithm 1 outlines the SAP methodology. The algorithm is summarized as follows: In a given dataset, descriptions for each class are acquired by querying the LLM (L1 - L4). Class description features are then derived by passing the descriptions through $\mathcal{T}$ (L5). Unprompted and prompted image features are obtained by processing images through

$\mathcal{I}$ (L7-L8). The description-guided image views are obtained via a parameter-free cross-attention between grid features and description features (L9-L10). The local image features are a weighted average of the description-guided views based on the relevance of each description to the image (L11 - L12). Finally, the local and global image features are fused to create the unified image feature (L13 - L14). Unprompted and prompted description-guided text views are obtained by passing the description-guided text templates through T (L15-L16). L_{ce}, L^{v}_{steer}, and L^{t}_{steer} loss functions are employed to train the prompts.

Algorithm 1 SAP Algorithm

Require: Dataset D = $\{\mathbf{x}_i, y_i\}_{i=1}^{B}$; Classification label space: $\mathcal{Y}$; Vision and Language encoders: ($\mathcal{I}$, $\mathcal{T}$); LLM: ChatGPT-3.5 model; Hyperparameters: coefficients λ_1,λ_2, scaling parameter s, learning rate δ; Learnable Prompts: $\theta = \{\mathcal{P}_t, \mathcal{P}_v\}$

Ensure: Trained parameters $\hat{\theta}$

/ Get descriptions for each class by querying LLM */*

for all y $\in \mathcal{Y}$ **do**

  A_y = LLM(Visual features for distinguishing y in a photo?)

end for

$A = \bigcup_{y \in \mathcal{Y}} A_y$

$\mathbf{t}^A = \mathcal{T}(A)$ */* Get class description features */*

for all epochs **do**

  / Get unprompted and prompted image features for every image* $\mathbf{x}$ *in the batch */*

  $\mathbf{v}_g, \_ = \mathcal{I}(\mathbf{x})$

  $\mathbf{v}_{pg}, \mathbf{v}_{pr} = \mathcal{I}(\mathbf{x}; P_v)$

  / Get description-guided image views using parameter-free cross-attention */*

  $\mathbf{x}_{att} = row\_softmax(\mathbf{t}^A \cdot \mathbf{v}_{pr}^{\intercal})$

  $\hat{\mathbf{v}}_{pr} = \mathbf{x}_{att} \cdot \mathbf{v}_{pr}$

  / Get local image feature using description relevance score and image views */*

  $\mathbf{r}_{att} = softmax(\mathbf{v}_{pg} \cdot (\mathbf{t}^A)^{\intercal})$

  $\mathbf{v}_{pl} = \mathbf{r}_{att} \cdot \hat{\mathbf{v}}_{pr}$

  / Get unified image feature by fusing global and local feature using local score* l_s **/*

  $l_s = \mathbf{r}_{att} \cdot max(\mathbf{x}_{att}, dim = 1)$

  $\mathbf{v}_{pu} = (1 - l_s) \cdot \mathbf{v}_{pg} + l_s \cdot \mathbf{v}_{pl}$

  / Get unprompted and prompted description guided text views for every class* y **/*

  $\mathbf{t}^{y,A_y} = \mathcal{T}(y, A_y)$

  $\mathbf{t}_p^{y,A_y} = \mathcal{T}(y, A_y; P_t)$

  / Similarity between an image and a class is the aggregate of similarities over pertinent descriptions of a class */*

  $O_\theta(\mathbf{x}_i, y) = \frac{1}{|A_y|} \sum_{a \in A_y} sim(\mathbf{v}_{pu}^{\mathbf{x}_i}(\theta), \mathbf{t}_p^{y,a}(\theta))$

  $L_{ce}(\theta) = -\frac{1}{B} \sum_{i=1}^{B} log \frac{exp(s \cdot O_\theta(\mathbf{x}_i, y_i))}{\sum_{y \in \mathcal{Y}} exp(s \cdot O_\theta(\mathbf{x}_i, y))}$

  / Compute Steering Losses */*

  $L^{v}_{steer}(\theta) = \frac{1}{B} \sum_{i=1}^{B} \|\mathbf{v}_{pg}^{\mathbf{x}_i}(\theta) - \mathbf{v}_g^{\mathbf{x}_i}\|_1$

  $L^{t}_{steer}(\theta) = \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} \|\mathbf{t}_p^{y,A_y}(\theta) - \mathbf{t}^{y,A_y}\|_1$

  / Perform gradient descent on the total loss */*

  $\mathcal{L}(\theta) = L_{ce}(\theta) + \lambda_1 L^{v}_{steer}(\theta) + \lambda_2 L^{t}_{steer}(\theta)$

  $\hat{\theta} = \theta - \delta \nabla \mathcal{L}(\theta)$

end for

return $\hat{\theta}$

E. Implementation Details

Training Details. We use the ViT-B/16 [5]-based CLIP model as our backbone. For the GZS and B2N benchmarks, we fine-tune the model on $K = 16$ shot training data from the base classes. Prompts are learned in the first three layers for the Cross-dataset benchmark and the first nine layers for the remaining two benchmarks. We introduce a d-dimensional bias as the sole additional parameter compared to [4]. The text prompts in the initial layer are initialized with the word embeddings of `'a photo of a'`, and the rest are randomly initialized from a normal distribution, similar to [4]. Our models are trained on a single Tesla V100 GPU with Nvidia driver version 470.199.02. We train for 20 epochs, with a batch size of 4 images, $\lambda_1 = 10$ and $\lambda_2 = 25$. The hyperparameter setup is common across all datasets. We use the SGD optimizer with a momentum of 0.9, a learning rate of 0.0025, and weight decay $5e-4$. A cosine learning rate scheduler is applied with a warmup epoch of 1. Image pre-processing involves random crops, random horizontal and vertical flips, and normalization using mean values of $[0.48, 0.46, 0.41]$ and standard deviation values of $[0.27, 0.26, 0.27]$. All baselines utilize publicly available codes and models. All results are averages over three seeds. We use PyTorch 1.12, CUDA 11.3, and build on the Dassl code repository: https://github.com/KaiyangZhou/Dassl.pytorch. We will open-source our code on acceptance.

F. Expanded Tables and Additional Results

Comparison against a recent concurrent method that uses External Knowledge. In Tab. A2 we compare SAP against CoPrompt[9] on the B2N benchmark. CoPrompt[9] is a concurrent work that uses class descriptions to tune prompts and adapters, with a total of $4.74M$ additional parameters over CLIP. SAP outperforms CoPrompt by **+0.46** average HM, despite only having $36K$ additional learnable parameters. We also compare SAP against a prompt-only version of CoPrompt, as indicated by CoPrompt* in Tab. A2, in which we outperform by **+0.92** in average HM.

		CoPrompt (ICLR '24) prompts+adapter both	**CoPrompt*** (ICLR '24) prompts only	**SAP (Ours)** prompts only
Average on 11 datasets	Base	84.00	83.40	**84.68** (+1.28)
	Novel	77.23	76.90	**77.51** (+0.61)
	HM	80.48	80.02	**80.94** (+0.92)

Table A2. Comparison on the B2N benchmark against a concurrent method CoPrompt. SAP outnumbers the prompt-only version by a margin on Base (by +1.28%), Novel (by +0.61%) and HM (by +0.92).

Class Activation Maps. We show CAMs for the ResNet-50[6] backbone encoder to visualize image regions that most correlate to a given description. In addition to Fig. 1, Fig. A1 shows the GradCAM [9] visualizations for novel class *"River"* w.r.t class description "A photo of a river, is a flowing body of water which has shores" and for class *"Spring Crocus"* w.r.t description "A photo of a Spring Crocus, has vibrant purple petals which grow in clusters". SAP effectively localizes the text semantics in image compared to baselines.

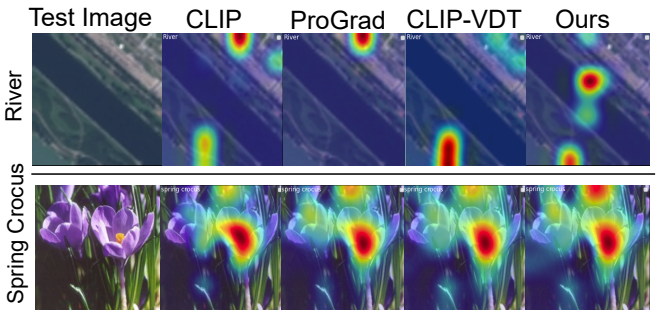

Figure A1. Class Activation Maps

Study on Leveraging Class Descriptions for Text Features. In this section we justify our design choice of averaging various class description guided semantic views to generate unified text feature. Our key contribution is not just integrating descriptions into prompt learning for VLMs, but *how* descriptions are integrated into *both* visual and text modalities. We show average HM results across all 11 datasets of other design choices herein, including CLIP-VDT[6]. We consider three alternative ways to incorporate class descriptions and show that our methodology leads to the best results. For our first alternative, we show that normalizing the unified text feature leads to a drop in performance (SAP w/ norm in Tab. A3). We also find that simply appending all class descriptions at once to generate a single semantic view also leads to a drop in performance (SAP w/ agg descriptions in Tab. A3). Finally we show that replacing our text modality construction with that used by CLIP-VDT leads to a significant drop in average HM. These experiments show that how we add class descriptions is important, and that our approach is different from recent approaches that uses external information.

Method	Avg HM
SAP	**80.94**
SAP w/ norm	80.31
SAP w/ agg descriptions	79.17
CLIP-VDT Text + SAP's Visual	78.63

Table A3. Comparison with other text design alternatives.

Effect of Removing Learnable Bias. To study the effect of adding a learnable bias to obtain grid features, we conduct an ablation study. Tab. A4 shows that adding a bias is a parameter-efficient way to learn good local image features.

Effect of Removing Class Descriptions. Our method SAP incorporates class descriptions in both image and text modalities. Here we study the effect of removing description guidance from both modalities. To remove description guidance from images, we simply use the global feature $\mathbf{v}_{pg}$ to compute L_{ce} instead of the unified feature $\mathbf{v}_{pu}$.

Method	Avg. Base	Avg. Novel	Avg. HM
Effect of Removing Learnable Bias			
SAP w/o bias	84.55	75.72	79.9
SAP	**84.68**	**77.51**	**80.94**
Effect of Removing Class descriptions			
SAP - TG	84.62	74.79	79.41
SAP - VG	84.56	77.04	80.63
SAP	**84.68**	**77.51**	**80.94**
Effect of Unified Image Features			
SAP w/ global	84.56	77.04	80.63
SAP w/ global & grid	84.66	76.81	80.55
SAP	**84.68**	**77.51**	**80.94**

Table A4. Ablation Studies. All results are on the B2N generalization benchmark, and are average results over 11 datasets.

We denote this by SAP-VG. Similarly, to remove description guidance from text, we just use the default class name template i.e. `'a photo of a [y]'`. We denote this baseline as SAP-TG. The results shown in Tab. A4 indicate that adding class descriptions to both modalities, as SAP does, is best performing. Furthermore our method to incorporate class descriptions into images is through a fully non-parametric cross-attention, and adds little computational overhead.

Effect of Unified Image Feature. Here we study the goodness of the unified image feature $\mathbf{v}_{pu}$. As described in the methodology, class descriptions are incorporated into the grid features to generate description guided image views which are averaged into local features and finally fused into unified features. We consider a baseline that uses just the global image feature $\mathbf{v}_{pg}$ instead of the unified feature. We call this SAP w/ global. Then, we consider a baseline that naively combines global and grid features (without constructing semantic-views) by averaging them and denoting it by SAP w/ global & grid. Note that both baselines construct description guided text views. The results presented in Tab. A4 justify our choice of using semantic-views of grid features to generate unified image features.

Few-shot setting. Our main objective is to train prompts that can generalize effectively to novel classes and datasets. As such, we present results primarily on settings that test generalizability such as the GZS benchmark, B2N benchmark, and the Classification without class names benchmark. For completeness, we present results in a few-shot classification setting, where limited training samples are provided for all classes. Note that there are no novel classes in this setting. We showcase outcomes for K=1, 2, 4, 8, and 16 shots. As shown in Fig A2, on average, across 11 datasets, we perform competitively against the best baseline PromptSRC.

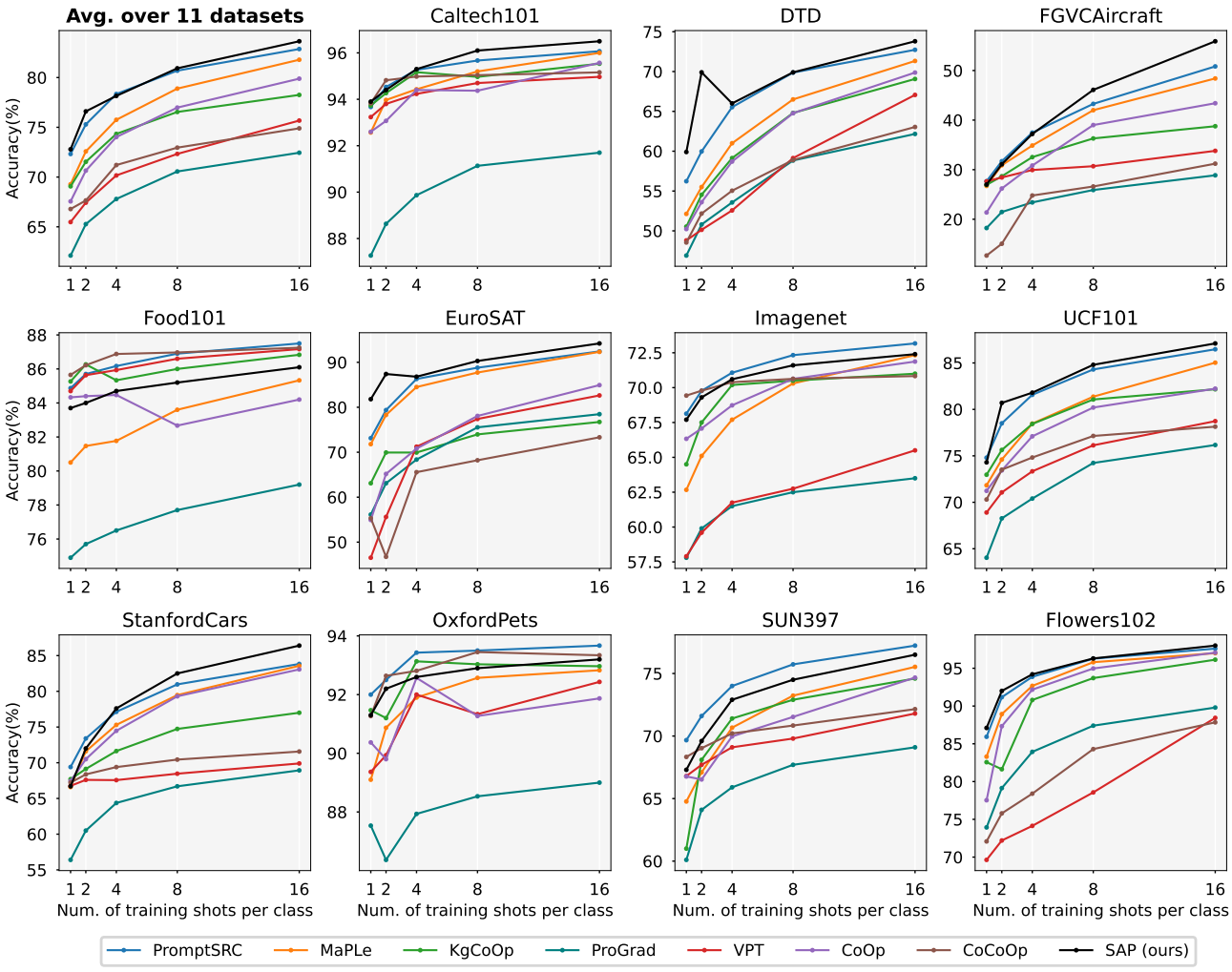

Figure A2. Performance of SAP in the few-shot setting. Our method achieves competitive performance compared to all baselines on average across 11 datasets.

Expanded Dataset-wise Tables. We present the elaborate tables dataset-wise for the Generalized Zero-Shot setting in Tab. A5 and Base-to-Novel generalization setting in Tab. A6. SAP outperforms the best-performing baseline, PSRC, in 7 of the 11 considered datasets. We perform very well in challenging datasets such as EuroSAT, DTD, and UCF-101. We present dataset-wise results for the Classification without class name benchmark in Tab. A7. Tab. A8 has the dataset-wise results for the Cross-Dataset generalization benchmark.

G. Generation of Class Descriptions

Tab. A9 shows class names sampled from different datasets and their respective descriptions retrieved using GPT-3.5 [10]. We use the query – ``What are useful visual features for distinguishing a [classname] in a photo? Answer concisely.''. Class descriptions differ from well-curated attributes found in datasets with annotated attributes such as AwA [11] and CUB [12] in three ways: (i) Our class descriptions may be noisy since no manual curation is used; (ii) They may not necessarily contain class-discriminative information, especially for similar classes; and (iii) Descriptions of a class are generated independently, and may not contain comparative traits w.r.t. other classes. These choices are primarily to keep our approach low-cost while integrating these finer details into fine-tuning of VLMs. It's important to note that our description generation occurs at the class level, not the image level, making it cost-efficient.

Dataset		CLIP (ICML '21)	CoOp (IJCV '22)	VPT (ECCV '22)	CoCoOp (CVPR '22)	MaPLe (CVPR '23)	KgCoOp (CVPR '23)	ProGrad (ICCV '23)	PSRC (ICCV '23)	CLIP-VDT (ICCVW '23)	SAP (Ours)
Average on 11 datasets	gBase	60.81	75.19	73.48	73.13	75.47	76.86	70.15	78.81	63.75	**79.47 (+0.66)**
	gNovel	63.21	60.39	66.62	65.23	67.09	62.12	55.07	68.13	63.89	**69.75 (+1.62)**
	gHM	61.99	66.99	69.89	68.96	71.04	68.71	61.70	73.08	63.82	**74.29 (+1.21)**
UCF101	gBase	62.70	80.26	75.76	76.56	76.90	78.96	74.63	**82.67**	66.19	82.23
	gNovel	64.40	**84.76**	67.73	64.76	70.40	62.33	51.36	71.40	67.00	76.40
	gHM	63.53	**82.45**	71.52	70.17	73.51	69.67	60.85	76.62	66.59	79.21
EuroSAT	gBase	51.40	69.26	88.22	70.86	84.06	82.02	76.26	86.60	55.09	**94.37**
	gNovel	38.90	36.26	53.36	41.03	43.90	31.26	23.43	54.16	50.79	**58.53**
	gHM	44.28	47.60	66.50	51.97	57.68	45.28	35.85	66.65	52.85	**72.25**
DTD	gBase	42.70	65.36	58.92	60.29	63.00	66.42	57.19	**68.73**	55.79	66.47
	gNovel	45.79	34.30	44.26	46.09	47.49	39.73	33.36	47.53	51.00	**54.27**
	gHM	44.19	44.99	50.55	52.25	54.16	49.72	42.14	56.20	53.28	**59.75**
Oxford Pets	gBase	84.80	89.56	89.06	91.12	91.69	91.99	88.36	**93.00**	83.80	91.97
	gNovel	90.19	90.46	93.23	92.50	**93.93**	92.69	87.76	91.00	90.40	92.30
	gHM	87.41	90.01	91.10	91.81	**92.80**	92.34	88.06	91.99	86.97	92.13
Stanford Cars	gBase	56.00	74.43	65.13	67.29	69.33	72.56	64.46	74.77	59.50	**76.40**
	gNovel	64.19	57.16	70.56	68.82	69.86	66.56	55.66	**71.23**	61.59	69.33
	gHM	59.81	64.67	67.74	68.05	69.61	69.43	59.74	**72.96**	60.52	72.69
Flowers102	gBase	62.09	93.40	83.12	87.36	91.19	92.80	84.86	95.00	69.90	**95.69**
	gNovel	69.80	56.92	65.56	65.53	68.29	65.76	62.39	71.00	77.00	**71.13**
	gHM	65.71	70.74	73.31	74.89	78.10	76.97	71.92	81.27	73.20	**81.60**
Food101	gBase	79.90	83.59	85.96	86.15	86.76	85.76	78.46	**87.07**	75.90	86.43
	gNovel	80.90	76.82	84.99	86.50	**87.20**	83.72	76.23	85.90	77.69	86.09
	gHM	80.39	80.07	85.49	86.33	**86.98**	84.73	77.33	86.48	76.78	86.26
FGVC Aircraft	gBase	14.50	29.92	25.12	25.90	25.90	32.69	23.93	34.90	16.10	**35.00**
	gNovel	23.79	22.83	28.03	26.36	28.53	22.06	15.63	28.40	18.60	**30.23**
	gHM	18.01	25.90	26.50	26.13	27.15	26.35	18.93	31.32	17.59	**32.44**
SUN397	gBase	60.50	72.56	69.40	71.19	72.76	73.36	67.69	**75.63**	63.09	75.40
	gNovel	63.70	56.52	67.50	67.26	68.93	61.75	57.00	68.70	66.00	**69.80**
	gHM	62.05	63.55	68.44	69.17	70.79	67.06	61.89	72.00	64.51	**72.30**
Caltech101	gBase	91.40	95.92	95.66	95.09	95.83	95.89	91.53	96.20	93.59	**96.30**
	gNovel	91.69	85.09	92.26	90.93	92.03	92.06	85.26	91.73	86.19	**92.82**
	gHM	91.54	90.19	93.94	92.97	93.89	93.94	88.29	93.91	89.73	**94.53**
Imagenet	gBase	63.00	72.80	71.9	72.59	72.80	73.00	64.19	72.30	61.79	**73.97**
	gNovel	62.00	63.20	65.40	67.80	67.40	65.40	57.70	**68.40**	56.59	66.66
	gHM	62.49	67.66	68.50	70.11	70.00	68.99	60.77	**70.30**	59.07	70.13

Table A5. Accuracy comparison on the GZS benchmark. gNovel & gBase indicate the accuracy of the novel classes and base classes respectively under the joint classification label space. gHM is the harmonic mean of gBase and gNovel. The best numbers are in bold, and the second best are underlined. As reported in the first row, SAP outperforms all baselines on average gBase (by $+0.66\%$), gNovel (by $+1.62\%$), and gHM (by 1.21%) computed across all datasets. We indicate the margin of improvement over the corresponding best-performing baseline for each metric in green.

Dataset		CLIP	CoOp	VPT	CoCoOp	ProDA	MaPLe	KgCoOp	ProGrad	PSRC	L.Prompt	CLIP-VDT	KAPT	SAP
Average on 11 datasets	Base	69.34	82.69	80.81	80.47	81.56	82.28	80.73	82.48	84.26	84.47	82.48	81.10	**84.68 (+0.21)**
	Novel	74.22	63.22	70.36	71.69	72.30	75.14	73.60	70.75	76.10	74.24	74.50	72.24	**77.51 (+1.41)**
	HM	71.70	71.66	70.36	75.83	76.65	78.55	77.00	76.16	79.97	79.03	78.28	76.41	**80.94 (+0.97)**
UCF101	Base	70.53	84.69	82.67	82.33	85.23	83.00	82.89	84.33	**87.10**	86.19	84.10	80.83	86.60
	Novel	77.50	56.05	74.54	77.64	78.04	80.77	76.67	76.94	78.80	73.07	76.40	67.10	**83.90**
	HM	73.85	67.46	78.39	77.64	78.04	80.77	79.65	79.35	82.74	79.09	80.07	73.33	**85.23**
EuroSAT	Base	56.48	92.19	93.01	87.49	83.90	94.07	85.64	90.11	92.90	93.67	88.50	84.80	**96.10**
	Novel	64.05	54.74	54.89	60.04	66.00	73.23	64.34	60.89	73.90	69.44	70.50	67.57	**81.13**
	HM	60.03	68.69	69.04	71.21	73.88	82.35	73.48	72.67	82.32	79.75	78.48	75.21	**87.98**
DTD	Base	53.24	79.44	79.15	77.01	80.67	80.36	77.55	77.35	83.37	82.87	81.80	75.97	**84.27**
	Novel	59.90	41.18	50.76	56.00	56.48	59.18	54.99	52.35	62.97	60.14	62.30	58.30	**67.03**
	HM	56.37	54.24	61.85	64.85	66.44	68.16	64.35	62.45	71.75	69.70	70.73	65.97	**74.67**
Oxford Pets	Base	91.17	93.67	94.81	95.20	95.43	95.43	94.65	95.07	95.33	**96.07**	94.40	93.13	95.27
	Novel	97.26	95.29	96.00	97.69	**97.83**	97.76	97.76	97.63	97.30	96.31	97.70	96.53	96.90
	HM	94.12	94.47	95.40	96.43	**96.62**	96.58	96.18	96.33	96.30	96.18	95.68	94.80	96.08
Stanford Cars	Base	63.37	78.12	72.46	70.49	74.70	72.94	71.76	77.68	78.27	78.36	76.80	69.47	**79.70**
	Novel	74.89	60.40	73.38	73.59	71.20	74.00	**75.04**	68.63	74.97	72.39	72.90	66.20	73.47
	HM	68.65	68.13	72.92	72.01	72.91	73.47	73.36	72.88	**76.58**	75.26	74.80	67.79	76.46
Flowers102	Base	72.08	97.60	95.39	94.87	97.70	95.92	95.00	95.54	98.07	**99.05**	97.40	95.00	97.83
	Novel	**77.80**	59.67	73.87	71.75	68.68	72.46	74.73	71.87	76.50	76.52	75.30	71.20	76.50
	HM	74.83	74.06	83.26	81.71	80.66	82.56	83.65	82.03	85.95	86.34	84.94	81.40	**86.86**
Food101	Base	90.10	88.33	89.88	90.70	90.30	90.71	90.50	90.37	90.67	**90.82**	90.40	86.13	90.40
	Novel	91.22	82.26	87.76	91.29	88.57	**92.05**	91.70	89.59	91.53	91.41	91.20	87.06	91.43
	HM	90.66	85.19	88.81	90.99	89.43	**91.38**	91.09	89.98	91.10	91.11	90.80	86.59	90.91
FGVC Aircraft	Base	27.19	40.44	33.10	33.41	36.90	37.44	36.21	40.54	42.73	**45.98**	37.80	29.67	42.93
	Novel	36.29	22.30	30.49	23.71	34.13	35.61	33.55	27.57	37.87	34.67	33.00	28.73	**38.87**
	HM	31.09	28.75	31.74	27.74	35.46	36.50	34.83	32.82	40.15	39.53	35.24	29.19	**40.80**
SUN397	Base	69.36	80.60	79.66	79.74	78.67	80.82	80.29	81.26	**82.67**	81.20	81.40	79.40	82.57
	Novel	75.35	65.89	72.68	76.86	76.93	78.70	76.53	74.17	78.47	78.12	76.80	74.33	**79.20**
	HM	72.23	72.51	79.63	78.27	77.79	79.75	78.36	77.55	80.52	79.63	79.03	76.78	**80.85**
Caltech101	Base	96.84	98.00	97.86	97.96	98.27	97.74	97.72	98.02	98.10	98.19	**98.30**	97.10	98.23
	Novel	94.00	89.91	93.76	93.81	93.23	94.36	94.39	93.89	94.03	93.78	**95.90**	93.53	94.37
	HM	95.40	93.73	95.77	95.84	95.68	96.02	96.03	95.91	96.02	95.93	**97.09**	95.28	96.26
ImageNet	Base	72.43	76.47	70.93	75.98	75.40	76.66	75.83	77.02	**77.60**	76.74	76.40	71.10	**77.60**
	Novel	68.14	67.88	65.90	70.43	70.23	70.54	69.96	66.66	70.73	**70.83**	68.30	65.20	69.83
	HM	70.22	71.92	73.66	73.10	72.72	73.47	72.78	71.46	**74.01**	73.66	72.12	68.02	73.51

Table A6. Accuracy comparison on Base-to-Novel Generalization benchmark. The best numbers are in bold, and the second best are underlined. SAP outperforms all baselines on average Base (by $+0.21\%$), Novel (by $+1.41\%$) and HM (by $+0.97\%$) computed over all datasets. We indicate the margin of improvement over the corresponding best-performing baseline for each metric in green.

Dataset		CLIP	CoOp	VPT	CoCoOp	MaPLe	KgCoOp	ProGrad	PSRC	SAP
Average on 11 datasets	Base	33.28	36.97	40.28	40.12	41.56	37.95	34.00	40.40	**43.31** (+1.75)
	Novel	38.55	43.90	43.72	40.80	43.30	40.69	35.01	43.78	**45.66** (+1.76)
	HM	35.72	40.14	41.93	40.46	42.41	39.27	34.50	42.02	**44.46** (+2.04)
UCF101	Base	56.60	61.20	61.20	61.70	64.20	62.00	59.70	63.10	**64.70**
	Novel	62.20	66.80	63.20	**70.70**	70.40	68.80	63.50	69.40	69.10
	HM	59.27	63.88	62.18	65.89	**67.16**	65.22	61.54	66.10	66.83
EuroSAT	Base	39.90	47.10	76.50	62.90	84.30	59.70	47.60	71.4	**88.70**
	Novel	71.10	78.70	**83.20**	49.00	58.30	57.60	45.80	82.10	80.90
	HM	51.12	58.93	79.71	55.09	68.93	58.63	46.68	76.38	**84.62**
DTD	Base	40.20	40.90	47.20	44.20	44.90	41.90	39.20	42.70	**52.40**
	Novel	42.40	44.10	44.30	47.10	42.90	44.40	40.20	44.00	**49.00**
	HM	41.27	42.44	45.70	45.60	43.88	43.11	39.69	43.34	**50.64**
Oxford Pets	Base	24.50	32.00	22.30	**34.20**	32.80	25.40	23.10	27.40	23.60
	Novel	35.20	40.80	40.70	44.10	**46.40**	39.70	36.00	41.60	44.10
	HM	28.89	35.87	28.81	**38.52**	38.43	30.98	28.14	33.04	30.75
Stanford Cars	Base	13.50	15.60	17.60	16.30	10.30	12.50	10.00	21.00	**22.50**
	Novel	15.90	20.70	18.90	11.70	**25.80**	15.30	8.50	20.40	23.40
	HM	14.60	17.79	18.23	13.62	14.72	13.76	9.19	20.70	**22.94**
Flowers102	Base	7.40	14.10	12.40	17.70	18.30	12.00	16.40	18.80	**19.60**
	Novel	9.30	20.40	18.40	17.60	23.20	12.30	13.80	19.30	**26.00**
	HM	8.24	16.67	14.82	17.65	20.46	12.15	14.99	19.05	**22.35**
Food101	Base	35.10	42.70	**44.00**	43.40	35.50	47.10	42.10	41.20	42.20
	Novel	33.80	**45.40**	44.80	44.40	38.90	44.60	41.80	40.50	44.20
	HM	34.44	44.01	44.40	43.89	37.12	**45.82**	41.95	40.85	43.18
FGVC Aircraft	Base	6.10	9.50	8.00	7.00	**13.40**	6.80	5.20	8.30	9.40
	Novel	7.90	**15.80**	12.80	8.30	15.50	10.70	8.20	12.30	12.30
	HM	6.88	11.87	9.85	7.59	**14.37**	8.32	6.36	9.91	10.66
SUN397	Base	46.60	49.20	50.50	51.30	50.20	50.10	40.10	50.00	**51.40**
	Novel	48.30	50.00	51.40	52.50	52.20	**53.20**	42.90	51.40	51.40
	HM	47.43	49.60	50.95	**51.89**	51.18	51.60	41.45	50.69	51.40
Caltech101	Base	77.80	76.00	**83.00**	**83.00**	82.30	80.80	72.30	81.10	81.70
	Novel	74.80	74.30	75.90	75.80	75.50	**76.20**	63.20	75.10	75.20
	HM	76.27	75.14	**79.29**	79.24	78.75	78.43	67.44	77.98	78.32
ImageNet	Base	18.40	18.40	20.40	19.70	**21.00**	19.20	18.30	19.4	20.30
	Novel	23.20	26.00	27.40	**27.60**	27.30	24.80	21.30	25.50	26.70
	HM	20.52	21.55	23.39	22.99	**23.74**	21.64	19.69	22.04	23.06

Table A7. Accuracy comparison in the classification without class names setting. We show average Base, Novel, and HM accuracies over all 11 datasets. During evaluation, descriptions of each class are provided instead of the class name, and visual recognition is conducted based on these descriptions. SAP outperforms baselines by average Base (by $+1.75\%$), Novel (by $+1.76\%$) and HM (by $+2.04\%$) computed over all datasets.

	Source	Target										
	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	Aircraft	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
VPT	70.60	91.80	90.40	63.70	67.30	83.10	22.70	66.10	46.10	37.10	65.90	63.42
MaPLe	70.72	93.53	90.49	65.57	72.23	86.20	24.74	67.01	46.49	48.06	68.69	66.30
KgCoOp	69.94	94.08	90.13	65.63	71.21	86.48	23.85	67.47	45.80	41.98	68.33	65.49
ProGrad	62.17	88.30	86.43	55.61	62.69	76.76	15.76	60.16	39.48	28.47	58.70	57.36
PSRC	71.27	93.60	90.25	65.70	70.25	86.15	23.90	67.10	46.87	45.50	68.75	65.81
CLIP-VDT	68.10	85.40	83.50	50.30	56.00	72.50	14.60	56.30	42.70	24.70	53.80	53.98
KAPT	N/A	88.90	89.40	58.15	68.00	79.95	17.95	N/A	44.80	41.35	65.05	61.50
SAP (Ours)	71.40	94.53	90.14	64.58	71.31	86.23	24.47	68.09	48.61	49.10	71.52	**66.85**

Table A8. Cross-Dataset Generalization benchmark. Models are trained on Imagenet and tested on the entire label space of new datasets without fine-tuning. SAP outperforms all baselines on average. N/A: not available in [2].

Class (Dataset)	Attributes	Class (Dataset)	Attributes
Breast stroke (UCF101)	1. Arms moving in a circular motion 2. Kicking legs in a frog-like motion 3. Head above water during stroke 4. Positioned horizontally in the water 5. Pushing water forward and outwards	Diving (UCF101)	1. Person in mid-air or jumping 2. Person wearing diving gear 3. water splashing or ripples 4. Person wearing gogglesr 5. Person wearing swim cap
Highway or road (EuroSAT)	1. Long and straight path 2. Multiple lanes for traffic 3. Traffic signs 4. Smooth and paved surface 5. Guardrails or barriers	Permanent cropland (EuroSAT)	1. Uniform vegetation or crops 2. Irrigation systems or canals 3. Organized rows or patterns 4. Fences or boundaries 5. Distinct crop types or varieties
Striped (DTD)	1. Alternating bands or lines 2. Regular pattern of stripes 3. Varying widths of stripes 4. Contrasting colors between stripes 5. Horizontal, vertical, diagonal stripes	Wrinkled (DTD)	1. Irregular and uneven surface 2. Creases or folds 3. Shadows indicating unevenness 4. Lack of smoothness 5. Distorted or crumpled appearance
Maine coon (Oxford Pets)	1. Large domestic cat 2. Long, bushy tail 3. Tufted ears with lynx-like tips 4. Rectangular body shape 5. Tufted paws	Chihuahua (Oxford Pets)	1. Small breed of dog 2. Rounded apple-shaped head 3. Erect, pointy ears 4. Short snout 5. Short legs and long tail
2008 chrysler pt cruiser convertible (Stanford Cars)	1. Convertible top 2. Chrome grille 3. PT cruiser badge 4. Alloy wheels 5. Boxy shape	2012 ferrari ff coupe (Stanford Cars)	1. Sleek and sporty design 2. Large and stylish alloy wheels 3. Low and wide stance 4. Ferrari logo on the front and rear 5. Dual exhaust pipes
Watercress (Flowers102)	1. Small, round-shaped leaves 2. Vibrant green color 3. Thin, delicate stems 4. Water or moist environments 5. Clusters of small white flowers	Trumpet creeper (Flowers102)	1. Bright orange or red flowers 2. Trumpet-shaped blossoms 3. Long, tubular petals 4. Green leaves with serrated edges 5. Hummingbirds and bees
Hot dog (Food101)	1. Cylindrical-shaped food 2. Bun or bread 3. Sausage or frankfurter 4. Visible grill marks 5. Toppings like onions or relish	Sushi (Food101)	1. Bite-sized and compact 2. Rice as a base 3. Raw or cooked fish 4. Seaweed wrapping (nori) 5. Served with soy sauce
737-200 (FGVC Aircraft)	1. Two engines on the wings 2. Low wing configuration 3. Narrow body 4. Distinctive short fuselage 5. Swept-back wings	Industrial area (SUN397)	1. Factories or warehouses 2. Smokestacks or chimneys 3. Cranes or heavy machinery 4. Conveyor belts or assembly lines 5. Trucks or shipping containers
Gramophone (Caltech101)	1. Phonograph Cylinder or Disc 2. Horn Speaker 3. Hand-Cranked Operation 4. Nostalgic and Vintage Appeal 5. Vinyl or Shellac Records	Buckle (Imagenet)	1. Metal or plastic object 2. Rectangular or circular shape 3. Fastening or securing 4. Opened and closed 5. Found on belts or straps

Table A9. Sample classes from various datasets and the corresponding attributes provided by GPT-3.5.

References

[1] Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, pages 4904–4916. PMLR, 2021.

[2] Menon, S. and Vondrick, C. Visual Classification via Description from Large Language Models. In *The Eleventh International Conference on Learning Representations*, 2023.

[3] Yang, Y., Panagopoulou, A., Zhou, S., Jin, D., Callison-Burch, C., and Yatskar, M. Language in a Bottle: Language Model Guided Concept Bottlenecks for Interpretable Image Classification. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19187–19197, 2022.

[4] Roy, S. and Etemad, A. Consistency-guided Prompt Learning for Vision-Language Models. In *The Twelfth International Conference on Learning Representations*, 2024.

[5] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*, 2021.

[6] He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. *2016 CVPR (CVPR)*, pages 770–778, 2015.

[7] Rasheed, H., Khattak, M.U., Maaz, M., Khan, S., and Khan, F.S. Fine-tuned CLIP Models are Efficient Video Learners. *2023 CVPR*, pages 6545–6554, 2022.

[8] Oord, A.v.d., Li, Y., and Vinyals, O. Representation Learning with Contrastive Predictive Coding. *ArXiv*, abs/1807.03748, 2018.

[9] Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017.

[10] Hagendorff, T., Fabi, S., and Kosinski, M. Machine intuition: Uncovering human-like intuitive decision-making in GPT-3.5. 2022. doi: 10.48550/arXiv.2212.05206.

[11] Lampert, C.H., Nickisch, H., and Harmeling, S. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR (CVPR)*, pages 951–958. IEEE, 2009.

[12] Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S.J. The Caltech-UCSD Birds-200-2011 Dataset. In .. California Institute of Technology, 2011.