

# Enhancing CLIP with GPT-4: Harnessing Visual Descriptions as Prompts

Mayug Maniparambil, Chris Vorster, Derek Molloy, Noel Murphy,  
Kevin McGuinness, Noel E. O'Connor  
ML Labs, Dublin City University,  
Dublin, Ireland

## Abstract

*Contrastive pretrained large Vision-Language Models (VLMs) like CLIP have revolutionized visual representation learning by providing good performance on downstream datasets. VLMs are 0-shot adapted to a downstream dataset by designing prompts that are relevant to the dataset. Such prompt engineering makes use of domain expertise and a validation dataset. Meanwhile, recent developments in generative pretrained models like GPT-4 mean they can be used as advanced internet search tools. They can also be manipulated to provide visual information in any structure. In this work, we show that GPT-4 can be used to generate text that is visually descriptive and how this can be used to adapt CLIP to downstream tasks. We show considerable improvements in 0-shot transfer accuracy on specialized fine-grained datasets like EuroSAT ( $\sim 7\%$ ), DTD ( $\sim 7\%$ ), SUN397 ( $\sim 4.6\%$ ), and CUB ( $\sim 3.3\%$ ) when compared to CLIP's default prompt. We also design a simple few-shot adapter that learns to choose the best possible sentences to construct generalizable classifiers that outperform the recently proposed CoCoOP by  $\sim 2\%$  on average and by over  $4\%$  on 4 specialized fine-grained datasets. The code, prompts, and auxiliary text dataset is available at [github.com/mayug/VDT-Adapter](https://github.com/mayug/VDT-Adapter).*

## 1. Introduction

Contrastive pre-training of large-scale VLMs has demonstrated remarkable image classification performance on open-set classes. Models like CLIP [25] and ALIGN [12] are pretrained on web-scale datasets consisting of image-text pairs (over 400 million and 1.8 billion respectively), resulting in a highly generalizable model with competent 0-shot domain adaptation capabilities. While vanilla supervised training is performed on a closed set of concepts or classes, CLIP pretraining uses natural language. This results in a joint text-vision embedding space that is not

constrained to a fixed set of classes. In CLIP, the classifier is constructed by plugging the class name into a predetermined prompt template like ‘a photo of {class name}’. A straightforward way to adapt CLIP to different domains is by prompt engineering, which usually involves modifying the prompt template to include semantic information about the target task. For example, to classify bird images, one could construct a prompt ‘a photo of {classname}, a type of bird’. This prompt engineering process, however, is not optimal because it: 1.) requires domain expertise in the target domain; 2.) has high variance – small changes to the prompt result in large variation in performance; 3.) has a fixed prompt template for all the classes, therefore only the class name in the prompt provides the classification anchor, which might not contain enough information to distinguish different classes. For example, in Fig 1 we see an image of a Green Heron, which from the name would suggest that it is predominantly a green-colored bird and we would assume that it is similar to Green Woodpecker if we have never seen either bird. However, we can see that it is in fact a blackish-brown bird with a chestnut-colored neck and visually more similar to a bird like the Black Bittern. For 0-shot transfer to fine-grained datasets like this to work well, CLIP has to either have seen and associated images of a Green Heron to the text ‘Green Heron’ from its large pretraining dataset or additional information in the form of *visually descriptive textual* (VDT) information is required. Here we define VDT as a set of sentences that describe the visual features of the class under consideration including shape, size, color, environment, patterns, composition, etc. While most humans can identify many different common bird species just from their names, they would need access to an ornithology taxonomy of bird descriptions to identify more rare bird species. Similarly, we argue that CLIP’s 0-shot accuracy can be improved by incorporating VDT information into the prompts. As shown, in Fig 1, including VDT information like *black crown* and *black rump* moves the classification prototype of Green Heron away from the classification prototype of Green Woodpecker and towards that of Black Bittern in the text-encoder’s embedding space.

<sup>1</sup>Corresponding author: mayugmaniparambil@gmail.com

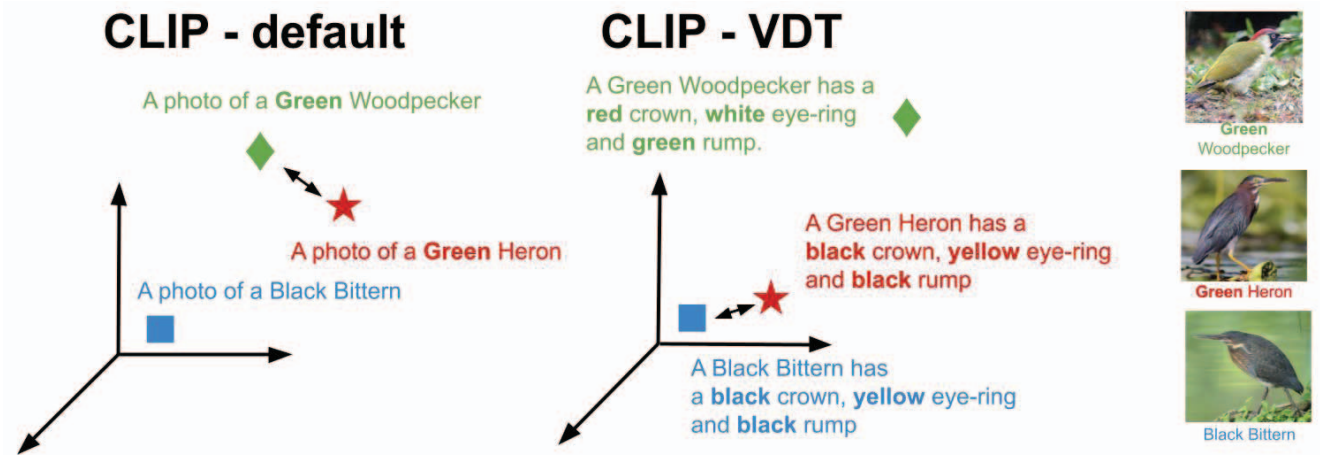


Figure 1: An example showing three birds, Green Heron, Green Woodpecker, and Black Bittern. Green Heron and Green Woodpecker have close-by classification prototypes by virtue of not having enough details in the prompt template. Here we see that adding visual descriptions to the prompt resolves this issue and moves the classification prototypes in the word-encoder’s space such that classification prototypes for visually similar birds (Green Woodpecker and Black Bittern) lie together. Only the text-encoder’s embedding space is visualized.

In this work, we first show that we can use VDT information for each class in the target domain to construct class conditional prompts that achieve performance improvements over CLIP’s default prompt. We show this on the CUB dataset [29] by constructing sentences from domain experts about the bird species in Section 3.2.1 as they are readily available as part of the dataset.

However, we acknowledge that domain expert annotations are costly and time-consuming to obtain, hampering the scalability of our method to other datasets. To address this, we focus on the recent advances in *generative pre-trained Large Language Models (LLMs)* like GPT-4 to construct these class conditional prompts in a manner easily scalable to other datasets. These models are a good fit for the task of constructing sophisticated prompts, because: 1) they are a condensed form of human knowledge (trained on web-scale text data) [33]; 2) they can be manipulated to produce information in any form or structure which makes compatibility with CLIP’s prompt style relatively simple. Therefore we use GPT-4 to construct visually descriptive textual information about the classes with special emphasis in the GPT-4 prompts about visual cues like shape, color, structure, and compositionality. We use the generated VDT information to construct prompt ensembles that are passed through CLIP’s text encoder and aggregated to generate classifiers that are then used for 0-shot classification. Using GPT-4 circumvents the need for domain knowledge and conveniently provides class conditional prompts. Prompt ensembling the VDT sentences reduce CLIP’s performance sensitivity to small changes in the prompt. We show per-

formance improvements over vanilla CLIP with the default prompt on 12 datasets with an average improvement of 2% and even better improvements in fine-grained datasets like EuroSAT ( $\sim 7\%$ ), DTD ( $\sim 7\%$ ), SUN397 ( $\sim 4.6\%$ ), and CUB ( $\sim 3.3\%$ ). The prompts and all the auxiliary class information are released publicly along with the code.

Finally, we design a simple adapter that learns to adaptively select and aggregate the best sentences for any given dataset and show that making use of this additional VDT information improves the few-shot domain transfer performance of CLIP as well. We demonstrate the few-shot adaptation performance for the recently proposed Base-to-New setting on a benchmark of 12 datasets and outperform recent methods like CoOp [36] and CoCoOp [35] despite having fewer model parameters, shorter training time, and a simpler model architecture.

In short, our contributions are as follows:

1. We show that including visually descriptive textual (VDT) information in prompts results in better 0-shot domain transfer performance of CLIP.
2. We use GPT-4 to generate VDT sentences in a scalable manner and show consistent performance improvements over CLIP in 0-shot domain transfer.
3. We design a simple adapter network to make use of this extra information for few-shot transfer and show performance improvements over methods like CLIP-Adapter and CoCoOp [35] for few-shot domain transfer in the Base-to-New setting.

4. We release all the VDT information for all 12 datasets to promote further research in multi-modal prompt and adapter design for low-shot domain transfer of large VLMs.

## 2. Related Works

### 2.1. Vision Language Models

Recent VLMs [12, 25, 8] jointly learn the vision and language encoders and have demonstrated impressive 0-shot domain transfer performance. As mentioned in [36], this can be attributed to transformer networks [28], contrastive losses [3, 10], and web-scale training datasets [25, 13].

While our GPT-generated prompt ensembles are similar to CLIP’s prompt ensembles, CLIP’s prompt ensembles were constructed and tuned manually, and are class agnostic, while ours were generated by GPT models that were prompted to provide VDT information for each class.

### 2.2. Prompt Learning

CoOp [36] successfully used prompt learning in VLMs but had generalizability limitations due to overfitting on the few-shot dataset [35]. In response, CoCoOp was proposed, enhancing performance with image-conditioned prompt learning using a meta-network, albeit at a higher resource cost. We address generalizability differently by using class conditional VDT information. Our simpler and more efficient model, CLIP-A-self, outperforms CoCoOp in the Base-to-New few-shot setting.

### 2.3. Few-shot adapters for Vision Language models

CLIP-Adapter [9] (CLIP-A) offers a simpler few-shot transfer method for VLMs, utilizing an MLP trained on fixed image/text encoders. Our CLIP-A-self is different from CLIP-A in that we apply a self-attention mechanism on the set of all sentences for any class, learning to select and aggregate the best subset of VDT information for the dataset from the few-shot training set. Although Tip-adapter [34] showed superior performance on base classes with a cache model, it’s inapplicable in the Base-to-New setting due to its reliance on few-shot test class examples, making it irrelevant for our comparison.

### 2.4. Semantic information from Large Language Models

Recent advancements in transformer-based language models, particularly the GPT family [2, 22], have demonstrated exceptional abilities in semantic extraction from intricate texts. Their application to vision tasks has emerged as an active area of research. [20] employs Palm540B LLM [4] to generate semantic data for unsupervised class embedding vectors in 0-shot classification, but only tests on three

legacy datasets. Our research presents results on a modern benchmark of 12 datasets. Recently, [24, 19] leverage GPT-3 for class conditional prompts to enhance CLIP’s 0-shot domain transfer on 6 datasets. While [19] focuses on using GPT-3 to construct visual descriptors that aid in the interpretability of CLIP’s predictions during 0-shot domain transfer, we argue that 0-shot domain transfer performance improves with the inclusion of high-quality VDT information. Hence, we make use of GPT-4 for richer, more diverse, and more accurate VDT information. Concurrent work [16] leverages GPT-3 to generate image-specific fine-grained text by matching GPT-generated text candidates in CLIP’s embedding space and uses them for improving the downstream classification.

While [19] utilize GPT-3, probability ensemble, and highlight VDT’s role in 0-shot transfer, our method differs. We use GPT-4 for auxiliary data collection, perform ensemble in word-encoder space, and introduce a few-shot adapter for optimal VDT selection in few-shot transfer. [27] uses GPT-3 for prompt construction in diffusion models to generate images for support sets while our work only uses GPT4 to acquire auxiliary text data. To our knowledge, we are the first to prompt GPT-4 for visually descriptive sentences to improve CLIP’s 0-shot and few-shot transfer.

## 3. Methodology

### 3.1. Review of CLIP and CLIP-Adapter

Through contrastive pretraining on large image-text datasets, CLIP performs classification on various concepts, aligning related images and texts in a shared embedding space, while separating dissimilar ones. After pretraining, CLIP directly performs image classification on the target dataset without any finetuning. First, we review how the CLIP model performs 0-shot classification on an open set.

The CLIP model, comprising a vision and language model, encodes an image and its corresponding caption into visual and textual embeddings, respectively. During inference, these embeddings are compared using *cosine similarity*. Given an image  $I \in \mathbb{R}^{H \times W \times C}$ , where  $H$ ,  $W$ ,  $C$  denotes the height, width, and number of channels of the image, the vision encoder transforms the image into the joint embedding space to get the image features  $f \in \mathbb{R}^D$  where  $D$  represents the dimension of the features.

During inference, a prompt template such as ‘A photo of {classname}’ is used to generate sentences for  $K$  different classes and passed through the text-encoder to yield classifier weight matrix  $W \in \mathbb{R}^{D \times K}$ . Prediction probabilities are then calculated by multiplying image feature  $f$  with  $W$  and applying a softmax function:

$$f = \text{Backbone}(\mathbf{I}), \quad p_i = \frac{\exp(\mathbf{W}_i^T f) / \tau}{\sum_{j=1}^K \exp(\mathbf{W}_j^T f) / \tau}, \quad (1)$$

In CLIP [25], 0-shot domain transfer utilizes domain-specific information in the prompt template, such as ‘A photo of a {class-name}, a type of bird’ for bird images. [25] reports that careful prompt design and prompt ensembling are important to improve 0-shot classification accuracy. Prompt ensembling is achieved by constructing several prompts for each class and then averaging the classification vectors. In our work, we show that prompt ensembles of VDT improve CLIP’s 0-shot domain transfer.

CLIP-A [9] is a learnable MLP adapter applied to image and/or word encoder features for few-shot transfer to target datasets. During few-shot transfer, given  $N$  images per class with labels, denoted as  $(x_{i,k}, y_{i,k})_{i=1, k=1}^{i=N, j=K}$ ,  $K$  classifier weights are constructed using the prompt template  $H$  and text encoder  $g$  as  $W = g(H(classname(\{y_{i,k}\})))$ . The image features  $f$  and text features  $W$  pass through the learnable adapters  $A_v, A_t$  to get adapted features as follows.

$$f^* = \alpha A_v(f)^T + (1 - \alpha)f, \quad (2)$$

$$W^* = \beta A_t(W)^T + (1 - \beta)W. \quad (3)$$

The hyperparameters  $\alpha$  and  $\beta$  blend CLIP’s knowledge with fine-tuned knowledge to avoid CLIP-Adapter overfitting. Logits are calculated as per Eqn 1, and cross entropy loss over the entire training set  $(x_{i,k}, y_{i,k})_{i=1, k=1}^{i=N, j=K}$  is used to optimize  $A_v, A_t$ .

In the *All* setting, few-shot transfer is tested on a hold-out dataset with images from the  $K$  classes used in training. In the *Base-to-New* setting, proposed by [35], the evaluation occurs on  $U$  non-overlapping classes. Our model is evaluated in the more practical *Base-to-New* setting.

### 3.2. Language Model Prompt Design

In this section, we demonstrate how VDT enhances CLIP’s 0-shot transfer capabilities and outline our method for generating class-specific prompts with an LLM.

#### 3.2.1 Visual Descriptive Sentences

[25] demonstrates that careful prompt design and prompt ensembling improve the 0-shot classification performance of CLIP. Here we ask the question: What type of information can be appended to the prompt template to improve the 0-shot domain transfer performance? We show that appending visually descriptive information to the prompt template and ensembling improves the 0-shot performance over the default prompt and prompts containing non-visual information.

Using the CUB dataset with expert annotations, we contrast the 0-shot performance of visual and non-visual prompt ensembles. For the visual prompts, we take class attribute vectors detailing attributes like color, pattern, shape, etc. for 28 bird body parts, leading to 312 scores per bird.

Table 1: Comparing visual and non-visual prompt ensembles for 0-shot domain transfer to the CUB dataset.

Prompting	Default	Non-Visual-GT	Visual-GT	Visual-GPT
Accuracy	54.7	53.0	57.7	57.4

Table 2: Results of including LLM generated VDT on 6 datasets for comparison with other works. We see that higher quality VDT from GPT-4 outperforms GPT-3 generated VDT on specialized datasets like DTD OxfordPets and EuroSAT.

Methods	EuroSAT	Food101	DTD	Oxford Pets	CUB	ImageNet	Average
CLIP	47.69	85.97	43.09	89.07	54.70	64.51	64.17
DCLIP[19]	48.82	<b>88.50</b>	45.59	86.92	<b>57.75</b>	68.03	65.93
CLIP-GPT	<b>54.86</b>	86.43	<b>50.15</b>	<b>91.54</b>	57.43	<b>68.92</b>	<b>68.21</b>

We use the most pronounced attribute-value pairs to form 28 visual prompts (denoted *Visual-GT*) such as ‘A photo of Green Heron. Green Heron has a greenish-black head cap.’ Conversely, for non-visual prompts (denoted *Non-Visual-GT*), we collect information on bird calls, migration, behavior, and habitat, yielding 12 different prompts like ‘A photo of Green Heron. The green heron’s bird call is a loud, harsh ‘skeow’ per class.

We derive classification vectors for *Visual-GT* and *Non-Visual-GT* by averaging class-level sentence embeddings within CLIP’s joint embedding space, considering its 77-token limit. Table 1 shows no improvement using *Non-Visual-GT* prompts over the default, yet a 4% improvement with *Visual-GT*.

#### 3.2.2 Prompting LLMs for visually descriptive information

In the prior section, we highlighted the use of expert VDT information in creating class-specific prompts to enhance CLIP’s 0-shot performance. However, acquiring expert annotations is both expensive and time-consuming. To overcome this, we utilize GPT language models, known for their large-scale knowledge and flexibility [33]. Our approach involves using GPT-4 to generate visual descriptions for any given dataset thereby aiding in the construction of prompt ensembles for CLIP in a scalable manner.

Our prompting strategy takes inspiration from chain-of-thought prompting [30] and is as follows: First, we ask GPT-4 to list all the attributes that may be necessary to discriminate between images of the  $K$  classes under consideration. Second, we ask GPT-4 to provide the values for all these attributes for all the  $K$  classes as sentences. Example for the CUB dataset is shown on the left side of Fig 1.

The last row in Table 1 shows that the GPT-4 generated



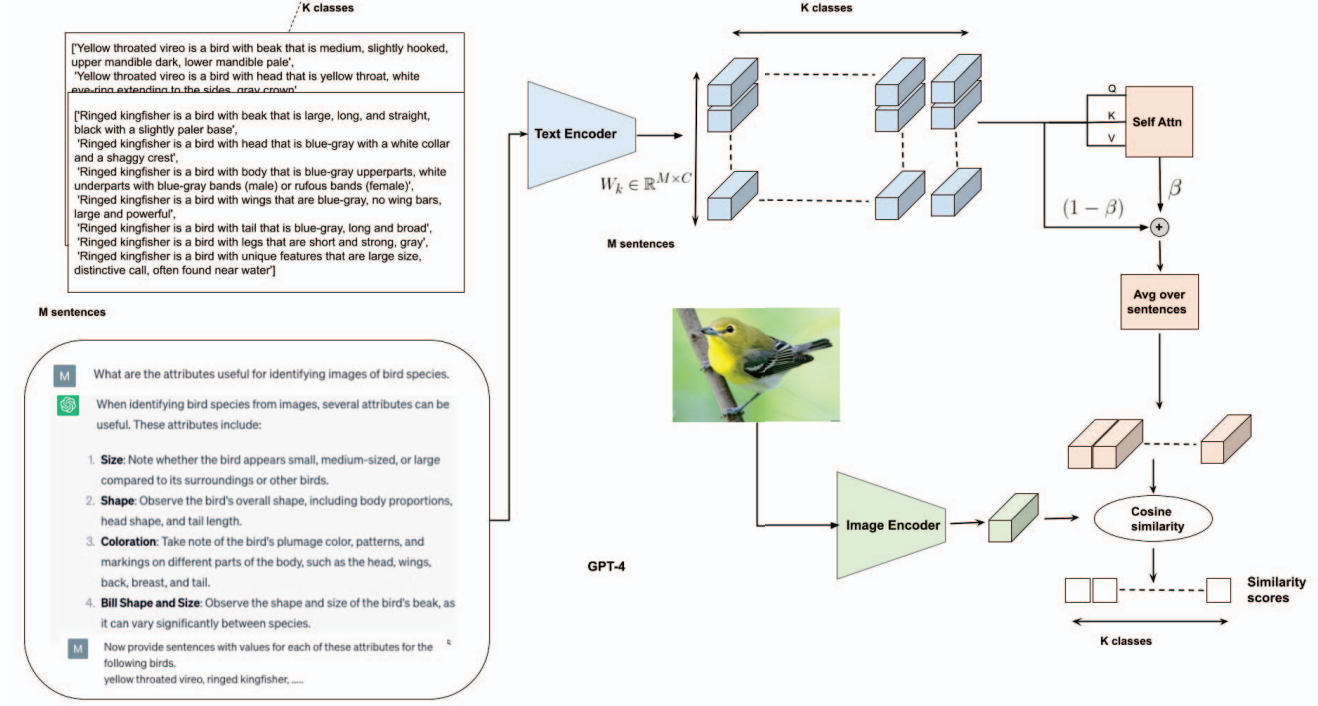


Figure 2: CLIP-A-self, our simple self-attention based adapter learns to select and aggregate the most relevant subset of Visually Descriptive Text (VDT) to generate more generalizable classifiers. First, we prompt GPT-4 to generate VDT,  $N$  sentences for  $K$  classes that are then passed through the text encoder to get embeddings for each of the  $N \times K$  sentences. Self-attention is applied over the  $N$  sentences of each class and averaged to get  $K$  adapted classifier embeddings.

Table 3: Results of 0-shot transfer on 12 datasets with ViT-B/16.

Methods	EuroSAT	Caltech101	Oxford Flowers	Food101	FGVC Aircraft	DTD	Oxford Pets	Stanford Cars	Sun397	UCF101	CUB	ImageNet	Average
CLIP	47.69	93.75	70.69	85.97	<b>24.81</b>	43.09	89.07	<b>65.55</b>	62.61	<b>67.54</b>	54.70	64.51	64.16
CLIP-GPT	<b>54.86</b>	<b>94.51</b>	<b>73.40</b>	<b>86.43</b>	23.42	<b>50.15</b>	<b>91.54</b>	65.01	<b>67.24</b>	65.51	<b>57.43</b>	<b>68.9</b>	<b>66.53</b>

visual sentences' performance is similar to that of sentences generated from the class attribute vectors annotated by domain experts. We follow the same simple strategy for all the datasets in the benchmark suite to generate visually descriptive sentences in a scalable and flexible manner and use them to construct prompt ensembles.

### 3.3. Simple few-shot adapters for visual sentences

We design a simple adapter that can use VDT information to improve the few-shot transfer of CLIP to the target datasets. Similar to the CLIP-A text, we append a small set of learnable parameters to the output of the word encoder and train the adapter using cross-entropy loss. Our CLIP-A-self uses a self-attention layer that applies attention over the embeddings of the different sentences for each class and averages the output to get the final classification vector.

	Base	New	H
CLIP	68.45	73.89	71.05
CoOp	<b>82.39</b>	62.39	70.99
CoCoOp	79.35	71.89	75.37
CLIP-A	78.90	72.14	75.07
CLIP-A-self	82.12	74.20	<b>77.78</b>

Table 4: Comparing our CLIP-A-self against other methods on average accuracy over 12 datasets.

Given we have  $M$  GPT generated sentences for each of the  $K$  classes  $t_{m,k}$ , we construct  $M$  prompts by appending each sentence to the prompt template like  $H(\text{classname}(y_{i,k}), \{t_{m,k}\})$  and pass them through CLIP's word encoder to get  $W^{sent} \in \mathbb{R}^{D \times M \times K}$ .

For the self-attention adapter, we apply vanilla self-attention [28] over all the visual descriptive sentences such that during training it learns to select and aggregate the most relevant visual sentences for identifying each class. Just like before, we first obtain the classification vector for all sentences  $W^s \in \mathbb{R}^{K \times M \times D}$  and pass them as the key, query, and value to the self-attention module  $B_{self}$  and average out the output tokens to get the final classification vector  $W^*$ . Here the attention is applied over the  $M$  different visually descriptive sentences.

$$W_{avg} = 1/M \sum_{m=1}^M W_{m,k}^s \quad (4)$$

$$\{W_{m,k}^a\}_1^M = B_{self}(\{W_{m,k}^s\}_1^M, \{W_{m,k}^s\}_1^M, \{W_{m,k}^s\}_1^M) \quad (5)$$

$$W_{a-mean} = 1/M \sum_{m=1}^M W_{m,k}^a \quad (6)$$

$$W^* = \beta W_{a-mean}^T + (1 - \beta) W_{avg} \quad (7)$$

We finally obtain the new adapter classifier weights  $W^* \in \mathbb{R}^{D \times K}$  that have been adapted to focus on the most visually discriminative information among the  $M$  visually descriptive sentences for any given dataset. We make use of 1 to calculate the probabilities and predict the image category by selecting the class with the highest probability.

During the few-shot training only the weights of the adapter network  $B_{self}$  are trained using cross-entropy loss.

## 4. Experiments

We assess the significance of visual sentence ensembles in two scenarios: (i) we gauge visual sentence quality by comparing an ensemble of these prompts with CLIP’s default prompts across 12 benchmark datasets; (ii) we contrast the performance of adapters using these visual prompts against other few-shot transfer techniques in Base-to-New class generalization within a dataset. Prior to discussing the results, we detail the datasets and experimental setup.

### 4.1. Datasets

We use 11 diverse image recognition datasets from [36] and the bird species CUB dataset [29] for both study settings, extending our suite to 12. These include generic object datasets ImageNet [6] and Caltech101 [7]; fine-grained classification datasets OxfordPets [23], StanfordCars [15], Flowers102 [21], Food101 [1] and FGVCAircraft [17]; SUN397 [32] for scene recognition; UCF101 [26] for action recognition; DTD [5] for texture classification; EuroSAT [11] for satellite imagery; and CUB for bird identification.

For 0-shot transfer with visual sentences, we test on *All* classes across these datasets while for the Base-to-New set-

ting, following [35], we equally sample classes for base and new sets without overlap. We use the 150-base and 50-new class split from ZSL and few-shot literature [31, 18] for CUB. Like [35], our CLIP-A-self is evaluated on the 16-shot setting for easier comparison with other methods.

### 4.2. Baselines

We compare the performance of visual sentences ensemble on 0-shot transfer against the CLIP model [25] whose default prompts for each dataset have been extensively fine-tuned using a test set. We also compare against DCLIP [19] a recent work that uses GPT-3 to generate VDT information for 0-shot transfer. We compare our CLIP-A-self against two prompt learning methods CoOp [36] which learns static prompts and CoCoOp [35] which learns a dynamic prompt that is specifically designed to improve Base-to-New transfer. We also compare our CLIP-A-self against CLIP-A [9] due to the similarity in architecture and to show that the performance improvements are from making use of the visual sentences and not from the just adapting the text features.

### 4.3. Training settings

Our implementation is based on CoOp’s and CLIP-A’s code. We make all our comparisons on VIT CLIP backbone i.e., VIT-B/16. We take the results for CoOp and CoCoOp for all datasets (except CUB) from their respective papers, while we make use of practices from the respective papers like context length set to 4 and context initialization to “a photo of” to ensure the best results on the CUB dataset. For CLIP-A, we re-run all experiments on VIT-B/16 backbone as they were not reported in the paper. For all adapter models including ours, we only tune the residual ratio  $\beta$  hyperparameter. For CLIP-A, we use the version where the MLP is applied on top of the visual encoder as it performed the best [9]. We make use of May version of GPT-4 for obtaining the auxiliary dataset.

### 4.4. GPT generated visual sentences improve 0-shot transfer.

We compare the performance of CLIP-GPT prompt ensemble with the default prompts of CLIP in Table 3. GPT-generated prompt ensemble improves upon the performance of CLIP 0-shot by 2% on average over 12 datasets. The improvement over CLIP-ZS is significant; over 5% for specialized fine-grained datasets like CUB, SUN397, EuroSAT, and DTD and over 2% for oxford-flowers and oxford-pets. This shows that CLIP does not recognize several of the classnames in these datasets and describing the class in the form of visually descriptive sentences results in better classifiers from the text-encoder and better classification accuracy. It is also worth noting that only including the visually descriptive sentences in the prompts can help improve the performance of general datasets like Imagenet (over 4%)

Table 5: **Comparison of CLIP-A-self with CLIP, CoOp and CoCoOp in the Base-to-New generalization setting.** For prompt learning-based methods (CoOp and CoCoOp), their prompts are learned from the base classes (16 shots). The results strongly justify the importance of including extra VDT information. H denotes Harmonic mean (to highlight the generalization trade-off [31]).

(a) CUB.				(b) Caltech101.				(c) OxfordPets.			
	Base	New	H		Base	New	H		Base	New	H
CLIP	58.7	70.3	63.90	CLIP	96.84	94.00	95.40	CLIP	91.17	97.26	94.12
CoOp	79.2	53.3	63.71	CoOp	98.00	89.81	93.73	CoOp	93.67	95.29	94.47
CoCoOp	67.1	74.1	70.40	CoCoOp	97.96	93.81	95.84	CoCoOp	<b>95.20</b>	<b>97.69</b>	<b>96.43</b>
CLIP-A	68.3	70.8	69.53	CLIP-A	97.7	93.6	95.61	CLIP-A	94.8	97.0	95.89
CLIP-A-self	<b>78.6</b>	<b>71.3</b>	<b>74.77</b>	CLIP-A-self	<b>98.3</b>	<b>95.9</b>	<b>97.09</b>	CLIP-A-self	94.4	97.0	95.68
(d) StanfordCars.				(e) Flowers102.				(f) Food101.			
	Base	New	H		Base	New	H		Base	New	H
CLIP	63.37	<b>74.89</b>	68.65	CLIP	72.08	<b>77.80</b>	74.83	CLIP	90.10	91.22	90.66
CoOp	<b>78.12</b>	60.40	68.13	CoOp	<b>97.60</b>	59.67	74.06	CoOp	88.33	82.26	85.19
CoCoOp	70.49	73.59	72.01	CoCoOp	94.87	71.75	81.71	CoCoOp	<b>90.70</b>	<b>91.29</b>	<b>90.99</b>
CLIP-A	70.5	73.3	71.87	CLIP-A	94.6	71.5	81.44	CLIP-A	90.3	91.2	90.75
CLIP-A-self	76.8	72.9	<b>74.80</b>	CLIP-A-self	97.4	75.3	<b>84.94</b>	CLIP-A-self	90.4	91.2	90.80
(g) FGVCAircraft.				(h) SUN397.				(i) DTD.			
	Base	New	H		Base	New	H		Base	New	H
CLIP	27.19	<b>36.29</b>	31.09	CLIP	69.36	75.35	72.23	CLIP	53.24	59.90	56.37
CoOp	<b>40.44</b>	22.30	28.75	CoOp	<b>80.60</b>	65.89	72.51	CoOp	79.44	41.18	54.24
CoCoOp	33.41	23.71	27.74	CoCoOp	79.74	76.86	78.27	CoCoOp	77.01	56.00	64.85
CLIP-A	34.9	33.5	34.19	CLIP-A	80.1	75.9	77.94	CLIP-A	74.9	53.0	62.08
CLIP-A-self	37.8	33.0	<b>35.24</b>	CLIP-A-self	81.4	76.8	79.03	CLIP-A-self	<b>81.8</b>	<b>62.3</b>	<b>70.73</b>
(j) EuroSAT.				(k) UCF101.				(l) ImageNet.			
	Base	New	H		Base	New	H		Base	New	H
CLIP	56.48	64.05	60.03	CLIP	70.53	<b>77.50</b>	73.85	CLIP	72.43	68.14	70.22
CoOp	<b>92.19</b>	54.74	68.69	CoOp	<b>84.69</b>	56.05	67.46	CoOp	<b>76.47</b>	67.88	71.92
CoCoOp	87.49	60.04	71.21	CoCoOp	82.33	73.45	77.64	CoCoOp	75.98	<b>70.43</b>	<b>73.10</b>
CLIP-A	82.5	62.4	71.06	CLIP-A	82.9	74.9	78.70	CLIP-A	75.4	68.6	71.84
CLIP-A-self	88.5	70.5	78.48	CLIP-A-self	84.1	76.4	<b>80.07</b>	CLIP-A-self	76.4	68.3	72.12

and Caltech-101 (over 1%) too. For all other datasets, the transfer performance matches that of CLIP, with the exception being the action recognition dataset UCF-101. We inspected the sentences generated for UCF-101 and notice that several of the sentences generated by GPT involves temporal information instead of visual descriptions and we believe this could be the reason for the drop in accuracy. However, we notice in Section 4.5.1 that the self-attention module of the few-shot adapter learns to emphasize the visual sentences out of the generated sentences which might explain the improvement in the performance of few-shot adapters in the new setting in Section 4.5. We also compare against recent work [19] on their subset of 6 datasets

for ViT-B/16 encoder in 2. We see that using the larger GPT-4 model over the GPT-3 model results in much higher improvements for specialized datasets like DTD ( $\sim 5\%$ ) and EuroSAT ( $\sim 6\%$ ). We compare the text used by [19] against our GPT4-generated VDT in the supplementary.

#### 4.5. GPT-Adapters improve few-shot transfer performance.

We compare the performance of our CLIP-A-self against CLIP, CoOp, and CoCoOp on the benchmark suite of 12 datasets in the Base-to-New setting in Table 5. Here we see that GPT-Adapters that make use of the VDT information outperform CoCoOp by 3% in the new setting while

Prompting	ZS	Base	New	H
Default	54.7	NA	NA	NA
OpenAssistant	56.0	78.3	69.8	73.80
GPT-3.5	55.7	78.1	70.6	74.16
GPT-4	57.4	78.6	71.3	74.77

Table 6: Comparing different GPT models for obtaining the VDT information. We see that the larger models provide higher quality VDT information but CLIP-A-self is capable of producing generalizable classifiers even with smaller models like OpenAssistant.

maintaining similar performance to that of CoOp in the base setting on the average accuracy over 12 datasets. This is impressive considering that CoCoOp makes use of a meta-network and forward pass through the text encoder making it computationally intensive to train. CoCoOp takes up to 5 hours to train on 16-shot ImageNet for ViT-B/16 encoder, in comparison, our CLIP-A-self takes only 10 mins (on an RTX 3090 GPU). The Base-to-New generalization ability of our adapters is even more impressive for fine-grained, specialized datasets as evidenced by the gains over CoCoOp in Harmonic mean of base and new accuracy. For example, CLIP-A-self demonstrates gains in datasets like FGVC-Aircraft ( 7.5%), EuroSat ( 7.4%), DTD ( 5.8%), CUB ( 4.3%), Flowers102 ( 4%), Stanford Cars ( 2.4%) and UCF-101 ( 2.4% ). This demonstrates that our adapters make use of semantic information in the form of visually descriptive sentences and fuse this with CLIP’s 0-shot knowledge to build more generalizable classifiers that transfer well to unseen classes within the same dataset. It is also worth noting that even though the same set of VDT did not provide any improvements in 0-shot domain transfer for datasets like FGVC-Aircraft, Stanford-Cars, and UCF-101, our self-attention adapter was able to choose the most informative subset of VDT and produce few-shot classifiers that provide substantial few-shot transfer performance gains in comparison to CoCoOp. We show in Section 4.5.1 the sentences picked by the attention mechanism for these datasets to qualitatively verify this.

#### 4.5.1 Attention weights Analysis

We note that even though CLIP-GPT ensembles were outperformed by CLIP default prompt on FGVC Aircraft, UCF-101, and Stanford Cars dataset, we see that CLIP-A-self outperforms CLIP-A and CoCoOp [35] on these datasets in the few-shot transfer setting. We believe that this is because, during few-shot training, the self-attention mechanism learns to select the most relevant visual sentences out of the set of visually descriptive text and helps produce generalizable classifiers. In Table 1 in supplement-

tary, we show the top 3 and bottom 3 attributes picked by attention scores for each of these datasets and show that the sentences with the highest attention scores correspond to visually descriptive attributes in the set and vice versa for the lowest scored attributes. For example, for both Stanford Cars and FGVC it is interesting to see that the color scheme is one of the least used attributes as it’s difficult to identify a car or a plane from its color or livery. For UCF-101, information like the force involved or temporal information like speed and range of motion of the action is unlikely to be encoded in the image and hence is not selected by the attention mechanism. Information regarding the subject and the object of the action, like the posture of the person, description of the object, and interaction between objects are visible in the images and hence weighted highly by the attention mechanism.

#### 4.6. Ablation over different GPT models

In this section, we see if other GPT models like GPT-3.5 and the open-source model, OpenAssistant [14], are as capable as GPT-4 in generating visually descriptive information. We explore this on the CUB dataset as it is fine-grained and specialized. The results are presented in Table 6. We find that the performance improves with larger models which are more capable of memorizing accurate class information with less hallucination [33]. Even though we obtain decent performance with the open-source model OpenAssistant, the outputs were always inconsistent and noisy, resulting in a lot of clean-up effort in comparison to GPT-3.5 and GPT-4 where the outputs were in the form of concise sentences following a dictionary format. It is worth noting that our few-shot adapter is capable of picking out the best VDT information even from a noisy set, pushing the Base-to-New generalization performance of OpenAssistant, and GPT-3.5 close to that of GPT-4.

### 5. Conclusion

In this work, we show that using visually descriptive textual (VDT) information can improve the 0-shot domain transfer performance of CLIP over non-visual information and the default prompts. We demonstrate GPT-4 to be an accurate and flexible source of VDT information by improving the 0-shot domain transfer performances on a suite of 12 benchmark datasets. Our few-shot adapter CLIP-A-self learns to pick the best VDT information from the GPT generated set and improve the few-shot domain transfer in the Base-to-New setting even when the quality of the generated text deteriorates. We release all prompts and VDT information for all 12 datasets to promote further research in the fertile research direction of using LLMs for learning multimodal adapters for foundation models.



## Acknowledgement

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant number 18/CRT/6183. For the purpose of Open Access, the author has applied a CC BY public copyright license to any Author Accepted Manuscript version arising from this submission.

## References

- [1] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101—mining discriminative components with random forests. In *ECCV*, 2014. 6
- [2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020. 3
- [3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 3
- [4] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayanan Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways, 2022. 3
- [5] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 6
- [6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 6
- [7] 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 *CVPR-W*, 2004. 6
- [8] Andreas Fürst, Elisabeth Rumetshofer, Johannes Lehner, Viet T Tran, Fei Tang, Hubert Ramsauer, David Kreil, Michael Kopp, Günter Klambauer, Angela Bitto, et al. Cloob: Modern hopfield networks with infoloob outperform clip. *Advances in neural information processing systems*, 35:20450–20468, 2022. 3
- [9] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021. 3, 4, 6
- [10] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 3
- [11] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019. 6
- [12] 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, 3
- [13] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 3
- [14] Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Richárd Nagyfi, et al. Openassistant conversations—democratizing large language model alignment. *arXiv preprint arXiv:2304.07327*, 2023. 8
- [15] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV-W*, 2013. 6
- [16] Kathleen M Lewis\*, Emily Mu\*, Adrian V Dalca, and John Gutttag. Gist: Generating image-specific text for fine-grained object classification. *arXiv preprint arXiv:2307.11315*, 2023. 3
- [17] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 6
- [18] Mayug Maniparambil, Kevin McGuinness, and Noel E. O’Connor. Basetransformers: Attention over base datapoints for one shot learning. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, page 482. BMVA Press, 2022. 6
- [19] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. *ICLR*, 2023. 3, 4, 6, 7

- [20] Muhammad Ferjad Naeem, Muhammad Gul Zain Ali Khan, Yongqin Xian, Muhammad Zeshan Afzal, Didier Stricker, Luc Van Gool, and Federico Tombari. I2mvformer: Large language model generated multi-view document supervision for zero-shot image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15169–15179, 2023. 3
- [21] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008. 6
- [22] OpenAI. Gpt-4 technical report, 2023. 3
- [23] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *CVPR*, 2012. 6
- [24] Sarah Pratt, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. *arXiv preprint arXiv:2209.03320*, 2022. 3
- [25] 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, 3, 4, 6
- [26] 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. 6
- [27] Vishaal Udandaraao, Ankush Gupta, and Samuel Albanie. Sus-x: Training-free name-only transfer of vision-language models. *arXiv preprint arXiv:2211.16198*, 2022. 3
- [28] 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. 3, 6
- [29] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. Caltech-UCSD Birds 200, 2011. 2, 6
- [30] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022. 4
- [31] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2251–2265, 2018. 6, 7
- [32] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010. 6
- [33] Jifan Yu, Xiaozhi Wang, Shangqing Tu, Shulin Cao, Daniel Zhang-Li, Xin Lv, Hao Peng, Zijun Yao, Xiaohan Zhang, Hanming Li, et al. Kola: Carefully benchmarking world knowledge of large language models. *arXiv preprint arXiv:2306.09296*, 2023. 2, 4, 8
- [34] Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of clip for few-shot classification. In *European Conference on Computer Vision*, pages 493–510. Springer, 2022. 3
- [35] Kaiyang Zhou, Jingkan 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. 2, 3, 4, 6, 8
- [36] Kaiyang Zhou, Jingkan Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 2, 3, 6



# 中国科学技术大学

University of Science and Technology of China

## 计算机视觉

题目： 利用 GPT4 加强 CLIP

姓名： SA23011253 任永文

专业： 电子信息

学院： 计算机科学与技术学院

2023 年 11 月 18 日

## 目录

1 论文目的	3
2 论文原理	3
2.1 CLIP 和 CLIP 适配器 . . . . .	3
2.2 语言模型提示设计 . . . . .	4
2.3 简单的视觉句子少样本适配器 . . . . .	4
3 论文结果	5
3.1 GPT 适配器提高了少样本迁移能力 . . . . .	5
3.2 GPT 生成的视觉句子提高了零样本迁移能力 . . . . .	5
3.3 在不同的 GPT 模型上进行的消融实验 . . . . .	6
4 论文不足	6



## 1 论文目的

CLIP 等对比性预训练大型视觉语言模型 (VLMs) 彻底改变了视觉表示学习。通过设计与数据集相关的提示, VLM 能够在零样本情况下适应下游数据集。这种提示工程利用了领域的专业知识和验证数据集。同时, 像 GPT-4 这样的生成式预训练模型的发展意味着它们可以成为先进的互联网搜索工具, 也能够提供结构中的任何视觉信息。在这项工作中, 我们展示了 GPT-4 可用于生成视觉描述, 并展示了如何利用它来使 CLIP 适应下游任务。

我们的贡献如下: 1. 我们表明, 在提示中包含视觉描述性文本 (VDT) 信息可以提高 CLIP 的零样本领域迁移性能。2. 我们以可伸缩的方式利用 GPT-4 生成 VDT 句子, 并在零样本领域迁移中展示了与 CLIP 相比的一致性能改进。3. 我们设计了一个简单的适配器网络, 利用这些额外信息进行少样本迁移, 并展示了在从基础到新设置中 CoCoOp 等方法的性能改进。4. 我们发布了所有 12 个数据集的所有 VDT 信息, 以促进对大型 VLMs 在少样本迁移方面的提示集成和适配器设计的进一步研究。

## 2 论文原理

### 2.1 CLIP 和 CLIP 适配器

通过对大型图像-文本数据集进行对比预训练, CLIP 能对各种内容进行分类, 并将相关的图像和文本置于一个共享的嵌入空间中, 而将不相关的图像分离开来。经过预训练后, CLIP 直接对目标数据集进行图像分类, 而无需进行微调。

CLIP 模型由视觉模型和语言模型组成, 分别将图像和相应标题编码为视觉嵌入和文本嵌入。在推理过程中, 利用余弦相似度比较这些嵌入。在推理过程中, 给定一个  $H \times W \times C$  的图像, 视觉编码器将图像转换到联合嵌入空间, 得到图像  $D$  维的特征  $f$ 。使用提示模板如“类名的照片”为  $K$  个不同的类生成句子, 并通过文本编码器生成  $D \times K$  的类名权重矩阵  $W$ 。然后, 通过将图像特征  $f$  与  $W$  相乘, 并应用 softmax 函数计算预测概率。

$$f = \text{Backbone}(\mathbf{I}), p_i = \frac{\exp(\mathbf{W}_i^T f)/\tau}{\sum_{j=1}^K \exp(\mathbf{W}_j^T f)/\tau}, \quad (1)$$

在 CLIP 中, 零样本领域迁移利用了提示模板中的领域特定信息, 例如“类名的照片, 一种鸟的类型”。仔细的提示设计对于提高零样本分类精度至关重要。我们通过为每个类构造多个提示, 然后平均这些分类向量来实现提示集成。在我们的工作中, 我们证明了视觉描述性文本 (VDT) 的提示集成能够改善 CLIP 的零样本领域迁移性能。

CLIP-A 是一个可学习的 MLP 适配器, 用于辅助图像和/或单词编码器进行少样本迁移到目标数据集。在少样本迁移过程中, 对于每个类给定  $N$  个带标签的图像,  $K$  类权重通过提示模板  $H$  和文本编码器  $g$  来构造。图像特征  $f$  和文本特征  $W$  通过可学习的适配器  $A_v$  和  $A_t$  得到被适配的特征。超参数  $\alpha$  和  $\beta$  用于混合 CLIP 的知识与微调知识, 以避免 CLIP-适配器的过拟合, 使用整个训练集的交叉熵损失来优化  $A_v$  和  $A_t$ 。

$$f^* = \alpha A_v(f)^T + (1 - \alpha)f, \quad (2)$$

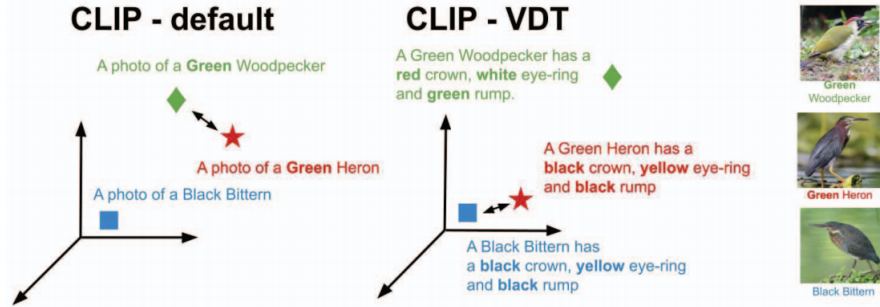
$$\mathbf{W}^* = \beta A_t(\mathbf{W})^T + (1 - \beta)\mathbf{W}. \quad (3)$$

## 2.2 语言模型提示设计

本节中演示了 VDT 如何增强 CLIP 的零样本迁移能力，并概述了使用 LLM 生成类特定提示的方法。精心设计和整合提示可以提升 CLIP 在零样本分类方面的性能。与默认提示以及包含非视觉信息的提示相比，在提示模板中添加视觉描述信息和集成可以提高零样本性能。

Methods	EuroSAT	Food101	DTD	Oxford Pets	CUB	ImageNet	Average
CLIP	47.69	85.97	43.09	89.07	54.70	64.51	64.17
DCLIP[19]	48.82	<b>88.50</b>	45.59	86.92	<b>57.75</b>	68.03	65.93
CLIP-GPT	<b>54.86</b>	86.43	<b>50.15</b>	<b>91.54</b>	57.43	<b>68.92</b>	<b>68.21</b>

但获取专家注释既昂贵又耗时，为了解决这一问题，我们利用了以其大规模知识和灵活性而闻名的 GPT 语言模型。我们的提示策略从思维链提示中得到启发，步骤如下：首先，我们要求 GPT-4 列出区分所考虑的 K 类图像所需的所有属性。其次，我们要求 GPT-4 为所有 K 个类提供这些属性的所有值作为句子。图的左侧展示了 CUB 数据集的示例。



表中的最后一行显示，生成的 GPT-4 视觉句子的表现与领域专家注释的类属性向量生成的句子相似。我们在基准测试套件中对所有数据集遵循相同的简单策略，以可扩展和灵活的方式生成可视化描述性句子，并利用它们构建提示集成。

## 2.3 简单的视觉句子少样本适配器

我们设计了一个简单的适配器，利用 VDT 信息改进了 CLIP 到目标数据集的少样本迁移。类似于 CLIP-A 文本，我们在单词编码器的输出中添加了一小组可学习的参数，并利用交叉熵损失训练适配器。我们的 CLIP-A-self 采用了自注意层，这层将注意力应用于每个类别不同句子的嵌入上，并对输出进行平均，得到最终的分类向量。

假设 K 类中每个类别都有 M 个 GPT 生成的句子，我们通过将每个句子附加到提示模板中构建了 M 个提示，并利用 CLIP 的单词编码器得到了  $D \times M \times K$  的  $\mathbf{W}_{\text{sent}}$ 。对于自注意适配

器，我们在所有的视觉描述句子上应用了基本的自注意机制，这样在训练过程中，它学习选择和聚合最相关的视觉句子来识别每个类别。和之前一样，我们首先得到了所有句子的分类向量  $K \times M \times D$  的  $W_s$ ，并将它们作为键、查询和值传递给自注意模块  $B_{self}$ ，并对输出进行平均得到最终的分类向量  $W^*$ ，这里的注意力被应用在  $M$  个不同的视觉描述性句子上。

$$W_{avg} = 1/M \sum_{m=1}^M W_{m,k}^s \quad (4)$$

$$\{W_{m,k}^a\}_1^M = B_{self}(\{W_{m,k}^s\}_1^M, \{W_{m,k}^s\}_1^M, \{W_{m,k}^s\}_1^M) \quad (5)$$

$$W_{a-mean} = 1/M \sum_{m=1}^M W_{m,k}^a \quad (6)$$

$$W^* = \beta W_{a-mean}^T + (1 - \beta) W_{avg} \quad (7)$$

最后，我们得到了新的适配器分类器权重  $D \times K$  的  $W^*$ ，这些权重对于在任何给定数据集的  $M$  个视觉描述性句子中聚焦于最具视觉辨别性的信息。我们计算概率，并通过选择概率最高的类别来预测图像类别。在少样本训练中，只使用交叉熵损失来训练适配器网络  $B_{self}$  的权重。

### 3 论文结果

#### 3.1 GPT 适配器提高了少样本迁移能力

Methods	EuroSAT	Caltech101	Oxford Flowers	Food101	FGVC Aircraft	DTD	Oxford Pets	Stanford Cars	Sun397	UCF101	CUB	ImageNet	Average
CLIP	47.69	93.75	70.69	85.97	<b>24.81</b>	43.09	89.07	<b>65.55</b>	62.61	<b>67.54</b>	54.70	64.51	64.16
CLIP-GPT	<b>54.86</b>	<b>94.51</b>	<b>73.40</b>	<b>86.43</b>	23.42	<b>50.15</b>	<b>91.54</b>	65.01	<b>67.24</b>	65.51	<b>57.43</b>	<b>68.9</b>	<b>66.53</b>

我们对比了使用 CLIP-GPT 提示集成和表 3 中 CLIP 默认提示的性能。GPT 生成的提示集成相较于 12 个数据集的 CLIP 零样本性能平均提高了 2%。对于专门的细粒度数据集性能提升较高，表明 CLIP 在识别这些数据集中的某些类别时存在困难，但当以视觉描述性句子的形式提供类别信息时，从文本编码器中获取的分类器和精度更佳。

值得注意的是，仅当提示中包含视觉描述性句子时，才能帮助提高诸如 Imagenet 等通用数据集的性能。对于动作识别数据集 UCF-101 之类的数据集效果较差，我们检查了为 UCF-101 生成的句子，发现其中有几个涉及时间信息而非视觉描述，这可能是准确性下降的原因。少样本适配器的自注意模块学会了从生成的句子中强调视觉方面，这可能解释了在新设置中少样本适配器性能的提高。

#### 3.2 GPT 生成的视觉句子提高了零样本迁移能力

下图是 CLIP-A-self 与 CLIP、CoOp 和 CoCoOp 在从基础到新的泛化设置中的比较。对于基于提示学习的方法（CoOp 和 CoCoOp），他们的提示是从基础类（16 个样本）中学习的。

这些结果有力地证明了包含额外的 VDT 信息的重要性。H 表示谐波平均值（以突出泛化的权衡）。

(a) CUB.				(b) Caltech101.				(c) OxfordPets.			
	Base	New	H		Base	New	H		Base	New	H
CLIP	58.7	70.3	63.90	CLIP	96.84	94.00	95.40	CLIP	91.17	97.26	94.12
CoOp	79.2	53.3	63.71	CoOp	98.00	89.81	93.73	CoOp	93.67	95.29	94.47
CoCoOp	67.1	74.1	70.40	CoCoOp	97.96	93.81	95.84	CoCoOp	<b>95.20</b>	<b>97.69</b>	<b>96.43</b>
CLIP-A	68.3	70.8	69.53	CLIP-A	97.7	93.6	95.61	CLIP-A	94.8	97.0	95.89
CLIP-A-self	<b>78.6</b>	<b>71.3</b>	<b>74.77</b>	CLIP-A-self	<b>98.3</b>	<b>95.9</b>	<b>97.09</b>	CLIP-A-self	94.4	97.0	95.68
(d) StanfordCars.				(e) Flowers102.				(f) Food101.			
	Base	New	H		Base	New	H		Base	New	H
CLIP	63.37	<b>74.89</b>	68.65	CLIP	72.08	<b>77.80</b>	74.83	CLIP	90.10	91.22	90.66
CoOp	<b>78.12</b>	60.40	68.13	CoOp	<b>97.60</b>	59.67	74.06	CoOp	88.33	82.26	85.19
CoCoOp	70.49	73.59	72.01	CoCoOp	94.87	71.75	81.71	CoCoOp	<b>90.70</b>	<b>91.29</b>	<b>90.99</b>
CLIP-A	70.5	73.3	71.87	CLIP-A	94.6	71.5	81.44	CLIP-A	90.3	91.2	90.75
CLIP-A-self	76.8	72.9	<b>74.80</b>	CLIP-A-self	97.4	75.3	<b>84.94</b>	CLIP-A-self	90.4	91.2	90.80
(g) FGVC Aircraft.				(h) SUN397.				(i) DTD.			
	Base	New	H		Base	New	H		Base	New	H
CLIP	27.19	<b>36.29</b>	31.09	CLIP	69.36	75.35	72.23	CLIP	53.24	59.90	56.37
CoOp	<b>40.44</b>	22.30	28.75	CoOp	<b>80.60</b>	65.89	72.51	CoOp	79.44	41.18	54.24
CoCoOp	33.41	23.71	27.74	CoCoOp	79.74	76.86	78.27	CoCoOp	77.01	56.00	64.85
CLIP-A	34.9	33.5	34.19	CLIP-A	80.1	75.9	77.94	CLIP-A	74.9	53.0	62.08
CLIP-A-self	37.8	33.0	<b>35.24</b>	CLIP-A-self	81.4	76.8	79.03	CLIP-A-self	<b>81.8</b>	<b>62.3</b>	<b>70.73</b>
(j) EuroSAT.				(k) UCF101.				(l) ImageNet.			
	Base	New	H		Base	New	H		Base	New	H
CLIP	56.48	64.05	60.03	CLIP	70.53	<b>77.50</b>	73.85	CLIP	72.43	68.14	70.22
CoOp	<b>92.19</b>	54.74	68.69	CoOp	<b>84.69</b>	56.05	67.46	CoOp	<b>76.47</b>	67.88	71.92
CoCoOp	87.49	60.04	71.21	CoCoOp	82.33	73.45	77.64	CoCoOp	75.98	<b>70.43</b>	<b>73.10</b>
CLIP-A	82.5	62.4	71.06	CLIP-A	82.9	74.9	78.70	CLIP-A	75.4	68.6	71.84
CLIP-A-self	88.5	70.5	78.48	CLIP-A-self	84.1	76.4	<b>80.07</b>	CLIP-A-self	76.4	68.3	72.12

### 3.3 在不同的 GPT 模型上进行的消融实验

Prompting	ZS	Base	New	H
Default	54.7	NA	NA	NA
OpenAssistant	56.0	78.3	69.8	73.80
GPT-3.5	55.7	78.1	70.6	74.16
GPT-4	57.4	78.6	71.3	74.77

比较不同的 GPT 模型获取 VDT 信息的能力。我们看到，较大的模型能提供更高质量的 VDT 信息，但 CLIP-A-self 即使使用像 OpenAssistant 这样较小的模型，也能够产生可泛化的分类器。

## 4 论文不足

实验结果表明该方法在类别数较少或者分类粒度较小的数据集中表现较好，但是对于其他对类别数或者自动化要求较高的情况来说，使用 GPT4 获得类定制的提示可能不太可靠。