What does a platypus look like?
Generating customized prompts for zero-shot image classification

Sarah Pratt[1*] Ian Covert[1] Rosanne Liu[2, 3] Ali Farhadi[1]

[1]University of Washington [2]Google DeepMind [3]ML Collective

Abstract

Open-vocabulary models are a promising new paradigm for image classification. Unlike traditional classification models, open-vocabulary models classify among any arbitrary set of categories specified with natural language during inference. This natural language, called "prompts", typically consists of a set of hand-written templates (e.g., "a photo of a {}") which are completed with each of the category names. This work introduces a simple method to generate higher accuracy prompts, without relying on any explicit knowledge of the task domain and with far fewer hand-constructed sentences. To achieve this, we combine open-vocabulary models with large language models (LLMs) to create Customized Prompts via Language models (CuPL, pronounced "couple"). In particular, we leverage the knowledge contained in LLMs in order to generate many descriptive sentences that contain important discriminating characteristics of the image categories. This allows the model to place a greater importance on these regions in the image when making predictions. We find that this straightforward and general approach improves accuracy on a range of zero-shot image classification benchmarks, including over one percentage point gain on ImageNet. Finally, this simple baseline requires no additional training and remains completely zero-shot. Code available at `https://github.com/sarahpratt/CuPL`.

1. Introduction

Open-vocabulary models [40, 23, 42, 63] achieve high classification accuracy across a large number of datasets without labeled training data for those tasks. To accomplish this, these models leverage the massive amounts of image-text pairs available on the internet by learning to associate the images with their correct caption, leading to greater flexibility during inference. Unlike standard models, these models classify images by providing a similarity score between an image and a caption. To perform inference, one can generate a caption or "prompt" associated with each of the desired categories, and match each image to the best prompt. This means that categories can be selected ad hoc and adjusted without additional training.

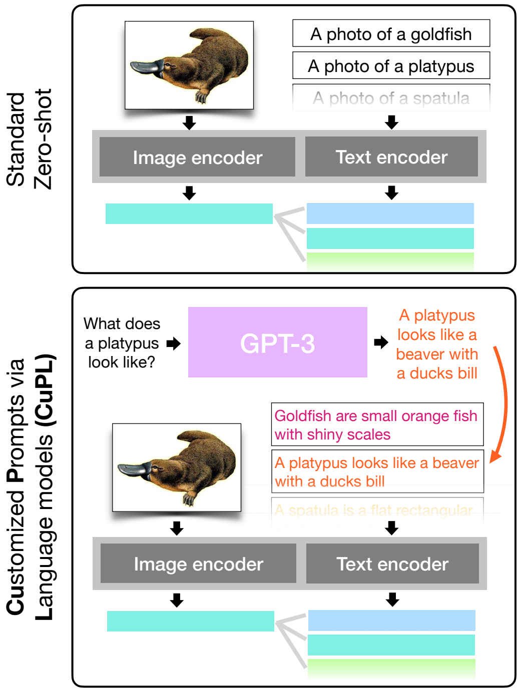
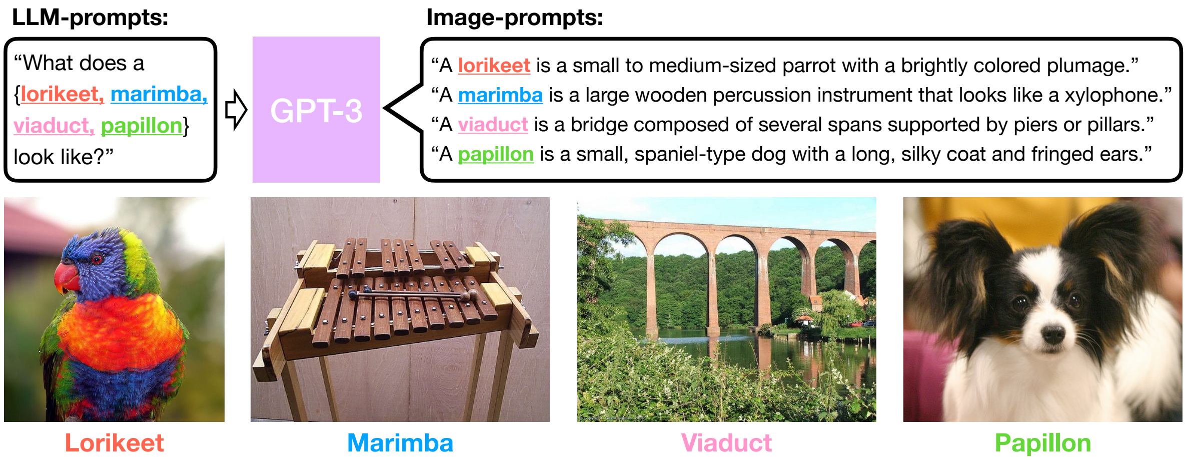


Figure 1. **Schematic of the method.** (Top) The standard method of a zero-shot open-vocabulary image classification model (e.g., CLIP [42]). (Bottom) Our method of CuPL. First, an LLM generates descriptive captions for given class categories. Next, an open-vocabulary model uses these captions as prompts for performing classification.

However, this new paradigm poses a challenge:

How can we best represent an image category through natural language prompts?

[*]Correspondence to `spratt3@uw.edu`.



"What does a {lorikeet, marimba, viaduct, papillon} look like?"

"A lorikeet is a small to medium-sized parrot with a brightly colored plumage."
"A marimba is a large wooden percussion instrument that looks like a xylophone."
"A viaduct is a bridge composed of several spans supported by piers or pillars."
"A papillon is a small, spaniel-type dog with a long, silky coat and fringed ears."

Figure 2. **Example CuPL LLM-prompts and Image-prompts.** LLM-prompts are filled in with a class name and then used as input to GPT-3, which then outputs image-prompts. Example LLM generated image-prompts and associated images from ImageNet are shown. Only image-prompts are used for the downstream image classification.

The standard approach is to hand write a number of prompt templates [42] (e.g.,"a photo of a {}"), compile a natural language label for each category in the dataset, and create a set of prompts for each category by filling in each template with the natural language labels. Then, image embeddings are matched to the nearest set of prompt embeddings and labelled with the category associated with that set of prompts (more details in Section 2).

This method has three major drawbacks. Firstly, each prompt template has to be hand-written, so having twice as many prompts for a category requires twice as much human effort. This can become costly as each new dataset typically has a different set of prompt templates [42].

Secondly, the prompt templates must be general enough to apply to all image categories. For example, a prompt for the ImageNet [13] category "platypus" could only be as specific as "a photo of a {platypus}", and could not be something like "a photo of a {platypus}, a type of aquatic mammal" as that template would no longer be relevant for other image categories. This is limiting, as descriptive details are useful for fine-grained classification. For example, different species of frogs share many of the same characteristics. However, tree frogs can be distinguished with their distinct large eyes. This is a valuable detail for classification but cannot be included in a general template. Therefore, when using these basic templates, the model may not take advantage of this detail in the image, leading to an incorrect categorization as demonstrated in Figure 5.

Lastly, writing high performing prompt templates currently requires prior information about the contents of the dataset. For example, the list of hand-written ImageNet prompts [42] includes "a black and white photo of the {}.", "a low resolution photo of a {}.", and "a toy {}." all of which demonstrate prior knowledge about the type of representations present in the dataset. This information is not generalizable to other datasets, as ImageNet contains "black and white" and "toy" representations of its categories, but other datasets do not (e.g., FVGC Aircraft [32]).

To overcome these challenges, we propose Customized Prompts via Language models (CuPL). In this algorithm, we couple a large language model (LLM) with a zero-shot open-vocabulary image classification model. We use the LLM to generate prompts for each of the image categories in a dataset. Using an LLM allows us to generate an arbitrary number of prompts with a fixed number of hand-written sentences. Additionally, these prompts are now customized to each category and contain specified visual descriptions while still remaining zero-shot. This allows prompts to contain details about a class which distinguish it from other similar classes. For example, to describe a tree frog, the LLM generates the sentence "A tree frog looks like a small frog with large eyes." This not only describes the category, but specifically mentions the eyes, the feature which distinguishes the Tree frog class from the most visually similar classes - other types of frogs. We find that CuPL prompts are rich with these discriminating details and show that the model is able to leverage these details to place more importance on relevant parts of the image when classifying between similar, commonly confused categories (Figure 5).

We find these customized prompts outperform the hand-written templates on 15 zero-shot image classification benchmarks, including a greater than 1 percentage point gain on ImageNet [13] Top-1 accuracy and a greater than 6 percentage point gain on Describable Textures Dataset [11], with fewer hand-written prompts when compared to the standard method used in [42]. Finally, this method requires no additional training or labeled data for either model.

2. Methods

The CuPL algorithm consists of two steps: (1) generating customized prompts for each of the categories in a given dataset and (2) using these prompts to perform zero-shot image classification.

2.1. Generating Customized Prompts

This step consists of generating prompts using an LLM. For clarity, we distinguish between two different kind of prompts. The first are the prompts which cue the LLM to generate the descriptions of the dataset categories. These prompts do not describe an object, but rather prompt the description of an object (e.g., "What does a platypus look like?"). We will refer to these as "LLM-prompts".

Secondly, there are the prompts to be matched with images in the zero-shot image classification model. These are the prompts that describe a category (e.g., "A platypus looks like ..."). We call them "image-prompts." In CuPL, these are the output of the LLM, as exemplified in Figure 2.

In this work, we use GPT-3 [5] as our LLM. To generate our image-prompts, we must first construct a number of LLM-prompt templates. While this does require some engineering by hand, it is significantly less than the amount of hand-engineered sentences used in the standard method of creating image-prompt templates for CLIP. For example, in our ImageNet experiments, we construct 5 LLM-prompt templates compared to the 80 image-prompts used by CLIP for zero-shot ImageNet classification.

After constructing these LLM-prompts, we generate 10 different image-prompts for each of the LLM-prompts. This means for ImageNet we use an LLM to generate a total of 50 customized image-prompts for each image category. For each of these, we generate a maximum of 50 tokens, but halt a generation early if it produces a period. Additionally, we generate with a high temperature of 0.99, which encourages more diversity among the 10 generated image-prompts. We also clean each generated sentence by deleting any blank lines and adding a period at the end.

2.2. Utilizing Customized Prompts

After generating image-prompts for each of the categories, we then perform zero-shot image classification. While there are a number of open-vocabulary models [40, 23, 42, 63], we report our results using CLIP [42] as this is the most popular publicly available open-vocabulary model.

CLIP consists of a text encoder and and image encoder (schematic on the top of Figure 1). In the standard setting, there are a number of hand-written templates which can be completed with the relevant category names (e.g. "A photo of a {}", "A photo of many {}"). To classify the images in a dataset, each of these templates is filled in with a given category name. Then each of these sentences is embedded via the text encoder, and all sentences completed with the same category name are averaged and normalized. This results in n embeddings where n is the number of categories in the dataset. Each of these n embeddings is the mean of many different sentence embeddings. Then each image in the dataset is embedded using the image encoder. This embedding is compared to each of the n text embeddings using cosine similarity and is labeled with the most similar one.

CuPL requires only a small adjustment from this standard practice. Instead of filling in the hand-written templates for each category, we simply replace these altogether with the sentences output by GPT-3. This means that for CuPL, hand-written templates are only used as input for the LLM, while the prompts for CLIP are entirely generated text. We present 2 different setting of CuPL (as shown in Table 1), each representing a different trade-off between accuracy and hand-engineering.

1. CuPL (base). This setting uses three hand-written sentences across all 15 examined datasets. We do this by constructing general LLM-prompt templates which are filled in with the category names for each dataset. Our three general templates are as follows:

Describe what a/the ___ looks like:
Describe a/the ___ :
What are the identifying characteristics of a/the ___ ?

The blank portion of this template is either filled in with the category type plus the category name (e.g. "pet" + {} for the Oxford Pets dataset [38] or "aircraft" + {} for FGVC Aircraft [32]) or just the category name for more general datasets like ImageNet [13]. Type specification is necessary because of words that have multiple meanings. For example "boxer" from the Oxford Pets dataset can also mean a person who boxes, as opposed to a dog breed, so it is necessary to specify "Describe a pet boxer:". Similarly, "Tornado" from the FGVC Aircraft dataset can be a type of aircraft or a type of weather.

2. CuPL (full). In this setting we use different LLM-prompt templates for each dataset, just as [42] uses different image-prompt templates for each dataset. However, we use fewer hand-written templates overall and also contain less specific information about each dataset in the templates. For this work, each dataset has between 2 and 9 LLM-prompts which generate between 20 and 90 image-prompt per category (10 generated sentences per LLM-prompt). For ImageNet, we use the following 5 LLM-prompts: (1) "Describe what a(n) {} looks like", (2) "How can you identify a(n) {}?", (3) "What does a(n) {} look like?", (4) "A caption of an image of a(n) {}", (5) "Describe an image from the internet of a(n) {}". Full LLM-prompts for all datasets as well as example image-prompts are given the Appendix.

	ImageNet	DTD	Stanford Cars	SUN397	Food101	FGVC Aircraft	Oxford Pets	Caltech101	Flowers 102	UCF101	Kinetics-700	RESISC45	CIFAR-10	CIFAR-100	Birdsnap	mean	Total	Unique
std	75.54	55.20	77.53	69.31	93.08	32.88	93.33	93.24	78.53	77.45	60.07	71.10	95.59	78.26	50.43	73.43		
# hw	80	8	8	2	1	2	1	34	1	48	28	18	18	18	1		268	175
CuPL (base)	76.19	58.90	76.49	72.74	93.33	36.69	93.37	93.45	78.83	77.74	60.24	68.96	95.81	78.47	51.11	74.15		
Δ std	+0.65	+3.70	-1.04	+3.43	+0.25	+3.81	+0.04	+0.21	+0.30	+0.29	+0.17	-2.14	+0.22	+0.21	+0.63			
# hw	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3		45	3
CuPL (full)	76.69	61.70	77.63	73.31	93.36	36.11	93.81	93.45	79.67	78.36	60.63	71.69	95.84	78.57	51.11	74.80		
Δ std	+1.15	+6.50	+0.10	+4.00	+0.28	+3.23	+0.48	+0.21	+1.14	+0.91	+0.56	+0.59	+0.25	+0.31	+0.63			
# hw	5	6	9	3	3	2	2	3	2	5	4	5	3	4	3		59	45

Table 1. **Performance of CuPL prompts compared to the standard, hand-written prompts in CLIP [42] on 15 zero-shot image classification benchmarks.** "Δstd" stands for the difference; green shows improvement. In addition to accuracy, we show number of prompt templates that are hand-written ("# hw") for each dataset using each method, as well as the total and unique number of hand-written templates for each method (unique number only counts templates once even if used for multiple datasets). Note that CuPL (base) uses just three hand-constructed sentence across all datasets compared to 175 in the standard method.

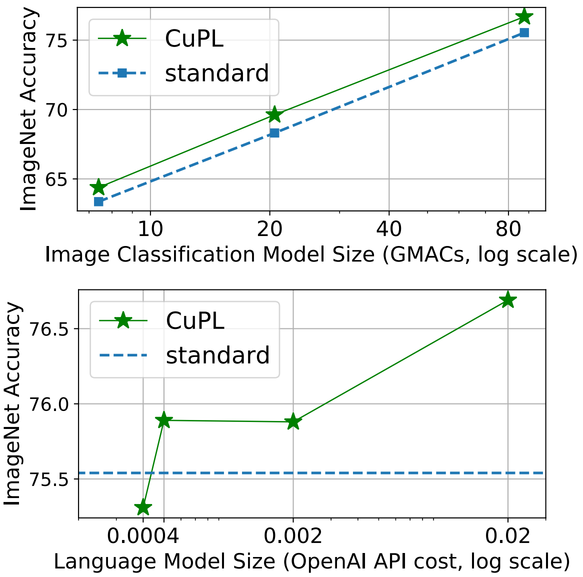


Figure 3. **Performance of CuPL as models scale.** (Top) ImageNet Top-1 accuracy for various scales of CLIP. CuPL prompts remain consistently better than standard prompts even we adjust CLIP model size (ViT-B/32, ViT-B/16, ViT-L/14). GPT-3 model set as DaVinci-002. (Bottom) ImageNet Top-1 accuracy for various scales of GPT-3 (ada, babbage, curie, davinci-002). Larger models produce higher accuracy. CLIP model set as ViT-L/14.

3. Experiments and Results

We first discuss the details of our experimental setup. We next show improvements on a wide range of image classification benchmarks. We then examine the scaling behavior with respect to the model size and report observations regarding hyperparameters such as the number of hand-written prompts. Finally, we provide evidence that the model is able to use CuPL prompts to place more importance on the most relavent parts of the image.

3.1. Setup

Unless specified otherwise, we use CLIP with a backbone of ViT-L/14 [14] and the GPT-3 DaVinci-002 model. Additionally, in order to perform open-vocabulary image classification, each image category needs a natural language label. This is sometimes provided by the dataset, but not always (e.g. ImageNet categories are described by an id number which can map to multiple synonyms). For this work, we use the same natural language labels specified in [42].

We report our findings on 15 zero-shot image recognition benchmarks: ImageNet [13], Describable Textures Dataset (DTD) [11], Stanford Cars [26], Scene UNderstanding (SUN397) [60], Food101 [4], FGVC Aircraft [32], Oxford Pets [38], Caltech101 [16], Flowers 102 [36], UCF101 [52], Kinetics-700 [8], Remote Sensing Image Scene Classification (RESISC45) [10], CIFAR-10 [27], CIFAR-100 [27], and Birdsnap [2]. For the two video datasets, we extract the middle frame of the video, as is done in Radford *et al.* [42].

3.2. Results

Our results for the base prompts setting and the full prompts setting are in Table 1. We present our method's performance on 15 different image classification benchmarks, comparing both the classification accuracy and the number of hand-written sentence templates needed for each method. Note that for the standard method [42], the hand-written sentences refer to the image-prompts, while for

Standard	**CuPL**	**Menon *et al.*** [33]
75.54	76.69	75.00

Table 2. **Comparison with Menon *et al.* [33]** on Top-1 Imagenet accuracy with ViT L/14.

CuPL the hand-written sentences refer to the LLM-prompts, with which image-prompts are generated.

1. CuPL (base). In this setting, we see performance gains in 13 out of the 15 examined datasets. Note this setting uses *just three hand-constructed sentence across all datasets*. This is in comparison to the nearly 175 unique image-prompt templates that are hand-written across all of these datasets in the standard setting. Additionally, in the standard setting these hand-constructed prompts must be very specific to the dataset (e.g., "a black and white photo of a {}.", "a plastic {}."). In comparison, CuPL (base) requires only the category type of the overall dataset and still outperforms the hand-written, domain specified baseline in almost all cases. Thus, we present this base prompt setting as a simple standard that matches or exceeds prompt engineering open-vocabulary models.

2. CuPL (full prompts). Here we see improvements on all examined datasets. This includes large (over 1 percentage point) gains on ImageNet Top-1, DTD (texture classification), SUN397 (scene classification), FGVC Aircraft (fine-grained aircraft classification), and Flowers 102 (flower classification). While this setting requires more hand-written prompts than setting (1), it still requires significantly fewer than the baseline method (5 sentences versus 80 sentence for ImageNet), and does not include knowledge about the image domain. The full list of hand-constructed sentences for CuPL (full prompts) and the baseline method [42] can be found in the Appendix.

3.3. Analysis and Ablations

Other prompting techniques. Concurrent work by Menon *et al.* [33] also explores LLM generated descriptions for image classification. This work differs from CuPL as it generates a structured list of identifying attributes in a single generation, which are reformatted into multiple sentences. In contrast, CuPL outputs a single sentence for multiple generations, with no enforced format. The benefit of the structured output used in Menon *et al.* [33] is that the authors can examine the similarity of a given image with each individual attribute to understand which ones most contribute to a prediction. However, unlike CuPL, this method performs worse than standard human-written prompts, as shown in Table 2. This is potentially because this work focuses on explainability, and therefore enforces a strict format on the generated prompts, likely reducing overall accuracy.

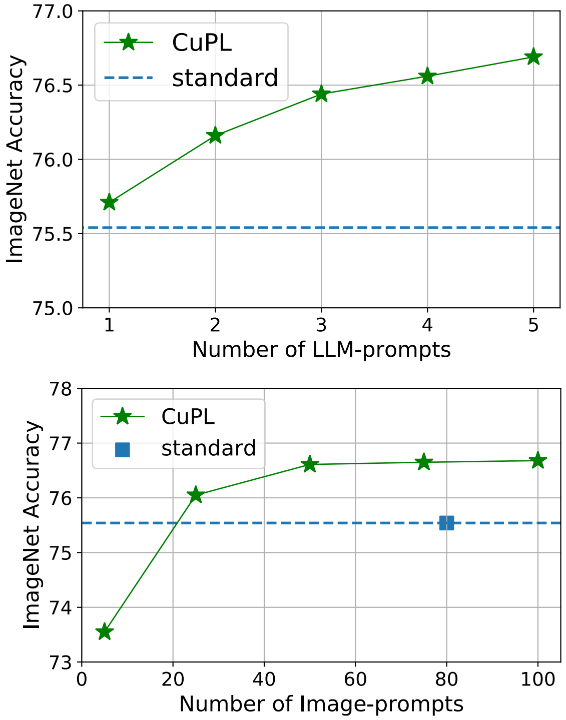


Figure 4. **Ablation on number of LLM-prompts (top) and image-prompts (bottom).** (Top) As number of hand-written LLM-prompts increases, so does accuracy. 10 image-prompts are generated for each LLM-prompt. Note that CuPL outperforms the baseline even with just one hand-written sentence. We add the prompts in a greedy manner, at each step adding the 10 prompts which lead to the largest performance gain. (Bottom) We adjust the number of image-prompts generated by a fixed number (5) of LLM-prompts. Even at 5 Image-prompts per LLM-prompt (25 prompts total), we outperform the baseline which uses 80 image-prompts.

Model Size. In Figure 3, we show CuPL (full prompts) at different model scales. As there are two different zero-shot models in the CuPL algorithm, we show the effects of varying each model individually. On the top, we vary the CLIP model used while holding the LLM constant. We see consistent gains across all model sizes. On the bottom, we vary the size of the LLM. We plot the accuracy of the baseline as well, which does not vary as it does not utilize an LLM. We find larger models lead to higher accuracy, though the 2nd and 3rd largest models perform similarly.

Number of Prompts. In Figure 4, we present ablations on the number of LLM-prompts and image-prompts for CuPL (full prompts). On the top, we show ImageNet accuracy as we increase the number of LLM-prompts. This also corresponds to the number of sentences that have to be hand-written. Notably, this method outperforms the baseline even when using prompts generated from a single hand-written sentence. On the bottom, we hold the number of

LLM-prompts constant at 5 and adjust how many image-prompts we generate per LLM-prompt. We plot the accuracy given the total number of image-prompts (so 10 generated image-prompt per LLM-prompt corresponds to 50 total image-prompts). We see that CuPL begins to outperform the baseline at just 25 image-prompts, well below the 80 image-prompts used in the baseline.

Additional Analysis. In the Appendix, we provide comparisons between CuPL prompts and descriptive prompts generated with definitions of ImageNet classes as well as with Wikipedia descriptions of ImageNet classes. We find that CuPL prompts outperform both of these baselines. Additionally, we provide results of ensembling CuPL prompts and the baseline hand-written prompts used in [42]. We find that this ensemble outperforms just baseline prompts for all datasets, and outperforms just CuPL prompts for some datasets.

3.4. Shapley Value Analysis

We show that CuPL descriptions allow CLIP to place more importance on image regions that are most relevant for the correct classification. In order to measure the importance of regions in the image, we invoke Shapley values [49], a tool from game theory that has become popular for understanding which input information contributes to a model's final prediction [31, 9]. Shapley values can be computed for any model, and although there are methods designed specifically for vision transformers [12] (the underlying architecture of CLIP), we use a simple model-agnostic calculation [35]. We employ Shapley values to understand the importance of different image regions with CuPL prompts versus baseline prompts, and we find that CuPL places more value on regions that are emphasized in object descriptions, and thus are likely important for obtaining correct classifications. We demonstrate this correlation in two ways: (1) visualizing heatmaps of importance over images, and (2) measuring the importance of segmented image parts annotated by the PartImageNet Dataset [18].

Importance Heatmaps To understand how CuPL captions lead to a change in importance of different image regions, we calculate the Shapley value of small image patches when using CuPL prompts versus when using baseline prompts. We calculate the Shapley values with respect to a binary classification probability between the correct class and a similar distractor class in order to understand how CuPL corrects these errors. As shown in Figure 5, we examine the important regions of an image of a dog when classifying between two very similar dog categories: a "Schipperke dog" versus a "Groenendael dog". Both of these classes are Belgian dogs that are black with pointy ears. However, they have a few subtle differences including the typical appearance of their tails. Additionally, we show the important regions of an image when classifying between a "Tree frog" and a "Tailed frog", which also look very similar.

For each binary classification, we show four heatmaps: (1) the regions that contribute to a higher probability of the correct class when using CuPL prompts, (2) the regions that contribute to a higher probability of the incorrect class when using CuPL prompts, (3) the regions that contribute to a higher probability of the correct class when using baseline prompts, (4) the regions that contribute to a higher probability of the incorrect class when using baseline prompts. Interestingly, we find that not only does CuPL place importance on different regions of the image, but these regions correspond to descriptions in the CuPL prompts. For example, the tail of the dog is very important to the "Schipperke" probability when using CuPL prompts, but not when using baseline prompts, and the tail of the Schipperke dog is described 10 times in the CuPL descriptions of this class. Similarly, we find that the eyes in the image of the frog are much more important when classifying with CuPL than with the baseline, and that the eyes are mentioned 10 times in the CuPL description of a tree frog. We provide more examples of this phenomenon in the Appendix.

Importance of Segmented Parts In order to understand the correlation between the importance of an image region and its frequency in CuPL prompts on a larger scale, we utilize the PartImageNet Dataset [18]. This dataset contains segmentation maps of the different parts of a class for a subset of ImageNet classes. For example, the dog classes have the parts: 'head', 'body', 'leg' and 'tail'. We use these segmentation maps to obtain the Shapley value for each part of the animal with respect to the final probability of the ground truth class. To understand the effect of changing to CuPL prompts, we calculate the difference between the Shapley values with CuPL prompts and with baseline prompts, and we average across all images in a class. So for each part in each examined class we calculate the following (where SV denotes the Shapley value):

$$\frac{1}{|\text{class}|}\sum_{\text{image}\in\text{class}} \text{SV}_{\text{CuPL}}(\text{image}, \text{part}) - \text{SV}_{\text{base}}(\text{image}, \text{part})$$

This gives us a score for how much more important a part of an animal is to CuPL compared to the baseline for classification. Additionally, we quantify how prevalent each body part is in the CuPL descriptions. We do this using the WordNet [34] database to tag each words as part of the 'leg', 'head', etc. More details of this tagging system are given in the Appendix. We present our findings in Figure 6. We find that the parts that are more important to CuPL are highly correlated with the parts that are present in the descriptions of the animals (and thus likely important to the identification of the animal). For example, head-related attributes of the Japanese Spaniel class are frequently mentioned in the descriptions. Additionally, the 'head' in the image is much more important to the final prediction for CuPL than for

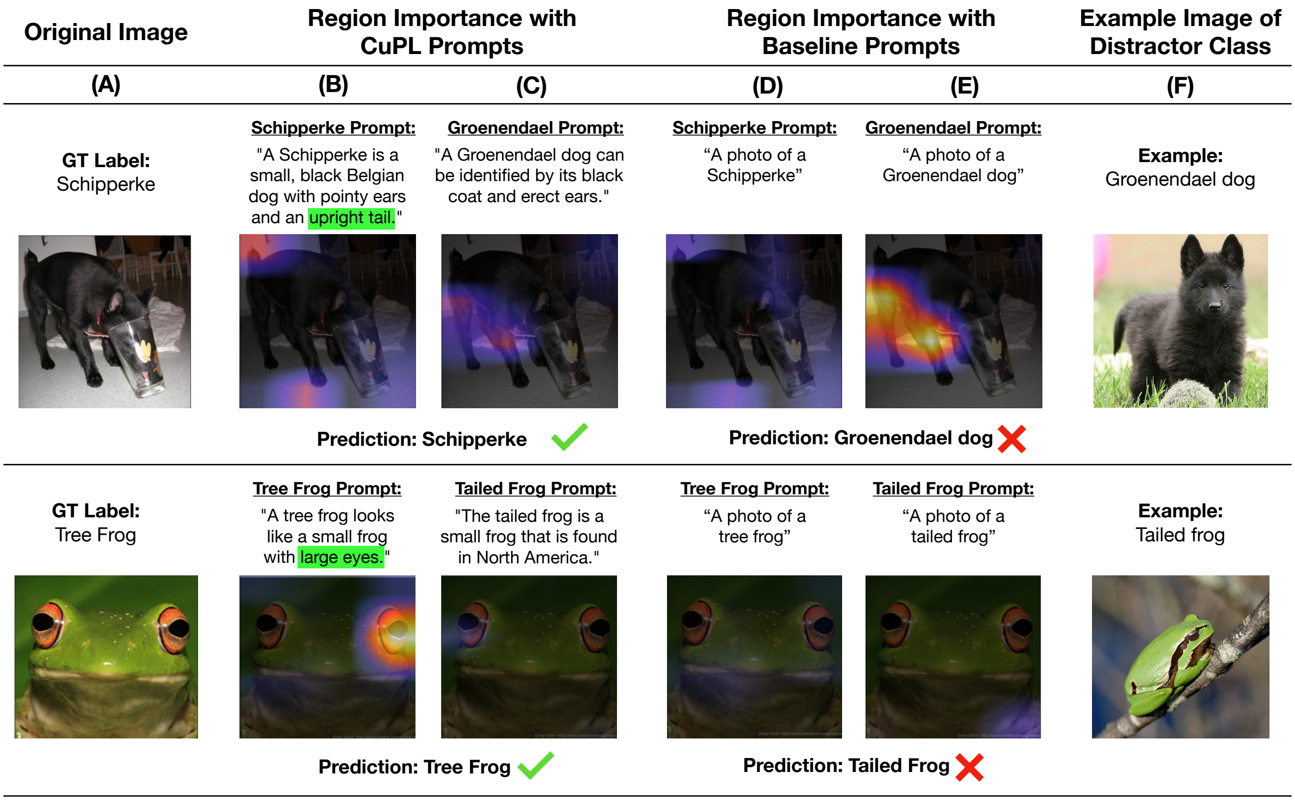

Original Image
Region Importance with CuPL Prompts
Region Importance with Baseline Prompts
Example Image of Distractor Class
(A)
(B)
(C)
(D)
(E)
(F)
GT Label: Schipperke
Schipperke Prompt:
"A Schipperke is a small, black Belgian dog with pointy ears and an upright tail."
Groenendael Prompt:
"A Groenendael dog can be identified by its black coat and erect ears."
Schipperke Prompt:
"A photo of a Schipperke"
Groenendael Prompt:
"A photo of a Groenendael dog"
Example: Groenendael dog
Prediction: Schipperke ✓
Prediction: Groenendael dog ✗
GT Label: Tree Frog
Tree Frog Prompt:
"A tree frog looks like a small frog with large eyes."
Tailed Frog Prompt:
"The tailed frog is a small frog that is found in North America."
Tree Frog Prompt:
"A photo of a tree frog"
Tailed Frog Prompt:
"A photo of a tailed frog"
Example: Tailed frog
Prediction: Tree Frog ✓
Prediction: Tailed Frog ✗

Figure 5. **CuPL prompts lead the model to focus on semantically important regions of the image.** We use Shapley values (Section 3.4) to visualize the importance of each region in a binary classification problem. We examine which parts of an image lead the model to classify it as the correct class versus a commonly confused class. We present the original image (column A), as well as four heatmaps showing which regions raise the probability of the *correct* class for the *CuPL* model (column B), the *incorrect* class for the *CuPL* model (column C), the *correct* class for the *baseline* model (column D), and the *incorrect* class for the *baseline* model (column E). Additionally, we show that the regions that are more important to CuPL than to the baseline correspond to regions mentioned in the CuPL prompts (i.e. "tail" which is a commonly mentioned word in Schipperke Dog CuPL prompts and "eyes" which is a common word in Tree Frog prompts). We also show an example image from the distractor class to demonstrate the level of similarity between these fine-grained classes (column F). Finally, we see that CuPL scores the correct class higher, whereas the baseline scores the incorrect class higher. This series of observations lead us to believe that CuPL is able to correct errors because the descriptive prompts cause the model to weigh semantically important regions more heavily.

baseline. Thus, CuPL is able to extract important information for identifying the animal from the text and incorporate it into classification predictions.

4. Related Work

4.1. Natural Language Descriptions for Image Classification

Several prior works use text-based knowledge of image categories to improve classification accuracy. [15] extract visual information from unstructured text descriptions collected from the internet to recognize parts of object and classify them in a zero-shot way. [45] and [19] use natural language descriptions of bird types to train a multimodal classification model. [21] use hand-collected attribute tags to attend over relevant features in images. [39] extract visual information from Wikipedia descriptions to enable zero-shot bird classification. Additional works [50, 6] show improvements on large datasets (e.g., ImageNet) using external information from external databases such as Imagenet-wiki and Wordnet. While these works show the effectiveness of augmenting zero-shot models with descriptive text, all of these prior works rely on external natural language databases for descriptions. This often limits the possible categories that can be classified and can require extensive preprocessing to extract visual descriptions from noisy natural language.

4.2. Generated Text for Downstream Tasks

Recent work has utilized text generated from LLMs in a number of ways. [47] use an LLM to paraphrase existing image captions to use as data augmentation for CLIP. [30]

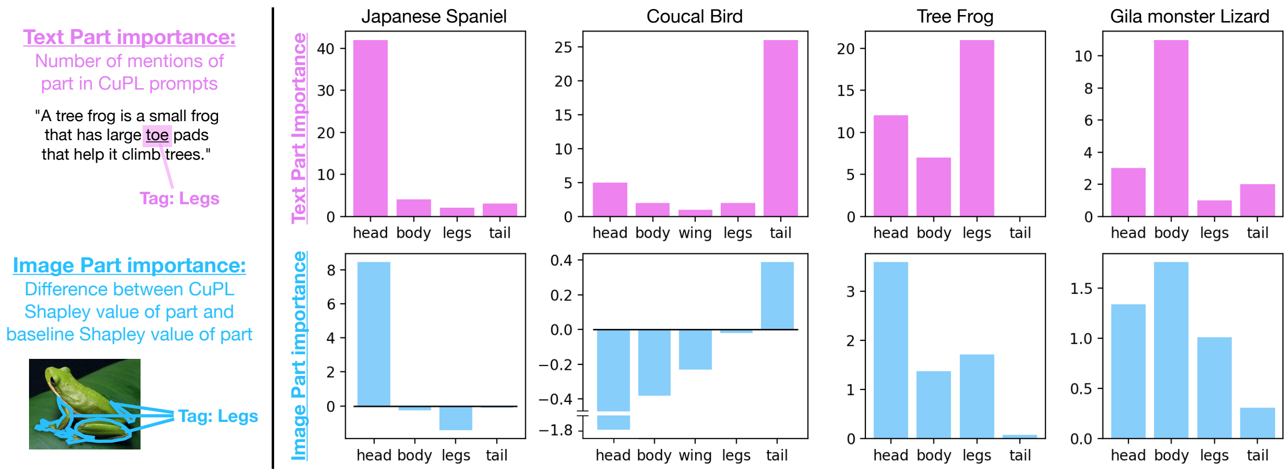


Figure 6. **When specific parts of an animal/object are frequently mentioned in CuPL prompts, the CuPL model places more importance on these parts in the image compared to the baseline model.** The PartImageNet dataset [18] provides segmentation maps of ImageNet images broken down into parts. For example, Tree Frog is broken down into the parts: 'head', 'leg', 'body' and 'tail'. We use the WordNet database [34] to tag words in CuPL prompts as belonging to one of these parts. We refer to the number of mentions of the part as the Text Part Importance. We then use the PartImageNet segmentations to compare the Shapley value of each part when using CuPL prompts and baseline prompts, which we call the Image Part Importance. We find a strong correlation between the Text Part Importance and the Image Part Importance, leading to the conclusion that CuPL is able to take advantage of the knowledge contained in the descriptions when making its predictions.

use GPT-3 to generate knowledge on a topic when given a number of demonstrations, which is then used to improve accuracy on common sense reasoning questions. [20] use a LLM to add labels to text to improve text classification accuracy. In [64], the outputs of a GPT-2 model are used to train an encoder on top of a vision model to generate multimodal image representations for a variety of tasks. [53] utilize a language model to perform image captioning by iteritively generating candidate image captions with a LLM and then using feedback from an open-vocabulary model to align it to a given image. Similarly, [62] use GPT-3 along with text descriptions of images for the Visual Question Answering (VQA) task. However, unlike CuPL these prior works are either purely language tasks (common sense reasoning, text classification) or multimodal with some language component (image captioning, VQA). Most similarly, [33] use and LLM to generate a structured list of attributes which are then reformatted into captions for CLIP. However this work differs from ours as it does not improve over human written templates. Additionally, [61] use an LLM to generate a list of natural language attributes for ImageNet classes and then select a subset of these attributes for each class in a few-shot manner. Our work differs from this as we remain in the zero-shot setting.

4.3. Prompt Engineering

Previous efforts have explored methods for obtaining successful natural language prompts. For both open-vocabulary image classification models as well as LLMs, the format of prompts is known to highly affect accuracy [48, 42, 5, 17]. This has led to a large effort to find optimal prompt formats. Proposed methods include crowd-sourcing high performing prompts [1] as well as framing prompts to induce models to give explanations as well as answers [57, 25, 37]. Additional works have proposed learning prompts via gradient based methods [65, 41, 29, 28, 51], retrieval from a database [46], or reformatting/rephrasing existing prompts [24, 46].

Most relevant to this work are a number of methods for designing optimal prompts for zero-shot image classification with open-vocabulary models. These methods learn prompts formats which yield high accuracy for image classification using either supervised [66, 43] or unsupervised [22] methods. However, unlike these prior works this work requires no additional training or labeled data.

5. Conclusion

We demonstrate that leveraging knowledge from an LLM can immediately improve zero-shot accuracy on a variety of image classification tasks, with much less hand-engineering efforts to craft natural language prompts. Furthermore, prompts can be customized to the desired categories, rather than a general template that applies to all categories. Finally, using prompts generated by LLMs lowers the barrier of prior knowledge about the dataset, which is often required when crafting prompt templates.

Querying an LLM for prompt construction is simple, straightforward and as our results suggested, immediately beneficial. The hypothesis that a joint force of LLMs and open vocabulary models would improve zero-shot image classification is thoroughly tested in this work. We hope these findings serve as a useful tool towards understanding and improving zero-shot image classification, and more generally, the consolidation of model capacities and modalities through natural language.

6. Acknowledgements

We thank Mitchell Wortsman, Gabriel Ilharco, Vivek Ramanujan, Aditya Kusupati, Jason Wei, and Ofir Press for helpful discussions and draft feedback. We also thank Samir Yitzhak Gadre and Alex Fang for their useful experimental suggestions. Finally, we thank our reviewers and meta-reviewers for their time and feedback during the peer review process. This work is in part supported by NSF IIS 1652052, IIS 17303166, DARPA N66001-19-2-4031, DARPA W911NF-15-1-0543 and gifts from Allen Institute for Artificial Intelligence, Google, and Apple.

References

[1] Stephen H. Bach, Victor Sanh, Zheng Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M SAIFUL BARI, Thibault Févry, Zaid Alyafeai, Manan Dey, Andrea Santilli, Zhiqing Sun, Srulik Ben-David, Canwen Xu, Gunjan Chhablani, Han Wang, Jason Alan Fries, Maged S. Al-shaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Mike Tian-Jian Jiang, and Alexander M. Rush. Promptsource: An integrated development environment and repository for natural language prompts. *ArXiv*, abs/2202.01279, 2022. 8

[2] Thomas Berg, Jiongxin Liu, Seung Woo Lee, Michelle L Alexander, David W Jacobs, and Peter N Belhumeur. Birdsnap: Large-scale fine-grained visual categorization of birds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2011–2018, 2014. 4

[3] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020. 24

[4] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014. 4

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3, 8, 20, 21

[6] Sebastian Bujwid and Josephine Sullivan. Large-scale zero-shot image classification from rich and diverse textual descriptions. *ArXiv*, abs/2103.09669, 2021. 7

[7] Sebastian Bujwid and Josephine Sullivan. Large-scale zero-shot image classification from rich and diverse textual descriptions. *arXiv preprint arXiv:2103.09669*, 2021. 19

[8] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019. 4

[9] Hugh Chen, Ian C Covert, Scott M Lundberg, and Su-In Lee. Algorithms to estimate Shapley value feature attributions. *arXiv preprint arXiv:2207.07605*, 2022. 6

[10] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017. 4

[11] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2, 4

[12] Ian Covert, Chanwoo Kim, and Su-In Lee. Learning to estimate Shapley values with vision transformers. *arXiv preprint arXiv:2206.05282*, 2022. 6

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2, 3, 4, 22

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 4

[15] Mohamed Elhoseiny, Yizhe Zhu, Han Zhang, and A. Elgammal. Link the head to the ”beak”: Zero shot learning from noisy text description at part precision. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6288–6297, 2017. 7

[16] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004. 4

[17] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*, 2020. 8

[18] Ju He, Shuo Yang, Shaokang Yang, Adam Kortylewski, Xiaoding Yuan, Jie-Neng Chen, Shuai Liu, Cheng Yang, Qihang Yu, and Alan Yuille. Partimagenet: A large, high-quality dataset of parts. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VIII*, pages 128–145. Springer, 2022. 6, 8, 22

[19] Xiangteng He and Yuxin Peng. Fine-grained image classification via combining vision and language. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7332–7340, 2017. 7

[20] Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Juan-Zi Li, and Maosong Sun. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. In *ACL*, 2022. 8

[21] Siteng Huang, Min Zhang, Yachen Kang, and Donglin Wang. Attributes-guided and pure-visual attention alignment for few-shot recognition. In *AAAI*, 2021. 7

[22] Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022. 8

[23] 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

[24] Zhengbao Jiang, Frank F. Xu, J. Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020. 8

[25] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *ArXiv*, abs/2205.11916, 2022. 8

[26] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013. 4

[27] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 4

[28] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *ArXiv*, abs/2104.08691, 2021. 8

[29] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, abs/2101.00190, 2021. 8

[30] Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. Generated knowledge prompting for commonsense reasoning. In *ACL*, 2022. 7

[31] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 2017. 6

[32] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013. 2, 3, 4

[33] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. *arXiv preprint arXiv:2210.07183*, 2022. 5, 8

[34] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 6, 8, 19, 22, 24, 25

[35] Rory Mitchell, Joshua Cooper, Eibe Frank, and Geoffrey Holmes. Sampling permutations for Shapley value estimation. *Journal of Machine Learning Research*, 23(43):1–46, 2022. 6

[36] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008. 4

[37] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. *ArXiv*, abs/2112.00114, 2021. 8

[38] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012. 3, 4

[39] Tzuf Paz-Argaman, Yuval Atzmon, Gal Chechik, and Reut Tsarfaty. Zest: Zero-shot learning from text descriptions using textual similarity and visual summarization. In *FINDINGS*, 2020. 7

[40] Hieu Pham, Zihang Dai, Golnaz Ghiasi, Kenji Kawaguchi, Hanxiao Liu, Adams Wei Yu, Jiahui Yu, Yi-Ting Chen, Minh-Thang Luong, Yonghui Wu, et al. Combined scaling for open-vocabulary image classification. *arXiv preprint arXiv:2111.10050*, 2021. 1, 3

[41] Guanghui Qin and Jas' Eisner. Learning how to ask: Querying lms with mixtures of soft prompts. *ArXiv*, abs/2104.06599, 2021. 8

[42] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 2, 3, 4, 5, 6, 8, 12, 20, 22, 23

[43] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18082–18091, 2022. 8

[44] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pages 5389–5400. PMLR, 2019. 22

[45] Scott E. Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 49–58, 2016. 7

[46] Ohad Rubin, Jonathan Herzig, and Jonathan Berant. Learning to retrieve prompts for in-context learning. In *NAACL*, 2022. 8

[47] Shibani Santurkar, Yann Dubois, Rohan Taori, Percy Liang, and Tatsunori Hashimoto. Is a caption worth a thousand images? a controlled study for representation learning. *arXiv preprint arXiv:2207.07635*, 2022. 7

[48] Timo Schick and Hinrich Schütze. Exploiting cloze-questions for few-shot text classification and natural language inference. In *EACL*, 2021. 8

[49] Lloyd S Shapley et al. A value for n-person games. 1953. 6

[50] Sheng Shen, Chunyuan Li, Xiaowei Hu, Yujia Xie, Jianwei Yang, Pengchuan Zhang, Anna Rohrbach, Zhe Gan, Lijuan

Wang, Lu Yuan, Ce Liu, Kurt Keutzer, Trevor Darrell, and Jianfeng Gao. K-lite: Learning transferable visual models with external knowledge. *ArXiv*, abs/2204.09222, 2022. 7

[51] Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Eliciting knowledge from language models using automatically generated prompts. *ArXiv*, abs/2010.15980, 2020. 8

[52] 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. 4

[53] Yixuan Su, Tian Lan, Yahui Liu, Fangyu Liu, Dani Yogatama, Yan Wang, Lingpeng Kong, and Nigel Collier. Language models can see: Plugging visual controls in text generation. *ArXiv*, abs/2205.02655, 2022. 8

[54] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 23, 24, 25, 26

[55] Ben Wang and Aran Komatsuzaki. Gpt-j-6b: A 6 billion parameter autoregressive language model, 2021. 20

[56] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019. 22

[57] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903, 2022. 8

[58] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019. 20

[59] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7959–7971, 2022. 22

[60] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010. 4

[61] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a bottle: Language model guided concept bottlenecks for interpretable image classification. *arXiv preprint arXiv:2211.11158*, 2022. 8

[62] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. An empirical study of gpt-3 for few-shot knowledge-based vqa. In *AAAI*, 2022. 8

[63] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 1, 3

[64] Youngjae Yu, Jiwan Chung, Heeseung Yun, Jack Hessel, Jae Sung Park, Ximing Lu, Prithviraj Ammanabrolu, Rowan Zellers, Ronan Le Bras, Gunhee Kim, and Yejin Choi. Multimodal knowledge alignment with reinforcement learning. *ArXiv*, abs/2205.12630, 2022. 8

[65] Ningyu Zhang, Luoqiu Li, Xiang Chen, Shumin Deng, Zhen Bi, Chuanqi Tan, Fei Huang, and Huajun Chen. Differentiable prompt makes pre-trained language models better few-shot learners. *ArXiv*, abs/2108.13161, 2021. 8

[66] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, pages 1–12, 2022. 8