Semantically-Prompted Language Models Improve Visual Descriptions

Anonymous ACL submission

Abstract

Language-vision models like CLIP have made significant progress in zero-shot vision tasks, such as zero-shot image classification (ZSIC). However, generating specific and expressive visual descriptions remains a challenge as current methods produce descriptions that lack granularity and are ambiguous. To tackle these challenges, we propose V-GLOSS: Visual Glosses, a novel method that prompts language models with semantic knowledge to produce improved visual descriptions. We demonstrate that V-GLOSS can be used to achieve state-of-the-art results on benchmark ZSIC datasets, such as ImageNet and STL-10. In addition, we introduce a silver dataset with visual descriptions generated by V-GLOSS and demonstrate its utility for language-vision tasks.

1 Introduction

Language-vision models (Radford et al., 2021; Jia et al., 2021) have made significant progress in zero-shot vision tasks. However, we hypothesize that their accuracy is limited by a lack of visual concept descriptions that are both expressive and specific, that is, glosses that describe what images depicting a concept look like. In this work, we investigate this hypothesis by creating and testing a novel method for producing visual descriptions.

Improving visual descriptions is crucial for enhancing system performance in zero-shot vision tasks. Such descriptions facilitate the creation of more useful representations. Additionally, being able to describe a concept in terms of its appearance is essential for developing more robust and adaptable methods incorporating diverse visual information across various domains, without the need for extensive re-training.

Existing approaches to generating visual descriptions, such as those employed by CLIP (Radford et al., 2021) and CuPL (Pratt et al., 2022), involve directly plugging class labels into fixed templates (e.g., *a photo of* X), or using large language models such as InstructGPT (Ouyang et al., 2022) to generate descriptions based on class labels (e.g., *what does* X *look like?*). These methods suffer from two main issues: class granularity and label ambiguity. Class granularity refers to the difficulty in distinguishing between visually similar classes, such as ALLIGATOR and CROCODILE. Label ambiguity is caused by using polysemous words as labels for distinct concepts. For example, CRANE can refer to either a bird or a construction machine. These issues limit the performance of existing models (Radford et al., 2021).

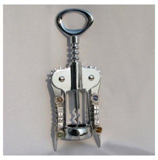
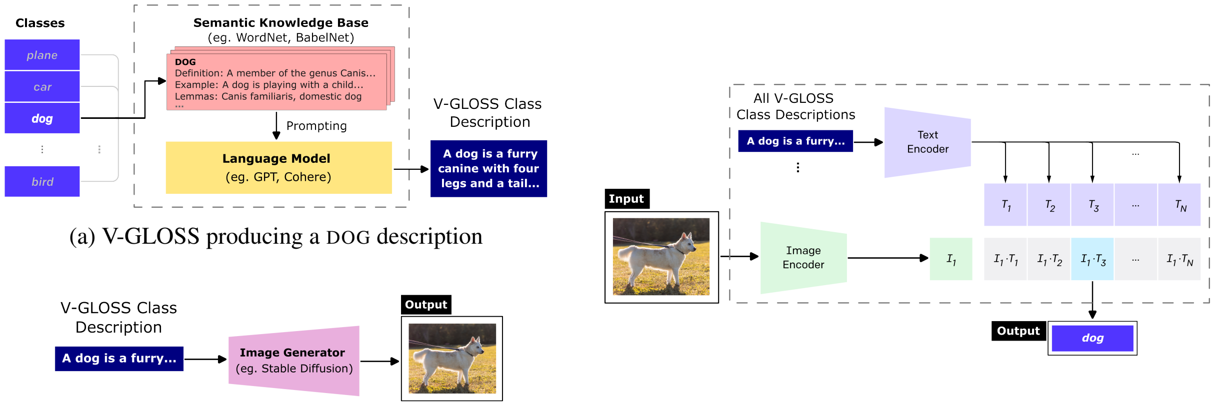
Class / Concept	WordNet Gloss	V-GLOSS (Ours)
CORKSCREW 	A bottle opener that pulls corks.	A **tool** with a **spiral blade** that is used to remove corks from bottles.
BRAMBLING 	Eurasian finch.	A **small brown** bird with a **black head** and a **white patch** on its chest.
BROCCOLI 	Branched green undeveloped flower heads.	A **green vegetable** with a **thick stalk** and **florets** that grow in a **dense** head.

Table 1: A qualitative comparison between WordNet concept glosses and V-GLOSS (Silver) class descriptions for some ImageNet classes. Our method describes what a class *looks like*, instead of what it *does* or *is*.

To address these challenges, we introduce V-GLOSS, a novel method that leverages language models (LMs) and semantic knowledge bases (SKBs) to generate improved visual descrip-



(a) V-GLOSS producing a DOG description

(b) V-GLOSS for ZSCIG: generating a DOG image

(c) V-GLOSS for ZSIC: classifying a test image

Figure 1: For the DOG class, we depict (a) V-GLOSS's architecture (Section 4.2.1), along with adaptations: (b) ZSIC (Section 5.4.1) and (c) ZSCIG (Section 5.4.1)

tions – visual glosses. Table 1 shows some examples. By combining structured semantic information from SKBs such as WordNet (Miller, 1998), with a contrastive algorithm to distinguish similar classes, V-GLOSS is designed to mitigate the dual issues of granularity and ambiguity.

Our results demonstrate the effectiveness of V-GLOSS in improving the performance of ZSIC systems. We achieve state-of-the-art (SOTA) results on benchmark datasets such as ImageNet (Deng et al., 2009), CIFAR-10, and CIFAR-100 (Krizhevsky et al., 2009) in the zero-shot setting, and STL-10 (Coates et al., 2011) in both the zero-shot and supervised settings. Additionally, we introduce V-GLOSS Silver, a silver dataset constructed by V-GLOSS, consisting of a visual gloss for each ImageNet class. We show that V-GLOSS Silver is useful for language-vision tasks such as ZSIC and ZSCIG, comparing favorably to WordNet glosses.

2 Tasks

Our main task is to generate a description for a given class or concept. For example, if an image classification dataset has the class DOG, we aim to produce a description such as *"A dog is a furry, four-legged canine..."* We consider such a description to be a specific kind of gloss.

We use two downstream tasks to compare methods of generating class descriptions: zero-shot image classification (ZSIC), and zero-shot class-conditional image generation (ZSCIG).

In ZSIC, the goal is to classify an image based on a set of classes, without having seen any labeled images belonging to those classes. The set of classes depends on the dataset. For example, given an image depicting a dog, we aim to predict the class DOG.

In ZSCIG, the goal is to generate an image that corresponds to a specific class, again without having seen any labeled examples. For example, given a class DOG, we aim to generate an image of a dog.

In short, ZSIC is the task of classifying a given image, while ZSCIG is the task of generating an image given a class. Both involve classes and images. Visual descriptions of classes provide useful information which can facilitate both tasks, by making it easier to either recognize or generate images of each class. Therefore, by developing a novel method to improve the generation of such descriptions, we hypothesize that performance on ZSIC and ZSCIG can be improved.

3 Related Work

Language Models The advent of transformer-based language models has revolutionized many natural language processing tasks (Radford et al., 2018; Devlin et al., 2018; Radford et al., 2019; Brown et al., 2020; Black et al., 2022; Ouyang et al., 2022). As these models are scaled up by their number of parameters and quantity of training data, they exhibit emergent abilities such as few-shot and zero-shot learning (Wei et al., 2022).

Language-Vision Models Significant strides have been made in the field of language-vision models such as CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021). These models apply contrastive pre-training approaches on large image-text datasets, leading to improved representation learning for both text and images and enhanced performance on several multi-modal tasks (Mokady et al., 2021; Song et al., 2022). Further advancements have been achieved by scaling up pre-training and incorporating auxiliary training objectives (Pham et al., 2021; Yu et al., 2022).

Producing Descriptions & Prompting The generation of descriptions and prompting has been explored in various studies. Radford et al. (2021) introduced the template ensemble (TE) method, which uses a custom set of class labels and a fixed set of templates. Each label is inserted into these templates, and the completed templates for each class are aggregated into a single representation of the class. The CuPL method (Pratt et al., 2022) utilizes InstructGPT (Brown et al., 2020; Ouyang et al., 2022) to generate descriptions for ImageNet classes. Both TE and CuPL can be used for zero-shot image classification. Hao et al. (2022) fine-tuned GPT models (Radford et al., 2018, 2019) to rephrase image-generation prompts, resulting in improved images. (Zhou et al., 2022) learned soft prompts that improve performance, but are intractable to humans. In this work, we prompt language models with semantic knowledge to generate visual descriptions.

4 Method

We begin by describing how we map classes to concepts in a semantic knowledge base (SKB), in order to leverage the concept-specific information the SKB contains. We then introduce of our novel method V-GLOSS, which has two variants, *normal* and *contrastive*. We conclude by describing the construction of V-GLOSS Silver, a set of class descriptions produced using V-GLOSS.

4.1 Mapping Classes to WordNet Synsets

The ImageNet classes are already mapped to WordNet synsets by the dataset's creators. For the other datasets, we employ a heuristic that starts by mapping each class to the most frequent sense of the class label, as determined by WordNet[1]. For CIFAR-10 and STL-10, this heuristic is sufficient. For CIFAR-100, we manually re-map 18 classes. For instance, we needed to re-map RAY from *light* to *sea creature*, as the *light* sense is the most frequent according to WordNet, but the RAY images in the dataset depict sea creatures.

[1] https://www.nltk.org/

```
What does a platypus look like?
A platypus looks like a beaver with a duck's bill
```

(a) CuPL Pratt et al. (2022)

```
...
Concept name: eagle
Hypernyms: bird or prey
Hyponyms: bald eagle, eaglet, golden eagle, harpy
Gloss: any of various large keen-sighted diurnal birds
of prey noted for their broad wings and strong...
Unique and expressive visual description: Eagles are
large birds of prey with dark brown bodies and wings...
...
Concept name: platypus
Hypernyms: duckbill, duckbilled platypus, ...
Hyponyms: egg-laying mammal
Gloss: small densely furred aquatic monotreme of
Australia and Tasmania having a broad bill...
Unique and expressive visual description:

Platypuses are water-dwelling mammals that have
broad duck-like bills and hind legs with a foot web that
has an intricate web of keratinised spongy hairs
```

(b) V-GLOSS

Figure 2: Class descriptions for PLATYPUS generated by two different methods that use LMs. Input prompts, output descriptions, and **plugged values** are shown.

4.2 V-GLOSS

We discuss the two variants of V-GLOSS below, *normal* and *contrastive*. In both, for each class, we produce multiple descriptions resulting in an ensemble.

4.2.1 *Normal* V-GLOSS

We generate normal descriptions via in-context learning with an LM, beginning by providing the LM with a description of the task to be performed, followed by multiple input-output examples. The examples are fixed, involving the concepts EAGLE, BAT (animal), BAT (baseball), and TELEVISION. We selected these to expose the model to ambiguous class labels (*bat*), a natural object (*eagle*), and an artificial object via (*television*). For each class, we obtain from WordNet the hypernyms, hyponyms, usage examples, synonyms, and glosses of the sense to which the class is mapped, and provide this to the LM. Figure 2b shows a session with the LM, beginning with the example of *eagle*, with output generated for the class *platypus*. Table 1 compares our descriptions to WordNet glosses.

4.2.2 *Contrastive* V-GLOSS

During development, we observed that many errors were caused by false positives involving visually similar classes. For example, the classes

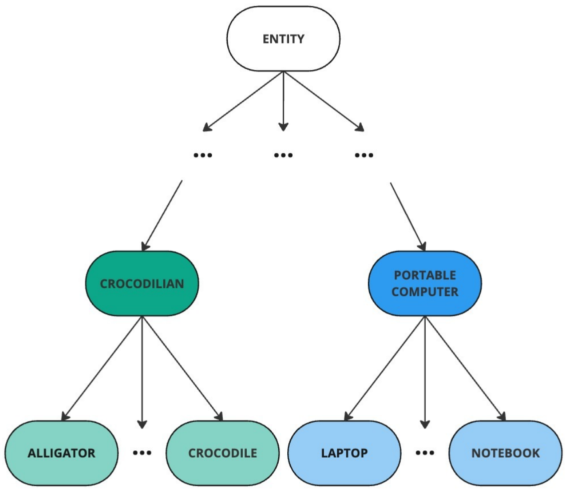


Figure 3: A sample of WordNet hypernym hierarchy. For *contrastive* prompting, we only distinguish classes that are semantically similar to the target class, like ALLIGATOR to CROCODILE.

Class / Concept	*Normal*	*Contrastive*
ALLIGATOR 	A large reptile with a long snout, a broad head, and a long tail.	A large, **dark-colored** reptile with a **rounded snout**, found in **freshwater**.
CROCODILE 	A reptile with a broad, flat snout, a long tail, and a long, pointed snout.	A **grayish-green** reptile with a **v-shaped snout**, found in **brackish** or **saltwater**.

Table 2: Two similar classes with **key differences** between their *normal* and *contrastive* descriptions.

CROCODILE for ALLIGATOR refer to similar-looking animals, and are often confused for one another. The contrastive variant of V-GLOSS is designed to address this by using semantic similarity between classes as a heuristic to estimate visual similarity. For each class, we search for other classes that are semantically similar, and if any are found, we add a negative instruction to the LM prompt, e.g. we generate a description for an ALLIGATOR *but not* a CROCODILE, using the same prompt structure as for normal V-GLOSS.

We create a similarity matrix M as follows:

$$M_{i,j} = Sim(S[i], S[j]) \quad (1)$$

$Sim(s_1, s_2)$ is the Wu-Palmer path-similarity function (Wu and Palmer, 1994) comparing synsets s_1 and s_2; this similarity function uses the path between two concepts in the WordNet hypernym hierarchy (Figure 3) to measure semantic relatedness. S is the set of all classes in a dataset, $\mathcal{D}$, and i and j are indices ranging from 1 to $|S|$. Concisely, Equation 1 defines a similarity matrix containing similarity scores between all classes in a dataset. M is one of the inputs to our contrastive V-GLOSS variant, shown in Algorithm 1.

In Algorithm 1, λ is a threshold for minimum similarity. We only generate contrastive descriptions when classes have a similarity that exceeds or is equal to λ. N indicates the maximum number of classes to generate contrastive descriptions for. k is the number of distinct descriptions to generate for a class pair. LM_c takes in the $target$ class, a neighbor class, and k, then prompts the LM to generate k descriptions that distinguish the $target$ and neighbor classes. In summary, for each class, Algorithm 1 identifies the classes most similar to it, excluding itself, and generates descriptions that distinguish them. Table 2 compares the normal and contrastive descriptions for ALLIGATOR and CROCODILE; note that distinguishing features of the two classes are included in the LM's output. Table 3 shows examples of classes with high false positive rates, and the classes they are contrasted with.

Algorithm 1 Generate Contrastive Descriptions: We generate contrastive descriptions to help distinguish the most similar classes.

Require: M: Equation 1 result
Require: λ, N, k: Hyperparameters
Require: S: All classes in dataset, $\mathcal{D}$
Require: LM_c: LM prompted contrastively

```
G ← empty |S|-list for class descriptions
for i ← 0 to |S| − 1 do
    target ← S[i]
    S* ← top N classes : λ ≤ M_{i,*} ≤ 1
    for s* in S* do
        samples ← LM_c(target, s*, k)
        G[i].insert(samples)
return G
```

5 Evaluation

Toward evaluating V-GLOSS, we describe our datasets, evaluation metrics, baselines, previous methods, and experiments.

Class	False Positives	Contrastives
African Elephant	Tusker (44), Asian Elephant (6)	Tusker, Asian Elephant
Notebook	Laptop (22), Desktop (10), Space bar (2)	Laptop, Desktop, Space bar

Table 3: False positives and their counts vs. classes selected by the contrastive algorithm (see Equation 1 and Algorithm 1). Hits and misses are shown.

5.1 Datasets

We evaluate our method on the test splits of four widely used benchmark datasets: ImageNet (Deng et al., 2009) consists of 50,000 images equally distributed across 1,000 classes, and serves as our *primary* benchmark. CIFAR-10 and CIFAR-100 (Krizhevsky et al., 2009) both comprise 10,000 test samples across 10 and 100 classes, respectively. Finally, STL-10 (Coates et al., 2011) comprises 100,000 test samples and is designed for unsupervised learning. For CIFAR-10, CIFAR-100, and STL-10, which are not pre-mapped to WordNet, we employ the two-step process detailed in Section 4.1 to map each class to a WordNet synset.

Experiment 1 (Section 5.4) involves ImageNet alone and covers both the ZSCIG and ZSIC tasks. In contrast, Experiment 2 (Section 5.5), which is our main experiment, tests the impact of various class description methods on the ZSIC task and uses all datasets. In Experiment 2, we allow methods to use ensembles of descriptions of each class, while in Experiment 1, we experiment with only a single description.

The datasets we selected to evaluate the following properties of V-GLOSS:

1. **Performance on benchmark datasets with varying numbers of classes.** Each dataset has its own set of classes, ranging from ImageNet with 1,000 classes, to CIFAR-100 with 100 classes, to CIFAR-10 and STL-10, each with 10 classes.

2. **Ability to represent diverse concepts at varying levels of granularity.** The datasets we use contain a wide range of concepts across various domains, rather than those targeting specific subareas such as pets (Parkhi et al., 2012), foods (Bossard et al., 2014), cars (Krause et al., 2013), scenes (Xiao et al., 2010), or airplanes (Maji et al., 2013).

5.2 Evaluation Metrics

Top-1 Accuracy In ZSIC, this metric is the frequency with which the model's top prediction for an image matches the gold label.

Fréchet Inception Distance (FID) For ZSCIG, FID (Heusel et al., 2017) quantifies the divergence between ground truth and generated images, with lower scores signifying a better ability to produce images similar to the ground truth.

Inception Score Also for ZSCIG, the inception score (Salimans et al., 2016) uses an Inception model's (Szegedy et al., 2015) output probability distribution to assess the diversity and realism of generated images, with higher scores indicating more diverse and convincing images. Unlike the above metrics, this does not require ground-truth images for comparison.

5.3 Baseline & Previous Methods

In this section, we describe the methods to which we compare V-GLOSS. For methods that produce ensembles of class descriptions (i.e. multiple descriptions per class), a single representation of the class is obtained by averaging individual representations for each description.

First, the **1-Template baseline** inserts a class label into a *single* specific template. For example, given the class DOG, the baseline produces *"A photo of a dog."*

Template Ensemble (Radford et al., 2021) generates an ensemble of descriptions for a class by inserting the class label into each of a set of templates. For example, some descriptions for DOG are: *"A photo of a dog."*, *"A blurry photo of a dog."*, and *"An origami dog."* This method uses a modified list of class labels[2] designed to reduce ambiguity.

CuPL (Pratt et al., 2022) also generates an ensemble of descriptions for each class. The descriptions are generated by prompting a LLM, InstructGPT (Ouyang et al., 2022), with questions such as: *"What does a dog look like?"* and *"Describe an image of a dog from the internet."* CuPL uses the same class labels as Template Ensemble.

The authors of CuPL also combined their method with Template Ensemble. The resulting method, **CuPL + Template Ensemble**, combines the class descriptions from both methods.

[2] https://github.com/anishathalye/imagenet-simple-labels

5.4 Experiment 1: V-GLOSS Silver

This experiment evaluates V-GLOSS's ability to generate a *single* description for each class, without relying on ensembling. We then evaluate the V-GLOSS description of each class against its WordNet gloss.

To construct this set of class descriptions, which we view as a silver dataset of such descriptions, we generate a *single*, *normal* description for each ImageNet class via greedy decoding. We generate only *normal* descriptions because they outperform *contrastive* ones when only a single description is used. We call the resulting dataset *V-GLOSS Silver*.

We extrinsically evaluate V-GLOSS Silver by using it for the ZSIC and ZSCIG tasks, and comparing the results to those achieved using the 1-Template baseline, and WordNet glosses. We do not compare V-GLOSS Silver to CuPL or other previous methods which do not produce a single description for each class.

5.4.1 Technical Details

ZSIC We employ CLIP (Radford et al., 2021), which comprises an image encoder and a text encoder, as the ZSIC backbone model. Our procedure consists of three steps: First, we use the CLIP text encoder to create an aggregate representation for each class based on its description(s). Then, at test time, we employ the CLIP image encoder to generate a representation of the input image. Finally, we predict the class which maximizes the cosine similarity between the representation of its description(s), and the image representation (see Figure 1c). We evaluate the predictions using top-1 accuracy.

ZSCIG For ZSCIG (see Figure 1b), we condition Stable Diffusion (Rombach et al., 2022) on each class description before generating an image. We use a guidance scale of 7.5 and run 50 diffusion steps. We evaluate the generated images using Inception and FID scores.

5.4.2 Results

The results of Experiment 1 are shown in Table 4. Based on our extrinsic evaluation in the ZSIC and ZSCIG tasks, *V-GLOSS Silver* descriptions yield better performance compared to baseline and WordNet Glosses. On ZSIC, we improve accuracy by 1.3%; on ZSCIG, we improve Inception and FID scores by 9.9 and 5.7, respectively. This demonstrates the effectiveness and utility of V-GLOSS: our visual descriptions yield better results on ZSIC and ZSCIG.

	ZSIC	ZSCIG	
	Accuracy ↑	Inception ↑	FID ↓
Baseline (1-Template)	71.0	99.7	25.7
WordNet Glosses	44.7	58.5	30.0
V-GLOSS Silver	**72.3**	**109.6**	**20.0**

Table 4: Extrinsic evaluation on the tasks of ZSIC and ZSCIG. ↓ means that lower is better.

5.4.3 Analysis

V-GLOSS Silver descriptions are considerably more detailed, more expressive, and better visually grounded than their WordNet gloss counterparts (see Figure 1). Specifically, we observe that V-GLOSS descriptions make greater use of descriptive words and phrases, e.g. *spiral, brown, green, thick, small*, etc.

5.5 Experiment 2: ZSIC

Our second experiment assesses the effectiveness of V-GLOSS descriptions in facilitating ZSIC. The details for the ZSIC pipeline are largely similar to those described in Experiment 1 (Section 5.4), except that we generate an ensemble of descriptions per class, as opposed to only one description. We also experiment with two image encoder variants: ViT (Dosovitskiy et al., 2020) and RN50 (He et al., 2016). For all baselines and methods (Section 5.3, Section 4.2.1), we follow the same evaluation procedure after generating class descriptions.

5.5.1 Technical Details

We generate class descriptions using the 6.1B-parameter Cohere LM[3]. We choose Cohere over alternatives due to its extensive cost-free availability, reducing the cost of our experiments. Cohere has comparable performance to the similarly-sized InstructGPT (Brown et al., 2020; Ouyang et al., 2022) variant, as demonstrated by Liang et al. (2022) across various benchmarks. Therefore, we do not gain any advantage by using Cohere instead of InstructGPT.

When generating class descriptions with *normal* V-GLOSS, we use a temperature of 2.5 to produce an ensemble of 50 descriptions per class. When generating *contrastively*, we use a temperature of 1.5 to generate an ensemble of 20 descrip-

[3] https://docs.cohere.com/docs/models

Method	Model	Accuracy (%) on Datasets				# LM Parameters
		ImageNet	CIFAR-100	CIFAR-10	STL-10	
Baseline (1-Template)	ViT RN50	72.4 68.7	77.3 57.7	95.2 81.0	99.5 98.4	0
Template Ensemble	ViT RN50	76.2 73.2	77.9 61.3	96.2 86.8	99.4 98.3	0
CuPL	ViT	76.7	-	-	-	175B
CuPL + Template Ensemble	ViT RN50	77.6 75.1	- -	- -	- -	175B
V-GLOSS (*Normal-Only*)	ViT RN50	77.3 73.3	77.5 63.5	95.6 86.8	99.4 98.3	6.1B
V-GLOSS (*Normal + Contrastive*)	ViT RN50	**78.5** 74.5	**78.2** 64.6	**97.0** 87.8	**99.6** 98.8	6.1B

Table 5: Top-1 accuracy on ZSIC. ViT and RN are Transformer- and ResNet-based CLIP variants.

tions per class. Like Pratt et al. (2022), we observe that performance saturates around 50 descriptions for *normal* V-GLOSS, but we also observe saturation at around 20 descriptions for *contrastive* V-GLOSS. Based on tuning on development data, we set $N = 5$, $\lambda = 0.5$, and $k = 4$ (see Algorithm 1). In total, we obtain 70 class descriptions. During generation, we set the maximum number of tokens to 35, but also terminate generation when the *boundary parameter* or *newline* token is reached.

5.5.2 Results

The results of Experiment 2 are shown in Table 5. V-GLOSS yields better accuracy than the baseline by an average of 3.60% overall (2.22% with ViT and 4.98% with RN50). V-GLOSS also outperforms Template Ensemble and CuPL + Template Ensemble, by 1.21% and 0.15% respectively. This improvement is especially notable since the top 15 results on the ImageNet benchmark differ by less than 1% accuracy.[4]

In addition, we make the following observations. (1) V-GLOSS (*Normal + Contrastive*) surpasses V-GLOSS (*Normal-Only*), by an average of 0.91% accuracy. (2) We outperform *CuPL + Template Ensemble* using an LLM with 28.7x fewer parameters. (3) The RN backbone (He et al., 2016), which is generally less capable than ViT (Dosovitskiy et al., 2020), sees a more significant benefit from the *V-GLOSS* method, on average 3.8%. (4) For STL-10, V-GLOSS matches the top-performing supervised system (Gesmundo, 2022) with a score of 99.6%.

We also note that the *contrastive* component is more helpful on the larger datasets: CIFAR-100 and ImageNet, which have more opportunities for mutual ambiguity between different classes, than on the smaller ones: CIFAR-10 and STL-10. Concretely, this improvement is 1.05%, on average. Later, in Section 6, we discuss these results and their implications more extensively.

5.5.3 Analysis

In Section 1, we pointed out several problems in previous methods. Here, we carefully analyze how our V-GLOSS method addresses these issues.

Label Ambiguity: Without adequate context, text models may fail to grasp the intended meaning of a polysemous word. *Crane* is a polysemous word, and ImageNet (Deng et al., 2009) has two classes that refer to different senses of the word: *construction machine* and *wading bird*, but use the same label. Thus, in *1-Template*, for example, both classes have the same description. This point highlights an important benefit of linking classes to WordNet, which resolves such ambiguity. Empirically, when compared with a ViT backbone to the *Lex Baseline*, our accuracy on CRANE (machine) and CRANE (bird) increase from 0% and 46% to 76% and 78%, respectively.

Relationship Between Performance & Context: When comparing the baselines to the other methods, we observe that accuracy generally improves as the amount of surrounding context increases. On one hand, if a sentence consists of *"my crane."* alone, the sense of *crane* is unclear. On the other, if the sentence is *"my construction crane,"* the meaning of *crane* becomes clear. We see that providing additional context helps to disambiguate words. When a description provides more useful context,

[4] https://paperswithcode.com/sota/image-classification-on-imagenet

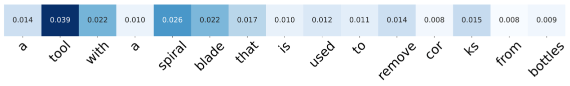


Figure 4: V-GLOSS Attention Map

Figure 5: WordNet Gloss Attention Map

models can form better representations of specific classes. By comparing V-GLOSS to the baselines (see Table 5), we can observe that the benefits of additional context extend to the vision-language setting. Concretely, providing visually-grounded context in the description improves performance.

Class Granularity: We consider pairs of classes that are similar enough to be mistaken, such as ALLIGATOR and CROCODILE. In WordNet, relationships between synsets are modeled through *is-a* (hyponymy-hypernymy) and *part-of* (meronymy-holonymy) relationships. For example, CROCODILIAN is a hypernym of both ALLIGATOR and CROCODILE, while only ALLIGATOR is a holonym of SNOUT, since alligators have snouts while crocodiles do not. Using our contrastive algorithm, we generate descriptions that highlight how images of a CROCODILE should depict a greener animal with a rounded snout. Empirically, using ViT, the average accuracy of V-GLOSS across these two classes jumps from 36% to 68% when contrastive glosses are used. This improvement highlights the effectiveness of our contrastive V-GLOSS variant in reducing false positives between visually similar classes.

Attention To Relevant Context: We analyze the model's attention maps to better understand V-GLOSS's impact. Figure 4 shows the attention map for V-GLOSS (see Table 1 for descriptions), indicating effective utilization of visually-relevant context. Conversely, Figure 5 shows the attention map for the WordNet glosses (baseline), where the attention score on *bottle* is 3.5x higher, implying less distraction in V-GLOSS. These maps demonstrate success in steering the model's attention toward relevant context, thus improving classification accuracy across different classes and descriptions.

6 Discussion

When looking at our results, a pertinent question arises: Why does an SKB, such as WordNet, help us do better on tasks related to vision? In this section, we formulate two insights on how the synergy between SKBs and LMs supports our improvements.

Insight #1: SKBs represent concepts precisely When LMs are prompted with better information, they produce better output (Borgeaud et al., 2022). WordNet provides a precise representation of a class and its relationship to other classes, leaving minimal room for ambiguity. Afterward, we can prompt an LM with this precise information to produce unambiguous and high-quality class descriptions.

Insight #2: Semantic similarity is a useful proxy for visual similarity WordNet models lexical semantics as a graph (see Figure 3), with synsets as nodes and *is-a* relationships as directed edges. The distance between different nodes reflects the level of semantic similarity and is by extension an indicator of the level of visual similarity between synsets. ALLIGATOR and CROCODILE are semantically similar because they are both kinds of CROCODILIAN, but they are visually similar as well (see Table 2). Semantic similarity informs what classes we distinguish with our contrastive descriptions and why they work (see Table 3). This is because semantic and visual similarity are highly correlated.

7 Conclusion

This study concentrates on generating visual class descriptions for ZSIC and ZSCIG tasks. We utilize a unique method that merges Semantic Knowledge Bases (SKBs) and Language Models (LMs) to create high-quality descriptions. Our findings reveal that the semantic information from SKBs can condition an LM to generate accurate, expressive, and visually grounded descriptions. Furthermore, we observe that LMs, although pre-trained solely on text, contain latent knowledge about the visual properties of concepts. This knowledge can be harnessed using our novel V-GLOSS method, thus improving the accuracy of zero-shot image classification and generation models. This underscores the strong relationship between language and vision, suggesting potential for LMs in future multi-modal tasks.

Limitations

The dataset must be mapped to an SKB. As described earlier, mapping the dataset to WordNet, although a one-time step, is not fully automatic. In future work, we look to fully automate this step, possibly by selecting a synset based on the similarity between sample class images and potential senses of the class label.

We are limited in terms of language, dataset class count, and our SKB's size. First, our English-focused stance may prove a limiting factor in our method being applied to ZSIC or ZSCIG tasks based in other languages. Some classes are strongly related to non-English languages.

Second, our largest evaluation dataset, ImageNet (Deng et al., 2009), has 1,000 classes, representing just 0.64% coverage of WordNet. We look forward to evaluating our methods on a larger ImageNet set: ImageNet-21k, which would cover 14.06% of WordNet.

Third, although our method can be applied to BabelNet (Navigli and Ponzetto, 2012), which has over 1.5 billion synsets, we focus on WordNet, which has 155,287. We look to explore alternative SKBs such as BabelNet, or non-English wordnets, both of which offer the benefit of being multilingual.

Ethics Statement

In normal use, we discover no direct ethical issues with our method. Note, however, that we may inherit ethical problems from the components used by our method. Both CLIP (Agarwal et al., 2021) and LMs (Liang et al., 2021) have independently been shown to exhibit some level of bias. Also, semantic resources such as WordNet (Miller, 1995) tend to focus on formalized concepts. This poses a problem if our method's use concerns people on the fringes of society.

We noted earlier that our method is mostly English-focused. This could be a source of bias if our method is applied in a multilingual context. We ask that people do not apply our method to real-world problems where multilingual knowledge is required. There is also the issue of semantic resources for low-resource languages not being extensive enough (Magueresse et al., 2020).

References

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

Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, et al. 2022. Gpt-neox-20b: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pages 2206–2240. PMLR.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 446–461. Springer.

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

Adam Coates, Andrew Ng, and Honglak Lee. 2011. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings.

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

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

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

Andrea Gesmundo. 2022. A continual development methodology for large-scale multitask dynamic ml systems. *arXiv preprint arXiv:2209.07326*.

Yaru Hao, Zewen Chi, Li Dong, and Furu Wei. 2022. Optimizing prompts for text-to-image generation. *arXiv preprint arXiv:2212.09611*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

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

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

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

Paul Pu Liang, Chiyu Wu, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2021. Towards understanding and mitigating social biases in language models. In *International Conference on Machine Learning*, pages 6565–6576. PMLR.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Alexandre Magueresse, Vincent Carles, and Evan Heetderks. 2020. Low-resource languages: A review of past work and future challenges. *arXiv preprint arXiv:2006.07264*.

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

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

George A Miller. 1998. *WordNet: An electronic lexical database*. MIT press.

Ron Mokady, Amir Hertz, and Amit H Bermano. 2021. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*.

Roberto Navigli and Simone Paolo Ponzetto. 2012. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial intelligence*, 193:217–250.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

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

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. 2021. Combined scaling for open-vocabulary image classification. *arXiv preprint arXiv: 2111.10050*.

Sarah Pratt, Rosanne Liu, and Ali Farhadi. 2022. What does a platypus look like? generating customized prompts for zero-shot image classification. *arXiv preprint arXiv:2209.03320*.

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

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

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

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans. *Advances in neural information processing systems*, 29.

Haoyu Song, Li Dong, Wei-Nan Zhang, Ting Liu, and Furu Wei. 2022. Clip models are few-shot learners: Empirical studies on vqa and visual entailment. *arXiv preprint arXiv:2203.07190*.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich.

2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Zhibiao Wu and Martha Palmer. 1994. Verb semantics and lexical selection. *arXiv preprint cmp-lg/9406033*.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. 2010. 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.

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

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