EXPLORING WEAK-TO-STRONG GENERALIZATION FOR CLIP-BASED CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Aligning large-scale commercial models with user intent is crucial to preventing harmful outputs. Current methods rely on human supervision but become impractical as model complexity increases. When models surpass human knowledge, providing accurate feedback becomes challenging and inefficient. A novel solution proposed recently is using a weaker model to supervise a stronger model. This concept leverages the ability of weaker models to perform evaluations, thereby reducing the workload on human supervisors. Previous work has shown the effectiveness of weak-to-strong generalization in the context of language-only models. Extending this concept to vision-language models leverages these insights, adapting the proven benefits to a multi-modal context. In our study, we explore weak-to-strong generalization for CLIP-based classification. We propose a method, *class prototype learning* (CPL), which aims to enhance the classification capabilities of the CLIP model, by learning more representative prototypes for each category. Our findings indicate that despite the simple loss function under weak supervision, CPL yields robust results. Our experiments are conducted on challenging datasets to evaluate our method. Extensive experiments show that our method is effective, achieving a 3.67% improvement over baseline methods.

1 INTRODUCTION

Large language models (LLMs), such as GPT 4o (Achiam et al., 2023), Claude 3 (Anthropic, 2024) and Gemini 1.5 (Reid et al., 2024), have made significant strides in enhancing performance across a spectrum of natural language processing tasks. However, despite their successes, ensuring that these models align with human expectations and intentions remains a formidable challenge (Burns et al., 2023). Increasing the size of language models does not necessarily improve their ability to follow user intent, as they can still produce untruthful, toxic, or unhelpful outputs, indicating a lack of alignment with their users (Ouyang et al., 2022). Alignment with user intent is crucial for deploying these models effectively in practice (Bai et al., 2022). Traditional alignment techniques often rely heavily on human supervision (Christiano et al., 2017; Stiennon et al., 2020; Ouyang et al., 2022; Glaese et al., 2022; Bai et al., 2022), requiring evaluators to provide feedback on model outputs. However, as the complexity and intricacy of model outputs increase, the feasibility and scalability of this approach diminishes (Burns et al., 2023). As a result, there is a need to effectively align LLMs with human values without overly burdening human evaluators.

Burns et al. (2023) explored a novel approach known as weak-to-strong generalization to address the challenge of aligning strong models with human feedback. This strategy leverages a weaker model to supervise a more robust one, presenting a promising method to enhance model alignment. The study by Burns et al. (2023) demonstrates the effectiveness of this weak-to-strong learning approach, where finetuning strong models with knowledge generated by their weaker counterparts consistently improves performance. For example, in natural language processing tasks, finetuning GPT-4 with supervision from a GPT-2-level model significantly enhances GPT-4's performance. This approach highlights the viability of weak-to-strong learning as a solution for better model alignment, demonstrating that even weaker models can provide valuable guidance for improving stronger models. While the technique proves effective, applying it to Vision-Language Models (VLMs) is far from straightforward. VLMs face unique challenges in aligning complex multimodal tasks, making it essential to thoroughly explore the method's applicability and limitations in this context. Unlike in natural language tasks, where text-based guidance can be more straightforward, VLMs must

align both visual and textual information, making supervision from weaker models significantly more challenging. The complexity of managing two distinct modalities introduces difficulties in ensuring coherent feedback across image and text domains, necessitating a more nuanced approach when adapting weak-to-strong generalization to VLMs. Our goal is to rigorously investigate the weak-to-strong paradigm within VLMs, as this problem extends beyond a mere adaptation of previous work.

In this study, we explore weak-to-strong generalization for CLIP-based classification, recognizing it as a crucial starting point in VLMs. Existing VLMs (Radford et al., 2021; Jia et al., 2021) take classification as the basic task to evaluate the alignment of images and texts. Also in our task setting, classification makes it easier for us to design simulation experiments and establish a benchmark. In this context, we introduce a method called *class prototype learning* (CPL). CPL involves generating class prototypes that encapsulate the characteristics of each class using weak supervision. This method effectively mitigates the false signals typically generated by weak supervision, thereby showcasing superior performance. Moreover, when compared to conventional methods for adapting VLMs to downstream tasks, such as prompt tuning (Zhou et al., 2022b; Jia et al., 2022), our CPL approach proves to be more efficient. This efficiency arises from the fact that CPL eliminates the need for employing a text encoder during the fine-tuning phase. By streamlining the adaptation process, CPL offers a more resource-effective solution while maintaining high-performance levels.

We conduct extensive experiments to evaluate the performance of the proposed method, CPL, using the DomainNet dataset (Peng et al., 2019), which includes six diverse visual domains. The dataset is divided into training and test sets, and the experiment involves training weak models, generating weak supervision sets, fine-tuning strong models with weak supervision, and comparing the results to strong model training with ground truth labels. Various baselines are used for comparison. Our results show that the CPL achieves the highest average accuracy across all domains, significantly outperforming other methods. In particular, CPL shows substantial improvements for challenging domains like Infograph and handles domain-specific features effectively, despite CLIP's lower zero-shot performance in domains like QuickDraw. This illustrates the robustness and effectiveness of CPL in weak-to-strong generalization scenarios.

We summarize the main contributions of our work:

(i) **Exploring weak-to-strong generalization for CLIP-based classification**: Previous work (Burns et al., 2023; Guo et al., 2024) has shown the effectiveness of weak-to-strong generalization in LLMs. Extending this concept to VLMs leverages these insights, adapting the proven benefits to a multi-modal context.

(ii) **Proposing CPL**: We present a method that effectively leverages class prototype representations through weak supervision to enhance the classification performance of VLMs, such as CLIP (Radford et al., 2021).

(iii) **Conducting simulation experiments**: We design a simulation experiment within the VLMs framework based on DomainNet (Peng et al., 2019) to study this problem, and establish a benchmark in this context. Our experiment resulted in a 3.67% improvement over baseline methods.

2 Related Work

Vision-language models. VLMs integrate visual and textual information, enabling a multifaceted understanding and interaction with multimodal content. CLIP (Radford et al., 2021) exemplifies this approach, leveraging contrastive learning to align images with textual descriptions effectively. This model demonstrates robust zero-shot capabilities, where it can recognize images or concepts it was not explicitly trained on. The effectiveness of CLIP and similar models, such as ALIGN (Jia et al., 2021), Flamingo (Alayrac et al., 2022), BLIP (Li et al., 2022) and Llava (Liu et al., 2023), arises from their ability to generalize from vast amounts of web-collected data, learning nuanced, multimodal representations that are applicable across various tasks and domains.

Vision-language prompt tuning. Research has also focused on improving prompt-based learning and fine-tuning methods, such as CoOp (Zhou et al., 2022b) and CoCoOp (Zhou et al., 2022a), which adapt VLMs more effectively to specific tasks by learning customized prompt strategies. CoOp

transforms static text prompts into dynamic, learnable components. This allows prompts to adjust during training, aligning model responses with task-specific needs, and improving performance, especially in zero-shot or few-shot settings. Following CoOp, several studies Lu et al. (2022); Sun et al. (2022); Derakhshani et al. (2023); Zhu et al. (2023); Gao et al. (2024) have advanced prompt tuning to enhance model performance.

Knowledge distillation. Knowledge distillation (Hinton et al., 2015; Ahn et al., 2019; Zhao et al., 2022; Jin et al., 2023) is an effective model compression technique in which a smaller, more efficient student model learns from a larger, more complex teacher model. The conventional method for knowledge distillation involves training the student model to minimize the difference between its predicted probability distribution and that of the teacher model, often measured using Kullback-Leibler (KL) divergence. However, weak-to-strong generalization offers an alternative by having strong models supervised by weaker models.

Weak-to-strong generalization. The concept of weak-to-strong generalization, initially introduced by Burns et al. (2023), presents a promising approach for aligning super-intelligent models with human values. This study emphasizes the significance of the issue and provides experimental evidence to support its feasibility. Building on this framework, Guo et al. (2024) introduces a dynamically adjusted confidence loss and demonstrates the effectiveness of their method in the context of visual foundation models. Therefore, based on those previous work, we explore the weak-to-strong generalization for VLMs.

3 PRELIMINARIES

In this section, we outline the preliminary studies considered in this paper.

CLIP-like vision-language models. The CLIP model (Radford et al., 2021) employs a vision encoder $f^{\text{s}}_{\text{vision}}$ and a text encoder $f^{\text{s}}_{\text{text}}$, which jointly learn to map visual inputs x_i and textual inputs $\boldsymbol{t}_j$ into feature embeddings $\boldsymbol{r}_i = f^{\text{s}}_{\text{vision}}(\boldsymbol{x}_i)$ and $\boldsymbol{r}_j = f^{\text{s}}_{\text{text}}(\boldsymbol{t}_j)$, respectively. These embeddings are projected into a shared latent space where their similarity is measured by cosine similarity, $\cos(\boldsymbol{r}_i, \boldsymbol{r}_j)$. By maximizing the similarity of positive pairs $(\boldsymbol{r}_i, \boldsymbol{r}_j)$ and minimizing the similarity of negative pairs sampled from the dataset, CLIP optimizes the contrastive loss function:

$$\mathcal{L}_{\text{CLIP}} = \frac{1}{N}\sum_{n=1}^{N} \log \frac{\exp(\cos(\boldsymbol{r}_{i_n}, \boldsymbol{r}_{j_n})/\tau)}{\sum_{k=1}^{N} \exp(\cos(\boldsymbol{r}_{i_n}, \boldsymbol{r}_{j_k})/\tau)} + \frac{1}{N}\sum_{n=1}^{N} \log \frac{\exp(\cos(\boldsymbol{r}_{j_n}, \boldsymbol{r}_{i_n})/\tau)}{\sum_{k=1}^{N} \exp(\cos(\boldsymbol{r}_{j_n}, \boldsymbol{r}_{i_k})/\tau)},$$

where N is the batch size, $(\boldsymbol{i}_n, \boldsymbol{j}_n)$ denotes the index pairs of positive examples, and τ is a temperature parameter. This contrastive learning approach enables CLIP to achieve remarkable zero-shot classification performance across various tasks, leveraging its pretrained representations $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ without task-specific training.

CLIP linear probs. The standard method to fine-tune pre-trained VLMs, e.g., CLIP, involves training a linear classifier on the feature representations extracted from these pre-trained models. This approach mirrors how Radford et al. (2021) evaluated the transferability of CLIP, treating pre-trained models primarily as feature extractors. This method is generally more efficient because only the parameters of the additional classification heads need to be trained. The formula is:

$$\hat{\boldsymbol{p}} = \text{softmax}(\boldsymbol{W} \cdot f^{\text{s}}_{\text{vision}}(\boldsymbol{x}) + b) \tag{1}$$

where $f^{\text{s}}_{\text{vision}}(\boldsymbol{x})$ denotes the feature representation extracted from the pre-trained CLIP model for an input $\boldsymbol{x}$, $\boldsymbol{W}$ represents the weights of the linear classifier, b is the bias term, and $\hat{p}$ is the predicted probability distribution over the classes. The parameters $\boldsymbol{W}$ and b are learned during the training process on the downstream task's labeled data. This approach leverages the rich feature representations learned by CLIP during its pre-training phase, enabling efficient and effective adaptation to new tasks with minimal additional training. In this method, $f^{\text{s}}_{\text{text}}$ is not used during this training process.

CLIP prompt tuning. A recent mainstream approach to more effectively adapt VLMs involves learning customized prompts (Zhou et al., 2022b). This method fine-tunes the input prompts that

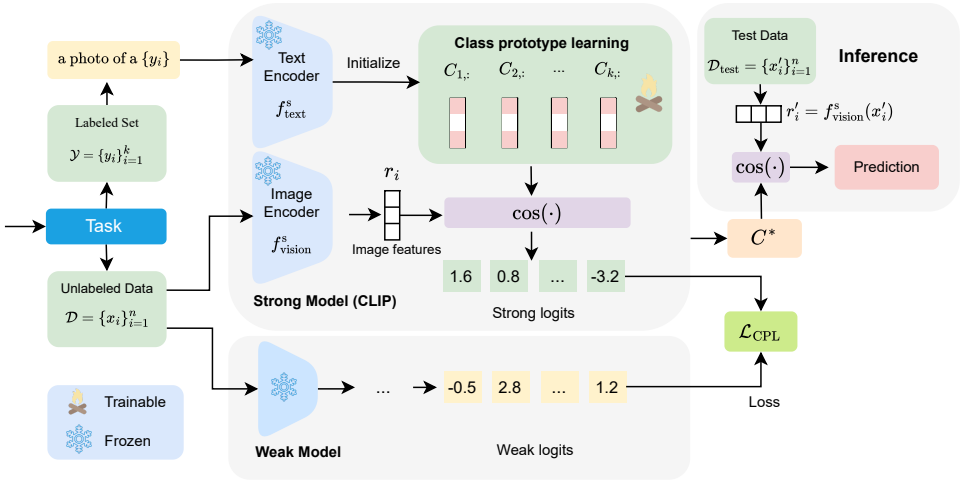


Figure 1: **Overview of the weak-to-strong process for enhancing strong model performance using weak model supervision**. Unlabeled data from a given task is fed into both a strong model (CLIP) and a weak model. The strong model uses an image encoder to generate image features ($\boldsymbol{r}_i$), which are compared with learnable class prototypes ($\boldsymbol{C}_{1,:}, \boldsymbol{C}_{2,:}, ..., \boldsymbol{C}_{k,:}$) through cosine similarity to produce strong logits. Concurrently, the weak model generates weak logits from the same data. Our alignment loss (L_{CPL} in Eq. 5) is computed between the strong logits (based on the prototype matrix $\boldsymbol{C}$) and weak logits. For test data, the image features ($\boldsymbol{r}'_i$) extracted from the strong model f^{s} are compared with the learned prototype matrix $\boldsymbol{C}^*$ to make predictions, aiming to improve the strong model's classification performance in the given task.

guide the model's attention and feature extraction processes. Mathematically, this approach can be represented as:

$$\hat{\boldsymbol{p}} = \cos(f^{\text{s}}_{\text{vision}}(\boldsymbol{x}), f^{\text{s}}_{\text{text}}(\{\boldsymbol{t}_i\}_{i=1}^{k})) \tag{2}$$

where $f^{\text{s}}_{\text{vision}}(\boldsymbol{x})$ denotes the feature representation extracted from the vision encoder for an input $\boldsymbol{x}$, and $f^{\text{s}}_{\text{text}}(\{\boldsymbol{t}_i\}_{i=1}^{k})$ denotes the feature representation extracted from the text encoder for a set of prompts $\{\boldsymbol{t}_i\}_{i=1}^{k}$, where $\boldsymbol{t}_i = \{\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_k, \{\text{classname}_i\}\}$, with $\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_k$ representing the learned prompt vectors and $\{\text{classname}_i\}$ being the target class name. The function cos represents the cosine similarity between the vision and text feature representations. The parameters of the prompt vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_k$ are learned during the training process, enabling the model to better align the vision and language features for the specific downstream task. Unlike in the previous method, $f^{\text{s}}_{\text{text}}$ is utilized during the training.

4 Weak-to-Strong Learning for CLIP-based Classification

In this section, we first introduce the problem formulation and describe our proposed method CPL. Additionally, the overall procedure is shown in Figure 1, and the algorithm is shown in Algorithm 1.

Problem formulation. In this paper, we consider a scenario involving a weakly pre-trained model f^{w} and a strongly pre-trained model f^{s}, where f^{s} generally exhibits better generalization due to more parameters or extensive training data. In this paper, we consider f^{s} to be a VLM model, e.g., CLIP (Radford et al., 2021), while f^{w} is a vision model. Given a target task, we have a dataset consisting of n unlabeled samples $\mathcal{D} = \{\boldsymbol{x}_i\}_{i=1}^{n}$, m labeled test samples[1] $\mathcal{D}_{\text{test}} = \{(\boldsymbol{x}_i^{\text{te}}, \boldsymbol{y}_i)\}_{i=1}^{m}$ and a label set $\mathcal{Y} = \{\boldsymbol{y}_i\}_{i=1}^{k}$, where k represents the number of category and each $\boldsymbol{y}_i$ represent one semantic label. We apply the weak model f^{w} to $\mathcal{D}$ to generate predictions, which we refer to as *weak supervision*, represented by a weakly supervised dataset $\mathcal{D}_w = \{(\boldsymbol{x}_i, f^{\text{w}}(\boldsymbol{x}_i)\}_{i=1}^{n}$. The task of weak-to-strong

[1]This test set is only used for testing phrase.

generalization is to fine-tune the strong model f^{s} with the weakly supervised dataset $\mathcal{D}_w$ to enhance its classification capabilities on the test dataset $\mathcal{D}_{\text{test}}$.

Class prototype learning. Empirical evidence (Figure 2a and 2b) indicates that previous VLM fine-tuning approaches, including linear probs and prompt tuning, applied in weak-to-strong generalization often result in strong models overfitting to the weak models. Consequently, this leads to the strong models performing close to the weak models on test sets. To address this, we aim to learn the set of class prototypes as a matrix $\boldsymbol{C} \in \mathbb{R}^{k \times d}$, where k is the total number of classes and each row $\boldsymbol{C}_{i,:} \in \mathrm{R}^{1 \times d}$ is the class prototype for each class i based on the feature embeddings of training images belonging to that class, which encapsulate the characteristics of each class. The prototype representation $\boldsymbol{C}_{i,:}$ for each class i can be initialized by the text embedding corresponding to a textual description of the class label. For instance, $\boldsymbol{C}_{i,:}$ could be initialized with the text embedding of "a photo of a {label}" extracted by CLIP text encoder, where label represents the class label name.

When presented with an input image $\boldsymbol{x} \in \mathbb{R}^{h \times w \times c}$ in $\mathcal{D}$, we compute its visual feature embedding $f^{\mathrm{s}}_{\text{vision}}(\boldsymbol{x})$, where $f^{\mathrm{s}}_{\text{vision}}(\boldsymbol{x}) \in \mathbb{R}^{d \times 1}$. Then, the cosine similarity between the image embedding $f^{\mathrm{s}}_{\text{vision}}(\boldsymbol{x})$ and each class prototype $\boldsymbol{C}_{i,:}$ is calculated as the logits $\boldsymbol{z}^{\mathrm{s}}$. Mathematically, the unnormalized logit for i-th class (i.e., the i-th element in $\boldsymbol{z}$) regarding $\boldsymbol{x}$ is computed as:

$$z_i^{\mathrm{s}}(\boldsymbol{C}_{i,:}, \boldsymbol{x}) = \frac{\boldsymbol{C}_{i,:} f^{\mathrm{s}}_{\text{vision}}(\boldsymbol{x})}{\|\boldsymbol{C}_{i,:}\| \|f^{\mathrm{s}}_{\text{vision}}(\boldsymbol{x})\|}, \tag{3}$$

where this cosine similarity operation measures the alignment between the image and class centroids, providing a measure of the image's association with each class. Subsequently, a softmax function can be applied to the logits to obtain class probabilities.

Weak-to-strong alignment. The ultimate aim of weak-to-strong alignment is to elicit the capabilities of a much stronger model using weak supervision from a weaker model (Burns et al., 2023). Unlike knowledge distillation, where the stronger model serves as the teacher and the weaker model as the student, weak-to-strong alignment reverses these roles. Here, the weaker models act as students guiding the stronger model. A straightforward approach to this challenge is to use knowledge distillation methods (Hinton et al., 2015) to make the strong model's behavior agree with that of the weak model. Most logit-based KD methods utilize the KL divergence, which quantifies the amount of information lost when approximating one probability distribution with another. Therefore, for each $\boldsymbol{x}$ in $\mathcal{D}$, given the logits of the weak model, $\boldsymbol{z}^{\mathrm{w}}(\boldsymbol{x}) = f^{\mathrm{w}}(\boldsymbol{x})$, and those of the strong model $\boldsymbol{z}^{\mathrm{s}}$ (using Eq. 3), we convert them into the softened probability vector $\boldsymbol{p}^{\mathrm{w}}$ and $\boldsymbol{p}^{\mathrm{s}}$. The i-th value of $\boldsymbol{p}^{\mathrm{w}}$ or $\boldsymbol{p}^{\mathrm{s}}$ is computed by a *softmax* function with a temperature hyperparameter τ, which is denoted by

$$p_i^{\mathrm{w}}(\boldsymbol{x}) = \frac{\exp(z_i^{\mathrm{w}}(\boldsymbol{x})/\tau)}{\sum_{j=1}^{k} \exp(z_j^{\mathrm{w}}(\boldsymbol{x})/\tau)}, \quad p_i^{\mathrm{s}}(\boldsymbol{C}_{i,:}, \boldsymbol{x}) = \frac{\exp(z_i^{\mathrm{s}}(\boldsymbol{C}_{i,:}, \boldsymbol{x})/\tau)}{\sum_{j=1}^{k} \exp(z_j^{\mathrm{s}}(\boldsymbol{C}_{j,:}, \boldsymbol{x})/\tau)}. \tag{4}$$

Thus, the loss value of each $\boldsymbol{x}$ in $\mathcal{D}$ is realized by minimizing the KL divergence between softened probability vectors of weak and strong models, which is defined as:

$$\mathcal{L}_{\mathrm{CPL}}(\boldsymbol{C}, \boldsymbol{x}) = \mathrm{KL}(\boldsymbol{p}^{\mathrm{s}}(\boldsymbol{C}, \boldsymbol{x}) \parallel \boldsymbol{p}^{\mathrm{w}}(\boldsymbol{x})) = \sum_{i=1}^{k} p_i^{\mathrm{w}}(\boldsymbol{C}_{i,:}) \log \frac{p_i^{\mathrm{w}}(\boldsymbol{C}_{i,:})}{p_i^{\mathrm{s}}(\boldsymbol{C}_{i,:}, \boldsymbol{x})}. \tag{5}$$

We demonstrate the overall algorithm in Algorithm 1. The algorithm for weak-to-strong generalization in VLMs begins by initializing class prototypes using text embeddings from the strong model. During training, mini-batches of unlabeled data are processed to obtain feature embeddings and generate logits from both the strong and weak models. The alignment loss between these logits is computed to update the class prototypes iteratively. Once training is complete, the learned class prototypes are used to compute feature embeddings from the test data, generate logits through cosine similarity, and predict the labels by selecting the class with the highest logit value.

5 Experiments

In this section, we evaluate the performance of our method by a series of experiments and various ablation studies. The implementation details can be found in Appendix 5.

Algorithm 1: Weak-to-strong Generalization for VLMs

Input : An unlabeled set: $\mathcal{D} = \{\boldsymbol{x}_i\}_{i=1}^n$; a test set: $\mathcal{D}_{\text{test}} = \{\boldsymbol{x}_i^{\text{te}}\}_{i=1}^m$; a label set: $\mathcal{Y} = \{\boldsymbol{y}_i\}_{i=1}^k$; a weak model: $f^{\text{w}}(\cdot)$; a strong model: $f^{\text{s}}(\cdot)$; learnable class prototypes: $\boldsymbol{C} \in \mathbb{R}^{k \times d}$; maximum epochs: $T_{\max}$; alignment loss function: $\mathcal{L}_{\text{CPL}}(\cdot, \cdot)$.

1: Obtain $f^{\text{s}}_{\text{vision}}(\cdot)$, $f^{\text{s}}_{\text{text}}(\cdot)$ from $f^{\text{s}}(\cdot)$;
2: Initialize class prototypes $\boldsymbol{C}$ where $\boldsymbol{C}_{i,:} = f^{\text{s}}_{\text{text}}(\text{“a photo of a } \{\boldsymbol{y}_i\}\text{”})$;
for $T = 1$ **to** $T_{\max}$ **do**
  3: Fetch mini-batch $\mathcal{B}$ in $\mathcal{D}$;
  4: Compute the average loss $\mathcal{L} = \frac{1}{|\mathcal{B}|} \sum_{\boldsymbol{x} \in \mathcal{B}} \mathcal{L}_{\text{CPL}}(\boldsymbol{C}, \boldsymbol{x})$;
  5: Update class prototypes $\boldsymbol{C}$ using Adam (Kingma & Ba, 2014) and the average loss $\mathcal{L}$;
end
6: Get learned class prototypes $\boldsymbol{C}^*$;
7: Obtain test feature embeddings $\{\boldsymbol{r}_i\}_{i=1}^m = \{f^{\text{s}}_{\text{vision}}(\boldsymbol{x}^{\text{te}})\}_{\boldsymbol{x}^{\text{te}} \in \mathcal{D}_{\text{test}}}$;
8: Compute predicted logits $\{\boldsymbol{z}_i\}_{i=1}^m$ where $\boldsymbol{z}_i = \cos(\boldsymbol{C}^*, r_i)$;
9: Compute prediction $\hat{\mathcal{Y}} = \{\arg\max_j \boldsymbol{z}^{\text{s}}_{i,j}\}_{i=1}^n$;

Output : $\hat{\mathcal{Y}}$

Table 1: **DomainNet statistics.** This table provides statistics for the DomainNet dataset (Peng et al., 2019) across different styles: Clipart, Infograph, Painting, Quickdraw, Real, and Sketch. It includes the number of classes (#Classes), the number of training samples (#Train), the number of test samples (#Test), and the total number of samples (#Total) for each style. Each style has 345 classes, with varying numbers of training and test samples.

	Clipart	Infograph	Painting	Quickdraw	Real	Sketch
#Classes	345	345	345	345	345	345
#Train	33525	36023	50416	120750	120906	48212
#Test	14604	15582	21850	51750	52041	20916
#Total	48,129	51,605	72,266	172,500	172,947	69,128

Datasets. In our exploration of weak-to-strong scenarios, we turn to the challenging and relatively large dataset: *DomainNet* (Peng et al., 2019). Comprising six diverse domains, each housing 345 categories of common objects, *DomainNet* offers a rich landscape for analysis. These domains encompass a range of visual styles and sources: Clipart, featuring a collection of clipart images; Infograph, presenting infographic images with specific objects; Painting, showcasing artistic renditions of objects in the form of paintings; Quickdraw, housing drawings from the popular game "Quick Draw!" by worldwide players; Real, encompassing photographs and real-world images; and Sketch, containing sketches of various objects. Refer to Table 1 for detailed statistics into each domain.

Experimental setup. In our experiments, each domain within the DomainNet dataset is treated as an individual task, resulting in a total of 6 tasks under consideration. To investigate weak-to-strong generalization within the setting of VLMs, we design these steps to simulate the problem:

(1) **Dataset splitting**: Referring to Table 1, each domain is divided into a training set $\mathcal{D}_{\text{train}}$ and a test set $\mathcal{D}_{\text{test}}$. The test set $\mathcal{D}_{\text{test}}$ is further partitioned into $\mathcal{D}_{\text{hold}}$ and $\mathcal{D}'_{\text{test}}$, comprising 80% and 20% of $\mathcal{D}_{\text{test}}$ respectively. (2) **Create the weak model**: The training data $\mathcal{D}_{\text{train}}$ is utilized to fine-tune the weak model, employing ground truth labels. Evaluation occurs in $\mathcal{D}_{\text{test}}$, termed as *weak performance*. (3) **Weak supervision set generation**: The *weak supervision* set $\mathcal{D}'_{\text{hold}}$ is generated by the weak model from $\mathcal{D}_{\text{hold}}$, replacing ground labels with logits produced by the weak model. (4) **Strong model training with weak supervisor**: Initially, $\mathcal{D}'_{\text{hold}}$ is split into 80% and 20% portions for strong model fine-tuning and parameter tuning, respectively. The strong model is then fine-tuned in the holdout training set. The final performance is assessed in $\mathcal{D}_{\text{test}}$, labeled as *weak-to-strong performance*. (5) **Strong model training with ground truth labels as ceiling**: Finally, the strong model undergoes fine-tuning on $\mathcal{D}_{\text{hold}}$ (with ground truth labels) to represent *strong ceiling performance*.

Table 2: **Performance on DomainNet datasets across different methods and styles**. This table showcases the results in accuracy (%) of various methods on different styles within the DomainNet datasets, including Clipart, Infograph, Painting, Quickdraw, Real, and Sketch. The average performance across all styles is also listed. The compared methods include CE+LP, KD+LP, AuxConf+LP, AdaptConf+LP, CE+TP, KD+TP, AuxConf+TP, and AdaptConf+TP, with CPL yielding the highest performance in most categories. The final row, Δ, represents the improvement margin of CPL over other methods. CPL is used as the strong ceiling performance, which is the best among the LP, TP, and CPL

Method	DomainNet						Avg.
	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	
Weak	67.15	31.71	67.90	46.70	85.10	52.26	-
Strong Ceiling	74.27	50.84	72.24	49.92	85.34	66.64	-
CE+LP	66.97	30.55	64.90	45.59	82.69	55.28	57.66
KD+LP	70.69	35.78	68.28	**48.15**	83.66	59.88	61.07
AuxConf+LP	67.41	18.37	66.92	30.59	84.00	56.79	54.01
AdaptConf+LP	70.68	33.78	67.86	47.80	83.84	60.18	60.69
CE+TP	62.02	29.25	62.68	44.82	81.16	52.51	55.41
KD+TP	69.57	36.03	68.37	47.34	83.70	59.93	60.82
AuxConf+TP	69.20	20.97	68.61	43.60	83.92	58.49	57.47
AdaptConf+TP	69.97	35.96	68.18	47.42	83.59	59.95	60.85
CPL (Ours)	**73.10**	**46.14**	**71.80**	47.96	**85.41**	**64.01**	**64.74**
Δ	2.41	10.11	3.19	-0.19	1.41	3.83	3.67

Baselines. In exploring the weak-to-strong problem within the VLM setting, we investigate different fine-tuning strategies. Initially, Radford et al. (2021) assessed CLIP's transferability via *linear probs* (LP) across many datasets. Subsequent research focused on *textual prompting* (TP) (Zhou et al., 2022b), where a learnable prompt is learned from a small target dataset. This method is data-efficient and demonstrates good generalization effects. Prompt tuning has emerged as a popular method for adapting VLMs to downstream tasks (Wu et al., 2023). Thus, we adopt linear probs and prompt tuning as our foundational fine-tuning strategies within the realm of weak-to-strong generalization. In addition, we compare our method with the following learning strategies:

(1) **Cross entropy** (CE): Utilized in studies by (Radford et al., 2021; Zhou et al., 2022b), cross-entropy measures the disparity between one-hot ground truth label distribution and model prediction probability. It serves as a straightforward baseline for this task. (2) **Knowledge distillation** (KD) (Hinton et al., 2015) transfer knowledge from a strong model to a smaller one, serving as a fundamental baseline due to its simplicity and effectiveness. (3) **Auxiliary confidence loss** (AuxConf) is proposed by Burns et al. (2023), which excels in balancing direct learning from the weak model with the inherent capacity of the strong model. (4) **Adaptive Confidence loss** (AdaptConf) is introduced by Guo et al. (2024) that dynamically adjusts weights based on confidence levels, enabling the strong model to discern when to prioritize its predictions or follow the guidance of the weak model.

Implementation details. In this section, we provide an overview of the implementation details regarding our proposed method and comparative baseline methods on simulation experiments. The code is mainly based on Pytorch and the Huggingface library. We employed ResNet and ViT as the weak model and CLIP as the strong model, for our task. The evaluation is performed in five random seeds. During training, we used a test batch size of 2048 for evaluation. The weak model was trained for 3 epochs with a batch size of 512 and a learning rate of 1e-3, whereas the strong model underwent 10 epochs with the same batch size and a learning rate of 1e-2. The learning rate was adjusted dynamically, and a warm-up ratio of 0.1 was utilized. We also ensured the loading of the best model at the end of training based on the validation set. All our experiments are conducted using a single V100 GPU with 40GB of memory, supported by 8 CPU workers and 64GB of RAM.

Table 3: **Performance Comparison of Different Weak Models**. This table presents the results in accuracy (%) of various methods applied to weak models, including Resnet-18, Resnet-26, Resnet-34, Cvt-13, and Convnext-tiny-224. The average performance across all models is also provided. The strong ceiling performance is given for reference. The methods compared are CE+LP, KD+LP, AuxConf+LP, AdaptConf+LP, CE+TP, KD+TP, AuxConf+TP, and AdaptConf+TP, with CPL showing the best performance. The final row, Δ, indicates the improvement margin of CPL over other methods.

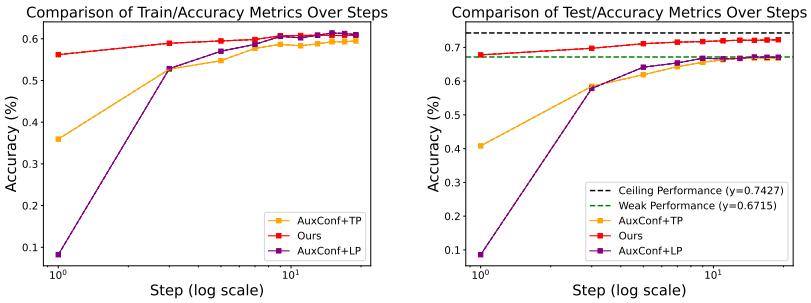
Method	Weak Models					Avg.
	Resnet-18	Resnet-26	Resnet-34	Cvt-13	Convnext-tiny-224	
Weak	55.22	57.2	59.96	51.33	69.47	-
Strong Ceiling			74.27			-
CE+LP	64.28	64.19	64.72	62.19	69.5	64.98
KD+LP	66.80	67.26	68.53	65.89	71.53	68.00
AuxConf+LP	66.19	67.02	66.21	62.52	71.10	66.61
AdaptConf+LP	67.72	68.13	68.83	66.95	71.42	68.61
CE+TP	62.22	63.35	64.04	61.02	67.71	63.67
KD+TP	65.71	66.43	67.34	64.50	69.31	66.66
AuxConf+TP	66.39	67.04	67.47	66.04	69.56	67.30
AdaptConf+TP	65.91	66.69	67.42	65.31	69.04	66.87
CPL (Ours)	**72.25**	**71.84**	**72.06**	**71.91**	**72.47**	**72.11**
Δ	4.53	3.71	3.23	4.96	0.94	3.50

Experiment results. The results presented in Table 2 provide a comprehensive evaluation of various methods across multiple domains within the DomainNet dataset. Each method's efficacy is assessed based on its accuracy in six distinct domains: Clipart, Infograph, Painting, Quickdraw, Real, and Sketch. Notably, our proposed method (CPL) exhibits remarkable performance, achieving the highest average accuracy of 64.74%. This signifies a substantial improvement over baseline methods, with CPL outperforming the best-performing baseline by notable margins, showcasing gains of 2.41%, 10.11%, 3.19%, -0.19%, 1.41%, and 3.67% across respective domains.

In the domain of QuickDraw, it is evident that CLIP demonstrates a lower zero-shot ability, suggesting significant disparities between the data distribution in QuickDraw and the CLIP training data. This observation underscores the challenge of generalizing CLIP to the QuickDraw domain effectively. Surprisingly, in such cases, the straightforward KD approach emerges as the most effective method, outperforming more sophisticated techniques. This phenomenon suggests that the inherent structure of the KD method enables it to leverage available information optimally, leading to superior performance despite the substantial dissimilarities between the CLIP and QuickDraw domains.

In the context of the Infograph domain, the weak model exhibits notably inferior performance compared to all other domains. Conversely, our proposed method demonstrates the most substantial performance improvement, showcasing a significant gain in accuracy as compared to both the weak model and other competing methods. This highlights the effectiveness of our approach in addressing the challenges specific to the Infograph domain, where the weak model struggles to generalize effectively. The considerable performance gain achieved by our method underscores its ability to capture and leverage domain-specific features, resulting in improved accuracy and robustness in handling Infograph data.

Ablation on different weak supervision. Table 3 illustrates our approach to various forms of weak supervision across different models in detail, such as Resnet models (He et al., 2016) (Resnet-18, Resnet-26, Resnet-34), Cvt-13 (Wu et al., 2021), and Convnext-tiny-224 (Liu et al., 2022). The experiment was conducted on the DomainNet Clipart domain, revealing a diverse range of performances from different weak models, with accuracy scores spanning from 55.2% to 69.47%. Notably, our method consistently achieved the best weak-to-strong generalization performance among all the weak models tested, closely approximating the strong ceiling performance, which was benchmarked at 74.27%.



(a) Train Accuracy. (b) Test Accuracy.

Figure 2: **Comparison of train and test accuracy metrics over training steps**. It shows the comparison of train (a) and test (b) accuracy metrics for different methods over training steps. Methods include AuxConf+TP, Ours, and AuxConf+LP. "Ours" demonstrates the highest accuracy, nearing the ceiling performance (y = 0.7427) and surpassing weak performance (y = 0.6715) in both the training and testing phases.

Table 4: **Average Performance Comparison of Different Tuning Methods in Accutacy (%)**.

Method	Performance
Text Encoder	70.34
$\boldsymbol{C}$	74.42

Our method's superior performance is evident across various weak supervision techniques. The results show that while other methods improved performance to varying degrees, none matched the consistency and high performance of our method. For instance, our approach significantly outperformed the baseline weak supervision models, achieving top accuracy scores such as 72.25% for Resnet-18, 71.84% for Resnet-26, 72.06% for Resnet-34, 71.91% for Cvt-13, and 72.47% for Convnext-tiny-224. On average, our method achieved a performance gain of 3.5%, underscoring its superior ability to enhance model accuracy through improved weak supervision techniques.

The performance gains highlight the incremental improvements our method brings compared to other approaches. These improvements range from 0.94% to 4.96%, demonstrating our method's ability to consistently push model performance closer to the strong ceiling benchmark. From this table, it is evident that weak-to-strong generalization is feasible in the VLMs setting. By utilizing supervision from weak models, our strong model has attained results that are close to the ceiling performance.

Ablation on different tuning methods. We have conducted an ablation study to compare the performance of tuning C versus tuning the text encoder. The results have been shown in Table 4. Our findings indicate that tuning $\boldsymbol{C}$ yields better performance than tuning the text encoder. The study by Wu et al. (2023) shows that prompt tuning for VLMs is more robust to noisy labels compared to fine-tuning.

Analysis of our method. In Figures 2a and 2a, we demonstrate the training and test accuracy over training steps for our method compared to two baseline methods, AuxConf+TP and AuxConf+LP. The training accuracy plot shows that all methods eventually converge to an accuracy of around 0.6. Specifically, our method shows a rapid and consistent improvement, achieving high training accuracy more quickly than the other methods. The baseline methods, AuxConf+TP and AuxConf+LP, also improve but at different rates, with AuxConf+TP showing a steadier progression and AuxConf+LP catching up later in the process. In the test accuracy plot, the differences between the methods become more pronounced. While AuxConf+TP and AuxConf+LP exhibit similar weak performance levels, struggling to surpass a certain threshold, our method showcases significantly better performance. It not only achieves higher test accuracy but also maintains this performance consistently over the steps, closely approaching the ceiling performance.

6 CONCLUSION

In conclusion, traditional alignment techniques for LLMs, which rely heavily on human supervision, such as RLFH, face significant challenges due to the intricacy of model outputs and the inefficiency of requiring substantial human feedback. To address this, a novel approach has recently been proposed where a weaker model supervises a much stronger one. Extending this concept to VLMs leverages these insights, adapting the proven benefits to a multi-modal context. Hence, we introduced a method called CPL, which effectively enhances the classification capabilities of VLMs with weak supervision. Our simulation experiments validate the effectiveness of this weak-to-strong approach. Extensive experimental results demonstrate that our method significantly improves performance across various benchmarks. These results underscore the potential of weak supervision as a powerful tool in the alignment, offering a promising avenue for future research and application.

7 LIMITATION

Since the core problem we aim to address in this research has not yet emerged, we currently lack access to superintelligence models. Although our experiments rely on simulations, these simulated scenarios do not fully replicate the complexities of the actual challenge we anticipate. Consequently, there exists a significant gap between our simulated experiments and the real-world problem. This discrepancy implies that the methods demonstrating success in our current simulations may not necessarily prove effective when applied to the final real-world task. Therefore, while our current research provides valuable insights and progress, it remains crucial to acknowledge these limitations and continue refining our approaches to better align with the ultimate goal of weak-to-strong alignment.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9163–9171, 2019.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. *arXiv preprint arXiv:2312.09390*, 2023.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Mohammad Mahdi Derakhshani, Enrique Sanchez, Adrian Bulat, Victor G Turrisi da Costa, Cees GM Snoek, Georgios Tzimiropoulos, and Brais Martinez. Bayesian prompt learning for image-language model generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15237–15246, 2023.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2):581–595, 2024.

Amelia Glaese, Nat McAleese, Maja Trębacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, et al. Improving alignment of dialogue agents via targeted human judgements. *arXiv preprint arXiv:2209.14375*, 2022.

Jianyuan Guo, Hanting Chen, Chengcheng Wang, Kai Han, Chang Xu, and Yunhe Wang. Vision superalignment: Weak-to-strong generalization for vision foundation models. *arXiv preprint arXiv:2402.03749*, 2024.

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

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

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*, pp. 4904–4916. PMLR, 2021.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pp. 709–727. Springer, 2022.

Ying Jin, Jiaqi Wang, and Dahua Lin. Multi-level logit distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24276–24285, 2023.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pp. 12888–12900. PMLR, 2022.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11976–11986, 2022.

Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5206–5215, 2022.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1406–1415, 2019.

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*, pp. 8748–8763. PMLR, 2021.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

Ximeng Sun, Ping Hu, and Kate Saenko. Dualcoop: Fast adaptation to multi-label recognition with limited annotations. *Advances in Neural Information Processing Systems*, 35:30569–30582, 2022.

Cheng-En Wu, Yu Tian, Haichao Yu, Heng Wang, Pedro Morgado, Yu Hen Hu, and Linjie Yang. Why is prompt tuning for vision-language models robust to noisy labels? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15488–15497, 2023.

Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 22–31, 2021.

Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pp. 11953–11962, 2022.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16816–16825, 2022a.

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

Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15659–15669, 2023.