CLIP meets DINO for Tuning Zero-Shot Classifier using Unlabeled Image Collections

Mohamed Fazli Imam[1], **Rufael Fekadu Marew**[1], **Jameel Hassan**[1,2], **Mustansar Fiaz**[3], **Alham Fikri Aji**[1], **Hisham Cholakkal**[1]

[1]Mohamed Bin Zayed University of AI [2]The Johns Hopkins University [3]IBM Research

mohamed.imam@mbzuai.ac.ae rufael.marew@mbzuai.ac.ae jameel.hassan@mbzuai.ac.ae
mustansar.fiaz@ibm.com alham.fikri@mbzuai.ac.ae hisham.cholakkal@mbzuai.ac.ae

Abstract

In the era of foundation models, CLIP has emerged as a powerful tool for aligning text and visual modalities into a common embedding space. However, the alignment objective used to train CLIP often results in subpar visual features for fine-grained tasks. In contrast, SSL-pretrained models like DINO excel at extracting rich visual features due to their specialized training paradigm. Yet, these SSL models require an additional supervised linear probing step, which relies on fully labeled data—often expensive and difficult to obtain at scale. In this paper, we propose a label-free prompt-tuning method that leverages the rich visual features of self-supervised learning models (DINO) and the broad textual knowledge of large language models (LLMs) to largely enhance CLIP-based image classification performance using unlabelled images. Our approach unfolds in three key steps: (i) We generate robust textual feature embeddings that more accurately represent object classes by leveraging class-specific descriptions from LLMs, enabling more effective zero-shot classification compared to CLIP's default name-specific prompts. (ii) These textual embeddings are then used to produce pseudo-labels to train an alignment module that integrates the complementary strengths of LLM description-based textual embeddings and DINO's visual features. (iii) Finally, we prompt-tune CLIP's vision encoder through DINO-assisted supervision using the trained alignment module. This three-step process allows us to harness the best of visual and textual foundation models, resulting in a powerful and efficient approach that surpasses state-of-the-art label-free classification methods. Notably, our framework, **NoLA** (No Labels Attached), achieves an average absolute gain of 3.6% over the state-of-the-art LaFter across 11 diverse image classification datasets. Our code and models can be found at https://github.com/fazliimam/NoLA.

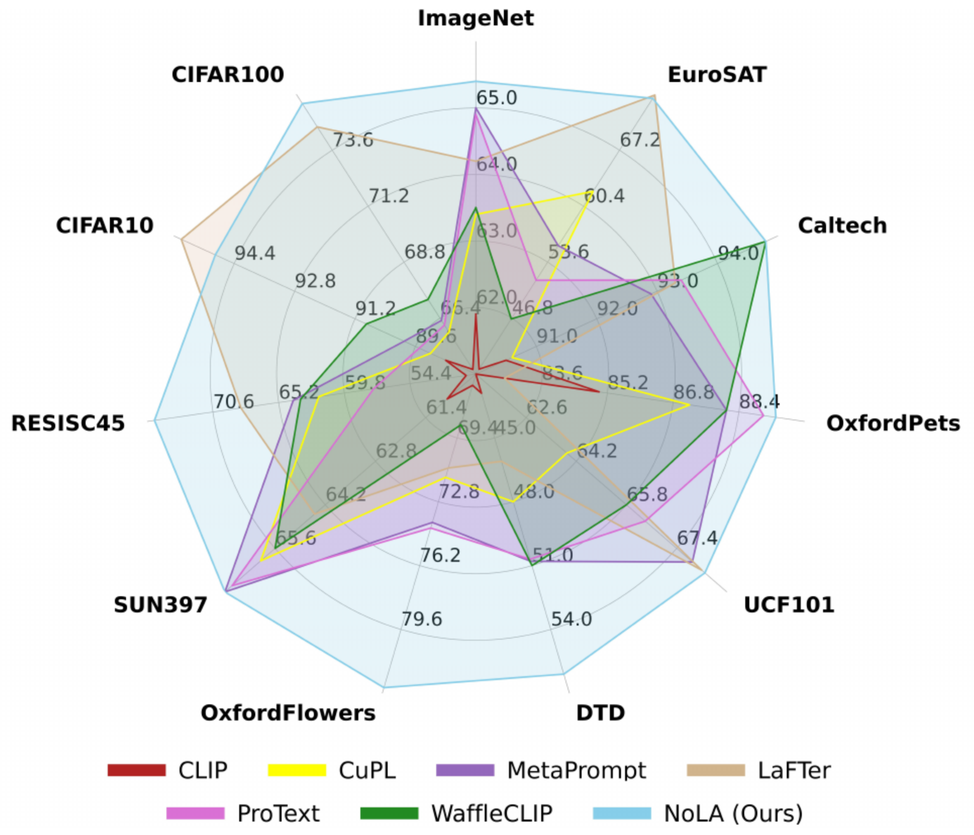

Figure 1: Top-1 accuracy (%) comparison with recent label-free method on 11 diverse image classification datasets. NoLA (Ours) achieves state-of-the-art performance in 9 out of 11 datasets, outperforming the state-of-the-art LaFter by an average absolute gain of 3.6%.

Introduction

The vision-language research landscape is rapidly evolving with foundational models like CLIP (Radford et al. 2021), ALIGN (Jia et al. 2021), and BLIP (Li et al. 2022) leading the charge. These models are composed of dual encoders for both text and images and map inputs to a shared embedding space, enabling the comparison of test image embeddings with text embeddings representing different classes. Among these, CLIP has gained particular attention for its ability to leverage contrastive learning on extensive image-text pairs. This innovative approach by aligning images and text representations without the need for additional data or training excels in many other computer vision tasks including medical imaging (Zhang et al. 2024; Zhao et al. 2023; You et al. 2023), remote sensing (Qiu et al. 2022; Chen et al. 2023; Yuan, Zhan, and Xiong 2023; Yuan et al. 2021b; Li et al. 2023a; Bazi et al. 2022), video anomaly detection (Wu et al. 2024; Joo et al. 2023) and more. Though these models offer impressive flexibility in recognizing a wide range of categories, they often require further supervised fine-tuning to match the performance of traditional methods on specific closed-set tasks. Nonetheless, the scalability and flexibility of VLMs like CLIP, ALIGN, and BLIP have significantly advanced zero-shot recognition by creating a unified representation between visual and language domains.

During pre-training, CLIP is designed to align image-text pairs within a shared feature space, enabling it to encode open-vocabulary concepts and perform effectively on zero-shot recognition tasks. CLIP includes two separate encoders—one for images and one for text. A manually crafted

prompt like "`a photo of a [CLS]`" serves as the text input during inference. The model compares the text features of different classes with the visual features, assigning the predicted label to the class with the highest similarity. CLIP exhibits remarkable zero-shot capabilities, allowing it to make accurate predictions on tasks it has not explicitly been trained for. This is achieved through its foundation in zero-shot transfer learning, where the model leverages its extensive training on a diverse dataset of 400 million image-text pairs. By learning to associate images with their corresponding textual descriptions, CLIP can generalize its understanding to new and unseen classes. This flexibility enables it to perform various tasks, such as image classification and object detection, without requiring additional fine-tuning on labeled datasets. Consequently, CLIP represents a significant advancement in multimodal AI, effectively bridging the gap between natural language understanding and computer vision. The joint vision-language embedding nature of foundational VLMs such as CLIP renders such models an ideal choice for zero-shot recognition, image captioning, visual question answering, and many other tasks. However, the zero-shot recognition capability of such vision-language foundation models often falls behind the visual recognition methods trained on the target dataset. Hence, to achieve the full potential of foundation models, it is desired to adapt them towards the target dataset. Specifically, the model needs specialized adaptation to the inherent challenges in the target dataset.

On the other hand, self-supervised learning (SSL) methods have gained prominence for their ability to leverage large volumes of unlabeled data to learn meaningful representations (Schiappa, Rawat, and Shah 2023; Caron et al. 2021; Eldele et al. 2023; Liu et al. 2023). Unlike traditional supervised learning, which requires extensive labeled datasets, SSL techniques generate their own supervisory signals from the data itself. SSL methods can uncover intricate patterns and relationships within the data, making them particularly valuable in scenarios where labeled data is scarce or costly to obtain. Generally, SSL methods (Zhu, Liu, and Huang 2023; Park and Van Hentenryck 2023; Koçyiğit, Hospedales, and Bilen 2023; Stojnic and Risojevic 2021; Akiva, Purri, and Leotta 2022) are task-agnostic, enabling a richer feature representation learning. Although SSL methods are initially trained using unlabeled data, their performance is often evaluated and enhanced in a fully supervised setting through techniques such as linear probing. In this approach, a simple linear classifier is trained on top of the features learned by the SSL model using a labeled dataset. This process allows researchers to assess the quality of the representations and fine-tune them for specific tasks. By leveraging supervised learning in this manner, SSL methods can effectively bridge the gap between unsupervised feature learning and practical, task-specific applications, ensuring that the learned representations are robust and useful for downstream classification and other supervised tasks.

In this paper, we introduce ***NoLA (No Labels Attached)***, an efficient method for fine-tuning CLIP to a set of finite classes without relying on any labels. Our goal is to eliminate the need for costly image labels by employing weakly supervised fine-tuning of the CLIP model. Specifically, following (Pratt et al. 2023), we develop an enriched Class Description Embedding (CDE) classifier using descriptions generated by large language models (LLMs). This process distills the knowledge of LLMs, resulting in a more robust classifier. The LLM-enriched CDE classifier is subsequently employed to construct a DINO-based labeling (DL) network, leveraging DINO (Caron et al. 2021)—a self-supervised learning (SSL) pre-trained vision encoder—to align with the VLM joint embedding space. Once trained, the DINO-based labeling network serves as a pseudo-labeler to learn target dataset-specific prompts, which are subsequently appended to the frozen CLIP vision encoder in a FixMatch (Sohn et al. 2020) fashion.

We demonstrate the effectiveness of NoLA across 11 popular image classification datasets in a label-free evaluation, where it surpasses existing state-of-the-art methods that use LLM-generated descriptions, achieving an average gain of 3.6% while maintaining a lightweight auto-labeling approach.

Our contributions can be summarized as follows:

- We introduce a lightweight auto-labelled adaptation of vision language models for the classification using prompt tuning, called **No Labels Attached** *NoLA*.
- Leveraging the rich contextual knowledge base of LLMs, we compose a class description embedding (CDE) classifier to generate pseudo labels. The enriched class description embedding (CDE) is used to align a pretrained SSL encoder to the VLM joint embedding space as a DINO-based labelling (DL) network. The strong visual SSL encoder, aligned towards the VLM embedding space is used as the auto-labeller towards adapting the VLM vision encoder using prompt tuning.
- Through an extensive evaluation on 11 widely recognized image classification datasets, we demonstrate that our method, NoLA (i) achieves an 11.91% average improvement over zero-shot CLIP and (ii) surpasses the previous state-of-the-art in a label-free setting on 9 out of the 11 datasets. Morover, our method (NoLA) achieves an average absolute gain of 3.6% over the state-of-the-art LaFTer, across 11 datasets.

Related Works

Vision-Language Models

The vision language models (VLMs) (Radford et al. 2021; Jia et al. 2021; Naeem et al. 2023b; Yuan et al. 2021a; Yao et al. 2021; Yu et al. 2022) architecture involves three key components: firstly, utilizing a visual backbone (Dosovitskiy et al. 2020) to encode visual representations; secondly, engaging a language model (Vaswani et al. 2017) to interpret the text description and generate appropriate text embeddings; and finally, consolidating a contrastive learning objective to unify the visual representation along with language models. These VLMs are designed to attract the rich multimodal features together for the aligned image-text pairs as well as keep distance for the un-paired image-text features which are disjoint in a unified manner. For example, CLIP

(Radford et al. 2021), ALIGN (Jia et al. 2021), and Florence (Yuan et al. 2021a) demonstrate remarkable performance in visual representation learning and transfer learning for natural scenes. The resulting models act like open-vocabulary concepts and are capable of achieving promising performance in many downstream tasks including zero-shot downstream tasks; such as open-vocabulary image classification (Khattak et al. 2023; Naeem et al. 2023a), object detection (Cozzolino et al. 2024; Pan et al. 2024), and segmentation (Liang et al. 2023; Wysoczańska et al. 2024). Although these VLMs exhibit great performance, maintaining generalization capabilities remains a crucial challenge.

Zero-shot Learning

Provided the labels for the seen categories, the main objective of zero-shot learning (ZSL) is to learn a classifier that can discriminate the test samples of the unseen classes (Pourpanah et al. 2022; Xu et al. 2020; Hou et al. 2024). Recently, researchers have developed methods to enhance the zero-shot capabilities of CLIP by utilizing class-specific descriptions generated from large language models (LLMs). These methods demonstrate how a pretrained language model can create improved language prompts for open vocabulary tasks. In these studies, hand-crafted prompts are used to query the LLM, generating visual and distinguishing attributes for the classes within the respective datasets.

CuPL (Pratt et al. 2023) was one of the earliest works to demonstrate that the prompts generated by this method can achieve superior performance on zero-shot image classification benchmarks. LaFTer (Mirza et al. 2024b) trains a classifier on the embeddings of generated texts for zero-shot classes, which can also be applied to image features. ProText (Khattak et al. 2024) utilizes a text-only supervision approach to effectively learn prompts, leveraging the capabilities of large language models (LLMs). MetaPrompting (Mirza et al. 2024a) introduces an automatic prompt generation technique to generate high-level textual information to find a way to generate diverse category-level prompts for the zero-shot classification task. AdaptCLIPZS (Saha, Van Horn, and Maji 2024) proposes fine-grained labeling by pairing images with coarse-level descriptions which emphasizes key attributes of classes to bridge the gap between the image-level captions and generalized information of the category object, resulting in generalization to several tasks. WaffleCLIP (Roth et al. 2023) proposes to introduce random descriptors for zero-shot accuracy.

Pseudo Labelling/ Semi-Supervised Learning

Pseudo-labeling or semi-supervised learning is a powerful machine learning method to generate pseudo-labels of a large amount of data without requiring a large amount of labels (Sohn et al. 2020; Hoyer et al. 2023; Berthelot et al. 2019). It mitigates the requirement of labeled data by selecting confident ones to train models. Inn order to boost the performance of semi-supervised learning, recent works leverage both pseudo-labeling and consistency regularization to benefit from similar predictions between the two different views of an image (Kurakin et al. 2020; Li, Li, and Wang 2023). (Wei and Gan 2023) propose an adaptive consistency regularizer (ACR) method to handle the semi-supervised learning for the long-tailed classification problem. Further, the pseudo-labeling was extended to utilize augmentation (Nguyen and Yang 2023) and consistency regularization (Yan et al. 2024). (Li et al. 2023b) presents a novel open-set semi-supervised framework exploiting both inliers and outliers when they are hard to distinguish. FixMatch (Sohn et al. 2020) combined consistency regularization to estimate the pseudo-label using a high-confidence prediction. It was later extended via non-parametrically predicting view assignments with support samples (Assran et al. 2021).

Prompt Learning

Over the years, machine-learning approaches generally focused on fully supervised learning, which employs task-specific models that are exclusively trained on instances with labels relevant to the target task (Krizhevsky, Sutskever, and Hinton 2012; Alom et al. 2018). In the recent era of foundation models, the learning paradigms have undergone considerable modernization and are moving away from fully supervised learning to a *pre-training* and *fine-tuning* learning frameworks for the downstream tasks (Zhou et al. 2022a,b; Gao et al. 2024). These approaches leverage the models to acquire generalized feature learning during the pre-training and do not require exclusively adapting the model to downstream tasks. On the contrary, researchers are redesigning the inputs using prompts to revamp the downstream task ensuring that it corresponds with the original pre-training task (Lester, Al-Rfou, and Constant 2021; Lu et al. 2022). Prompt learning has shown the promising potential to minimize semantic discrepancies and bridge the gap between pre-training and fine-tuning to overcome the issues related to the overfitting problem (Lu et al. 2022; Liu et al. 2024; Khattak et al. 2023; Lee et al. 2023). CoOP(Zhou et al. 2022b) proposes that the prompt vectors by employing the cross-entropy loss can reduce the prediction error. CoCoOp (Zhou et al. 2022a) introduce which generates image-adaptive prompts resulting in enhanced generalization to the distribution shifts. The unsupervised prompt learning (UPL) approach (Huang, Chu, and Wei 2022) avoids the prompt engineering. Whereas, TPT optimizes the prompt by minimizing the entropy with confidence selection (Shu et al. 2022) by introducing a test-time prompt learning framework.

Methodology

We first provide an overview of CLIP and prompt learning. We then introduce our framework, No Labels Attached (NoLA) tuning and provide a detailed explanation of how we apply our method, combining the strengths of VLMs and pre-trained self-supervised learned vision backbones, for improved performance.

Preliminaries

Contrastive Language-Image Pre-training (CLIP): CLIP comprises two parallel encoders, mapping the visual and textual inputs into feature vectors in the joint embedding space. Here, the CLIP image and text encoders are denoted

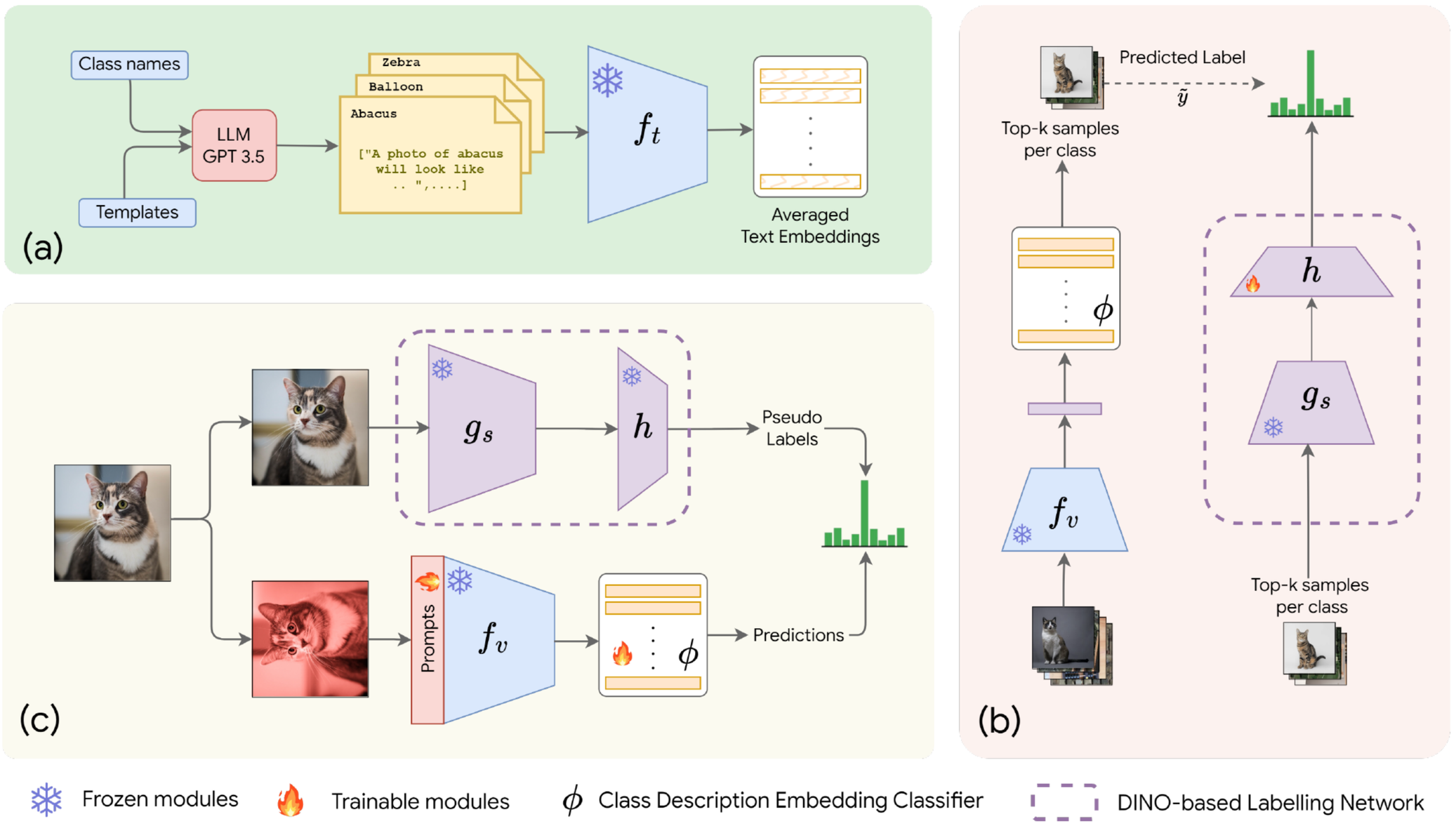

Figure 2: **Overview of proposed *NoLA* (No Labels Attached) method,** (a) A set of templates and the class names are fed through an LLM to generate context-enriched text descriptions per class. The description embeddings obtained from the CLIP text encoder $\boldsymbol{f_t}$ are averaged to compose the class description-based embedding (CDE) classifier $\boldsymbol{\phi}$. (b) Zero-shot inference is obtained for the training set by using the CLIP vision encoder $\boldsymbol{f_v}$ and the CDE classifier. From the predictions, top-k confident training samples are selected to train the alignment module h which utilizes a self-supervised learned (SSL) $\boldsymbol{g_s}$ backbone (DINO). (c) The DINO-based labelling network consisting of the alignment module h is then used to generate pseudo labels and learn dataset specific visual prompts which are prepended to the frozen CLIP vision encoder.

by $\mathcal{F}_v$ and $\mathcal{F}_t$ respectively, and their pre-trained parameters are represented by $\theta_{\texttt{CLIP}} = \{\theta_v, \theta_t\}$ respectively. An input image I is converted to M patches, which are projected to produce patch tokens, and a class token CLS is prepended to it, resulting in $\boldsymbol{X}_0 = \{\texttt{CLS}, \boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_M\}$ where e_i is the embedding of the i^{th} patch. The image encoder $\mathcal{F}_v$ encodes the input patches via transformer blocks to produce a latent visual feature representation $\boldsymbol{f_v} = \mathcal{F}_v(\boldsymbol{X}_0; \theta_v)$. The corresponding class label y is embedded within a text template or description, such as '`a photo of a` $<$ CLS $>$', which is tokenized to form $\boldsymbol{Y}_0$. The text encoder $\mathcal{F}_t$ encodes $\boldsymbol{Y}_0$ through transformer blocks to compute the latent textual feature as $\boldsymbol{f_t} = \mathcal{F}_t(\boldsymbol{Y}_0; \theta_t)$. At zero-shot inference, the similarity of each text feature with class labels $y = \{1, 2, \cdots, C\}$ is computed with that of the image feature as $\boldsymbol{s_i} = \texttt{sim}(\boldsymbol{f_{t_i}} \cdot \boldsymbol{f_v})$, where $\texttt{sim}(.)$ denotes the cosine similarity, $\boldsymbol{s_i}$ denotes the similarity score of i^{th} class with the text feature $\boldsymbol{f_{t_i}}$. The prediction probability $p(y_i|X)$ on X can be defined as:

$$p(y_i|X) = \frac{\texttt{exp}(\texttt{sim}(\boldsymbol{f_t} \cdot \boldsymbol{f_v})\tau)}{\sum_{i=1}^{C} \texttt{exp}(\texttt{sim}(\boldsymbol{f_t} \cdot \boldsymbol{f_v})\tau)}, \tag{1}$$

where τ is the temperature of the softmax function.

Prompt Learning: CLIP contains a plethora of knowledge leveraged from training on millions of noisy image-text pairs. To effectively extract the rich features learned by the CLIP model, recent approaches (Zhou et al. 2022b,a; Khattak et al. 2023; Lu et al. 2022) append extra learnable prompts while keeping image and text encoders frozen. These prompts modify the context of the model input without distorting the pre-trained CLIP features. We prepend visual prompts on the vision encoder of CLIP, denoted by $\theta_{\texttt{P}}$.

Motivation

While CLIP's native zero-shot classification capability shows promising results without requiring training data, it is usually surpassed by networks trained on data specific to the target domain. Existing methods (Zhou et al. 2022b,a; Saha, Van Horn, and Maji 2024) narrow this performance gap through fine-tuning the zero-shot classifier in a few-shot setting. While improving accuracy they also incur additional costs associated with curating and annotating training data.

Additionally, the features extracted using the pretrained CLIP vision encoder are often sub-optimal for many fine-

grained visual recognition tasks, as they do not effectively discriminate between the distinguishing characteristics of similar-looking image categories. While there are models that demonstrate superior feature extraction capabilities—specifically, those trained in a self-supervised manner like DINO (Caron et al. 2021), SimCLR(Chen et al. 2020) —these models often require labeled datasets to learn a linear probing adapter for downstream datasets. We aim to leverage the rich visual features learned in self-supervised models, particularly DINO pretrained on ImageNet (Deng et al. 2009), to adapt large VLMs such as CLIP by utilizing only unlabeled training images. DINO pre-trained on ImageNet is particularly advantageous because it captures a wide range of visual features across diverse categories, enabling more effective adaptation to fine-grained tasks.

Overview

An overview of the proposed approach is shown in Figure 2. which comprises the following three components (i) A class description-based embedding (CDE) classifier that builds textual embeddings using class descriptions prompted through an LLM, enriched by their vast knowledge base (As shown in Figure 2.-(a)). (ii) We then use a stronger SSL pre-trained visual backbone such as DINO and align it to the joint embedding space of the VLM to be used as the auto-labelling network (As shown in Figure 2.-(b)). (iii) This stronger SSL pre-trained visual backbone is then used for the DINO-assisted prompt learning in the vision encoder (As shown in Figure 2.-(c)). Next, we provide detailed explanation of these three key components.

Class Description based Embedding (CDE) classifier: The generic vision-language model setting uses their text encoder to build the classifier to classify the image embedding, given the class names of the target dataset. We compose the classifier by generating finer descriptions catered towards the target dataset by prompting an LLM model, a technique adopted from (Pratt et al. 2023). We prompt the LLM with the class names and N template questions, specific to the target dataset as shown in Figure 2.-(a). This generates K descriptions $\boldsymbol{\omega^C}$ for each class C, enriched by the domain knowledge of the LLM, giving $K \times C$, class descriptions. The class description based embedding classifier $\boldsymbol{\phi} \in \mathbb{R}^{C \times d}$, where d is the embedding dimension, can be formulated as follows:

$$\begin{aligned} \boldsymbol{\phi_C} &= \frac{1}{K} \sum_{i=1}^{K} \mathcal{F}_t(\boldsymbol{\omega}_i^{\boldsymbol{C}}; \theta_t) \\ \boldsymbol{\phi} &= \texttt{Concat}[\boldsymbol{\phi_1}, \boldsymbol{\phi_2}, \ldots, \boldsymbol{\phi_C}]. \end{aligned} \tag{2}$$

DINO-based Labelling (DL) Network: We seek to improve the visual embedding by utilizing a strong self-supervised pre-trained visual backbone $\boldsymbol{g_s}$. To this end, we make use of the LLM-enriched CDE classifier, to align a self-supervised learning (SSL) pre-trained vision backbone to the joint embedding space of the VLM.

In order to obtain the text-aligned visual embedding $\boldsymbol{h}$ for the target dataset, we first input the target image to a CLIP visual encoder $\boldsymbol{f_v}$ to output visual features. These features are further fed to the CDE classifier ($\boldsymbol{\phi_C}$) to obtain the top-k samples for each class $\boldsymbol{C}$, here k is proportional to the number of training images and the number of classes in the respective target dataset. It is only fair to select a higher value for k in datasets with more samples per class, and a lower value otherwise. Since our method is entirely label-free, we do not use information about the number of samples per class. Instead, we determine k using the data available to us: the number of training images and the number of classes. First, we calculate the average number of images per class by dividing the total number of training images by the number of classes. To account for inherent class imbalance that will be present in most datasets, we then select 20% of this average. The choice of 20% as the optimal percentage is justified by empirical analysis, which we present in the supplementary material. Additionally, if the calculated k is less than 16, we set k to 16, and if it exceeds 512, we cap k at 512. Later, the alignment module $\boldsymbol{h}$ is then optimized utilizing smoothed cross-entropy loss function (Szegedy et al. 2016), to obtain a DINO-based labelling (DL) network (comprising of $\boldsymbol{g_s}$ and $\boldsymbol{h}$), using the top-k samples per class, where all other components are frozen (Fig 2-(b)).

DINO-assisted prompt learning: In order to adapt the vision encoder of the VLM, we set up learnable visual prompt tokens $\theta_{\texttt{P}}$ to the vision encoder. Specifically, we append learnable V visual prompts with the visual input tokens. The image encoder processes the input to generate a prompted visual feature representation denoted as $\boldsymbol{f_v^p}$ can be represented as:

$$\boldsymbol{f_v^p} = \mathcal{F}_v(\boldsymbol{X}_0; \theta_v, \theta_{\texttt{P}}). \tag{3}$$

This facilitates a lightweight adaptation of the vision encoder as opposed to fine-tuning the vision encoder. To do so, motivated by Fixmatch (Sohn et al. 2020), we generate two separate views for each target input i.e., weak transformation as identity ($\boldsymbol{I_0}$) and strong augmentation ($\boldsymbol{I_s}$). This approach is effective because weak augmentation tends to preserve the intrinsic characteristics of the input data, facilitating the creation of pseudo-labels that are more reliable and, strong augmentation introduces perturbations that encourage the model to learn robust and invariant features, thus improving its generalization capability. Thus, striking a balance between ensuring label quality and improving the model's capacity to generalize to previously unseen data.

We now use the DL Network –a vision encoder with finer visual cues, aligned to the VLM embedding space– as an auto-labeller to train the visual prompts and the CDE classifier for the target dataset, through the training objective given in Eq. 4.

$$\min_{\theta_{\texttt{P}}, \phi} \mathcal{L}_{\texttt{SCE}} \Big(\boldsymbol{\phi}\big(\mathcal{F}_v(\boldsymbol{X}_s; \theta_v, \theta_{\texttt{P}})\big), \boldsymbol{h}\big(\boldsymbol{g_s}(\boldsymbol{X}_0; \theta_g)\big) \Big), \tag{4}$$

where SCE represents the smoothed cross-entropy loss function (Szegedy et al. 2016), the θ_g denotes the pre-trained parameters of $\boldsymbol{g_s}$. The $\boldsymbol{X_s}$ and $\boldsymbol{X_0}$ are the patchified input of strong and weak augmented images $\boldsymbol{I_s}$ and $\boldsymbol{I_0}$, respectively.

Through lightweight auto-labelled prompt tuning setting (Figure 2.-(c)), we harmonically combine the domain

knowledge distilled from the LLM using the CDE classifier and stronger visual cues from a pre-trained SSL encoder towards better performance for the label-free classification task.

Experiments

Datasets

We extensively evaluate our approach across 11 diverse datasets, each representing distinct domains. Among these, four datasets—ImageNet (Deng et al. 2009), CIFAR-10/100 (Krizhevsky, Hinton et al. 2009), and Caltech-101 (Fei-Fei, Fergus, and Perona 2006)—focus on common natural categories. EuroSAT (Helber et al. 2019) and RESISC45 (Cheng, Han, and Lu 2017), each containing 10 and 45 classes respectively, provide satellite imagery for geographical and environmental analysis. The UCF-101 dataset (Soomro, Zamir, and Shah 2012) is used for action recognition, while SUN-397 (Xiao et al. 2016) offers images from 397 naturally occurring scenes. Flowers-102 (Nilsback and Zisserman 2008) is a fine-grained classification dataset containing 102 different categories of flowers. The Describable Textures Dataset (DTD) (Cimpoi et al. 2014) comprises 47 categories of images, designed to study texture perception through various describable attributes. Lastly, the Oxford Pets (Parkhi et al. 2012) dataset features 37 categories of pet images, covering a range of cat and dog breeds. For all the datasets, we either use the splits provided by the author or, if unavailable, we use the split provided by (Zhou et al. 2022b).

Implementation Details

As discussed earlier, our pipeline includes three main stages. In the first step, we utilize the descriptions obtained from an LLM i.e., GPT3.5, we use the descriptions dataset obtained by (Pratt et al. 2023) in which they prompt with the class names of the target dataset and N dataset specific questions to generate K class-specific descriptions. For the obtained descriptions, we also add dataset-specific prompt templates provided by (Radford et al. 2021).

In the second stage, we build the DINO-based labelling (DL) network, using a self-supervised vision encoder aligned to the VLM joint embedding space. We keep the DINO ViT-B/16 (Caron et al. 2021), Imagenet pre-trained backbone, $\boldsymbol{g_s}$ (in Fig 2 (b)) frozen and train alignment module $\boldsymbol{h}$ on the target dataset. The choice of k value in the top-k samples to be selected is different for each dataset and it is propotional to the number of training images and number of categories in the dataset. The specifications, training details of the alignment module $\boldsymbol{h}$ and reasoning behind k value selection are mentioned in the supplementary material.

In the final DINO-assisted prompt learning stage, we include learnable prompts in the vision encoder of the VLM. The DINO-assisted prompt learning is performed using the AdamW (Kingma and Ba 2014) optimizer with a learning rate of $2e^{-3}$ and a batch size of 512. To obtain the augmented view of the image, we employ augmentations from SimSiam (Chen and He 2021): Gaussian blur, random resized crop, random horizontal flip, color jitter, random scaling, and, random perspective. All experiments are conducted using a single Nvidia A100 GPU.

Results and discussion

We evaluate the image classification performance of NoLA across the 11 datasets presented in Table 1 using ViT-B/32 (Dosovitskiy et al. 2020) CLIP variant. We compare the performace our method with six label-free methods CuPL (Pratt et al. 2023), MetaPrompt (Mirza et al. 2024a), LaFTer (Mirza et al. 2024b), ProText (Khattak et al. 2024), WaffleCLIP (Roth et al. 2023), and also CLIP zero-shot performance. We also conduct a quantitative analysis by comparing our method with CoOp (Zhou et al. 2022b), a dew-shot method. As seen in Figure 1, our approach demonstrates state-of-the-art performance in 9 out of 11 datasets when compared with label-free methods and even, outperforms few-shots methods on certain datasets.

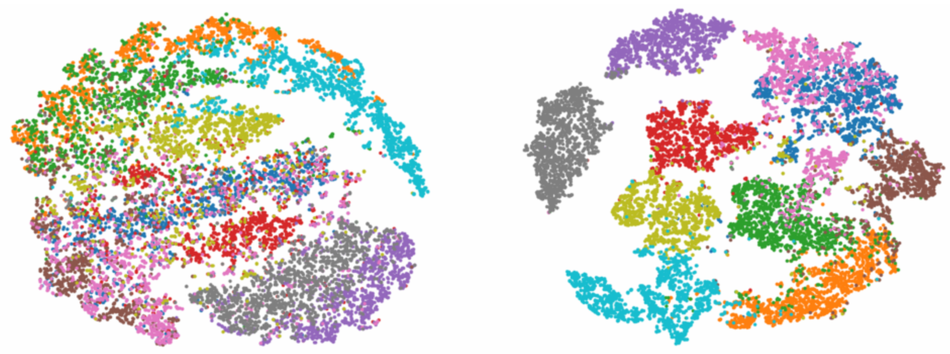

Figure 3: TSNE projections comparison of EuroSAT embeddings obtained from the base CLIP ViT-B/32 (left) and CLIP ViT-B/32 adapted with our NoLA framework (right).

In addition, in Figure 3, we compare visualizations of embeddings from base CLIP and our method. It is evident that our method produces better pronounced, discriminative clustered embeddings as same class features are closer to each other, while different class features are far apart.

Table 3: **Ablation of integration of individual components that makes up the NoLA framework.** The top-1 accuracy for each variant is averaged across six datasets for their respective test sets.

	Avg. Top-1 Acc.
CLIP zero-shot	67.9
(+) CDE classifier	72.0
(+) DL network	73.9
(+) DINO-assisted Prompt Learning	80.5

Ablation Study

We conduct an ablative study for proposed method NoLA. The experiments are conducted across six datasets namely, EuroSAT, Caltech101, OxfordPets, OxfordFlowers, SUN397, and CIFAR100 for all ablation experiments.

	Venue	ImageNet	EuroSAT	Caltech101	OxfordPets	UCF101	DTD	Flowers102	SUN397	RESISC45	CIFAR10	CIFAR100	Average
Few-Shot Methods													
CoOp (1-Shot) (Zhou et al. 2022b)	IJCV (‘22)	60.6	58.4	91.7	-	63.8	40.1	71.2	64.1	-	83	55.6	-
CoOp (5-Shot) (Zhou et al. 2022b)	IJCV (‘22)	61.3	71.8	93.2	-	74.3	41.1	85.8	67.3	-	86.6	63.2	-
CoOp (10-Shot) (Zhou et al. 2022b)	IJCV (‘22)	62.3	81.6	94.6	-	77.2	65.8	92.1	69	-	88.5	66.6	-
Label-Free Methods													
CLIP (Radford et al. 2021)	ICML (‘21)	61.9	40.6	90.5	85.0	61.0	42.9	66.6	60.8	49.8	88.8	64.2	64.7
CuPL (Pratt et al. 2023)	ICCV (‘23)	63.4	62.2	90.6	87.2	63.9	48.0	71.5	66.0	61.9	89.2	65.8	70.0
MetaPrompt (Mirza et al. 2024a)	ECCV (‘24)	65.0	55.6	92.9	88.1	67.9	50.8	73.9	67.0	64.0	89.9	66.3	71.0
LaFTer (Mirza et al. 2024b)	NeurIPS (‘24)	64.2	**73.9**	93.3	82.7	68.2	46.1	71.0	64.5	68.3	**95.8**	74.6	73.0
ProText (Khattak et al. 2024)	Arxiv	64.9	51.4	93.4	89.0	66.4	50.7	74.2	66.8	57.4	89.5	66.1	70.0
WaffleCLIP (Roth et al. 2023)	ICCV (‘23)	63.5	46.7	**94.8**	88.1	65.8	51.0	68.7	65.6	63.4	90.9	67.2	69.6
Ours		**65.4**	73.5	**94.8**	**89.3**	**68.3**	**56.1**	**82.7**	**67.0**	**75.4**	94.9	**75.6**	**76.6**

Table 1: Top-1 accuracy (%) for 11 datasets by using ViT-B/32 CLIP variant

	EuroSAT	Caltech101	OxfordPets	Flowers102	SUN397	CIFAR100	Average
AdaptCLIPZS (16-shot) (Saha, Van Horn, and Maji 2024)	81.8	95.3	93.7	81.3	72.1	73.9	83.0
LaFTer (Mirza et al. 2024b)	72.1	94.3	81.5	65.9	65.9	76.3	76.0
Ours	**79.1**	**95.3**	**91.7**	**84.3**	**69.3**	**77.5**	**82.9**

Table 2: We compare our methodology with CLIP ViT-B/16 variant across six datasets on few-shot (AdaptCLIPZS) and label-free (LaFTer) methods.

Using CLIP ViT-B/16 variant: Table 2 demonstrates that our methodology maintains strong performance with the CLIP ViT-B/16 variant. Additionally, our approach achieves results comparable to AdaptCLIPZS (Saha, Van Horn, and Maji 2024), a few-shot method utilizing 16 shots.

The different stages of NoLA: Table 3 summarizes the contribution of each stage-wise component in our proposed method using CLIP ViT-B/32 variant. As observed, the class-specific LLM knowledge distilled through the CDE provides a significant boost in performance with an increase of 4.1% against the zero-shot accuracy of CLIP. Next, the performance of the trained DL network shows a further improvement of 1.9%. This validates the importance of combining the strengths of VLMs and enriched visual features from a stronger visual backbone. A further boost of 6.6% in the performance is obtained through the DINO-assisted prompt learning method, adapting the vision encoder of the VLM using prompt learning, thus showcasing the significance of each stage.

Ablation of design choices: Table 4 illustrates two ablations where in ablation 1 shows the averaged Top 1 % accuracy obtained when the DINO vision encoder is replaced with CLIP vision encoder and ablation2 shows the averaged Top 1 % accuracy obtained when trained DL network is completely replaced with CDE classifier. While both these setups are able to get much better performance in accuracy when compared to CLIP zero-shot, the incorporation of a rich feature extractor like DINO and enriched class embeddings brings in a more nuanced understanding of visual features, leading to further improvements in NoLA's discriminative ability as shown in the overall accuracy.

Table 4: **Alternate design choices:** Ablation 1 refers to the ablation of using CLIP vision encoder in the DL network of (b) in Figure 2. Ablation 2 refers to the ablation of using the CDE classifier as the pseudo-labeller of (c) in Figure 2. The top-1 accuracy for each ablation is averaged across six datasets for their respective test sets.

Method	Avg. Top-1 Acc.
Ours: NoLA (using DINO based DL)	80.5
Ablation 1: Replace DINO with CLIP in DL	77.8
Ablation 2: Replace DL with CDE	76.2

Conclusion

In this work, we propose a label-free lightweight prompt tuning for vision language models. Particularly, we leverage knowledge from the Large Language Model (LLM) to build a class description embedding (CDE) classifier and use pseudo-labels from the CDE classifier to align an SSL pretrained vision encoder, DINO, to the vision-language joint embedding space, to build the DINO-based Labelling (DL) network. Finally, we employ our trained DL network as an auto-labeller to adapt the vision-language vision encoder

through prompt tuning. We perform extensive experiments over 11 popular image classification datasets and our study reveals that our framework, NoLA, performs favorably compared to existing VLMs-based state-of-the-art methods.

Supplementary Material: CLIP meets DINO for Tuning Zero-Shot Classifier using Unlabeled Image Collections

In this supplementary, we provide,

- Ablation of trainable components in NoLA
- Ablation of using GPT-4o descriptions
- Analysis on k (number of confident pseudo-labels per class) selection
- Implementation of class description embedding (CDE) classifier
- Additional implementation details

All ablations and experiments in this supplementary material are conducted using the ViT-B/32 CLIP variant unless specified otherwise.

Ablation of trainable components in NoLA

We summarize the findings in Table 5 to evaluate the impact of different trainable components when adapting the vision encoder through DINO-assisted prompt learning. We observe that the combined training of both the visual prompts and the learnable CDE (as shown in Fig 2-(c) in the main paper) yields a reasonable improvement compared to making only one of these components trainable (Settings 2 or 3). This enhancement can be attributed to the synergistic benefits of visual adaptation via prompts and the domain knowledge captured by the LLM-derived CDE classifier.

Table 5: **Ablation of the trainable components in NoLA.** The Top-1 accuracy is averaged across six datasets, namingly, EuroSAT, Caltech101, Oxford-pets, Flowers-102, SUN397, CIFAR100.

Trainable components →	Prompts	CDE	Avg. Top-1 Acc.
Setting 1	✗	✗	77.9
Setting 2	✓	✗	78.2
Setting 3	✗	✓	79.6
NoLA	✓	✓	**80.5**

Ablation of using GPT-4o descriptions

While we utlize the descriptions dataset obtained from (Pratt et al. 2023), which is generated using GPT3.5, we also experiment the performance of our framework when paired with richer descriptions generated from GPT-4o. The prompts used to generate the descriptions are provided in the Appendix. The findings of this ablation is presented in Table 6, where we compared Top-1 accuracy obtained for the six datasets which were used in the ablations. This comparison allows us to assess the impact of more detailed and contextually rich descriptions on the overall performance of our framework. Notably, GPT-4o descriptions achieve an average accuracy that is 0.64 higher than with GPT-3.5.

Table 6: Comparison of NoLA's performance with GPT-3.5 descriptions and GPT-4o descriptions over six datasets.

LLM	EuroSAT	Caltech101	Oxford-pets	Flowers-102	SUN397	CIFAR100	Average
GPT-3.5	73.47	94.84	**89.34**	82.70	**67.02**	**75.62**	80.50
GPT-4o	**74.75**	**95.01**	88.66	**85.91**	66.69	75.20	**81.04**

Analysis on k selection

The alignment module h within the DL network is trained on samples selected using the CDE classifier, making the number of confident pseudo-labels per class, k, crucial for training. (Pantazis et al. 2022; Huang, Chu, and Wei 2022) argued that choosing k as 16 is optimal for any dataset. However, we hypothesize, for a dataset which has a higher number of images per class, it's reasonable to select a higher number of pseudo-labels. To explore this, we analyzed the impact of different values of k on DL network performance.

Since our method is entirely label-free, we cannot directly use the information on the number of training images available for each class. Instead, we estimate the average number of images per class by dividing the total number of training images by the number of classes. However, this estimate may not accurately represent the true distribution due to the imbalanced nature of datasets in the wild. To account for the long-tailed distributions, we select only a proportion of this estimated value. We experiment with different proportions of the average number of images per class to determine the optimal proportion. Specifically, we test selecting between 10% and 30% with 5% increments.

As shown in Figure 4 (bottom), for smaller datasets, setting k to 16 yields better performance (Pantazis et al. 2022; Huang, Chu, and Wei 2022). In contrast, for larger datasets, we find that k set to 16 is suboptimal. Empirically, setting k to around 20% of the average number of images per class achieves better accuracy (see Figure 4 - top). Thus, we adopt the following strategy: if 20% of the average number of images per class is less than 16, we select 16 confident samples. Otherwise, we select 20% of the confident samples, with a cap of 512 if the number exceeds this limit.

Implementation of class description embedding (CDE) classifier

We develop a class description embedding (CDE) classifier, enhanced by descriptions generated from the extensive knowledge base of large language models (LLMs), a technique we adopt from (Pratt et al. 2023). For a dataset with N classes, for any given class $n \in \{1, \cdots, N\}$ we create textual descriptions $\{T_{n,m}\}_{m=1}^{M}$, where $T_{n,m}$ denotes the m^{th} description of the n^{th} class. These descriptions capture

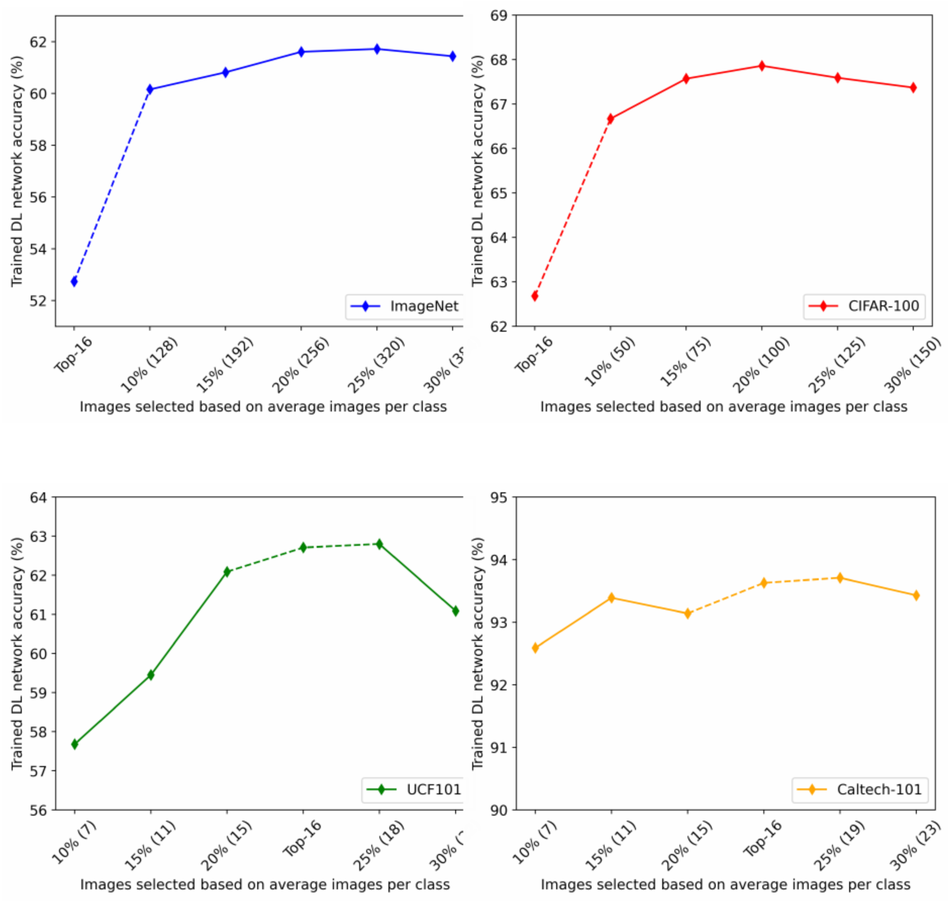

Figure 4: **Top-1 Accuracy of trained DL network with different values for k.** The top row shows the performance of different values for k in large datasets, namely, ImageNet (top-left) and CIFAR-100 (top-right). The bottom row shows the performance of different values for k in small datasets, namely, UCF101 (bottom-left) and Caltech101 (bottom-right). The value inside the parentheses on the x-axis represents the number of pseudo labels selected according to the specified percentage.

a wide range of semantic information, enriching the classifier's ability to distinguish between different classes.

For each description $T_{n,m}$, we find the corresponding text embedding $\phi_{n,m} = \mathcal{F}_t(T_{n,m})$ and with this an average text embedding is computed for each class n as,

$$\boldsymbol{\phi_n} = \frac{1}{M}\sum_{i=1}^{M}\phi_{n,m} \tag{5}$$

Finally the CDE classifier ϕ is constructed by concatinating the classwise average text embeddings as, $\boldsymbol{\phi} = \texttt{concat}[\boldsymbol{\phi_1}, \boldsymbol{\phi_2}, \ldots, \boldsymbol{\phi_N}]$.

Additional implementation details

Table 7 provides detailed hyperparameters for training the alignment module $\boldsymbol{h}$ (as shown in Fig 2-(b) of the main paper) within the DINO-based labeling (DL) network. Similarly, Table 8 presents the hyperparameters used for the DINO-assisted prompt learning stage (as shown in Fig 2-(c) of the main paper).

Table 7: **The list of hyperparameters used to optimize alignment module h within the DL network.**

Hyperparameters	Value
GPU	Nvidia A100 80GB
Backbone	Pretrained DINO ViT B/16
Pretrained	ImageNet
Input Size	224x224
Epochs	50
Optimizer	AdamW
Learning Rate	$1e^{-3}$
Batch Size	32
Samples trained on	Top-k strategy
Loss	Smoothed Cross-Entropy

Table 8: **The list of hyperparameters used for DINO-assisted prompt learning stage in the NoLA framework.**

Hyperparameters	Value
GPU	Nvidia A100 80GB
Backbone	ViT B/32
Input Size	224x224
Prompt Tuning Method	VPT(Jia et al. 2022)
Learnable Tokens	16
Batch Size	512
Optimizer	Adam
Learning Rate	$4e^{-3}$
Loss	Smoothed Cross Entropy

References

Akiva, P.; Purri, M.; and Leotta, M. 2022. Self-supervised material and texture representation learning for remote sensing tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8203–8215.

Alom, M. Z.; Taha, T. M.; Yakopcic, C.; Westberg, S.; Sidike, P.; Nasrin, M. S.; Van Esesn, B. C.; Awwal, A. A. S.; and Asari, V. K. 2018. The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164*.

Assran, M.; Caron, M.; Misra, I.; Bojanowski, P.; Joulin, A.; Ballas, N.; and Rabbat, M. 2021. Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8443–8452.

Bazi, Y.; Al Rahhal, M. M.; Mekhalfi, M. L.; Al Zuair, M. A.; and Melgani, F. 2022. Bi-modal transformer-based approach for visual question answering in remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60: 1–11.

Berthelot, D.; Carlini, N.; Goodfellow, I.; Papernot, N.; Oliver, A.; and Raffel, C. A. 2019. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties

in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9650–9660.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A Simple Framework for Contrastive Learning of Visual Representations. arXiv:2002.05709.

Chen, X.; and He, K. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15750–15758.

Chen, Y.; Huang, J.; Li, X.; Xiong, S.; and Lu, X. 2023. Multiscale Salient Alignment Learning for Remote Sensing Image-Text Retrieval. *IEEE Transactions on Geoscience and Remote Sensing*.

Cheng, G.; Han, J.; and Lu, X. 2017. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10): 1865–1883.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; and Vedaldi, A. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3606–3613.

Cozzolino, D.; Poggi, G.; Corvi, R.; Nießner, M.; and Verdoliva, L. 2024. Raising the Bar of AI-generated Image Detection with CLIP. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4356–4366.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Eldele, E.; Ragab, M.; Chen, Z.; Wu, M.; Kwoh, C.-K.; Li, X.; and Guan, C. 2023. Self-supervised contrastive representation learning for semi-supervised time-series classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Fei-Fei, L.; Fergus, R.; and Perona, P. 2006. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4): 594–611.

Gao, P.; Geng, S.; Zhang, R.; Ma, T.; Fang, R.; Zhang, Y.; Li, H.; and Qiao, Y. 2024. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2): 581–595.

Helber, P.; Bischke, B.; Dengel, A.; and Borth, D. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7): 2217–2226.

Hou, W.; Chen, S.; Chen, S.; Hong, Z.; Wang, Y.; Feng, X.; Khan, S.; Khan, F. S.; and You, X. 2024. Visual-Augmented Dynamic Semantic Prototype for Generative Zero-Shot Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23627–23637.

Hoyer, L.; Tan, D. J.; Naeem, M. F.; Van Gool, L.; and Tombari, F. 2023. SemiVL: Semi-Supervised Semantic Segmentation with Vision-Language Guidance. *arXiv preprint arXiv:2311.16241*.

Huang, T.; Chu, J.; and Wei, F. 2022. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q.; Sung, Y.-H.; Li, Z.; and Duerig, T. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, 4904–4916. PMLR.

Jia, M.; Tang, L.; Chen, B.-C.; Cardie, C.; Belongie, S.; Hariharan, B.; and Lim, S.-N. 2022. Visual prompt tuning. In *ECCV*, 709–727. Springer.

Joo, H. K.; Vo, K.; Yamazaki, K.; and Le, N. 2023. Clip-tsa: Clip-assisted temporal self-attention for weakly-supervised video anomaly detection. In *2023 IEEE International Conference on Image Processing (ICIP)*, 3230–3234. IEEE.

Khattak, M. U.; Naeem, M. F.; Naseer, M.; Van Gool, L.; and Tombari, F. 2024. Learning to Prompt with Text Only Supervision for Vision-Language Models. *arXiv preprint arXiv:2401.02418*.

Khattak, M. U.; Rasheed, H.; Maaz, M.; Khan, S.; and Khan, F. S. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19113–19122.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koçyiğit, M. T.; Hospedales, T. M.; and Bilen, H. 2023. Accelerating Self-Supervised Learning via Efficient Training Strategies. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 5654–5664.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Kurakin, A.; Raffel, C.; Berthelot, D.; Cubuk, E. D.; Zhang, H.; Sohn, K.; and Carlini, N. 2020. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring.

Lee, Y.-L.; Tsai, Y.-H.; Chiu, W.-C.; and Lee, C.-Y. 2023. Multimodal Prompting with Missing Modalities for Visual Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14943–14952.

Lester, B.; Al-Rfou, R.; and Constant, N. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 12888–12900. PMLR.

Li, M.; Li, Q.; and Wang, Y. 2023. Class Balanced Adaptive Pseudo Labeling for Federated Semi-Supervised Learning.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16292–16301.

Li, X.; Wen, C.; Hu, Y.; and Zhou, N. 2023a. Rs-clip: Zero shot remote sensing scene classification via contrastive vision-language supervision. *International Journal of Applied Earth Observation and Geoinformation*, 124: 103497.

Li, Z.; Qi, L.; Shi, Y.; and Gao, Y. 2023b. IOMatch: Simplifying open-set semi-supervised learning with joint inliers and outliers utilization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15870–15879.

Liang, F.; Wu, B.; Dai, X.; Li, K.; Zhao, Y.; Zhang, H.; Zhang, P.; Vajda, P.; and Marculescu, D. 2023. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7061–7070.

Liu, C.; Zhang, W.; Lin, X.; Zhang, W.; Tan, X.; Han, J.; Li, X.; Ding, E.; and Wang, J. 2023. Ambiguity-Resistant Semi-Supervised Learning for Dense Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15579–15588.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2024. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Lu, Y.; Liu, J.; Zhang, Y.; Liu, Y.; and Tian, X. 2022. Prompt distribution learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5206–5215.

Mirza, M. J.; Karlinsky, L.; Lin, W.; Doveh, S.; ; Micorek, J.; Kozinski, M.; Kuhene, H.; and Possegger, H. 2024a. Meta-Prompting for Automating Zero-shot Visual Recognition with LLMs. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Mirza, M. J.; Karlinsky, L.; Lin, W.; Possegger, H.; Kozinski, M.; Feris, R.; and Bischof, H. 2024b. Lafter: Label-free tuning of zero-shot classifier using language and unlabeled image collections. *Advances in Neural Information Processing Systems*, 36.

Naeem, M. F.; Khan, M. G. Z. A.; Xian, Y.; Afzal, M. Z.; Stricker, D.; Van Gool, L.; and Tombari, F. 2023a. I2mvformer: Large language model generated multi-view document supervision for zero-shot image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15169–15179.

Naeem, M. F.; Xian, Y.; Zhai, X.; Hoyer, L.; Van Gool, L.; and Tombari, F. 2023b. Silc: Improving vision language pretraining with self-distillation. *arXiv preprint arXiv:2310.13355*.

Nguyen, K.-B.; and Yang, J.-S. 2023. Boosting Semi-Supervised Learning by bridging high and low-confidence predictions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1028–1038.

Nilsback, M.-E.; and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, 722–729. IEEE.

Pan, C.; Yaman, B.; Velipasalar, S.; and Ren, L. 2024. Clip-bevformer: Enhancing multi-view image-based bev detector with ground truth flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15216–15225.

Pantazis, O.; Brostow, G.; Jones, K.; and Mac Aodha, O. 2022. Svl-adapter: Self-supervised adapter for vision-language pretrained models. *arXiv preprint arXiv:2210.03794*.

Park, S.; and Van Hentenryck, P. 2023. Self-supervised primal-dual learning for constrained optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4052–4060.

Parkhi, O. M.; Vedaldi, A.; Zisserman, A.; and Jawahar, C. 2012. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, 3498–3505. IEEE.

Pourpanah, F.; Abdar, M.; Luo, Y.; Zhou, X.; Wang, R.; Lim, C. P.; Wang, X.-Z.; and Wu, Q. J. 2022. A review of generalized zero-shot learning methods. *IEEE transactions on pattern analysis and machine intelligence*, 45(4): 4051–4070.

Pratt, S.; Covert, I.; Liu, R.; and Farhadi, A. 2023. What does a platypus look like? generating customized prompts for zero-shot image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15691–15701.

Qiu, C.; Yu, A.; Yi, X.; Guan, N.; Shi, D.; and Tong, X. 2022. Open Self-Supervised Features for Remote-Sensing Image Scene Classification Using Very Few Samples. *IEEE Geoscience and Remote Sensing Letters*, 20: 1–5.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Roth, K.; Kim, J. M.; Koepke, A.; Vinyals, O.; Schmid, C.; and Akata, Z. 2023. Waffling around for performance: Visual classification with random words and broad concepts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15746–15757.

Saha, O.; Van Horn, G.; and Maji, S. 2024. Improved Zero-Shot Classification by Adapting VLMs with Text Descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17542–17552.

Schiappa, M. C.; Rawat, Y. S.; and Shah, M. 2023. Self-supervised learning for videos: A survey. *ACM Computing Surveys*, 55(13s): 1–37.

Shu, M.; Nie, W.; Huang, D.-A.; Yu, Z.; Goldstein, T.; Anandkumar, A.; and Xiao, C. 2022. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems*, 35: 14274–14289.

Sohn, K.; Berthelot, D.; Carlini, N.; Zhang, Z.; Zhang, H.; Raffel, C. A.; Cubuk, E. D.; Kurakin, A.; and Li, C.-L. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33: 596–608.

Soomro, K.; Zamir, A. R.; and Shah, M. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Stojnic, V.; and Risojevic, V. 2021. Self-supervised learning of remote sensing scene representations using contrastive multiview coding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1182–1191.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wei, T.; and Gan, K. 2023. Towards Realistic Long-Tailed Semi-Supervised Learning: Consistency Is All You Need. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3469–3478.

Wu, P.; Zhou, X.; Pang, G.; Zhou, L.; Yan, Q.; Wang, P.; and Zhang, Y. 2024. Vadclip: Adapting vision-language models for weakly supervised video anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 6074–6082.

Wysoczańska, M.; Ramamonjisoa, M.; Trzciński, T.; and Siméoni, O. 2024. Clip-diy: Clip dense inference yields open-vocabulary semantic segmentation for-free. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 1403–1413.

Xiao, J.; Ehinger, K. A.; Hays, J.; Torralba, A.; and Oliva, A. 2016. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision*, 119: 3–22.

Xu, W.; Xian, Y.; Wang, J.; Schiele, B.; and Akata, Z. 2020. Attribute prototype network for zero-shot learning. *Advances in Neural Information Processing Systems*, 33: 21969–21980.

Yan, Z.; Wu, Y.; Qin, Y.; Han, X.; Cui, S.; and Li, G. 2024. Universal semi-supervised model adaptation via collaborative consistency training. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 872–882.

Yao, L.; Huang, R.; Hou, L.; Lu, G.; Niu, M.; Xu, H.; Liang, X.; Li, Z.; Jiang, X.; and Xu, C. 2021. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*.

You, K.; Gu, J.; Ham, J.; Park, B.; Kim, J.; Hong, E. K.; Baek, W.; and Roh, B. 2023. Cxr-clip: Toward large scale chest x-ray language-image pre-training. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 101–111. Springer.

Yu, J.; Wang, Z.; Vasudevan, V.; Yeung, L.; Seyedhosseini, M.; and Wu, Y. 2022. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*.

Yuan, L.; Chen, D.; Chen, Y.-L.; Codella, N.; Dai, X.; Gao, J.; Hu, H.; Huang, X.; Li, B.; Li, C.; et al. 2021a. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*.

Yuan, Y.; Zhan, Y.; and Xiong, Z. 2023. Parameter-Efficient Transfer Learning for Remote Sensing Image-Text Retrieval. *IEEE Transactions on Geoscience and Remote Sensing*.

Yuan, Z.; Zhang, W.; Rong, X.; Li, X.; Chen, J.; Wang, H.; Fu, K.; and Sun, X. 2021b. A lightweight multi-scale cross-modal text-image retrieval method in remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 60: 1–19.

Zhang, X.; Xu, M.; Qiu, D.; Yan, R.; Lang, N.; and Zhou, X. 2024. MediCLIP: Adapting CLIP for Few-shot Medical Image Anomaly Detection. *arXiv preprint arXiv:2405.11315*.

Zhao, Z.; Liu, Y.; Wu, H.; Li, Y.; Wang, S.; Teng, L.; Liu, D.; Li, X.; Cui, Z.; Wang, Q.; et al. 2023. Clip in medical imaging: A comprehensive survey. *arXiv preprint arXiv:2312.07353*.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022a. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16816–16825.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9): 2337–2348.

Zhu, W.; Liu, J.; and Huang, Y. 2023. Hnssl: Hard negative-based self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4777–4786.

APPENDIX: *CLIP meets DINO for Tuning Zero-Shot Classifier using Unlabeled Image Collections*

Below we have listed the prompts used to generate descriptions for various datasets from GPT-4o.

ImageNet, Caltech101, CIFAR10, CIFAR100, SUN397

Prompts
Describe what a(n) {} looks like
What does a(n) {} look like?
What characteristics can be used to differentiate a(n) {} from others based on just a photo?
Describe an image from the internet of a(n) {}
A caption of an image of a(n) {}:
List how one can recognize the a(n) {} within an image.
List the distinguishing features of the a(n) {}.
List the visual cues that help in identifying the a(n) {}.
List the visual characteristics that make the a(n) {} easily identifiable.
List how one can identify the a(n) {} based on visual cues.

EuroSAT, RESISC45

Prompts
Describe a satellite photo of a(n) {}
Describe a(n) {} as it would appear in an aerial image
How can you identify a(n) {} in an aerial photo?
Describe the satellite photo of a(n) {}
Describe an aerial photo of a(n) {}
List how one can recognize the a(n) {} within an aerial image.
List the distinguishing features of the a(n) {} in a satellite photo.
List the visual cues that help in identifying the a(n) {} in an aerial image.
List the visual characteristics that make the a(n) {} easily identifiable in a satellite image.
List how one can identify the a(n) {} based on visual cues in an aerial photo.

Flowers102

Prompts
Describe how to identify a(n) {}, a type of flower.
Describe a photo of a(n) {}, a type of flower.
What does a(n) {} flower look like?
List the distinguishing features of a(n) {} flower.
How can you recognize a(n) {} flower in a photo?
Describe the visual characteristics of a(n) {}, a flower of the {} category.
What visual cues help identify a(n) {} flower?

Oxford-Pets

Prompts
Describe what a pet a(n) {} looks like.
Describe a photo of a(n) {}, a type of pet.
Visually describe a(n) {}, a type of pet.
List the distinguishing features of a(n) {} pet.
How can you recognize a(n) {} pet in a photo?
Describe the visual characteristics of a pet a(n) {}.
What visual cues help identify a(n) {} pet?
Describe how to identify a pet a(n) {} in an image.