Learning to Prompt with Text Only Supervision for Vision-Language Models

Muhammad Uzair Khattak[1] Muhammad Ferjad Naeem[2] Muzammal Naseer[1]
Luc Van Gool[2] Federico Tombari[3,4]
[1]Mohamed bin Zayed University of AI [2]ETH Zurich [3]TU Munich [4]Google

Abstract

Foundational vision-language models such as CLIP are becoming a new paradigm in vision, due to their excellent generalization abilities. However, adapting these models for downstream tasks while maintaining their generalization remains a challenge. In literature, one branch of methods adapts CLIP by learning prompts using visual information. While effective, most of these works require labeled data which is not practical, and often struggle to generalize towards new datasets due to over-fitting on the source data. An alternative approach resorts to training-free methods by generating class descriptions from large language models (LLMs) and perform prompt ensembling. However, these methods often generate class specific prompts that cannot be transferred to other classes, which incur higher costs by generating LLM descriptions for each class separately. In this work, we propose to combine the strengths of these both streams of methods by learning prompts using only text data derived from LLMs. As supervised training of prompts is not trivial due to absence of images, we develop a training approach that allows prompts to extract rich contextual knowledge from LLM data. Moreover, with LLM contextual data mapped within the learned prompts, it enables zero-shot transfer of prompts to new classes and datasets potentially cutting the LLM prompt engineering cost. We perform evaluations in cross-dataset transfer setting where our method improves over prior ensembling works while being competitive to those utilizing labeled images.

1. Introduction

The Vision field is experiencing a new paradigm in its model-building approach with the emergence of foundational models [5, 10], which are large DNNs pre-trained on web-scale data. Among these, Vision-Language models (VLMs) such as CLIP [10] stand out as the latest highlights which leverage contrastive pre-training on massive image-text pairs from the internet. During pre-training, CLIP learns to align image-text samples in a shared feature space. This allows CLIP to encode open-vocabulary concepts and generalize well to zero-shot recognition tasks.

In literature, numerous techniques have been proposed to adapt CLIP for downstream recognition tasks. One branch of methods [12, 13] treat text prompts as learnable vectors and optimize them using task-specific objectives such as cross-entropy. As prompts are learned in the embedding space, this allows them to be used with classes and datasets beyond those on which they were trained on. While effective, most of these methods require annotated image labels to optimize the prompts which is often impractical, especially in real-world scenarios such as medical imaging, security, surveillance, etc. Moreover, these methods tend to overfit on few-shot source samples and struggle to retain CLIP's generalization, especially in cross-dataset settings. Alternatively, several methods [8, 9] use the training-free approach of prompt ensembling by leveraging the capabilities of Large Language Models (LLMs). Instead of using hand-crafted templates, these methods mine dataset or class-specific descriptors and captions from LLMs to enrich text features. These enriched features aim to better represent content that could occur in test images, leading to improvements over baseline CLIP. Although these methods do not require image information, the knowledge acquired from LLMs is mostly specific to each class and not directly transferable across unseen classes and datasets since no optimization is performed. Additionally, generating LLM descriptions for each concept separately incurs additional LLM serving and prompt engineering costs.

In this work, we present a new paradigm to improve CLIP's generalization. Our motivation comes from combining the strengths of prompt learning and prompt ensembling approaches while addressing their limitations. We introduce ProText: **Pro**mpt Learning with **Text**-Only Supervision. In contrast to previous methods, our approach instead proposes to learn prompts using text only data obtained from LLMs. As supervised training of prompts is not trivial due to image-free setting, we develop a novel training framework that allows prompts to learn and extract rich contextual knowledge from LLM data.

Our contributions are as follows: 1. We present a new approach for prompt learning in CLIP using text-only supervision. Our method, ProText harmonically combines the strengths of prompt learning and prompt ensembling methods to improve CLIP's generalization. 2. To optimize prompts with text-only data, we develop a training approach that allows prompts to learn a mapping by extracting contextual information from LLM data. 3. As LLM knowledge is mapped within the learned prompts, this enables prompts to be directly used with new classes and datasets potentially cutting the additional LLM serving and prompt engineering cost. 4. We validate the effectiveness of our method through extensive experiments. ProText improves the generalization of CLIP across cross-dataset transfer settings and fares competitive to approaches that explicitly use labeled image samples for training.

2. Method

Our proposed adaptation framework, ProText: **Prom**pt Learning with **Text** only supervision aims to address the challenges of existing approaches by learning *transferable* prompts without *relying* on images. Fig. 1 shows our ProText framework at the training and inference stages.

2.1. Preliminaries

Contrastive Language-Image Pre-training (CLIP). CLIP consist of an image encoder f and a text encoder g which maps image and text input into visual and textual feature respectively. We denote CLIP parameters as $\theta_{\texttt{CLIP}} = \{\theta_f, \theta_g\}$ where θ_f and θ_g refer to the image and text encoder parameters, respectively. Input image $\boldsymbol{X}$ is divided into M patches which are linearly projected to produce patch tokens and a learnable class token $\texttt{CLS}$ is prepended resulting in the final sequence as $\tilde{\boldsymbol{X}} = \{\texttt{CLS}, \boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_M\}$. The image encoder f encodes the input patches via multiple transformer blocks to produce a latent visual feature representation $\tilde{\boldsymbol{f}} = f(\tilde{\boldsymbol{X}}, \theta_f)$, where $\tilde{\boldsymbol{f}} \in \mathbb{R}^d$. Next, the corresponding class label y is embedded in a text template, such as '$\texttt{a photo of a [CLASS]}$' which can be formulated as $\tilde{\boldsymbol{Y}} = \{\text{SOS}, \boldsymbol{t}_1, \boldsymbol{t}_2, \cdots, \boldsymbol{t}_L, \boldsymbol{c}_k, \text{EOS}\}$. Here $\{\boldsymbol{t}_l|_{l=1}^{L}\}$ and $\boldsymbol{c}_k$ are the word embeddings corresponding to the text template and the label y, respectively while SOS and EOS are the learnable start and end token embeddings. The text encoder g encodes $\tilde{\boldsymbol{Y}}$ via multiple transformer blocks to produce the latent text feature as $\tilde{\boldsymbol{g}} = g(\tilde{\boldsymbol{Y}}, \theta_g)$, where $\tilde{\boldsymbol{g}} \in \mathbb{R}^d$. For zero-shot inference, text features of text template with class labels $\{1, 2, \cdots, C\}$ are matched with image feature $\tilde{\boldsymbol{f}}$ as $\frac{\texttt{exp}(\texttt{sim}(\tilde{\boldsymbol{g}} \cdot \tilde{\boldsymbol{f}})\tau)}{\sum_{i=1}^{C} \texttt{exp}(\texttt{sim}(\tilde{\boldsymbol{g}}_i \cdot \tilde{\boldsymbol{f}})\tau)}$, where $\texttt{sim}()$ denotes the cosine similarity and τ is the temperature.

Prompt Learning with CLIP. Being a parameter efficient tuning method, prompt learning has emerged as a popular technique to adapt vision-language models like CLIP. Since most of the model is kept frozen during adaptation, prompt learning aims to reduce overfitting. Learnable prompts are appended either at the image side [2], text encoder side [12, 13], or both sides. In this work, we learn hierarchical prompts at the text encoder named Deep Language Prompting (DLP) [6] formulated as follows.

T learnable language prompts $\boldsymbol{P_t} = \{\boldsymbol{p_t^1}, \boldsymbol{p_t^2}, \cdots, \boldsymbol{p_t^T}\}$ are appended with text input tokens, resulting in $\tilde{\boldsymbol{Y}}_{\boldsymbol{p}} = \{\text{SOS}, \boldsymbol{P_t}, \boldsymbol{t}_1, \boldsymbol{t}_2, \cdots, \boldsymbol{t}_L, \boldsymbol{c}_k, \text{EOS}\}$. The text encoder processes $\tilde{\boldsymbol{Y}}_{\boldsymbol{p}}$ and prompted text feature is obtained as $\tilde{\boldsymbol{g}}_{\boldsymbol{p}} = g(\tilde{\boldsymbol{Y}}_{\boldsymbol{p}}, \theta_g)$. We use deep prompting which learns hierarchical prompts at subsequent transformer blocks of text encoder. Visual feature $\tilde{\boldsymbol{f}}$ is obtained without utilizing learnable prompts. To adapt CLIP on image classification task on dataset $\mathcal{D}$, prompts $\boldsymbol{P_t}$ are optimized in a supervised fashion using labeled image samples with cross-entropy loss, $\mathcal{L}_{\text{CE}}$.

$$\mathcal{L}_{\text{CE}} = \arg\min_{\boldsymbol{P_t}} \mathbb{E}_{(\boldsymbol{X}, y) \sim \mathcal{D}} \, \mathcal{L}(\text{sim}(\tilde{\boldsymbol{f}}, \tilde{\boldsymbol{g}}_{\boldsymbol{p}}), y). \quad (1)$$

Prompt Ensembling with LLM descriptions. Several methods have recently proposed to adapt CLIP via training-free prompt ensembling techniques. We focus our comparison with a strong ensembling baseline CuPL [9]. Specifically, a Large Language Model $\mathcal{F}$ such as GPT-3 [3] is used to generate class-specific descriptions for class labels $\{1, 2, \cdots, C\}$ using queries such as *'How does a* CLASS *look like'*. Text features of the same class description are averaged together, which serves as the ensembled text features. Finally, zero-shot inference is performed with those ensembled text features.

2.2. Prompt Learning with Text-Only Supervision

Our work aims to address the Visual data dependency and transferability limitations of Image-supervised prompt learning and LLM-based prompt ensembling methods within a unified framework. Below we detail our strategy for curating text-to-text data via LLMs for training, followed by our text-only prompt learning framework.

Text-Only LLM data for Prompt Learning. As discussed in Sec. 2.1, optimizing prompts for downstream datasets typically requires image-labels pairs. Since we explicitly aim to bypass this requirement, we first leverage LLMs to curate text data for prompt learning which consists of text inputs and text outputs. Given a set of classes $\{c_i\}_{i=1}^{C}$, we prepare text inputs $\{L_{\text{inputs}}^i\}_{i=1}^{C}$ by wrapping each class name in a standard hand-written text template, $L_{\text{inputs}}^i =$ '$\texttt{a photo of a}$ c_i'. Next, we prepare text outputs corresponding to the L_{inputs}. Specifically, we query GPT-3 model to generate detailed descriptions for each class name c_i. Similar to CuPL [9], we prompt GPT-3 with different queries Q conditioned on class names such as *'How does a c_i look like?'* and *'How can you identify a c_i?"* to obtain text outputs, $L_{\text{outputs}}^i = \mathcal{F}(Q|c_i)$. Similar to [9], we generate M text outputs per query Q and use N different queries, resulting in $M \times N$ text outputs per class category. We associate all L_{outputs} with the corresponding single L_{inputs}

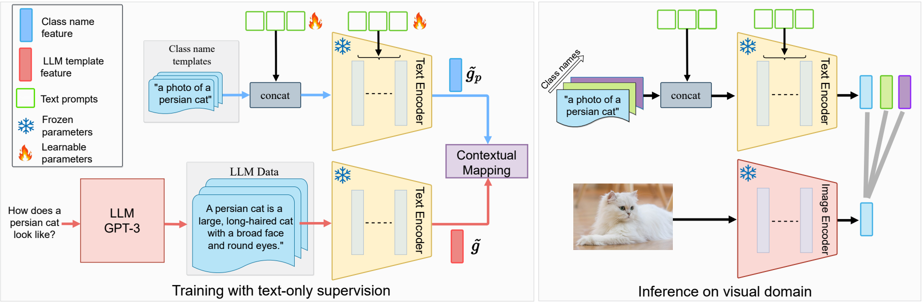

Figure 1. Overview of ProText framework. **(Left)** First, captions are generated for training classes using LLM. During training, CLIP text encoders generate **prompted class-name feature** ($\tilde{\boldsymbol{g}}_{\boldsymbol{p}}$) with learnable prompts and **frozen LLM template feature** ($\tilde{\boldsymbol{g}}$). Next, we employ contextual mapping loss to guide learnable prompts to learn a mapping from the prompted class-name feature to the LLM template feature. This allows the learned prompts to exploit internal knowledge of text encoder complemented by LLM descriptions. **(Right)** At inference, learned prompts are used with class-name templates. Moreover, rich contextual information from LLM descriptions mapped within the learned prompts enables its transferability to new classes and datasets.

for each class c_i. As LLMs are pre-trained on internet-scale text corpora, they possess the capability of generating very diverse and high-quality descriptions and captions for different class categories which results in high-quality text outputs. Finally we combine L_{inputs} and L_{outputs} to create LLM based text-to-text data for text only prompt learning, $\mathcal{D}_{\text{PROMPT}} = \{L^i_{\text{inputs}}, L^i_{\text{outputs}}\}_{i=1}^{M\times N\times C}$. We refer the readers to supplementary for additional details on the choice of LLM prompts and examples of $\mathcal{D}_{\text{PROMPT}}$.

Contextual mapping with Prompt Learning To leverage LLM text-to-text data $\mathcal{D}_{\text{PROMPT}}$ for learning transferable prompts, we propose a contextual mapping strategy that learns a mapping function that maps standard class name templates such as '`a photo of a` c_i' to the text feature generated from a LLM description which contains more information about the class c_i. In other words, contextual mapping allows learnable prompts to map L_{inputs} to L_{outputs} in the text space. The mapping function is realized in the form of learnable prompt vectors. For an i_{th} training sample from $\mathcal{D}_{\text{PROMPT}}$ consisting of a text-to-text pair $\{L_{\text{inputs}}, L_{\text{outputs}}\}_i$, we obtain prompted class-name feature $\tilde{\boldsymbol{g}}_p$ for L^i_{inputs} using learnable prompts and frozen LLM feature $\tilde{\boldsymbol{g}}$ for L^i_{outputs} without the prompt vectors within the pre-trained latent space of CLIP. We then impose a contextual mapping constraint between $\tilde{\boldsymbol{g}}_p$ and $\tilde{\boldsymbol{g}}$ text features as follows,

$$\mathcal{L}_{\text{mapping}} = \frac{1}{d}\sum_{i=1}^{d} ||\tilde{\boldsymbol{g}}_{\boldsymbol{p}} - \tilde{\boldsymbol{g}}||_2^2. \tag{2}$$

Motivation for $\mathcal{L}_{\text{mapping}}$. Contextual mapping objective allows learnable prompts to exploit internal knowledge of text encoder of CLIP to generate rich contextual features aligned with the LLM descriptions (L^i_{outputs}) for a given class. This strategy effectively learns prompts without using any visual information and when trained using all training classes together, it enables prompts to capture versatile and generalized context from the LLM descriptions.

Method	ImageNet Acc.
1: CLIP (ICML'21)	66.72
2: CLIP-Attribute	67.60
3: CLIP-80	68.32
4: DCLIP (ICLR'23)	68.03
5: Waffle CLIP (ICCV'23)	68.34
6: CuPL (ICCV'23)	69.62
7: ProText-Attribute	68.05
8: ProText-80	68.48
9: ProText-CuPL	**70.22**

Table 1. With same amount of text data, learning contextual prompts with text-only supervision improves CLIP performance in comparison to the prompt ensembling techniques.

3. Experiments

We perform comparisons in cross-dataset transfer settings.

Cross-dataset transfer. This setting evaluates the generalization ability of models trained on ImageNet-1k [4] source dataset by directly transferring it on cross-datasets.

Implementation details. We use a publically available pretrained ViT-B/16 CLIP model from OpenAI [10]. Refer to the supplementary material for additional details.

Effectiveness of Text-Only Supervision We first present an ablation to motivate our approach of learning prompts with text-only supervision. We train ProText with 3 types of text data and evaluate performance on ImageNet-1k [4]. ProText-Attribute uses 46 templates from [1]. ProText-80 is trained on standard 80 templates provided by CLIP [10] and ProText-CuPL is trained on class-specific LLM data employed by our baseline CuPL [9]. As shown in Tab. 1, prompt ensembling with attribute templates and 80 templates improves over CLIP single template. Among the LLM-based ensembling methods, CuPL provide highest performance of 69.62%. In contrast, ProText uses a learning-based approach and shows competitive performance against prompt ensembling methods using the same text data. When equipped with CuPL LLM text-data, ProText surpasses CuPL by 0.60% leading to highest performance. These results motivate our approach that instead of prompt ensembling, one can achieve competitive results by utilizing the same available text data to learn prompts.

	Source	Target										
	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	Aircraft	SUN397	DTD	EuroSAT	UCF101	*Average*
Methods utilizing labeled visual samples												
CoOp	**71.51**	93.70	89.14	64.51	68.71	85.30	18.47	64.15	41.92	46.39	66.55	63.88
Co-CoOp	71.02	94.43	90.14	65.32	71.88	86.06	22.94	**67.36**	45.73	45.37	68.21	65.74
MaPLe	70.72	93.53	90.49	65.57	72.23	86.20	24.74	67.01	46.49	48.06	68.69	66.30
PromptSRC	71.27	93.60	90.25	65.70	70.25	86.15	23.90	67.10	46.87	45.50	68.75	65.81
Zero-shot & LLM Prompt ensembling methods												
CLIP	66.72	92.98	89.13	65.29	71.30	86.11	**24.90**	62.59	44.56	47.84	66.83	65.15
CuPL	69.62	92.98	89.13	65.29	71.30	86.11	**24.90**	62.59	44.56	47.84	66.83	65.15
Prompt learning with text-only supervision												
ProText (Ours)	69.80	**94.81**	**91.01**	**66.00**	**72.35**	**86.66**	24.72	67.34	**47.93**	**51.86**	**69.60**	**67.23**

Table 2. Cross-dataset benchmark evaluation.

Method	ImageNet
1: CuPL-Mixtral-7x8B	64.69
2: ProText-Mixtral-7x8B	69.07
3: CuPL-GPT	69.62
4: ProText-GPT-3	**70.22**

Table 3. Choice of text data.

Method	ImageNet
1: Linear Adapter	69.36
2: MLP Adapter	69.24
3: LoRA (r=8)	70.19
4: Prompt Learning	**70.22**

Table 4. Mapping networks.

Cross-dataset transfer We next show results in cross-dataset transfer settings in Tab. 2. We assign CLIP results to CuPL for cross-datasets as class-specific ImageNet LLM prompts limit its transfer to new datasets. CuPL improves ImageNet performance of CLIP by ensembling ImageNet LLM prompts, while its cross-dataset results remain the same as CLIP. In contrast, ProText effectively addresses the transferability challenges of CuPL using generalized prompts trained with the same ImageNet LLM data. Since ProText allows generalization to unseen datasets, these learned prompts can directly be used with CLIP for cross-datasets leading to absolute average gains of +2.1% against CLIP and CuPL. We next compare ProText with 16-shot image-supervised methods. Without using any visual samples, ProText shows effective generalization on cross-datasets and surpasses previous best method MaPLe on 9/10 datasets leading to the highest average accuracy of 67.23%.

Ablations: Choice of LLM for generating text data. Similar to CuPL [9], ProText by default uses GPT-3 [3] descriptions. Here we ablate on a recent open-source Mixtral-8x7B LLM in Tab. 3. ProText improves over CuPL-Mixtral by 4.38% and fares competitively to ProText-GPT. **Contextual mapping network.** While ProText employs prompt learning to learn contextual mapping from LLM descriptions, here we ablate on other choices in Tab. 4. Adapters at the output of CLIP encoders perform relatively lower. ProText using LoRA provides 70.19% IN-Acc., suggesting ProText compatibility with recent finetuning methods.

4. Conclusion

Prompt learning and LLM-based ensembling are effective techniques to improve CLIP's generalization. However, prompt learning often requires labeled images, which is less practical, while LLM-based ensembling methods are dominantly class-specific and not directly transferable to new classes. To address these challenges, we propose a new direction to adapt CLIP by learning generalized prompts with text-only supervision, without relying on visual data. We introduce a text-only training strategy for prompts to learn a mapping function that embeds rich contextual knowledge from LLM data within the prompts. The context learned by these prompts transfers well to unseen classes and datasets, reducing the LLM prompt engineering and serving cost. We perform extensive evaluations where our text-only approach performs favorably well over previous methods, including those utilizing labeled images.

References

[1] Bang An, Sicheng Zhu, Michael-Andrei Panaitescu-Liess, Chaithanya Kumar Mummadi, and Furong Huang. More context, less distraction: Improving zero-shot inference of clip by inferring and describing spurious features. In *Workshop on Efficient Systems for Foundation Models*, 2023. 3

[2] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Visual prompting: Modifying pixel space to adapt pre-trained models. *arXiv preprint 2203.17274*. 2

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 2020. 2, 4

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*. Ieee, 2009. 3

[5] 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 *ICLR*. PMLR. 1

[6] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *CVPR*, 2023. 2

[7] Muhammad Uzair Khattak, Syed Talal Wasim, Muzammal Naseer, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Self-regulating prompts: Foundational model adaptation without forgetting. In *ICCV*, 2023. 2

[8] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. In *ICLR*, 2023. 1

[9] Sarah Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. In *ICCV*, 2023. 1, 2, 3, 4

[10] 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 *ICML*. PMLR, 2021. 1, 3, 2

[11] Karsten Roth, Jae Myung Kim, A Koepke, Oriol Vinyals, Cordelia Schmid, and Zeynep Akata. Waffling around for performance: Visual classification with random words and broad concepts. 2023. 1

[12] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*. 1, 2, 3

[13] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, 2022. 1, 2, 3

Learning to Prompt with Text Only Supervision for Vision-Language Models

Supplementary Material

The following sections provide supplementary material for our main paper. This includes additional experiments, implementation details, and specifics of our text-to-text data used for training. The contents are organized as follows:

- Additional experiments (Section A)
- Implementation details (Section B)
- Details on Text-Only Data (Section C)

A. Additional Experiments

A.1. Additional comparisons with WaffleCLIP.

We present additional comparisons between ProText and WaffleCLIP [11] approach. WaffleCLIP employs prompt ensembling by introducing random descriptors and characters alongside class names. Specifically, we perform comparison with a WaffleCLIP-Concept variant, which incorporates high-level dataset concepts in its text prompts, such as 'a photo of a flower: a CLS ' for OxfordFlowers. Further details on the WaffleCLIP framework and its variants can be found in [11].

Cross-dataset transfer. For cross-dataset transfer settings, all methods only utilize ImageNet source dataset LLM prompt information. The results are shown in Tab. 5. CuPL shows same performance as CLIP for cross-datasets as class-specific descriptions for new datasets are not available in this setting. Overall, WaffleCLIP uses random descriptors which leads to improvements over CLIP and CuPL. In contrast, ProText with text-only training with ImageNet-1k LLM templates shows consistent improvements over WaffleCLIP by surpassing on 9/10 cross-datasets and leads to the averaged accuracy of 67.23% in the challenging cross-dataset transfer setting.

Text-only supervised setting. We additionally compare WaffleCLIP in text-only supervised setting. As shown in Tab. 6, WaffleCLIP improves over CLIP but lags behind CuPL as it only relies on high-level dataset concepts and random descriptors. CuPL uses class-specific LLM descriptions for prompt ensembling and shows improved results. In contrast to these approaches, ProText adopts a learning-based approach using text data and shows the highest performance by surpassing both WaffleCLIP and CuPL in 10/11 datasets. This suggests that text-only prompt learning can serve as a better alternative to training-free prompt ensembling methods.

A.2. Comparison with image-supervised methods.

We show additional comparisons of ProText with image-supervised methods in terms of generalization performance. In base to novel class generalization setting, we include prompt learning methods utilizing 16-shot image data where we mainly focus on novel class performance for comparison. In text-only supervised setting, we compare ProText with few-shot image supervised methods including CLIP Linear Probe, CoOp, and CoCoOp, which are trained up to 2-shot data.

Unseen class generalization. All methods are trained on seen classes of each dataset and we specifically analyze their performance on unseen classes to study generalization. Results are shown in Tab. 7. Image-supervised prompt learning methods utilize 16-shot base-class labeled data and demonstrate improved accuracy for novel classes. For example, the previous state-of-the-art method, PrompSRC, achieves a substantial accuracy of 70.73% on ImageNet for novel classes. In comparison, ProText, leveraging text-only data, shows an improvement of +0.65% against PromptSRC for novel classes on ImageNet. In summary, ProText consistently outperforms PromptSRC on 9 out of 11 datasets for novel classes, leading to the highest novel class accuracy of 76.98% averaged over 11 datasets.

Supervised setting. In Tab. 8, compare ProText with few-shot image-supervised methods including CLIP Linear Probe, CoOp, and CoCoOp. ProText shows improved averaged performance over 1 & 2 shot Linear Probe. Similarly, ProText without using any images for training improves on most datasets against CoOp and CoCoOp trained with 1 and 2 shots. ProText, without using any images for training, outperforms CoOp and CoCoOp trained with 1 and 2 shots on most datasets. This suggests that text-only training can be considered an effective alternative approach to image-supervised methods under extreme low-data regimes.

A.3. Additional ablation studies.

We present additional ablation experiments conducted on ProText as outlined below.

Combining prompt ensembling and prompt learning. In our ProText approach, learnable prompts for inference are trained on text data. Here, we explore an alternative experiment by averaging the text features with ProText-learned prompts and text features of LLM templates obtained via prompt ensembling. Specifically, we average the LLM prompt features (e.g., CuPL features) and ProText features for the same classes to study if prompt learning and prompt ensembling could be complementary. The results are shown in Table 9. Combining ProText and CuPL features leads to marginal improvement compared to ProText alone. We conjecture that since ProText uses the same LLM template data to learn prompts, the LLM template features and ProText features might not be strongly complementary.

	Source	Target										
	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	Aircraft	SUN397	DTD	EuroSAT	UCF101	*Average*
Zero-shot & Prompt ensembling methods												
CLIP	66.72	92.98	89.13	65.29	71.30	86.11	**24.90**	62.59	44.56	47.84	66.83	65.15
CuPL	69.62	92.98	89.13	65.29	71.30	86.11	**24.90**	62.59	44.56	47.84	66.83	65.15
WaffleCLIP-Concept	68.34	94.01	89.57	63.42	72.00	**86.84**	24.49	66.17	45.15	47.74	67.96	65.74
Prompt learning with text-only supervision												
ProText (Ours)	**69.80**	**94.81**	**91.01**	**66.00**	**72.35**	86.66	24.72	**67.34**	**47.93**	**51.86**	**69.60**	**67.23**

Table 5. **Cross-dataset transfer setting**. Results comparison of ProText with CLIP, CuPL, and Waffle-CLIP. ProText overall shows consistent improvements over LLM-based prompt ensembling methods.

Dataset	CLIP	CuPL	WaffleCLIP-C	ProText	Δ
ImageNet	66.72	69.60	68.34	**70.22**	+0.62
Caltech101	92.98	94.32	94.01	**95.29**	+0.97
DTD	44.56	53.96	45.15	**54.04**	+0.06
EuroSAT	47.84	**60.27**	47.74	58.53	-1.74
StanfordCars	65.29	65.95	63.42	**66.77**	+0.82
Flowers102	71.30	73.85	72.00	**74.42**	+0.57
Aircraft	24.90	27.66	24.49	**29.01**	+1.35
SUN397	62.59	69.00	66.17	**69.76**	+0.76
OxfordPets	89.13	91.11	89.57	**92.72**	+1.61
UCF101	66.83	70.63	67.96	**71.45**	+0.82
Food101	86.11	86.11	**86.84**	86.68	+0.57
Average	65.15	69.31	65.97	**69.90**	+0.59

Table 6. ProText results with text supervision on each dataset. We compare ProText with CLIP and CuPL and WaffleCLIP-Concept. Gains of ProText over CuPL are shown in blue.

Dataset	CuPL	ProText	CoOp	CoCoOp	MaPLe	PromptSRC	Δ
	[9]	Ours	[12]	[13]	[6]	[7]	
ImageNet	68.14	**71.38**	67.88	70.43	70.54	70.73	+3.2
Caltech101	94.00	**95.63**	89.81	93.81	94.36	94.03	+1.6
DTD	59.90	61.59	41.18	56.00	59.18	**62.97**	+1.7
EuroSAT	64.05	**80.97**	54.74	60.04	73.23	73.90	+17
StanfordCars	74.89	**76.08**	60.40	73.59	74.00	74.97	+1.2
Flowers102	77.80	**78.44**	59.67	71.75	72.46	76.50	+0.6
Aircraft	36.29	34.13	22.30	23.71	35.61	**37.87**	-2.2
SUN397	75.35	**79.14**	65.89	76.86	78.70	78.47	+3.8
OxfordPets	97.26	**98.00**	95.29	97.69	97.76	97.30	+0.7
UCF101	77.50	**79.50**	56.05	73.45	78.66	78.80	+2.0
Food101	91.22	91.98	82.26	91.29	**92.05**	91.53	+0.8
Average	74.22	**76.98**	63.22	71.69	75.14	76.10	+2.8

Table 7. **Novel-class generalization comparison.**We compare ProText with prompt ensembling and image-supervised methods on unseen class performance in base-to-novel class generalization setting. Gains of ProText over CuPL are shown in blue.

B. Additional Implementation details

Training details. For training ProText, we use a publically available CLIP ViT-B/16 model from OpenAI [10]. Language prompts for each training are initialized with ‘a photo of a” for the first layer and randomly initialized for the remaining transformer layers of the text encoder of CLIP. All models are trained using the AdamW optimizer on a single 16-GB V100 GPU. For cross-dataset and domain generalization benchmarks, we train ProText using $T = 4$ and $T = 16$ language prompts, respectively, for 10 and 200 epochs, respectively. The warm-up epochs are set to 5 during training.

As text data from LLMs varies in quality and size across datasets, we have observed that training ProText on each dataset requires custom training configurations to achieve the best performance. Therefore, ProText employs optimal prompt length and epoch configuration for each dataset. The optimal training configurations are obtained through the validation splits of each dataset.

Base-to-novel generalization setting. In Tab. 10, we show the hyperparameters used for training models in base-to-novel generalization settings. We use a learning rate of 0.03 for all datasets except UCF101, FOOD101, and Oxford-Flowers where learning rate of 0.0025 is used.

Text-only supervised setting. For our comparison with CuPL [9] in Table 8, ProText models are trained using the same LLM text data as utilized by CuPL. Hyperparameter values are shown in Table 11. All models are trained using a learning rate of 0.03, except for UCF101, EuroSAT, and Oxford-Flowers, where a learning rate of 0.0025 is used.

C. Details on Text-Only Data

As discussed in Sec. 2.2, our ProText approach relies on text-only data ($\mathcal{D}$PROMPT) curated from Language Models (LLMs) for training its language prompts. Here, we provide additional details on the curation of text-only data. Specifically, we first provide information on the text queries used

Dataset	CLIP	CuPL	ProText	Linear Probe		CoOp		CoCoOp	
				K=1	K=2	K=1	K=2	K=1	K=2
ImageNet	66.70	69.62	**70.22**	32.13	44.88	66.33	67.07	69.43	69.78
Caltech101	92.98	94.32	**95.29**	79.88	89.01	92.60	93.07	93.83	94.82
DTD	44.56	53.96	**54.04**	34.59	40.76	50.23	53.60	48.54	52.17
EuroSAT	47.84	**60.27**	58.53	49.23	61.98	54.93	65.17	55.33	46.74
StanfordCars	65.29	65.95	**66.77**	35.66	50.28	67.43	70.50	67.22	68.37
Flowers102	71.30	73.85	**74.42**	69.74	85.07	77.53	87.33	72.08	75.79
Aircraft	24.90	27.66	**29.01**	19.61	26.41	21.37	26.20	12.68	15.06
SUN397	62.59	69.00	**69.76**	41.58	53.70	66.77	66.53	68.33	69.03
OxfordPets	89.13	91.11	**92.72**	44.06	58.37	90.37	89.80	91.27	92.64
UCF101	66.83	70.63	**71.45**	53.66	65.78	71.23	73.43	70.30	73.51
Food101	86.11	86.11	**86.68**	43.96	61.51	84.33	84.40	85.65	86.22
Average	65.15	69.31	**69.90**	45.83	57.98	67.56	70.65	66.79	67.65

Table 8. ProText results with text supervision on each dataset. We compare ProText with CLIP [10], CuPL [9] and image supervised Linear Probe [10], CoOp [12] and CoCoOp [13] methods.

Method	ImageNet Top1.
1: CuPL	69.62
2: ProText	70.22
3: Ensembling: ProText + CuPL	**70.28**

Table 9. Ablation on combining CuPL and ProText text features.

H.parameter	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	Aircraft	SUN397	DTD	EuroSAT	UCF101
Epochs	30	30	50	30	150	50	200	30	200	30	20
# Prompts (T)	4	16	4	8	4	8	4	8	4	16	8

Table 10. Hyper-parameters setting used for base-to-novel generalization setting. Optimal configuration is set using validation splits of each dataset.

H.parameter	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	Aircraft	SUN397	DTD	EuroSAT	UCF101
Epochs	200	30	50	20	300	30	200	200	200	300	100
# Prompts (T)	16	16	4	16	4	16	4	16	16	4	8

Table 11. Hyper-parameters used for text-only supervised setting.

as input to LLMs for generating prompts, followed by qualitative examples of $\mathcal{D}_{\text{PROMPT}}$.

C.1. Queries to LLMs to curate Text-Only Data

Following [9], we obtain class descriptions from LLMs by providing various queries as inputs. Specifically, we utilize queries termed as *Full prompts* by CuPL [9]. For instance, to generate class descriptions of ImageNet-1k classes, we prompt GPT-3 with the following 5 queries:

- ‘Describe what a(n) CLS looks like.’
- ‘How can you identify a(n) CLS? ’
- ‘What does a(n) look like? ’
- ‘Describe an image from the internet of a(n) CLS.’
- ‘A caption of an image of a(n) CLS.’

Here, CLS denotes the class names present in the dataset. After generating LLM class descriptions, we associate all descriptions of the same class with its class-name template given as ‘A photo of a CLS’. This results in our text-only training data $\mathcal{D}_{\text{PROMPT}}$ with text-to-text mapping pairs used to train ProText. Refer to [9] for LLM queries of other datasets used to generate class-specific descriptions. For standardized comparisons, we use publicly available CuPL data and generate descriptions for datasets not provided by CuPL.

C.2. Qualitative examples

As LLMs are pre-trained on internet-scale text corpora, they possess the capability of generating diverse and high-quality descriptions and captions for different class categories, resulting in high-quality text outputs. Below we show some examples of $\mathcal{D}_{\text{PROMPT}}$ text-to-text pairs for the ImageNet-1k dataset.

Class: Tench

Class-name template: ‘A photo of a Tench’

Associated LLM descriptions:

- ‘A tench is a freshwater fish with a dark green back and light-colored sides.’
- ‘A tench looks like a freshwater fish with a dark olive-green back, fading to yellowish-brown on the sides.’
- ‘Tench are a freshwater fish that can grow up to 70cm long! They have olive-brown skin with dark spots, and their meat is white and firm.’
- ‘This image shows a large, dark green tench swimming in a pond.’

Class: bath towel

Class-name template: ‘A photo of a bath towel’

Associated LLM descriptions:

- ‘A bath towel typically has a loops on one side and a smooth surface on the other.’
- ‘A bath towel is a rectangular piece of fabric, usually Cotton, that is used to dry oneself after a bath or shower.’
- ‘The image is of a white bath towel with a blue and green stripes.’
- ‘A fluffy white bath towel draped over a towel rack.’

Class: sandal

Class-name template: ‘A photo of a sandal’

Associated LLM descriptions:

- ‘A sandal is a shoe typically made of leather or synthetic material that has an open toe and a strap or straps that go around the foot or up the ankle.’
- ‘A sandal is usually a flat shoe with a strap that goes around the foot or ankle.’
- ‘This sandal is from the ancient Egyptian city of Thebes.’
- ‘When you are looking to identify a sandal, the first place to start is by looking at the features of the shoe.’