

---

# PROBABILISTIC LANGUAGE-IMAGE PRE-TRAINING

**Sanghyuk Chun**  
NAVER AI Lab

**Wonjae Kim**

**Song Park**

**Sangdoo Yun**

## ABSTRACT

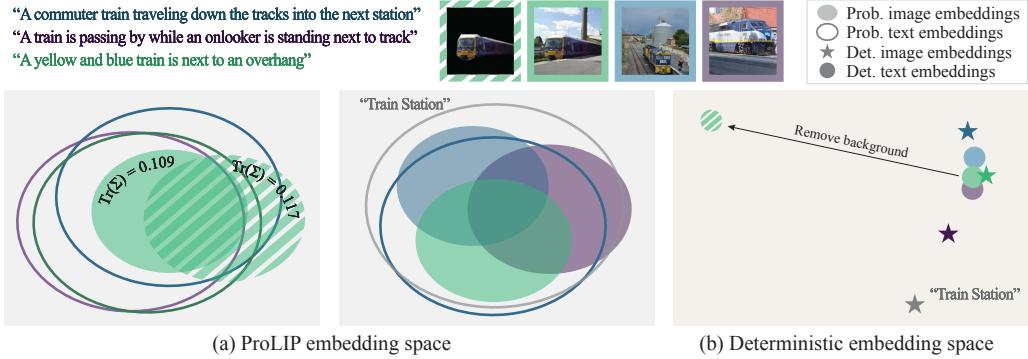
Vision-language models (VLMs) embed aligned image-text pairs into a joint space but often rely on deterministic embeddings, assuming a one-to-one correspondence between images and texts. This oversimplifies real-world relationships, which are inherently many-to-many, with multiple captions describing a single image and vice versa. We introduce Probabilistic Language-Image Pre-training (ProLIP), the first probabilistic VLM pre-trained on a billion-scale image-text dataset using only probabilistic objectives, achieving a strong zero-shot capability (*e.g.*, 74.6% ImageNet zero-shot accuracy with ViT-B/16). ProLIP efficiently estimates uncertainty by an “uncertainty token” without extra parameters. We also introduce a novel inclusion loss that enforces distributional inclusion relationships between image-text pairs and between original and masked inputs. Experiments demonstrate that, by leveraging uncertainty estimates, ProLIP benefits downstream tasks and aligns with intuitive notions of uncertainty, *e.g.*, shorter texts being more uncertain and more general inputs including specific ones. Utilizing text uncertainties, we further improve ImageNet accuracy from 74.6% to 75.8% (under a few-shot setting), supporting the practical advantages of our probabilistic approach. The code is available at <https://github.com/naver-ai/prolip>.

## 1 INTRODUCTION

Vision-language models (VLMs) aim for a joint vision-language embedding space, and have become a cornerstone in the recent advance of machine learning (Radford et al., 2021; Jia et al., 2021; Li et al., 2022; Zhai et al., 2023). For training, VLMs map an aligned image-text pair (*e.g.*, an image and its corresponding caption) into the same space using contrastive learning. Their rich joint representations learned from large-scale image-text aligned datasets have achieved significant success in various downstream tasks, such as zero-shot classification (by treating class labels as a templated text, *e.g.*, a photo of  $\{\cdot\}$ ) or image-text cross-modal retrieval.

Despite their great success, most VLMs encode representations into a deterministic Euclidean space. This assumes a one-to-one correspondence between images and texts, which oversimplifies the complex nature of real-world relationships. In practice, image-text matching is inherently many-to-many. Multiple captions can accurately describe an image, each highlighting different aspects of the visual content. For example, a train image can be described by multiple captions, *e.g.*, “a train”, “train station” or “train parked next to a station”. Conversely, a caption may correspond to several images describing similar scenes or objects, *e.g.*, “a train” can be matched to all the train images. However, as shown in Figure 1 (b), a deterministic model (*e.g.*, CLIP (Radford et al., 2021)) fails to capture the multiplicity, *e.g.*, “Train Station” embedding is located to an irrelevant point to the other train images and captions. This is because the CLIP loss forces the positive pairs close and random negative pairs far away, which has no stable solution when we map them onto a point in Euclidean space.

Instead of representing an input to a deterministic point vector, we aim to map an input to a random variable. As shown in Figure 1 (a), our probabilistic VLM (PrVLM) approach can handle the multiplicity, *e.g.*, the distribution of “Train Station” covers all the train image distributions. This paper introduces Probabilistic Language-Image Pre-training (ProLIP), the first PrVLM pre-trained on billion-scale image-text pairs only using probabilistic objectives. Compared to previous PrVLM works (Chun et al., 2021; Ji et al., 2023; Upadhyay et al., 2023; Chun, 2024), ProLIP has several advantages. First, while the previous methods need a dedicated module to predict the uncertainty, ProLIP estimates uncertainty very efficiently simply by adding an “uncertainty token” ([UNC]) to input without other additional parameters. Second, we introduce a novel inclusion loss, which en-



**Figure 1: Comparison of ProLIP and deterministic embedding spaces.** We visualize images and captions from MS-COCO Caption (Chen et al., 2015) using models trained on DataComp 1B (Gadre et al., 2024) with 1.28B seen samples (See Appendix A.1 for more details of visualization method). ProLIP can capture multiplicity of image-text matching (*e.g.*, the text embedding of “Train station” covers all three train images), while deterministic embeddings fail to capture the ambiguity. Furthermore, when we synthetically remove the background, ProLIP maps the new embedding near the original embedding but with a larger uncertainty value ( $0.109 \rightarrow 0.117$ , while the deterministic model maps the new embedding very far from the original embedding).

forces the distributional inclusion relationship between an image-text pair and between the original input data and the masked one. Our new objective helps embeddings be more interpretable by humans. Third, ProLIP can be trained from scratch without needing any pre-trained models and achieve state-of-the-art zero-shot capability without fine-tuning. Furthermore, ProLIP achieves strong zero-shot capability, *e.g.*, 74.6% ImageNet zero-shot accuracy with the ViT-B/16 backbone, where the CLIP model with the same number of seen samples achieves 73.5% (Ilharco et al., 2021).

In the experiments, ProLIP slightly outperforms the deterministic CLIP model in zero-shot classification (ZSC) tasks (*e.g.*, CLIP shows 67.2 ImageNet ZSC accuracy, while ProLIP shows 67.6). We also show the benefits of using uncertainty estimates for image-text tasks. First, we observe that our intuition and the learned uncertainty are aligned well. For example, (1) texts generally “include” images (*i.e.*, texts are more uncertain than images), (2) shorter texts tend to be more uncertain, (3) more general texts/images tend to be more uncertain and include more specific ones (*e.g.*, masked image and “Train Station” in Figure 1). Furthermore, we show two applications when a proper uncertainty estimate is helpful; Bayesian Prompt Re-Weighting (BPRW), a fully Bayesian approach to seek better ImageNet zero-shot prompts which improves accuracy from  $74.6\% \rightarrow 75.8\%$ , and the uncertainty-based dataset traversal, which provides a better understanding of dataset hierarchy.

## 2 PRELIMINARY

### 2.1 INHERENT AMBIGUITY INDUCED BY THE MULTIPLICITY OF IMAGE-TEXT PAIRS

The nature of image-text matching is many-to-many. Unfortunately, in practice, this multiplicity is not fully annotated in VL datasets; we only treat one corresponding caption as “positive” caption, while the others are considered as “negative”. For example, in COCO Caption (Chen et al., 2015), more than 80% of positive correspondences are labeled as “negative” (Chun et al., 2022). As observed by Chun (2024), this hidden multiplicity inherently causes ambiguity for VL datasets. For example, assume we have the caption “a train is next to a train station” and three semantically similar images showing a train next to a train station. Here, there will be only one positive image for the caption due to the construction protocol of VL datasets. If we approximate three image embeddings as the same image embedding, the correspondence between this approximated embedding, and the caption will be uncertain (*i.e.*, either positive or negative). Suppose we use deterministic matching loss, such as contrastive loss used by CLIP (Radford et al., 2021). In this case, the best deterministic mapping will map the caption embedding not very close to the image embeddings but “properly” far away from them. CLIP does not have enough capability to capture multiplicity and ambiguity. We aim to achieve an embedding space that can represent the inherent uncertainty of the input (also known as “aleatoric uncertainty”) for a more interpretable and understandable embedding space.

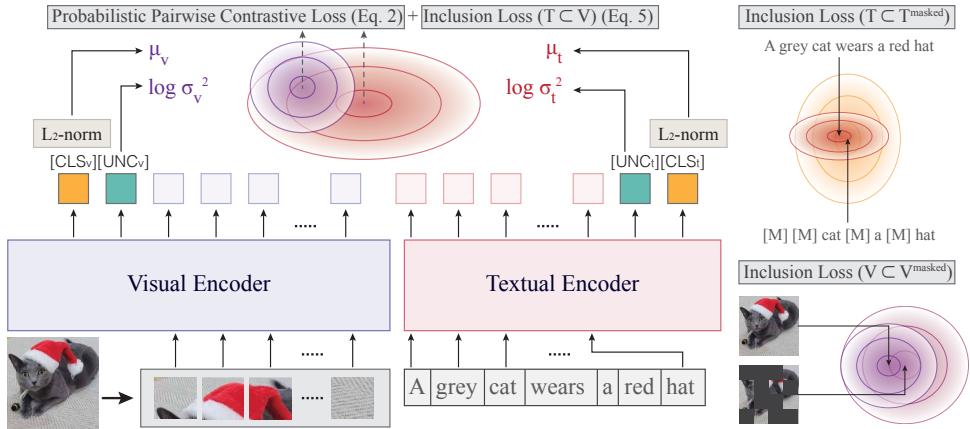


Figure 2: **Overview of ProLIP.** [CLS] and [UNC] tokens are used for  $\mu$  and  $\log \sigma^2$ , respectively.

## 2.2 PROBABILISTIC IMAGE-TEXT REPRESENTATIONS

Probabilistic embeddings map each data point as a random variable (*e.g.*, a Gaussian distribution) rather than a fixed vector, capturing the inherent uncertainty and diversity. This approach offers a better understanding of the semantic space by providing an extra axis of uncertainty, *e.g.*, we can quantify the uncertainty of the input by using the estimated uncertainty (*e.g.*, covariance of Gaussian). Recently, Kirchhof et al. (2023) theoretically have shown that a probabilistic representation learning with a proper probabilistic matching loss can recover the correct aleatoric uncertainty. Namely, a probabilistic mapping can capture the ambiguity of the inputs. Probabilistic embeddings have been actively studied for applications with inherent ambiguity, such as word embeddings (Nguyen et al., 2017), image embeddings (Oh et al., 2019), face understanding (Shi & Jain, 2019; Chang et al., 2020), 2D-to-3D pose estimation (Sun et al., 2020), speaker diarization (Silnova et al., 2020), video understanding (Park et al., 2022), and composed image retrieval (Neculai et al., 2022).

As we investigated in Section 2.1, VL tasks also suffer from aleatoric uncertainty caused by the inherent multiplicity of image-text matching and sparse annotations. Recently, there have been attempts to tackle the inherent ambiguity of VL tasks with probabilistic embeddings (Chun et al., 2021; Ji et al., 2023; Upadhyay et al., 2023; Chun, 2024). However, these methods have a very limited scale to be used as a generic purpose VLM, such as CLIP. For example, ProbVLM (Upadhyay et al., 2023) is an ad-hoc module top on the frozen pre-trained CLIP, limiting the full exploration of the probabilistic space. Furthermore, ProbVLM is only trained on small image caption datasets, such as CUB (Wah et al., 2011) or COCO caption (Chen et al., 2015), which makes it not applicable to more practical zero-shot classification applications. MAP (Ji et al., 2023) proposes a pre-training method using a cross-attention Transformer. However, it has a limited zero-shot capability, resulting in the need to fine-tune the model for each downstream task. Furthermore, its structure is highly inefficient for retrieval systems; it needs both image and text pair to compute a similarity between them, *i.e.*, we have to compute all possible image-text pairs to get the full similarity. Lastly, PCME++ (Chun, 2024) showed a possibility of pre-trained PrVLM, but their scalability is still limited (*e.g.*, achieving 34% ImageNet zero-shot accuracy). We empirically observe that the objective function of PCME++ shows slow or unstable training under large-scale image-text pairs. Furthermore, all these PrVLMs need heavy additional parameters to estimate uncertainty from data. ProLIP does not need a dedicated module for uncertainty estimate but employs a very efficient strategy using [UNC].

## 3 PROBABILISTIC LANGUAGE-IMAGE PRE-TRAINING

### 3.1 ARCHITECTURE

We model an input as a Gaussian random variable with a diagonal covariance by estimating mean  $\mu$  and variance  $\sigma^2$  vectors from the input. Similar to CLIP (Radford et al., 2021), ProLIP has separate visual and textual encoders. We use VisionTransformer (ViT) (Dosovitskiy et al., 2021) for the visual encoder and Transformer (Vaswani et al., 2017) for the textual encoder.

---

Previous Probabilistic VLMs (PrVLMs) introduce additional parameters for estimating uncertainty. For example, PCME++ (Chun, 2024) uses one multi-head self-attention block for this. However, this approach will require additional parameters and computational costs, limiting usability. Instead of an additional uncertainty module, we simply introduce a new uncertainty token [UNC], along with the class token [CLS] (See Figure 2). Compared to the previous PrVLMs, [UNC] requires almost negligible additional parameters. The visual encoder takes [CLS] and [UNC] at the beginning of the input sequences, while the textual encoder takes [UNC] and [CLS] at the end of the input. This is because the textual encoder of the original CLIP uses the end-of-sentence token rather than [CLS] at the beginning. Note that we assume diagonal covariance for simplicity, namely, [UNC] is the same dimension with [CLS]. We use the L2-normalized [CLS] output as  $\mu$  and [UNC] output as  $\log \sigma^2$ . Similar to [CLS], [UNC] is projected to the final embedding space using a linear layer. We initialize the bias value of this layer to a small value (e.g.,  $-10$ ) to initialize the initial  $\sigma^2$  scale small (e.g.,  $\exp(-10) \approx 5 \times 10^{-5}$  for each dimension). This simple trick helps stable training.

### 3.2 PROBABILISTIC PAIRWISE CONTRASTIVE LOSS

In this subsection, we introduce the probabilistic pairwise contrastive loss (PPCL), the main objective function of ProLIP. PPCL is similar to the probabilistic matching loss (PML) of PCME++, but we modify PML for stable training based on the log sigmoid loss by SigLIP (Zhai et al., 2023).

Following PCME++, we use the closed-form sampled distance (CSD) as our probabilistic distance:

$$d_{\text{CSD}}(Z_1, Z_2) = \mathbb{E}_{Z_1, Z_2} \|Z_1 - Z_2\|_2^2 = \|\mu_1 - \mu_2\|_2^2 + \text{tr}(\Sigma_1 + \Sigma_2) = \|\mu_1 - \mu_2\|_2^2 + \|\sigma_1 + \sigma_2\|_1, \quad (1)$$

where  $Z_1$  and  $Z_2$  are Gaussian random variables with diagonal covariances. The probabilistic matching loss by PCME++ uses pairwise binary cross entropy (BCE) by taking  $-a \cdot d_{\text{CSD}}(Z_1, Z_2) + b$  as logits, where  $a$  and  $b$  are learnable scalars. However, we empirically observe that PML fastly converges to a small value, and its gradient is dramatically small, which makes the overall learning procedure slow or unstable (see Appendix A.2 for details). To solve the problem, we employ log sigmoid loss (Zhai et al., 2023). By replacing the squared L2 distance  $\|\mu_1 - \mu_2\|_2^2$  to Equation (1) (details are in Appendix A.3), we have a new probabilistic pairwise contrastive loss (PPCL):

$$\mathcal{L}_{\text{PPCL}}(Z_v, Z_t) = -\log \frac{1}{1 + \exp(y_{vt}(-a(\mu_v^\top \mu_t - \frac{1}{2}\text{tr}(\Sigma_v + \Sigma_t)) + b))}, \quad (2)$$

where  $a$  and  $b$  are learnable scalar values and  $y_{vt}$  is 1 if  $v$  and  $t$  are matched otherwise -1.

### 3.3 INCLUSION LOSS

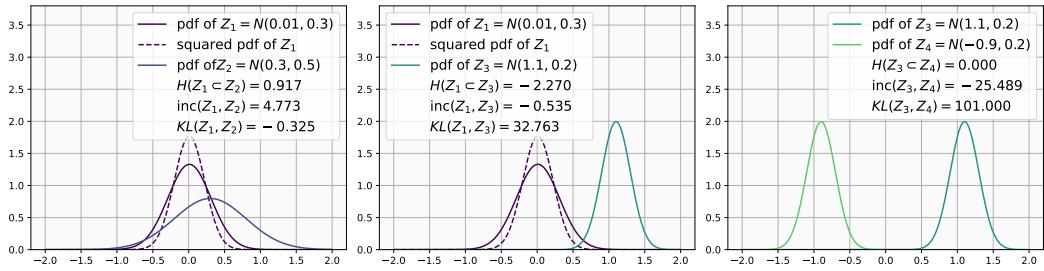
Although PPCL enables to learn probabilistic representations, we empirically observe that the learned uncertainty from data is often counterintuitive to humans. For example, we may expect that text captions with a general meaning (e.g., “photo”) has a very large covariance that can cover all the photographic image embeddings, but sometimes it does not. Similarly, we may expect that if a text or an image loses some information (e.g., some tokens are randomly masked), its probability distribution will entail the distribution of the original sample. However, it is not always guaranteed that a model will learn desired uncertainty, especially under noisy image-text correspondences.

To tackle the issue, we introduce a novel objective function enforcing a random variable  $Z_1$  to be included in another random variable  $Z_2$ . Let  $p_1$  and  $p_2$  be their corresponding probability density function (pdf). If  $Z_1$  is included in  $Z_2$ , then we can presume that the area with high  $p_1$  will be overlapped to the area with high  $p_2$ . From this observation, we propose a novel inclusion measure by emphasizing the area with high  $p_1$  and compute expectation of the emphasized  $p_1$  under the distribution  $p_2$ . Specifically, we take the square to  $p_1$ , and compute  $\int p_1(x)^2 p_2(x) dx$ . This measure is related to Bhattacharyya distance ( $\int \sqrt{p_1 p_2} dx$ ) or the inner product ( $\int p_1 p_2 dx$ ), but our measure is designed for measuring “inclusion” (it becomes high if  $Z_1$  is included in  $Z_2$  and otherwise low) while the others are designed for measuring “distance” or “dissimilarity” between distributions.

The log inclusion measure (omitting constants) can be derived as follows:

$$\text{inc}(Z_1, Z_2) = \log \int_{-\infty}^{\infty} p_1^2(x) p_2(x) dx = -2 \log \sigma_1^2 - \log \sigma_2^2 - \frac{1}{2} \log(A) + \frac{B^2}{4A} - C, \quad (3)$$

where,  $A = \frac{1}{\sigma_1^2} + \frac{1}{2\sigma_2^2}$ ,  $B = \frac{2\mu_1}{\sigma_1^2} + \frac{\mu_2}{\sigma_2^2}$ ,  $C = \frac{\mu_1^2}{\sigma_1^2} + \frac{\mu_2^2}{2\sigma_2^2}$ .



**Figure 3: Visual understanding of the proposed inclusion loss.** We plot probabilistic density functions (pdfs) of three pairs of Gaussian distributions and their inclusion hypothesis,  $\mathcal{H}(Z_1 \subset Z_2)$  (Equation (4)), log inclusion (Equation (3)) and KL divergence. The dashed line denotes the squared pdf, *i.e.*,  $p^2(x)$ .  $\mathcal{H}(Z_1 \subset Z_2)$  becomes (a) positive if  $Z_1$  is included in  $Z_2$  and (b) otherwise negative. (c) If  $Z_1$  and  $Z_2$  are the same level, then  $\mathcal{H}$  will become zero. While log inclusion represents how  $Z_1$  is included in  $Z_2$ , KL measures the “dissimilarity” between distributions (*e.g.*, (c) has the largest KL, but the smallest log inclusion)

The full derivation is in Appendix A.4.

Now, using Equation (3), we introduce a hypothesis test whether  $Z_1$  is included in  $Z_2$  as follows:

$$\mathcal{H}(Z_1 \subset Z_2) = \log \int_{-\infty}^{\infty} p_1^2(x)p_2(x)dx - \log \int_{-\infty}^{\infty} p_1(x)p_2^2(x)dx. \quad (4)$$

$\mathcal{H}$  is positive if the hypothesis is true and otherwise negative (See Figure 3). It has two distinct properties compared to other probabilistic measures. First,  $\mathcal{H}$  is asymmetric. The most probabilistic measures aim to measure the “distance”, “overlapping” or “dissimilarity” between two distributions, therefore symmetric (*e.g.*, Wasserstein distance and Bhattacharyya distance). Meanwhile, we measure the level of “inclusion” of two random variables, therefore asymmetric: if  $Z_1$  is included in  $Z_2$ , then  $Z_2$  will not be included in  $Z_1$ , *i.e.*,  $\mathcal{H}(Z_1, Z_2) \neq \mathcal{H}(Z_2, Z_1)$ . Compared to the asymmetric measure, such as KL divergence, we aim to measure how  $Z_1$  is “included” in  $Z_2$ . In contrast, KL measures the dissimilarity between them based on relative entropy. As shown in Figure 3, even if  $Z_1$  and  $Z_2$  have the same variance, we have very high KL, while our measure becomes very small.

Similarly to PPCL, we use the log sigmoid loss for stable convergence. We use  $\log \int_{-\infty}^{\infty} p_1^2(x)p_2(x)dx - \log \int_{-\infty}^{\infty} p_1(x)p_2^2(x)dx$  as the logit value, where the logit becomes positive if  $Z_1$  is included in  $Z_2$ . Now, we introduce our novel inclusion loss as follows:

$$\mathcal{L}_{\text{inclusion}}(Z_1 \subset Z_2) = -\log \frac{1}{1 + \exp(-c\mathcal{H}(Z_1 \subset Z_2))}, \quad (5)$$

where  $c$  is a positive scalar value. For stability, we fix  $c$  as a large scalar value, such as 1000. Like KL divergence, our inclusion loss can be volatile if variance values become extremely small. To prevent a loss explosion, during training, we multiply a small  $\varepsilon$  to  $\frac{1}{\sigma^2}$  for computing  $A, B, C$  in Equation (3), mimicking each Gaussian has sufficiently large variances multiplied by  $1/\varepsilon$ .

Using the inclusion loss, we enforce two properties to the model. First, we let text distribution include image distribution, *i.e.*,  $\mathcal{L}_{\text{inclusion}}(Z_v \subset Z_t)$ . This intuition is from observations from the previous studies showing that “text entails image” (Chun et al., 2021; Desai et al., 2023; Chun, 2024; Kim et al., 2024). Conceptually, when we describe an image, we select the product of the relevant concepts by an arbitrary choice. Therefore, text usually has more general information than images. As another property, we let an embedding of partial information include the embedding of its full information, *i.e.*,  $\mathcal{L}_{\text{inclusion}}(Z \subset Z^{\text{partial}})$ . For example, we generate a text containing partial information of the original caption by masking out random tokens (Devlin, 2018). Similarly, we generate a partial image by masking out the image tokens (He et al., 2022). In practice, we mask out 75% of the input tokens to generate partial information. For text, we replace the input tokens with [MASK], and for image, we drop the input patch tokens for efficient computation (Li et al., 2023).

Finally, we use the VIB loss as a regularization of each Gaussian embedding (*i.e.*, preventing too small  $\sigma^2$ ) following Chun et al. (2021) and Chun (2024). Putting Equation (2) and Equation (5) all

---

together, we have the following learning objective:

$$\begin{aligned} \mathcal{L} = & \sum_{Z_v \in \mathcal{V}} \sum_{Z_t \in \mathcal{T}} \mathcal{L}_{\text{PPCL}}(Z_v, Z_t) + \sum_{Z_v \in \mathcal{V}} [\alpha_2 \mathcal{L}_{\text{inclusion}}(Z_v \subset Z_{v^{\text{masked}}}) + \beta \mathcal{L}_{\text{VIB}}(Z_v)] \\ & + \sum_{Z_t \in \mathcal{T}} [\alpha_2 \mathcal{L}_{\text{inclusion}}(Z_t \subset Z_{t^{\text{masked}}}) + \beta \mathcal{L}_{\text{VIB}}(Z_t)] + \sum_{(v,t) \in (\mathcal{V}, \mathcal{T})} \alpha_1 \mathcal{L}_{\text{inclusion}}(Z_v \subset Z_t), \end{aligned} \quad (6)$$

where  $\alpha_1, \alpha_2, \beta$  are control hyperparameters for each loss function. For computational efficiency, we generate masked samples only for 12.5% samples in the mini-batch and compute inclusion loss using them. VIB loss is computed for all samples.

### 3.4 PROMPT TUNING WITH UNCERTAINTY ESTIMATES

As observed by Chun (2024), the estimated uncertainty is not only beneficial to understanding the input data uncertainty but also effective to zero-shot classification (ZSC). In practice, we use multiple templated text prompts to estimate the textual embedding of class names. For example, the original CLIP paper uses 80 prompts, including “a photo of {·}”. Although this prompt engineering with a mixture of templates significantly improves ZSC performances, it is still unclear which template benefits ZSC. Furthermore, if we carefully explore images of each class, we can conjecture that each class might need different templates. For example, in ImageNet, “ferret” images often co-occur with the ferret’s owner. In this case, prompts like “a photo of **my** ferret” can be helpful to estimate a text feature corresponding to the images, compared to using “a origami ferret”.

How can we select the most informative text prompts? One possible solution will be filtering out highly uncertain text prompts. For example, for “black-footed ferret” class, “the embroidered {·}”, “the origami {·}” or “the plastic {·}” have high uncertainty values, while “a low resolution photo of {·}” or “a cropped photo of {·}” have small uncertainty values. Our experiment shows this strategy is moderately effective: it improves ImageNet ZSC accuracy +0.1pp. We presume that it is because the variety of text prompt uncertainties for each class is not significantly large. Also, we presume that the suitability of text prompts is not solely dependent on the text itself; we may need to consider how texts describe the corresponding images well. We propose Bayesian Prompt Re-Weighting (BPRW), a simple probabilistic approach to find the optimal weight of prompts for each class.

Let  $\pi_c \in \mathbb{R}^N$  be the weight of each prompt, where  $N$  is the number of prompts (*i.e.*, 80 for ImageNet) and  $c$  is the class index. Our goal is to find the best  $\pi_i$  that the new text embedding  $Z_t^{\text{new}} = \sum_i \pi_c^i Z_t^i$  describe the given  $M$  image embeddings  $Z_v^j$ . To achieve this goal, we optimize  $\pi_c$  to have the best posterior for  $Z_t$  and  $Z_v$ . First, we sample  $K$  point vectors for each  $Z_v^j$  and assume they are observations (total  $K \times M$  point vectors). Next, we optimize a simple Expectation-Maximization (EM) algorithm to find the best  $\pi$  achieving the best log-likelihood. Here, we set a Dirichlet prior for  $\pi_c$  using the uncertainty values of each  $Z_t$ , *i.e.*, a prompt with higher uncertainty has a smaller prior. Due to the page limit, we describe the detailed algorithm in Appendix A.5.

Although our algorithm is theoretically well-founded and flexible by a Bayesian approach (*e.g.*, setting prior assumption using  $\sigma_t^2$ ), it needs the image embeddings corresponding to the target class, which violates the ZSC assumption. We tackle this issue by collecting corresponding image embeddings using KNN for each text class embedding. If we can use some true pairs under a few-shot setting (*e.g.*, 5 true images for each class), we observe a significant performance improvement.

## 4 EXPERIMENTS

### 4.1 IMPLEMENTATION DETAILS AND EXPERIMENTAL PROTOCOL

**Model.** We use ViT-B/16 (Dosovitskiy et al., 2021) as our image encoder and a 12-layer 768-wide Transformer (Vaswani et al., 2017) as our text encoder. We set the embedding dimension to 768 and the context length to 64 tokens, following SigLIP ViT-B/16 (Zhai et al., 2023).

**Optimization.** We implement ProLIP based on openclip (Ilharco et al., 2021) and the DataComp-1B dataset (Gadre et al., 2024). We list the optimization hyperparameters in Appendix B.1. We train ProLIP models using 32 NVIDIA H100 GPUs with Bfloat16 precision, taking

about one day to train a ViT-B/16 model with 1.28B seen samples. We initialize the bias value of the linear projection top on [UNC] to  $-10$  to initialize  $\log \sigma^2$  with a small value. We set the initial scale and bias parameters ( $a$  and  $b$  in Equation (2)) to 10 and  $-10$ , following Zhai et al. (2023). We randomly select 12.5% image-text pairs from the mini-batch and masking out their 75% information using [MASK] for texts (Devlin, 2018) and token drop for images (He et al., 2022).

**Evaluation.** We evaluate the models on 38 tasks of Datacomp evaluation suite (Gadre et al., 2024) – the full evaluation datasets are listed in Appendix B.2. We report five categories: ImageNet, 6 ImageNet variants with distribution shifts, 13 VTAB tasks, 3 retrieval tasks, and the average of 38 tasks. In addition, we employ the HierarCaps dataset (Alper & Averbuch-Elor, 2024) which provides captions with four different hierarchies (e.g., “water sports”  $\Rightarrow$  “kite surfing”  $\Rightarrow$  “kite surfer on top of the board”  $\Rightarrow$  “kite surfer in the air on top of a red board”). Similarly, we construct new HierarImg dataset, which provides images with four different hierarchies (See Figure B.1 for examples). We will describe the details of HierarCaps, HierarImg, and their evaluation in Section 4.4.

## 4.2 MAIN RESULTS

Table 1 shows the main result. We use multiple prompts for each task following the DataComp evaluation suite. Similar to CLIP zero-shot classification (ZSC), ProLIP uses the ensemble of multiple prompts by  $Z_t^{\text{mixed}} = \mathcal{N}(\frac{1}{N} \sum_i \mu_i, \frac{1}{N} \sum_i \sigma_i^2)$ , where  $\mu_i$  and  $\sigma_i^2$  denote the mean and variance of  $i$ -th prompt and  $N$  is the number of prompts (e.g., 80 for ImageNet). Note that if we treat this operation as the “average” of  $N$  random variables, then the variance should be  $\frac{1}{N^2} \sum_i \sigma_i^2$ , but we empirically observe that the division decreases the final ZSC performance, e.g., 74.51 where our approach shows 74.58 on ImageNet. We did not use the uncertainty-based ZSC described in Section 3.4 for evaluating 38 tasks. Instead, we use CSD to find the nearest class text embedding.

Table 1 shows that ProLIP outperforms CLIP in all metrics with 1.28B seen samples. Furthermore, when we train ProLIP with 12.8 billion seen samples, we achieve a high-performing PrVLM backbone. We show more experiments in Appendix C.2, including ablation study and other backbones.

Table 1: **Zero-shot classification results.** The full results for each task can be found in Table C.1.

	# Samples Seen	ImageNet	IN dist. shifts	VTAB	Retrieval	Average
ViT-B/16	CLIP	1.28B	67.2	55.1	56.9	53.4
	SigLIP	1.28B	67.4	55.4	55.7	53.4
	ProLIP	1.28B	67.6	55.0	57.1	53.4
ProLIP	12.8B	74.6	63.0	63.7	59.6	63.3

## 4.3 UNDERSTANDING THE LEARNED UNCERTAINTY

**Uncertain samples visualization.** From Equation (1), we can define the uncertainty of the given input by measuring  $tr(\Sigma)$ , namely,  $\sum_i \sigma_i^2$  (we simply denote  $\sigma_v^2$  for image uncertainty and  $\sigma_t^2$  similarly). Figure 4 shows the samples with low and high uncertainty values using this value. We extract samples from the 3.5M subset of 12.8B DataComp CommonCrawl small (Gadre et al., 2024) filtered by CLIP similarity and English filtering<sup>1</sup>. We use ProLIP with 12.8B seen samples for the analyses.

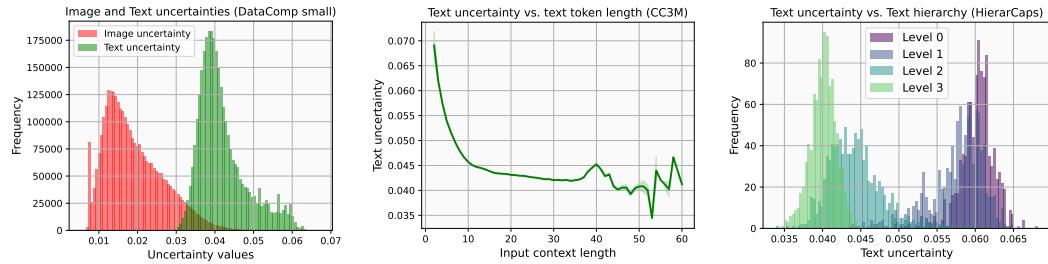
Figure 4 shows that the texts with more general meanings have high uncertainty, e.g. “Screenshot” or “graphic”. This is because a shorter text with more general meaning has more opportunity to be matched to various images. In contrast, certain captions describe a longer and distinct context, such as the exact address or proper nouns, which is unlikely matched to multiple images. We will show that the context length of the text is highly correlated to the uncertainty value (Figure 6). Interestingly, there are captions with high uncertainty despite long context lengths (e.g., more than the specified context length). We empirically observe that such captions have almost no information, showing that ProLIP captures the text uncertainty well. The examples are shown in Appendix C.1.

We can see certain and uncertain images in the upper row of Figure 4. On the uncertain image side, we can find images solely with an object (e.g., a clock, a book, or a clipart) on a white background.

<sup>1</sup><https://huggingface.co/datasets/nielsr/datacomp-small-filtered>



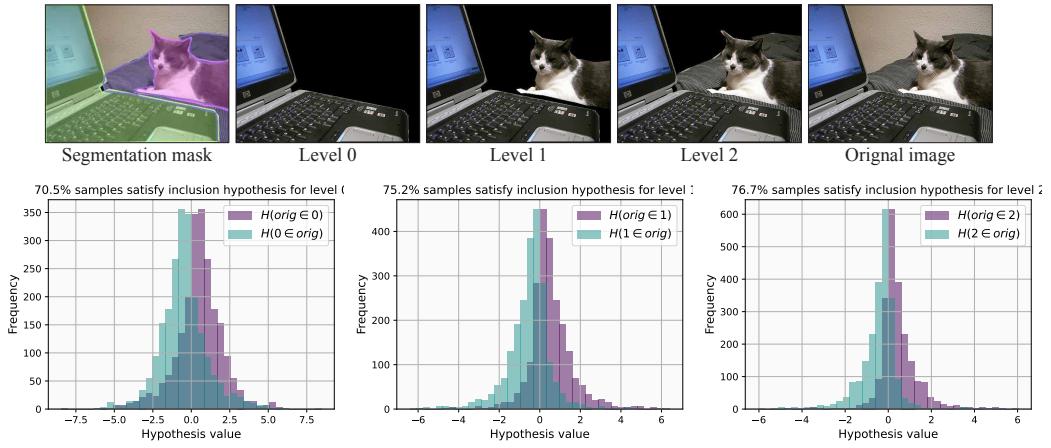
Figure 4: **Uncertain & certain samples.** Visualization from the 3.5M filtered DataComp Small pool.



Generally, such images can be matched to multiple possible descriptions, *e.g.*, the name of the product, the detailed explanation of the product, or the written text in the image (*e.g.*, the book title). On the other hand, certain images have more complex visual cues that can be described with more and specific captions. More samples with high and low uncertainty can be found in Figure C.1.

**Statistics of  $\sigma_v^2$  and  $\sigma_t^2$ .** In Figure 4, we also observe that  $\sigma_v^2$  is generally smaller than  $\sigma_t^2$ , would be originated from the inclusion loss  $\mathcal{L}_{\text{inclusion}}(Z_v, Z_t)$  in Equation (6). Figure 5 shows that the image embeddings and text embeddings have distinct uncertainty values, as we expected.

**What is the source of the uncertainty?** We answer this question by analyzing text context length and data hierarchy. First, we plot the text uncertainty by the context length on the ConceptualCaption 3M (CC3M) captions (Sharma et al., 2018). As shown in Figure 6, a short caption tends to have a large uncertainty value. For example, we observe that the caption “*film series*” has the largest uncertainty value in CC3M, while the caption “*gangsta rap artist told by person @ person I almost died you have to see this!*” is the most certain caption. Namely, more uncertain captions tend to be more logically “general” captions, such as “*dress - sewing pattern*” or “*person – before & after*”, while more certain captions specify a particular situation. From this observation, we explore the relationship between the uncertainty and varying levels of description. We employ the HierarCaps dataset, whose images have four levels of descriptions from the full caption of COCO Caption and its logical entailment hierarchy with three different levels (*e.g.*, “bird”  $\Rightarrow$  “blue bird” …). Examples are shown in Figure 9. Figure 7 shows the relationship between the text uncertainty and text hierarchy levels. Here, Level 0 denotes the most general captions, *e.g.*, “chair” or “bird”, and Level 4 represents the original COCO Caption. As shown in the figure, more general captions (lower levels) tend to be more uncertain, while more specific captions (higher levels) tend to be less uncertain.



**Figure 8:  $\sigma_v^2$  by image hierarchy using HierarImgs.** We tested the inclusion hypothesis of the original image and masked images from level  $i$ ,  $\mathcal{H}(\text{orig} \subset i)$ , and its inverse hypothesis,  $\mathcal{H}(i \subset \text{orig})$ . In all tests, more than 70% images are included in their lower-level images (purple histogram bars with positive hypothesis values). The dataset construction of HierarImgs and related discussions can be found in Appendix B.3 and C.4.

We further investigate whether a similar phenomenon happens for the visual modality by constructing a new HierarImgs dataset to represent the logical visual hierarchy. As shown in the upper row of Figure 8, our HierarImgs dataset consists of four levels: Level 4 is the original image and Level 0 is the largest visual segment. The details of the dataset construction can be found in Appendix B.3.

Using the images, we analyze the relationship between the visual uncertainty and varying levels of visual information. We test whether each image becomes more uncertain than the original image by applying the inclusion hypothesis. Namely, we test if a lower-level image includes its original image by Equation (4). As shown in Figure 8, most of the images satisfy the inclusion hypothesis (*e.g.*, more than 70%), implying that ProLIP also captures image hierarchy. In Appendix C.4, we explain more details of the image hierarchy experiments, including the absolute  $\sigma_v^2$  value by different levels and possible pitfalls of HierarImgs (*e.g.*, we need more careful filtering for a reliable evaluation).

#### 4.4 APPLICATION USING UNCERTAINTY

**Image traversals.** We follow the protocol by Desai et al. (2023). We first set the [ROOT] embedding. Then, we retrieve the nearest caption of the given image, and interpolate [ROOT] and the text embedding with 50 equally spaced steps. Desai et al. (2023) used the null text embedding "" as the [ROOT] of the CLIP embedding space. In ProLIP case, we can utilize the additional uncertainty information and the inclusion hypothesis. Hence, we search the root embedding of the given embedding by searching the text embedding that includes the given image embedding most. The other procedure equals to Desai et al. (2023). We perform traversals on HierarCaps.

We show image traversal results in Figure 9. Interestingly, the most inclusive caption (*i.e.*, [ROOT]) for each image is not always same to the ground truth level 0 caption of HierarCaps. For example, our estimated [ROOT] for the vase picture is "vase", while the GT level 0 caption is "object". From the observation, we can presume that although our retrieval results are plausible, it could lead to inferior HierarCaps retrieval results. To ensure more diversity, we take an average of "" and the most inclusive text embedding and use it as the root embedding. Using the newly proposed root embedding, we quantitatively measure the performance of our traversal in Table 2. First, our new [ROOT] embedding is more specialized to the inputs, rather than only using ""; we can achieve higher R@1<sup>[ROOT]</sup> using our approach. The table shows that the proposed probabilistic image traversal achieves higher recall and precision than deterministic traversal using "" as [ROOT]. Namely, the probabilistic approach gives more opportunities to get more precise captions during the traversal.

**Uncertainty-based ImageNet prompt enhancement.** As described in Section 3.4, we propose BPRW, a new prompt re-weighting method to find a weight  $\pi_c$  for each class. A text embedding weighted by  $\pi_c$ , *i.e.*,  $Z_t^{\text{new}} = \sum_i \pi_c^i Z_t^i$  will be used as a new class embedding for ZSC. Table 3

Query images					
Top-1 Retrieved captions	Kite surfer in the air on top of a red board	A blue bird sitting on the top of a branch with autumn leaves	An outdoor table containing assorted bowls of food and beer	A large red chair has a horse statue on it	Leaves and purple flowers come out of a brown vase on a desk
	Kite surfer in the air on top of a red board	A blue bird sitting on the top of a branch with autumn leaves	An outdoor table containing assorted bowls of food and beer	A large red chair has a horse statue on it	Leaves and purple flowers come out of a brown vase on a desk
	Kite surfer in the air	Blue bird sitting on top of branch	Outdoor picnic	Red chair has a horse statue	Vase with flowers in it
	<b>kite surfing</b>		<b>picnic</b>	<b>red chair</b>	<b>vase</b>
HierarCaps GTs	Level 3	Kite surfer in the air on top of a red board	A blue bird sitting on the top of a branch with autumn leaves	An outdoor table containing assorted bowls of food and beer	A large red chair has a horse statue on it
	Level 2	Kite surfer on top of the board	Blue bird sitting on top of branch	Outdoor table with food and beer	Red chair has a horse statue
	Level 1	kite surfing	blue bird	outdoor table	red chair
	Level 0	<b>water sports</b>	<b>bird</b>	<b>outdoor</b>	<b>vase</b>
					<b>object</b>

Figure 9: **Image traversals with ProLIP.** For each image, we estimate the [ROOT] caption which include the image most using Equation (4). Then, we interpolate the [ROOT] caption and the retrieved caption. We compare our interpolation and HierarCaps GTs. Red denotes when the estimated and GT roots are different.

Table 2: **HierarCaps retrieval.** We measure the precision and recall of 50 traversal results. “Prob?” denotes inclusion-based traversal, using the average of “” and “[ROOT]” in Figure 9 as a new [ROOT]. R@1<sup>[ROOT]</sup> is recall of [ROOT] embeddings.

	Prob?	Prec	R@1	R@1 <sup>[ROOT]</sup>
CLIP (1.28B)	✗	25.0	63.0	0.1
ProLIP (1.28B)	✗	28.4	62.6	0.1
ProLIP (1.28B)	✓	35.9	62.9	15.3
ProLIP (12.8B)	✗	31.7	67.8	0.1
ProLIP (12.8B)	✓	41.1	68.0	23.3

Table 3: **ImageNet prompt tuning by ProLIP.**  $K$  denotes the number of few-shot samples for each class (if required). “OpenAI 80 prompts” is the same as “ImageNet” of ProLIP 12.8B in Table 1.

Prompt strategy	Accuracy	$K$
“a photo of {·}”	73.7	-
OpenAI 80 prompts	74.6	-
Filtering by $\sigma$ stats	74.6 (+0.03)	-
Top-K prompts	74.7 (+0.07)	-
BPRW (proposed)	74.7 (+0.12)	0
BPRW (proposed)	75.6 (+0.99)	5
BPRW (proposed)	75.8 (+1.21)	9

shows the ImageNet classification results with different strategies. First, solely using text uncertainty and filtering out uncertain texts are not sufficiently effective. They only improve about +0.05pp top-1 accuracy. On the other hand, BPRW achieves better accuracy by using additional visual information; we have +0.12pp for ImageNet ZSC. If we can use a few labeled images per class (e.g., five for each class), we can get over 1.2% accuracy improvement by adjusting their weight. In Appendix C.5, we describe more details of BPRW, including hyperparameters and more visualization results of the learned weights by BPRW. Interestingly, the learned weights follow the actual image distributions; if the images are mostly low resolution, “a low resolution” prompt becomes the most significant.

## 5 CONCLUSION

In this work, we introduced Probabilistic Language-Image Pre-training (ProLIP), a fully probabilistic vision-language model that addresses the limitations of deterministic embeddings by capturing the inherent multiplicity in image-text relationships. By mapping inputs to random variables and efficiently estimating uncertainty through an “uncertainty token” ([UNC]), ProLIP models distributional relationships without additional parameters. The inclusion loss further enhances interpretability by enforcing distributional inclusion between image-text pairs and between original and masked inputs. Our experiments demonstrate that ProLIP is not only beneficial in zero-shot classification tasks but also provides additional axis of understanding input data by capturing their uncertainty. Our approach highlights the potential of uncertainty modeling in vision-language applications.

---

## AUTHOR CONTRIBUTIONS

This project is the extension of Chun et al. (2021; 2022); Chun (2024). S Chun led the project; the other authors actively and significantly contributed to the project with advice and feedback. W Kim and S Yun contributed to the baseline OpenCLIP implementation and evaluation toolkits from Kim et al. (2024); S Chun developed the main module upon the baseline implementation. The main ideas, such as [UNC], probabilistic pairwise contrastive loss, and inclusion loss, are designed and implemented by S Chun. S Park contributed to constructing the HierarImg dataset. S Chun wrote the initial version of the manuscript. S Park contributed to a better visualization. All authors contributed to the final manuscript.

## ACKNOWLEDGEMENTS

We thank the internal infrastructure team who helped to secure the DataComp 1B dataset.

## REFERENCES

- Morris Alper and Hadar Averbuch-Elor. Emergent visual-semantic hierarchies in image-text representations. In *European Conference on Computer Vision (ECCV)*, 2024.
- Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019.
- Yonatan Bitton, Nitzan Bitton Guetta, Ron Yosef, Yuval Elovici, Mohit Bansal, Gabriel Stanovsky, and Roy Schwartz. Winogavil: Gamified association benchmark to challenge vision-and-language models. *Advances in Neural Information Processing Systems*, 35:26549–26564, 2022.
- Jie Chang, Zhonghao Lan, Changmao Cheng, and Yichen Wei. Data uncertainty learning in face recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 5710–5719, 2020.
- Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 3558–3568, 2021.
- Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.
- Sanghyuk Chun. Improved probabilistic image-text representations. In *International Conference on Learning Representations (ICLR)*, 2024.
- Sanghyuk Chun, Seong Joon Oh, Rafael Sampaio De Rezende, Yannis Kalantidis, and Diane Larlus. Probabilistic embeddings for cross-modal retrieval. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021.
- Sanghyuk Chun, Wonjae Kim, Song Park, Minsuk Chang Chang, and Seong Joon Oh. Eccv caption: Correcting false negatives by collecting machine-and-human-verified image-caption associations for ms-coco. In *Eur. Conf. Comput. Vis.*, 2022.
- Karan Desai, Gaurav Kaul, Zubin Aysola, and Justin Johnson. RedCaps: Web-curated image-text data created by the people, for the people. In *NeurIPS Datasets and Benchmarks*, 2021.
- Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. Hyperbolic image-text representations. In *International Conference on Machine Learning*, pp. 7694–7731. PMLR, 2023.
- Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

- 
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. Learn. Represent.*, 2021. URL <https://openreview.net/forum?id=YicbFdNTTy>.
- Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *Advances in Neural Information Processing Systems*, 36, 2024.
- Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.
- Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8340–8349, 2021a.
- Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15262–15271, 2021b.
- Byeongho Heo, Sanghyuk Chun, Seong Joon Oh, Dongyo Han, Sangdoo Yun, Gyuwan Kim, Youngjung Uh, and Jung-Woo Ha. Adam: Slowing down the slowdown for momentum optimizers on scale-invariant weights. In *Int. Conf. Learn. Represent.*, 2021.
- Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL <https://doi.org/10.5281/zenodo.5143773>. If you use this software, please cite it as below.
- Yatai Ji, Junjie Wang, Yuan Gong, Lin Zhang, Yanru Zhu, Hongfa Wang, Jiaxing Zhang, Tetsuya Sakai, and Yujiu Yang. Map: Multimodal uncertainty-aware vision-language pre-training model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23262–23271, 2023.
- 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 *Int. Conf. Mach. Learn.*, pp. 4904–4916. PMLR, 2021.
- Wonjae Kim, Sanghyuk Chun, Taekyung Kim, Dongyo Han, and Sangdoo Yun. Hype: Hyperbolic entailment filtering for underspecified images and texts. In *European Conference on Computer Vision (ECCV)*, 2024.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Int. Conf. Learn. Represent.*, 2015.
- Michael Kirchhof, Enkelejda Kasneci, and Seong Joon Oh. Probabilistic contrastive learning recovers the correct aleatoric uncertainty of ambiguous inputs. In *International Conference on Machine Learning*, 2023.
- Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022.
- Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. Scaling language-image pre-training via masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23390–23400, 2023.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, 2014.

- 
- Andrei Neculai, Yanbei Chen, and Zeynep Akata. Probabilistic compositional embeddings for multimodal image retrieval. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 4547–4557, 2022.
- Dat Quoc Nguyen, Ashutosh Modi, Stefan Thater, Manfred Pinkal, et al. A mixture model for learning multi-sense word embeddings. In *Proc. of the 6th Joint Conference on Lexical and Computational Semantics (\* SEM 2017)*, pp. 121–127, 2017.
- Seong Joon Oh, Kevin Murphy, Jiyan Pan, Joseph Roth, Florian Schroff, and Andrew Gallagher. Modeling uncertainty with hedged instance embedding. In *Int. Conf. Learn. Represent.*, 2019.
- Jungin Park, Jiyoung Lee, Ig-Jae Kim, and Kwanghoon Sohn. Probabilistic representations for video contrastive learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 14711–14721, 2022.
- 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 *Int. Conf. Mach. Learn.*, pp. 8748–8763. PMLR, 2021.
- Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5389–5400. PMLR, 09–15 Jun 2019. URL <https://proceedings.mlr.press/v97/recht19a.html>.
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.
- Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.
- Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Association for Computational Linguistics*, pp. 2556–2565, 2018.
- Yichun Shi and Anil K Jain. Probabilistic face embeddings. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 6902–6911, 2019.
- Anna Silnova, Niko Brummer, Johan Rohdin, Themos Stafylakis, and Lukas Burget. Probabilistic embeddings for speaker diarization. In *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, pp. 24–31, 2020.
- Jennifer J Sun, Jiaping Zhao, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, and Ting Liu. View-invariant probabilistic embedding for human pose. In *Eur. Conf. Comput. Vis.*, 2020.
- Uddesha Upadhyay, Shyamgopal Karthik, Massimiliano Mancini, and Zeynep Akata. Probvilm: Probabilistic adapter for frozen vision-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1899–1910, 2023.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Adv. Neural Inform. Process. Syst.*, pp. 5998–6008, 2017.
- C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.
- Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019.

---

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Int. Conf. Comput. Vis.*, 2019.

Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11975–11986, 2023.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *Int. Conf. Learn. Represent.*, 2018.

---

## APPENDIX

### A MORE DETAILS OF PROLIP

#### A.1 2D VISUALIZATION FOR FIGURE 1

We use a linear projection, such as Principal Component Analysis (PCA), for the visualization. If we use a non-linear projection from a high-dimensional space to a 2-dimensional space, there is no guarantee that the projected Gaussian distribution still follows a Gaussian distribution. Therefore, we project the high-dimensional embeddings to 2-dimensional space using PCA. All models are trained on the DataComp 1B dataset with 1.28B seen samples (*i.e.*, the 1.28B models in Table 1).

For both models, we select ten similar images and their corresponding captions from the Hierar-Caps dataset. Then, for ProLIP, we randomly sample ten embeddings for each embedding using the learned Gaussian distributions. We then apply PCA to the sampled embeddings for visualization.

#### A.2 PROBABILISTIC MATCHING LOSS VS. PROBABILISTIC PAIRWISE CONTRASTIVE LOSS

Probabilistic matching loss (PML) by PCME++ (Chun, 2024) and the proposed probabilistic pairwise contrastive loss (PPCL) use almost the same logit based on CSD (Equation (1)). However, they have differences in (1) PML is based on the original CSD, which should compute L2-distance between  $\mu$ s and (2) PML uses binary cross entropy loss (BCE) while PPCL uses log sigmoid loss.

The first difference can make numerical errors when the difference between  $\mu$ s is extremely small. It is because L2 distance should compute a “square” operation, *i.e.*,  $\sum_{i=1}^D (a_i^2 - b_i^2)^2$ , where  $a_i$  and  $b_i$  are scalar value of  $i$ -th dimension of vector  $a$  and  $b$ . On the other hand, we use logit based on matrix multiplication  $\mu_1^\top \mu_2$  (derived in Appendix A.3), showing a more stable and accurate computation in terms of float precision.

The second difference can cause a fast gradient vanishing as already discussed by Chun (2024). Chun (2024) thus employed additional techniques to mitigate the issue, *e.g.*, pseudo positives and mixed data sample augmentation method, such as Mixup (Zhang et al., 2018) and CutMix (Yun et al., 2019). However, in this paper, we omit the techniques because simplicity is important when we train with a large-scale training samples. To tackle the issue, we employ log sigmoid loss and a multiplication-based logit, resulting in a fast and stable convergence.

In our ablation study (Appendix C.2), we empirically show that the PML loss used by PCME++ fails to be converged when we use the proposed [UNC] architecture, rather than the inefficient multi-head architecture.

#### A.3 DERIVATION OF EQUATION (2)

We first start from the fact that  $d_{L2} = \|\mu_1 - \mu_2\|_2^2 = 2 - 2\mu_1^\top \mu_2 = 2 - 2 \cdot d_{L2cos}$  when  $\mu_1$  and  $\mu_2$  are L2-normalized (*i.e.*,  $\|\mu\|_2^2 = 1$ ). Namely, a cosine similarity  $d_{L2cos} = 1 - \frac{1}{2}d_{L2}$ . By replacing  $d_{L2}$  to  $d_{CSD}$  (Equation (1)), we can conclude the derivation.

$$d_{CSDcos} = 1 - \frac{1}{2}d_{CSD} = 1 - \frac{1}{2}d_{L2} - \frac{1}{2}tr(\sigma_v^2 + \sigma_t^2) = d_{L2cos} - \frac{1}{2}tr(\Sigma_v + \Sigma_t). \quad (\text{A.1})$$

#### A.4 DERIVATION OF EQUATION (3)

Let us assume two Gaussian distributions:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left(-\frac{(x - \mu_1)^2}{2\sigma_1^2}\right), \quad q(x) = \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp\left(-\frac{(x - \mu_2)^2}{2\sigma_2^2}\right)$$

First, we take the square of  $p(x)$ :

$$p(x)^2 = \left( \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left(-\frac{(x - \mu_1)^2}{2\sigma_1^2}\right) \right)^2 = \frac{1}{2\pi\sigma_1^2} \exp\left(-\frac{(x - \mu_1)^2}{\sigma_1^2}\right)$$

Now, we compute the integral:

$$\begin{aligned} \int p(x)^2 q(x) dx &= \int_{-\infty}^{\infty} \frac{1}{2\pi\sigma_1^2} \exp\left(-\frac{(x-\mu_1)^2}{\sigma_1^2}\right) \cdot \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp\left(-\frac{(x-\mu_2)^2}{2\sigma_2^2}\right) dx \\ &= \frac{1}{2\pi\sigma_1^2\sqrt{2\pi\sigma_2^2}} \int_{-\infty}^{\infty} \exp\left(-\frac{(x-\mu_1)^2}{\sigma_1^2} - \frac{(x-\mu_2)^2}{2\sigma_2^2}\right) dx \end{aligned}$$

We expand the terms in the exponent as follows:

$$-\frac{(x-\mu_1)^2}{\sigma_1^2} - \frac{(x-\mu_2)^2}{2\sigma_2^2} = -\left(\frac{1}{\sigma_1^2} + \frac{1}{2\sigma_2^2}\right)x^2 + \left(\frac{2\mu_1}{\sigma_1^2} + \frac{\mu_2}{\sigma_2^2}\right)x - \left(\frac{\mu_1^2}{\sigma_1^2} + \frac{\mu_2^2}{2\sigma_2^2}\right)$$

Let:

$$A = \frac{1}{\sigma_1^2} + \frac{1}{2\sigma_2^2}, \quad B = \frac{2\mu_1}{\sigma_1^2} + \frac{\mu_2}{\sigma_2^2}, \quad C = \frac{\mu_1^2}{\sigma_1^2} + \frac{\mu_2^2}{2\sigma_2^2}$$

The term in the exponent simplifies to:

$$-A\left(x - \frac{B}{2A}\right)^2 + \frac{B^2}{4A} - C$$

Now, the integral becomes:

$$\frac{1}{2\pi\sigma_1^2\sqrt{2\pi\sigma_2^2}} \exp\left(\frac{B^2}{4A} - C\right) \int_{-\infty}^{\infty} \exp\left(-A\left(x - \frac{B}{2A}\right)^2\right) dx$$

Using the Gaussian integral formula, we have:

$$\int_{-\infty}^{\infty} \exp\left(-A\left(x - \frac{B}{2A}\right)^2\right) dx = \sqrt{\frac{\pi}{A}}$$

Thus, the integral becomes:

$$\int_{-\infty}^{\infty} p(x)^2 q(x) dx = \frac{1}{2\pi\sigma_1^2\sqrt{2\pi\sigma_2^2}} \exp\left(\frac{B^2}{4A} - C\right) \sqrt{\frac{\pi}{A}}$$

By taking logarithm and omitting constant terms, we have Equation (3).

## A.5 ALGORITHM OF UNCERTAINTY-AWARE ZERO-SHOT PROMPT SELECTION

**Remark:** Let  $\pi_c \in \mathbb{R}^N$  be the weight of each prompt, where  $N$  is the number of prompts (*i.e.*, 80 for ImageNet) and  $c$  is the class index. Our goal is to find the best  $\pi_i$  that the new text embedding  $Z_t^{\text{new}} = \sum_i \pi_c^i Z_t^i$  describe the given  $M$  image embeddings  $Z_v^j$ . To achieve this goal, we optimize  $\pi_c$  to have the best posterior for  $Z_t$  and  $Z_v$ . For each class  $c$ , we sample  $K$  points from each  $Z_v^j$ . Namely, we have total  $M' = M \times K$  point embeddings as observations for class  $c$ .

We assume a Dirichlet prior for  $\pi_c$  with  $\alpha$ , where  $\alpha$  controls the “uniformity” of the posterior. If we choose large  $\alpha$ , such as 10, then  $\pi$  will be more uniform and if we choose small  $\alpha$ , such as 0.1, then  $\pi$  almost becomes a Dirac-delta distribution. We also set the initial  $\pi$  with the reversed uncertainty score, *i.e.*, the normalized  $1/\text{tr}(\Sigma_t)$  for each prompt. We also employ a small trick for stability. After we got the prior value, we add a small  $\varepsilon$  to  $\Sigma_t$  to make the overall operation stable. Now, we explain the Expectation-Maximization (EM) algorithm for estimating the mixing proportions of a mixture of  $N$  Gaussian components, incorporating a Dirichlet prior over the mixing proportions. For simplicity, we omit the class index  $c$  for the remaining section.

We place a Dirichlet prior on the mixing proportions for the given  $\alpha$ :

$$p(\pi) = \text{Dirichlet}(\pi | \alpha), \tag{A.2}$$

We have a dataset of  $M'$  observations  $\{x_j\}_{j=1}^{M'}$  and assume that the data is generated from a mixture of  $N$  Gaussian distributions (*i.e.*, the number of prompts).

---

**E-Step:** Compute the responsibilities  $\gamma_{jn}$ , which represent the probability that observation  $x_j$  was generated by component  $n$ :

$$\gamma_{jn} = \frac{\pi_n f_n(x_j)}{\sum_{i=1}^N \pi_i f_i(x_j)},$$

where  $f_n(x_j)$  is the Gaussian probability density function of component  $n$  evaluated at  $x_j$ :

$$f_n(x_j) = \frac{1}{(2\pi)^{D/2} |\Sigma_n|^{1/2}} \exp\left(-\frac{1}{2}(x_j - \mu_n)^\top \Sigma_n^{-1} (x_j - \mu_n)\right).$$

Since  $\Sigma_n$  is diagonal, the computation simplifies to:

$$f_n(x_j) = \prod_{d=1}^D \frac{1}{\sqrt{2\pi\sigma_{nd}^2}} \exp\left(-\frac{(x_{jd} - \mu_{nd})^2}{2\sigma_{nd}^2}\right).$$

**M-Step:** Compute the effective number of observations assigned to each component:

$$N_n = \sum_{j=1}^{M'} \gamma_{jn}.$$

Then, update the mixing proportions for each prompt  $n$ :

$$\pi_n = \frac{N_n + \alpha - 1}{M' + \sum_{i=1}^N (\alpha_i - 1)} = \frac{N_n + \alpha - 1}{M' + N(\alpha - 1)}.$$

Ensure that  $\pi_n \geq 0$  and  $\sum_{n=1}^N \pi_n = 1$ . We can simplify the algorithm as follows:

---

**Algorithm 1** Bayesian Prompt Re-Weighting (BPRW)

---

```

1: Initialization ( $\alpha$ : the Dirichlet prior hyperparameter,  $\varepsilon$ : the stability hyperparameter)
2: for each class  $c = 1$  to  $C$  do
3:   Collect M' observations by choosing  $M$  number of  $Z_v$  and sample  $K$  points from each  $Z_v$ .
    We can choose  $M$  “ground truth” observations where  $Z_v$  is the embedding, including the  $c$ 
    class. Otherwise, we select  $M$  nearest samples from the base text prompt embedding of  $c$ .
4:   for each prompt index  $k = 1$  to  $K$  do
5:     Set initial mixing proportion:  $\pi_n \leftarrow \left(\frac{1}{Tr(\Sigma_n)}\right) / \left(\sum_{n=1}^N \frac{1}{Tr(\Sigma_n)}\right)$ .
6:     Modify covariance matrix for stability:  $\Sigma_k \leftarrow \Sigma_k + \varepsilon I$ .
7:   end for
8:   repeat
9:     // E-Step:
10:    for each observation  $j = 1$  to  $M'$  do
11:      for each prompt index  $n = 1$  to  $N$  do
12:        Compute the responsibility  $\gamma_{jn} = (\pi_n f_n(x_j)) / \left(\sum_{i=1}^N \pi_i f_i(x_j)\right)$ .
13:      end for
14:    end for
15:    // M-Step:
16:    for each prompt index  $n = 1$  to  $N$  do
17:      Compute  $N_n = \sum_{j=1}^{M'} \gamma_{jn}$ 
18:      Update mixing proportion  $\pi_n = \frac{N_n + \alpha - 1}{M' + N(\alpha - 1)}$ 
19:    end for
20:  until convergence
21: end for
```

---

---

## B MORE EXPERIMENTAL DETAILS

### B.1 HYPERPARAMETERS

We use the AdamW (Kingma & Ba, 2015) optimizer following the official openclip implementation. We also tried AdamP (Heo et al., 2021), which was the main optimizer of PCME++ (Chun, 2024). We empirically observe that AdamP can improve the overall performances (*e.g.*, average performance from 57.3 to 57.8 under the same setting), but for a fair comparison, we follow the protocol of OpenCLIP (Ilharco et al., 2021). We use learning rate of 0.0005, beta1 of 0.9, beta2 of 0.95, weight decay of 0.2 and batch size of 512 for each GPU (*i.e.*, the full batch size is  $512 \times 32 = 16384$ ). We apply 10000 warmup steps and then learning rate is decayed by cosine learning rate scheduling. We use image augmentations of scaling 0.8 to 1.0, color jittering and gray scale. Among the mini-batch, we select 12.5% image-text pairs and drop 75% tokens of them to compute  $\mathcal{L}_{\text{inc}}(x \subset x^{\text{masked}})$ . In all experiments, we fix  $\alpha_1$  and  $\alpha_2$  in Equation (6) as  $10^{-7}$  and 0.001, respectively. We tried various combinations of  $\alpha$ , but we found that the final result is relatively robust to the choice of  $\alpha$  unless we choose very large  $\alpha$ . Similarly, we tested various  $\varepsilon$  and  $c$  for the inclusion loss (Equation (5)), but as shown in Table C.5, the final result is relatively robust to the choice of the parameters. If not specified, we simply chose  $\varepsilon = -20$  and  $c = 1000$ , but for future research, we recommend using  $\varepsilon = -10$  and  $c = 10$ , which empirically performs the best (57.3 vs. 57.9 in the average performance) and is a reasonable scale.

### B.2 TRAINING AND EVALUATION DATASETS

We mainly use the DataComp 1B dataset (Gadre et al., 2024), a filtered version of the LAION-5B dataset (Schuhmann et al., 2022), as our training dataset. We had 1,121,356,767 number of valid URLs among 1.39 billion URLs and 1,118,443,492 number of unique images after de-duplicating URLs (there were 147,676,246 number of duplicated URLs in the DataComp 1B URLs). For the ablation study, we use ConceptualCaption 3M (Sharma et al., 2018), ConceptualCaption 12M (Changpinho et al., 2021) and RedCaps (Desai et al., 2021) datasets.

We use 38 tasks from the DataComp evaluation suite, including: ImageNet (Russakovsky et al., 2015), 6 benchmarks for evaluating robustness under ImageNet distribution shifts, including ImageNet-A, ImageNet-O (Hendrycks et al., 2021b), ImageNet-R (Hendrycks et al., 2021a), ImageNet v2 (Recht et al., 2019), ImageNet-Sketch (Wang et al., 2019) and ObjectNet (Barbu et al., 2019), and 13 VTAB task (Zhai et al., 2019), including Caltech-101, CIFAR-100, CLEVR Counts, CLEVR Distance, Describable Textures, EuroSAT, KITTI Vehicle Distance, Oxford Flowers-102, Oxford-IIIT Pet, PatchCamelyon, RESISC45, SVHN and SUN397. We also employ three retrieval tasks, including Flickr (Young et al., 2014), MS-COCO Caption (Chen et al., 2015) and WinoGAVIL (Bitton et al., 2022).

### B.3 HIERARCHICAL IMAGES DATASET CONSTRUCTION

We construct the hierarchical image dataset by using the validation set of the COCO dataset (Lin et al., 2014), which includes both images and their corresponding segmentation maps. First, the images were filtered to retain only those where the smallest segment was larger than 5000 pixels. Then, the three largest segments were extracted from each image and pasted onto a blank canvas in descending order of size. We illustrate the generated samples in Figure B.1.

## C MORE RESULTS

### C.1 UNCERTAIN AND LONG CAPTIONS

For example, we observe that the following single caption shows a very high uncertainty value despite its length (*e.g.*, top 20% uncertain samples in Figure 4). Note that Figure 6 shows that a caption with longer length might have a higher uncertainty, but we empirically observe that it is not always true if the caption only has almost useless information. Below, “UNC” denotes the uncertainty value of the corresponding caption, namely  $tr(\Sigma_t)$ .



**Figure B.1: Hierarchical COCO samples.**

UNC: 0.0415 "Searching for Meaning: Idealism, Bright Minds, Disillusionment, and Hope (Third in a Series of See Jane Win(tm) Books) Cover Image"

UNC: 0.0415 "Graphic design profession workdesk monitor printer books lamp pc computer stock illustration"

UNC: 0.0416 "Photo ID: 2299331 Views: 14221 UK - Air Force Eurofighter EF-2000 Typhoon FGR4 (ZK306) shot at Fairford (FFD / EGVA) UK - England July 21, 2013 By Michael Brazier-Aviation-Images"

UNC: 0.0416 "Vigo 36 inch Farmhouse Apron Single Bowl 16 Gauge Stainless Steel Kitchen Sink with Zurich Chrome Faucet, Grid, Strainer and Soap Dispenser"

UNC: 0.0418 "XIAOMI iHealth PT-101B Medical Baby High Sensitivity LED Electric Thermometer Underarm/Oral Soft Head Thermometer Adult BabyThermometer Sensor"

UNC: 0.0462 "Maidofhonortoastus Scenic Sample Invoices Created With Our Online Invoicing Software With Licious Sample Invoice Template With Comely Free Excel Invoice Also Invoice Template For Self Employed In Addition Used Vehicle Invoice And Invoice Payment Reminder As Well As Print Invoice Template Additionally It Services Invoice Template From Invoiceberrycom With Maidofhonortoastus Licious Sample Invoices Created With Our Online Invoicing Software With Comely Sample Invoice Template And Scenic Free Excel Invoice Also Invoice Template For Self Employed In Addition Used Vehicle Invoice From Invoiceberrycom 0."

UNC: 0.0462 "Carterusaus Ravishing Invoice Freewordtemplatesnet With Inspiring Proforma Invoice With Easy On The Eye Chicken Receipts Also I Receipt Notice In Addition Cash Receipts Template And Read Receipt For Gmail As Well As Annual Gross Receipts Additionally Platepass Receipt From Freewordtemplatesnet With Carterusaus Inspiring Invoice Freewordtemplatesnet With Easy On The Eye Proforma Invoice And Ravishing Chicken Receipts Also I Receipt Notice In Addition Cash Receipts Template From Freewordtemplatesnet"

UNC: 0.0466 "Aldiablosus Ravishing Addition Worksheets Dynamically Created Addition Worksheets With Entrancing Addition Worksheets With Appealing Phonics Worksheets Free Also Functions Worksheet Algebra In Addition Cellular Respiration Worksheet Middle School And Beginning Addition Worksheets As Well As Real World Math Worksheets Additionally Chemical Change Worksheet From Mathaidscom With Aldiablosus Entrancing Addition Worksheets Dynamically Created Addition Worksheets With Appealing Addition Worksheets And Ravishing Phonics Worksheets Free Also Functions Worksheet Algebra In Addition Cellular Respiration Worksheet Middle School From Mathaidscom"

## C.2 MORE ZERO-SHOT CLASSIFICATION EXPERIMENTS

We show the full results of 38 tasks in Table C.1. We can observe that in most benchmarks, ProLIP outperforms CLIP and SigLIP. We also show the result of the AdamP-optimized ViT-B/16 backbone and the fine-tuned ViT-L/16 backbone in the table and Table C.2.

In addition, we conduct the ablation study of our design choices. Appendix C.3 shows the results. We train the models on Conceptual Caption 3M (Sharma et al., 2018), Conceptual Caption 12M (Changpinyo et al., 2021) and RedCaps (Desai et al., 2021) datasets with 96M seen samples. This efficient setting enables to train a model in 7 hours with 8 A100 NVIDIA GPUs. We measure the effectiveness of our design choice in three categories. First, we measure ImageNet zero-shot top-1 accuracy (IN-Top1). Second, we measure the average  $\sigma^2$  values of visual and textual modalities.

Table C.1: **Zero-shot classification full results.** ProLIP (AdamP) denotes a model optimized by the AdamP (Heo et al., 2021) optimizer. ProLIP (ViT-L FT) denotes the fine-tuned ViT-L/16 backbone from SigLIP ViT-L/16 (Zhai et al., 2023).

	Caltech-101	CIFAR-10	CIFAR-100	CLEVR Counts	CLEVR Distance	Country211	Describable Textures	EuroSAT	FGVC Aircraft	Food-101	GTSRB	ImageNet 1k	ImageNet Sketch	ImageNet v2	ImageNet-A	ImageNet-O	ImageNet-R	KITTI Vehicle Distance	MNIST	ObjectNet
1.28B seen samples																				
CLIP	91.75	94.34	77.85	18.01	15.79	15.63	60.43	45.09	20.77	86.55	50.27	67.24	55.70	58.95	30.73	53.70	76.35	30.80	72.96	55.39
SigLIP	92.77	94.17	78.25	20.11	15.87	15.18	60.90	39.15	21.01	85.96	40.71	67.37	55.93	59.67	30.80	54.25	76.41	18.00	73.17	55.60
ProLIP	92.68	94.58	77.73	27.64	17.30	15.54	60.80	39.07	26.46	85.33	46.17	67.84	56.85	60.03	27.28	54.50	76.74	36.99	70.37	53.36
ProLIP (AdamP)	93.10	94.42	77.57	27.86	21.50	15.13	62.39	54.30	23.90	85.37	42.10	67.37	55.99	59.64	30.29	54.95	76.54	32.35	77.15	54.25
ProLIP (ViT-L FT)	94.58	98.28	87.79	23.31	16.27	26.56	70.53	71.26	49.91	94.20	52.76	79.40	70.28	72.77	67.67	33.45	92.26	19.97	85.05	75.37
ProLIP (12.8B)	96.42	83.25	29.78	15.13	21.29	66.91	60.89	91.03	52.79	74.56	63.65	66.66	50.25	45.40	86.00	32.21	84.47	65.80	81.74	61.41
Oxford Flowers-102																				
	Oxford-IIIT Pet	Pascal VOC 2007	PatchCamelyon	Rendered SST2	RESISC45	Stanford Cars	STL-10	SUN397	SVHN	iWildCam	Camelyon17	FMoW	Flickr	MSCOCO	WindGAVIL	Dollar Street	GeoDE	Average		
CLIP	69.67	88.57	81.38	56.67	50.08	58.57	83.09	96.78	66.43	59.68	11.16	56.03	10.57	71.09	45.49	43.46	56.43	86.97	57.12	
SigLIP	68.62	88.31	81.46	56.66	49.20	55.62	83.82	96.23	67.60	61.70	10.07	63.84	11.69	71.76	45.47	43.09	57.48	87.56	56.72	
ProLIP	74.04	90.05	80.91	54.05	49.70	54.70	86.49	96.31	66.60	59.00	70.45	45.65	10.42	44.96	9.53	40.46	55.84	85.68	56.90	
ProLIP (AdamP)	71.44	89.22	80.36	50.70	50.25	57.83	86.01	96.34	65.73	63.81	9.12	49.23	12.26	72.32	45.83	43.76	58.18	86.56	57.77	
ProLIP (ViT-L FT)	83.24	95.27	81.13	58.07	57.66	71.59	93.72	99.02	73.97	66.53	82.26	55.76	16.23	66.76	18.24	45.86	67.06	91.15	65.93	
ProLIP (12.8B)	54.31	68.27	91.33	97.91	71.35	72.77	57.85	79.97	53.18	93.61	38.02	78.37	93.45	45.56	12.64	15.12	62.27	90.31	63.31	

Table C.2: **More zero-shot classification results of ProLIP.** \* denotes that the backbone is fine-tuned with the number of highlighted seen samples. (AdamP) denotes that the model is optimized by AdamP.

Backbone	# Samples Seen	ImageNet	IN dist. shifts	VTAB	Retrieval	Average
ViT-B/16	1.28B	67.6	55.0	57.1	53.4	57.3
ViT-B/16 (AdamP)	1.28B	67.4	55.3	59.1	54.0	57.8
ViT-B/16	12.8B	74.6	63.0	63.7	59.6	63.3
ViT-L/16	1.28B*	79.4	68.6	64.0	61.3	65.9

If a model captures the inherent uncertainty well, we assume that it will have a higher uncertainty for captions, rather than images. Finally, we measure the HierarCaps recall, to measure whether the learned uncertainty captures the hierarchy of captions well.

Table C.3 shows that (1) PCME++ loss is not trainable when we use more limited architecture than multi-head architecture for estimating uncertainty. It supports our assumption discussed in Appendix A.2. (2) PCME++ loss becomes learnable if we use deterministic loss together. (3)  $\mathcal{L}_{\text{inc}(v \subset t)}$  enforces images to belong into texts, namely image uncertainty tends to be smaller than text uncertainty with the inclusion loss. (4)  $\mathcal{L}_{\text{inc}(x \subset x^{\text{masked}})}$  makes the embedding space can capture the hierarchy of data, namely, improves HierarCaps recall.

Although Table C.3 shows a good intuition for each loss function, we remark that the learned uncertainty might need more training time to be more accurate (Table 2). From this observation, we also compare ProLIP models by ablating the main losses on DataComp 1B with 1.28B seen samples. Table C.4 shows that using all proposed loss functions achieves the best ImageNet accuracies and VTAB results. Although it shows slightly worse performance in retrieval, we found that its performance is reasonable good compared to other variants, especially for the baseline (first row) in all measurements.

Table C.3: **Ablation study.** All models are trained on Conceptual Caption 3M (Sharma et al., 2018), 12M (Changpinyo et al., 2021) and RedCaps (Desai et al., 2021) where the number of seen samples is 96M.

Loss	Unc Arch	$\mathcal{L}_{\text{inc}(v \subset t)}$	$\mathcal{L}_{\text{inc}(x \subset x_{\text{masked}})}$	IN-Top1	avg $\sigma_v^2$	avg $\sigma_t^2$	H.Cap Recall
CLIP	-	-	-	35.5	-	-	48.4
PCME++ + CLIP	multi-head	✗	✗	33.6	0.0160	0.0077	55.2
PCME++	[UNC]	✗	✗	0.1	0.0354	0.0347	-
PCME++ + CLIP	[UNC]	✗	✗	36.1	0.2552	0.2118	53.8
ProLIP	[UNC]	✗	✗	37.4	0.3276	0.0745	44.8
ProLIP	[UNC]	✓	✗	36.8	0.0076	0.2324	46.7
ProLIP	[UNC]	✗	✓	37.5	0.3319	0.0610	47.9
ProLIP	[UNC]	✓	✓	37.0	0.0086	0.2254	54.8

Table C.4: **Impact of losses for zero-shot classification tasks.** All models are ViT-B/16 trained on DataComp 1B with 1.28B seen samples.

$\mathcal{L}_{\text{inc}(v \subset t)}$	$\mathcal{L}_{\text{inc}(x \subset x_{\text{masked}})}$	ImageNet	IN dist. shifts	VTAB	Retrieval	Average
✗	✗	67.0	54.6	56.2	<b>53.6</b>	56.6
✓	✗	67.3	54.6	56.4	53.2	57.0
✗	✓	67.4	54.4	56.4	53.2	56.7
✓	✓	<b>67.6</b>	<b>55.0</b>	<b>57.1</b>	<u>53.4</u>	<b>57.3</b>

Table C.5: **Impact of  $\epsilon$  and  $c$  for the inclusion loss.** All models are ViT-B/16 trained on DataComp 1B with 1.28B seen samples.

$\epsilon$	$c$	ImageNet	IN dist. shifts	VTAB	Retrieval	Average
-20	1000	67.6	55.0	57.1	53.4	57.3
-10	1000	67.4	55.1	57.3	53.1	57.1
-5	1000	67.7	55.2	55.9	52.9	56.6
-5	100	67.8	55.3	56.7	53.0	57.5
-5	10	68.0	55.5	56.8	53.6	57.4
-10	10	67.8	55.3	58.5	53.0	57.9

### C.3 MORE VISUAL EXAMPLES

We visualize more samples with combinations of various image and text uncertainties. Figure C.1 shows the example image-text pairs with their uncertainty values and similarity score measured by CSD (Equation (1)). The results are similar to Figure 5.

### C.4 MORE RESULTS FOR HIERARIMGS

Unlike text uncertainty (Figure 7), we observe that the visual uncertainty values are not discriminative by the levels in contrast to text uncertainty – See Figure C.2a. Lower-level images tend to have a larger average uncertainty (0.015) than the original images (0.014), but their differences are not significant between levels as texts. Instead of plotting every image in the same histogram, we plot the difference of uncertainty between the original image and its masked versions in Figure C.2 (b-d).

To understand why some images have reversed image uncertainty by their hierarchy, we visualize the images whose original image is not included by level 0 images. Interestingly, as shown in Figure C.3, we can observe that many images with improper uncertainty estimate by hierarchy actually have wrong visual semantic hierarchy with severely occluded main objects. For example, in the upper row image, the dog in the image appears at the second level, but it only reveals its part rather than the whole body. These results show that a proper filtering on the HierarImgs dataset should be required for a more reliable evaluation.

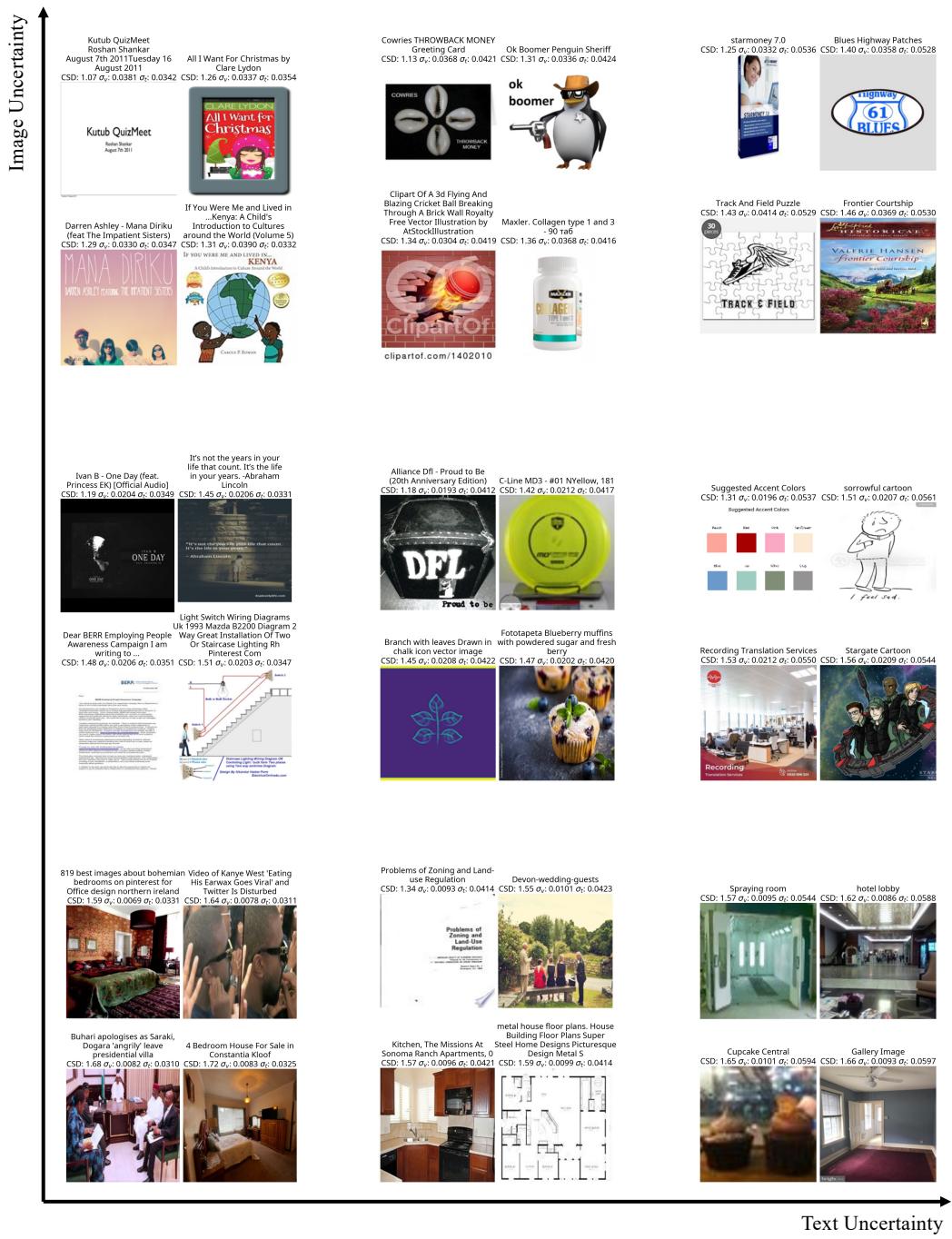


Figure C.1: **Samples with high/low image/text uncertainty.** Samples are drawn from the DataComp small subset. We also report CSD (similarity, lower is closer) between the pair,  $\sigma_v^2$ , and  $\sigma_t^2$  (lower is certain).

## C.5 MORE DISCUSSIONS FOR BAYSIAN PROMPT RE-WEIGHTING (BPRW)

**Hyperparameters.** When ground-truth labels are not accessible (*i.e.*,  $K = 0$  in Table 3), we set the  $\alpha$  to 5. Then, we select 5 nearest images from each class embedding made by 80 prompts. We then sample 10 samples from the selected image embeddings and get 100 point image embeddings. Now, the sampled embeddings are used as observation of the algorithm (See Appendix A.5). We use  $\varepsilon = 0.02$  for a stable convergence. For few-shot settings with  $K > 0$ , we set  $\alpha$  to 2 and select  $\frac{100}{K}$  samples (*e.g.*, if  $K = 9$ , then we sample 11 point vectors from the image embeddings).

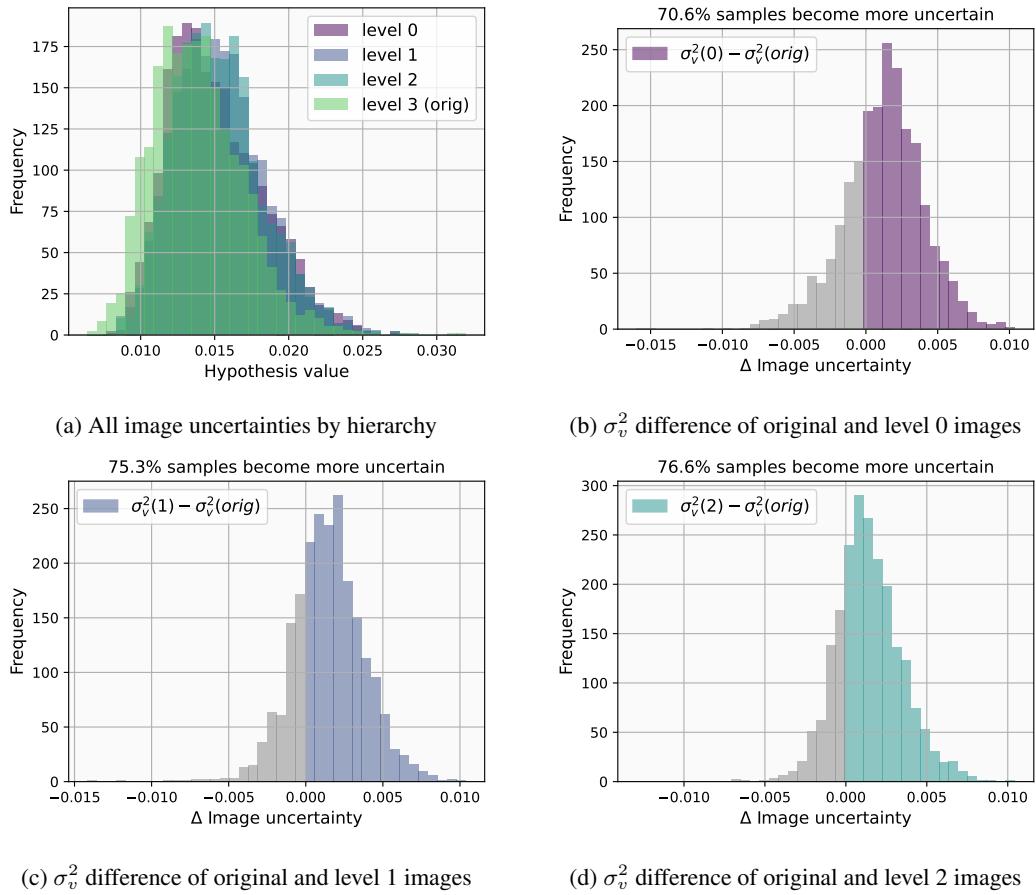


Figure C.2: **HierarImg $\sigma_v^2$  statistics.**

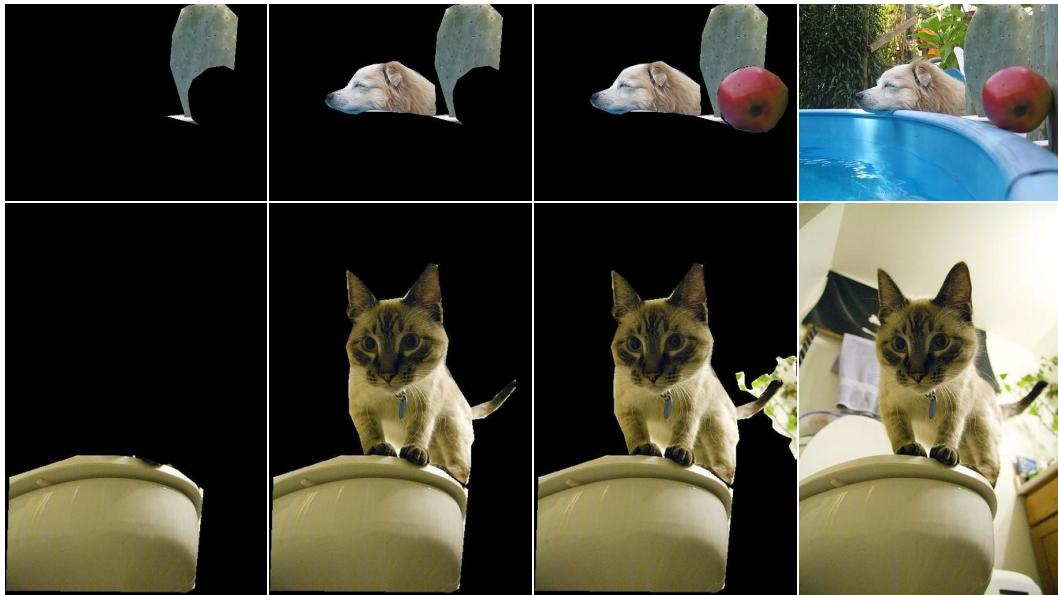
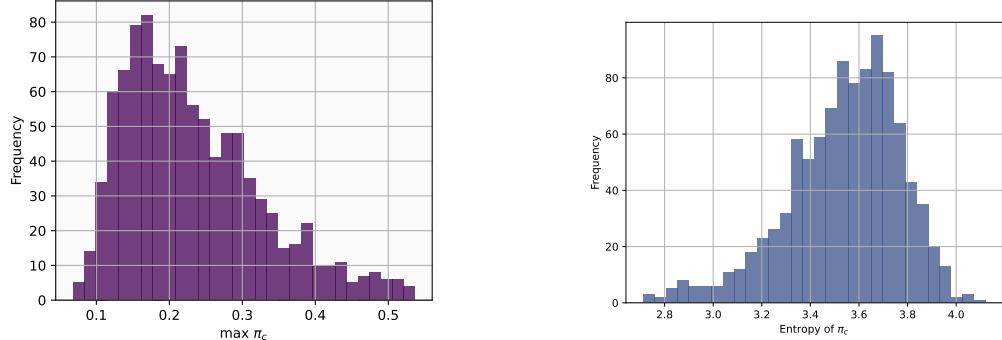


Figure C.3: **Examples of HierarImg when ProLIP fails to capture  $\sigma_v^2$  by hierarchy.**



(a) **Histogram of  $\max \pi_c$ .** The uniform distribution will have 0.001  $\max \pi$ .

(b) **Histogram of entropy of  $\pi_c$ .** The uniform distribution will have 6.9078 entropy value.

Figure C.4: **Statistics of the learned  $\pi$  by BPRW.** We use  $\pi$  obtained from the few-shot setting with  $K = 5$ .

**Visualization of the learned weight.** Using the  $\pi$  with a few-shot setting ( $K = 5$ ), we visualize the learned prompt weight. Figure C.4a shows the histogram of the maximum  $\pi_c$  for each  $c$ ; an uniform distribution will have 0.01, while larger  $\max \pi_c$  denotes that specific prompts specifically selected for the class. In the figure, we can observe that the  $\pi$  is generally larger than uniform. In addition, we plot the entropy of  $\pi$  in Figure C.4b, where it shows a similar result.

Finally, we show examples of  $\pi$  and its corresponding images. Figure C.5 shows the examples. We select three classes as examples and show their images and the learned  $\pi$ . Interestingly, for “black-footed ferret” images, we found that the context “my” has more than 0.5 weight. The actual images of black-footed ferrets are mostly composed of pet images, which makes sense that “my” prompt matches the images the most. Similarly, we observe that “front curtain” images are mostly low resolution due to the insufficient light in theaters, resulting in “a low resolution” or “a dark photo” becoming the most important prompts. Lastly, we see the missile images are mostly in the museums, resulting in “a close-up” prompt becoming the most important, but not significantly (0.1162) as much as the most contributing prompt of black-footed ferret (0.5130) and curtain (0.3921) images.

Class name: Black-footed ferret

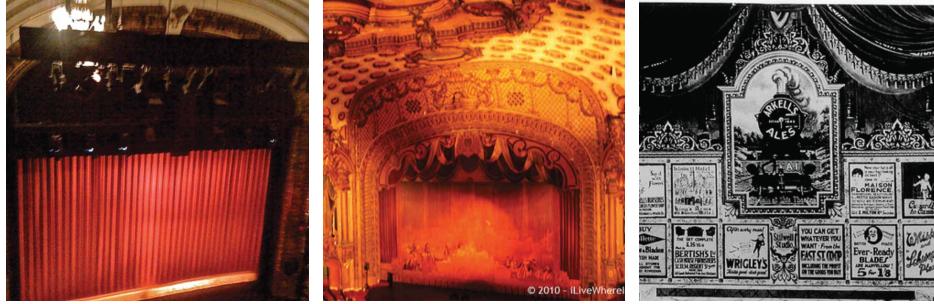


[0.5130] a photo of my black-footed ferret.  
[0.0250] itap of my black-footed ferret.

...

[0.0056] a origami black-footed ferret  
[0.0056] a sculpture of a black-footed ferret

Class name: Front curtain

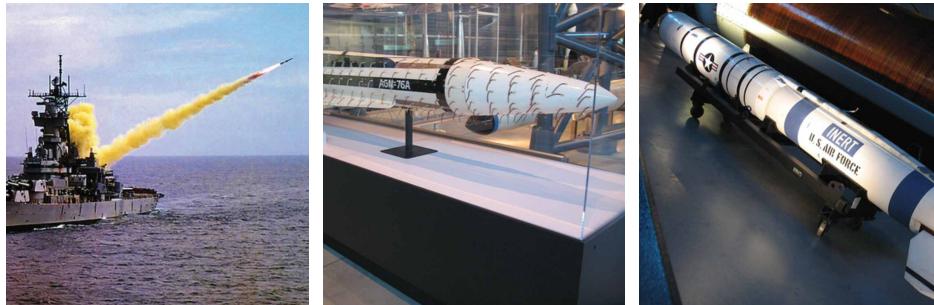


[0.3921] a low resolution photo of the front curtain.  
[0.1102] a dark photo of a front curtain.

...

[0.0056] a drawing of a front curtain  
[0.0056] a cartoon front curtain

Class name: Missile



[0.1162] a close-up photo of a missile.  
[0.0842] a black and white photo of the missile.

...

[0.0056] the plushie missile  
[0.0056] a doodle of the missile

Figure C.5: **Visualization of learned  $\pi$  by BPRW for each class.**