-Supplementary Material-

Active Prompt Learning in Vision Language Models

This supplementray material presents additional analysis and explanation of our paper, "Active Prompt Learning in Vision Language Models", that are not included in the main manuscript due to the page limitation. Appendix A analyzes the reason why VLMs make imbalance during the active learning pipeline. Appendix B addresses the method details for generating the descriptions of each class. Appendix C describes the detailed experimental settings such as datasets and active learning baselines. Also, Appendix D shows the effectiveness of our method in large datasets. Appendix E addresses the phenomenon where PCB shows reduced effectiveness compared to zero-shot pretrained CLIP. Last, Appendix F describes the additional results under not only BADGE but also Entropy and Coreset with various architectures of an image encoder.

A. Why Imbalane Occurs in VLMs

Biased knowledge of pretrained CLIP. Figure 6 indicates the zero-shot accuracy of each class when using pretrained CLIP for all the datasets. While pretrained CLIP has powerful knowledge for some classes, it has weakness for the other classes. For instance, in the case of Flowers102, pretrained CLIP has no knowledge in terms of *stemless gentian* by showing zero accuracy. On the contrary, it has perfect knowledge about *moon orchid* by indicating 100% accuracy. As such, we can conclude that imbalanced knowledge of the pretrained CLIP causes imbalanced querying by active learning algorithms.

Imbalanced dataset degrades the performance. Table 4 illustrates both accuracy and imbalance of labeled datasets after the final round. For the Oxford Pets, Stanford Cars, and FGVC aircraft datasets, where active learning algorithms can be worse than random sampling, the imbalances of Entropy and Coreset are higher than that of random sampling. It indicates that the large imbalance degrades the performance even if these datasets consist of uncertain or diversified data. As described in Section 4.3, Table 4 also shows that getting informative data enhances the accuracy after achieving a certain level of balance. For instance, despite the imbalance of BADGE being greater than that of Coreset paired with PCB, the accuracy of the combination of Coreset and PCB is still lower than that of BADGE without PCB.

B. Details for Generating Descriptions

As extending Section 3.2, we delve into generating description methods in details. In the NLP community, *few-shot learning* is one of the popular prompt engineering skills for LLMs, which enhances the performance of whole tasks. It adds a few question and answer pairs, which consist of similar types of what we ask, into the prompt. To generate the best quality descriptions for each class, we also leverage two-shot learning to LLM. Here, we show the full prompt template to get the descriptions (Figure 7). Since the text files for DTD, EuroSAT, and Oxford Pets are included in [67], we simply use them. We obtain descriptions for the remaining datasets through the use of GPT-3, whenever feasible. However, there are instances, such as with fine-grained datasets like Cars, where it proves impossible to generate descriptions for certain classes. Take, for example, the class "Audi V8 Sedan 1994" within the Cars dataset. When prompted, GPT-3 fails to provide any description, whereas GPT-3.5-turbo produces the following output: "[four-door sedan body style, Audi logo on the front grille, distinctive headlights and taillights, sleek and aerodynamic design, alloy wheels, side mirrors with integrated turn signals, V8 badge on the side or rear of the car, license plate with a specific state or country, specific color and trim options for the 1994 model year]".

C. Experimental Settings

Datasets. We select seven publicly available image classification datasets that have been previous utilized in the CLIP model. Here are the details of each dataset.

- Flowers102 [41] consists of 102 different categoris of flowers, each representing a distinct flower species, such as roses, sunflowers, and daisies. There are 8,189 image and label pairs in total. The distribution of images across categories is imbalanced, similar to typical real-world datasets, with a range of 40 to 258 samples.
- DTD [6], abbreviated from Describable Texture Dataset, is designed for a texture classification task. This dataset consists of 47 distinct classes, including the categories like fabrics and natural materials. In total, DTD comprises 5,640 samples. Notably, when examining the performance reported in [46], it becomes evdient that DTD poses a challenging problem for pre-trained CLIP models, as the texture is not easily recognizable.
- Oxford Pets [42] consists of 37 different pet categories, including various dogs and cats. This dataset contains 7,400 samples (4,978 dog images and 2,371 cat images). Additionally, it provides both class and segmentation labels for each image, though we use only the class labels in this experiment.
- EuroSAT [17] comprises 10 distinct classes that represent various land use and land cover categories. In total, this dataset includes 27,000 satelite images, with 2,700 im-

Method	Flowers102		DTD		Oxford Pets		EuroSAT		Caltech101		Stanford Cars		Aircraft	
	Acc	Imbal	Acc	Imbal	Acc	Imbal	Acc	Imbal	Acc	Imbal	Acc	Imbal	Acc	Imbal
CLIP (zero-shot)	66.7	-	44.5	-	87.0	-	49.4	-	87.9	-	59.4	-	21.2	-
Random	92.92	24.31	58.77	6.77	78.30	7.17	77.62	9.50	89.55	48.52	65.96	8.09	30.69	6.02
Entropy [18]	94.80	20.54	59.18	6.64	76.81	7.31	75.46	10.87	91.67	15.03	66.68	10.69	25.80	17.11
+ PCB	96.16	13.41	59.73	5.62	80.44	3.01	80.80	2.40	92.41	9.83	67.18	7.75	26.78	11.75
+ PCB(AE)	96.33	14.86	60.07	6.34	80.87	4.85	81.72	2.53	93.14	12.16	66.42	10.13	27.09	12.38
+ PCB(AS)	**96.94**	13.59	59.50	4.94	80.94	2.76	80.75	3.93	93.48	11.47	68.93	9.34	27.58	14.23
Coreset [50]	88.65	30.72	50.39	39.92	76.70	18.58	68.09	37.87	88.78	48.52	61.75	24.53	24.32	21.58
+ PCB	91.30	21.24	55.77	15.50	76.84	8.74	77.50	2.07	89.96	22.08	63.63	13.44	25.38	14.27
+ PCB(AE)	91.70	21.59	57.09	16.34	78.60	8.97	79.28	1.00	90.29	20.33	62.08	14.38	26.19	14.27
+ PCB(AS)	92.33	21.50	56.38	14.98	79.50	9.86	79.28	1.13	91.70	22.11	65.75	12.51	26.22	13.74
BADGE [2]	96.33	17.89	58.98	5.90	80.03	5.71	79.79	5.47	92.54	13.19	68.07	5.77	31.25	6.87
+ PCB	96.12	13.07	60.28	5.39	80.22	1.51	**81.98**	1.73	92.21	11.73	68.50	4.91	31.35	6.46
+ PCB(AE)	96.35	12.69	61.92	4.56	81.93	2.45	80.70	1.20	92.52	12.77	67.70	4.94	31.80	6.04
+ PCB(AS)	96.71	12.47	**62.33**	3.55	**83.16**	2.43	81.50	1.47	**93.85**	12.54	**70.70**	4.52	**32.27**	4.98
Full data	97.9	-	74.7	-	89.3	-	94.5	-	94.4	-	80.8	-	43.4	-

Table 4. **Final accuracy and imbalance on seven downstream tasks with the ViT-B/32 image encoder.**

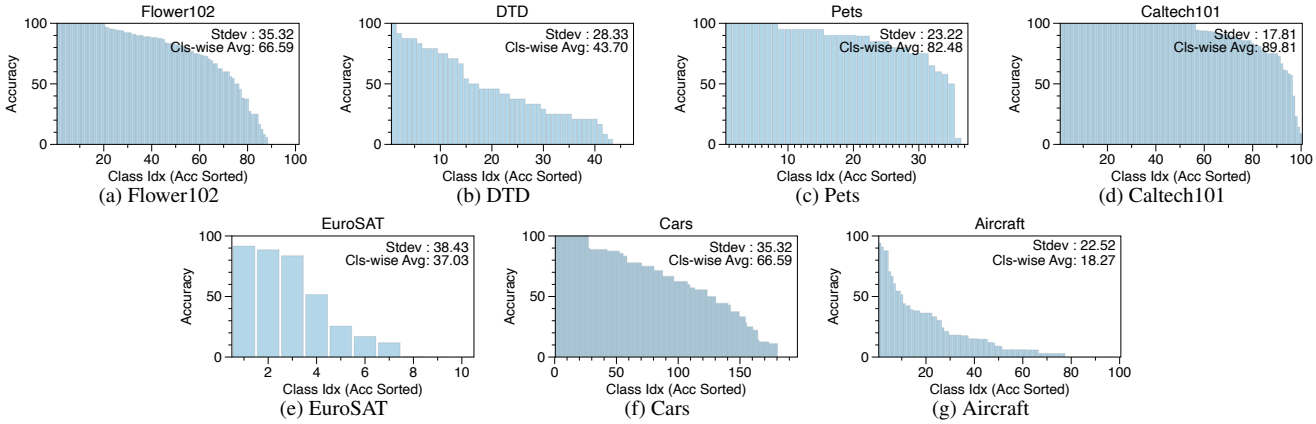

Figure 6. **Per class zero-shot accuracy from the pretrained CLIP with the ViT-B/32 image encoder for each dataset.**

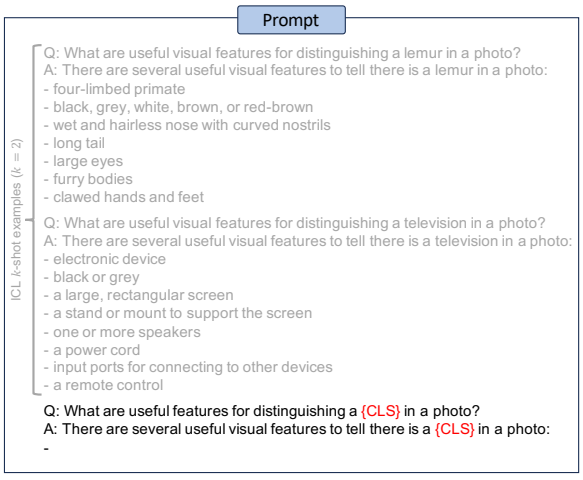

Figure 7. **Prompt template applied two-shot learning for generating descriptions.**

ages allocated to each of the 10 classes. Notably, each class contains an equal number of images, ensuring a balanced distribution within the dataset.

- Caltech101 [11] is composed of 101 unique object categories, each corresponding to a different type of objects or scenes. These categories encompass a wide range of objects, such as various animals, vehicles, and more. The dataset comprises a total of 9,000 images with varying numbers of images allocated to each category. Notably, it is considered a severely imbalanced dataset due to the uneven distribution of images across its categories.
- Stanford Cars [27] consists of a collection of 16,185 images categorized into 196 different classes, with each class typically representing a specific car make, model, and year, *e.g.* 2012 Tesla Model S.
- FGVC-Aircraft [37] encompasses a total of 10,200 images depicting various aircrafts. This dataset is organized into 102 distinct classes, and each class corresponds to

Method	ImageNet-100	Food101	SUN397	UCF101
Random	61.96	68.14	60.73	71.55
Entropy	61.26	66.47	60.57	71.48
+ PCB	62.78	68.60	61.14	72.48
Coreset	61.20	63.76	55.68	63.47
+ PCB	62.06	65.13	58.21	70.50
BADGE	62.66	69.11	61.84	74.49
+ PCB	64.42	70.45	62.80	75.84

Table 5. **Final accuracy with the ViT-B/32 CLIP image encoder on four large scaled datasets.**

a specific aircraft model variant. Notably, there are 100 images available for each of these 102 different aircraft model variants. A class is named using the make, model, and specific variant, *e.g.* Boeing 737-76J.

Active learning methods. To validate the effectiveness of PCB, we select three representative active learning methods.

1. Entropy [18] selects the most uncertain examples with the highest entropy value from logits in the prediction. Specifically, a selected query set $\mathcal{Q}$ with size d is defined as follows:

$$\mathcal{Q} = \underset{Q \subset \mathcal{D}_u, |Q|=d}{\arg\max} \sum_{x_i \in Q} \mathcal{H}\left(f(x_i)\right),$$

where $\mathcal{H}(f(x))$ denotes the entropy of a softmax output $f(x)$.

2. Coreset [50] queries the most diverse examples using embeddings from the model (*i.e.* image encoder). More precisely, it selects the examples that are the least relevant to the queried dataset. For this purpose, the authors proposed the K-Center-Greedy and Robust K-Center algorithms, and we choose the former one.
3. BADGE [2] considers both uncertainty and diversity by selecting the examples via k-means++ clustering in the gradient space. The gradient embeddings for all examples are defined as follows:

$$g_x = \frac{\partial}{\partial \theta_{\text{out}}} \mathcal{L}_{CE}(f(x; \theta_t), \hat{y}(x)) \tag{1}$$

where θ_t and θ_{out} refer to the parameters of the model and the final layer at round t, respectively. $\hat{y}(x)$ denotes the pseudo label.

D. Large Dataset

Datasets. We select additional four large datasets that have been previously utilized in the CLIP model. Due to limited resources, we utilized the subset that consists of 16 samples per class from the original dataset. Here are the details of each dataset.

- ImageNet-100 is a subset from ImageNet [9], consisting of randomly selected 100 categories. ImageNet is a substantial collection of images, with 1,281,167 designated for training, 50,000 set aside for validation, and 100,000 for testing purposes.
- Food-101 [4] consists of 101 food categories with 750 training and 250 test images per category, making a total of 101,000 images. The labels for the test images have been manually cleaned, while the training set contains some noise.
- SUN397 [57] is the database for scene regonition, which contains 397 categories and 130,519 images.
- UCF101 [53] is an extension of UCF50 and consists of 13,320 video clips, which are classified into 101 categories. These 101 categories can be classified into 5 types (Body motion, Human-human interactions, Human-object interactions, Playing musical instruments, and Sports). The total length of these video clips is over 27 hours. All the videos are collected from YouTube and have a fixed frame rate of 25 FPS with the resolution of 320 × 240. In this work, the middle frame of each video is used as an input to the image encoder.

Method	N	Model RN50	RN101	ViT-B/32	ViT-B/16
CLIP (zero-shot)	0	85.4	86.2	87.0	88.9
Random	37	74.65±0.50	79.08±1.39	78.30±0.74	84.36±1.34
BADGE [2]	37	75.06±0.50	80.77±1.31	80.03±1.19	85.54±1.30
+ PCB	37	76.51±1.83	80.94±0.42	80.22±1.69	86.22±0.71
+ PCB(AE)	37	76.77±0.65	83.02±0.89	81.93±0.88	87.23±0.35
+ PCB(AS)	37	80.09±0.85	83.48±2.13	83.16±0.18	88.10±1.49
BADGE [2]	148	86.48±0.12	89.10±0.16	88.28±0.33	91.74±0.23
+ PCB	148	86.70±0.07	89.19±0.21	88.03±0.33	91.92±0.81
+ PCB(AE)	148	86.54±0.55	89.52±0.17	88.30±0.34	91.72±0.07
+ PCB(AS)	148	87.74±0.32	90.16±0.12	89.29±0.15	92.64±0.14
Full data	-	88.0	91.1	89.3	92.7

Table 6. **Ablation study as increasing query size N on Oxford Pets.** It shows that the small amount of training set is the crucial reason why finetuning methods underperform zero-shot CLIP.

Results. Table 5 presents further experimental results on the four large datasets, following the outcomes shown in Table 1. Due to limited resources, we conducted our experiments without description augmentation, applying only PCB, and all the experiments are conducted only once. When comparing these results to the baselines, we observed that using PCB has 1%–2% points performance improvement compared to only employing conventional active learning techniques, and it is a similar trend to what was observed in Table 1.

E. Larger Size of N for Oxford Pets

As shown in Table 1 and Table 2, zero-shot CLIP outperforms PCB combined with all the active learning algorithms in the case of Oxford Pets. Here, Table 6 shows that increasing query size N enhances the performance. The performance when N is 4 times of the number of classes (*i.e.* 148) surpasses the performance when N is the number of classes (*i.e.* 37) with 4%–7% points for all the architec-

tures of an image encoder. Moreover, PCB (AS) combined with BADGE when N=148 almost reaches the performance when training with all the data (Full data). Through this phenomenon, setting an appropriate query size N is important to achieve the performance that we expect, and it should be determined by learning difficulty of the dataset.

F. Additional Results

Table 2 indicates the performance on various types of architectures of an image encoder under BADGE active learning. To extend it, we conduct the experiment on various types of architectures under not only BADGE but also Entropy and Coreset, and summarize the results in Table 7. Regardless of the architecture types of the image encoder, PCB combined with BADGE still has the best performance among the other baselines, but sometimes, PCB combined with Entropy beats combination of PCB and BADGE by a narrow margin. It indicates that a subset P sampled through Entropy has many informative examples similar to a subset P sampled through BADGE, where the size of P is 10% of the whole dataset.

Model	Method	Final Accuracy (↑) Flowers102	DTD	Oxford Pets	EuroSAT	Caltech101	Stanford Cars	Aircraft	Avg Acc (↑)
RN50	CLIP (zero-shot)	65.9	41.7	85.4	41.1	82.1	55.8	19.3	55.9
	Random	92.06±0.54	56.62±0.97	74.65±0.50	79.10±2.31	84.11±0.75	61.34±0.57	29.15±0.32	68.18
	Entropy [18]	95.19±0.09	57.62±2.13	72.74±0.97	75.73±4.28	88.21±0.42	61.32±0.80	25.13±0.96	67.99
	+ PCB	95.30±0.59	56.44±0.39	75.49±0.45	81.69±1.63	88.78±0.43	62.02±0.17	25.75±0.35	69.35
	+ PCB(AE)	95.75±0.23	59.02±0.59	76.59±0.12	81.77±1.51	89.41±0.53	61.05±0.99	26.44±0.81	70.00
	+ PCB(AS)	96.17±0.27	**59.34**±1.09	78.59±1.41	**83.26**±0.35	90.49±0.02	63.52±0.31	26.46±0.99	71.12
	Coreset [50]	85.02±1.51	48.74±1.00	69.87±2.36	70.02±4.16	83.34±1.33	57.93±0.56	25.38±0.62	62.90
	+ PCB	88.79±0.98	51.63±0.30	71.75±1.64	77.74±2.13	85.54±0.84	58.67±0.37	25.33±0.63	65.64
	+ PCB(AE)	89.27±1.69	51.69±1.25	73.70±0.27	77.74±3.33	86.69±0.57	57.63±0.55	25.17±0.37	65.98
	+ PCB(AS)	89.50±1.39	53.15±1.37	75.53±1.64	79.79±1.06	87.15±1.14	60.61±0.54	25.88±0.10	67.37
	BADGE [2]	95.56±0.54	58.35±1.20	75.06±0.50	80.94±0.55	89.67±0.30	63.96±0.53	28.12±1.03	70.24
	+ PCB	95.66±0.28	57.41±0.17	76.51±1.83	80.06±0.97	89.06±0.21	63.18±0.77	29.23±0.35	70.16
	+ PCB(AE)	95.72±0.31	59.20±1.25	76.77±0.65	81.96±0.60	89.57±0.19	62.62±0.26	28.85±1.59	70.67
	+ PCB(AS)	**96.18**±0.07	59.14±1.08	**80.09**±0.85	81.60±2.89	**90.76**±0.34	**66.20**±0.69	**29.61**±0.78	**71.94**
	Full data	97.6	71.6	88.0	93.6	92.8	78.8	42.6	80.71
RN101	CLIP (zero-shot)	65.7	43.9	86.2	33.1	85.1	62.3	19.5	56.54
	Random	92.87±0.43	58.29±1.24	79.08±1.39	77.21±4.13	87.55±0.75	70.02±0.36	32.76±0.29	71.11
	Entropy [18]	96.26±0.11	57.17±1.54	78.63±0.99	74.88±1.26	91.02±0.48	70.09±0.16	27.49±0.69	70.79
	+ PCB	96.26±0.25	58.81±1.39	80.14±1.27	79.91±2.06	91.62±0.30	70.87±0.45	28.11±0.38	72.25
	+ PCB(AE)	96.47±0.39	59.81±1.34	82.65±0.99	81.23±1.26	92.16±0.90	70.14±0.56	27.96±1.63	72.92
	+ PCB(AS)	**96.49**±0.17	60.70±1.09	**83.64**±1.02	**82.43**±1.35	92.86±0.20	73.62±0.67	28.68±0.83	74.06
	Coreset [50]	87.90±0.92	52.23±1.76	74.02±1.81	66.62±0.54	87.23±1.18	65.83±0.43	26.37±0.42	65.74
	+ PCB	91.08±0.37	54.75±2.93	76.43±1.61	75.39±1.94	89.36±0.28	66.97±0.75	27.28±0.33	68.75
	+ PCB(AE)	91.61±1.30	56.38±1.55	77.11±1.86	76.99±0.65	89.90±0.06	65.38±0.62	27.72±0.39	69.30
	+ PCB(AS)	91.80±0.28	57.31±2.07	81.14±0.24	78.49±1.99	90.11±0.30	69.11±0.73	28.31±0.78	70.90
	BADGE [2]	96.26±0.07	59.93±1.25	80.77±1.31	78.23±2.22	91.35±0.32	71.43±0.97	32.56±0.64	72.93
	+ PCB	95.79±0.38	60.20±1.89	80.94±0.42	79.55±1.37	91.75±0.44	71.35±0.39	32.62±1.48	73.17
	+ PCB(AE)	**96.49**±0.26	**62.59**±0.84	83.02±0.89	81.50±0.69	92.51±0.32	71.42±0.77	32.76±0.76	74.33
	+ PCB(AS)	96.47±0.18	62.17±1.04	83.48±2.13	81.14±1.57	**92.87**±0.18	**74.04**±0.39	**32.84**±0.85	**75.43**
	Full data	97.8	74.2	91.1	92.9	94.7	83.7	46.0	82.91
ViT-B/16	CLIP (zero-shot)	70.4	46.0	88.9	54.1	88.9	65.6	27.1	63.0
	Random	94.98±0.06	62.63±1.81	84.36±1.34	81.14±1.83	90.95±0.85	73.62±0.30	38.88±0.25	75.22
	Entropy [18]	97.63±0.42	62.49±0.39	82.56±0.49	77.93±0.90	93.04±0.41	74.35±0.59	33.27±0.72	74.47
	+ PCB	97.75±0.08	64.93±1.02	84.89±0.59	83.48±1.37	94.23±0.23	75.68±0.26	36.03±0.43	76.71
	+ PCB(AE)	98.06±0.35	64.36±0.47	87.08±0.90	83.55±1.95	94.56±0.34	75.15±0.55	35.60±1.58	76.91
	+ PCB(AS)	**98.48**±0.14	63.81±1.24	88.03±0.60	**85.92**±0.85	94.89±0.28	77.58±0.43	35.84±1.71	77.79
	Coreset [50]	92.12±1.45	56.07±0.90	82.17±1.82	72.17±2.72	90.66±0.45	70.12±0.83	33.28±0.45	70.94
	+ PCB	94.79±0.31	59.07±0.63	83.09±1.19	80.25±3.12	90.60±0.80	71.27±0.19	34.06±0.66	73.30
	+ PCB(AE)	94.94±0.55	60.54±0.86	84.52±0.23	84.04±2.92	92.15±0.09	70.10±1.03	33.36±0.03	74.24
	+ PCB(AS)	95.44±0.82	61.98±1.04	86.77±0.69	83.85±2.45	92.97±0.29	72.96±0.63	35.24±0.49	75.60
	BADGE [2]	97.97±0.41	62.84±2.17	85.54±1.30	82.22±1.94	93.77±0.51	76.55±0.78	39.64±0.14	76.93
	+ PCB	98.32±0.21	64.89±1.45	86.22±0.71	81.53±3.11	93.75±0.28	76.36±0.27	40.20±0.30	77.32
	+ PCB(AE)	98.21±0.21	**65.25**±1.28	87.23±0.35	84.04±2.92	94.51±0.29	75.84±0.44	39.93±0.21	77.86
	+ PCB(AS)	98.19±0.17	64.95±1.47	**88.10**±1.49	83.85±2.45	**95.12**±0.26	**78.19**±0.48	**40.56**±0.51	**78.42**
	Full data	99.0	77.7	92.7	95.1	95.3	85.3	53.6	85.53

Table 7. **Various architectures of an image encoder as an extension of Table 2.** We include both all the conventional active learning algorithms and PCB combined with them in terms of various architectures of the image encoder.