

Supplementary Materials: CLIPCleaner: Cleaning Noisy Labels with CLIP

ANONYMOUS AUTHORS

A SAMPLE SELECTION WITH OTHER VISION-LANGUAGE MODELS

Here, we compare CLIP with another vision-language model - ALIGN [3]. Specifically, we compare their performance on sample selection based on the CIFAR10 dataset with instance-dependent noise [2]. In table 1, we can see ALIGN behaves similarly well as CLIP concerning precision with even higher recall. This demonstrates that our proposed idea of using vision-language models for sample selection is widely effective.

Table 1. Precision-Recall of sample selection results on CIFAR10 with instance-dependent noise with CLIP and ALIGN.

Noise ratio	0.1		0.2		0.3		0.4	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
CLIP	99.73	70.75	99.53	75.07	99.25	77.77	99.03	79.23
ALIGN	99.47	72.47	99.13	78.64	99.01	81.22	98.74	84.53

B PROMPTS GENERATION AND FURTHER ANALYSIS

How multiple prompts with class-specific features are generated? Regarding the generation multiple prompts based on class-specific features, motivated by recent work [5], we first generate multiple features for each class by asking ChatGPT about each category’s characteristics. We use below question for ChatGPT 3.5:

For CLIP model, the prompts matter a lot. can you give me some discriminative features of some classes? Please list it in nested python as each class has multiple descriptions. Please ensure it is formatted as ‘which has ...’ or ‘which is ...’ or ‘which ...’. For example, [‘Cat’, ‘Lynx’, ‘Wolf’, ‘Coyote’, ‘jaguar’, ‘Cheetah’, ‘Chimpanzee’, ‘Orangutan’, ‘Hamster’, ‘Guinea pig’].

We then generate multiple prompts with template in SECTION 3.2. We will include our generated class-specific prompts along with the code upon acceptance.

Comparison of class-specific prompts with other prompt style. To experimentally validate the superiority of our prompt style based on class-specific features, we conduct a comparative analysis of its zero-shot classification performance against alternative prompt styles. Specifically, we consider three empirical variants including ours:

- (1) Single prompt: ‘A photo of {class name of y_i }.’;
- (2) Multiple prompts with different templates: ‘A good photo of {class name of y_i }.’/‘An old picture of {class name of y_i }.’ .etc;
- (3) Multiple prompts with class-specific features: ‘A photo of {class name of y_i }, which is/has {class-specific feature j of class y_i }.’ with features such as the color, shape, etc.

In table 2, we present zero-shot classification results on six noisy datasets using the three prompt styles mentioned above and different backbones for CLIP model (VIT-B/32 and VIT-L/14@336px). We observe that, in most cases, the effectiveness of our prompting style is at its best, especially when employing a larger-scale CLIP backbone (VIT-L/14@336px). This aligns with our theoretical analysis.

Table 2. Zero-shot classification with different prompt styles.

Model	Prompt technique	CIFAR10	CIFAR100	Red Mini-ImageNet	WebVision	Clothing1M	ANIMAL-10N
CLIP (ViT-B/32)	1	88.29	61.62	74.40	72.40	39.80	75.08
	2	89.73	63.65	75.14	68.12	39.68	75.70
	3	87.97	63.72	78.12	73.36	37.73	74.62
CLIP (ViT-L/14@336px)	1	94.78	74.36	80.20	45.13	85.18	85.12
	2	95.17	74.96	79.88	47.26	85.78	87.00
	3	95.19	76.78	81.96	48.15	85.36	87.98

C UTILIZING CLIPCLEANER WITH OTHER METHODS

Current mainstream methods for learning with noisy labels usually involve an iterative process consists of sample selection and model training. Normally, these methods require a warm-up stage, i.e., training with whole dataset for some epochs to learn an usable model before the iteration process. Here, we consider to utilize the selected samples by *CLIPCleaner* for the warmup stage to validate if it can also bring improvement for existing methods. In table 3, we validate that *CLIPCleaner* brings steady improvement over original DivideMix [4].

Table 3. Testing accuracy (%) of *CLIPCleaner* utilized along with DivideMix.

Dataset	CIFAR10					CIFAR100			
Noise type	Symmetric				Assymmetric	Symmetric			
Noise ratio	20%	50%	80%	90%	40%	20%	50%	80%	90%
Cross-Entropy	86.8	79.4	62.9	42.7	85.0	62.0	46.7	19.9	10.1
Co-teaching+ [12]	89.5	85.7	67.4	47.9	-	65.6	51.8	27.9	13.7
F-correction [7]	86.8	79.8	63.3	42.9	87.2	61.5	46.6	19.9	10.2
PENCIL [11]	92.4	89.1	77.5	58.9	88.5	69.4	57.5	31.1	15.3
LossModelling [1]	94.0	92.0	86.8	69.1	87.4	73.9	66.1	48.2	24.3
DivideMix [4]	96.1	94.6	93.2	76.0	93.4	77.3	74.6	60.2	31.5
DivideMix + CLIPCleaner	96.3 (0.2↑)	95.7 (1.1↑)	94.3 (1.1↑)	88.7 (12.7↑)	94.2 (0.8↑)	78.1 (0.8↑)	75.2 (0.6↑)	71.3 (9.1↑)	46.6 (15.1↑)

D PER-CLASS SEPERATE GMM VS WHOLE SINGLE GMM

In this section, we compare the differences between using seperate GMM for each class and a single GMM for all classes in sample selection. We conduct experiments on the CIFAR10 dataset with instance-dependent noise. As shown in table 4, we observe that the seperate GMM yields a higher recall while maintaining competitive precision in sample selection. In table 5, we find and validate that the seperate GMM allows us to obtain a more balanced subset, thereby mitigating class imbalance issues and partially explaining why we achieve a better recall above.

Table 4. Precision and recall of sample selection on CIFAR10 dataset with instance-dependent noise with Separate and Single GMM.

Noise ratio	0.1		0.2		0.3		0.4	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
Separate GMM	99.73	70.75	99.53	75.09	99.25	77.77	99.04	79.26
Single GMM	99.77	68.90	99.61	71.88	99.43	73.68	99.29	72.67

Table 5. Max-Min number of selected samples from each class.

Noise ratio	0.1		0.2		0.3		0.4	
	Max	Min	Max	Min	Max	Min	Max	Min
Separate GMM	4061	2228	3938	2148	3656	1851	3312	1440
Single GMM	4188	1720	4038	1757	3682	1455	3403	947

E FULL DERIVATION IN SECTION 3.1 AND SECTION 3.2

In this section, we first provide the full derivation of the weighted empirical risk and the solution of optimal weight. We then briefly explain the relation of output similarity in CLIP model and the joint probability.

Weighted empirical risk minimization in sample selection. For better clarification, we here repeat the problem formulation in Section 3.1. Given a dataset of training samples $(\mathbf{x}_i, y_i)_{i=1}^N$ *i.i.d* sampled from a hidden joint distribution $P(\mathbf{x}, y)$ with $\text{supp}(P) = \{\mathbf{x} \in \mathbb{R}^{C \times H \times W}, y \in \{1, \dots, K\}\}$ and K denotes the number of semantic classes, the goal of supervised learning is to learn a model f that can accurately predict the true labels y for new, unseen examples. Mathematically, we often optimize the empirical risk with samples *i.i.d* sampled from noisy distribution $P(\mathbf{x}, y)$:

$$\hat{R}^P(f) = \frac{1}{N} \sum_{i=1}^N L(\mathbf{x}_i, y_i; f)$$

Here L can be any applicable classification-calibrated surrogate loss to 0-1 loss, normally we use *Cross-Entropy loss*:

$$L(\mathbf{x}_i, y_i; f) = -\log \frac{\exp(f(\mathbf{x}_i)_{y_i})}{\sum_{j=1}^K \exp(f(\mathbf{x}_i)_j)}.$$

Owing to the *ERM principle*, we can uniformly minimize *w.r.t* the expected risk by minimizing above empirical risk:

$$R^P(f) = E_{P(\mathbf{x}, y)} L(\mathbf{x}, y; f)$$

However, in this work we focus on learning with noisy labels, that is to say, there exist discrepancy between the noisy training distribution $P(\mathbf{x}, y)$ and clean unknown distribution $P^{true}(\mathbf{x}, y)$. In this condition, for the same specific model f , we have the expected risk on real distribution as:

$$R^{true}(f) \triangleq R^{P^{true}}(f) = E_{P^{true}(\mathbf{x}, y)} L(\mathbf{x}, y; f)$$

To bridge the distribution discrepancy, we can easily find that:

$$R^{true}(f) = E_{P^{true}(\mathbf{x}, y)} L(\mathbf{x}, y; f) = E_{P(\mathbf{x}, y)} \frac{P^{true}(\mathbf{x}, y)}{P(\mathbf{x}, y)} L(\mathbf{x}, y; f).$$

Further, we assume $\frac{P^{true}(\mathbf{x}, y)}{P(\mathbf{x}, y)} = \frac{P^{true}(y|\mathbf{x})P^{true}(\mathbf{x})}{P(y|\mathbf{x})P(\mathbf{x})} = \frac{P^{true}(y|\mathbf{x})}{P(y|\mathbf{x})}$ as label noise normally does not affect the sample itself ($P(\mathbf{x}) = P^{true}(\mathbf{x})$). We then get the corresponding weighted empirical risk with noisy labels,

$$\hat{R}^{true}(f) = \frac{1}{N} \sum_{i=1}^N \frac{P^{true}(y_i|\mathbf{x}_i)}{P(y_i|\mathbf{x}_i)} L(\mathbf{x}_i, y_i; f)$$

with which we can ensure a risk-consistent classifier *w.r.t* clean distribution learned with even noisy labels.

More than sample selection? Other than sample selection, another applicable direction is the so-called *risk-consistent* methods, for example, to estimate the noise transition matrix by assuming $P^{true}(y^{true}|\mathbf{x}) = T(y^{true}|\mathbf{y})P(\mathbf{y}|\mathbf{x})$. A common assumption here is to assume the noise transition is instance-independent and label-dependent only thus to alleviate it from $T(y^{true}|\mathbf{y}, \mathbf{x})$ to $T(y^{true}|\mathbf{y})$. Please refer to related paper [9, 10] for more details. Though theoretically consistent, these methods often achieves relative sub-optimal performance, since noise modes in real-world datasets are extremely complex, and current noise models cannot accurately simulate them.

Relation of joint probability $Q(\mathbf{x}_i, \mathbf{z}_i)$ and the CLIP similarity $\exp(g(\mathbf{x}_i)^T h(\mathbf{z}_i))$. The zero-shot classification paradigm in eq. (4) is widely applied, however without clear theoretical explanation. In this work, we bridge the CLIP model with zero-shot classification by eq. (3). Please note we mark the equations from the main paper are colored by blue. We here explain the probabilistic relation of the learned similarity value ($\exp(g(\mathbf{x}_i)^T h(\mathbf{z}_i))$) and the joint probability $Q(\mathbf{x}_i, \mathbf{z}_i)$. Specifically, we can easily write the empirical risk with CLIP loss function in eq. (2) as:

$$\begin{aligned}\hat{R}^Q(g, h) &= \frac{1}{2} \sum_{i=1}^M \left(-\log \frac{\exp(g(\mathbf{x}_i)^T h(\mathbf{z}_i))}{\sum_{j=1}^M \exp(g(\mathbf{x}_j)^T h(\mathbf{z}_i))} - \log \frac{\exp(g(\mathbf{x}_i)^T h(\mathbf{z}_i))}{\sum_{j=1}^M \exp(g(\mathbf{x}_i)^T h(\mathbf{z}_j))} \right) \\ &= -\frac{1}{2} \log \prod_{i=1}^M \frac{\exp(g(\mathbf{x}_i)^T h(\mathbf{z}_i))}{\sum_{j=1}^M \exp(g(\mathbf{x}_j)^T h(\mathbf{z}_i))} \frac{\exp(g(\mathbf{x}_i)^T h(\mathbf{z}_i))}{\sum_{j=1}^M \exp(g(\mathbf{x}_i)^T h(\mathbf{z}_j))}\end{aligned}$$

For the specific *i.i.d* sampled dataset, based on *MLE principle* we have the negative log-likelihood as:

$$\begin{aligned}\mathcal{L}(g, h; (\mathbf{x}_i, \mathbf{z}_i)_{i=1}^M) &= -\log \prod_{i=1}^M Q_{g,h}(\mathbf{x}_i | \mathbf{z}_i, \mathbf{x} \in \{\mathbf{x}_j\}_{j=1}^M; g, h) Q_{g,h}(\mathbf{z}_i | \mathbf{x}_i, \mathbf{z} \in \{\mathbf{z}_j\}_{j=1}^M; g, h) \\ &= -\log \prod_{i=1}^M \frac{Q_{g,h}(\mathbf{x}_i, \mathbf{z}_i)}{\sum_{j=1}^M Q_{g,h}(\mathbf{x}_j, \mathbf{z}_i)} \frac{Q_{g,h}(\mathbf{x}_i, \mathbf{z}_i)}{\sum_{j=1}^M Q_{g,h}(\mathbf{x}_i, \mathbf{z}_j)}\end{aligned}$$

Comparing $\hat{R}^Q(g, h)$ with $\mathcal{L}(g, h; (\mathbf{x}_i, \mathbf{z}_i)_{i=1}^M)$, we have: $\exp(g(\mathbf{x}_i)^T h(\mathbf{z}_j)) \propto Q_{g,h}(\mathbf{x}_i, \mathbf{z}_j)$, where latter serves as an estimation of $Q(\mathbf{x}_i, \mathbf{z}_j)$ after training.

F DERIVATION OF THEOREM 1 AND THEOREM 2

In this section, we provide full derivation of THEOREM 1 and THEOREM 2 in Section 3.4. To start with, we first state the essential generalization error bound based on Rademacher complexity (\mathfrak{R}):

LEMMA F.1 (RADEMACHER GENERALIZATION ERROR BOUND [6]). *Supposing we have N i.i.d samples $\{\mathbf{x}_i\}_{i=1}^N$ from distribution $P(\mathbf{x})$. Let \mathcal{F} be the hypothesis space of model f and L be any classification-calibrated surrogate loss function of 0-1 loss ranging from $[a, b]$. Then, for any $\delta > 0$, with probability at least $1 - \delta$ we have the following holds for all $f \in \mathcal{F}$:*

$$R(f) \leq \hat{R}(f) + 2\mathfrak{R}(\mathcal{F}) + (b - a) \sqrt{\frac{\log(1/\delta)}{2N}}$$

Here, $R^P(f) = E_{P(\mathbf{x})} L(\mathbf{x}; f)$ denotes the expected risk with f and $\hat{R}^P(f) = \frac{1}{N} \sum_{i=1}^N L(\mathbf{x}_i; f)$ denotes the empirical one. Please do not confuse the notations here with other notations.

F.1 Derivation of THEOREM 1

Let us recall the formulation of CLIP model. CLIP aims to learn from a dataset of image-text pairs, denoted as $(\mathbf{x}_i, \mathbf{z}_i)_{i=1}^M$, which is *i.i.d.* sampled from a hidden joint distribution $Q(\mathbf{x}, \mathbf{z})$ with $\text{sup}(Q) = \{\mathbf{x} \in \mathbb{R}^{C \times H \times W}, \mathbf{z} \in \mathbb{R}^d\}$. As the dataset

for training CLIP is often also considered ‘noisy’¹. Here, we denote the clean joint distribution for CLIP training dataset as $Q'(\mathbf{x}, \mathbf{z})$ and the corresponding clean dataset as $(\mathbf{x}_i, \mathbf{z}'_i)_{i=1}^M$.

According to eq. (3) in main paper, to measure the distance between $\tilde{P}_{zeroshot}(y|\mathbf{x})$ with $P^{true}(y|\mathbf{x})$, we then divide it into two parts, i.e, the distance between $\tilde{P}_{zeroshot}(y|\mathbf{x})$ and $Q'(y|\mathbf{x})$ (*Model error*) and the distance between $Q'(y|\mathbf{x})$ and $P^{true}(y|\mathbf{x})$ (*Domain gap*).

On the one hand, we simply define the *domain gap* as ε_{domain} here, which represents how different the true prediction distribution ($Q'(y|\mathbf{x})$) of CLIP training dataset is than the true prediction distribution ($P^{true}(y|\mathbf{x})$) of out targeted classification problem. This is technically irreducible but can be improved by making the CLIP training dataset more abundant and reduce its domain gap with the targeted classification dataset.

On the other hand, the model error is further divided into two parts:

- (1) the distance between $Q_{g,h}(\mathbf{z}|\mathbf{x})$ and $Q'(\mathbf{z}|\mathbf{x})$ (*CLIP generalization error*);
- (2) the error induced by eq. (4) when estimating $\tilde{P}_{zeroshot}(y|\mathbf{x})$ based on $Q'(\mathbf{x}, \mathbf{z})$ (*Prompt sampling and designing*).

Intuitively, the first part represents how good our CLIP model learn and generalize, and the second part represents how much extra bias we introduce when we try to approximate the integral with sampling (eq. (4)).

CLIP generalization error. Following main paper’s notations, let us recall here the empirical risk on *i.i.d* sampled dataset from the noisy CLIP distribution Q as $\hat{R}^Q(f)$:

$$\hat{R}^Q(g, h) = \frac{1}{M} \sum_{i=1}^M L_{clip}(\mathbf{x}_i, \mathbf{z}_i; g, h),$$

and the corresponding empirical risk *w.r.t* clean dataset as:

$$\hat{R}^{Q'}(g, h) = \frac{1}{M} \sum_{i=1}^M L_{clip}(\mathbf{x}_i, \mathbf{z}'_i; g, h),$$

while the expected risk on the unknown clean CLIP distribution Q' as $R^{Q'}(g, h)$, as:

$$R^{Q'}(g, h) = E_{Q'} L_{clip}(\mathbf{x}, \mathbf{z}; g, h)$$

Below we present how to bound the CLIP generalization error. We denote $(\hat{g}, \hat{h}) = \arg \min_{g \in \mathcal{G}, h \in \mathcal{H}} \hat{R}^Q(g, h)$ as the empirical optimal model *w.r.t* *i.i.d* sampled dataset from Q , $(g^*, h^*) = \arg \min_{g \in \mathcal{G}, h \in \mathcal{H}} R^{Q'}(g, h)$ as the best-achievable model *w.r.t* clean distribution Q' and $(g_{bayes}, h_{bayes}) = \arg \min_{g, h} R^{Q'}(g, h)$ as the Bayes optimal model *w.r.t* clean distribution Q' . We can decompose the excess risk of our learned empirical optimal model \hat{f} over the Bayes optimal model f_{bayes} as:

$$\begin{aligned} R^{Q'}(\hat{g}, \hat{h}) - R^{Q'}(g_{bayes}, h_{bayes}) &= \underbrace{R^{Q'}(\hat{g}, \hat{h}) - R^{Q'}(g^*, h^*)}_{\text{estimation error}} \\ &\quad + \underbrace{R^{Q'}(g^*, h^*) - R^{Q'}(g_{bayes}, h_{bayes})}_{\text{approximation error}} \\ &= R^{Q'}(\hat{g}, \hat{h}) - R^{Q'}(g^*, h^*) + \mathcal{B}_{approx} \\ &\approx R^{Q'}(\hat{g}, \hat{h}) - R^{Q'}(g^*, h^*) \end{aligned} \tag{1}$$

¹The image description sometimes can be random due to the data collection process [3, 8]. We here also consider this into consideration. **Please note this is different with our interested label noise in this work.**

Exact analysis of approximation error is often intractable, we thus abbreviate it as \mathcal{B}_{approx} and omit it in subsequent analysis. For estimation error, we have:

$$\begin{aligned}
R^{Q'}(\hat{g}, \hat{h}) - R^{Q'}(g^*, h^*) &= R^{Q'}(\hat{g}, \hat{h}) - \hat{R}^Q(\hat{g}, \hat{h}) + \hat{R}^Q(\hat{g}, \hat{h}) \\
&\quad - \hat{R}^Q(g^*, h^*) + \hat{R}^Q(g^*, h^*) - R^{Q'}(g^*, h^*) \\
&\stackrel{\hat{R}^Q(\hat{g}, \hat{h}) - \hat{R}^Q(g^*, h^*) \leq 0}{\leq} \\
&\leq R^{Q'}(\hat{g}, \hat{h}) - \hat{R}^Q(\hat{g}, \hat{h}) + \hat{R}^Q(g^*, h^*) - R^{Q'}(g^*, h^*) \\
&\leq 2 \sup_{g \in \mathcal{G}, h \in \mathcal{H}} |R^{Q'}(g, h) - \hat{R}^Q(g, h)|
\end{aligned} \tag{2}$$

Supposing the range of L_{clip} as $[0, l_{\infty}^{clip}]$ for all (\mathbf{x}, \mathbf{z}) in $\text{sup}(Q)$ with $g, h \in \mathcal{G}, \mathcal{H}$ and L_{clip} is λ -Lipschitz continuous w.r.t \mathbf{z}_i , according to Lemma F.1 and triangle inequality, we have:

$$\begin{aligned}
|R^{Q'}(g, h) - \hat{R}^Q(g, h)| &\stackrel{\text{Lemma F.1}}{\leq} |R^{Q'}(g, h) - \hat{R}^{Q'}(g, h)| \stackrel{\text{Lipschitz continuous}}{+} |\hat{R}^{Q'}(g, h) - \hat{R}^Q(g, h)| \\
&\leq 2\mathfrak{R}(\mathcal{G} \circ \mathcal{H}) + l_{\infty}^{clip} \sqrt{\frac{\log(1/\delta)}{2M}} + \lambda \frac{1}{M} \sum_{i=1}^M \|\mathbf{z}_i - \mathbf{z}'_i\|_2 \\
&\leq 2\mathfrak{R}(\mathcal{G} \circ \mathcal{H}) + l_{\infty}^{clip} \sqrt{\frac{\log(1/\delta)}{2M}} + \varepsilon_n
\end{aligned} \tag{3}$$

Here, we rewrite $\lambda \frac{1}{M} \sum_{i=1}^M \|\mathbf{z}_i - \mathbf{z}'_i\|_2$ as ε_n which is the error term induced by language noise ($\mathbf{z}_i \neq \mathbf{z}'_i$). With eq. (1) and eq. (3), we have:

$$R^{Q'}(\hat{g}, \hat{h}) - R^{Q'}(g_{bayes}, h_{bayes}) \leq 2(2\mathfrak{R}(\mathcal{G} \circ \mathcal{H}) + l_{\infty}^{clip} \sqrt{\frac{\log(1/\delta)}{2M}} + \varepsilon_n) \tag{4}$$

To further connect the generalization error bound above and the distance of estimated probability $Q_{g,h}(\mathbf{z}|\mathbf{x})$ and $Q'(\mathbf{z}|\mathbf{x})$, we have:

$$\begin{aligned}
R^{Q'}(g, h) &= E_{Q' L_{clip}}(\mathbf{x}, \mathbf{z}; g, h) \\
&= -\frac{1}{2} \int Q'(\mathbf{x}) \int Q'(\mathbf{z}|\mathbf{x}) \log Q_{g,h}(\mathbf{z}|\mathbf{x}) d\mathbf{z} d\mathbf{x} \\
&\quad - \frac{1}{2} \int Q'(\mathbf{z}) \int Q'(\mathbf{x}|\mathbf{z}) \log Q_{g,h}(\mathbf{x}|\mathbf{z}) d\mathbf{x} d\mathbf{z} \\
&= \frac{1}{2} \int Q'(\mathbf{x}) D_{KL}(Q'(\mathbf{z}|\mathbf{x}), Q_{g,h}(\mathbf{z}|\mathbf{x})) d\mathbf{x} \\
&\quad - \frac{1}{2} \int Q'(\mathbf{x}) \int Q'(\mathbf{z}|\mathbf{x}) \log Q'(\mathbf{z}|\mathbf{x}) d\mathbf{z} d\mathbf{x} \\
&\quad - \frac{1}{2} \int Q'(\mathbf{z}) \int Q'(\mathbf{x}|\mathbf{z}) \log Q_{g,h}(\mathbf{x}|\mathbf{z}) d\mathbf{x} d\mathbf{z} \\
&\geq \frac{1}{2} \int Q'(\mathbf{x}) D_{KL}(Q'(\mathbf{z}|\mathbf{x}), Q_{g,h}(\mathbf{z}|\mathbf{x})) d\mathbf{x} \\
&\geq d(Q_{g,h}(\mathbf{z}|\mathbf{x}), Q'(\mathbf{z}|\mathbf{x}))
\end{aligned} \tag{5}$$

Specifically, we have $(g_{bayes}, h_{bayes}) = \arg \min R^Q(g, h)$ when and only when $Q_{g,h}(\mathbf{z}|\mathbf{x}) = Q'(\mathbf{z}|\mathbf{x})$. Intuitively, when and only when the learned model is Bayes optimal, we have a zero distance between the estimated probability and the

ground-truth probability. According to eq. (4), we thus have:

$$\begin{aligned}
 R^{Q'}(\hat{g}, \hat{h}) - R^{Q'}(g_{bayes}, h_{bayes}) &\leq 2(2\mathfrak{R}(\mathcal{G} \circ \mathcal{H}) + l_{\infty}^{clip} \sqrt{\frac{\log(1/\delta)}{2M}} + \varepsilon_n) \implies \\
 d(Q_{\hat{g}, \hat{h}}(z|\mathbf{x}), Q'(z|\mathbf{x})) &\leq R^{Q'}(g_{bayes}, h_{bayes}) \\
 &\quad + 2(2\mathfrak{R}(\mathcal{G} \circ \mathcal{H}) + l_{\infty}^{clip} \sqrt{\frac{\log(1/\delta)}{2M}} + \varepsilon_n) \\
 &\leq 2(2\mathfrak{R}(\mathcal{G} \circ \mathcal{H}) + l_{\infty}^{clip} \sqrt{\frac{\log(1/\delta)}{2M}} + \varepsilon_n)
 \end{aligned} \tag{6}$$

Prompt sampling and designing. We then take step two into consideration. According to eq. (3), with $Q_{\hat{g}, \hat{h}}(z|\mathbf{x})$ we can estimate $\tilde{P}_{zeroshot}(y|\mathbf{x})$. To quantify the additional error of the sampling process (eq. (4)), we denote as Δ a error coefficient which represents how much extra error been induced. Let us recall the domain gap (ε_{domain}) before, we thus have [THEOREM 1](#) below:

THEOREM F.2 (ESTIMATION WITH ZERO-SHOT CLASSIFIER). *Let \mathcal{G}, \mathcal{H} be the hypothesis space of vision encoder g and language encoder h . Let us denote the rademacher complexity as $\mathfrak{R}(\mathcal{G} \circ \mathcal{H})$ of the combined CLIP model. Supposing the range of L from eq. (2) as $[0, l_{\infty}^{clip}]$ for all (x, z) in $\text{sup}(Q)$ with $g, h \in \mathcal{G}, \mathcal{H}$. Then, for any $\delta > 0$, with probability at least $1 - \delta$ we have the following holds:*

$$d(\tilde{P}_{zeroshot}(y_i|\mathbf{x}_i), p^{true}(y_i|\mathbf{x}_i)) \leq \varepsilon_{domain} + \Delta(\lambda_1 \mathfrak{R}(\mathcal{G} \circ \mathcal{H}) + \lambda_2 l_{\infty}^{clip} \sqrt{\frac{\log 1/\delta}{M}} + \lambda_3 \varepsilon_n)$$

with $\lambda_1, \lambda_2, \lambda_3 > 0$. Here, ε_{domain} denotes the bias term induced by the domain gap between Q and p^{true} , and $\Delta \geq 1$ denotes the bias coefficient induced in designing prompts and sampling in ??.

F.2 Derivation of [THEOREM 2](#)

The derivation of [THEOREM 2](#) follows a similar but rather simpler process. Specifically, with Q, Q', z_i, z'_i, M replaced by P, P', y_i, y'_i, N , similar to eq. (4), we have:

$$R^{true}(\hat{f}) - R^{true}(f_{bayes}) \leq 2(2\mathfrak{R}(\mathcal{F}) + l_{\infty}^{noisy} \sqrt{\frac{\log(1/\delta)}{2N}} + \varepsilon_{noise}) \tag{7}$$

To similarly connect the generalization error bound above and the distance of estimated probability $P_f(y|\mathbf{x})$ and $p^{true}(y|\mathbf{x})$, with L_{noisy} as the cross-entropy loss, we have:

$$\begin{aligned}
 R^{true}(f) &= E_{p^{true}} L_{noisy}(\mathbf{x}, y; f) \\
 &= - \int p^{true}(\mathbf{x}) \int p^{true}(y|\mathbf{x}) \log P_f(y|\mathbf{x}) dy d\mathbf{x} \\
 &= \int p^{true}(\mathbf{x}) D_{KL}(p^{true}(y|\mathbf{x}), P_f(y|\mathbf{x})) d\mathbf{x} \\
 &\quad - \int p^{true}(\mathbf{x}) \int p^{true}(y|\mathbf{x}) \log p^{true}(y|\mathbf{x}) dy d\mathbf{x} \\
 &\geq \int p^{true}(\mathbf{x}) D_{KL}(p^{true}(y|\mathbf{x}), P_f(y|\mathbf{x})) d\mathbf{x} \\
 &\geq 2d(P_f(y|\mathbf{x}), p^{true}(y|\mathbf{x}))
 \end{aligned} \tag{8}$$

Similarly, we then have [THEOREM 2](#):

THEOREM F.3 (ESTIMATION WITH TRAINED CLASSIFIER). *Let \mathcal{F} be the hypothesis space of trained classifier f' . Let us denote the rademacher complexity as $\mathfrak{R}(\mathcal{F})$ of the trained classifier. Supposing the range of L for training f' as $[0, l_\infty^{\text{noisy}}]$ for all (x, y) in $\text{sup}(P)$ with $f' \in \mathcal{F}$. Then, for any $\delta > 0$, with probability at least $1 - \delta$ we have the following holds:*

$$d(\tilde{P}_{\text{trained}}(y_i|\mathbf{x}_i), P^{\text{true}}(y_i|\mathbf{x}_i)) \leq \varepsilon_{\text{noise}} + \lambda_1 \mathfrak{R}(\mathcal{F}) + \lambda_2 l_\infty^{\text{noisy}} \sqrt{\frac{\log 1/\delta}{N}}$$

with $\lambda_1, \lambda_2 > 0$. Here, $\varepsilon_{\text{noise}}$ denotes the difference term induced by the distribution difference between P and P^{true} .

REFERENCES

- [1] Eric Arazo, Diego Ortego, Paul Albert, Noel O'Connor, and Kevin McGuinness. 2019. Unsupervised label noise modeling and loss correction. In *International Conference on Machine Learning*. PMLR, 312–321.
- [2] Pengfei Chen, Junjie Ye, Guangyong Chen, Jingwei Zhao, and Pheng-Ann Heng. 2021. Beyond class-conditional assumption: A primary attempt to combat instance-dependent label noise. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11442–11450.
- [3] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*. PMLR, 4904–4916.
- [4] Junnan Li, Richard Socher, and Steven CH Hoi. 2020. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394* (2020).
- [5] Sachit Menon and Carl Vondrick. 2022. Visual Classification via Description from Large Language Models. *arXiv preprint arXiv:2210.07183* (2022).
- [6] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. 2018. *Foundations of machine learning*. MIT press.
- [7] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. 2017. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1944–1952.
- [8] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 8748–8763.
- [9] Xiaobo Xia, Bo Han, Nannan Wang, Jiankang Deng, Jiatong Li, Yinian Mao, and Tongliang Liu. 2022. Extended T: Learning With Mixed Closed-Set and Open-Set Noisy Labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 3 (2022), 3047–3058.
- [10] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. 2019. Are anchor points really indispensable in label-noise learning? *Advances in neural information processing systems* 32 (2019).
- [11] Kun Yi and Jianxin Wu. 2019. Probabilistic end-to-end noise correction for learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7017–7025.
- [12] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. 2019. How does disagreement help generalization against label corruption?. In *International Conference on Machine Learning*. PMLR, 7164–7173.