J. Vis. Commun. Image R. 111 (2025) 104550

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Full Length Article

Adaptive Vision-Language Prompt Learners for Learning with Noisy Labels ☆

Changhui Hu [a,1], Bhalaji Nagarajan [a,*,1], Ricardo Marques [b,2], Petia Radeva [a,c,2]

[a] Dept. de Matemàtiques i Informàtica, Universitat de Barcelona, Gran Via de les Corts Catalanes 585, Barcelona, 08007, Spain
[b] Grup de Recerca de Tecnologies Interactives (GTI), Universitat Pompeu Fabra (UPF), Barcelona, Spain
[c] Institut de Neurosciències, Universitat de Barcelona, Barcelona, Spain

ARTICLE INFO

MSC:
68T05
68T10
68T30
68T35
68T37
68W40

Keywords:
Learning with noisy labels
Visual-language models
Prompt learning

ABSTRACT

Training deep learning models requires manual labelling of a large volume of diverse data that is a tedious and time-consuming process. As humans are prone to errors, large-scale data labelling often introduces label noise, leading to degradation in the performance of deep neural networks. Recently, pre-trained models on extensive multi-modal data have shown remarkable performance in computer vision tasks. However, their use to tackle the problem of learning with noisy labels is still in its infancy, due to high computational complexity and training costs. In this work, we propose a novel approach, AVL-Prompter, to effectively leverage vision-language-pre-trained models for learning with noisy labels. The key idea of our method is the use of shared deep learnable prompts between visual and textual encoders, allowing us to effectively adapt large V-L models to the downstream task of learning with noisy labels. Our technique exhibits superior performance, particularly in high-noise settings, outperforming state-of-the-art methods in several datasets with synthetic and real label noise. Our contribution comes from a novel, simple, but highly efficient methodological path to learning with noisy labels while remaining straightforward to implement. The code is available at https://github.com/bhalajin/AVL-Prompter.

1. Introduction

Deep learning has exhibited remarkable performance on a wide range of complex and challenging computer vision tasks, including classification [1,2], object detection [3,4], and segmentation [5]. This success can be attributed to the powerful model architectures and the ability to learn from large-scale datasets. The performance of a deep learning model is largely influenced by the quality and quantity of training data it leverages [6]. However, the process of data collection and labelling data in real-world scenarios is prone to various uncertainties [7], which can result in the inclusion of incorrect labels *(label noise)* in the training data, thereby significantly impacting the performance of the model [8]. Effectively training models in the presence of label noise [9–12] has become an issue of great concern in the realm of deep learning. Noisy labels affect the generalization of the models [13,14].

Learning with Noisy Labels (LNL) has been long focussed in the machine learning community [15]. The primary objective is to develop models that are robust towards label noise. Recent LNL algorithms employ various strategies [16], including modifying the loss function [17,18], re-weighting samples [19], and using robust loss functions [20,21]. Sample selection methods, which leverage the idea of selecting clean samples for training models, are a promising approach that operates under the intuition that less noisy data contribute to more robust models [8]. However, recent models have shown a trend towards increased complexity, resulting in a substantial increase in training time [22]. Owing to this complexity, most recent methods opt for smaller models to mitigate computational overhead. In addition, recent studies reveal that the increase in algorithm complexity is not directly related to the increased performance improvements [23]. Additionally, a common approach in LNL algorithms is to initialize models with weights pretrained on ImageNet, which, however, limits the semantic information contained in the category labels [24].

The latest breakthroughs in self-supervised learning have facilitated the pre-training of large and powerful models on general large-scale corpora. These models exhibit increased generalization capabilities, allowing for fine-tuning on diverse downstream tasks. Pre-trained language models such as BERT [25] and GPT-3 [26] have inspired the development of potent computer vision models such as the vision

☆ This paper has been recommended for acceptance by Zicheng Liu.

* Corresponding author at: Dept. de Matemàtiques i Informàtica, Universitat de Barcelona, Gran Via de les Corts Catalanes 585, Barcelona, 08007, Spain.
E-mail addresses: chhuhuxx121@alumnes.ub.edu (C. Hu), bhalaji.nagarajan@ub.edu (B. Nagarajan), ricardo.marques@upf.edu (R. Marques), petia.ivanova@ub.edu (P. Radeva).

[1] Equal contribution.
[2] Equal supervision.

https://doi.org/10.1016/j.jvcir.2025.104550
Received 28 November 2024; Received in revised form 21 July 2025; Accepted 28 July 2025
Available online 5 August 2025

transformers [27] and ConvNext [28]. Large-scale V-L pretraining models, such as CLIP [29] and ALIGN [30], learn a shared embedding space by aligning vision and language modalities [31]. Contrastive Language-Image Pretraining (CLIP) [29], for instance, uses contrastive learning to extract representations from noisy vision-language pairs on the web. CLIP models excel in zero-shot recognition, leveraging the rich supervision provided by natural language. For an image classification downstream task, the prediction for an image is computed as the output with the highest similarity score between language and text representations. In V-L models, two separate encoders — one for each modality — are used to transform the high-dimensional data into low-dimensional embedding space. V-L models provide a joint embedding space between the text and image modalities and offer an improved classification space, where classes have well-defined boundary space [32]. This is particularly helpful in noisy labels, as the improved space can better handle data through modality alignment, compared to having only one single modality. This reducing overfitting due to label noise.

One of the important issues that arises with the advent of large V-L models is to investigate effective ways to adapt these models for various downstream tasks. Prompt learning [33] is a recent alternative to fine-tuning models, where learnable prompt vectors are optimized using a task-specific objective to tailor large CLIP-like models for downstream tasks [34]. Initial implementation of prompt learning in V-L models used prompts only on the text encoder [34]. Context Optimization [34] involves modelling the context words of the prompts using learnable vectors, while keeping the entire pre-trained parameters fixed. Conversely, Conditional Context Optimization [35] utilizes a neural network to generate an input-conditional token for each image. Recent prompt learning methods such as MaPLe [36] use shared prompts for both the vision and language encoders. Prompt learning research has shown rapid progress due to their wide applicability and data-efficiency [32]. However, very few works have been devoted to their label noise robustness [37,38], especially in downstream image classification tasks.

Our proposal. With recent advancements in multi-modal pre-training models and prompt learning techniques for adapting their representations, a myriad of potential solutions emerge for addressing the issue of label noise and overcoming the bottlenecks inherent in the existing LNL algorithms. There has been very limited research in leveraging pre-trained models for noisy label datasets in downstream classification tasks. With this regard, we propose, **Adaptive Vision-Language Prompt Learners**, in short, AVL-Prompter to address the problem of label noise. AVL-Prompter uses a vision-language prompt learning model with shared deep learnable prompts between the visual and textual encoders. These shared deep learnable prompts ensure both the vision and language encoders mutually interact to learn the context prompts simultaneously. This interaction is not present in V-L models in general, as both modalities work separately towards context optimization [36]. This V-L model is embedded in a pipeline comprising of semi-supervised learning methodology to enable models to learn in the presence of label noise. To the best of our knowledge, AVL-Prompter is the first work to utilize multi-modal prompt learning to tackle LNL problems and, as shown by our results, leads to a very significant improvement compared to the SoTA. Although highly innovative in the sense that it proposes a new methodological approach to LNL, our method is relatively simple to implement. On this front, our main contributions to this work can be outlined as follows:

- We propose **AVL-Prompter**, a novel vision-language prompt learning model for learning with noisy labels with shared deep learnable prompts between the vision and text encoders. We fine-tune the vision and text encoders along with learnable shared prompts suitable for handling label noise.
- We use a simple and efficient semi-supervised learning technique encompassing our AVL-Prompter to achieve a highly effective small-loss-based sample-selection LNL framework.
- We evaluate the efficacy of AVL-Prompter using different synthetic and real-noise benchmarks across various noise ratios. We obtain performance gains across various benchmark datasets, particularly in challenging high noise settings and in real-noise benchmarks. We perform an extensive analysis of the results and compare them to several state-of-the-art methods.

2. Related works

In this section, we provide an overview of the latest literature that is highly relevant to our work.

2.1. Learning with noisy labels

In recent years, LNL has been a focal point of research interest, leading to several deep-learning algorithms aimed at addressing the problem of label noise [16]. LNL algorithms can be categorized into various families of methods based on their operational models. An overview of different families of methods are detailed in [16].

2.1.1. Taxonomy of LNL methods

Sample selection methods aim to identify clean samples through various strategies, such as utilizing small loss [39]. Loss correction involves adjusting the loss weights to prevent overfitting on noisy samples, thereby reducing errors caused by incorrectly labelled samples [17,40]. The transition matrix is a common technique used in loss correction methods [40]. In terms of sample reweighting, the samples that are noisiest are identified and weighted differently compared to the clean samples [41]. Regularization-based methods are widely utilized due to their ability to mitigate the memorization of labels and enhance the robustness of the algorithms [42–45]. Meta-learning [46] strategies are used in identifying and correcting potential noisy labels. Noise-robust loss functions such as Mean Absolute Error [47], Symmetric cross-entropy learning [20], Active Passive Loss [21] increase the robustness of LNL algorithms.

2.1.2. Sample selection methods

Sample selection techniques have gained significant popularity within the LNL community. These methods select a possible clean subset of data and provide different training strategies for the clean and noisy subsets. Identifying the criteria used in selecting the samples is a challenging task. Two-component beta-mixture model [48] and Gaussian mixture model [49] are used in selecting clean and noisy samples. Sample selection-based methods often suffer from error accumulation.

2.1.3. Multi-network methods

Multi-network learning frameworks [39,49–51] used two cooperative 'peer' networks to reduce the flow of error introduced by the noisy labels. DivideMix [49], one of the most popular LNL benchmark methods, used two homogeneous networks to train each other by employing GMMs to split the training data into clean and noisy samples. Contrast2Divide [52] studied the importance of warm-up in LNL algorithms and proposed the use of self-supervised pre-training to create better feature extractors. AugDesc [53] explored different data augmentation techniques in LNL algorithms and used two different augmentations — one for analysing the loss and the other for backpropagation. DISC [54] used a dynamic threshold strategy to select clean and noisy samples along with selecting hard samples. RankMatch [55] employs confidence and consistency to combat label noise. Bayesian DivideMix++ [56] uses a combination of techniques to address the challenge of memorization and label uncertainty. CCLM [57] used a per-class-based local distribution of samples to address the challenges in Global Noise modelling in other DivideMix-inspired works. Manifold

DivideMix [58] used a filtering procedure to remove out-of-distribution samples before the semi-supervised learning phase. Probabilistic Noise Perception [59], ProMix [60], ULC [61], and SplitNet [62] are recent sample selection methods that achieved state-or-the-art performance.

2.2. Vision-language models

V-L models have been able to encode multi-modal representations learned in a shared embedding space. CLIP [29], ALIGN [30], and Florence [63] have been effective in a wide spectrum of downstream tasks. With large V-L pre-trained models available, several research focuses on the effective adaptation of these models to the downstream tasks. "Prompting" [33] is an alternative to fine-tuning the pre-trained models, where prompts are used to tailor the models for downstream tasks.

2.2.1. Prompt tuning

Prompt Tuning [32] is a technique where the model uses a small target dataset to learn the prompts using back-propagation. Prompt Aligned Gradient [64] used selective gradient updates to prevent prompt tuning from forgetting the knowledge learned from the V-L models. Read-only Prompt Optimization [65] used masked attention to prevent internal representation shift in the pre-trained model.

2.2.2. Prompt learning

Prompt Learning is another class of algorithms, where the prompts are learned automatically during the fine-tuning stage. CoOp [34] models the context works of the prompts, keeping the pre-trained parameters fixed. CoCoOp [35] uses an additional network to generate input-conditional tokens for each image. MaPLe [36] utilizes shared learnable prompts across vision and language encoders. PromptSRC [31] optimizes the prompts for both task-specific and task-agnostic general representations.

2.2.3. Prompt learning in LNL

Although several strategies exist to adapt the pre-trained models for downstream tasks, there has been very little work on utilizing them for LNL methods. NLIP [66] used noise-harmonization and noise-completion to mitigate the impact of noise (wrong or irrelevant content). Prompt Tuning [32] for CLIP has been found robust to label noise. TURN [22] is based on fine-tuning of pre-trained models to learn target datasets in the presence of label noise. EPL [23] used linear probing strategies to improve the robustness of LNL methods. The above LNL methods have instigated a paradigm shift in using large pre-trained models through different adaptations. In this work, we present, AVL-Prompter, a simple and novel approach to learning with noisy labels, using prompt learning techniques to adapt vision-language pre-trained models for LNL problems.

3. AVL-prompter for learning with noisy labels

In this section, we introduce our proposal, **AVL-Prompter** for learning with noisy labels. The comprehensive overview of AVL-Prompter is illustrated in Fig. 1. We follow the widely adapted **multi-network learning framework** for sample-selection-based LNL problems [49,52,56] and simultaneously train two homogeneous models. In contrast to the state-of-the-art methodologies, which typically rely on convolutional neural networks, we use a CLIP-like vision-language prompt learning model as the model architecture [36]. This choice enables us to leverage the rich supervision offered by natural language, enhancing the model's performance. The proposed AVL-Prompter falls into the category of small-loss-based sample-selection methods. During each epoch, we model the sample loss of the models to select the clean and noisy samples. At each mini-batch, the models use both the clean (labelled) and noisy (unlabelled) data identified by the other model to perform semi-supervised learning. We detail each component in the subsequent sections.

3.1. Vision-language prompt learning architecture

3.1.1. Vision-language model

CLIP-like [29] models consists of an image encoder ($\mathcal{V}$) and a text encoder ($\mathcal{L}$) of $\mathcal{K}_v$ and $\mathcal{K}_l$ transformer layers, respectively. The image encoder, $\mathcal{V}$, maps high-dimensional image representations into a low-dimensional embedding space, whereas the text encoder, $\mathcal{L}$, outputs text embeddings from natural language. An overview of the V-L prompt learning architecture can be seen in Fig. 2.

Image encoder. Each image $\mathcal{X}$ is divided into $\mathcal{M}$ fixed-size patches, followed by projection to produce patch tokens $\{e_1, e_2, \dots, e_{\mathcal{M}}\}$, $e_i \in \mathbb{R}^{\mathcal{M} \times d_v}$, d_v is the dimension of the output of the last transformer layer. A learnable class token, cls, is appended with the patch tokens constituting the input tokens $\hat{\mathcal{X}} = \{cls, e_1, e_2, \dots, e_{\mathcal{M}}\}$. The tokens in $\hat{\mathcal{X}}$ are sequentially processed through the $\mathcal{K}_v$ transformer blocks of the image encoder $\mathcal{V}$ to produce a latent visual feature representation, $x \in \mathbb{R}^{d_{vl}}$, projected from a common $\mathcal{V}-\mathcal{L}$ latent space.

Text encoder. The text encoder, $\mathcal{L}$, generates latent text feature representations, z, from the text descriptions. The text descriptions are produced from the class label, $y \in \{1, 2, \dots, C\}$ enclosed within a text template *"a photo of a {class label}"*. The text descriptions are tokenized and projected into word embeddings, $\hat{\mathcal{W}} = \{w_i\} \in \mathbb{R}^{\mathcal{N}_e \times d_l}$, $\mathcal{N}_e$ being the number of embeddings and d_l being the dimension of the output of the last transformer layer. The $\mathcal{K}_l$ transformer blocks of the text encoder $\mathcal{L}$, in their turn, output the latent textual feature representation $z \in \mathbb{R}^{d_{vl}}$, from the common $\mathcal{V}-\mathcal{L}$ latent space.

Prediction probability. For the image classification task, the prediction $\hat{y}$ for image $\mathcal{X}$ is computed as the output with the highest similarity score. The prediction probability is computed as:

$$p(\hat{y}|x) = \frac{\exp(\text{sim}(x, z_{\hat{y}})/\mathcal{T}_{CLIP})}{\sum_{c=1}^{C} \exp(\text{sim}(x, z_c)/\mathcal{T}_{CLIP})}, \tag{1}$$

where x is the latent visual feature representation of $\mathcal{X}$ generated by the image encoder $\mathcal{V}$, z_c is the latent text feature representation of the caption *"a photo of a {class label c}"*, generated by the text encoder $\mathcal{L}$, $\mathcal{T}_{CLIP}$ is the temperature parameter learned by CLIP and $\text{sim}(\cdot)$ is the cosine similarity score.

3.1.2. Prompt learning

Prompt Learning [67] is an alternate technique for fine-tuning the large V-L models. Prompt learning approaches can be uni-modal [34,35], where learnable prompt tokens are appended to the text encoders or multi-model [31,36], where the prompts are appended to both image and text encoders. We use a *branch-aware multi-modal deep prompting* technique, where the vision and language branches are fine-tuned together using shared prompts. The deep prompting [36] allows the models to learn separate sets of prompts at every transformer block, rather than introducing the prompts only at the first transformer block as in the case of shallow prompts [34,35], thereby offering increased flexibility in aligning the representations of vision and language.

Learnable prompts. To learn context prompts, we append learnable prompts in the language branch and the vision branch of the V-L model. At each transformer block of the language model, a set of $\mathcal{N}$ learnable tokens $\mathcal{P}^{(k)} = \{P_i^{(k)}\}_{i=1}^{\mathcal{N}}$ is introduced up to a specific depth D_T, with k being the index of the transformer linear layer. After each D_T^{th} transformer layer, the subsequent layers process previous block prompts. The text encoder $\mathcal{L}$ processes the modified word embeddings:

$$\hat{\mathcal{W}}_P^{(k)} = \{P_1^{(k)}, P_2^{(k)}, \dots, P_{\mathcal{N}}^{(k)}, w_1, w_2, \dots, w_{\mathcal{N}}\}$$

to generate the prompt-enhanced latent textual feature representation, z_P. To learn the prompts in a shared embedding space, we use the $\mathcal{V}-\mathcal{L}$ coupling function, $\mathcal{F}(\cdot)$, where $\tilde{\mathcal{P}}^{(k)} = \mathcal{F}(\mathcal{P}^{(k)})$ [36]. The coupling

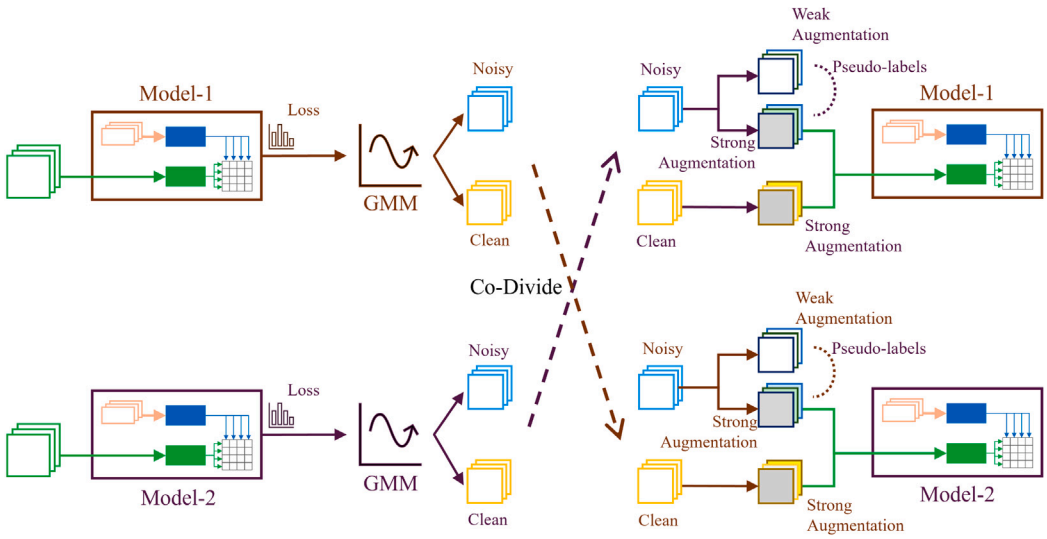


Fig. 1. Overview of our AVL-Prompter. Loss from two homogeneous peer networks (Model-1 and Model-2) are modelled using a GMM to split the training data into clean and noisy subsets. The subsets identified by the models are used to perform semi-supervised learning, treating the clean set as labelled and the noisy set as unlabelled.

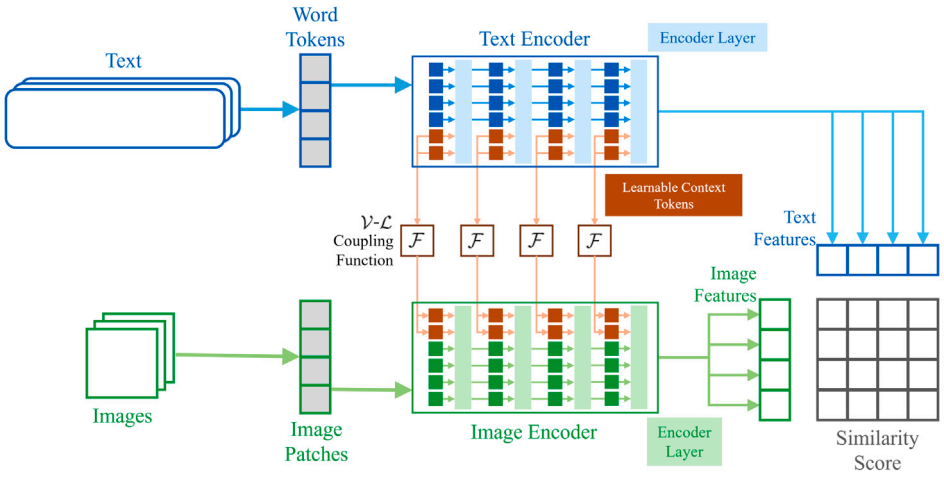


Fig. 2. Vision-Language Prompt Learner Architecture. The model comprises a text encoder and an image encoder that uses the class labels and images to create text and image features. Additionally, the model uses shared learnable context tokens realized using the coupling function $\mathcal{F}(\cdot)$, which enhances the latent features of both encoders.

function $\mathcal{F}(\cdot)$ is a linear layer that maps d_l dimensional inputs into d_v. The image encoder $\mathcal{V}$ employs the modified input token,

$$\hat{\mathcal{X}}_P = \{\tilde{P}_1^{(k)}, \tilde{P}_2^{(k)}, \ldots, \tilde{P}_{\mathcal{M}}^{(k)}, cls, e_1, e_2, \ldots, e_{\mathcal{M}}\},$$

resulting from applying the $\mathcal{V}-\mathcal{L}$ coupling function to the learnable prompts $\mathcal{P}^{(k)}$, to generate the prompt-enhanced latent visual feature representation, x_P. The prediction probability is computed as in Eq. (1) using the latent representations from the modified input tokens. Shared prompts allow the V-L model to learn more correlated features to successive transformer blocks. In contrast to the existing prompt learning technique [34,36], where the learned parameters of the V-L models are kept intact, we fine-tune the $\mathcal{V}-\mathcal{L}$ encoders along with learning the context prompts. This is done to allow the V-L model to better adapt to the downstream task of learning with noisy labels.

3.2. Training pipeline

Next, we provide a concise overview of the key components of the training pipeline: the loss modelling step and the semi-supervised learning step. Our proposed AVL-Prompter framework is based on the hypothesis that clean samples are faster to learn compared to noisy samples [8]. This hypothesis is supported by several LNL sample-selection methods [49,52,56], where clean samples are selected based on small losses. One critical design consideration revolves around using the model to make predictions (and selection of samples) on the same noisy data that it was trained on, which could lead to confirmation bias [68]. To mitigate this bias, following several benchmark LNL methods [49,52,55,56,69], we adopt training two homogeneous models simultaneously. The fundamental concept underlying this approach is to leverage the prediction (sample selections) of one model to derive the decisions of the other model, thereby promoting more robust learning.

3.2.1. Loss modelling step

For a dataset D containing $\mathcal{N}$ samples with possible label noise, $D = \{(x_i, \tilde{y}_i)\}_{i=1}^{\mathcal{N}}$, where x_i is the ith sample and $\tilde{y}_i$ is the one-hot label over C classes. We initialize two homogeneous V-L models with image and text encoder parameters, $\theta_{\mathcal{V}}$ and $\theta_{\mathcal{L}}$, respectively. The primary objective at this step is to fit the loss $\mathcal{L}(\theta_{\mathcal{V}}, \theta_{\mathcal{L}})$, to distinguish between clean and noisy samples. To achieve this, we employ a two-component Gaussian Mixture Model (GMM), g, to fit the loss distribution using the Expectation-Maximization algorithm. The resulting probability density function, $p(\cdot)$ gives the posterior probability of x_i being clean given its

loss, $l_i(\theta_{\mathcal{V}}, \theta_{\mathcal{L}})$. Using a threshold τ over $p(\cdot)$, we partition the samples in the training set into clean and noisy subsets. Next, we employ the co-divide step [49,52,56], where one model divides the dataset into clean and noisy samples, which are to be used by the other model. The clean split is treated as the labelled set, $\mathcal{X}$, whereas the noisy samples are treated as unlabelled set, $\mathcal{U}$. Pseudo-labels are assigned to the unlabelled data. Formally this step is represented as below:

$$\begin{aligned} \bigcup_{i \in \mathcal{N}} \mathcal{X} &= \left\{ (x_i, y_i) \mid p(g|l_i) \geq \tau \right\}_{(x_i, y_i) \in \mathcal{D}} \\ \bigcup_{i \in \mathcal{N}} \mathcal{U} &= \left\{ (x_i, y_i) \mid p(g|l_i) < \tau \right\}_{(x_i, y_i) \in \mathcal{D}} \end{aligned} \tag{2}$$

3.2.2. Semi-supervised learning step

The next phase involves using the labelled ($\mathcal{X}$) and unlabelled ($\mathcal{U}$) subsets to foster the learning of the models. At each training epoch, we train the two V-L models one at a time, keeping the other one fixed. We follow the FixMatch [70] semi-supervised learning method to obtain pseudo-labels for the unlabelled data.

Pseudo-labelling and consistency regularization. FixMatch works on the premise that model's predictions should remain consistent when presented with perturbed versions of the same image [71–73]. To achieve pseudo-labels, FixMatch leverages two kinds of augmentations: "weak" and "strong". Pseudo-labels are derived from the "weak" augmentation samples of the unlabelled set. The largest class probability is assigned as the pseudo-labels for each sample in the unlabelled set. Subsequently, these pseudo-labels are used as the targets for the "strong" augmentation of the unlabelled set, which is incorporated into the loss computation. In AVL-Prompter, we adopt a similar pseudo-labelling strategy. For the "weak" augmentations, we use standard flip-and-shift augmentations [70]. For the "strong" augmentations, we additionally integrate RandAugment [74]. RandAugment randomly selects transformations for each sample in the mini-batch, further enhancing the diversity of the augmentations applied. We use the same set of image transformations as those used in RandAugment [74].

3.2.3. Loss function

The overall loss function of AVL-Prompter is similar to the loss function of DivideMix [49], however with significant differences in the usage of clean and noisy subsets. The loss function is composed of (i) the loss associated with the labelled set $\mathcal{L}_x$, (ii) the loss corresponding to the unlabelled set $\mathcal{L}_u$, and, (iii) a regularization term $\mathcal{L}_r$:

$$\mathcal{L}(\theta_{\mathcal{V}}, \theta_{\mathcal{L}}) = \mathcal{L}_x + \lambda_u \mathcal{L}_u + \lambda_r \mathcal{L}_r. \tag{3}$$

L_x corresponds to the standard cross-entropy loss. In contrast to DivideMix, we use only "weak" augmentations on the samples of the labelled set ($\mathcal{X}$). Weak augmentations preserve the image-label alignment, making it easier for the model to learn accurate class boundaries. L_u also corresponds to the standard cross-entropy loss. However, in this case, using "strong" augmentations on the samples of the unlabelled set ($\mathcal{X}$), where the targets are derived from the "weak" augmentations of the samples in the unlabelled set ($\mathcal{U}$). This introduces the consistency regularization in our framework, which is based on the assumption that the model outputs similar predictions for perturbed versions of the same image [71]. λ_u corresponds to a fixed scalar hyper-parameter to denote the weight of the unlabelled loss. Cross-entropy loss of both L_x and L_u is computed using the prediction probability of the V-L model (Eq. (1)). L_r is the regularization term (same as in DivideMix [49]), used to regularize the model's average output across all samples in the mini-batch and λ_r corresponds to the weight of the regularization term. L_r follows a uniform prior distribution to regularize the model's average output across all samples of the mini-batch, preventing the model from assigning all samples to a single class, particularly in high-noise settings.

4. Experiments

First, we detail the experimental setup to validate our proposed AVL-Prompter method. We evaluate the performance of different synthetic and real noise datasets and compare them against various state-of-the-art methods to highlight the effectiveness of our approach. Finally, we provide a detailed analysis of the results to show the significance of our method, followed by the different design decisions used in establishing AVL-Prompter. The ablations show the importance of each component of our proposed method.

4.1. Experimental setup

Datasets. We evaluate the performance of AVL-Prompter across different datasets of varying characteristics: CIFAR-10 and CIFAR-100 [75], EuroSAT [76], Tiny-ImageNet [77], Oxford-IIIT Pets [78], CIFAR-10N and CIFAR-100N [79], and WebVision [80]. CIFAR-10/100 and CIFAR-10N/100N comprise 50k training samples and 10k testing samples, each of size 32×32. EuroSAT is a satellite image dataset comprising 27000 labelled samples distributed across 10 classes. Tiny-ImageNet is a reduced version of the ILSVRC12 ImageNet [81], featuring 200 classes, each with 500 images of size 64×64. Oxford-IIIT Pets has 37 categories, each class with around 200 images. WebVision 1.0 (mini WebVision) contains 2.4 million images sourced from 1000 ILSVRC12 ImageNet [81] classes. Of the datasets used for validation, CIFAR-10N/100N and WebVision are real noise datasets, while the rest are synthetic noise datasets.

Noise settings. For the synthetic noise datasets, we simulate two forms of label noise - *symmetric* and *asymmetric*. Symmetric noise is introduced by randomly substituting the labels of a certain percentage of samples with arbitrary labels drawn from all possible classes. Asymmetric noise involves selectively replacing labels of similar classes. For CIFAR-10, the labels are flipped as Truck → Automobile, Bird → Airplace, Deer → Horse, Dog ↔ Cat. For CIFAR-100, the classes are grouped into 20 super-classes of five (e.g. Aquatic Mammals contain Beaver, Dolphin, Otter, Seal and Whale), and the noise flips each class into the next circularly. For the other synthetic datasets (Tiny ImageNet, EuroSAT, Oxford-IIIT Pets), we use pair flip noise introduced by UNICON [69]. Following the established experiment settings for CIFAR datasets in LNL problems [49,52,56], we evaluate AVL-Prompter against varying noise ratios: 20%, 50%, 80%, and 90% symmetric noise, as well as 40% asymmetric noise. For Tiny-ImageNet, we validate on Symmetric 20% and 50% noise settings following UNICON [69], whereas for EuroSAT and Oxford-IIIT Pets, we use Symmetric 60% and Asymmetric 40% following EPL [23]. All the real noise datasets provide standard clean evaluation sets. CIFAR-10N/100N use the same training data as the corresponding CIFAR datasets, however, are provided with human-annotated real-world noisy labels. CIFAR-10N has five noisy label sets — Aggregate, Random1, Random2, Random3 and Worst, whereas CIFAR-100N has fine and coarse labels. WebVision is the largest real noise dataset and the label noise of WebVision is estimated to be around 20% [16].

Implementation details. To establish the Vision Language model in AVL-Prompter, we use ViT-B/32-CLIP [29] for all datasets except WebVision, for which we use ViT-L/14-CLIP [29]. We use a stochastic gradient descent (SGD) optimizer for all datasets except EuroSAT and Oxford-IIIT Pets, for which we use a Resilient Backpropagation (Rprop) optimizer. We parameterize SGD with an initial learning rate of 0.02, a momentum of 0.9, and a weight decay of 5e-4. For CIFAR, EuroSAT and Oxford-IIIT Pets, we set a batch size of 64 and train the model for a total of 60 epochs. For Tiny-ImageNet, we set the batch size as 128 and train the model for 25 epochs, whereas for WebVision, we use a batch size of 8 and train the model for 10 epochs. For the synthetic noise experiments, we set the clean probability threshold (τ) as 0.5, except for symmetric 90% noise, where τ is set to 0.6. For WebVision, we set

Table 1
Comparison with SoTA methods on CIFAR-10 dataset under symmetric and asymmetric noise. †-*Results updated from UNICON [69].* ‡-*Results updated from RankMatch [55].*

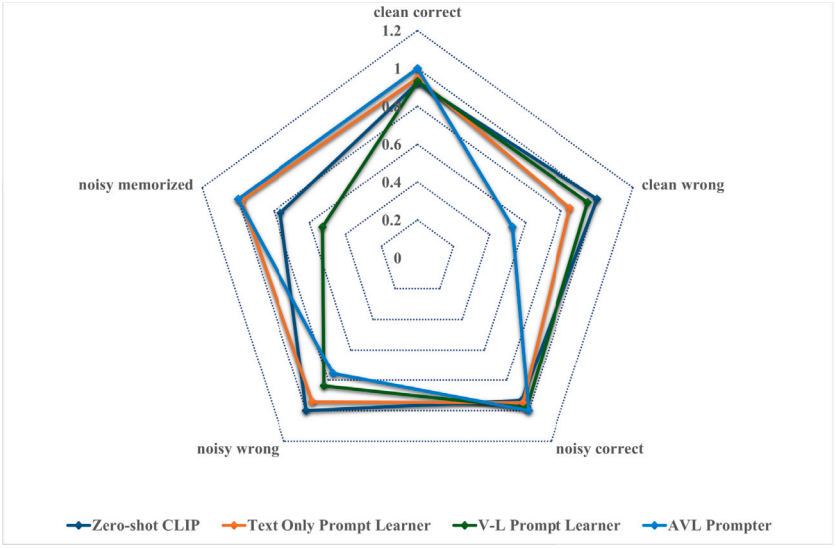
Noise Type		Sym.				Asym.
Method / Noise Ratio		20%	50%	80%	90%	40%
DivideMix [49]	Best	96.1	94.6	93.2	76.0	93.4
	Last	95.7	94.4	92.9	75.4	92.1
C2D [52]	Best	96.43±0.07	95.32±0.12	94.40±0.04	93.57±0.09	93.45±0.07
	Last	96.23±0.09	95.15±0.16	94.30±0.12	93.42±0.09	90.75±0.16
MOIT† [11]	Best	94.1	91.1	75.8	70.1	93.2
RRL‡ [82]	Best	96.4	95.3	93.3	77.4	93.3
UNICON [69]	Best	96.0	95.6	93.9	90.8	94.1
ULC [61]	Best	96.1	95.2	94.0	86.4	-
	Last	95.9	94.7	93.2	85.8	-
Rank Match [55]	Best	96.5	95.6	94.5	92.6	94.7
	Last	96.4	95.4	94.2	92.1	94.4
Bayesian DivideMix++ [56]	Best	96.39±0.06	95.68±0.09	95.25±0.08	94.46±0.15	-
	Last	96.13±0.07	95.40±0.11	94.97±0.02	94.20±0.12	-
SplitNet [62]	Best	96.5	96.3	95.2	94.0	95.4
	Last	96.3	96.0	95.0	93.9	95.3
AVL-Prompter	Best	**98.1**	**97.5**	**96.3**	**92.9**	**96.9**
	Last	**98.0**	**97.4**	**96.2**	**92.9**	**96.6**

Table 2
Comparison with SoTA methods on CIFAR-100 dataset under symmetric and asymmetric noise. †-*Results updated from UNICON [69].* ‡-*Results updated from RankMatch [55].* ⋆-*Results updated from C2D [52].*

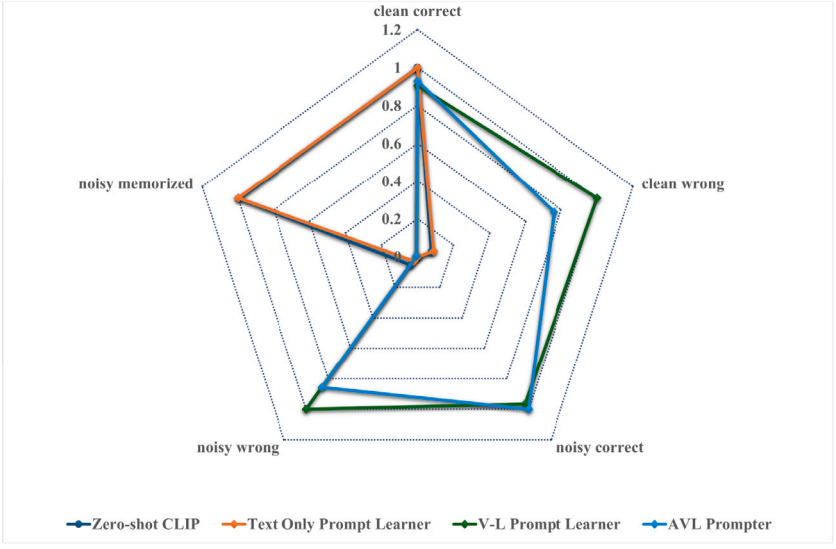
Noise Type		Sym.				Asym.
Method / Noise Ratio		20%	50%	80%	90%	40%
DivideMix [49]	Best	77.3	74.6	60.2	31.5	72.2⋆
	Last	76.9	74.2	59.6	31.0	72.4⋆
C2D [52]	Best	78.69±0.17	76.43±0.25	67.78±0.30	58.70±0.31	75.48±0.16
	Last	78.32±0.35	76.07±0.41	67.43±0.30	58.45±0.30	75.06±0.16
MOIT † [11]	Best	75.9	70.1	51.4	24.5	74.0
RRL ‡ [82]	Best	80.3	76.0	61.1	33.1	-
UNICON [69]	Best	77.6	63.9	44.8	74.8	-
ULC [61]	Best	77.3	74.9	61.2	34.5	-
	Last	77.1	74.3	60.8	34.1	-
Rank Match [55]	Best	79.5	77.9	67.6	50.6	-
	Last	79.3	77.6	67.2	49.9	-
Bayesian DivideMix++ [56]	Best	80.02±0.03	78.31±0.14	70.01±0.23	61.15±0.34	76.52±0.12
	Last	79.56±0.13	77.71±0.13	69.55±0.22	60.70±0.42	76.06±0.13
SplitNet [62]	Best	80.6	77.8	70.3	50.7	-
	Last	80.3	77.5	70.2	50.4	-
AVL-Prompter	Best	**87.3**	**86.3**	**83.0**	**78.6**	**85.8**
	Last	**87.3**	**86.2**	**82.8**	**78.1**	**85.7**

the value of τ as 0.5. We set λ_u and λ_r as used in DivideMix [49]. For the V-L Prompt Learner, following [36], we set the number of context vectors to 2 and prompt depth to 9. We initialize the learnable prompts with the phrase "a photo of a {class}". To measure the similarity scores between the text features and image features, we use a logit scale of 100 for the image features. For the "Text only prompt learner" model (used in ablations), we set the number of context vectors to 16 and we use the 'END' token position. We do not use any initialization words. We use the default CLIP logit scale. For the "Zero-shot CLIP" model (used in ablations), we use the text prompt template as *"a photo of a {}."*. All experiments in this work are implemented using Pytorch, on an NVIDIA RTX 3090 GPU. However, experiments related to smaller datasets like CIFAR can also be run on less powerful GPUs, such as NVIDIA RTX 2080Ti.

Comparison methods. We benchmark the performance of the proposed AVL-Prompter against several state-of-the-art LNL models, particularly focusing on sample-selection methods that leverage multi-network learning and include semi-supervised learning techniques. DivideMix [49] utilizes a semi-supervised learning framework, that employs per-sample loss distribution using GMMs. Contrast to Divide [52] addresses warm-up obstacles using self-supervised pre-training. MOIT [11] adopts joint training via supervised contrastive learning and semi-supervised classification. RRL [82] uses low-dimensional subspace to mitigate label noise. UNICON [69] performs sample selection using Jensen–Shannon diverge. ULC [61] leverages uncertainty measurements for label correction. LongReMix [83] uses oversampling of high-confidence clean set. RankMatch [55] utilizes confidence representation voting for sample selection. Bayesian DivideMix++ [56] enhances DivideMix through self-supervised pre-training, tailored augmentations and uncertainty-aware sample selection. CCLM [57] uses local noise modelling instead of Global noise modelling. PES [45] proposes a progressive early stopping method to alleviate the memorization effect of DNN. SplitNet [62] used a learnable module for clean-noise label splitting. EPL [23] utilizes pre-trained models to correct noisy samples. The comparisons enable us to ascertain the effectiveness of AVL-Prompter and its efficiency in handling label noise. *It is important to note that there is a lack of literature on using large vision-language models for LNL, underscoring the novelty of our approach. We provide comparisons with vision-language models wherever applicable.*



(a) Clean and noisy samples identified by the best models



(b) Clean and noisy samples identified by the final models

Fig. 3. Categorizing clean and noisy samples. *Each axis is normalized (0–1) for better visualization.*

4.2. Results

We brief the performance metrics used to validate AVL-Prompter against state-of-the-art methods. Later, we highlight the performance of our method across multiple datasets, providing detailed insights and discussing notable improvements.

Performance metrics. We follow the established experimental settings consistent with the different comparison methods. For all synthetic noise experiments (CIFAR10/100, EuroSAT, Oxford-IIIT Pets and Tiny-ImageNet), we report the "best" and the "last" accuracy concerning the test set. The "best" corresponds to the best test accuracy over all the epochs. The "last" accuracy corresponds to the average of the last ten epochs and is used as a measure of robustness [56,84,85]. Larger gaps between the "best" and "last" accuracy indicate potential overfitting to the noise model. For CIFAR10N/100N, we show the test accuracy of the given clean test set. For WebVision, we report the top-1 and top-5 validation accuracy of both WebVision and ILSVRC12 validation sets [49,52,56].

Table 3
Comparison on Tiny-ImageNet *(showing Best/Last Test accuracy)* [86]. †-*Results updated from UNICON [69].*

Noise Ratio	Std. CE †	NCT [86]	UNICON [69]	AVL-Prompter
Sym. 20%	35.8/35.6	58.0/57.2	59.2/58.4	**77.3** / **76.9**
Sym. 50%	19.8/19.6	47.8/47.4	52.7/52.4	**69.0** / **67.5**

Performance on synthetic noise. We report the performance of our approach on CIFAR-10 and CIFAR-100 datasets across various noise ratios in Tables 1 and 2, respectively. AVL-Prompter consistently outperforms existing state-of-the-art methods by a significant margin in all noise settings on both CIFAR-10 and CIFAR-100. It is noteworthy that the proposed AVL-Prompter employs a simple LNL strategy compared to the complex methodologies presented by the comparison methods, yet achieves superior results. The high performance can be attributed to starting at a better state by utilizing multi-modal pre-trained models with deep learnable prompts. The efficacy of AVL-Prompter is particularly pronounced in high-noise settings (symmetric 80% and 90% noise

Table 4
Comparison on EuroSAT and Oxford-IIIT Pets *(showing Best/Last Test accuracy)*. †-*Results updated from EPL [23].*

Noise Ratio	Sym. 60%	Asym. 40%	Sym. 60%	Asym. 40%
Std. CE†	70.8/70.0	75.9/71.4	52.3/47.1	59.5/58.5
EPL [23]	93.6/92.0	92.0/91.4	82.0/81.7	80.7/80.4
AVL-Prompter	**98.5** / **98.4**	**95.2** / **93.3**	**87.2** / **86.8**	**91.6** / **90.7**

Table 5
Comparison with SoTA methods on (mini) WebVision Dataset. Results are reported for both WebVision and ILSVRC12 (ImageNet) Validation sets. †-*Results updated from UNICON [69].* ‡-*Results updated from RankMatch [55].* ⋆-*Results obtained using ViT-L/14-CLIP.*

Dataset	WebVision			
Validation	WebVision		ILSVRC12	
Performance	Top-1	Top-5	Top-1	Top-5
DivideMix [49]	77.32	91.64	75.20	90.84
RRL‡ [82]	77.80	91.30	74.40	90.90
C2D [52]	$78.57_{\pm 0.37}$	$93.04_{\pm 0.10}$	$79.42_{\pm 0.34}$	$92.32_{\pm 0.33}$
UNICON [69]	77.60	93.44	75.29	93.72
LongReMix [83]	78.92	92.32	-	-
RankMatch [55]	79.91	93.61	77.39	94.26
Bayesian DivideMix++ [56]	$80.12_{\pm 0.28}$	$92.40_{\pm 0.30}$	$78.51_{\pm 0.28}$	$92.67_{\pm 0.42}$
SplitNet [62]	81.34	93.82	76.11	94.24
EPL⋆ (DM) [23]	77.53	92.89	75.47	91.74
EPL⋆ (ELR+) [23]	77.94	92.92	73.11	90.21
EPL⋆ (UNICON) [23]	77.75	93.74	75.93	93.79
AVL-Prompter	**83.00**	**95.92**	**82.72**	**97.40**

settings). Notably, in CIFAR-100 experiments, AVL-Prompter demonstrates substantial improvements, with a 28% increase in performance in symmetric 90% settings and a 16% increase in symmetric 80% settings. Furthermore, AVL-Prompter exhibits notable performance gains in asymmetric noise settings, where sample selection poses significant challenges, highlighting its remarkable capability.

We further compare the performance of the proposed AVL-Prompter using Tiny-ImageNet (Table 3), EuroSAT and Oxford-IIIT Pets (Table 4). Across all three datasets, AVL-Prompter consistently outperforms state-of-the-art methods. Particularly in the case of EuroSAT and Oxford-IIIT Pets, AVL-Prompter achieves higher performance compared to EPL which uses a much larger ViT-L/14-CLIP, whereas in AVL-Prompter, we use ViT-B/32-CLIP, further underscoring the effectiveness of our proposed method.

Performance on real noise. We show the results of AVL-Prompter for WebVision in Table 5. We achieve 1.7% improvement over SoTA on the Top-1 WebVision validation accuracy and 2.1% improvement on the Top-5 validation accuracy. Note that these are notable performance gains as they improve the SoTA by a much larger margin than previous approaches. Similarly, AVL-Prompter achieves a Top-1 performance gain of 3% and 5% Top-5 performance improvement in ILSVRC12 validation set. These results underscore the efficiency of AVL-Prompter, particularly in handling large noisy datasets, thereby demonstrating its scalability.

We demonstrate the performance of AVL-Prompter on CIFAR-10N/100N datasets in Table 6. The results show considerable improvements in all considered noise settings on both datasets. The gains are particularly evident in the Noisy Fine setting of CIFAR-100N (a comparatively more complex dataset than CIFAR-10N), where AVL-Prompter shows an improvement of ≈6.5%.

Summary of results. Overall, our proposed AVL-Prompter consistently outperforms other state-of-art methods across both synthetic and real noise benchmarks. The performance gains are especially pronounced under complex noise settings such as high-noise conditions (28% gain at 90% symmetric noise and 16% at 80% symmetric noise on CIFAR-100), highlighting the method's robustness, scalability, and effectiveness in challenging label noise scenarios. Remarkably, our proposed method uses simple LNL strategy and manages to outperfom more complex methods, including using smaller backbone (ViT-B/32) compared to EPL which uses a ViT-L/14 backbone.

4.3. Analysis

In this section, we provide in-depth insights into the results delineated within the manuscript, elucidating their significance and implications.

Memorization of noisy labels. Deep neural networks often learn the underlying patterns first and subsequently memorize all the samples [8]. When confronted with noisy labels, there is a tendency of the models to learn the clean samples first and later memorize the label noise, leading to overfitting of the models to label noise [56]. Accessing the effectiveness of LNL algorithms against memorization of noisy samples thus becomes pivotal. We analyse the performance of the best and the final AVL-Prompter models against other alternative models (used in ablation study) in this regard and report the categorization in Tables 7 and 8. We also visually show the percentage of samples categorized in Fig. 3. We categorize the number of clean samples correctly classified and clean samples misclassified. We also determine the number of noisy samples that are correctly classified after noise correction, the number of noisy samples that are still misclassified after noise correction (not the same as noisy labels) and the number of noisy samples retaining the noisy labels (i.e. memorized).

In the 'best' models case (Table 7), Zero-shot CLIP and Text Only Prompt Learner exhibit comparative categorization correctness to that of the AVL-Prompter models. However, they notably fall behind in the 'final' model case (Table 8), indicating a decline in their generalization ability. Indeed, the performance of the 'final' models reveals significant overfitting for Zero-shot CLIP and Text Only Prompt Learner models, as shown by the large amount of memorized samples. In contrast, the 'best' AVL-Prompter model demonstrates a lower number of clean samples that are wrongly classified, while correctly detecting a higher number of clean samples. Despite observing a small number of samples that are memorized, it remains consistent with the best and final models, highlighting the absence of overfitting in the proposed method. These findings underscore the robustness of AVL-Prompter against noise memorization.

UMAP-visualizations. Fig. 4 depicts the visual-language features and image features of our proposed model using UMAP [87]. We extract the image features from the image encoder, whereas the V-L features represent the combined features of both text and image encoders. Although the pre-trained initialization serves as a zero-shot classification model, it lacks discriminative features, leading to a loosely connected set of coarse groups (Fig. 4(a)). In contrast, our proposed method exhibits distinctly superior features, evident from the separation between the coarse groups (Fig. 4(b)). By comparing the features derived solely from the image encoder (left) with those derived from both encoders (right), it is evident that the combined features result in enhanced classification ability.

Table 6
Comparison on CIFAR-N datasets showing Best Test accuracy. †-*Results updated from CCLM [57].* ‡-*Results updated from SplitNet [62]* .

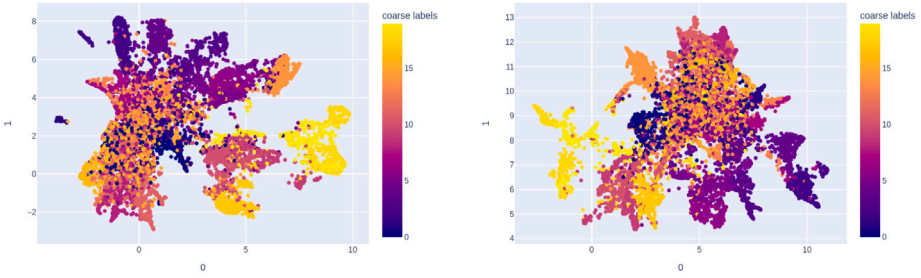
Method	CIFAR-10N					CIFAR-100N
	Agg.	Ran. 1	Ran. 2	Ran. 3	Worst	Noisy Fine
DivideMix† [49]	95.3	95.6	95.6	95.6	93.2	69.4
C2D† [52]	95.7	95.9	95.8	93.8	92.8	70.8
CCLM(DM) [57]	95.6	95.6	95.4	95.6	93.0	70.7
CCLM(C2D) [57]	95.8	95.7	95.7	95.9	92.4	70.6
PES(Semi)‡ [45]	94.66	95.06	95.19	95.22	92.68	70.36
SplitNet [62]	96.50	96.47	96.42	96.27	94.22	72.61
AVL-Prompter	**97.62**	**97.56**	**97.22**	**97.42**	**95.14**	**79.04**

Table 7
Categorizing clean and noisy samples identified by the **best** model in CIFAR-100 Sym. 80% noise setting. *Numbers in brackets indicate total samples based on ground truth labels.*

Method	Clean (10389)		Noisy (39611)		
	Correct	Wrong	Correct	Wrong	Memorized
Zero-shot CLIP	8849	1540	30436	8988	187
Text Only Prompt Learner	9082	1307	30887	8484	240
V-L Prompt Learner	8930	1459	31942	7540	129
AVL-Prompter	9577	812	32564	6802	245

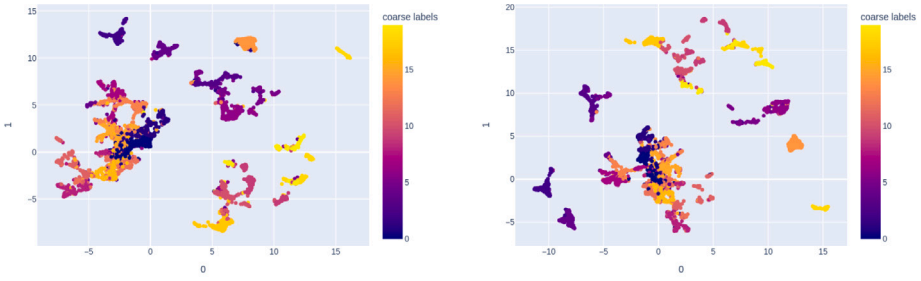
Table 8
Categorizing clean and noisy samples identified by the **final** model (last epoch) in CIFAR-100 Sym. 80% noise setting. *Numbers in brackets indicate total samples based on ground truth labels.*

Method	Clean (10389)		Noisy (39611)		
	Correct	Wrong	Correct	Wrong	Memorized
Zero-shot CLIP	10307	82	121	444	39046
Text Only Prompt Learner	10293	96	60	244	39307
V-L Prompt Learner	9327	1062	31488	7905	218
AVL-Prompter	9582	807	32582	6783	246



V-L Features Image Features

(a) Pre-trained AVL-Prompter (Model Initialization)



V-L Features Image Features

(b) After proposed AVL-Prompter Training

Fig. 4. UMAP visualization of our proposed AVL-Prompter Method. *(We use the coarse labels of CIFAR-100 for better visualizations.*

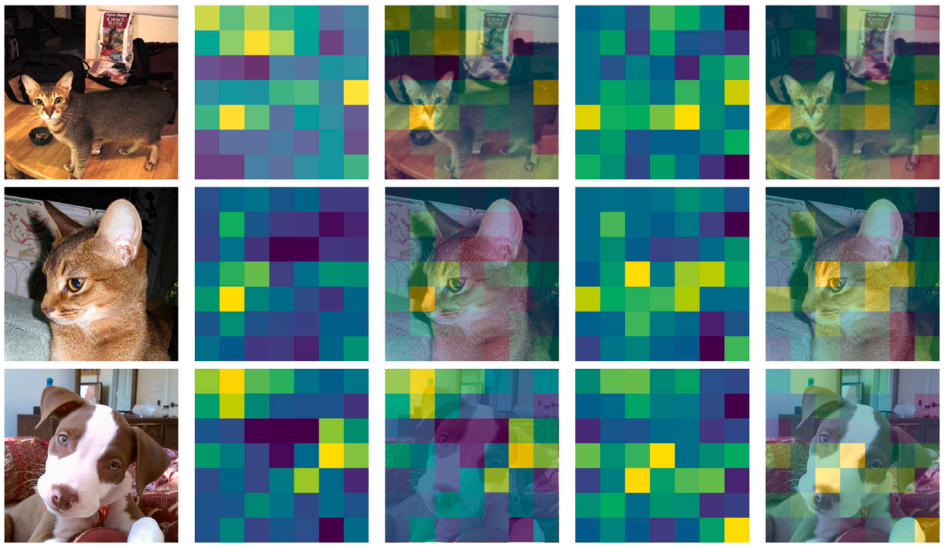


Fig. 5. Attention Map Visualizations of AVL-Prompter in Oxford-IIIT Pets dataset (60% Sym. noise). From the left: first column shows original images, second and third shows the attention maps and images overlayed during model initialization, fourth and fifth shows the attention maps and images overlayed after training.

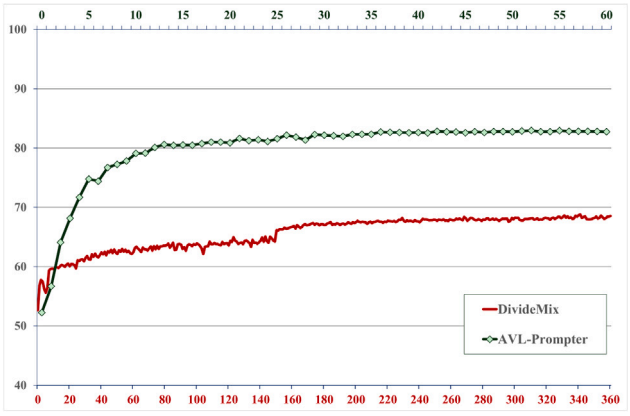


Fig. 6. Training Progression of DivideMix against AVL-Prompter in CIFAR-100 Sym. 80% noise setting. *y-axis represents the model accuracy. Lower x-axis shows the training epochs of DivideMix, while the upper x-axis shows that of AVL-Prompter.*

Attention maps. We show the attention maps of AVL-Prompter during model initialization and after adaptive training on the Oxford-IIIT Pets dataset (60% Sym. noise) in Fig. 5. During initialization, the model activates both the pet body and the background noise, reflecting the lack of discrimination for high-noise data. After training, attention is significantly focused on the semantic core such as the pet's head, eyes and ears, and the background noise response is suppressed, highlighting strong robustness in high-noise scenarios. Over training, AVL-Prompter readjusts the prompt embedding space to better equip for handling noisy embeddings.

Training progression. We present a comparative analysis of the training progression between our proposed AVL-Prompter and DivideMix [49] in Fig. 6. Leveraging a pre-trained visual language model affords AVL-Prompter a more robust initialization and notably, eliminates the necessity for a warmup phase, contrasting with DivideMix. This mitigates the risk of learning noisy samples during the warmup phase, as highlighted in [52].

4.4. Ablation study

In this section, we provide insights into the design decisions underlying our proposed AVL-Prompter method. To illustrate these decisions, we utilize CIFAR-100 symmetric 60% noise settings. Following this, we show a comprehensive analysis of each component of AVL-Prompter, conducted using CIFAR-100 symmetric 80% noise settings.

Table 9
Design decisions of AVL-Prompter. *Results shown for CIFAR-100 Sym. 60%.*

Experiment	Description	Test Accuracy
Selection of Prompting Technique	LP	82.7
	FT	**85.6**
Fine-Tuning Encoders	Only text encoder	83.3
	Only image encoder	85.1
	Both text and image encoders	**85.6**

Selection of prompting technique. Prompt Learning introduces learnable prompts to adapt models for downstream tasks. While conventional prompt learning methods maintain frozen pre-trained model weights, learnable prompts can lead to overfitting task-specific data distributions [31]. In this analysis, we investigate the behaviour of prompt learning across different model settings. Linear Probing (LP) involves updating only the parameters of the last fully connected layer, along with the learnable prompts, while keeping the feature extractors frozen. Fine-tuning (FT) entails updating all parameters of both encoders during the learning process. We report the performance of AVL-Prompter across different prompting techniques in Table 9. Notably, fine-tuning the encoders with learnable prompts demonstrates superiority compared to linear probing. This advantage could stem from the model's ability to adapt and learn noise patterns when coupled with the semi-supervised learning component of AVL-Prompter.

Effect of fine-tuning encoders. AVL-Prompter comprises two encoders - a text encoder and an image encoder, both sharing learnable prompts. In this analysis (Table 9), we examine the behaviour of these encoders when fine-tuning them. Note that, when only one encoder is fine-tuned, the other is kept frozen. The results indicate that fine-tuning both encoders yields better performance compared to fine-tuning only one of them. This observation underscores the effectiveness of aligning both encoders, enhancing the overall model's capability. Furthermore, it can be noted that fine-tuning only the text encoder results in lower performance compared to fine-tuning the image encoder. This difference can be attributed to the model adapting to the label noise without the corrective information from the image encoder.

Table 10
Ablation study of AVL-Promptercomponents. *Results shown for CIFAR-100 Sym. 80% noise. Models without Semi-supervised Learning (×) are trained with standard Cross Entropy Loss.*

S.No	Model	Prompt Learning	Semi-supervised Learning	Test Accuracy
1	(Pre-trained) ResNet-50	×	×	67.1
2	Zero-shot CLIP	×	×	77.1
3	Text Only Prompt Learner	✓	×	78.3
4	V-L Prompt Learner	✓	×	80.9
5	V-L Prompt Learner	✓	✓	**83.0**

Effect of the proposed components. We investigate the roles of prompt learning and semi-supervised learning components within the AVL-Prompter (Table 10). Firstly, we assess the significance of prompt learning. To do so, we substitute the proposed V-L prompt learning model with ImageNet pre-trained ResNet-50 [2] and Zero-shot CLIP model [29] (*#1 and #2* in Table 10, respectively), and compare their results to that of the V-L prompt training model without semi-supervised learning (*#4* in Table 10). The V-L prompt learning model outperforms by substantial margins. Subsequently, we emphasize the importance of learnable prompts (*#3 and #4*) over static prompts in LNL settings (*#2*). We can notice that Zero-shot CLIP uses a static prompt of the format "a photo of a class". The observed performance improvements across all models utilizing prompt learning underscore the importance of the learnable prompts. Furthermore, we compare the impact of employing prompt learning solely on the text encoder [35] (*#3*) with using shared learnable prompts (*#4*). This comparison shows substantial improvements achieved through the utilization of shared prompts across both encoders.

Lastly, we study the role of semi-supervised learning within AVL-Prompter by removing the semi-supervised learning component (*#4 and #5* in Table 10). The results validate the effectiveness of the proposed components, underlying their significance in enhancing model performance.

4.5. Limitations

The proposed AVL-Prompter for LNL has shown promising results. However, during our research, we identified certain limitations in our approach that can pave the way for future research directions:

- **Fine-Tuning Transformers:** The core component of AVL-Prompter lies in fine-tuning of V-L models. Fine-tuning transformer architectures, particularly large models such as ViT-L, present a complex challenge [88]. ViT-B/32 has 151M params, while, ViT-L/14 has 428M params. Selecting appropriate hyperparameters, batch sizes, and learning schedules becomes difficult due to the multitude of potential configurations [89].
- **Computational Complexity:** Fine-tuning yields significant performance enhancements to linear probing. However, the computational cost associated with fine-tuning models poses a notable challenge, which is particularly pronounced in places of fixed computing budget. Additionally, selecting the right architecture for a given dataset is challenging, often requiring costly trial-and-error procedures.
- **Noise Level Estimation:** The efficacy of AVL-Prompter hinges on accurately estimating the noise level in the dataset. However, in real-world datasets, noise distributions are often unknown and heterogeneous. Inaccurate noise estimation could result in improper sample selection, consequently affecting the model performance.
- **Pre-trained Data Dependence:** Pre-trained V-L models are trained on large-scale web-curated data. The quality and diversity of this pre-training data significantly influence downstream performance. Inherent biases or lack of domain-relevant information affect the learned representations.

5. Conclusions

In this paper, we propose a novel, simple, but highly efficient approach, AVL-Prompter, to the learning with noisy labels problem. By resorting to shared deep learnable prompts, we adapt large vision-language models to the challenging task of learning with noisy labels, hence fully benefiting from their superior capabilities. Our results confirm that vision-language models can be effectively trained in noisy label settings, bringing significant improvements over SoTA, even in the most challenging cases. We believe that our approach sets a new standard in learning with noisy labels, and opens a new research avenue in this domain. Future work could aim at reducing the computational cost of the training step by developing an efficient strategy where only the crucial part of the model is trained.

CRediT authorship contribution statement

Changhui Hu: Writing – original draft, Validation, Software, Methodology, Conceptualization. **Bhalaji Nagarajan:** Writing – original draft, Validation, Methodology, Conceptualization. **Ricardo Marques:** Writing – review & editing, Supervision, Methodology, Investigation, Conceptualization. **Petia Radeva:** Writing – review & editing, Supervision, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was partially funded by the EU project MUSAE (No. 01070421), 2021-SGR-01094 (AGAUR), Icrea Academia'2022 (Generalitat de Catalunya), Robo STEAM (2022-1-BG01-KA220-VET-000089434, Erasmus+ EU), DeepSense (ACE053/22/000029, ACCIÓ), DeepFoodVol (AEI-MICINN, PDC2022-133642-I00), IDEATE (AEI-MICINN, PID2022-141566NB-I00), and A-BMC (AEI-MICINN, CNS2022-135480). C. Hu acknowledges China Scholarship Council (CSC) Fellowship (Ministry of Education, China, No. 202208410100).

Data availability

Data will be made available on request.

References

[1] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (2012).
[2] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[3] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[4] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst. 28 (2015).
[5] Y. Yao, T. Chen, G.-S. Xie, C. Zhang, F. Shen, Q. Wu, Z. Tang, J. Zhang, Non-salient region object mining for weakly supervised semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 2623–2632.
[6] R. Wang, H. Sun, Y. Ma, X. Xi, Y. Yin, MetaViewer: Towards a unified multi-view representation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 11590–11599.
[7] T. Xiao, T. Xia, Y. Yang, C. Huang, X. Wang, Learning from massive noisy labeled data for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2691–2699.

[8] D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M.S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, et al., A closer look at memorization in deep networks, in: International Conference on Machine Learning, PMLR, 2017, pp. 233–242.

[9] S. Liu, J. Niles-Weed, N. Razavian, C. Fernandez-Granda, Early-learning regularization prevents memorization of noisy labels, Adv. Neural Inf. Process. Syst. 33 (2020) 20331–20342.

[10] Y. Xu, P. Cao, Y. Kong, Y. Wang, L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise, Adv. Neural Inf. Process. Syst. 32 (2019).

[11] D. Ortego, E. Arazo, P. Albert, N.E. O'Connor, K. McGuinness, Multi-objective interpolation training for robustness to label noise, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 6606–6615.

[12] H. Wei, L. Feng, X. Chen, B. An, Combating noisy labels by agreement: A joint training method with co-regularization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13726–13735.

[13] B. Han, G. Niu, X. Yu, Q. Yao, M. Xu, I. Tsang, M. Sugiyama, Sigua: Forgetting may make learning with noisy labels more robust, in: International Conference on Machine Learning, PMLR, 2020, pp. 4006–4016.

[14] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning (still) requires rethinking generalization, Commun. ACM 64 (3) (2021) 107–115.

[15] D. Angluin, P. Laird, Learning from noisy examples, Mach. Learn. 2 (1988) 343–370.

[16] H. Song, M. Kim, D. Park, Y. Shin, J.-G. Lee, Learning from noisy labels with deep neural networks: A survey, IEEE Trans. Neural Netw. Learn. Syst. (2022).

[17] Y. Zhang, G. Niu, M. Sugiyama, Learning noise transition matrix from only noisy labels via total variation regularization, in: International Conference on Machine Learning, PMLR, 2021, pp. 12501–12512.

[18] S. Wu, X. Xia, T. Liu, B. Han, M. Gong, N. Wang, H. Liu, G. Niu, Class2simi: A noise reduction perspective on learning with noisy labels, in: International Conference on Machine Learning, PMLR, 2021, pp. 11285–11295.

[19] M. Ren, W. Zeng, B. Yang, R. Urtasun, Learning to reweight examples for robust deep learning, in: International Conference on Machine Learning, PMLR, 2018, pp. 4334–4343.

[20] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, J. Bailey, Symmetric cross entropy for robust learning with noisy labels, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 322–330.

[21] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, J. Bailey, Normalized loss functions for deep learning with noisy labels, in: International Conference on Machine Learning, PMLR, 2020, pp. 6543–6553.

[22] S. Ahn, S. Kim, J. Ko, S.-Y. Yun, Fine tuning pre trained models for robustness under noisy labels, 2023, arXiv preprint arXiv:2310.17668.

[23] J. Ko, S. Ahn, S.-Y. Yun, Efficient utilization of pre-trained model for learning with noisy labels, in: ICLR 2023 Workshop on Pitfalls of Limited Data and Computation for Trustworthy ML, 2023.

[24] Y. Wei, H. Hu, Z. Xie, Z. Liu, Z. Zhang, Y. Cao, J. Bao, D. Chen, B. Guo, Improving CLIP fine-tuning performance, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 5439–5449.

[25] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.

[26] T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Adv. Neural Inf. Process. Syst. 33 (2020) 1877–1901.

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

[28] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, S. Xie, A convnet for the 2020s, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11976–11986.

[29] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PMLR, 2021, pp. 8748–8763.

[30] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, T. Duerig, Scaling up visual and vision-language representation learning with noisy text supervision, in: International Conference on Machine Learning, PMLR, 2021, pp. 4904–4916.

[31] M.U. Khattak, S.T. Wasim, M. Naseer, S. Khan, M.-H. Yang, F.S. Khan, Self-regulating prompts: Foundational model adaptation without forgetting, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 15190–15200.

[32] C.-E. Wu, Y. Tian, H. Yu, H. Wang, P. Morgado, Y.H. Hu, L. Yang, Why is prompt tuning for vision-language models robust to noisy labels? in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 15488–15497.

[33] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, G. Neubig, Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, ACM Comput. Surv. 55 (9) (2023) 1–35.

[34] K. Zhou, J. Yang, C.C. Loy, Z. Liu, Learning to prompt for vision-language models, Int. J. Comput. Vis. 130 (9) (2022) 2337–2348.

[35] K. Zhou, J. Yang, C.C. Loy, Z. Liu, Conditional prompt learning for vision-language models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 16816–16825.

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

[37] T. Wei, H.-T. Li, C. Li, J.-X. Shi, Y.-F. Li, M.-L. Zhang, Vision-language models are strong noisy label detectors, Adv. Neural Inf. Process. Syst. 37 (2024) 58154–58173.

[38] B. Pan, Q. Li, X. Tang, W. Huang, Z. Fang, F. Liu, J. Wang, J. Yu, Y. Shi, NLPrompt: Noise-label prompt learning for vision-language models, in: Proceedings of the Computer Vision and Pattern Recognition Conference, 2025, pp. 19963–19973.

[39] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, M. Sugiyama, Co-teaching: Robust training of deep neural networks with extremely noisy labels, Adv. Neural Inf. Process. Syst. 31 (2018).

[40] G. Patrini, A. Rozza, A. Menon, R. Nock, L. Qu, Making neural networks robust to label noise: a loss correction approach, Stat 1050 (2016) 13.

[41] X. Liu, S. Luo, L. Pan, Robust boosting via self-sampling, Knowl.-Based Syst. 193 (2020) 105424.

[42] H. Zhang, M. Cisse, Y.N. Dauphin, D. Lopez-Paz, Mixup: Beyond empirical risk minimization, 2017, arXiv preprint arXiv:1710.09412.

[43] Y. Chen, S.X. Hu, X. Shen, C. Ai, J.A. Suykens, Compressing features for learning with noisy labels, IEEE Trans. Neural Netw. Learn. Syst. (2022).

[44] A.K. Menon, A.S. Rawat, S.J. Reddi, S. Kumar, Can gradient clipping mitigate label noise? in: International Conference on Learning Representations, 2020.

[45] Y. Bai, E. Yang, B. Han, Y. Yang, J. Li, Y. Mao, G. Niu, T. Liu, Understanding and improving early stopping for learning with noisy labels, Adv. Neural Inf. Process. Syst. 34 (2021) 24392–24403.

[46] Y. Tu, B. Zhang, Y. Li, L. Liu, J. Li, Y. Wang, C. Wang, C.R. Zhao, Learning from noisy labels with decoupled meta label purifier, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 19934–19943.

[47] A. Ghosh, H. Kumar, P.S. Sastry, Robust loss functions under label noise for deep neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, no. 1, 2017.

[48] E. Arazo, D. Ortego, P. Albert, N. O'Connor, K. McGuinness, Unsupervised label noise modeling and loss correction, in: International Conference on Machine Learning, PMLR, 2019, pp. 312–321.

[49] J. Li, R. Socher, S.C. Hoi, Dividemix: Learning with noisy labels as semi-supervised learning, 2020, arXiv preprint arXiv:2002.07394.

[50] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, L. Fei-Fei, Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels, in: International Conference on Machine Learning, PMLR, 2018, pp. 2304–2313.

[51] X. Yu, B. Han, J. Yao, G. Niu, I. Tsang, M. Sugiyama, How does disagreement help generalization against label corruption? in: International Conference on Machine Learning, PMLR, 2019, pp. 7164–7173.

[52] E. Zheltonozhskii, C. Baskin, A. Mendelson, A.M. Bronstein, O. Litany, Contrast to divide: Self-supervised pre-training for learning with noisy labels, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 1657–1667.

[53] K. Nishi, Y. Ding, A. Rich, T. Hollerer, Augmentation strategies for learning with noisy labels, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8022–8031.

[54] Y. Li, H. Han, S. Shan, X. Chen, DISC: Learning from noisy labels via dynamic instance-specific selection and correction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 24070–24079.

[55] Z. Zhang, W. Chen, C. Fang, Z. Li, L. Chen, L. Lin, G. Li, RankMatch: Fostering confidence and consistency in learning with noisy labels, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 1644–1654.

[56] B. Nagarajan, R. Marques, E. Aguilar, P. Radeva, Bayesian DivideMix++ for enhanced learning with noisy labels, Neural Netw. 172 (2024) 106122.

[57] A. Tatjer, B. Nagarajan, R. Marques, P. Radeva, Decoding class dynamics in learning with noisy labels, Pattern Recognit. Lett. (2024).

[58] F. Fooladgar, M.N.N. To, P. Mousavi, P. Abolmaesumi, Manifold DivideMix: A semi-supervised contrastive learning framework for severe label noise, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 4012–4021.

[59] Z. Sun, F. Shen, D. Huang, Q. Wang, X. Shu, Y. Yao, J. Tang, Pnp: Robust learning from noisy labels by probabilistic noise prediction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 5311–5320.

[60] R. Xiao, Y. Dong, H. Wang, L. Feng, R. Wu, G. Chen, J. Zhao, ProMix: Combating label noise via maximizing clean sample utility, in: Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23, 2023, pp. 4442–4450.
[61] Y. Huang, B. Bai, S. Zhao, K. Bai, F. Wang, Uncertainty-aware learning against label noise on imbalanced datasets, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, no. 6, 2022, pp. 6960–6969.
[62] D. Kim, K. Ryoo, H. Cho, S. Kim, SplitNet: learnable clean-noisy label splitting for learning with noisy labels, Int. J. Comput. Vis. (2024) 1–18.
[63] L. Yuan, D. Chen, Y.-L. Chen, N. Codella, X. Dai, J. Gao, H. Hu, X. Huang, B. Li, C. Li, et al., Florence: A new foundation model for computer vision, 2021, arXiv preprint arXiv:2111.11432.
[64] B. Zhu, Y. Niu, Y. Han, Y. Wu, H. Zhang, Prompt-aligned gradient for prompt tuning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 15659–15669.
[65] D. Lee, S. Song, J. Suh, J. Choi, S. Lee, H.J. Kim, Read-only prompt optimization for vision-language few-shot learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 1401–1411.
[66] R. Huang, Y. Long, J. Han, H. Xu, X. Liang, C. Xu, X. Liang, Nlip: Noise-robust language-image pre-training, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 1, 2023, pp. 926–934.
[67] T. Shin, Y. Razeghi, R.L. Logan IV, E. Wallace, S. Singh, Autoprompt: Eliciting knowledge from language models with automatically generated prompts, 2020, arXiv preprint arXiv:2010.15980.
[68] A. Tarvainen, H. Valpola, Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, Adv. Neural Inf. Process. Syst. 30 (2017).
[69] N. Karim, M.N. Rizve, N. Rahnavard, A. Mian, M. Shah, Unicon: Combating label noise through uniform selection and contrastive learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9676–9686.
[70] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C.A. Raffel, E.D. Cubuk, A. Kurakin, C.-L. Li, Fixmatch: Simplifying semi-supervised learning with consistency and confidence, Adv. Neural Inf. Process. Syst. 33 (2020) 596–608.
[71] P. Bachman, O. Alsharif, D. Precup, Learning with pseudo-ensembles, Adv. Neural Inf. Process. Syst. 27 (2014).
[72] S. Laine, T. Aila, Temporal ensembling for semi-supervised learning, 2016, arXiv preprint arXiv:1610.02242.
[73] M. Sajjadi, M. Javanmardi, T. Tasdizen, Regularization with stochastic transformations and perturbations for deep semi-supervised learning, Adv. Neural Inf. Process. Syst. 29 (2016).
[74] E.D. Cubuk, B. Zoph, J. Shlens, Q.V. Le, Randaugment: Practical automated data augmentation with a reduced search space, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 702–703.
[75] A. Krizhevsky, G. Hinton, et al., Learning multiple layers of features from tiny images, 2009.
[76] P. Helber, B. Bischke, A. Dengel, D. Borth, Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification, IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens. 12 (7) (2019) 2217–2226.
[77] Y. Le, X. Yang, Tiny imagenet visual recognition challenge, 7(7), 2015, p. 3, CS 231N.
[78] O.M. Parkhi, A. Vedaldi, A. Zisserman, C. Jawahar, Cats and dogs, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 3498–3505.
[79] J. Wei, Z. Zhu, H. Cheng, T. Liu, G. Niu, Y. Liu, Learning with noisy labels revisited: A study using real-world human annotations, in: International Conference on Learning Representations, 2022.
[80] W. Li, L. Wang, W. Li, E. Agustsson, L. Van Gool, Webvision database: Visual learning and understanding from web data, 2017, arXiv preprint arXiv:1708.02862.
[81] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (2015) 211–252.
[82] J. Li, C. Xiong, S.C. Hoi, Learning from noisy data with robust representation learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9485–9494.
[83] F.R. Cordeiro, R. Sachdeva, V. Belagiannis, I. Reid, G. Carneiro, Longremix: Robust learning with high confidence samples in a noisy label environment, Pattern Recognit. 133 (2023) 109013.
[84] D. Tanaka, D. Ikami, T. Yamasaki, K. Aizawa, Joint optimization framework for learning with noisy labels, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5552–5560.
[85] K. Yi, J. Wu, Probabilistic end-to-end noise correction for learning with noisy labels, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 7017–7025.
[86] F. Sarfraz, E. Arani, B. Zonooz, Noisy concurrent training for efficient learning under label noise, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 3159–3168.
[87] L. McInnes, J. Healy, J. Melville, Umap: Uniform manifold approximation and projection for dimension reduction, 2018, arXiv preprint arXiv:1802.03426.
[88] E. Radiya-Dixit, X. Wang, How fine can fine-tuning be? learning efficient language models, in: International Conference on Artificial Intelligence and Statistics, PMLR, 2020, pp. 2435–2443.
[89] Y. Tay, M. Dehghani, J. Rao, W. Fedus, S. Abnar, H.W. Chung, S. Narang, D. Yogatama, A. Vaswani, D. Metzler, Scale efficiently: Insights from pre-training and fine-tuning transformers, 2021, arXiv preprint arXiv:2109.10686.