Weakly Supervised Learning with Side Information for Noisy Labeled Images

Lele Cheng(✉), Xiangzeng Zhou(✉), Liming Zhao(✉), Dangwei Li(✉), Hong Shang(✉), Yun Zheng(✉), Pan Pan(✉), and Yinghui Xu(✉)

Machine Intelligence Technology Lab, Damo Academy, Alibaba Group, Hangzhou, China
{yinan.cll,xiangzeng.zxz,lingchen.zlm,dangwei.ldw,shanghong.sh, zhengyun.zy,panpan.pp}@alibaba-inc.com, renji.xyh@taobao.com

Abstract. In many real-world datasets, like WebVision, the performance of DNN based classifier is often limited by the noisy labeled data. To tackle this problem, some image related side information, such as captions and tags, often reveal underlying relationships across images. In this paper, we present an efficient weakly-supervised learning by using a Side Information Network (SINet), which aims to effectively carry out a large scale classification with severely noisy labels. The proposed SINet consists of a visual prototype module and a noise weighting module. The visual prototype module is designed to generate a compact representation for each category by introducing the side information. The noise weighting module aims to estimate the correctness of each noisy image and produce a confidence score for image ranking during the training procedure. The propsed SINet can largely alleviate the negative impact of noisy image labels, and is beneficial to train a high performance CNN based classifier. Besides, we released a fine-grained product dataset called AliProducts, which contains more than 2.5 million noisy web images crawled from the internet by using queries generated from 50,000 fine-grained semantic classes. Extensive experiments on several popular benchmarks (i.e. Webvision, ImageNet and Clothing-1M) and our proposed AliProducts achieve state-of-the-art performance. The SINet has won the first place in the 5000 category classification task on WebVision Challenge 2019, and outperforms other competitors by a large margin.

Keywords: Weakly supervised learning · Noisy labels · Side information · Large scale web images

1 Introduction

In recent years, the computer vision community has witnessed the significant success of Deep Neural Networks (DNNs) on several benchmark datasets of image classification, such as ImageNet [1] and MS-COCO [22]. However, obtaining large-scale data with clean and reliable labels is expensive and time-consuming.

A. Vedaldi et al. (Eds.): ECCV 2020, LNCS 12375, pp. 306–321, 2020.
https://doi.org/10.1007/978-3-030-58577-8_19

When noisy labels are introduced in training data, it is widely known that the performance of a deep model can be significantly degraded [2,3,23,36], which prevents deep models from being quickly employed in real-world noisy scenarios.

A common solution is to collect a large amount of image related side information (e.g. surrounding texts, tags and descriptions) from the internet, and directly take them as the ground-truth for model training. Though this solution is more efficient than manual annotation, the obtained labels usually contain noise due to the heterogeneous sources. Therefore, improving the robustness of deep learning models against noisy labels has become a critical issue.

To estimate the noise in labels, some works propose new layers [26,27] or loss functions [18,28–30] to correct the noisy label during training. However, these works rely on a strict assumption that there is a single transition probability between the noisy labels and the ground-truth labels. Owning to this assumption, these methods may show good performance on hand-crafted noisy datasets but are inefficient on real noisy datasets such as Clothing1M [36]. In some situations, it is possible to annotate a small fraction of training samples as additional supervision. By using additional supervision, works like [11,31,32] could improve the robustness of deep networks against label noises. But still, the requirement on clean samples make them less flexible to apply in large scale real-world scenarios.

Many data cleaning algorithms [33–35] are developed to discard those samples with wrong label ahead of the training procedure. The major difficulty of these algorithms is how to distinguish informative hard samples from harmful mislabeled ones. CleanNet [11] achieves state-of-the-art performance on the real-world noisy dataset Clothing1M [36]. CleanNet generates a single representative sample (class prototype) for each class and uses it to estimate the correctness of sample labels. With the observation that samples have wide-spread distribution in noisy classes, SMP [20] takes multiple prototypes to represent a noisy class instead of single prototype in CleanNet. In both CleanNet and SMP, extra clean supervision is required to train models.

In most of previous works, image related side information or annotations (e.g. titles and tags) from web are commonly regarded as noisy labels. These works may not fully take advantage of the side information. Based on our observations, these image related side information reveal underlying similarity among images and classes, which has great potential to help tackle label noises. By analyzing the label structure and text descriptions, we explore an weakly-supervised learning strategy to deal with noisy samples. For example, the label "apple" may refer to a fruit or an Apple mobile phone. When acquiring images from web using the label "apple", images of apple fruits and Apple mobile phones will be wrongly put under a same class. Fortunately, titles or text descriptions about the images could imply the misplacement. In this paper, we propose an efficient weakly-supervised learning strategy to evaluate the correctness of each image sample in each class by exploiting the label structure and label descriptions. Moreover, we release a large scale fine-grained product dataset to facilitate further research

on visual recognition. To our knowledge, the proposed product dataset contains the largest number of product categories by now.

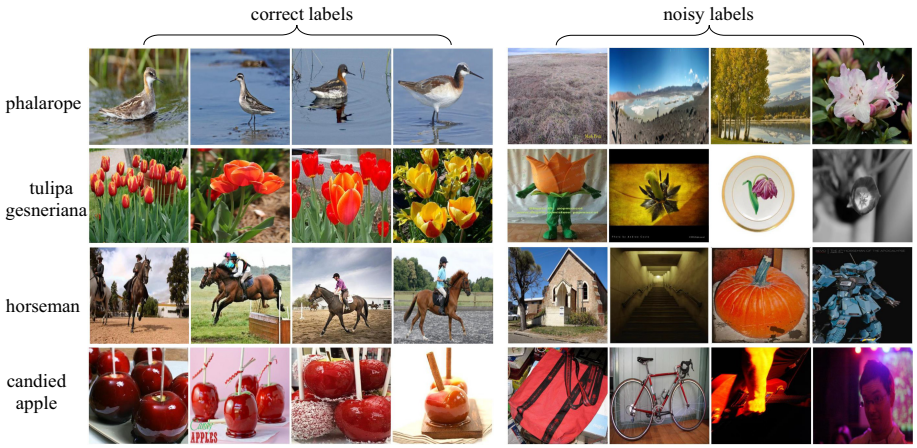

Fig. 1. Images of WebVision 2019 dataset [37] from the categories of phalarope, horseman, candied apple, tulipa, gesneriana. The dataset was collected from the Internet by textual queries generated from 5, 000 semantic concepts in WordNet. Obviously, each category includes a lot of noisy images as shown above.

Our contributions in this paper are summarized as follows:

1) A weakly supervised learning with side information network (SINet) is proposed for noisy labeled image classification. SINet infers the relationship between images and labels without any human annotation, and enable us to train high-performance and robust CNN models against large scale label noises.
2) A noisy and fine-grained product dataset called AliProducts is released, which contains more than 2.5 million web images crawled from the Internet by using queries generated from the 50, 000 fine-grained semantic classes. In addition, side information (e.g., hierarchical relationships between classes) are also provided for the convenience of extending research.
3) Extensive experiments are conducted on a number of benchmarks, including WebVision, ImageNet, Clothing1M and AliProducts, in which the proposed SINet obtains the state-of-the-art performance. Our SINET also won the first place on the WebVision Challenge 2019, and outperforms the other competitors by a large margin.

2 Related Work

Recent studies have shown that the performances of DNNs degraded substantially when training on data with noisy labels [2,3]. To alleviate this problem, a

number of approaches have been introduced and can be generally summarized as below.

Some methods design robust loss functions against label noises [4–9]. Zhang et al. [5] found that the mean absolute error (MAE) is inherently more robust to label noises than the commonly-used categorical cross entropy (CCE) in many circumstances. However, MAE performs poorly with DNNs and challenging datasets due to slow convergence. Generalized Cross Entropy (GCE) loss [9] applies a Box-Cox transformation to probabilities (power law function of probability with exponent q) and can be seen as a generalization of MAE and CCE, thus can be easily applied with existing DNN architecture and yield good performance in certain noisy datasets.

Re-weighting training samples aims to evaluate the correctness of each sample on a given label, and has been widely studied in [10–16,21]. In [12], meta learning paradigm is used to determine the sample weighting factors. [13] takes open-set noisy labels into consideration and train a Siamese network to detect noisy labels. In each iteration, sample weighting factors will be re-estimated, and the classifier will be updated at the same time. [14] also presents a method to separate clean samples from noisy samples in an iterative fashion. The biggest challenge encountering these data cleaning algorithms is how to distinguish informative hard samples from harmful mislabeled ones. To prevent discarding valuable hard samples, noisy samples is weighted according to their noisiness level which is estimated by pLOF [15]. In CleanNet [11], an additional network is designed to decide whether a sample is mislabelled or not. CleanNet aims to produce weights of samples during the training procedure. CurriculumNet [16] designs a learning curriculum by measuring the complexity of data and ranking samples in an unsupervised manner. However, most of these approaches either requires extra clean samples as additional information or adopts a complicated training procedure, making them less suitable for being widely applied in many real-world scenarios.

Self-learning pseudo-labels has been studied in many scenarios to deal with noisy labels. Reed et al. [17] propose to jointly train model with both noisy labels and pseudo-labels. However, [17] over-simplifies the assumption of the noisy label distribution, which leads to sub-optimal results. In the joint optimization process of [18], original noisy labels are completely replaced by pseudo-labels. This often discards some valuable information in the original noisy labels. Li et al. [19] proposes to simulate actual training by generating synthetic noisy labels, and train the model such that after one gradient update using each set of synthetic noisy labels, thus the model does not overfit to the specific noise.

Our method is similar to the work of [20], in which each class is represented by a learnable prototype. For each sample, a similarity is calculated between the sample and the corresponding prototype to correct its label. A final classifier is trained by using both the corrected label and the noisy label. However, [20] only takes visual information into consideration to construct class prototypes. Our approach integrate visual with side information to generate more reliable prototype for each class.

3 Approach

We focus on learning a robust image classifier from large-scale noisy images with side information. Let $\mathbb{D} = \{(x_1, y_1), ..., (x_N, y_N)\}$ be a noisily labeled dataset of N images. $y_i \in \{1, 2, ..., C\}$ is the noisy label corresponding to the image x_i, and C is the number of classes in the dataset.

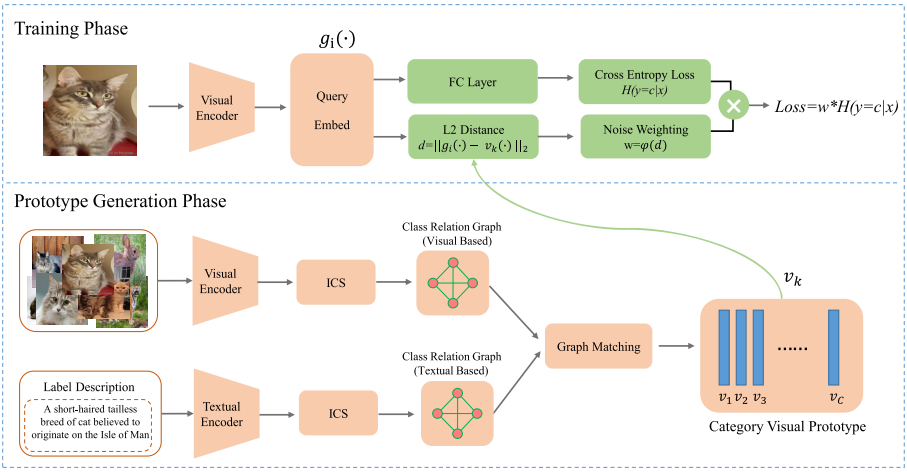

Fig. 2. Illustration of the framework of SINet on the noisy dataset, which includes three sub-modules, including *Class Relation Graph*, *Visual Prototype Generation*, and *Noise Weighting*. First, we construct visual-based and textual-based class relation graph with representation of image and textual using Inter Class Similarity (ICS), respectively. Second, we generate a visual prototype for each class using the compliance between the visual and textual class relations. Finally, *Noise Weighting* is used to weight all noisily labeled images with class prototypes before the training procedure.

Training with noisily labeled images, deep neural networks may over-fit these noisy labels and perform poorly. To alleviate this problem, we introduce a conceptually simple but effective side information network (SINet) for training against noisy labels. Based on the knowledge of "different classes look different" [38], we train the network under the constraint that the produced visual similarity between classes should have potential relevance with their natural semantic similarity. The semantic similarity is derived from a class relation graph constructed with the image related side information such as image titles, long text descriptions and image tags.

For each class, a prototype v_k, $k \in \{1, 2, ..C\}$ is generated from reliable training samples whose visual similarity graph aligns well with the constructed Class Relation Graph. Subsequently, we can decide whether an training sample is mislabeled or not by comparing its visual representation with v_k which is considered as a clean and reliable representation of k-th class. During the training

phrase, an image is recognized as a noisy sample or not in the light of the distance of the image feature and the class prototype. Instead of directly discarding noisy images by a predefined threshold, we assign each image a correctness weight in a noise weighting module.

Training a deep model $\mathcal{G}$ parameterized by θ on the dataset $\mathbb{D}$, the overall optimization objective is formulated as

$$\theta^* = argmin \sum_{i=1}^{N} w_i * L(y_i, \mathcal{G}(\theta, x_i)) \tag{1}$$

where L is a conventional cross entropy loss, and w_i is the image weight generated by the noise weighting module.

In the following sections, we elaborate the proposed SINet that using image related side information to facilitate a classification task on noisy images. The SINet comprises three modules, i.e. class relation graph generation, visual prototype generation and noise weighting. As shown in Fig. 2, an overview of the SINet architecture is illustrated. In Sect. 3.1, two kinds of category relation graphs are constructed using label embeddings and WordNet information, respectively. In Sect. 3.2, the KL divergence is used to estimated the compliance between the two kinds of graphs, so as to generate a visual prototype for each class. Given the class prototypes, a noise weighting module is presented to weight all noisily labeled images before the training procedure in Sect. 3.3.

3.1 Class Relation Graph

In some classification scenarios, for each class we can obtain both the long-text description of label and the hierarchical structure of class relationships using WordNet [40]. Both the label descriptions and WordNet structure reveal rich semantic information across classes. In this section, we attempt to exploit the inter-class semantic knowledge by constructing two kinds of class relation graphs. In the graphs, each node represents a class and the edges between nodes are built using two different similarity metrics.

Firstly, a straightforward way to build a class relation (marked as $\mathbb{G}_w$) is using the tree structure of WordNet. In the graph $\mathbb{G}_w$, an edge of two class nodes is created using the distance of the shortest path in the WordNet tree. Here we represent the WordNet-based class relation graph $\mathbb{G}_w$ as a matrix $S_w \in R^{C \times C}$.

Secondly, we learn a label embedding for each class node with the text description of label from WordNet. For instance, a label description *Manx cat: A short-haired tailless breed of cat believed to originate on the Isle of Man* provides rich semantic knowledge of the corresponding class. Moreover, these text descriptions reveal underlying relationship across classes from the perspective of natural language. To obtain a semantic representation of each class, we use a BERT [39] language model to learn a sequence of word embeddings from text description of each class label. We then feed them into a bidirectional LSTM module to achieve a class label level embedding (called label embedding). Please note that the number of label text descriptions available for training is too small,

so we use a pretrained BERT and freeze it in the training procedure, and only finetune the LSTM module. Meanwhile, the number of trainable parameters is significantly reduced.

We then build a graph $\mathbb{G}_l$ based on the label embeddings, in which the set of nodes is $\mathcal{V} = \{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_C\}$, and $\mathbf{v}_i \in R^d$ represents the label embedding of the i-th class. We then calculate the cosine similarity between all pairs of label embeddings to build edges of the graph $\mathbb{G}_l$. For convenience, the graph $\mathbb{G}_l$ is formulated as a inter-class similarity (ICS) matrix $S_l \in R^{C \times C}$ as below.

$$S_l^{ij} = \frac{\mathbf{v}_i^T \mathbf{v}_j}{||\mathbf{v}_i||_2 \, ||\mathbf{v}_j||_2} \tag{2}$$

Then S_l^{ij} is regarded as a kind of similarity between two class embeddings $\mathbf{v}_i$ and $\mathbf{v}_j$. Larger S_l^{ij} indicates higher similarity between the classes i and j.

Eventually, we blend the two class relation graphs $\mathbb{G}_w$ and $\mathbb{G}_l$ generated using two kinds of semantic knowledge, and obtain a hybrid graph $\mathbb{G}_t$, formulated as below.

$$S_t = S_l + S_w \tag{3}$$

3.2 Visual Prototype Generation

This section introduce an effective visual prototype generation module for training robust CNNs with noisy images. The key idea of visual prototype module is to generate a clean visual prototype v_k, $k \in \{1, 2, ..C\}$ for each class. The visual prototype v_k can be interpreted as a reliable and effective representation of k-th class, and can be used to identify the reliability of all training data.

In order to generate visual prototype v_k, we need to obtain some reliable images from k-th class, and evaluate their contributions to v_k. Since noisy images is ubiquitous within each class, it is an intractable problem to directly collect reliable images. In this paper, we resort to the class relation graph $\mathbb{G}_t$ constructed in Sect. 3.1 to help this collection. Intuitively, the inter-class relation in visual representation space for reliable images should be closely related to that in class relation graph. For example, the k nearest classes of *siamese cat* in class relation graph are *persian cat*, *tiger cat*, *manx cat*, etc. If the k nearest classes of an image in visual representation space are also the same, then this image is probably reliable, and should contribute to the generation of visual prototype in a high confidence.

To be specific, we consider an image sample x_i and its current labelled class c. The semantic similarity vector of class c can be obtained from the class relation graph $\mathbb{G}_t$, and is denoted as a vector s_t^i of length C. To compute the visual similarity vector between x_i and all C classes, it is required to generate an initial prototype for each class first. We first extract visual features from all images using the CNN model in the proposed SINet. Then for each class, top-k ranked image features according to their classification confidence score are averaged to generate initial class prototype. Then the visual similarity vector

of x_i is computed as the cosine similarity score of the CNN feature g_i with all initial class prototypes, which is denoted a vector s_v^i of length C. Finally the consistence score p_i of image sample x_i is estimated based on KL divergence between s_t^i and s_v^i.

$$p_i = \frac{1}{(KL\left(\psi(s_t^i), \psi(s_v^i)\right) + \epsilon)^\gamma} \tag{4}$$

where ψ is a normalize function, e.g. $L2$ norm or softmax, γ is used to control the "contrast" of two similarity vectors, and ϵ is a small positive constant to prevent the denominator going to zero.

Eventually, we generate a visual prototype v_c for class c by using the weighted sum of the image features, as formulated in Eq. (5):

$$v_c = \frac{\sum_{i=1}^{N} g_i p_i}{\sum_{i=1}^{N} p_i} \tag{5}$$

where the g_i is the visual CNN feature of image x_i in the base model.

As the training proceeds, the visual prototype for each class will be updated iteratively, then more reliable samples could contribute to train the CNN model better.

3.3 Noise Weighting

In this section, we use the class prototypes generated above to weight noisily labelled images before training. Considering an image x_i and its current labelled class c, we estimate an importance weight $w_{i,c}$ by calculating the Euclidean distance of the visual feature g_i of image x_i and the prototype v_c. As formulated in Eq. (6), the importance weight $w_{i,c}$ is computed with two hyper-parameters α and β to control the shift and contrast of different visual features.

$$w_{i,c} = max\{0, [\alpha - ||v_c - g_i||_2]^\beta\} \tag{6}$$

We finally use a weighted cross entropy loss for model training as shown in Equ.(7).

$$\text{Loss}_{ce} = \sum_{i=1}^{N} \sum_{c=1}^{C} w_{i,c} \cdot log(p_{i,c}) \tag{7}$$

where $p_{i,c}$ is the softmax output of image x_i on class c.

3.4 Implementation Details and WebVision Challenge

Implementation Details. The scale of WebVision data is significantly larger than that of ImageNet, it is important to considering the computational cost when extensive experiments are conducted in evaluation and comparisons. In our experiments, we employ the resnext-101 as our standard architecture. The resnext-101 model is trained

by adopting the proposed SINet. The network weights are optimized with mini-batch stochastic gradient decent (SGD), where the batch size is set to 2,500. The learning rate starts from 0.1, and decreases by a factor of 10 at the epochs of 30, 60, 80, 90. The whole training process stop at 100 epochs. To reduce the risk of over-fitting, we use common data augmentation technologies which include random cropping, mirror flip and autoaugment. We also add a dropout operation with a ratio of 0.25 after the global pooling layer.

Topk Label Smoothing. Since there exists massive noise images in WebVision, if we directly utilize the one-hot target of ground truth to train CNN, it is inevitable to over-fit the noisy labels. To alleviate this problem, we proposed Adaptive Label Smoothing to assist the model training. Specifically, we select a small subset of high confidence images to train an initial model, and then we use the model to predict probability distribution of rest images. We use the topk predictions and ground truth to construct a smoothing label, and use this smoothing label to train the model. The Adaptive Label Smoothing enhance the tolerance of noisy labels, leading to about 0.2% performance improvements on top-5 accuracy in WebVision challenge.

Adaptive Spatial Resolution. There exists a lot of fine-grained categories in WebVision, which are hard to distinguish. Many studies have show that high-resolution images can improve the performance of fine-grained recognition. Inspired by this, we first train an initial model with fixed image resolution of 224×224, and then finetune the model with large input resolutions, e.g. 256×256 and 312×312. Specifically, the adaptive average pooling is used before the classifier layer to keep the feature dimension unchanged. The large input resolutions enhance the tolerance of noisy labels, leading to about 0.5% performance improvements on top-5 accuracy in WebVision challenge.

4 Experiments

In this section, we mainly evaluate our SINet on four popular benchmarks for noisy-labeled visual recognition, i.e., WebVision, ImageNet, Clothing1M and AliProducts. Particularly, we investigate the learning capability on large-scale web images without any human annotation.

4.1 Datasets

WebVision 1.0 [37] is an object-centric dataset, which is larger than ImageNet for object recognition and classification. The images are crawled from both Flickr and Google images search, by using queries generated from the 1, 000 semantic concepts of the WordNet. Meta information along with those web images (e.g., title, description, tags, etc.) are also crawled. The dataset contains 1,000 object categories, including 2.4 millions images in total, but without any human annotation. 50K images with human annotation are used as validation set, and another 50K images with human annotation for testing. The evaluation measure is based on top-5 accuracy, where each algorithm provides a list of at most 5 object categories to match the ground truth.

WebVision 2.0 is similar with WebVision 1.0 [37]. It also contains images crawled from the Flickr website and Google Images search. The number of visual concepts was extended from 1,000 to 5,000, and the total number of training images reaches 16 million. It includes massive noisy labels, as shown in Fig. 1. There are 290K images

Fig. 3. Image samples of AliProducts, (a) fine-grained categories: from top to bottom are respectively chips in different flavors , diaper with different sizes, formula with different stages, cosmetic with different functions, and each column is a fine-grained category; (b) noisy labeled images: images from coke zero, bbq-flavored chips, milk candy, soap. As can been see, each category includes massive noisy images.

with human annotation are used as validation set, and another 290K images with human annotation for testing. The evaluation measure is the same as WebVision 1.0.

ImageNet. [1] is an image classification dataset, which contains 1000 classes. The original dataset has been splitted into 1.28 million training images, 50k validation images. In this paper, we randomly select 40% training images for each category and assign them with class label uniformly sampled from the rest categories. The generate new dataset thus have lot of noises which could be used to evaluate the effectiveness of popular algorithms on noisy image classification.

Clothing1M. [36] is a large-scale fashion dataset, which includes 14 clothes categories. It contains 1 million noise label images and 74,000 manually annotated images. We call the annotated images as clean set, which is divided into training data,validation data and testing data, with numbers of 50,000,14,000, and 10,000 images, respectively. There are some images overlap between the clean set and the noisy set. The dataset was designed for learning robust models from noisy data without human supervision.

AliProducts[1] is a large-scale noisy and fine-grained product dataset, which includes 50,000 categories. The images are crawled from image search engine and other web sources by using 50,000 product SKU names. The dataset covers foods, snacks, drinks, cosmetics and other daily products and the categories are in SKU (Stock Keeping Unit) level and specific to flavor, capacity, function or even the batch of the production. Therefore, some of the categories might have great difficulty in visual distinguishing due to the fine-grained attribute. AliProducts contains 2.5 million training images without any human annotation, consequently contain massive noisy labels, as show in Fig. 3. Totally 148K manually annotated images are used as validation set, and another 250K manually annotated images for testing. In addition, we released side information (e.g., hierarchical relationships between classes) concerning these image data, which could be exploited to learn better representations and models. The main difference

[1] https://tianchi.aliyun.com/competition/entrance/231780/information.

between AliProducts and other noisy datasets (e.g., Clothing 1M and WebVision) is that AliProducts contains massive fine-grained and real-world noisy images, which is relatively difficult for robust DNN methods to improve.

Table 1. Top1/Top5 accuracy of three different models with ResNext-101 architecture on validation set of WebVision.

Method	Model-A	Model-B	Model-C
Top1	51.05%	47.81%	55.57%
Top5	74.94%	72.08%	78.34%

4.2 Experiments on WebVision 2.0

In this subsection, we conduct extensive experiments on WebVision 2.0 dataset to evaluate and demonstrate the effectiveness of proposed SINet. All experiments are implemented using ResNext-101 backbone if there is no special instructions.

Training Strategy and Comparison. We conduct three training strategies with a standard ResNext-101 architecture, resulting in three models, which are described as follow.

Model-A. The model was trained by directly using all the training data.

Model-B. The model was trained by using the high-confidence images without reweighting in training loss.

Model-C. The model was trained with proposed training strategy, where the confidence score is multiplied on the loss of corresponding image for reweighting.

The top1/top5 results of three models on the validation set of WebVision are reported in Table 1. The result shows Model-A with all training data significantly outperforms the Model-B with subset of clean data, with improvements of 3.24%/2.86% for top1/top5 accuracy. This is due to that it is hard to distinguish all the clean labeled samples from those images with heavy noises. In addition, Model-C with our proposed method significantly outperforms Model-A, with improvements of 4.52%/3.40% on top1/top5 accuracy. It is obviously that our proposed method could better explore those clean labels from those noise samples. These improvements are significant on WebVision Challenge with such a large scale noisy dataset, which demonstrate the effectiveness of our method.

Class Relation Graph. We investigate different ways for constructing Class Relation Graph. (I) Category Name: We use category name, i.e., cat, lion, apple, with BERT model to extract the word embeddings, then use the similarity comparison of word embeddings to construct the Class Relation Graph. (II) Category Description: We use category descriptions in WordNet, i.e., *Siamese cat: A slender short-haired blue-eyed breed of cat having a pale coat with dark ears paws face and tail tip*, with BERT and LSTM to extract the textual embeddings, then use the similarity comparison of textual embeddings to construct the Class Relation Graph. (III) Hierachical WordNet: We directly use the prior knowledge of Hierachical WordNet based on the shortest path

Table 2. Class Relation Graph construction with different strategies. (I) Category Name (CN) (II) Category Description (CD) (III) Hierachical WordNet (HW)

Strategy	Top1	Top5
CN	54.18%	76.84%
CD	55.63%	78.15%
HW	55.71%	78.42%
CN+HW	55.79%	78.46%
CD+HW	**55.98%**	**78.62%**

Table 3. Visual Prototype Generation with different strategies. (I) with Reweighting: Topk matched images with matching score as weighting coefficient (II) without Reweighting: Topk matched images averaged

Strategy	Top1	Top5
Constant	55.68%	78.45%
Weighting	**55.98%**	**78.62%**

Table 4. The effect of Shift factor α on the performance of WebVision validation set.

α	Top1	Top5
0.8	53.25%	76.24%
1.0	55.43%	78.31%
1.2	**55.98%**	**78.62%**
1.4	55.76%	78.48%

Table 5. The effect of Contrast factor β on the performance of WebVision validation set.

β	Top1	Top5
1.0	55.68%	78.45%
1.5	**55.98%**	**78.62%**
2.0	55.82%	78.56%
2.5	55.35%	78.24%

between two category to establish the Class Relation Graph. Experimental results of using these three types of class relation graph are shown in Table 2. Obviously, by introducing these side information, the performance could improve a lot than original Model-A, which shows the effectiveness of proposed class relation graphs. Also, these three types of class relation graph are complementary, and combining of them could also boost the performance.

Visual Prototype. We investigate different ways for visual prototype generation. (I) Constant: We does not use any weight operations for the images in Visual Prototype candidates. In this case, the visual prototype representation in Eq. (5) is reduced as the mean of candidates representations. (II) Weighting: We use the method described in Eq. (5) as the weighting operation. Since p_i is the importance score, we use the soft weighted representation as the visual prototype. Experiments of using two types of strategies are shown in Table 3. It shows that when paying more attention on top-ranked images, the generated visual prototypes are better then simple average all the feature representations.

Noise Weighting. In this section, we conduct ablation analysis on the hyper-parameters for our proposed noisy weighting method, and discuss how they affect the recognition performance. (I) Shift factor α: α controls the amount of noisy data actually participating in the model training, the weight of images whose Euclid distance larger than α is 0, that is equivalent to deleting them from the training set, and only use the images with Euclid distance smaller than α to train the model. (II) Contrast factor β: We introduce the contrast parameter β to sharpen the differences of

the scores. Experiments of different α and β are shown in Table 4 and 5, respectively. Typically, $\alpha = 1.2$ could keep most of cleaning samples. $\beta = 1.5$ is a proper value to map the score to sampling weight and handle the noise data.

Final Results on the WebVision Challenge. We further evaluate the performance of our proposed SINet with various networks architectures, including ResNext101, SE-ResNext101, SE-Net154. Results are reported in Table 6. As can be found, SE-Net154 substantially performs ResNext101 and SE-ResNext101 on WebVision validation set, with top1/top5 improvements of 1.31%/1.46% and 1.30%/1.27%, while SE-ResNext101 and ResNext101 has similar performance with a marginal performance gain obtained. Our final results were obtained with ensemble of five models. We had the best performance at a Top 5 accuracy of 82.54% on the WebVision challenge 2019.

Table 6. Performance of SINet with various networks on WebVision validation set.

Method	ResNext101	SE-ResNext101	SENet154
Top1	55.56%	55.57%	56.87%
Top5	78.15%	78.34%	79.61%

4.3 Comparisons with the State-of-the-Art Methods

To further explore the effectiveness of our proposed SINet, we conduct extensive comparisons with recent state-of-the-art approaches developed specifically for learning from noisy labels, such as CleanNet, MetaCleaner and MentorNet. For fairness, our comparisons are based on the same CNN backbone, i.e., ResNet50.

WebVision1.0 and ImageNet. We evaluate our SINet on ImageNet, by adding 40% noise ratio with uniform flip. The top-1 accuracy is 66.47/69.12 for ResNet50 without/with SINet. It further shows the power of SINet for large-scale noisy image recognition. By following [16], we use the training set of WebVision1.0 to train the models, and test on the validation sets of WebVision1.0 and ImageNet. Both of them has same 1000 categories. Full results are presented in Table 7. SINet improves the performance of our baseline significantly, and our results compare favorably against recent CurriculumNet, CleanNet and MentorNet with consistent improvements.

Table 7. Comparisons on Webvision1.0 and ImageNet. The models are trained on WebVision1.0 training set and tested on WebVision1.0 and ImageNet validation sets.

Method	WebVision1.0	ImageNet
	Top1/Top5	Top1/Top5
Baseline [11]	67.8(85.8)	58.9(79.8)
CleanNet [11]	70.3(87.8)	63.4(84.6)
MentorNet [10]	70.8(88.0)	62.5(83.0)
CurriculumNet [16]	72.1(89.2)	64.8(84.9)
Our Baseline	69.9(87.4)	63.2(83.8)
SINet	**73.8(90.6)**	**66.8(85.9)**

Clothing1M. For Clothing1M, we consider the state of the art results in [21], which use both noisy and clean set to train the model. Following [21], we conduct two experiments. First we use the 25k clean set to construct the visual prototype and apply noise weighting to one million noisy data, and then use the images with confidence score to train the model. Second, we conduct the same experiment, but with all the clean training set (50k). As shown in Table 8, our SINet outperforms CleanNet, MetaCleaner and DeepSelf, which demonstrates its effectiveness.

Table 8. Experimental results on Clothing1M. Clean set is used in CleanNet [11] to obtain the validation set. To keep same data setting, we use the 25k clean images to construct the visual prototype and use 1M noisy training set with confidence scores to train our SINet, and then fine-tune it on 25k clean images. Furthermore, we achieve the state-of-the-art performance on the setting of Noise1M+Clean(50k), which illustrate the robustness of our SINet on noisy label recognition.

Noise1M+Clean(25k)	Method	CleanNet [11]	MetaCleaner [21]	DeepSelf [20]	Ours
	Accuray	74.69	76.00	76.44	**77.26**
Noise1M+Clean(50k)	Method	CleanNet [11]	MetaCleaner [21]	DeepSelf [20]	Ours
	Accuray	79.9	80.78	81.16	**81.32**

AliProducts. The existed benchmarks with noisy labels are relatively small in the scale of categories or images. To further explore the effectiveness of SINet, we conduct experiments on our released AliProducts, which is a large-scale product dataset with noisy labels and the hierarchical category relations is also provided. We use the hierarchical category relations to construct the class relation graph, and then combine it with images to construct the visual prototype. Finally, we use the images with confidence scores to train the model. As shown in Table 9, our SINet outperforms all other approaches, which illustrates that SINet is more robust to noisy labels.

Table 9. Comparison with the state-of-the-art on AliProducts dataset.

Method	Baseline	CurriculumNet [16]	CleanNet [11]	MetaCleaner [21]	Ours
Accuray	85.35%	85.69%	86.13%	85.92%	**86.29%**

5 Conclusions

In this paper, we presented a novel method, which can learn to generate a visual prototype for each category, for training deep CNNs with large-scale real-world noisy labels. It mainly consists of two submodules. The first module, Visual Prototype can generate a clean representation from the noisy images for every category by integrate the noisy images with side information. The second module, namely Noise Weighting, can estimate the confidence scores of all the noisy images and rank images with confidence scores by analyzing their deep features and Visual Prototype. Via SINet, we can

train a high-performance CNN model, where the negative impact of noisy labels can be reduced substantially. We conduct extensive experiments on WebVision, ImageNet, Clothing1M, as well as collected AliProducts, where it achieves state-of-the-art performance on all benchmarks. Future work could aim to train an end-to-end DNNs with Side Information to handle the noisy label recognition.

References

1. Deng, J., Dong, W., Socher, R., Li, J., Li, K., Li, F.: Imagenet: a large-scale hierarchical image database. In: CVPR (2009)
2. Nettleton, D., Orriols, P., Fornells, A.: A study of the effect of different types of noise on the precision of supervised learning techniques. Artif. Intell. Rev. **33**(4), 275–306 (2010)
3. Pechenizkiy, M., Tsymbal, A., Puuronen, S., Pechenizkiy, O.: Class noise and supervised learning in medical domains: the effect of feature extraction. In: IEEE Symposium on Computer-Based Medical Systems (CBMS), pp. 708–713 (2006)
4. Brooks, J.: Support vector machines with the ramp loss and the hard margin loss. Oper. Res. **59**(2), 467–479 (2011)
5. Ghosh, A., Kumar, H., Sastry, P.: Robust loss functions under label noise for deep neural networks. In: AAAI (2017)
6. Ghosh, A., Manwani, N., Sastry, P.: Making risk minimization tolerant to label noise. Neurocomputing **160**, 93–107 (2015)
7. Shirazi, H., Vasconcelos, N.: On the design of loss functions for classification: theory, robustness to outliers, and savageboost. In: NeurIPS (2009)
8. Rooyen, B., Menon, A., CWilliamson, R.: Learning with symmetric label noise: the importance of being unhinged. In: NeurIPS (2015)
9. Zhang, Z., Sabuncu, M.: Generalized cross entropy loss for training deep neural networks with noisy labels. In: NeurIPS (2018)
10. Jiang, L., Zhou, Z., Leung, T., Li, T., Li, F.: Mentornet: regularizing very deep neural networks on corrupted labels. arXiv preprint arXiv:1712.05055 (2017)
11. Lee, K., He, X., Zhang, L., Yang, L.: Cleannet: transfer learning for scalable image classifier training with label noise. arXiv preprint arXiv:1711.07131 (2017)
12. Ren, M., Zeng, W., Yang, B., Urtasun, R.: Learning to reweight examples for robust deep learning. arXiv preprint arXiv:1803.09050 (2018)
13. Wang, Y., et al.: Iterative learning with open-set noisy labels. In: CVPR (2018)
14. Xue, C., Dou, Q., Shi, X., Chen, H., Heng, P.: Robust learning at noisy labeled medical images: applied to skin lesion classification. arxiv.org (2019)
15. Kriegel, H., Kroger, P., Schubert, E., Zimek, A.: Loop: local outlier probabilities. In: CIKM (2009)
16. Guo, S., et al.: CurriculumNet: weakly supervised learning from large-scale web images. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11214, pp. 139–154. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01249-6_9
17. Reed, S., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., Rabinovich, A.: Training deep neural networks on noisy labels with bootstrapping. arXiv preprint arXiv:1412.6596 (2014)
18. Tanaka, D., Ikami, D., Yamasaki, T., Aizawa, K.: Joint optimization framework for learning with noisy labels. In: CVPR (2018)
19. Li, J., Wong, Y., Zhao, Q., Kankanhalli, M.: Learning to learn from noisy labeled data. In: CVPR (2019)

20. Han, J., Luo, P., Wang, X.: Deep self-learning from noisy labels. arXiv preprint arXiv:1908.02160 (2019)
21. Zhang, W., Wang, Y., Qiao, Y.: MetaCleaner: learning to hallucinate clean representations for noisy-labeled visual recognition. In: CVPR (2019)
22. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
23. Zhu, X., Wu, X.: Class noise vs attribute noise: a quantitative study. Artif. Intell. Rev. **22**(3), 177–210 (2004)
24. Simonyan K., Zisserman A.: Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556 (2014)
25. Szegedy, C., et al.: Going deeper with convolutions. arXiv:1409.4842 (2014)
26. Sukhbaatar, S., Bruna, J., Paluri, M., Bourdev, L., Fergus, R.: Training convolutional networks with noisy labels. arXiv preprint arXiv:1406.2080 (2014)
27. Goldberger, J., Reuven, E.: Training deep neural-networks using a noise adaptation layer. In: ICLR (2017)
28. Patrini, G., Rozza, A., Menon, A., Nock, R., Qu, L.: Making deep neural networks robust to label noise: a loss correction approach. In: CVPR (2017)
29. Hendrycks, D., Mazeika, M., Wilson, D., Gimpel, K.: Using trusted data to train deep networks on labels corrupted by severe noise. In: NeurIPS (2018)
30. Zhang Z., Sabuncu, M.: Generalized cross entropy loss for training deep neural networks with noisy labels. In: NeurIPS (2018)
31. Li, Y., Yang, J., Song, Y., Cao, L., Luo, J., Li, L.: Learning from noisy labels with distillation. In: CVPR (2017)
32. Veit, A., Alldrin, N., Chechik, G., Krasin, I., Gupta, A., Belongie, S.: Learning from noisy large-scale datasets with minimal supervision. In: CVPR (2017)
33. Brodley, C., Friedl, M.: Identifying mislabeled training data. arXiv:1106.0219 (2011)
34. Miranda, A., Garcia, L., Carvalho A., Lorena, A.: Use of classification algorithms in noise detection and elimination. In: HAIS (2009)
35. Barandela, R., Gasca, E.: Decontamination of training samples for supervised pattern recognition methods. In: ICAPR (2000)
36. Xiao, T., Xia, T., Yang, Y., Huang, C., Wang, X.: Learning from massive noisy labeled data for image classification. In: CVPR (2015)
37. Li, W., Wang, L., Li, W., Agustsson, E., Gool, L.: Webvision database: visual learning and understanding from web data. CoRR abs/1708.02862 (2017)
38. Alexander, B., Denzler, J.: Not just a matter of semantics: the relationship between visual and semantic similarity. In: German Conference on Pattern Recognition (2019)
39. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: ACL (2019)
40. Miller, G.A.: WordNet: a lexical database for English. Commun. ACM **38**(11), 39–41 (1995)