Holistic Label Correction for Noisy Multi-Label Classification

Xiaobo Xia[1], Jiankang Deng[2], Wei Bao[1],
Yuxuan Du[3], Bo Han[4], Shiguang Shan[5,6], Tongliang Liu[1*]
[1]The University of Sydney [2]Imperial College London
[3]JD Explore Academy [4]Hong Kong Baptist University
[5]Chinese Academy of Sciences [6]University of Chinese Academy of Sciences

Abstract

Multi-label classification aims to learn classification models from instances associated with multiple labels. It is pivotal to learn and utilize the label dependence among multiple labels in multi-label classification. As a result of today's big and complex data, noisy labels are inevitable, making it looming to target multi-label classification with noisy labels. Although the importance of label dependence has been shown in multi-label classification with clean labels, it is challenging and hard to bring label dependence to the problem of multi-label classification with noisy labels. The issues are, that we do not understand why label dependence is helpful in the problem, and how to learn and utilize label dependence only using training data with noisy multiple labels. In this paper, we bring label dependence to tackle the problem of multi-label classification with noisy labels. Specifically, we first provide a high-level understanding of why label dependence helps distinguish the examples with clean/noisy multiple labels. Benefiting from the memorization effect in handling noisy labels, a novel algorithm is then proposed to learn the label dependence by only employing training data with noisy multiple labels, and utilize the learned dependence to help correct noisy multiple labels to clean ones. We prove that the use of label dependence could bring a higher success rate for recovering correct multiple labels. Empirical evaluations justify our claims and demonstrate the superiority of our algorithm.

1. Introduction

Multi-label classification assigns a set of *multiple labels* for each instance [71]. As a practical learning paradigm, multi-label classification has been widely applied in various domains, ranging from computer vision [7] and natural language processing [41], to recommendation systems [69] and bioinformatics [8]. Consensually, compared with multi-class classification [20, 23, 24], where each instance is assigned with a single label, multi-label classification is more challenging [35]. Plenty of advanced methods are proposed in recent years for multi-label classification [77, 45, 14, 74, 37, 9, 19, 64].

The great majority of the methods assume that training data are annotated precisely. However, noisy labels are *inevitable* in multi-label classification [36], especially for classification with big and complex data. They may be resulted by unintentional mistakes of manual and automatic annotators [51, 75, 13], or intentional corruptions on clean labels [50, 44]. Noisy labels severely impair the generalization of learned models, *over-parameterized deep models* in particular [26, 61, 58, 59, 55, 56]. A straightforward way to address the problem of multi-label classification with noisy labels is to treat each label *in isolation* and convert the multi-label problem into a number of binary classification problems. Afterward, the methods in multi-class classification with noisy labels [16, 47] are applied to train *independent* binary classifiers, which capture instance-label dependence robustly to strengthen classification. This way is a remedy to handle noisy labels, but ignores the label dependence among multiple labels. It is essential to learn and utilize the label dependence in multi-label classification [70, 18, 11, 31].

Prior works [65, 6, 52] illustrate the successes of considering the label dependence among multiple labels in multi-label classification with clean labels. In different ways, *e.g.*, helping learn *inter-dependent classifiers* [7], the label dependence can be used to boost the learning of the instance-label dependence, which improves final classification. Inspired by the successes, it is concerned that label dependence could be exploited to handle the problem of multi-label classification with noisy labels. However, there are few attempts before for this important problem. At least *three questions* make the solution remain mysterious. First, in intuition, we need to understand why label dependence is helpful for the problem. Second, in technique, we need to know how to learn and utilize the label dependence in the

*Corresponding author (tongliang.liu@sydney.edu.au).

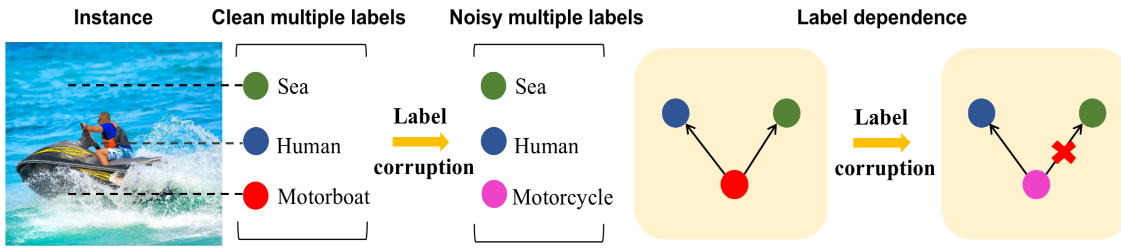

Figure 1. The illustration of why the label dependence among multiple labels helps distinguish the examples with noisy/clean multiple labels. The arrow presents the label dependence between a label pair. For the labels "a" and "b", "a $\rightarrow$ b" means that, when "a" appears, "b" will also occur with high probability. The example comes from a web search. The set of clean multiple labels is {Sea, Human, Motorboat}, where the label dependence is strong with both "Motorboat $\rightarrow$ Sea" and "Motorboat $\rightarrow$ Human". However, due to label corruption, Motorboat is flipped to be Motorcycle, which causes "Motorcycle $\nrightarrow$ Sea". Therefore, the label dependence among noisy multiple labels is *weaker* than the label dependence among corresponding clean ones.

problem. As we only have training data with noisy labels, both the accurate catch and application of the label dependence are challenging. Third, in verification, we need to know what improvements the label dependence can bring.

In this paper, we answer the three questions one by one. The first answer is illustrated in Figure 1. That is, compared with noisy multiple labels, the label dependence among clean multiple labels is *stronger with high probability*. Therefore, such dependence could help distinguish the examples with noisy/clean multiple labels for our problem. The second answer is given by the proposed holistic correction for multi-label classification with noisy labels (*aka* HLC). Specifically, HLC inherits the *memorization effect* in handling noisy labels [1, 25, 53]: the deep model would firstly memorize the training examples with clean labels, leading to reliable model predictions in early training. In HLC, the label dependence is learned by a dynamic graph [65], and then applied to correcting noisy multiple labels. In more detail, the *holistic score* in HLC is proposed to measure the instance-label and label dependencies in an example. The stronger instance-label and label dependencies make a larger holistic score. We compare the ratio between the holistic scores of the example with noisy multiple labels and its variant with predicted multiple labels, with an easily determined threshold. The noisy multiple labels are corrected or changeless based on the comparison result. Benefiting from the memorization effect, both dependence learning and multi-label correction are useful. Besides, they fulfill *a positive cycle* [3]. Namely, better dependence learning results in a better multi-label correction, and better multi-label correction makes better dependence learning, leading to final enhanced classification.

The third answer is given by both theoretical analyses and empirical evaluations. Theoretically, we show that the additional use of label dependence brings a higher probability to handle noisy multiple labels successfully than the sole use of instance-label dependence under some conditions. Empirically, we demonstrate the power of label dependence through experiments and show that, in most situations, HLC outperforms comparison methods with large margins.

The contributions of this paper are summarized as follows. (1) We focus on a realistic problem of multi-label classification with noisy labels. The challenges of using label dependency to address the problem are carefully analyzed, which benefits future research on the problem. (2) We propose an effective method to handle noisy labels in multi-label classification. The method measures simultaneous instance-label and label dependencies in an example for follow-up label correction. (3) Theoretical analysis is provided to explain the success rate of the proposed method. Besides, we confirm that the use of label dependence is indeed powerful under some conditions. (4) Extensive empirical results on multiple benchmarks demonstrate the superiority of our method. Detailed ablation studies and discussions are also provided. Codes are attached in the supplementary material.

2. Preliminaries

Problem setup. Let $\mathcal{X} \in \mathbb{R}^d$ denote the input space and $\mathcal{Y} \in \{l_1, \cdots, l_q\}$ denote the label space with q class labels. An example with multiple labels is denoted as $(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x} \in \mathcal{X}$ is the feature vector of an instance, and $\boldsymbol{y} \subseteq \mathcal{Y}$ is its set of associated labels. Denote the size of the label set $\boldsymbol{y}$ as $|\boldsymbol{y}|$. For the feature vector $\boldsymbol{x}$, its label set $\boldsymbol{y}$ may be corrupted and is flipped into $\bar{\boldsymbol{y}} \subseteq \mathcal{Y}$. We utilize a class-dependent noise transition matrix $\boldsymbol{T}$ [43, 46, 34, 60] to characterize the label flip process. Formally, for any $i \neq j$, $T_{ij} = \mathbb{P}(l_j \in \bar{\boldsymbol{y}} \wedge l_i \notin \bar{\boldsymbol{y}} | l_j \notin \boldsymbol{y} \wedge l_i \in \boldsymbol{y})$ represents the probability of the i-th class label to be flipped into the j-th class label. Consider a noisy multi-label dataset comprising several examples $(\boldsymbol{x}, \bar{\boldsymbol{y}})$. The aim is to learn a classification model *robustly* by *only* using the noisy dataset. Given an instance in testing, with the learned model, we can predict its relevant label set precisely.

Note that some works employ another problem setting that the total number of multiple labels can be changed after label flipping, which is referred to as multi-label classification with *missing* or *redundant* labels. For the former setting, it is not accurate to consider it as a classification with noisy labels, since all annotated labels are correct [67, 57]. For the latter setting, it is normally called partial multi-label learning [62], which is different from the problem setting of this paper, as detailed in Appendix C.4. Our setting, *i.e.*, the total number of labels is preserved after label flipping, is realistic. In many practical situations, it is easy to determine the number of objects in an image, in particular with object detection techniques. In contrast, it can be harder to annotate the objects perfectly, resulting in noisy labels. In addition, in Section 4.4, we will show that our problem setting well fits the realistic situation. That is to say, the proposed method can achieve superior performance on a realistic noisy multi-label dataset.

Preparation technology. As discussed, we need both the instance-label dependence and the label dependence among multiple labels. Given an example $(\boldsymbol{x}, \boldsymbol{y})$, for the instance-label dependence, it can be learned with the conditional probability of $l_i \in \boldsymbol{y}$ given $\boldsymbol{x}$ according to model's probability outputs. For the label dependence among multiple labels, it is often estimated by counting the occurrence of label pairs in training data [7].

Recently, the graph convolutional network (GCN) is used in multi-label classification and achieves great successes [7, 65, 6]. The advantage of the GCN-based methods is that they can capture the instance-label and label dependencies simultaneously during training. In this paper, we inherit the advantage of the GCN-based methods and build HLC based on ADDGCN [65]. ADDGCN designs a semantic attention module (SAM) to estimate the content-aware class-label representations for each class from the extracted feature map. The representations are fed into a GCN module (GCNM) for final classification. We provide the technical details of ADDGCN [65] in Appendix C.1. Before delving into the next section, readers only need to remember that the instance-label and label dependencies can be learned during training. Note that we also review prior works on multi-class classification with noisy labels and multi-label classification with clean/noisy labels in Appendix C.2 and Appendix C.3.

3. Proposed Method

3.1. Holistic Judgment in Multi-Label Classification

Holistic score. We begin with an example with clean multiple labels. Given an example $(\boldsymbol{x}, \boldsymbol{y})$, we can measure the instance-label dependence S^f, and the label dependence S^l. Denote the variable of clean multiple labels by $\boldsymbol{Y}$. Mathematically, we define two dependencies as $S^f_{\boldsymbol{z}}(\boldsymbol{x}) := \sum_{\{\boldsymbol{Y}=\boldsymbol{z}, l_i \in \boldsymbol{z}\}} \mathbb{P}(l_i|\boldsymbol{x})$ and $S^l_{\boldsymbol{z}}(\boldsymbol{x}) := \sum_{\{\boldsymbol{Y}=\boldsymbol{z}, l_i, l_j \in \boldsymbol{z}\}} \frac{1}{2}\left[\mathbb{P}(l_j|l_i, \boldsymbol{x}) + \mathbb{P}(l_i|l_j, \boldsymbol{x})\right]$, where $\boldsymbol{z}$ is the value of the random variable $\boldsymbol{Y}$. The holistic score of the example $(\boldsymbol{x}, \boldsymbol{y})$ considers two dependencies at the same time. Formally, we denote the holistic score of $(\boldsymbol{x}, \boldsymbol{y})$ as $S_{\boldsymbol{y}}(\boldsymbol{x})$ and define it as

$$S_{\boldsymbol{y}}(\boldsymbol{x}) := S^f_{\boldsymbol{y}}(\boldsymbol{x}) + S^l_{\boldsymbol{y}}(\boldsymbol{x}). \tag{1}$$

Afterward, denote the variable of noisy multiple labels by $\bar{\boldsymbol{Y}}$. For the example with noisy multiple labels, *i.e.*, $(\boldsymbol{x}, \bar{\boldsymbol{y}})$, the instance-label dependence and label dependence are measure by $\bar{S}^f_{\boldsymbol{z}}(\boldsymbol{x}) := \sum_{\{\bar{\boldsymbol{Y}}=\boldsymbol{z}, l_i \in \boldsymbol{z}\}} \mathbb{P}(l_i|\boldsymbol{x})$ and $\bar{S}^l_{\boldsymbol{z}}(\boldsymbol{x}) := \sum_{\{\bar{\boldsymbol{Y}}=\boldsymbol{z}, l_i, l_j \in \boldsymbol{z}\}} \frac{1}{2}\left[\mathbb{P}(l_j|l_i, \boldsymbol{x}) + \mathbb{P}(l_i|l_j, \boldsymbol{x})\right]$. Accordingly, the holistic score of the example $(\boldsymbol{x}, \bar{\boldsymbol{y}})$ is denoted by $\bar{S}_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$, which is defined as $\bar{S}_{\bar{\boldsymbol{y}}}(\boldsymbol{x}) := \bar{S}^f_{\bar{\boldsymbol{y}}}(\boldsymbol{x}) + \bar{S}^l_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$. Note that, during training, we cannot access $\bar{S}^f_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$ and $\bar{S}^l_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$. Instead, the estimated posterior probabilities are used. We denote the estimations of $\bar{S}^f_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$ and $\bar{S}^l_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$ as $\hat{\bar{S}}^f_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$ and $\hat{\bar{S}}^l_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$. The estimation of the holistic score is $\hat{\bar{S}}_{\bar{\boldsymbol{y}}}(\boldsymbol{x}) = \hat{\bar{S}}^f_{\bar{\boldsymbol{y}}}(\boldsymbol{x}) + \hat{\bar{S}}^l_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$. With preparation technology discussed in Section 2 and Appendix C.1, $\hat{\bar{S}}^f_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$ and $\hat{\bar{S}}^l_{\bar{\boldsymbol{y}}}(\boldsymbol{x})$ can be obtained.

Algorithm 1: Holistic Correction.

Input: $(\boldsymbol{x}, \bar{\boldsymbol{y}})$, h, and $\hat{\delta}$.
Output: $\bar{\boldsymbol{y}}_{new}$.
1: $\boldsymbol{y}^* = h(\boldsymbol{x})$;
2: $\kappa(h, \boldsymbol{x}, \bar{\boldsymbol{y}}) = \hat{\bar{S}}_{\bar{\boldsymbol{y}}}(\boldsymbol{x}) / \hat{\bar{S}}_{\boldsymbol{y}^*}(\boldsymbol{x})$;
3: **if** $\kappa(h, \boldsymbol{x}, \bar{\boldsymbol{y}}) \leq \hat{\delta}$ **then** $\bar{\boldsymbol{y}}_{new} = \boldsymbol{y}^*$;
4: **else** $\bar{\boldsymbol{y}}_{new} = \bar{\boldsymbol{y}}$.

Holistic correction. For the example $(\boldsymbol{x}, \bar{\boldsymbol{y}})$, we feed it into the deep network h included in ADDGCN [65]. The memorization effect in handling noisy labels [25, 33] shows that the deep network would first memorize the training data with clean labels and then the training data with noisy labels. Therefore, early in training, the outputs of the deep network are relatively reliable and can be used for label correction. For $(\boldsymbol{x}, \bar{\boldsymbol{y}})$, we denote its set of predicted multiple labels as $\boldsymbol{y}^*$. Here, the set of predicted labels is obtained with the top $|\bar{\boldsymbol{y}}|$ predictions based on the model's probability outputs.

Recall that the holistic score of an example holistically measures the instance-label dependence and label dependence among multiple labels simultaneously. From both human and machine cognition, if an example is annotated accurately, both dependencies should be strong [70, 18, 68, 28, 7] with high probability. Namely, the holistic score is large. We propose to check the ratio between the holistic score on $(\boldsymbol{x}, \bar{\boldsymbol{y}})$ and holistic score on $(\boldsymbol{x}, \boldsymbol{y}^*)$. Specifically, we check

$$\kappa(h, \boldsymbol{x}, \bar{\boldsymbol{y}}) = \hat{\bar{S}}_{\bar{\boldsymbol{y}}}(\boldsymbol{x}) / \hat{\bar{S}}_{\boldsymbol{y}^*}(\boldsymbol{x}). \quad (2)$$

We compare this ratio with a predetermined threshold $\hat{\delta}$. The value of $\hat{\delta}$ is given in the next subsection. If $\kappa(h, \boldsymbol{x}, \bar{\boldsymbol{y}}) \leq \hat{\delta}$, we flip the labels $\bar{\boldsymbol{y}}_{new} = \boldsymbol{y}^*$. Otherwise, the labels remain unchanged with $\bar{\boldsymbol{y}}_{new} = \bar{\boldsymbol{y}}$. The detailed algorithm of holistic correction for multi-label classification with noisy labels (*aka* HLC) is provided in Algorithm 1. After holistic correction for noisy labels, we use $(\boldsymbol{x}, \bar{\boldsymbol{y}}_{new})$ to train the deep network h based on ADDGCN [65].

3.2. Theoretical Insights

We extend the Tsybakov condition [75, 2, 15, 49] from multi-class classification to multi-label classification. Specifically, denote by $\boldsymbol{a_x}$ the label set predicted based on $S^f(\boldsymbol{x})$ with $\boldsymbol{a_x} := h^*(\boldsymbol{x}) = \arg\max_{\boldsymbol{z}} S^f_{\boldsymbol{z}}(\boldsymbol{x})$. Besides, denote by $\boldsymbol{b_x}$ the second best prediction with $\boldsymbol{b_x} := \arg\max_{\boldsymbol{z} \neq \boldsymbol{a_x}} S^f_{\boldsymbol{z}}(\boldsymbol{x})$. The maximum length of a label set is denoted as m ($m \ll q$). In this paper, we call the predicted label set by the Bayes optimal classifier for an instance as the correct label set. We present the Tsybakov condition on instance-label (abbreviated as ins.-label here) dependence and holistic Tsybakov condition as follows.

Definition 1 (Tsybakov condition on ins.-label dependence) *$\exists C_1, \lambda_1 > 0$ and $\exists t_0 \in (0, m]$, such that for all $t \leq t_0$, we have*

$$\mathbb{P}[S^f_{\boldsymbol{a_x}}(\boldsymbol{x}) - S^f_{\boldsymbol{b_x}}(\boldsymbol{x}) \leq t] \leq C_1 t^{\lambda_1}. \quad (3)$$

Definition 2 (Holistic Tsybakov condition) *$\exists C_2, \lambda_2 > 0$, and $\exists t_0 \in (0, m]$, such that for all $t \leq t_0$, we have*

$$\mathbb{P}[S_{\boldsymbol{a_x}}(\boldsymbol{x}) - S_{\boldsymbol{b_x}}(\boldsymbol{x}) \leq t] \leq C_2 t^{\lambda_2}. \quad (4)$$

Remark 1 *Definition 1 stipulates that the uncertainty of S^f is bounded. The margin region that is close to the decision boundary has a bounded volume. Definition 2 shares the similar idea and bound the uncertainty of S.*

Theorem 1 *Suppose $S(\boldsymbol{x})$ fulfills the holistic Tsybakov condition for constants C_2, $\lambda_2 > 0$, and $t_0 \in (0, m]$. We define $\epsilon := \max_{\boldsymbol{x},\boldsymbol{z}} \left[|\hat{\bar{S}}^f_{\boldsymbol{z}}(\boldsymbol{x}) - \bar{S}^f_{\boldsymbol{z}}(\boldsymbol{x})|, |\hat{\bar{S}}^l_{\boldsymbol{z}}(\boldsymbol{x}) - \bar{S}^l_{\boldsymbol{z}}(\boldsymbol{x})|, |\bar{S}^l_{\boldsymbol{z}}(\boldsymbol{x}) - S^l_{\boldsymbol{z}}(\boldsymbol{x})| \right]$ and $\tau := \min_i T_{ii}$. We analyze two cases:*
(1) If $\bar{\boldsymbol{y}}$ is corrected by $\kappa(h, \boldsymbol{x}, \bar{\boldsymbol{y}})$ with $\hat{\delta}$, let $\delta_1 = \min \left[\frac{\tau S_{\boldsymbol{b_x}}(\boldsymbol{x}) + \sum_{l_j \in \bar{\boldsymbol{y}}} \sum_{i \neq j} T_{ij} \mathbb{P}(l_i|\boldsymbol{x})}{\hat{\bar{S}}_{\boldsymbol{y}^*}(\boldsymbol{x})} \right]$ and $\rho_1 := |\hat{\delta} - \delta_1|$. Assume that $\epsilon \leq \frac{t_0 \tau - \rho_1 m}{3}$. Then, $\mathbb{P}[\bar{\boldsymbol{y}}_{new} = h^*(\boldsymbol{x}), \bar{\boldsymbol{y}}$ is flipped$]$ is at least $1 - C_2[O(\max(\epsilon, \rho_1))]^{\lambda_2} - \mathbb{P}[\boldsymbol{a_x} \neq \{\boldsymbol{y}^*, \bar{\boldsymbol{y}}\}]$.
(2) If $\bar{\boldsymbol{y}}$ is not corrected by $\kappa(h, \boldsymbol{x}, \bar{\boldsymbol{y}})$ with $\hat{\delta}$, let $\delta_2 = \max \left[\frac{\hat{\bar{S}}_{\bar{\boldsymbol{y}}}(\boldsymbol{x})}{\tau S_{\boldsymbol{b_x}}(\boldsymbol{x}) + \sum_{l_j \in \boldsymbol{y}^*} \sum_{i \neq j} T_{ij} \mathbb{P}(l_i|\boldsymbol{x})} \right]$ and $\rho_2 := |\hat{\delta} - \delta_2|$. Assume that $\epsilon \leq \frac{t_0 \delta_2^2 \tau - \rho_2 m - \rho_2^2 m}{3\delta_2^2}$. Then, $\mathbb{P}[\bar{\boldsymbol{y}}_{new} = h^*(\boldsymbol{x}), \bar{\boldsymbol{y}}$ is accepted$]$ is at least $1 - C_2[O(\max(\epsilon, \rho_2))]^{\lambda_2} - \mathbb{P}[\boldsymbol{a_x} \neq \{\boldsymbol{y}^*, \bar{\boldsymbol{y}}\}]$.

The proof of Theorem 1 is provided in Appendix B.1. Theorem 1 extends the theoretical results of [75] to multi-label classification with noisy labels. It claims that, even though with noisy multiple labels, the holistic correction has a guaranteed success rate to make proper corrections. Besides, if we can reasonably approximate the optimal δ with $\hat{\delta}$, our algorithm flips noisy multiple labels to correct ones with a good chance. Below, as a corollary of Theorem 1, we show that, there are certain circumstances, the use of holistic scores has a better chance to make corrections satisfactorily, than the sole use of instance-label dependence.

Corollary 1 *Suppose that $S(\boldsymbol{x})$ fulfills the holistic Tsybakov condition. Denote the set threshold $\hat{\delta}$ and optimal threshold δ. We define $\rho := \max |\hat{\delta} - \delta|$. We have that, $\exists \epsilon$ and ρ, if $C_2[O(\max(\epsilon, \rho))]^{\lambda_2} < C_1[O(\max(\epsilon, \rho))]^{\lambda_1}$, holistic correction brings higher probability to handle noisy labels successfully than instance-label dependence.*

The proof of Corollary 1 is provided in Appendix B.2. Corollary 1 claims that there exist cases where holistic scores better combat noisy labels. Note that, from a theoretical view, we do not state that holistic scores can work better in all circumstances of multi-label classification. Nevertheless, with the determination of the threshold $\hat{\delta}$, holistic scores can perform better in the experiments of this paper, which demonstrates the help of label dependence to handle noisy multiple labels.

4. Experiments

4.1. Experimental Setup

Datasets. We verify the effectiveness of the proposed method on the synthetic noisy versions of three datasets, *i.e.*, Pascal-VOC 2007 [12], Pascal-VOC 2012 [12], and MS-COCO [32]. Pascal-VOC 2007 contains 5,011 images in train and validation sets, while Pascal-VOC 2012 consists of 11,540 images in train and validation sets. The images come from 20 common object categories. For Pascal-VOC 2007 and Pascal-VOC 2012, we train methods using the noisy training and validation sets, and evaluate them on the test set of Pascal-VOC 2007 that has 4,952 images [14]. MS-COCO contains 82,081 training images and 40,137 validation images from 80 common object categories. As did in [74, 5, 65, 77], we evaluate the performance of methods using validation images.

Noisy-label generation. The class-dependent noise transition matrix $\boldsymbol{T}$ [42, 21, 46, 72] is used to corrupt the three

Table 1. Comparisons with advanced methods on noisy Pascal-VOC 2007. The mean and standard deviation of results (%) are presented.

Metrics	Methods / Noise	Sym. 30%	Sym. 40%	Sym. 50%	Pair. 20%	Pair. 30%	Pair. 40%
mAP ↑	BCE	64.50±1.20	58.65±2.16	48.19±0.23	71.77±1.15	60.94±4.25	48.72±2.13
	CSRA	66.99±0.48	59.62±0.61	46.97±0.48	72.45±0.69	63.58±1.48	52.72±1.52
	ADDGCN	63.89±0.94	55.75±1.98	44.14±1.37	71.02±0.95	61.05±0.06	50.18±2.70
	APL	66.79±1.19	58.86±1.53	47.64±1.81	72.61±0.99	61.99±0.78	49.10±0.15
	CDR	67.35±1.70	60.05±1.06	49.12±0.59	72.66±0.79	64.58±0.60	50.51±2.49
	JOINT	67.43±0.73	63.37±0.92	53.27±4.70	70.28±1.85	68.70±2.88	58.57±2.75
	WSIC	65.43±0.55	59.53±0.73	48.34±0.47	72.57±1.03	61.88±2.57	50.15±0.86
	CCMN	69.97±1.36	62.58±1.47	53.20±1.28	70.68±1.08	60.94±3.12	48.62±1.26
	HLC[†]	72.07±0.67	70.20±0.46	68.00±0.89	74.83±0.64	69.86±1.61	60.09±1.73
OF1 ↑	BCE	63.52±0.48	56.70±2.45	48.10±1.43	68.28±0.69	58.30±2.82	51.18±3.10
	CSRA	65.40±0.47	59.39±0.81	48.32±1.50	69.72±0.50	61.89±0.43	51.56±2.28
	ADDGCN	62.63±0.18	55.50±1.87	44.38±2.92	68.95±0.64	59.64±0.56	53.12±0.62
	APL	64.85±1.46	56.51±1.70	47.54±2.40	68.89±0.89	58.04±0.97	52.27±2.20
	CDR	65.31±0.99	57.93±1.05	48.86±1.71	69.53±0.65	59.89±1.07	51.68±3.83
	JOINT	69.72±0.88	67.93±0.77	61.62±1.40	71.24±1.03	64.20±0.88	60.30±1.24
	WSIC	63.45±0.97	57.96±1.25	48.38±2.41	69.88±1.22	57.97±2.19	51.99±1.65
	CCMN	69.66±1.55	60.43±1.31	53.84±0.69	67.12±0.61	59.55±1.45	53.46±1.04
	HLC[†]	71.03±0.33	69.08±1.00	68.62±0.48	72.09±0.74	65.76±2.39	60.71±1.37
CF1 ↑	BCE	58.91±1.34	53.21±2.04	43.66±0.53	65.93±0.81	57.03±3.43	47.21±1.89
	CSRA	62.31±0.50	55.67±0.61	43.11±0.76	67.39±0.80	59.66±1.04	51.13±1.12
	ADDGCN	60.41±1.04	53.72±1.38	42.42±0.59	66.05±0.97	57.81±0.58	48.89±2.64
	APL	60.23±1.53	52.85±2.18	42.38±1.67	66.59±0.71	58.33±0.49	47.67±1.83
	CDR	61.37±1.47	54.17±0.86	43.60±0.82	67.11±0.63	59.91±0.39	48.40±1.98
	JOINT	63.13±0.38	60.22±1.68	48.17±5.01	66.03±1.25	62.05±2.98	54.03±3.17
	WSIC	59.54±1.10	54.22±0.53	43.82±0.62	66.97±1.00	58.04±1.70	48.19±0.96
	CCMN	65.19±1.10	58.55±1.31	49.85±1.06	65.47±0.93	58.05±2.24	48.46±0.80
	HLC[†]	68.87±0.10	66.62±0.81	64.82±0.48	69.95±1.19	65.13±1.04	57.54±1.84

datasets. Here, for any $i \neq j$, $T_{ij} = \mathbb{P}(l_j \in \bar{\boldsymbol{y}} \wedge l_i \notin \bar{\boldsymbol{y}} | l_j \notin \boldsymbol{y} \wedge l_i \in \boldsymbol{y})$ represents the probability of the i-th class label to be flipped into the j-th class label. We consider both symmetric (abbreviated as Sym.) and pairflip (abbreviated as Pair.) noise settings [17]. The details of the transition matrix are provided in Appendix D.2. For symmetric noise, the noise rate is set to 30%, 40%, and 50% . For pairflip noise, the noise rate is set to 20%, 30%, and 40%.

Baselines. We exploit three types of baselines in total. Specifically, Type-I baselines contain the methods that are designed for multi-label classification with clean labels. Type-II baselines consider the methods for multi-class classification with noisy labels. Type-III baselines consider the methods that focus on multi-label classification with noisy labels. It should be noted that, there are relatively few methods belonging to this type [36]. More advanced methods belonging to Type-III baselines need to be investigated [36], which is also our focus in this paper. In more detail, Type-I baselines include CSRA [77] and ADDGCN [65]. Type-II baselines include APL [40], CDR [58], and JOINT [48]. Type-III baselines include WSIC [22] and CCMN [63]. As a simple baseline, we compare our method with the standard deep network that directly trains on noisy datasets (abbreviated as BCE). We detail all baselines in Appendix D.1.

Network & Optimizer. We use a ResNet-50 network [20] pretrained on ImageNet as the backbone for all methods. We train the models for 30 epochs in total. We utilize Adam [27] for the network optimization. The batch size is set to 128 for all the datasets. The learning rate is fixed to 5×10^{-5}. The images in Pascal-VOC 2007, Pascal-VOC 2012, and MS-COCO resize to 224×224. Note that, to make experiments more comprehensive, we also employ different experimental settings, *e.g.*, different networks and different image sizes. The details are provided in Section 4.3.

Measurement. As did in multi-label classification [77, 7], evaluation metrics include the mean average precision (mAP) [71], the average F1-measure (OF1), and the average per-class F1-measure (CF1). For fair comparison, we implement all methods with default parameters by PyTorch, and conduct all experiments on NVIDIA GTX3090 GPUs. All experiments are repeated three times with different random seeds. Following the works in learning with noisy labels [17, 54, 29, 30], the mean and standard deviation of results in the last epoch are reported. In addition, for different evaluation metrics, we report the mean and standard deviation of best results. Supplementary results are shown in Appendix E. Afterwards, the best mean results are highlighted in red. The second best mean results are also highlighted in blue.

4.2. Comparison with the State-of-the-Arts

The results on noisy Pascal-VOC 2007, Pascal-VOC 2012, and MS-COCO are shown in Table 1, Table 2, and Table 3 respectively. In summary, HLC consistently works

Table 2. Comparisons with advanced methods on noisy Pascal-VOC 2012. The mean and standard deviation of results (%) are presented.

Metrics	Methods / Noise	Sym. 30%	Sym. 40%	Sym. 50%	Pair. 20%	Pair. 30%	Pair. 40%
mAP ↑	BCE	66.74±0.80	56.07±0.50	45.15±1.56	70.91±1.13	57.61±1.14	49.85±0.36
	CSRA	66.35±0.50	56.20±1.35	45.54±1.14	71.29±0.83	60.71±1.18	47.63±1.56
	ADDGCN	63.34±0.96	54.54±0.86	44.88±1.71	70.41±0.54	57.96±0.68	47.66±1.08
	APL	67.07±1.04	56.79±1.86	43.51±1.93	71.32±1.60	59.59±1.27	48.14±1.16
	CDR	66.13±1.49	56.85±0.48	44.84±1.11	71.55±1.87	60.13±1.89	49.44±1.81
	JOINT	65.19±2.17	58.40±2.87	45.13±1.69	68.93±2.54	61.64±1.78	53.64±1.61
	WSIC	65.96±0.79	56.34±0.41	44.80±0.54	70.40±1.11	59.40±1.87	48.95±1.34
	CCMN	69.15±0.66	61.00±1.01	50.71±0.26	69.08±1.78	59.72±2.32	46.67±2.78
	HLC[†]	72.14±0.66	70.11±0.27	68.69±1.04	74.51±0.67	69.90±0.43	64.20±1.26
OF1 ↑	BCE	64.99±1.10	56.92±2.08	45.49±2.23	68.48±2.28	60.21±1.35	54.05±1.95
	CSRA	64.08±0.37	56.25±2.57	48.67±3.14	69.06±0.65	59.75±1.70	52.89±0.95
	ADDGCN	63.53±1.41	54.28±0.86	47.56±2.67	47.62±2.39	57.90±1.78	52.33±0.56
	APL	64.70±1.17	58.05±1.68	45.74±1.55	70.68±1.03	60.22±1.58	51.38±1.55
	CDR	64.06±1.38	57.31±1.21	46.51±0.95	70.45±1.44	60.57±1.24	52.26±2.42
	JOINT	67.35±1.86	64.57±2.39	54.37±3.33	70.81±1.40	64.40±1.76	56.27±1.29
	WSIC	62.74±2.10	57.13±0.73	45.52±1.28	69.72±1.19	59.11±2.04	52.49±1.38
	CCMN	65.77±0.23	59.91±0.93	51.45±0.94	67.93±1.73	59.26±0.51	48.61±4.71
	HLC[†]	71.14±0.60	69.50±0.40	67.80±0.33	72.13±0.26	67.59±0.96	64.28±0.81
CF1 ↑	BCE	62.47±0.44	53.26±0.41	43.43±1.67	66.03±1.69	55.90±0.70	49.29±0.64
	CSRA	62.08±0.70	53.23±1.27	43.23±1.25	66.02±0.74	57.71±1.02	47.46±1.68
	ADDGCN	59.67±1.14	52.61±0.52	44.33±1.99	65.22±0.86	55.32±0.76	47.30±1.12
	APL	62.99±1.07	53.69±1.80	41.72±1.42	66.44±1.40	57.52±0.91	48.14±1.02
	CDR	62.18±1.04	53.61±0.45	42.83±0.87	66.29±2.12	57.23±1.43	49.03±1.50
	JOINT	60.57±2.82	54.39±3.72	40.48±7.70	66.30±2.33	59.72±2.12	55.06±0.36
	WSIC	61.70±0.92	53.10±0.74	42.72±0.54	65.34±1.48	57.21±1.62	48.51±1.12
	CCMN	64.46±0.62	57.45±0.99	48.27±0.68	67.48±1.44	56.93±1.69	47.01±1.82
	HLC[†]	69.54±0.56	67.35±0.48	65.72±1.48	70.07±0.41	65.68±0.94	60.57±1.27

Table 3. Comparisons with advanced methods on noisy MS-COCO. The mean and standard deviation of results (%) are presented.

Metrics	Methods / Noise	Sym. 30%	Sym. 40%	Sym. 50%	Pair. 20%	Pair. 30%	Pair. 40%
mAP ↑	BCE	53.23±0.15	47.33±0.79	40.25±0.26	56.58±0.22	49.16±0.04	41.57±0.64
	CSRA	53.89±0.40	47.64±0.86	39.58±0.19	58.27±0.23	50.95±0.07	43.07±0.64
	ADDGCN	51.08±0.95	44.75±1.15	38.66±1.30	56.94±0.61	50.28±0.81	41.45±0.19
	APL	54.34±0.32	48.61±0.72	43.55±1.43	57.73±0.20	50.87±0.34	41.77±0.50
	CDR	54.01±0.04	49.01±0.26	43.94±1.25	57.03±0.28	50.99±0.77	42.71±0.09
	JOINT	53.93±0.41	48.01±1.04	45.27±0.68	57.30±0.33	51.94±0.20	42.74±0.55
	WSIC	52.99±0.53	46.84±0.86	39.76±0.64	56.66±0.31	49.46±0.25	42.52±0.62
	CCMN	51.73±0.18	50.36±0.71	45.32±0.89	58.13±0.44	51.17±0.29	42.12±0.76
	HLC[†]	54.87±0.68	51.09±0.53	48.15±0.50	58.55±0.09	53.41±0.13	45.91±0.39
OF1 ↑	BCE	51.34±1.70	44.36±0.82	34.85±1.24	59.16±0.95	52.44±0.81	42.94±1.13
	CSRA	52.03±1.86	41.63±1.41	33.47±3.18	59.17±0.14	50.27±0.88	41.75±1.36
	ADDGCN	55.67±1.48	47.79±0.40	35.95±3.73	60.96±0.65	55.05±1.78	47.47±0.77
	APL	51.07±1.32	43.93±2.70	33.90±4.00	60.04±1.16	50.64±2.86	44.34±1.99
	CDR	53.43±1.16	45.10±0.83	34.91±0.90	59.34±0.61	52.72±0.63	44.17±0.61
	JOINT	54.56±0.06	49.00±1.66	37.78±0.93	58.20±0.40	53.21±0.17	46.55±0.61
	WSIC	50.91±0.52	42.93±0.85	35.47±1.52	58.89±1.13	51.63±1.57	43.99±1.47
	CCMN	52.71±1.04	43.24±1.19	34.62±1.38	58.61±1.18	52.18±0.76	45.92±0.59
	HLC[†]	59.92±0.65	57.84±0.38	55.47±0.95	62.28±0.06	58.56±0.37	51.09±0.60
CF1 ↑	BCE	45.92±0.23	38.96±1.61	31.34±0.27	52.54±0.58	45.54±0.63	39.79±0.99
	CSRA	44.97±1.88	37.49±1.73	28.96±1.16	52.18±0.44	44.96±0.43	36.88±0.21
	ADDGCN	46.77±1.80	39.35±1.83	30.57±1.57	54.18±0.23	47.55±0.18	39.44±0.33
	APL	42.91±0.54	38.38±0.77	28.17±2.50	52.87±1.07	46.27±1.27	37.76±1.02
	CDR	46.62±0.42	39.47±0.54	29.59±2.52	52.51±0.69	45.75±0.81	39.15±0.53
	JOINT	49.51±0.81	42.38±1.21	24.24±0.61	54.39±0.17	49.90±0.85	38.34±0.55
	WSIC	45.30±1.09	39.15±1.62	31.42±0.94	52.04±0.28	45.76±0.70	39.44±1.11
	CCMN	44.20±1.19	35.18±1.01	27.90±1.25	53.23±0.58	46.88±0.92	40.55±0.89
	HLC[†]	51.94±0.63	49.24±0.30	46.69±0.66	55.44±0.13	50.91±0.48	43.35±0.82

best across all noise settings. In many cases, the best results achieved by HLC outperform the second best results by a large margin, especially when the noise level is high. Below, we further discuss the results based on the comparisons with three different types of baselines.

Compared with Type-I baselines. We first notice that Type-I baselines are fragile to noisy labels in multi-label classification. Without considering the side-effect of noisy labels, in many cases, they perform worse than BCE, which clearly illustrates the necessity for attention to handling noisy labels. Second, we compare HLC with ADDGCN. Without the proposed correction method for combating noisy labels, HLC will reduce to ADDGCN. As shown in the reported results, HLC performs much better than ADDGCN. To be specific, on noisy Pascal-VOC 2007, for Sym. 40%, HLC brings about +15% performance improvement *w.r.t.* three evaluation metrics over ADDGCN. For Sym. 50%, the performance improvement is increased to more than +20%. Also, for Pair. 30% and Pair. 40%, HLC enhances ADDGCN with about +10% improvement. On noisy Pascal-VOC 2012 and MS-COCO, the performance improvement is also very clear.

Compared with Type-II baselines. On noisy Pascal-VOC 2007, with Sym. noise, we can see that HLC outperforms APL, CDR, and JOINT clearly, especially for Sym. 50%. Additionaly, with Pair. noise, although the improvement is less than the cases with Sym. noise, HLC still performs best. On noisy Pascal-VOC 2012, for both Sym. and Pair. noise, the improvement is significant. Lastly, for noisy MS-COCO, HLC works better than all Type-II baselines with varying enhancement.

Note that, compared with APL and CDR, JOINT seems to be a stronger baseline. Benefiting from label correction, after a few training epochs, JOINT less overfits to wrong labels, following better performance. Nevertheless, the proposed label-correction paradigm is argued to be more advanced. As shown in all results, HLC surpasses JOINT, which verifies the effectiveness of our method.

Compared with Type-III baselines. On noisy Pascal-VOC 2007 and noisy Pascal-VOC 2012, HLC outperforms WSIC and CCMN distinctly. For example, with Sym. 50% noise, more than +10% performance promotion is brought by our method. On noisy MS-COCO, although WSIC and CCMN are sometimes competitive *w.r.t.* mAP, they are inferior *w.r.t.* both OF1 and CF1.

4.3. More Analyses and Justifications

In this subsection, we conduct performance analysis in more detail. The experiments are conducted with Sym. 50% noise, which is more challenging than the experiments in low-noise-rate cases.

Role of label dependence. We study the effect of removing the consideration of label dependence to provide insights into what makes HLC successful. The experiments are conducted on noisy Pascal-VOC 2007, Pascal-VOC 2012, and MS-COCO. The ResNet-50 network pretrained on ImageNet is used as the backbone. The image size is set to 224×224. Recall that HLC considers instance-label and label dependences simultaneously. When we remove the consideration of the label dependence in HLC, the corresponding method is named as HLC w/o l. here. For both HLC w/o l. and HLC, the value of the threshold $\hat{\delta}$ is searched in the range $\{0.25, 0.30, 0.35, 0.40, 0.45\}$. We use the 10% noisy training data as a validation set for the threshold determination and performance report. The results are shown in Table 4. As can be seen, HLC outperforms HLC w/o l.. The results justify our claims that the label dependence could help combat the noisy labels in multi-label classification, which demonstrate the effectiveness of the proposed holistic correction.

Table 4. Ablation study results on noisy Pascal-VOC 2007, Pascal-VOC 2012, and MS-COCO. The mean and standard deviation of results are presented. The best result in each case is in **bold**.

Dataset	Noisy Pascal-VOC 2007		
Methods	mAP ↑	OF1 ↑	CF1 ↑
HLC w/o l.	67.06±0.41	67.23±1.92	63.42±0.58
HLC	**68.00±0.89**	**68.62±0.48**	**64.82±0.48**
Dataset	Noisy Pascal-VOC 2012		
Methods	mAP ↑	OF1 ↑	CF1 ↑
HLC w/o l.	67.88±0.75	66.30±1.28	64.33±1.67
HLC	**68.69±1.04**	**67.80±0.33**	**65.72±1.48**
Dataset	Noisy MS-COCO		
Methods	mAP ↑	OF1 ↑	CF1 ↑
HLC w/o l.	46.21±0.36	52.90±0.92	44.51±1.29
HLC	**48.15±0.50**	**55.47±0.95**	**46.69±0.66**

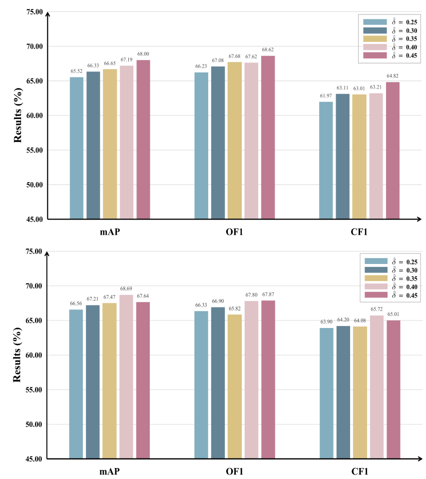

Figure 2. Ablation study results with different values of the set threshold $\hat{\delta}$. The experiments are conducted on noisy Pascal-VOC 2007 (**Top**) and noisy Pascal-VOC 2012 (**Bottom**).

Analysis of the threshold $\hat{\delta}$. We analyze the influence of different values of $\hat{\delta}$. The experiments are conducted on noisy Pascal-VOC 2007 and Pascal-VOC 2012. The ResNet-50 network pretrained on ImageNet is used as the backbone. The image size is set to 224×224. The value of the threshold $\hat{\delta}$ is chosen in $\{0.25, 0.30, 0.35, 0.40, 0.45\}$. Figure 2 shows that HLC is robust to the determination of the threshold $\hat{\delta}$ in the certain range, which facilitates the practical application of our method.

Evaluations with different networks. We use pretrained

Table 5. Comparisons with advanced methods on noisy MS COCO with different networks. The mean and standard deviation of results (%) are presented.

Metrics	Methods	ResNet-34	ResNet-101
mAP ↑	BCE	42.63±0.74	38.17±0.41
	CSRA	41.35±0.18	37.24±1.20
	ADDGCN	40.15±0.98	36.13±0.69
	APL	44.82±0.70	40.90±1.51
	CDR	45.43±0.65	41.00±0.38
	JOINT	44.81±0.77	39.96±1.30
	WSIC	41.86±0.62	37.49±0.68
	CCMN	45.31±0.47	46.01±1.01
	HLC†	46.05±0.81	46.24±2.13
OF1 ↑	BCE	37.65±2.46	38.65±2.50
	CSRA	35.05±0.78	34.28±2.40
	ADDGCN	35.11±1.26	37.18±0.50
	APL	34.31±1.73	37.89±1.30
	CDR	36.67±3.43	39.88±1.15
	JOINT	39.66±1.13	41.36±0.88
	WSIC	35.08±1.74	38.44±0.57
	CCMN	32.86±1.50	37.03±1.48
	HLC†	44.11±0.80	47.79±4.19
CF1 ↑	BCE	28.95±2.09	34.11±1.13
	CSRA	27.40±0.44	31.21±1.57
	ADDGCN	26.11±0.86	30.41±0.72
	APL	26.64±2.39	31.97±1.95
	CDR	27.13±2.39	35.17±1.06
	JOINT	30.77±1.63	37.63±0.81
	WSIC	27.62±0.65	33.83±0.61
	CCMN	24.75±0.48	26.65±0.26
	HLC†	34.79±1.41	40.88±2.42

ResNet-50 before. To show that our method is robust to the choice of network structures, we use different networks in experiments. Specifically, we employ pretrained ResNet-34 [20] and pretrained ResNet-101 [20] respectively. The noisy MS-COCO is considered. The image size is 224×224. The results on mAP are reported in Table 5. As can be seen, with different networks, HLC still works well.

Evaluations with different image sizes. We resize the image size to 224×224 before. To test the performance of advanced methods with different image sizes, we further consider 112×112, 384×384, and 448×448 image sizes. Pretrained ResNet-50 is used. The results are reported in Table 6. For mAP, we can see that HLC is competitive compared with CCMN and CSRA. For OF1 and CF1, HLC works better than all baselines with a clear margin.

4.4. Experiments on the Real-world Dataset

To demonstrate that our problem setting can be adapted to the real world and our method can well handle practical scenes, we employ the real-world dataset NUS-WIDE [10] that originally contained 269,648 images from Flicker, which have been manually annotated with 81 visual concepts. Since some urls for download have been deleted, we employ the dataset version in [45]. A standard 70-30 train-test split is used. The backbone is chosen as ResNet-101. As the computation cost of training on NUS-WIDE is relatively large, we run experiments one time. Here we compare our method with S-CLs [39], MS-CMA [66], SRN [76], ICME [7], and ASL [45]. For convenient comparison, we refer to the results of their original papers. Note that to further improve the performance on NUS-WIDE, we utilize ASL to replace the loss function of our method. We name the new method "HLC+ASL". Results are provided in Table 7, which demonstrate the effectiveness of our method on the real-world dataset.

Table 6. Comparisons with advanced methods on noisy MS COCO. The mean and standard deviation of results (%) are presented. Difference image sizes are considered here.

Metrics	Image sizes	112×112	384×384	448×448
mAP ↑	BCE	32.22±0.69	39.40±1.36	35.24±1.73
	CSRA	29.55±0.16	43.55±0.70	44.56±0.75
	ADDGCN	32.34±0.46	38.72±1.64	34.87±1.89
	APL	34.41±0.48	43.65±0.28	41.44±1.21
	CDR	34.75±0.39	43.26±0.72	39.97±1.40
	JOINT	32.89±0.16	42.95±0.88	40.17±1.26
	WSIC	31.98±0.23	39.57±1.02	36.08±0.23
	CCMN	36.17±0.41	44.39±0.39	44.03±0.17
	HLC†	35.98±1.05	45.12±0.13	44.23±1.20
OF1 ↑	BCE	26.70±0.88	34.71±2.76	26.72±3.35
	CSRA	20.14±1.28	36.41±0.71	38.54±0.78
	ADDGCN	26.36±2.29	42.83±2.05	40.73±1.04
	APL	24.02±1.22	34.68±1.46	30.73±2.64
	CDR	26.50±1.23	31.31±1.76	31.15±3.46
	JOINT	34.11±0.95	38.67±1.25	38.11±0.69
	WSIC	24.61±1.10	34.09±2.94	30.69±0.97
	CCMN	23.89±1.49	36.16±2.12	25.03±2.48
	HLC†	39.05±2.68	46.55±3.77	45.14±2.34
CF1 ↑	BCE	19.61±0.61	30.77±2.28	24.94±3.31
	CSRA	13.34±1.29	32.66±0.98	32.68±0.22
	ADDGCN	18.67±1.47	35.63±1.37	33.89±2.41
	APL	17.21±0.78	29.24±0.09	25.01±1.31
	CDR	18.04±1.07	29.62±1.19	26.28±1.23
	JOINT	20.76±0.75	34.90±1.88	33.75±1.31
	WSIC	19.07±0.57	31.63±2.49	28.50±1.36
	CCMN	16.75±1.21	30.09±2.32	26.68±1.57
	HLC†	29.70±2.07	40.34±2.14	37.48±2.36

Table 7. Comparison of our method to known state-of-the-art models on the NUS-WIDE dataset. Metrics are in %.

Method	mAP ↑	OF1 ↑	CF1 ↑
S-CLs [39]	60.1	73.7	58.7
MS-CMA [66]	61.4	73.8	60.5
SRN [76]	62.0	73.4	58.5
ICME [7]	62.8	74.1	60.7
ASL [45]	63.9	74.6	62.7
HLC†	63.1	74.6	62.9
HLC+ASL†	64.5	75.1	63.4

5. Conclusion

In this paper, we focus on the realistic problem of multi-label classification with noisy labels. We learn and utilize the label dependence among multiple labels to handle this problem. With the help of label dependence, a novel algorithm named HLC is proposed to correct noisy multiple labels to clean ones. We demonstrate the effectiveness of our algorithm both theoretically and empirically. For future work, we are interested in adapting HLC to other domains such as natural language processing and recommendation systems. We are also interested in promoting our algorithm to tackle instance-dependent label noise [73, 4, 78, 38] in multi-label classification.

Acknowledgements

Bo Han was supported by NSFC Young Scientists Fund No. 62006202 and Guangdong Basic and Applied Basic Research Foundation No. 2022A1515011652. Tongliang Liu is partially supported by ARC projects: FT220100318, DP220102121, LP220100527, LP220200949, and IC190100031.

References

[1] Devansh Arpit, Stanisław Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *ICML*, pages 233–242, 2017. 2

[2] Dara Bahri, Heinrich Jiang, and Maya Gupta. Deep k-nn for noisy labels. In *ICML*, pages 540–550, 2020. 4

[3] Yingbin Bai and Tongliang Liu. Me-momentum: Extracting hard confident examples from noisily labeled data. In *ICCV*, pages 9312–9321, 2021. 2

[4] Antonin Berthon, Bo Han, Gang Niu, Tongliang Liu, and Masashi Sugiyama. Confidence scores make instance-dependent label-noise learning possible. *arXiv preprint arXiv:2001.03772*, 2020. 9

[5] Tianshui Chen, Muxin Xu, Xiaolu Hui, Hefeng Wu, and Liang Lin. Learning semantic-specific graph representation for multi-label image recognition. In *ICCV*, pages 522–531, 2019. 4

[6] Zhaomin Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Learning graph convolutional networks for multi-label recognition and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 3

[7] Zhao-Min Chen, Xiu-Shen Wei, Xin Jin, and Yanwen Guo. Multi-label image recognition with joint class-aware map disentangling and label correlation embedding. In *ICME*, pages 622–627, 2019. 1, 3, 5, 8

[8] Xiang Cheng, Shu-Guang Zhao, Xuan Xiao, and Kuo-Chen Chou. iatc-misf: a multi-label classifier for predicting the classes of anatomical therapeutic chemicals. *Bioinformatics*, 33(3):341–346, 2017. 1

[9] Tejas Chheda, Purujit Goyal, Trang Tran, Dhruvesh Patel, Michael Boratko, Shib Sankar Dasgupta, and Andrew McCallum. Box embeddings: An open-source library for representation learning using geometric structures. *arXiv preprint arXiv:2109.04997*, 2021. 1

[10] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *ACM International Conference on Image and Video Retrieval*, pages 1–9, 2009. 8

[11] Zijun Cui, Yong Zhang, and Qiang Ji. Label error correction and generation through label relationships. In *AAAI*, pages 3693–3700, 2020. 1

[12] Mark Everingham, Andrew Zisserman, Christopher KI Williams, Luc Van Gool, Moray Allan, Christopher M Bishop, Olivier Chapelle, Navneet Dalal, Thomas Deselaers, Gyuri Dorkó, et al. The pascal visual object classes challenge 2007 (voc2007) results. 2008. 4

[13] Lei Feng, Senlin Shu, Zhuoyi Lin, Fengmao Lv, Li Li, and Bo An. Can cross entropy loss be robust to label noise? In *IJCAI*, pages 2206–2212, 2021. 1

[14] Bin-Bin Gao and Hong-Yu Zhou. Learning to discover multi-class attentional regions for multi-label image recognition. *IEEE Transactions on Image Processing*, 30:5920–5932, 2021. 1, 4

[15] Wei Gao, Bin-Bin Yang, and Zhi-Hua Zhou. On the resistance of nearest neighbor to random noisy labels. *arXiv preprint arXiv:1607.07526*, 2016. 4

[16] Bo Han, Gang Niu, Xingrui Yu, Quanming Yao, Miao Xu, Ivor Tsang, and Masashi Sugiyama. Sigua: Forgetting may make learning with noisy labels more robust. In *ICML*, pages 4006–4016, 2020. 1

[17] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *NeurIPS*, pages 8527–8537, 2018. 5

[18] Jun-Yi Hang and Min-Ling Zhang. Collaborative learning of label semantics and deep label-specific features for multi-label classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 3

[19] Jun-Yi Hang and Min-Ling Zhang. Dual perspective of label-specific feature learning for multi-label classification. In *ICML*, pages 8375–8386, 2022. 1

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 5, 8

[21] Dan Hendrycks, Mantas Mazeika, Duncan Wilson, and Kevin Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. In *NeurIPS*, 2018. 4

[22] Mengying Hu, Hu Han, Shiguang Shan, and Xilin Chen. Weakly supervised image classification through noise regularization. In *CVPR*, pages 11517–11525, 2019. 5

[23] Zhuo Huang, Xiaobo Xia, Li Shen, Bo Han, Mingming Gong, Chen Gong, and Tongliang Liu. Harnessing out-of-distribution examples via augmenting content and style. In *ICLR*, 2023. 1

[24] Zhuo Huang, Miaoxi Zhu, Xiaobo Xia, Li Shen, Jun Yu, Chen Gong, Bo Han, Bo Du, and Tongliang Liu. Robust generalization against photon-limited corruptions via worst-case sharpness minimization. In *CVPR*, pages 16175–16185, 2023. 1

[25] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*, pages 2309–2318, 2018. 2, 3

[26] Youngdong Kim, Junho Yim, Juseung Yun, and Junmo Kim. Nlnl: Negative learning for noisy labels. In *ICCV*, pages 101–110, 2019. 1
[27] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5
[28] Cheng Li, Bingyu Wang, Virgil Pavlu, and Javed Aslam. Conditional bernoulli mixtures for multi-label classification. In *ICML*, pages 2482–2491, 2016. 3
[29] Junnan Li, Richard Socher, and Steven C.H. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *ICLR*, 2020. 5
[30] Junnan Li, Caiming Xiong, and Steven CH Hoi. Learning from noisy data with robust representation learning. In *ICCV*, pages 9485–9494, 2021. 5
[31] Shikun Li, Xiaobo Xia, Hansong Zhang, Yibing Zhan, Shiming Ge, and Tongliang Liu. Estimating noise transition matrix with label correlations for noisy multi-label learning. In *NeurIPS*, 2022. 1
[32] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014. 4
[33] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. In *NeurIPS*, 2020. 3
[34] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(3):447–461, 2016. 2
[35] Weiwei Liu, Ivor W Tsang, and Klaus-Robert Müller. An easy-to-hard learning paradigm for multiple classes and multiple labels. *Journal of Machine Learning Research*, 18, 2017. 1
[36] Weiwei Liu, Haobo Wang, Xiaobo Shen, and Ivor Tsang. The emerging trends of multi-label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 5
[37] Weiwei Liu, Donna Xu, Ivor W Tsang, and Wenjie Zhang. Metric learning for multi-output tasks. *Transactions on Pattern Analysis and Machine Intelligence*, 41(2):408–422, 2018. 1
[38] Yang Liu. Identifiability of label noise transition matrix. *arXiv preprint arXiv:2202.02016*, 2022. 9
[39] Yongcheng Liu, Lu Sheng, Jing Shao, Junjie Yan, Shiming Xiang, and Chunhong Pan. Multi-label image classification via knowledge distillation from weakly-supervised detection. In *ACMMM*, pages 700–708, 2018. 8
[40] Xingjun Ma, Hanxun Huang, Yisen Wang, Simone Romano, Sarah Erfani, and James Bailey. Normalized loss functions for deep learning with noisy labels. In *ICML*, pages 6543–6553, 2020. 5
[41] Yasumasa Onoe, Michael Boratko, Andrew McCallum, and Greg Durrett. Modeling fine-grained entity types with box embeddings. In *ACL*, 2021. 1
[42] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *CVPR*, pages 1944–1952, 2017. 4
[43] Cosmin Octavian Pene, Amirmasoud Ghiassi, Taraneh Younesian, Robert Birke, and Lydia Y Chen. Multi-label gold asymmetric loss correction with single-label regulators. *arXiv preprint arXiv:2108.02032*, 2021. 2
[44] Geoff Pleiss, Tianyi Zhang, Ethan R Elenberg, and Kilian Q Weinberger. Identifying mislabeled data using the area under the margin ranking. In *NeurIPS*, 2020. 1
[45] Tal Ridnik, Emanuel Ben-Baruch, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, and Lihi Zelnik-Manor. Asymmetric loss for multi-label classification. In *ICCV*, 2021. 1, 8
[46] Jun Shu, Qian Zhao, Zengben Xu, and Deyu Meng. Meta transition adaptation for robust deep learning with noisy labels. *arXiv preprint arXiv:2006.05697*, 2020. 2, 4
[47] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022. 1
[48] Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *CVPR*, 2018. 5
[49] Alexander B Tsybakov. Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 32(1):135–166, 2004. 4
[50] Arash Vahdat. Toward robustness against label noise in training deep discriminative neural networks. In *NeurIPS*, pages 5596–5605, 2017. 1
[51] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *CVPR*, 2017. 1
[52] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. Cnn-rnn: A unified framework for multi-label image classification. In *CVPR*, pages 2285–2294, 2016. 1
[53] Xinshao Wang, Yang Hua, Elyor Kodirov, David A Clifton, and Neil M Robertson. Proselflc: Progressive self label correction for training robust deep neural networks. In *CVPR*, pages 752–761, 2021. 2
[54] Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *CVPR*, pages 8688–8696, 2018. 5
[55] Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In *CVPR*, pages 13726–13735, 2020. 1
[56] Hongxin Wei, Lue Tao, Renchunzi Xie, and Bo An. Open-set label noise can improve robustness against inherent label noise. In *NeurIPS*, pages 7978–7992, 2021. 1
[57] Baoyuan Wu, Siwei Lyu, and Bernard Ghanem. Ml-mg: Multi-label learning with missing labels using a mixed graph. In *ICCV*, pages 4157–4165, 2015. 3
[58] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In *ICLR*, 2021. 1, 5

[59] Xiaobo Xia, Tongliang Liu, Bo Han, Mingming Gong, Jun Yu, Gang Niu, and Masashi Sugiyama. Sample selection with uncertainty of losses for learning with noisy labels. In *ICLR*, 2022. 1

[60] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. In *NeurIPS*, 2020. 2

[61] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. Are anchor points really indispensable in label-noise learning? In *NeurIPS*, pages 6835–6846, 2019. 1

[62] Ming-Kun Xie and Sheng-Jun Huang. Partial multi-label learning. In *AAAI*, 2018. 3

[63] Ming-Kun Xie and Sheng-Jun Huang. Ccmn: A general framework for learning with class-conditional multi-label noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 5

[64] Ming-Kun Xie, Jia-Hao Xiao, and Sheng-Jun Huang. Label-aware global consistency for multi-label learning with single positive labels. In *NeurIPS*, 2022. 1

[65] Jin Ye, Junjun He, Xiaojiang Peng, Wenhao Wu, and Yu Qiao. Attention-driven dynamic graph convolutional network for multi-label image recognition. In *ECCV*, pages 649–665, 2020. 1, 2, 3, 4, 5

[66] Renchun You, Zhiyao Guo, Lei Cui, Xiang Long, Yingze Bao, and Shilei Wen. Cross-modality attention with semantic graph embedding for multi-label classification. In *AAAI*, pages 12709–12716, 2020. 8

[67] Hsiang-Fu Yu, Prateek Jain, Purushottam Kar, and Inderjit Dhillon. Large-scale multi-label learning with missing labels. In *ICML*, pages 593–601, 2014. 3

[68] Ze-Bang Yu and Min-Ling Zhang. Multi-label classification with label-specific feature generation: A wrapped approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 3

[69] Jiong Zhang, Wei-cheng Chang, Hsiang-fu Yu, and Inderjit Dhillon. Fast multi-resolution transformer fine-tuning for extreme multi-label text classification. In *NeurIPS*, 2021. 1

[70] Min-Ling Zhang and Kun Zhang. Multi-label learning by exploiting label dependency. In *KDD*, pages 999–1008, 2010. 1, 3

[71] Min-Ling Zhang and Zhi-Hua Zhou. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2013. 1, 5

[72] Yivan Zhang, Gang Niu, and Masashi Sugiyama. Learning noise transition matrix from only noisy labels via total variation regularization. In *ICML*, 2021. 4

[73] Yikai Zhang, Songzhu Zheng, Pengxiang Wu, Mayank Goswami, and Chao Chen. Learning with feature-dependent label noise: A progressive approach. In *ICLR*, 2021. 9

[74] Jiawei Zhao, Ke Yan, Yifan Zhao, Xiaowei Guo, Feiyue Huang, and Jia Li. Transformer-based dual relation graph for multi-label image recognition. In *ICCV*, pages 163–172, 2021. 1, 4

[75] Songzhu Zheng, Pengxiang Wu, Aman Goswami, Mayank Goswami, Dimitris Metaxas, and Chao Chen. Error-bounded correction of noisy labels. In *ICML*, pages 11447–11457, 2020. 1, 4

[76] Feng Zhu, Hongsheng Li, Wanli Ouyang, Nenghai Yu, and Xiaogang Wang. Learning spatial regularization with image-level supervisions for multi-label image classification. In *CVPR*, pages 5513–5522, 2017. 8

[77] Ke Zhu and Jianxin Wu. Residual attention: A simple but effective method for multi-label recognition. In *ICCV*, pages 184–193, 2021. 1, 4, 5

[78] Zhaowei Zhu, Tongliang Liu, and Yang Liu. A second-order approach to learning with instance-dependent label noise. In *CVPR*, pages 10113–10123, 2021. 9