Boosting Single Positive Multi-label Classification with Generalized Robust Loss*

Yanxi Chen, Chunxiao Li, Xinyang Dai, Jinhuan Li, Weiyu Sun, Yiming Wang, Renyuan Zhang, Tinghe Zhang and Bo Wang†

University of International Business and Economics
{202310411355, 202310411351, wangbo}@uibe.edu.cn

Abstract

Multi-label learning (MLL) requires comprehensive multi-semantic annotations that is hard to fully obtain, thus often resulting in missing labels scenarios. In this paper, we investigate Single Positive Multi-label Learning (SPML), where each image is associated with merely one positive label. Existing SPML methods only focus on designing losses using mechanisms such as hard pseudo-labeling and robust losses, mostly leading to unacceptable false negatives. To address this issue, we first propose a generalized loss framework based on expected risk minimization to provide soft pseudo labels, and point out that the former losses can be seamlessly converted into our framework. In particular, we design a novel robust loss based on our framework, which enjoys flexible coordination between false positives and false negatives, and can additionally deal with the imbalance between positive and negative samples. Extensive experiments show that our approach can significantly improve SPML performance and outperform the vast majority of state-of-the-art methods on all the four benchmarks. Our code is available at https://github.com/yan4xi1/GRLoss.

1 Introduction

Multi-label learning (MLL) constructs predictive models to provide multi-label assignments to unseen images [Zhao et al., 2022]. Conventionally, MLL assumes that each training instance is fully and accurately labeled with all relevant classes. However, obtaining such comprehensive labels is often expensive and even impractical in many real-world scenarios [Lv et al., 2019]. Due to these limitations of vanilla MLL, Multi-label Learning with Missing Labels (MLML) [Wu et al., 2014] received broad attentions, and was usually considered as a typical weakly supervised learning problem [Kim et al., 2022].

In this paper, we focus on an extreme variant of MLML, i.e., Single Positive Multi-label Learning (SPML), where only one label is verified as positive for a given image, leaving all the other labels as the unknown ones [Cole et al., 2021]. Compared to a fully labeled setting, SPML poses a practically more relevant yet substantially more challenging scenario. Since only one positive label is known, existing approaches for general MLML based on label relations, such as learning positive label correlations [Hayn and Elhamifar, 2020], creating label matrices [Feng et al., 2020], or learning to infer missing labels [Durand et al., 2019], are obstructed in the context of single labeled positive [Cole et al., 2021].

Nevertheless, in the realm of SPML, the priority still lies in effectively dealing with missing labels. Generally speaking, two strategies are commonly leveraged. The first strategy is to treat missing labels as unknown variables needed to be predicted [Rastogi and Mortaza, 2021]. As discussed above, predicting such unseen labels based on only one positive label is indeed challenging. The second one is to assume the missing labels are All Negative (AN), so that to transform SPML into a fully supervised MLL problem [Cole et al., 2021]. To our knowledge, AN assumption is still of the most popularity for MLML, including SPML (as a baseline) [Liu et al., 2023b], and normally trained with Binary Cross-entropy (BCE) loss. Following the relevant study, in this paper, AN with BCE is also considered as the baseline in our experimental part.

Unfortunately, AN assumption is oversimplified and inevitably results in a large number of false negative labels [Zhang et al., 2023]. Ignoring false negatives will introduce label noise, which substantially impairs SPML performance [Ghiassi et al., 2023b]. To circumvent the pitfalls of mislabeling in AN, pseudo-labeling has been explored [Hu et al., 2021]. In this work, we adapt the ideas of soft pseudo labeling to SPML. In addition, the remaining false negatives in pseudo-labeling are regarded as noise [Liu et al., 2023b], and properly handling these noisy labels can consistently improve SPML performance [Xia et al., 2023b]. Therefore, we specifically design a novel robust denoising loss to reduce the negative impact of noise in pseudo labels.

Another challenge in SPML is the extreme intra-class imbalance between positive and negative labels within one category, which is much severer than vanilla MLL and MLML [Ridnik et al., 2021; Tarekegn et al., 2021]. In order to tackle intra-class imbalance in SPML, a novel approach using both entropy maximization and asymmetric pseudo-labeling has been introduced [Zhou et al., 2022a], leading
to promising results. Nevertheless, the mechanism to address the issue of positive-negative label imbalance in SPML remains relatively unexplored. Furthermore, the imbalance across classes, a.k.a inter-class imbalance, easily causes the Long-tailed effect [Zhang et al., 2023], which is even aggravated in SPML. In this paper, we effectively mitigate both intra-class and inter-class imbalance issue by instance and class sensitive reweighting.

In this paper, aiming at solving the SPML problem, we propose a new loss function framework. Specifically, we first derive an empirical risk estimation for SPML based on single-class unbiased risk estimation (URE), and accordingly propose a novel loss function framework for SPML problems. To integrate Pseudo-labeling, Loss reweighting and Robust loss into our framework, we build Generalized Robust Loss (GR Loss) for SPML, allowing previous SPML methods to be naturally incorporated. With two mild assumptions, we offer the specific forms of the components in our GR Loss. Extensive experimental results on four benchmark datasets demonstrate that our GR Loss can achieve state-of-the-art performance. Our contributions can be summarized in four-fold:

• Framework level: We propose a novel SPML loss function framework to estimate the posterior probability of missing labels more accurately. In particular, our framework can unify existing SPML methods, enabling a comprehensive and holistic understanding of these former work.

• Formulation level: We design a tailored robust loss function for both positive and negative labels, as a novel surrogate for cross-entropy loss. The proposed robust loss is off-the-shell and can be seamlessly plug-in for our proposed framework to build the full Generalized Robust Loss.

• Methodology level: We elaborate the false negative and imbalance issues in SPML by gradient analysis and empirical study, and first propose a soft pseudo-labeling mechanism to address these issues.

• Experimental level: We demonstrate the superiority of our GR Loss by conducting extensive empirical analysis with performance comparison against state-of-the-art SPML methods across four benchmarks.

2 Related Work

Due to its highly incomplete labels, SPML presents unique challenge for MLL [Liu et al., 2023a], especially resulting in oversimplified prediction to falsely identify all labels as positive [Verelst et al., 2023]. In addition to SPML, incomplete supervision widely occurs in the context of weakly supervised learning, where pseudo-labeling has been widely employed [Liu et al., 2022; Chen et al., 2023; Wang et al., 2022; Zhou et al., 2022b]. The pseudo-labeling was initially applied to SPML to waive the issue of false negatives caused by AN assumption [Cole et al., 2021], where the pseudo labels were incorporated with regularization [Zhou et al., 2022a; Liu et al., 2023b]. In particular, label-aware global consistency regularization leverages contrastive learning to extract flow structure information, so that the latent soft labels can be accurately recovered [Xie et al., 2022].

Pseudo-labeling may introduce label noise [Xia et al., 2023a; Higashimoto et al., 2024]. In order to effectively cope with noisy labels, denoising robust losses [Ma et al., 2020], such as Generalized Cross Entropy (GCE) [Zhang and Sabuncu, 2018], Symmetric Cross Entropy (SCE) [Wang et al., 2019], and Taylor Cross Entropy [Feng et al., 2021], have been successively developed. In particular, robust Mean Absolute Error (MAE) loss was combined with Cross Entropy (CE) loss to simultaneously consider the robustness and generalization performance [Ghosh et al., 2017]. It has been shown that robust losses, typically reserved for multi-class scenario, are also effective in MLL, especially in the presence of incomplete labels [Zhang et al., 2021; Xu et al., 2022], where both false positive and false negative labels should be properly addressed [Ghia et al., 2023a]. Our method is under the same umbrella, and specifically tailors robust loss to SPML.

3 Method

3.1 Problem Definition

In partial multi-label learning [Xie and Huang, 2018; Xie and Huang, 2021], the full data can be represented as a triplet \((x, y, s)\), where \(x \in \mathcal{X}\) represents the input feature, \(y \in \mathcal{Y} = \{0, 1\}^C\) denotes the ground-truth label, and \(s \in \mathcal{S} = \{0, 1\}^C\) is the observed label. In addition, \(y_c \in \{0, 1\}\) denotes the \(c\)-th entry of \(y\), where \(y_c = 1\) indicates the relevance to the \(c\)-th class, while \(y_c = 0\) means the irrelevance. Besides, \(s_c \in \{0, 1\}\) denotes the \(c\)-th entry of \(s\), while \(s_c = 1\) indicates the relevance to the \(c\)-th class, while \(s_c = 0\) means either irrelevance or relevance, i.e., the unknown.

Let \(D = \{(x^n, s^n)\}_{n=1}^N\) be a partially labeled MLL training set with \(N\) instances. For a given sample \(x^n\), we only have access to the observed label \(s^n\), while the ground-truth label \(y^n\) remains unknown. In SPML, each instance \(x^n\) has only one observed positive label, leaving the observed labels for all other categories being missing, represented as \(\sum_{i=1}^C s^n_i = 1\).

3.2 Expected Risk Estimation

The goal of SPML is to find a function \(f : \mathcal{X} \rightarrow \mathcal{Y}\) from training set \(D\) to predict true labels for each \(x \in \mathcal{X}\). In multi-label classification, the standard approach is to independently train \(C\) binary classifiers \(f = [f_1, f_2, \ldots, f_C]\), using a shared backbone. For a certain class, it can be viewed as a PU-learning problem [Kiryo et al., 2017; Jiang et al., 2023], where \(y\) and \(s\) degenerate to scalars. Let \(s = 1\) if \(x\) is labeled, and \(s = 0\) if it is unlabeled. Accordingly, the surrogate expected risk function w.r.t. loss \(\mathcal{L}\) is:

\[
\mathbb{R}(f_i) = \mathbb{E}_{p(x,y,s)}[\mathcal{L}(f_i(x), y)].
\] (1)

Then, the empirical risk is derived from Eq.(1), and the details are provided in Appendix A of our full version:

\[
N \cdot \hat{\mathbb{R}}(f_i) = \sum_{(x,s=1)} \mathcal{L}(f_i(x), y = 1) + \sum_{(x,s=0)} k(x)\mathcal{L}(f_i(x), y = 1) + (1 - k(x))\mathcal{L}(f_i(x), y = 0),
\] (2)
where $k(x) = P(y = 1 \mid x, s = 0)$ represents the probability that an unlabeled sample is false negative under AN assumption. Consequently, for the classifier $f$, the empirical risk is:

$$N \cdot \hat{R}(f) = \sum_{n=1}^{N,C} s_n^o \mathcal{L}_i^{n+} + (1-s_n^o) \left[ k_n^o \mathcal{L}_i^{n+} + (1-k_n^o) \mathcal{L}_i^{n-} \right],$$

where $k_n^o = P(y_n^0 = 1 \mid x^n, s_n^0 = 0)$, $\mathcal{L}_i^{n+} = \mathcal{L}(f_i(x^n), y_n^0 = 1)$, and $\mathcal{L}_i^{n-} = \mathcal{L}(f_i(x^n), y_n^0 = 0)$, $n = 1, 2, \cdots, N$, $i = 1, 2, \cdots, C$.

### 3.3 Novel SPML Loss Framework

According to Eq. (3), we propose a novel loss function framework for SPML based on the following two considerations.

**Pseudo-labeling.** Given the difficulty in evaluating accuracy $k_i(x)$, we estimate it in an online manner: Let $p_i(x) = f_i(x)$ be the current model output, and we use $\hat{k}(p_i(x); \beta)$ to estimate $k_i(x)$, where $\beta$ is the parameters.

**Loss reweighting.** In order to deal with the class imbalance and false negatives, we propose the class-and-instance-specific weight $v(p_i^n; \alpha)$ to reweight different samples w.r.t. the category, where $\alpha$ is the parameters. Typically, for any certain class $i$, positive and negative losses are in a unified form for balanced classification, such as cross entropy loss $\mathcal{L}_i^+ = -\log(p_i)$, $\mathcal{L}_i^- = -\log(1-p_i)$. However, there is a severe intra-class and inter-class imbalance in SPML, together with noisy labels in the pseudo labels and negative samples, i.e., false negatives. Therefore, we add $v(p_i^n; \alpha_i)$ to the loss of each instance for each class, and use $\mathcal{L}_i^{n,1}, \mathcal{L}_i^{n,2}, \mathcal{L}_i^{n,3}$ to decouple the loss in Eq.(3) into three surrogate losses. Accordingly, the total loss can be expressed as:

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} \sum_{i=1}^{C} v(p_i^n; \alpha_i) \cdot \mathcal{L}_i^n,$$

and the class-and-instance-specific loss $\mathcal{L}_i^n$ is decoupled as:

$$\mathcal{L}_i^n = s_i^n \mathcal{L}_i^{n,1} + (1-s_i^n) \left[ \hat{k}(p_i^n; \beta_i) \mathcal{L}_i^{n,2} + (1-\hat{k}(p_i^n; \beta_i)) \mathcal{L}_i^{n,3} \right].$$

### The Analysis and Formulation of $\hat{k}(p; \beta)$

In this section, we present two assumptions on $\hat{k}(p; \beta)$. The superscript $n$ and subscript $i$ are properly omitted for simplicity. We defer to verify their validity in Appendix E.3 (see the full version). Based on these assumptions, we introduce the explicit form of the pseudo-labeling function $\hat{k}(p; \beta)$.

**Assumption 1.** In the initial training phase, $\hat{k}(p; \beta)$ is nearly a constant function.

**Remark 1.** In the initial phase of training, the model is unable to provide accurate prediction, thus the initial output can be approximately considered as random one. As a result, the probability $\hat{k}(p; \beta)$ is independent of the output and performs as a constant function. Furthermore, this constant is approximately equal to the number of false-negative labels divided by the number of missing labels, i.e., $\hat{k}(p; \beta) = \frac{\#\{y = 1, s = 0\}}{\#\{s = 0\}}$.

**Assumption 2.** In the final training stage, $\hat{k}(p; \beta)$ gradually becomes a monotonically increasing function.

**Remark 2.** In the final stage of training, the output of the well-trained model should perfectly fit the posterior probability $P(y = 1 \mid x)$. Therefore, we have:

$$\hat{k}(p; \beta) \approx \frac{P(y = 1 \mid x) \cdot P(s = 0 \mid y = 1, x)}{P(s = 0 \mid x)} \cdot \frac{1}{1-P(y = 1 \mid x) \cdot P(s = 1 \mid y = 1, x)} = \frac{(1-a) \cdot p}{1-a \cdot p},$$

where we use the noise-free fact of SPML, i.e., $P(s = 1 \mid y = 0, x) = 0$. Moreover, under the Selected Completely At Random (SCAR) assumption [Bekker and Davis, 2020], $a = P(s = 1 \mid y = 1) = P(s = 1 \mid y = 1, x)$ is a constant independent of the sample and ranges between 0 and 1. As a result, $\hat{k}(p; \beta)$ is a monotonically increasing function.

**The formulation.** According to the above two assumptions, we define the form of $\hat{k}(p; \beta)$ as Logistic function:

$$\hat{k}(p; \beta) = \frac{1}{1 + \exp\{-w \cdot p + b\}},$$

where the parameters $\beta = [w, b]$. We point out that there are other choices for the form of $\hat{k}(p; \beta)$, and we use Logistic because of its simplicity and it can easily satisfy the requirements in the assumptions. In detail, when $w = 0$, $\hat{k}(p; \beta)$ is a constant, corresponding to the early-stage training. On the other hand, if increasing $w > 0$ and decreasing $b < 0$, while maintaining their ratio within a certain range, the Logistic can reflect the characteristics of late-stage training. In particular, as $w \rightarrow +\infty$, $\frac{b}{w} = -\tau$, $\hat{k}(p; \beta)$ turns into the following threshold based hard pseudo-labeling strategy:

$$\hat{k}(p; \beta) = \frac{1}{1 + \exp\{-w \cdot (p - \tau)\}} \begin{cases} 1 & \text{if } p > \tau, \\ \frac{1}{2} & \text{if } p = \tau, \\ 0 & \text{if } p < \tau. \end{cases}$$

### The update of $\beta$

Since $\hat{k}(p; \beta)$ is only used as a calibration for the model output $p$ to better estimate $k(x)$, we stop the gradient backpropagation of $p$ in $\hat{k}(p; \beta)$, and instead consider $\beta$ as a trainable parameter. However, the performance is not satisfactory if we treat $\beta$ as parameter (see the details in the last row of Table 3). Consequently, we introduce linear regularization to deterministically update $\beta$ along with the training process. In other words, we assume that both $w, b$ linearly increase with the training epochs in the following manner:

$$w(t) = w(0) + \left(w(T) - w(0)\right) \cdot \frac{t}{T},$$

$$b(t) = b(0) + \left(b(T) - b(0)\right) \cdot \frac{t}{T}.$$
The Analysis and Formulation of $v(p; \alpha)$
When the model performs well, the output confidence $p$ can be largely trusted. In this circumstance, when the corresponding observed label $s = 0$, if $p$ is close to 1, it is probably that the corresponding instance is false negative or an outlier, and in either case, its weight should be reduced.

On the other hand, inspired by focal loss [Lin et al., 2017], in the case of imbalance between positive and negative samples, the weight of simple samples should be reasonably reduced. Hence, when $s = 0$, if $p$ is close to 0, the corresponding instance is probably a simple one, and its weight should be small. Conversely, if $p$ is close to 0.5, its weight should be larger. Furthermore, because positive label is too limited and definitely correct, we set its weight to 1.

The formulation. Based on the above discussions, we define $v(p; \alpha)$ as:

$$v(p; \alpha) = \begin{cases} 1 - \frac{p \cdot (1 - p)^2}{q_1}, & \text{if } s = 1, \\ \exp\left(-\frac{(p - \mu)^2}{2\sigma^2}\right), & \text{if } s = 0. \end{cases}$$

(11)

The update of $\alpha$. Similar to $\beta$, the parameters $\alpha = [\sigma, \mu]$ are instead set as linearly updated. The value of $\alpha(t)$ evolve over the training epochs in the following manner:

$$\mu(t) = \mu(0) + (\mu(T) - \mu(0)) \cdot \frac{t}{T},$$

(12)

$$\sigma(t) = \sigma(0) + (\sigma(T) - \sigma(0)) \cdot \frac{t}{T},$$

(13)

where $\mu(0), \sigma(0)$ are initially fixed, and $\mu(T), \sigma(T)$ are two hyperparameters to be determined.

The Analysis and Formulation of $L_1, L_2, L_3$

In order to get more accurate supervision information, for missing labels, we employ $\hat{k}(p; \beta)$ as a soft pseudo label to indicate the probability of its presence. However, in the training process, we still need to deal with a large amount of noise in pseudo labels. Training with robust loss can mitigate the noise issue. As a contrast, binary cross-entropy loss is good for fitting but does not enjoy robustness. In this work, we propose to combine MAE with BCE to build a novel robust loss.

Particularly, inspired by the Generalized Cross Entropy (GCE) loss, we define the loss functions as follows:

$$L_1 = 1 - \frac{p^{q_1}}{q_1}, \quad L_2 = 1 - \frac{p^{q_2}}{q_2}, \quad L_3 = 1 - \frac{(1 - p)^{q_3}}{q_3}.$$  

(14)

Here, $q_1, q_2, q_3$ are hyperparameters to adjust the robustness of $L_1, L_2, L_3$, which can be seen as a trade-off between MAE loss and BCE loss: When $q_1 = 1, L_j$ is the MAE loss; when $q_j \rightarrow 0, L_j$ asymptotically becomes BCE. The closer $q_j$ is to 0, the lower robustness the loss exhibits, allowing faster learning convergence. Conversely, the closer $q_j$ is to 1, the stronger robustness the loss equipped, making it more effective against incorrect supervision. Thus, controlling $q_j$ can balance the convergence speed and robustness. Detailed explanations can be found in Appendix D (see the full version). Figure (1a) shows the loss curves of $L_3$ for different values of $q_3$: When $q_3 = 0.001$, the curve approximates BCE loss, and when $q_3 = 1$, it represents MAE loss.

3.4 Relation to Other SPML Losses

In this section, we summarize the existing MLML and SPML losses and show how to integrate them into our unified loss function framework.

The core of our framework is to estimate the posterior probabilities of missing labels by adopting the label confidence $p$. As discussed above, $v(p; \alpha)$ and $\hat{k}(p; \beta)$ are both functions of $p$, which fundamentally aligns with the essence of the pseudo-labeling method. Transforming the pseudo-labeling method into the proposed framework, $v(p; \alpha)$ and $\hat{k}(p; \beta)$ can be represented as:

$$\hat{k}(p; \beta) = \begin{cases} 1 & p \geq \tau_1, \\ 0 & p \leq \tau_2, \\ \text{undefined} & \text{otherwise}, \end{cases}$$  

(15)

$$v(p; \alpha) = \begin{cases} 1 & s = 0 \text{ and } \tau_2 < p < \tau_1, \\ \text{otherwise}. \end{cases}$$  

(16)

Here, $\tau_1, \tau_2$ are adaptively changed with the training epoch $t$, acting as adaptive thresholds. Besides, in existing methods, $L_1, L_2, L_3$ typically employ cross-entropy loss:

$$L_1 = -\log(p), \quad L_2 = -\log(p), \quad L_3 = -\log(1 - p).$$  

(17)

We present specific forms of components in our framework corresponding to existing MLML/SPML methods in Table 1. More explanations are in Appendix B of full version.

3.5 Gradient Analysis

To better understand the performance of the proposed GR Loss, we conduct a gradient-based analysis, which is commonly used for in-depth study of loss functions [Ridnik et al., 2021]. Observing gradients is beneficial as, in practice, network weights are updated according to the gradient of loss. For convenience, let $z = z_c$ denote the output logit of the $c$-th class for $x$, and $L_\emptyset = \hat{k} \cdot \frac{1 - p^{q_2}}{q_2} + (1 - \hat{k}) \cdot \frac{1 - (1 - p)^{q_3}}{q_3}$ represent the loss for the unannotated label. The gradients of $L_\emptyset$ w.r.t. $z$ are given by:

$$g(p) = \left(\frac{\partial L_\emptyset}{\partial z} \cdot \frac{\partial L_\emptyset}{\partial p} \cdot \frac{\partial p}{\partial z}\right) = (1 - \hat{k}) \cdot (1 - p)^{q_1} \cdot p \cdot \hat{k} \cdot \frac{1}{q_2} \cdot (1 - p),$$

(18)

where $p = \sigma(z) = (1 + e^{-z})^{-1}$ and $\hat{k}$ is defined in Eq. (7).

An intuitive interpretation of the gradient is: When $g(p) > 0$, a decrease of $p$ leads to reduction in loss; conversely, when $g(p) < 0$, an increase of $p$ results in reduction in loss. Accordingly, we analyzed the gradient of GR Loss at different epochs and the loss gradients of other SPML methods, including EM loss and Hill loss. Figure (1b) illustrates the gradients of different losses, where $\beta^{(0)}$ and $\beta^{(T)}$ represent the gradients at the beginning and the end of training, respectively. Eventually, we reach the following conclusions:

- The imbalance between positive and negative labels is a key factor to affect SPML performance. By properly adjusting $q_1$, we can balance the supervision information.
- The issue of false negatives is also a key factor to impact the gradient (see Eq. (18)), which can be effectively addressed by adjusting $\hat{k}(p; \beta)$.

A detailed derivation process and specific explanations with evidence are provided in Appendix C (see the full version).
Table 1: Representing the existing MLML/SPML losses in our unified loss function framework.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{k}(p)$</td>
<td>$\begin{cases} 0 &amp; p \leq \tau_1 \ \text{undefined} &amp; \text{otherwise} \end{cases}$</td>
<td>$\begin{cases} 0 &amp; p \leq \tau_1 \ 1 &amp; \text{otherwise} \end{cases}$</td>
<td>0</td>
<td>$\begin{cases} 0 &amp; p \leq \tau_1 \ 1 &amp; \text{otherwise} \end{cases}$</td>
<td>$\frac{1}{1+\exp\left[-w(p-b)\right]}$</td>
</tr>
<tr>
<td>$v(p)$</td>
<td>$\begin{cases} 1 &amp; \text{otherwise} \ 0 &amp; s=0 \text{ and } p &gt; \tau_1 \end{cases}$</td>
<td>$\begin{cases} 1 &amp; \text{otherwise} \ \alpha &amp; s=0 \text{ and } p &gt; \tau_1 \ \beta &amp; s=0 \text{ and } p \leq \tau_1 \end{cases}$</td>
<td>1</td>
<td>1</td>
<td>$\begin{cases} 1 &amp; s=1 \ \exp\left{-\frac{(p-b)^2}{2\sigma^2}\right} &amp; s=0 \end{cases}$</td>
</tr>
<tr>
<td>$L_1$</td>
<td>$-\log(p)$</td>
<td>$-\log(p)$</td>
<td>$-\log(p)$</td>
<td>$(1-p_m)^\gamma \log(p_m)$</td>
<td>$\frac{1-p^T}{q_1}$</td>
</tr>
<tr>
<td>$L_2$</td>
<td>$-\log(p)$</td>
<td>$p \log(p) + (1-p) \log(1-p)$</td>
<td>$\text{undefined}$</td>
<td>$(1-p_m)^\gamma \log(p_m)$</td>
<td>$\frac{1-p^2}{q_2}$</td>
</tr>
<tr>
<td>$L_3$</td>
<td>$-\log(1-p)$</td>
<td>$-\hat{p} \log((1-p)) \log(1-p)$</td>
<td>$-(\lambda-p)^2$</td>
<td>$-(\lambda-p)^2$</td>
<td>$\frac{1-(1-p)^3}{q_3}$</td>
</tr>
</tbody>
</table>

Figure 1: The gradient analysis. Better viewed in color.

4 Experiments
We provide the main empirical results of the proposed GR Loss in this section, including the comparison with state-of-the-art methods, the ablation study of our framework and robust loss, and the hyperparameter analysis. More experimental results can be found in Appendix E of the full version.

4.1 Experimental Setup
Dataset. We evaluate our proposed GR Loss on four benchmark datasets: Pascal VOC-2012 (VOC) [Everingham and Winn, 2012], MS-COCO-2014(COCO) [Lin et al., 2014], NUS-WIDE(NU) [Chua et al., 2009], and CUB-200-2011(CUB) [Wah et al., 2011]. We first simulate the single-positive label training environments commonly used in SPML [Cole et al., 2021], and replicate their training, validation and testing samples. In these datasets, only one positive label is randomly selected for each training instance, while the validation and test sets remain fully labeled. More dataset descriptions are provided in Appendix E.1 of full version.

Implementation details and hyperparameters. For fair comparison, we follow the mainstream SPML implementation [Cole et al., 2021]. In detail, we employ ResNet-50 [He et al., 2016] architecture, which was pre-trained on ImageNet dataset [Russakovsky et al., 2015]. Each image is resized into 448×448, and performed data augmentation by randomly flipping an image horizontally. We initially conduct a search to determine and fix the hyperparameters $q_0$ and $q_3$ in Eq.(14), typically 0.01 and 1, respectively. Because the robust loss has a significant impact on training, which is also reflected in Table 3. Therefore, we only need to adjust four hyperparameters in $(\beta^T, \alpha^T)$. More details about hyperparameter settings are described in Appendix E.2 of our full version.

Comparing methods. In our empirical study, we compare our method to the following state-of-the-art methods: AN loss (assuming-negative loss) [Cole et al., 2021], AN-LS (AN loss combined with Label Smoothing) [Cole et al., 2021], Focal loss [Lin et al., 2017], ROLE (Regularised Online Label Estimation) [Cole et al., 2021], Hill loss [Zhang et al., 2021], SPLICL (Self-Paced Loss Correction) [Zhang et al., 2021], EM (Entropy-Maximization Loss) [Zhou et al., 2022a], EM+APL (Asymmetric Pseudo-Labeling) [Zhou et al., 2022a], SMILE (Single-positive Multi-label learning with Label Enhancement) [Xu et al., 2022], Large Loss (LL-R, LL-C, LL-Cp) [Kim et al., 2022], and MIMIE [Liu et al., 2023b]. The goal of all the above methods is to propose new SPML loss function, which is consistent with ours. Indeed, we are also concerned that there are methods that adopt pre-training strategies or use different backbones, such as: LAGC [Xie et al., 2022], DualCoop [Sun et al., 2022], HSPNet [Wang et al., 2023], CRISP [Liu et al., 2023a]. Due to the length limitation and in order to conduct a fair comparison, we only consider the former loss function modification approach in this study.

4.2 Results and Discussion
The experimental results of most existing MLML and SPML methods on four SPML benchmarks are reported in Table 2. It is observed that AN loss performs the worst almost on all four datasets, indicating that the false negative noise introduced by AN assumption has a significant negative impact on SPML. Meanwhile, Focal Loss, as a baseline method, is ineffective for imbalance in SPML, especially the inter-class imbalance.

Notably, our GR Loss outperforms existing methods in the first three SPML benchmarks, i.e., VOC, COCO, NUS, and achieves the second-highest result on CUB, slightly lower than MIME. The main reason for the second best on CUB is, our method treats SPML as C multiple binary classification tasks and the more categories means the more positive labels for each image, thus the stronger correlation between labels, leading to inferior performance of our method. In detail, for VOC (20 classes), the mAP is 0.63% higher than the second-best, and on COCO (80 classes) and NUS (81 classes), the improvements are 0.25% and 0.34%, respectively. However, there are 312 classes in CUB and an average of 31.5 positive
classes per image. As a result, our method is slightly worse than MIME, which considers label correlation and inter-class differences.

4.3 Ablation Study
In order to present an in-depth analysis on how the proposed method improves SPML performance, we conduct thorough ablation study on VOC and COCO and report the results in Table 3. It is observed that using only $\hat{k}(p; \beta)$ results in the greatest improvement, almost a 1% increase compared to using the other two items alone. Utilizing both $\hat{k}(p; \beta)$ and $L_1$, $L_2$, $L_3$ leads to a significant enhancement, with mAP reaching 89.35 and 72.86. Moreover, adding $v(p; \alpha)$ can further elevate the model’s performance, with mAP improving to 89.83 and 73.17, respectively. The last row indicates that treating $\beta$ and $\alpha$ as trainable parameters obtains inferior results, which verifies our analysis.

4.4 Hyperparameter and Distinguishability
The impact of $\beta$. We present the results of GR Loss with different $\beta(T)$ in Figure (2a). To facilitate the analysis, we adjust the threshold $\tau = -b/w$ while keeping $b(T)$ fixed at -2, -5, and -8, respectively. The results indicate that our model consistently achieves the optimal when $\tau = 1$, achieving a best mAP at 89.83 with $b = -2$. As the threshold decreasing, the performance gradually declines. However, it is still superior to the baseline, i.e., vanilla AN loss (the red dotted line).

The impact of $\alpha$. We present GR Loss performance with different $\alpha(T)$ in Figure (2b). By fixing $\sigma(T)$ at 0.1, 0.5, and 1, and varying the mean $\mu$ from 0.2 to 1, all the three curves peak at 0.8. Moreover, the mAPs for all parameter combinations are above 89 and mostly surpass both EM loss and AN loss, which indicating our method enjoys promising robustness.
Distinguishability of model predictions. A model with promising generalization should be capable of producing informative predictions for unannotated labels, i.e., the predicted probabilities for positive and negative labels should be clearly distinguishable for unannotated labels. Hence, we assess the confidence of model’s predictions on validation set for all classes and all unannotated labels at each epoch, and quantitatively measure the confidence distribution difference for positive and negative labels with Wasserstein distance. As shown in Figure (3a), the Wasserstein distances for GR Loss are much greater than those for AN loss at each epoch. Besides, as shown in Figure (3c), compared with AN loss in Figure (3b), for GR Loss, the confidence of negative labels is concentrated around 0.2, while that of positive labels is evenly distributed and becomes extremely low when smaller than 0.9, mainly clustering around 1.

5 Conclusion

In this paper, we proposed a novel loss function called Generalized Robust Loss (GR Loss) for SPML. We employ a soft pseudo-labeling mechanism to compensate the lack of labels. Meanwhile, we specifically design a robust loss to cope with the noise introduced in pseudo labels. In addition, we introduce two tunable functions $\hat{k}(.; \beta)$ and $v(.; \alpha)$ to calibrate model output $p$, so that to simultaneously deal with the intra-class and inter-class imbalance. Furthermore, we demonstrate the validity of GR Loss from both theoretical and experimental perspectives. From the theoretical perspective, we derive empirical risk estimates for SPML and perform a gradient analysis. Besides, in experiments, our method mostly achieves state-of-the-art results on all four benchmarks.

Nevertheless, there are three limitations in our study. First, $q_1, q_2, q_3$ in the robust loss Eq.(14) are all set as static hyperparameters. However, it is better to set the robust loss to be dynamically varying so that to balance the fitting ability and robustness. Second, as mentioned earlier, we did not evaluate our GR Loss on different backbones to fully validate its effectiveness. Third, our method does not consider the correlation among classes, resulting in an inferior performance compared with MIME on CUB. To overcome these limitations will shed light on the promising improvement of our current work.
Acknowledgements
This work was supported by National Natural Science Foundation of China (No.61702099), UIBE Excellent Young Scholar Project (21YQ10), and Scientific Research Laboratory of AI Technology and Applications, UIBE.

Contribution Statement
The joint first authors Yanxi Chen and Chunxiao Li are equally contributed to formulation and experiments. In addition, Xinyang Dai, Jinluan Li, WeiYu Sun, Yiming Wang, Renyuan Zhang, and Tinghe Zhang are equally contributed to writing and experiments. Authors are sorted alphabetically.

References


