Partial Multi-Label Learning via Probabilistic Graph Matching Mechanism

Gengyu Lyu
Beijing Key Laboratory of Traffic Data Analysis and Mining, Beijing Jiaotong University
Beijing 100044, China
18112030@bjtu.edu.cn

Songhe Feng
Beijing Key Laboratory of Traffic Data Analysis and Mining, Beijing Jiaotong University
Beijing 100044, China
shfeng@bjtu.edu.cn

Yidong Li*
School of Computer and Information Technology, Beijing Jiaotong University
Beijing 100044, China
ydli@bjtu.edu.cn

ABSTRACT

Partial Multi-Label learning (PML) learns from the ambiguous data where each instance is associated with a candidate label set, where only a part is correct. The key to solve such problem is to disambiguate the candidate label sets and identify the correct assignments between instances and their ground-truth labels. In this paper, we interpret such assignments as *instance-to-label* matchings, and formulate the task of PML as a matching selection problem. To model such problem, we propose a novel grap**H** m**A**tching based partial mu**L**ti-label l**E**arning (**HALE**) framework, where *Graph Matching* scheme is incorporated owing to its good performance of exploiting the instance and label relationship. Meanwhile, since conventional *one-to-one* graph matching algorithm does not satisfy the constraint of PML problem that multiple instances may correspond to multiple labels, we extend the traditional probabilistic graph matching algorithm from *one-to-one* constraint to *many-to-many* constraint, and make the proposed framework to accommodate to the PML problem. Moreover, to improve the performance of predictive model, both the minimum error reconstruction and k-nearest-neighbor weight voting scheme are employed to assign more accurate labels for unseen instances. Extensive experiments on various data sets demonstrate the superiority of our proposed method.

CCS CONCEPTS

• **Computing methodologies → Machine learning**.

KEYWORDS

partial multi-label learning; 'instance-to-label' matching; matching selection; graph matching; 'many-to-many' constraint

ACM Reference Format:
Gengyu Lyu, Songhe Feng, and Yidong Li. 2020. Partial Multi-Label Learning via Probabilistic Graph Matching Mechanism. In *26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394486.3403053

*Corresponding author.

KDD '20, August 23–27, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3403053

1 INTRODUCTION

In Partial Multi-Label learning (PML), each training instance is associated with a set of candidate labels, among which only a part is correct [30]. Recently, the need to learn from PML data naturally rises in many real-world scenarios. For example, in crowdsourcing image tagging (Figure 1), given a group of images together with corresponding candidate label sets, where some of labels (*cloud* and *people*) are incorrectly annotated owing to potential unreliable annotators, PML aims to learn an accurate multi-label classifier from such ambiguous training data and assign a set of proper labels for the unseen instances.

Evidently, the major difficulty to learn from PML data lies in that the ground-truth labels of each training instance are concealed in its candidate label set and not directly accessible to the learning algorithm. Thus, the common strategy to learn from PML data is disambiguation, i.e. dislodging the noisy labels from the candidate label set and then utilize the relatively credible labels for model induction. Following such strategy, some concrete approaches towards PML problem are proposed and can be roughly grouped into two categories: Xie et al. [30] and Sun et al. [25] disambiguate the candidate label set by assigning a confidence value for each candidate label, and then optimize it in an iterative manner. Fang et al.[8] and Wang et al. [27] also follow the disambiguation strategy and divide the training process into two stages, where the higher-level confidence labels are first selected from the ambiguous candidate label set, and then incorporated into some off-the-shelf MLL frameworks to induce the PML model. However, the above PML methods conduct the disambiguation operation by only utilizing either the instance correlations, label correlations, or both of them, while the direct assignment correlations between instances and labels are hardly taken into consideration, where such potential *instance-label* assignment correlations tends to have great contribution to improve the disambiguation capability of learning model.

In light of this observation, in this paper, we propose a novel grap**H** m**A**tching based partial mu**L**ti label l**E**arning (**HALE**) method, where such *instance-label* assignment correlations together with instance correlations and label correlations are simultaneously incorporated into the proposed framework. Specifically, we regard the correspondences between instances and their candidate labels as the instance-label matchings, and then reformulate the task of PML as an *instance-label* matching selection problem. Motivated by [3] [24], graph matching scheme is incorporated to solve such matching selection problem owing to its good performance on utilizing structural information of training data. However, existing graph matching algorithms are formulated with *one-to-one* constraint,

Figure 1: An exemplar of partial multi-label learning. In crowdsourcing image tagging, among the 7 candidate labels, only 5 of them are valid including *building*, *tree*, *car*, *sky* and *road*, while 2 of them are false including *cloud* and *people*.

which is not fully in accordance with the original task of PML problem that multiple instances can correspond to multiple labels. Therefore, we extend such *one-to-one* constraint to *many-to-many* constraint and propose a many-to-many probabilistic matching algorithm to make our method accommodate to the original PML problem. Moreover, to improve the predicted accuracy of learning model, both minimum error reconstruction scheme and k-nearest-neighbor weight voting scheme are simultaneously employed to assign more accurate labels for unseen examples. Extensive experiments demonstrate that our proposed method can achieve superior performance than state-of-the-art methods.

2 RELATED WORK

As a novel weakly supervised learning framework, partial multi-label learning can be regarded as an integration of *multi-label learning* [37] and *partial label learning* [4].

2.1 Multi-Label Learning (MLL)

Multi-label Learning aims to learn a multi-class classifier from the training data where each instance is associated with multiple valid labels [14]. Based on the order of correlations being exploited for model training, existing standard MLL methods can be roughly characterized into three categories: *first-order strategy*, *second-order strategy* and *high-order strategy*. For the first-order strategy, the MLL problem is decomposed into multiple binary classification problems [1] [33], where the classifier for each label is trained independently. For the second-order strategy, pairwise label correlations are considered, where the ranking between the relevant and irrelevant label [11], or any pair of labels [19] is often taken into consideration. For the high-order strategy, higher-level label correlations are considered, such as imposing all other labels' influences on each possible label [2] [17]. Recently, some weakly supervised MLL frameworks are proposed and most of them focus on solving the MLL problem with missing labels, such as [26] [28] [32].

2.2 Partial Label Learning (PLL)

Partial-label learning aims to induce a multi-class predictive model from the training data, where each instance is associated with a candidate label set, among which only one is ground-truth label [23] [10]. Existing methods to deal with such problem can be roughly grouped into three categories: *Averaging Disambiguation Strategy*, *Identification Disambiguation Strategy* and *Disambiguation-Free Strategy*. *Averaging Disambiguation Strategy*-based PLL methods usually treat each candidate label equally and they make prediction for unseen instances by averaging the outputs from all candidate labels [13] [34]. *Identification Disambiguation Strategy*-based PLL methods often view the ground-truth label as a latent variable first, and then refine the model parameter in an iterative manner [9] [16] [20] [24]. *Disambiguation-Free Strategy*-based methods learns from the partial label data by incorporating off-the-shelf learning techniques and they directly make prediction for unseen instance without any disambiguation operations [29] [35].

2.3 Partial Multi-label Learning (PML)

Partial multi-label learning learns from the ambiguous data, where partial labels in the candidate label set are correct [31]. Some of existing methods learns from the PML data by estimating confidence of each candidate label, and then incorporate the estimated confidence scores into an alternative optimization procedure for model induction [25] [30]. Others decompose the training process into two stages, where the high-level confidence labels are first selected from the candidate label set, and then employed for training the desired model via some off-the-shelf MLL methods[8] [27].

In this paper, from a completely new perspective, we formulate the task of PML into an *instance-label* matching selection problem, and propose a novel probabilistic matching algorithm to solve it.

3 THE PROPOSED METHOD

Formally speaking, we denote the d-dimensional feature space as $\mathcal{X} \in \mathbb{R}^d$, and the label space as $\mathcal{Y} = \{1, 2, \ldots, q\}$ with q class labels. PML aims to learn a classifier $\mathbf{f} : \mathcal{X} \mapsto \mathcal{Y}$ from the PML training data $\mathcal{D} = \{(\mathbf{x}_i, S_i)\}(1 \le i \le m)$, where the instance $\mathbf{x}_i$ is described as a d-dimensional feature vector, the candidate label set $S_i \subseteq \mathcal{Y}$ is associated with the instance $\mathbf{x}_i$ and m is the number of instances. In addition, we denote $\tilde{S}_i \subseteq S_i$ as the ground-truth label set for instance $\mathbf{x}_i$, and $\mathbf{y}_i, \tilde{\mathbf{y}}_i \in \{0, 1\}^{q \times 1}$ as the vector format of S_i and $\tilde{S}_i$, where each S_i (i.e. $\mathbf{y}_i$) corresponding to $\mathbf{x}_i$ is not directly accessible to the algorithm.

3.1 Formulation

HALE is a novel PML framework based on probabilistic graph matching scheme, which aims to fully explore the *instance-label* assignment correlations from the ambiguous PML data and establish an accurate assignment relationship between the instance space $\mathcal{X}$ and label space $\mathcal{Y}$. Although such strategy has been employed for PLL [24], to the best of our knowledge, this is the first attempt to resolve PML problem by graph matching strategy. To make the proposed method easily understanding, we illustrate the HALE method as a graph matching structure before the following detailed introduction.

As depicted in Figure 2, both the instance space and label space are formulated as two different undirected graphs $\mathbb{G}^i = (\mathbb{V}^i, \mathbb{E}^i)$ of size n_i, where $i \in \{1, 2\}$, and $n_1 = m$, $n_2 = q$. The nodes $\mathbb{V}^i$ in the two graphs represent the instances and labels respectively, while the edges $\mathbb{E}^i$ encode their correlations. The goal of HALE is to establish the graph nodes correspondence between $\mathbb{G}^1$ and $\mathbb{G}^2$.

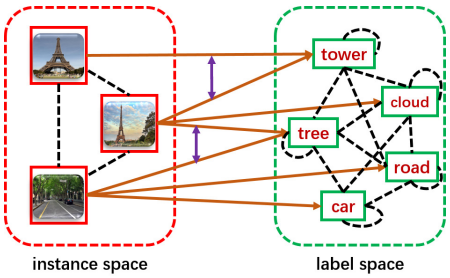

Figure 2: The graph matching structure of HALE. For convenience, we only illustrate remarkable labels for each image.

Here, we first denote the $\mathbf{A}^i$ as the adjacent matrix for each graph $\mathbb{G}^i$, where $i \in \{1, 2\}$. $\mathbf{A}^1 \in \mathbb{R}^{m\times m}$ encodes the instance similarity, which can be constructed by the symmetry-favored k-NN graph [21],

$$A^1_{ij} = \begin{cases} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{\frac{1}{m}\left(\sum_{i=1}^m \|\mathbf{x}_i - \mathbf{x}_{i_k}\|_2^2\right)}\right), & j \in \mathcal{N}_i, \\ 0, & \text{otherwise}. \end{cases} \tag{1}$$

Here, $\mathcal{N}_i$ saves the indices of the k-nearest neighbors of $\mathbf{x}_i$, and $\mathbf{x}_{i_k}$ represents the k-th nearest neighbor of $\mathbf{x}_i$. To ensure that $\mathbf{A}^1$ is symmetric, we further set $\mathbf{A}^1 = (\mathbf{A}^1 + \mathbf{A}^{1\top})/2$. $\mathbf{A}^2 \in \mathbb{R}^{q\times q}$ encodes the label similarity, which is built via the label co-occurrence in the training data:

$$A^2_{i'j'} = \frac{1}{m}\sum_{c=1}^{m} \mathbb{I}(Y_{i'c} = 1, Y_{j'c} = 1), \tag{2}$$

where $\mathbf{Y} \in \mathbb{R}^{q\times m}$ denotes the candidate label matrix for training instances, $\mathbb{I}(\triangle)$ is the indicator function and $\mathbb{I}(\triangle) = 1$ if $\triangle$ is true, $\mathbb{I}(\triangle) = 0$ otherwise.

Then, we define $\mathbf{P} \in \{0, 1\}^{m\times q}$ to describe the graph node correspondence between $\mathbb{G}^1$ and $\mathbb{G}^2$, where $P_{ij} = 1$ represents label j is assigned to instance $\mathbf{x}_i$, and $P_{ij} = 0$ otherwise. Among these correspondences that $P_{ij} = 0$, most of them are invaluable since label j is not contained in the candidate label set of instance $\mathbf{x}_i$. To reduce the complexity of learning model and establish the *instance-label* matching model conveniently, we remove the assignments between instances and their non-candidate labels, and obtain the row-wise vectorized replica $\mathbf{p} = [p_1, p_2, \ldots, p_u]^\top \in \mathbb{R}^{u\times 1}$, where each element of $\mathbf{p}$ is defined as:

$$p_e = \{\mathbf{x}_{i_e}, y_{l_e}\}, \tag{3}$$

here $i_e \in \{1, 2, \ldots, m\}$, $l_e \in \{1, 2, \ldots, |S_i|\}$, $|S_i|$ is cardinality of S_i. In addition, $e \in \{1, 2, \ldots, u\}$, $u = \sum_{i=1}^m |S_i|$ and $\{\mathbf{x}_{i_e}, y_{l_e}\}$ represents the confidence value of instance $\mathbf{x}_{i_e}$ assigned with its l_e-th candidate label.

Afterwards, motivated by [22], the correspondence between instances and their corresponding candidate labels can be obtained by solving the following optimization problem **OP (1)**:

$$\begin{aligned} \mathbf{p}^* &= \arg\max_{\mathbf{p}} \mathbf{p}^\top \mathbf{K}\mathbf{p} \\ s.t. \quad & \mathbf{p} \in \{0, 1\}^{u\times 1} \\ & 1 \le \sum_{j=1}^{q} P_{ij} \le |S_i|, \quad \forall i \in [m]. \end{aligned} \tag{4}$$

where $\mathbf{K} \in \mathbb{R}^{u\times u}$ is the affinity matrix that encodes the *instance-label* assignment correlations, i.e. $K_{ab} = \{p_a, p_b\} = \{\{\mathbf{x}_{i_a}, y_{l_a}\}, \{\mathbf{x}_{i_b}, y_{l_b}\}\}$. Here, $a, b \in \{1, 2, \ldots, u\}$, $\{\mathbf{x}_{i_a}, y_{l_a}\}$ represents the value of a-th element of $\mathbf{p}$ that indicating the correspondence between i_a-th instance $\mathbf{x}_{i_a}$ and its l_a-th candidate label y_{l_a}. Furthermore, K_{ab} can be initialized as

$$K_{ab} = A^1_{i_a j_b} \cdot A^2_{i'_a j'_b}. \tag{5}$$

It is worth noting that, compared with conventional PML methods, the proposed framework employs not only instance similarity but also label correlations, as well as *instance-label* assignment consistency, which leads the learning model to obtain more accurate instance-label matchings during the whole learning process.

3.2 Optimization

In this section, we extend the probabilistic graph matching scheme from [6] and derive a probabilistic graph matching PML algorithm, which can avoid the **OP (1)** falling into trivial solutions. The core of the proposed algorithm is based on the observation that we can use the solution of the spectral matching algorithm [18] to refine the estimation of the affinity matrix $\mathbf{K}$ and then solve a new assignment problem based on the refined matrix $\mathbf{K}$.

Concretely, we relax the first constraint of **OP (1)** to $\mathbf{p} \in [0, 1]^{u\times 1}$ and interpret $\mathbf{p}$ as matching probabilities $P(\{\mathbf{x}_i, y_l\})$. Then, the affinity matrix $\mathbf{K}$ can be further interpreted as a joint matching probability matrix, i.e. $K_{ab} = P(\{\mathbf{x}_{i_a}, y_{l_a}\}, \{\mathbf{x}_{i_b}, y_{l_b}\})$. Afterwards, we can iteratively refine $\mathbf{K}$ and $\mathbf{p}$ by solving the following problem **OP (2)**:

$$\begin{aligned} &\min_{K_{ab}, p_a} \sum_a \Big((\sum_b K_{ab}) - p_a\Big)^2 \\ =& \min_{p_{(a|b)}, p_a} \sum_a \Big((\sum_b p_{(a|b)} \cdot p_b) - p_a\Big)^2 \end{aligned} \tag{6}$$

where p_a is the assignment probability $P(\{\mathbf{x}_{i_a}, y_{l_a}\})$ and $p_{(a|b)}$ represents the conditional assignment probability $P(\{\mathbf{x}_{i_a}, y_{l_a}\}|\{\mathbf{x}_{i_b}, y_{l_b}\})$, which is the probability of assignment $\{\mathbf{x}_{i_a}, y_{l_a}\}$ when $\{\mathbf{x}_{i_b}, y_{l_b}\}$ is valid.

Note that, in **OP (2)**, the joint matching probability in $\mathbf{K}$ is not directly optimized as it cannot be easily updated: Having a high assignment probability $P(\{\mathbf{x}_{i_a}, y_{l_a}\}) \approx 1$ does not imply that $P(\{\mathbf{x}_{i_a}, y_{l_a}\}, \{\mathbf{x}_{i_b}, y_{l_b}\}) \approx 1$, as we might have $P(\{\mathbf{x}_{i_b}, y_{l_b}\}) \approx 0$. In contrast, $P(\{\mathbf{x}_{i_a}, y_{l_a}\}|\{\mathbf{x}_{i_b}, y_{l_b}\})$ is asymmetric, and given that $P(\{\mathbf{x}_{i_a}, y_{l_a}\}) \approx 1$, we can increase $P(\{\mathbf{x}_{i_a}, y_{l_a}\}|\{\mathbf{x}_{i_b}, y_{l_b}\})$ regardless of $P(\{\mathbf{x}_{i_b}, y_{l_b}\})$.

Next, we optimize the p_a and $p_{(a|b)}$ in an iterative manner. Specifically, in t-th iteration, we denote the estimations of $P^{(t)}(\{\mathbf{x}_{i_a}, y_{l_a}\}|\{\mathbf{x}_{i_b}, y_{l_b}\})$ by $p^{(t)}_{(a|b)}$ and $P^{(t)}(\{\mathbf{x}_{i_a}, y_{l_a}\})$ by $p^{(t)}_a$, respectively. Then, $p^{(t)}_{(a|b)}$ and $p^{(t)}_a$ can be separately updated following

$$p^{(t+1)}_a = \sum_b p^{(t)}_{(a,b)} = \sum_b p^{(t)}_{(a|b)} \cdot p^{(t)}_b, \tag{7}$$

and

$$p^{(t+1)}_{(a|b)} = p^{(t)}_{(a|b)} \cdot \frac{p^{(t+1)}_a}{p^{(t)}_a}. \tag{8}$$

Algorithm 1 The Training Algorithm of **HALE**

Inputs:
$\mathcal{D}$: the partial multi-label training set $\{(\mathbf{x}_i, S_i)\}$;
Process:
1. Calculate instance similarity matrix $\mathbf{A}^1$ and label similarity matrix $\mathbf{A}^2$ according to Eq (1) and Eq (2);
2. Calculate the affinity matrix $\mathbf{K}$ by Eq (5);
3. Set $\mathbf{K}^{(0)} = \mathbf{K}$ and $\mathbf{p}^{(0)} = \frac{1}{|S_i|}\mathbf{1}$ where $\mathbf{p}^{(0)} \in \mathbb{R}^{u\times 1}$;
4. for $t = 0$ **to** *iter*
5. $\quad \mathbf{p}^{(t+1)} = \mathbf{K}^{(t)}\mathbf{p}^{(t)}$;
6. $\quad \mathbf{p}^{(t+1)} = \text{Normalize}(\mathbf{p}^{(t+1)})$;
7. $\quad \mathbf{K}^{(t+1)}(a,b) = \mathbf{K}^{(t)}(a,b) \cdot (p_a^{(t+1)}/p_a^{(t)})$;
8. $\quad$ **if** $(||\mathbf{p}^{(t+1)} - \mathbf{p}^{(t)}||_2) < \delta$
9. $\quad\quad$ break;
10. $\quad$ **end if**
11. end for
12. Discretize $\mathbf{p}^{(t+1)}$, and derive the assignment $(\mathbf{x}_i, \tilde{S}_i)$;
Output:
$\tilde{S}_i$: the assigned labels for $\mathbf{x}_i$;

In order to explicitly formulate the HALE with *many-to-many* constraint and avoid the trivial solutions, we normalize $p_a^{(t+1)}$ in each optimization iteration following:

$$p_{a_c}^{(t+1)} = \frac{p_{a_c}^{(t+1)}}{\max\{p_{a_1}^{(t+1)}, p_{a_2}^{(t+1)}, \ldots, p_{a_{|S_i|}}^{(t+1)}\}}. \tag{9}$$

Here, $p_{a_c}^{(t+1)}$ is the element of *instance-label* assignment confidence subvector $[p_{a_1}^{(t+1)}, p_{a_2}^{(t+1)}, \ldots, p_{a_{|S_i|}}^{(t+1)}]$, which is separated from the assignment confidences between varying candidate labels and the same (i-th) instance.

During the entire process of optimization, we first initialize the required variables, and then repeat the above steps until the algorithm converges. Algorithm 1 summarized the pseudo-code of the proposed HALE.

3.3 Proof

In this subsection, inspired by [6], we show that the two-step iterative scheme presented in Eq. (7) and Eq. (8) monotonically reduces the objective function in **OP (2)**. The proof has two parts, the first derives a result that is used in the second part.

3.3.1 The First Step. Eq. (7) is a single iteration of the Power Iteration scheme that converges in the Frobenius norm, and thus decreases the objective function for each entry of $\mathbf{p}^{(t)}$.

$$\begin{aligned}\left(\left(\sum_b p_{(a|b)}^{(t)} p_b^{(t+1)}\right) - p_a^{(t+1)}\right)^2 \leq& \left(\left(\sum_b p_{(a|b)}^{(t)} p_b^{(t)}\right) - p_a^{(t)}\right)^2 \\ =& \left(p_a^{(t+1)} - p_a^{(t)}\right)^2.\end{aligned} \tag{10}$$

Denote by $S^{(t)} = \sum_b p_{(a|b)}^{(t)} p_b^{(t+1)}$, hence

$$\left(S^{(t)}\right)^2 - 2p_a^{(t+1)}S^{(t)} \leq \left(p_a^{(t)}\right)^2 - 2p_a^{(t+1)}p_a^{(t)}. \tag{11}$$

Assume $p_a^{(t+1)} \geq p_a^{(t)}$, then

$$\left(S^{(t)} - p_a^{(t)}\right)\left(S^{(t)} + p_a^{(t)} - 2p_a^{(t+1)}\right) \leq 0. \tag{12}$$

As $p_a^{(t+1)} > p_a^{(t)} \leq 0$ and $S^{(t)} \geq 0$, then

$$S^{(t)} - p_a^{(t)} > S^{(t)} + p_a^{(t)} - 2p_a^{(t+1)} \tag{13}$$

and

$$S^{(t)} - p_a^{(t)} > 0, \quad S^{(t)} + p_a^{(t)} - 2p_a^{(t+1)} < 0 \tag{14}$$

3.3.2 The Second Step. Eq. (8) also decreases the objective function. Namely, we aim to show that

$$\begin{aligned}\left(\left(\sum_b p_{(a|b)}^{(t)} p_b^{(t+1)}\right) - p_a^{(t+1)}\right)^2 \geq& \left(\left(\sum_b p_{(a|b)}^{(t+1)} p_b^{(t+1)}\right) - p_a^{(t+1)}\right)^2 \\ =& \left(\left(\sum_b p_{(a|b)}^{(t)} \frac{p_a^{(t+1)}}{p_a^{(t)}} p_a^{(t+1)}\right) - p_a^{(t+1)}\right)^2.\end{aligned} \tag{15}$$

Simplifying the above expression we get

$$S^{(t)}\left(\left(\frac{p_a^{(t+1)}}{p_a^{(t)}}\right)^2 - 1\right) - 2p_a^{(t+1)}\left(\frac{p_a^{(t+1)}}{p_a^{(t)}} - 1\right) \leq 0. \tag{16}$$

Assuming $p_a^{(t+1)} > p_a^{(t)}$ as before, we have that $\frac{p_a^{(t+1)}}{p_a^{(t)}} - 1 > 0$. Thus,

$$\begin{aligned}0 \geq& S^{(t)}\left(\frac{p_a^{(t+1)}}{p_a^{(t)}} + 1\right) - 2p_a^{(t+1)} \\ =& S^{(t)}\left(\frac{p_a^{(t+1)}}{p_a^{(t)}}\right) + S^{(t)} - 2p_a^{(t+1)} \geq S^{(t)} + p_a^{(t)} - 2p_a^{(t+1)}.\end{aligned} \tag{17}$$

Eq. (17) is validated by the first part of the proof (Eq. (14)), which implies the reduction of the objective function in **OP (2)**. The proof of the complementary case ($p_a^{(t+1)} < p_a^{(t)}$) can be derived mutatis mutandis.

In addition, during the optimization process, we normalize the vector $\mathbf{p}$ to satisfy the constraint of PML task and avoid the trivial solution. According to the illustrations in [15], such normalization operations does not hamper the convergence properties of our algorithm, since each of these operations can be considered a projection operator onto a closed and convex set.

3.4 Prediction

During the testing phase, the class label of each unseen instance $\mathbf{x}_i^*$ is predicted based on the disambiguated training examples $\{\mathbf{x}_i, \tilde{\mathbf{y}}_i\}$, where both minimum error reconstruction scheme and k-nearest-neighbor weight voting scheme are simultaneously incorporated to improve the predictive accuracy of the learning model.

Specifically, we calculate the k-nearest-neighbor weights $\mathbf{w}_i \in \mathbb{R}^{k\times 1}$ for each unseen instance $\mathbf{x}_i^*$ via minimum error reconstruction

scheme **OP (3)**:

$$w_{i_c}^* = \min_{w_{i_c}} \left\| \mathbf{x}_i^* - \sum_{c=1}^{k} w_{i_c} \cdot \mathbf{x}_{i_c} \right\|_2^2 \quad s.t. \quad w_{i_c} \geq 0, \quad \sum_{c=1}^{k} w_{i_c} = 1, \quad (\mathbf{x}_{i_c} \in \mathcal{N}(\mathbf{x}_i^*), 1 \leq c \leq k), \tag{18}$$

here, w_{i_c} is an element of $\mathbf{w}_i$ and $c \in \{1, 2, \ldots, k\}$. Thereafter, the unseen instance is classified by

$$\mathbf{y}_i^* = \sum_{c=1}^{k} w_{i_c} \cdot \tilde{\mathbf{y}}_{i_c}, \tag{19}$$

where $\tilde{\mathbf{y}}_{i_c} \in \mathbb{R}^{q \times 1}$ denotes the class vector of k-nearest-neighbor instance $\mathbf{x}_{i_c}$ and $\mathbf{y}_i^*$ represents the classification results of $\mathbf{x}_i^*$.

4 EXPERIMENT

4.1 Experimental Setup

To effectively evaluate the performance of the proposed HALE method, we implement experiments on 9 synthetic PML data sets and 3 real-world data sets, where the synthetic PML data sets are generated from the widely-used MLL data sets by randomly adding labeling noise under different configurations of the controlling parameter r. Here, $r \in \{1, 2, 3\}$ represents the average number of false candidate labels for each training example, and the candidate label set consists of relevant labels along with irrelevant labels that are randomly chosen from non-*ground-truth* label set. For the real-world data sets, candidate labels are collected from web users which are further examined by human labelers to specify the ground-truth labels. Table 1 summarizes the characteristics of these employed experimental data sets.

Table 1: Characteristics of the experimental data sets. For each PML data set, the number of examples (EXPs*), features (FEAs*), class labels (CLs*), the maximum number of ground-truth labels (M-GT*), the average number of ground-truth labels (A-GT*) and its corresponding domain (DOM*) are recorded. The last three PML data sets are real-world data sets.

Data set	EXPs*	FEAs*	CLs*	M-GT*	A-GT*	DOM*
Emotions	593	72	6	3	1.86	music
Birds	645	260	19	6	1.86	audio
Medical	978	1,449	45	3	1.25	text
Image	2,000	294	5	3	1.23	images
Scene	2,407	294	6	3	1.07	images
Bibtext	7,395	1,836	159	28	2.40	text
Eurlex-dc	19,348	5,000	412	7	1.01	text
Eurlex-sm	19,348	5,000	201	12	1.53	text
NUS-WIDE[1]	133,441	500	81	20	1.76	images
Music-emotion	6,833	98	11	7	2.42	music
Music-style	6,839	98	10	4	1.44	music
Mirflickr	10,433	100	7	5	1.77	images

[1] The original number of instances is 269,648 but some of them are unlabeled w.r.t the 81 class labels, thus we only utilized the remaining 133,441 instances to conduct experiments.

Meanwhile, we employ seven methods from three categories for comparative studies, including MLL methods [**ML-KNN** [36], **RankSVM** [7]], PLL methods [**IPAL** [34], **LALO** [9]], and PML methods [**PML-fp** [30], **PML-lc** [30], **PARTICLE** [8]], where the configured parameters are utilized via the suggestions in respective literatures. In addition, five popular multi-label metrics are employed to evaluate each comparing method, including *Hamming Loss*, *Ranking Loss*, *One-Error*, *Coverage* and *Average Precision*, whose detailed definitions can be found in [12]. Finally, we adopt ten-fold cross-validation to train the desired model and record the experimental results on each data set in Table 2 and Table 3.

4.2 Experimental Results

Due to page limit, we partially report the experimental results on synthetic data sets in Table 3, where the parameter is configured with $r = 3$, and the similar observations can be made when the data set is built under the configurations of $r = 1$ and $r = 2$. Table 2 summarizes the resulting win/tie/loss counts over 9 synthetic data sets and 5 evaluation metrics. Meanwhile, we also report the experimental results on real-world data sets in Table 4. Out of 150 statistical comparisons, the following observations can be made:

- For each comparing method, HALE separately achieves superior or comparable performance against tailored MLL and PLL methods in 85.5% and 89.4% cases. And, it also outperforms the counterpart PML methods in 84.0% cases.
- For each evaluation metric, HALE is superior or comparable to other comparing methods in 95.2% cases (*Hamming Loss*), 73.5% cases (*Ranking Loss*), 84.1% cases (*One Error*), 77.8% cases (*Coverage*) and 96.8% cases (*Average Precision*).
- For each data set, HALE outperforms most of comparing methods over 4/5 evaluation metrics. Particularly, on *Scene* data set, HALE achieves the best performance on all evaluation metrics.
- For large-scale data sets (such as *NUS-WIDE*), HALE can not only effectively learn from such large-scale data, but also achieve superior performance on most evaluation metrics.

In order to comprehensively evaluate the superiority of the proposed HALE, *Friedman test* [5] is utilized as the statistical test to analyze the relative performance among the comparing algorithms. According to Table 5, the null hypothesis of distinguishable performance among the comparing algorithms is rejected at 0.05 significance level. Therefore, we further employ the post-hoc Bonferroni-Dunn test [5] to show the relative performance among the comparing algorithms. Figure 4 illustrates the CD diagrams on each evaluation metric, where the average rank of each comparing algorithm is marked along the axis. According to Figure 4, it is observed that HALE performs significantly superiority against other comparing methods.

4.3 Robustness Analysis

In order to demonstrate the robustness of HALE, we conduct another group of comparative experiments, where the proportion of training examples decreases from 90% to 10%. Note that, it is the first time to evaluate the robustness of PML algorithm w.r.t the number of training examples. Figure 3 illustrates the comparative results

Table 2: Win/tie/loss counts of HALE's performance against comparing methods on synthetic data sets (pairwise t-test at 0.05 significance level).

Data set	Emotions	Birds	Medical	Image	Scene	Bibtext	Eurlex-dc	Eurlex-sm	NUS-WIDE	Sum
Hamming Loss	17/1/3	16/2/3	13/8/0	18/0/3	19/2/0	13/8/0	14/7/0	9/12/0	19/2/0	138/42/9
Ranking Loss	7/0/14	20/1/0	3/0/18	20/1/0	20/1/0	6/0/15	20/1/0	15/3/3	19/2/0	130/9/50
One Error	20/0/1	19/2/0	19/2/0	18/0/2	18/1/2	9/0/12	15/0/6	13/2/6	21/0/0	152/7/30
Coverage	21/0/0	13/2/6	11/2/8	20/1/0	21/0/0	11/1/9	12/2/7	9/0/12	21/0/0	139/8/42
Average Precision	18/1/2	20/1/0	21/0/0	21/0/0	21/0/0	20/1/0	17/3/1	18/0/3	20/1/0	176/7/6
Sum	83/2/20	88/8/9	67/12/26	97/2/6	99/4/2	59/10/36	78/13/14	64/17/24	100/5/0	735/73/137

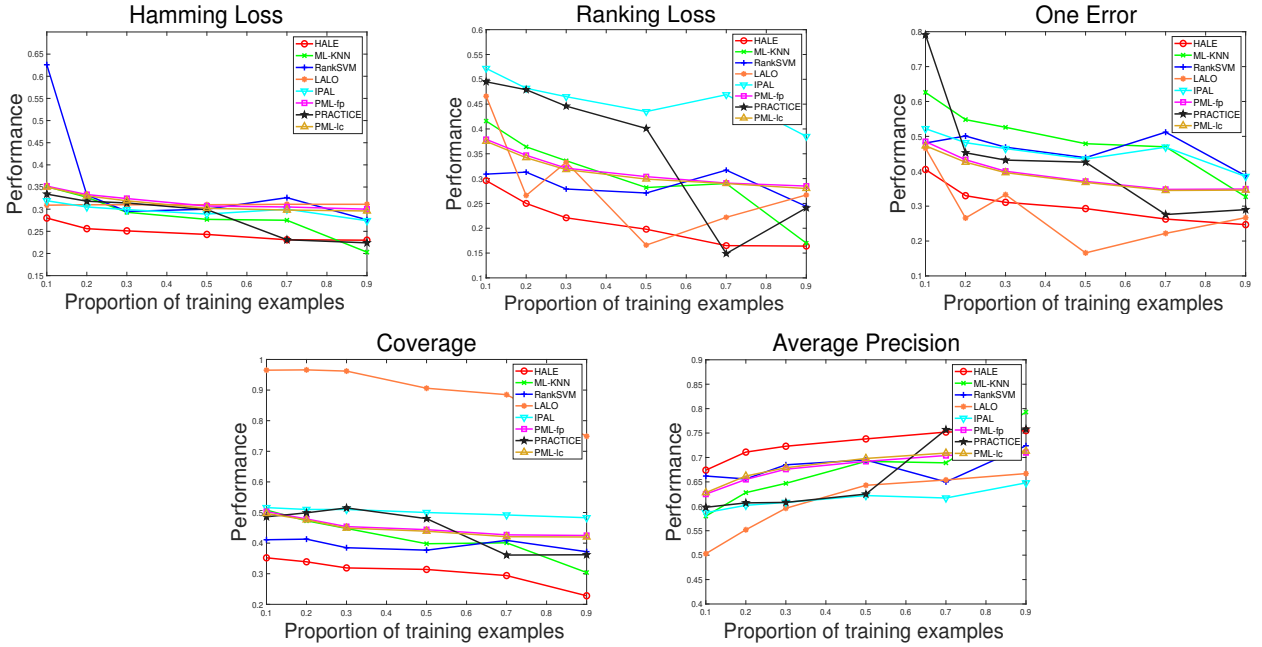

Figure 3: The performance of each comparing method on *Emotions* data set changes as the proportion of training examples increases from 0.1 to 0.9 (with one false candidate label [$r = 1$]).

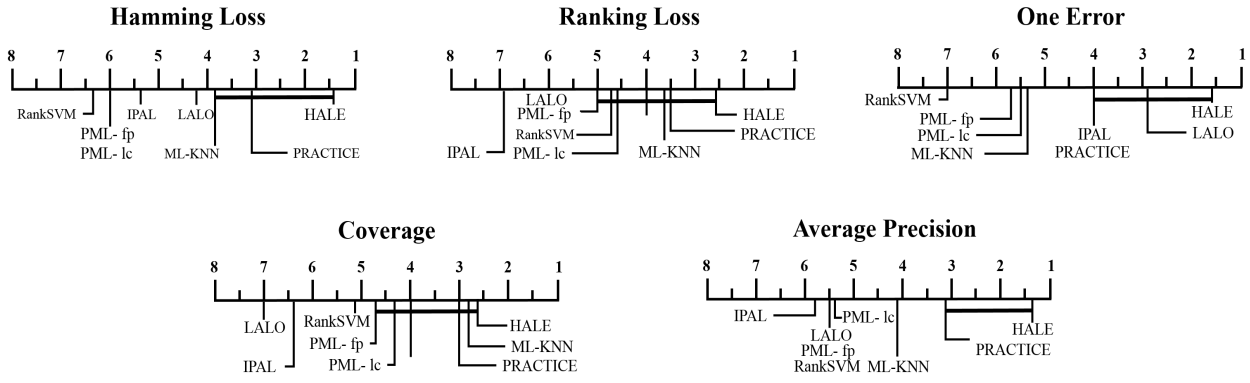

Figure 4: Comparison of HALE (control algorithm) against seven comparing algorithms with the Bonferroni-Dunn test. Algorithms not connected with HALE in the CD diagram are considered to have significantly different performance from the control algorithm (CD = 2.60 at 0.05 significance level)

Table 3: Comparison of HALE with state-of-the-art MLL, PLL, and PML approaches on five evaluation metrics, where the best performances are shown in bold face. (r = 3, pairwise t-test at 0.05 significance level)

	Hamming Loss (the lower the better)								
Data set	Emotions	Birds	Medical	Image	Scene	Bibtext	Eurlex-dc	Eurlex-sm	NUS-WIDE
HALE	**0.297±0.071**	0.096±0.009	**0.017±0.001**	**0.189±0.017**	**0.108±0.008**	0.016±0.001	**0.004±0.003**	**0.008±0.005**	0.031±0.008
ML-KNN	0.607±0.029	**0.053±0.006**	0.022±0.002	0.753±0.005	0.816±0.007	**0.014±0.000**	0.009±0.005	0.012±0.001	0.049±0.016
RankSVM	0.517±0.020	0.194±0.015	0.074±0.003	0.650±0.020	0.646±0.003	0.021±0.000	0.006±0.001	0.009±0.006	-
IPAL	0.314±0.021	0.125±0.009	0.018±0.003	0.199±0.015	0.173±0.001	0.020±0.000	0.010±0.004	0.013±0.005	0.059±0.017
LALO	0.311±0.002	0.098±0.012	0.027±0.001	0.244±0.005	0.107±0.009	0.015±0.000	0.007±0.002	0.009±0.002	-
PML-fp	0.437±0.027	0.157±0.007	0.056±0.005	0.448±0.030	0.362±0.017	0.019±0.000	0.011±0.006	0.016±0.002	-
PML-lc	0.437±0.027	0.132±0.012	0.063±0.003	0.443±0.014	0.363±0.008	0.021±0.001	0.013±0.005	0.019±0.002	-
PARTICLE	0.233±0.018	0.142±0.018	0.024±0.002	0.403±0.042	0.483±0.042	0.017±0.000	0.004±0.000	0.010±0.001	**0.029±0.006**
	Ranking Loss (the lower the better)								
Data set	Emotions	Birds	Medical	Image	Scene	Bibtext	Eurlex-dc	Eurlex-sm	NUS-WIDE
HALE	**0.235±0.037**	**0.271±0.061**	0.169±0.025	**0.192±0.015**	0.096±0.009	0.601±0.009	**0.079±0.002**	**0.040±0.002**	**0.239±0.012**
ML-KNN	0.241±0.026	0.304±0.048	0.088±0.019	0.342±0.026	0.179±0.016	0.232±0.006	0.086±0.012	0.043±0.003	0.301±0.011
RankSVM	0.235±0.037	0.291±0.067	0.103±0.018	0.247±0.023	0.155±0.009	**0.224±0.006**	0.134±0.012	0.085±0.005	-
IPAL	0.738±0.052	0.864±0.041	0.383±0.055	0.383±0.056	**0.085±0.008**	0.703±0.009	0.663±0.011	0.619±0.011	0.935±0.015
LALO	0.240±0.024	0.428±0.037	0.079±0.035	0.196±0.025	0.317±0.029	0.689±0.018	0.326±0.005	0.506±0.006	-
PML-fp	0.462±0.034	0.368±0.050	**0.052±0.016**	0.494±0.043	0.370±0.031	0.336±0.002	0.068±0.014	0.079±0.012	-
PML-lc	0.459±0.035	0.321±0.021	0.056±0.012	0.467±0.025	0.359±0.012	0.342±0.005	0.071±0.013	0.082±0.013	-
PARTICLE	0.259±0.019	0.301±0.032	0.100±0.021	0.315±0.073	0.175±0.066	0.287±0.010	**0.061±0.023**	0.053±0.002	0.240±0.015
	One Error (the lower the better)								
Data set	Emotions	Birds	Medical	Image	Scene	Bibtext	Eurlex-dc	Eurlex-sm	NUS-WIDE
HALE	**0.296±0.027**	**0.467±0.083**	**0.253±0.036**	0.255±0.016	**0.243±0.015**	0.491±0.014	0.292±0.012	**0.165±0.008**	**0.761±0.011**
ML-KNN	0.383±0.062	0.794±0.031	0.425±0.041	0.562±0.035	0.395±0.028	0.624±0.008	0.502±0.015	0.189±0.003	0.866±0.005
RankSVM	0.389±0.096	0.650±0.081	0.588±0.056	0.437±0.030	0.381±0.022	0.518±0.012	0.602±0.021	0.539±0.012	-
IPAL	0.511±0.093	0.760±0.072	0.285±0.059	0.289±0.052	0.262±0.013	**0.405±0.020**	0.244±0.015	0.172±0.012	0.883±0.011
LALO	0.300±0.153	0.836±0.079	0.285±0.096	**0.120±0.140**	0.286±0.028	0.443±0.018	**0.213±0.002**	0.179±0.009	-
PML-fp	0.527±0.049	0.747±0.060	0.295±0.035	0.712±0.053	0.757±0.038	0.465±0.013	0.429±0.008	0.292±0.015	-
PML-lc	0.531±0.045	0.589±0.036	0.325±0.043	0.683±0.027	0.773±0.034	0.468±0.016	0.432±0.009	0.306±0.013	-
PARTICLE	0.306±0.065	0.575±0.084	0.251±0.066	0.410±0.121	0.356±0.121	0.557±0.014	0.356±0.012	0.212±0.017	0.835±0.012
	Coverage (the lower the better)								
Data set	Emotions	Birds	Medical	Image	Scene	Bibtext	Eurlex-dc	Eurlex-sm	NUS-WIDE
HALE	**0.332±0.040**	**0.206±0.049**	0.095±0.016	**0.219±0.017**	**0.096±0.009**	0.478±0.007	0.235±0.009	0.243±0.010	**0.243±0.006**
ML-KNN	0.374±0.028	0.206±0.045	0.111±0.027	0.324±0.020	0.165±0.016	0.369±0.009	**0.098±0.002**	**0.061±0.008**	0.389±0.011
RankSVM	0.368±0.040	0.399±0.064	0.125±0.021	0.253±0.021	0.144±0.008	**0.294±0.008**	0.158±0.021	0.339±0.011	-
IPAL	0.514±0.036	0.584±0.042	0.220±0.031	0.225±0.027	0.900±0.029	0.498±0.023	0.239±0.012	0.689±0.012	0.628±0.026
LALO	0.815±0.064	0.595±0.075	0.084±0.026	0.917±0.042	0.162±0.015	0.499±0.004	0.291±0.013	0.602±0.004	-
PML-fp	0.528±0.035	0.485±0.054	**0.055±0.021**	0.448±0.032	0.321±0.027	0.356±0.016	0.121±0.019	0.166±0.011	-
PML-lc	0.526±0.036	0.453±0.046	0.062±0.026	0.430±0.022	0.312±0.011	0.372±0.016	0.132±0.023	0.176±0.025	-
PARTICLE	0.365±0.040	0.396±0.033	0.121±0.023	0.275±0.088	0.142±0.066	0.458±0.014	0.101±0.021	0.114±0.004	0.295±0.021
	Average Precision (the higher the better)								
Data set	Emotions	Birds	Medical	Image	Scene	Bibtext	Eurlex-dc	Eurlex-sm	NUS-WIDE
HALE	**0.751±0.035**	**0.505±0.054**	**0.769±0.023**	**0.762±0.016**	**0.849±0.010**	**0.353±0.010**	**0.673±0.006**	**0.739±0.013**	**0.230±0.007**
ML-KNN	0.741±0.029	0.453±0.052	0.672±0.039	0.627±0.022	0.744±0.019	0.306±0.006	0.603±0.012	0.761±0.012	0.171±0.015
RankSVM	0.722±0.050	0.443±0.070	0.544±0.044	0.714±0.021	0.760±0.009	0.329±0.002	0.413±0.015	0.530±0.012	-
IPAL	0.588±0.051	0.275±0.042	0.667±0.048	0.662±0.035	0.788±0.014	0.326±0.012	0.603±0.009	0.436±0.009	0.116±0.005
LALO	0.608±0.045	0.223±0.035	0.404±0.036	0.658±0.025	0.793±0.018	0.347±0.016	0.670±0.009	0.490±0.005	-
PML-fp	0.573±0.025	0.350±0.041	0.700±0.012	0.501±0.037	0.491±0.030	0.297±0.008	0.613±0.009	0.605±0.011	-
PML-lc	0.573±0.025	0.388±0.035	0.713±0.012	0.523±0.019	0.489±0.023	0.283±0.010	0.602±0.012	0.582±0.013	-
PARTICLE	0.745±0.024	0.431±0.051	0.720±0.044	0.689±0.096	0.750±0.098	0.313±0.012	0.630±0.016	0.695±0.012	0.206±0.017

'-' means overlong time consumption. Hence, the experimental results are not reported.

Table 4: Comparison of HALE with state-of-the-art MLL, PLL, and PML approaches on real-world data sets, where the best performances are shown in bold face. (pairwise t-test at 0.05 significance level).

Data sets	HALE	ML-KNN	RankSVM	IPAL	LALO	PML-fp	PML-lc	PARTICLE	Evaluation Metrics
Music-emotion	0.219±0.003	0.364±0.011	0.609±0.013	0.239±0.002	0.222±0.000	0.228±0.005	0.221±0.008	**0.206±0.003**	Hamming Loss ↓
Music-style	**0.122±0.005**	0.844±0.010	0.791±0.114	0.178±0.004	0.152±0.002	0.153±0.008	0.162±0.006	0.137±0.007	
Mirflickr	**0.156±0.003**	0.217±0.006	0.307±0.010	0.225±0.013	0.241±0.011	0.178±0.007	0.193±0.006	0.179±0.008	
Music-emotion	0.299±0.008	0.365±0.010	0.523±0.009	0.838±0.009	0.349±0.022	0.324±0.012	0.310±0.010	**0.261±0.007**	Ranking Loss ↓
Music-style	0.271±0.015	0.229±0.010	0.693±0.031	0.752±0.015	0.339±0.020	**0.195±0.012**	0.269±0.016	0.351±0.014	
Mirflickr	**0.143±0.009**	0.178±0.013	0.223±0.016	0.634±0.012	0.151±0.012	0.190±0.008	0.206±0.008	0.213±0.011	
Music-emotion	**0.356±0.009**	0.591±0.015	0.773±0.012	0.605±0.015	0.454±0.074	0.435±0.011	0.421±0.016	0.419±0.019	One Error ↓
Music-style	**0.318±0.016**	0.398±0.020	0.844±0.064	0.671±0.020	0.500±0.141	0.432±0.012	0.493±0.015	0.410±0.027	
Mirflickr	**0.139±0.009**	0.411±0.016	0.518±0.013	0.403±0.013	0.142±0.006	0.172±0.010	0.179±0.006	0.152±0.012	
Music-emotion	0.475±0.006	0.512±0.010	0.694±0.012	0.606±0.007	0.857±0.054	0.436±0.007	0.421±0.009	**0.409±0.009**	Coverage ↓
Music-style	**0.286±0.010**	0.295±0.014	0.713±0.010	0.382±0.010	0.815±0.014	0.398±0.017	0.467±0.011	0.368±0.017	
Mirflickr	**0.211±0.008**	0.262±0.011	0.309±0.007	0.564±0.012	0.786±0.021	0.286±0.009	0.309±0.009	0.271±0.013	
Music-emotion	0.543±0.005	0.505±0.010	0.377±0.005	0.423±0.008	0.411±0.025	0.556±0.016	0.539±0.013	**0.626±0.011**	Average Precision ↑
Music-style	**0.687±0.013**	0.659±0.014	0.269±0.026	0.473±0.011	0.354±0.030	0.653±0.011	0.598±0.009	0.621±0.016	
Mirflickr	**0.718±0.005**	0.698±0.013	0.671±0.012	0.566±0.016	0.671±0.009	0.683±0.007	0.675±0.010	0.690±0.012	

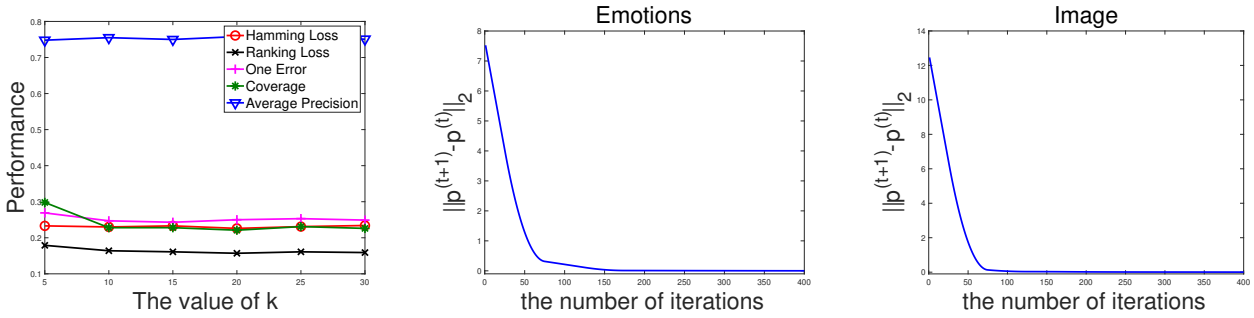

Figure 5: (I) The performance of HALE changes as each parameter increases with other parameters fixed [left]. (II) The convergence curves of HALE on *Emotions* [center] and *Image* [right] data sets with increasing number of iterations.

Table 5: Friedman statics τ_F in terms of each evaluation metric (at 0.05 significance level).

Evaluation Metric	τ_F	Average Precision
Hamming Loss	10.07	2.131 (Methods: 8, Data sets: 12)
Ranking Loss	4.28	
One Error	10.91	
Coverage	9.37	
Average Precision	8.22	

between HALE and other state-of-the-art methods on five evaluation metrics. As described in Figure 3, HALE is superior to all other comparing methods on most evaluation metrics. Especially, when the scale of training examples is extremely small (10%), HALE can significantly outperform all comparing methods on each evaluation metric. In summary, the robustness of HALE is demonstrated.

5 FURTHER ANALYSIS

Complexity Analysis: Theoretically, at each iteration of Algorithm 1, HALE consists of the $O(m^2q^2)$ operations required for the matrix-vector multiplication, and $O(mq)$ operations to row-normalize the assignment matrix, as well as $O(m^2q^2)$ operations needed to weigh the affinity matrix. Therefore, the overall computational complexity of HALE can reach to $O(m^2q^2)$. However, in practice, since we only utilize the k-nearest-neighbor instances to build the instance similarity matrices, and continuously conduct the

Table 6: Total running time (training time/testing time) comparison between HALE and other comparing methods on *Emotions*, *Image* and *NUS-WIDE* data sets.

Running time(s)	Emotions	Image	NUS-WIDE
HALE	0.845/0.050	8.905/0.338	1,495.508/76.529
ML-KNN	0.136/0.114	2.374/0.192	865.015/79.120
RankSVM	72.989/0.376	155.407/1.449	-
IPAL	0.259/0.027	2.446/0.219	2,152.075/200.147
LALO	6.294/1.029	64.098/11.360	-
PML-fp	494.88/0.080	2,753.531/0.100	-
PML-lc	316.35/0.060	2,265.123/0.106	-
PARTICLE	1.273/0.039	9.578/0.502	3,124.148/254.080

'-' means that the time consumption is over one week.

sparsity operation on all employed matrix during the whole learning process, thus the practical computational cost of HALE is far less than $O(m^2q^2)$. Table 6 illustrates the running time comparison between HALE and other comparing methods, measured within Matlab environment equipped with Intel E5-2650 CPU. According to Table 6, our proposed HALE is significantly effective than most comparing methods.

Parameter Analysis: We study the sensitivity analysis of HALE with respect to the crucial parameter k. Sub-figure [left] in Figure 5 illustrates the performance of HALE changes as k increases from 5 to 30 with step-size of 5 on *Emotions* data set. As shown in Figure 5, HALE is robust in terms of the parameter k and we empirically set $k = 10$ in our experiments.

Convergence Analysis: We conduct the convergence analysis of HALE on *Emotions* and *Image* data sets, where the convergence curves are separately shown in the right two sub-figures of Figure 5. We can easily observe that each $\|\mathbf{p}^{(t+1)} - \mathbf{p}^{(t)}\|_2$ gradually decreases to 0 as the number of iterations t increases. Therefore, the convergence of HALE is empirically demonstrated. In addition, in **Section 3.3**, we show that the two-step iterative scheme presented in Eq. (7) and Eq. (8) is monotonically decreasing the objective function in **OP (2)**. Therefore, the convergence of HALE is further demonstrated in theory.

6 CONCLUSION

In this paper, we proposed a novel probabilistic graph matching based partial multi-label learning framework named HALE. To the best of our knowledge, it is the first time to reformulate the PML problem into a graph matching structure. By incorporating the *instance-label* assignment correlations, the proposed HALE algorithm can effectively disambiguate the candidate label set and identify the credible labels for each training instance. Extensive comparative experiments demonstrate that HALE can achieve superior or comparable performance against state-of-the-art methods.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China (No. 61872032, No. U1934220), in part by the Beijing Natural Science Foundation (No. 4202058, No. 9192008), in part by the Fundamental Research Funds for the Central universities (2019JBM020, 2020YJS026, 2020YJS036), and in part by the Key R&D Program of Zhejiang Province (No.2019C01068).

REFERENCES

[1] M. Boutell, J. Luo, X. Shen, and C. Brown. 2004. Learning multi-label scene classification. *Pattern Recognition* 37, 9 (2004), 1757–1771.
[2] S. Burkhardt and S. Kramer. 2018. Online multi-label dependency topic models for text classification. *Machine Learning* 107, 5 (2018), 859–886.
[3] M. Chertok and Y. Keller. 2010. Spectral symmetry analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 7 (2010), 1227–1238.
[4] T. Cour, B. Sapp, and B. Taskar. 2011. Learning from partial labels. *IEEE Transactions on Knowledge and Data Engineering* 12, 5 (2011), 1501–1536.
[5] J. Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* 7, Jan (2006), 1–30.
[6] A. Egozi, Y. Keller, and H Guterman. 2013. A probabilistic approach to spectral graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 1 (2013), 18–27.
[7] André Elisseeff and Jason Weston. 2002. A kernel method for multi-labelled classification. In *Advances in Neural Information Processing Systems*. 681–687.
[8] J. Fang and M. Zhang. 2019. Partial multi-label learning via credible label elicitation. In *AAAI Conference on Artificial Intelligence*. 3518–3525.
[9] L. Feng and B. An. 2018. Leveraging latent label distributions for partial label learning. In *International Joint Conference on Artificial Intelligence*. 2107–2113.
[10] L. Feng and B. An. 2019. Partial label learning with self-guided retraining. In *AAAI Conference on Artificial Intelligence*. 3542–3549.
[11] J. Fürnkranz, E. Hüllermeier, E. Mencía, and K. Brinker. 2008. Multi label classification via calibrated label ranking. *Machine Learning* 73, 2 (2008), 133–153.
[12] E. Gibaja and S. Ventura. 2015. A tutorial on multi label learning. *ACM Computing Surveys (CSUR)* 47, 3 (2015), 52.
[13] C. Gong, T. Liu, Y. Tang, J. Yang, J. Yang, and D. Tao. 2017. A regularization approach for instance-based superset label learning. *IEEE Transactions on Cybernetics* 48, 3 (2017), 967–978.
[14] J. Huang, G. Li, Q. Huang, and X. Wu. 2016. Learning label-specific features and class-dependent labels for multi-label classification. *IEEE Transactions on Knowledge and Data Engineering* (2016), 3309–3323.
[15] R. Hummel and S. Zucker. 1983. On the foundations of relaxation labeling processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 3 (1983), 267–287.
[16] R. Jin and Z. Ghahramani. 2003. Learning with multiple labels. In *Advances in Neural Information Processing Systems*. 921–928.
[17] L. Jing, L. Yang, J. Yu, and M. Ng. 2015. Semi-supervised low-rank mapping learning for multi-label classification. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1483–1491.
[18] M. Leardeanu and M. Hebert. 2005. A spectral technique for correspondence problems using pairwise constraints. In *International Conference on Computer Vision*. 1482–1489.
[19] Y. Li, Y. Song, and J. Luo. 2017. Improving pairwise ranking for multi-label image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*. 3617–3625.
[20] L. Liu and T. Dietterich. 2012. A conditional multinomial mixture model for superset label learning. In *Advances in Neural Information Processing Systems*. 548–556.
[21] W. Liu and S. Chang. 2009. Robust multi-class transductive learning with graphs. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 381–388.
[22] Z. Liu and H. Qiao. 2013. Gnccp graduated non convexity and concavity procedure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 6 (2013), 1258–1267.
[23] G. Lyu, S. Feng, T. Wang, and C. Lang. 2020. A self-paced regularization framework for partial label learning. *IEEE Transactions on Cybernetics* (2020), 1–13. https://doi.org/10.1109/TCYB.2020.2990908
[24] G. Lyu, S. Feng, T. Wang, C. Lang, and Y Li. 2019. GM-PLL: graph matching based partial label learning. *IEEE Transactions on Knowledge and Data Engineering* (2019), 1–14. https://doi.org/10.1109/TKDE.2019.2933837
[25] L. Sun, S. Feng, T. Wang, C. Lang, and Y. Jin. 2019. Partial multi-label learning via low-rank and sparse decomposition. In *AAAI Conference on Artificial Intelligence*. 5016–5023.
[26] D. Thibaut, M. Nazanin, and M. Greg. 2019. Learning a deep convNet for multi-label classification with partial labels. In *IEEE Conference on Computer Vision and Pattern Recognition*. in press.
[27] H. Wang, W. Liu, Y. Zhao, C. Zhang, T. Hu, and G. Chen. 2019. Discriminative and correlative partial multi label learning. In *International Joint Conference on Artificial Intelligence*. 3691–3697.
[28] B. Wu, F. Jia, W. Liu, B. Ghanem, and S. Lyu. 2018. Multi-label learning with missing labels using mixed dependency graphs. *International Journal of Computer Vision* 126, 8 (2018), 875–896.
[29] X. Wu and M. Zhang. 2018. Towards enabling binary decomposition for partial label learning.. In *International Joint Conference on Artificial Intelligence*. 2868–2874.
[30] M. Xie and S. Huang. 2018. Partial multi-label learning. In *AAAI Conference on Artificial Intelligence*. 4302–4309.
[31] M. Xie and S. Huang. 2020. Partial multi-label learning with noisy label identification. In *AAAI Conference on Artificial Intelligence*. 1–8.
[32] C. Xu, D. Tao, and C. Xu. 2016. Robust extreme multi-label learning. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1275–1284.
[33] M. Zhang, Y. Li, X. Liu, and X. Geng. 2018. Binary relevance for multi-label learning: an overview. *Frontiers of Computer Science* 12, 2 (2018), 191–202.
[34] M. Zhang and F. Yu. 2015. Solving the partial label learning problem: an instance-based approach. In *International Joint Conference on Artificial Intelligence*. 4048–4054.
[35] M. Zhang, F. Yu, and C. Tang. 2017. Disambiguation-free partial label learning. *IEEE Transactions on Knowledge and Data Engineering* 29, 10 (2017), 2155–2167.
[36] M. Zhang and Z. Zhou. 2007. ML-KNN: a lazy learning approach to multi-label learning. *Pattern Recognition* 40, 7 (2007), 2038–2048.
[37] M. Zhang and Z. Zhou. 2013. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering* 26, 8 (2013), 1819–1837.