Multiple Noisy Label Distribution Propagation for Crowdsourcing

Hao Zhang, **Liangxiao Jiang*** and **Wenqiang Xu**
Department of Computer Science, China University of Geosciences, Wuhan 430074, China
ljiang@cug.edu.cn

Abstract

Crowdsourcing services provide a fast, efficient, and cost-effective means of obtaining large labeled data for supervised learning. Ground truth inference, also called label integration, designs proper aggregation strategies to infer the unknown true label of each instance from the multiple noisy label set provided by ordinary crowd workers. However, to the best of our knowledge, nearly all existing label integration methods focus solely on the multiple noisy label set itself of the individual instance while totally ignoring the intercorrelation among multiple noisy label sets of different instances. To solve this problem, a multiple noisy label distribution propagation (MNLDP) method is proposed in this study. MNLDP first transforms the multiple noisy label set of each instance into its multiple noisy label distribution and then propagates its multiple noisy label distribution to its nearest neighbors. Consequently, each instance absorbs a fraction of the multiple noisy label distributions from its nearest neighbors and yet simultaneously maintains a fraction of its own original multiple noisy label distribution. Promising experimental results on simulated and real-world datasets validate the effectiveness of our proposed method.

1 Introduction

In many pervasive applications of supervised learning, particularly with the rapid growth of deep learning, one frequently encounters the situation in which extensive data remain unlabeled. Traditionally, labeling tasks have been typically processed by domain experts. This provides accurate labels but is inefficient and involves a high cost [Tian and Zhu, 2015; Zhang *et al.*, 2016b]. Thus, crowdsourcing has emerged as an effective paradigm for annotating large datasets in domains such as computer vision and natural language processing [Liu *et al.*, 2012; Karger *et al.*, 2014]. Crowdsourcing platforms such as Amazon Mechanical Turk[1] and Crowdflower[2] provide participative online markets where the requesters publish specific types of human intelligence tasks (HITs) and collect numerous labels from ordinary labelers (or annotators) in a short time and at a relatively low cost. However, the qualities of collected labels provided by a single crowd labeler are often poor and the noise associated with the labels can compromise practical applications. To our knowledge, these unreliable labels may be caused by personal preference, low payment for each task, and varying cognitive abilities [Sheng *et al.*, 2008; Rodrigues and Pereira, 2017; Qiu *et al.*, 2018]. To solve this problem, multiple labels are frequently requested from different crowd labelers for a single instance. In other words, repeated labeling is performed [Li *et al.*, 2016; Zhang and Wu, 2018; Li *et al.*, 2019]. After acquiring multiple noisy label sets of each instance by repeated labeling, label integration methods can be used to infer (estimate) the unknown true label of each instance.

Consequently, integrating labels from multiple noisy labels has recently attracted considerable research attention [Sheshadri and Lease, 2013; Zhang *et al.*, 2016a]. The most straightforward label integration (consensus) method is majority voting (MV), which naively assumes that all labelers have the same reliability. In addition to MV, more sophisticated methods have been proposed to improve the performance of label integration [Dawid and Skene, 1979; Whitehill *et al.*, 2009; Raykar *et al.*, 2010; Demartini *et al.*, 2012; Karger *et al.*, 2014; Zhang *et al.*, 2016a; Rodrigues and Pereira, 2017]. These improved methods have clearly achieved remarkable progress to model crowdsourcing based on different parameters such as the reliabilities of labelers, the difficulties of instances, and labeling biases. However, to the best of our knowledge, nearly all existing label integration methods focus solely on the multiple noisy label set itself of the individual instance while totally ignoring the intercorrelation among multiple noisy label sets of different instances.

To solve this problem, in this study we propose a multiple noisy label distribution propagation (MNLDP) method. MNLDP focuses on the intercorrelation among multiple noisy label sets of different instances instead of directly estimating the aforementioned parameters. In MNLDP, the traditional multiple noisy labels are first transformed into multiple noisy label distributions, and a feature space based on the overlapped local linear neighborhood patches is constructed, where the edge weights of each patch can be calculated by a standard quadratic programming procedure. Then, the topo-

*Corresponding author
[1]http://www.mturk.com
[2]http://crowdflower.com

logical structure of the feature space is globally shared with the multiple noisy label space for simplicity. With the multiple noisy label space available, MNLDP then propagates the multiple noisy label distribution of each instance to its nearest neighbors. Consequently, each instance absorbs a fraction of the multiple noisy label distributions from its nearest neighbors and yet simultaneously maintains a fraction of its own multiple noisy label distribution.

The main contributions of this work are briefly summarized as follows: 1) The multiple noisy label integration problem is transformed into an MNLDP problem, which provides a new perspective from which to solve the multiple noisy label integration problem. 2) A novel MNLDP algorithm is designed to exploit the intercorrelation among multiple noisy label sets of different instances, and thus the effect of the multiple noisy label set itself of the individual instance is weakened.

The remainder of this paper is organized as follows. Section 2 introduces related work. Section 3 describes the proposed MNLDP method in detail. Section 4 presents a convergence analysis of MNLDP. Section 5 describes the experimental setup and results. Section 6 concludes the study and outlines the main directions for future work.

2 Related Work

Because of the openness of crowdsourcing, inferring the unknown true labels of instances from multiple noisy labels is a challenge. Many label integration (consensus) methods have been proposed for crowdsourcing. Majority voting (MV) [Sheng *et al.*, 2008] is the simplest and most effective method. However, it naively assumes that all labelers have the same reliability, which weakens its performance in many real-world crowdsourcing scenarios.

To improve the quality of integration labels, researchers have proposed numerous label integration (consensus) methods. [Dawid and Skene, 1979] proposed a method (DS) based on the expectation maximization algorithm, which uses the maximum likelihood estimation to estimate a confusion matrix for each labeler and a class prior. [Demartini *et al.*, 2012] proposed the ZenCrowd method (ZC), which uses only a two-element parameter to weight the reliability of a labeler. [Karger *et al.*, 2014] proposed a belief propagation-like label integration method (KOS) based on the reliabilities of labelers. [Zhang *et al.*, 2016a] proposed another novel method called the ground truth inference using clustering (GTIC) that is based on Bayesian statistics for multi-class labeling. Focusing on binary labeling, [Raykar *et al.*, 2010] proposed a Bayesian estimation-based method to model the two parameters of *sensitivity* and *specificity* to denote labelers' biases toward positive and negative instances, respectively. [Whitehill *et al.*, 2009] proposed a method to infer the true labels, the expertise of workers, and the difficulty of items simultaneously. [Rodrigues and Pereira, 2017] proposed a deep neural network layer known as a crowd layer. [Guan *et al.*, 2017] proposed an approach for training deep neural networks that exploits information about the annotators.

All of these improved methods have clearly achieved remarkable progress to model crowdsourcing based on different aspects such as the reliabilities of labelers, the difficulties of instances, and labeling biases. However, to the best of our knowledge, nearly all existing label integration methods focus solely on the multiple noisy label set itself of the individual instance while totally ignoring the intercorrelation among multiple noisy label sets of different instances. To solve this problem, we propose an MNLDP method. In MNLDP, each instance absorbs a fraction of the multiple noisy label distributions from its nearest neighbors and yet simultaneously maintains a fraction of its own multiple noisy label distribution.

Please note that our proposed MNLDP is entirely different from the well-known label distribution learning (LDL) [Geng, 2016], which is a new machine learning paradigm in which each instance is annotated by a label distribution. The label distribution in LDL covers a certain number of labels, representing the description degree of each label to the data point. One label may only partially describe the instance, but that the label describes the instance is completely true. However, in a multiple noisy label distribution, only one label is regarded as completely true, and the other labels are all wrong. Please also note that our proposed MNLDP is entirely different from the linear neighborhood propagation (LNP) [Wang and Zhang, 2008] for semi-supervised learning. LNP propagates the labels from the labeled points to the whole dataset using these linear neighborhoods with sufficient smoothness. By contrast, MNLDP propagates the multiple noisy label distributions rather than the multiple noisy labels themselves.

3 MNLDP Method

In crowdsourcing scenarios, a dataset can be expressed as $S = \{(\boldsymbol{x}_i, \mathcal{L}_i)|1 \leq i \leq n\}$, where $\boldsymbol{x}_i$ denotes the i-th $(i = 1, 2, \cdots, n)$ instance, and $\mathcal{L}_i = \{l_{ir}\}_{r=1}^{R}$ denotes its multiple noisy label set provided by R labelers. Each element l_{ir} represents the label provided by the r-th $(r = 1, 2, \cdots, R)$ labeler, which takes the value from the class label set $\mathcal{Y} = \{y_1, y_2, \cdots, y_c\}$. To construct the multiple noisy label space, $\mathcal{L}_i$ is first transformed into a multiple noisy label distribution $\mathcal{P}_i = \{p_i^{y_1}, p_i^{y_2}, \cdots, p_i^{y_c}\}$, where $p_i^{y_m}$ $(m = 1, 2, \cdots, c)$ denotes the probability (frequency) that all labelers label $\boldsymbol{x}_i$ as y_m. Obviously, $p_i^{y_m} \in [0, 1]$ and $\sum_{m=1}^{c} p_i^{y_m} = 1$. Based on the defined distribution, the dataset S can then be transformed into $S = \{(\boldsymbol{x}_i, \mathcal{P}_i)|1 \leq i \leq n\}$.

In addition, to construct the multiple noisy label space, we still must construct a feature space with the neighbor information of each data point. As with many common graph-based learning methods, the topological structure of the feature space can be represented by $\mathcal{G} =< \mathcal{V}, \mathcal{E}, \boldsymbol{W} >$, where $\mathcal{V}$ is the vertex set corresponding to data points, $\mathcal{E}$ is the edge set associated with each edge e_{ij} representing the relationship between each pair of data points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, and $\boldsymbol{W}$ is the weight matrix of $\mathcal{G}$ with each element w_{ij} measuring how similar $\boldsymbol{x}_i$ is to $\boldsymbol{x}_j$. Note that usually $w_{ij} \neq w_{ji}$.

For computational simplicity, we assume that each data point can be optimally reconstructed using a linear combination of its neighbors in the feature space [Roweis and Saul, 2000; Wang and Zhang, 2008]. Thus, our objective is to min-

imize the following:

$$\varepsilon(\boldsymbol{W}) = \sum_{i=1}^{n} \|\boldsymbol{x}_i - \sum_{j:\boldsymbol{x}_j \in \mathcal{N}(\boldsymbol{x}_i)} w_{ij}\boldsymbol{x}_j\|^2 \quad (1)$$

where $\mathcal{N}(\boldsymbol{x}_i)$ is the set of the k-nearest neighbors of $\boldsymbol{x}_i$. We further constrain $\mathbf{1}^{\mathrm{T}}\boldsymbol{w}_i = 1$, where

$$\boldsymbol{w}_i = \begin{cases} w_{ij}, & \text{if } \boldsymbol{x}_j \in \mathcal{N}(\boldsymbol{x}_i) \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$\sum_j w_{ij} = 1$ and $w_{ij} \geq 0$. For the local weight matrix $\boldsymbol{w}_i$, the minimization can be defined as:

$$\varepsilon(\boldsymbol{w}_i) = \sum_{j,k:\boldsymbol{x}_j,\boldsymbol{x}_k \in \mathcal{N}(\boldsymbol{x}_i)} w_{ij} G^i_{jk} w_{ik} \quad (3)$$

where $G^i_{jk} = (\boldsymbol{x}_i - \boldsymbol{x}_j)^T(\boldsymbol{x}_i - \boldsymbol{x}_k)$ is the (j,k)-th entry of the local Gram matrix $\boldsymbol{G}^i$.

Thus, the construction can be solved by the following n standard least square programming problems, which is formulated as Equation 4. Note that we first obtain a series of weight matrix $\boldsymbol{w}_i$ and then merge them to construct the whole space.

$$\begin{aligned} min_{\boldsymbol{w}_i} \quad & \boldsymbol{w}_i^T \boldsymbol{G}^i \boldsymbol{w}_i \\ s.t. \quad & \mathbf{1}^T \boldsymbol{w}_i = 1 \\ & \forall w_{ij} \in \boldsymbol{w}_i, w_{ij} \geq 0 \end{aligned} \quad (4)$$

Inspired by the smoothness assumption [Zhu *et al.*, 2005] and multi-label manifold learning [Hou *et al.*, 2016], we can naturally infer that the multiple noisy label space locally shares the topological structure of the feature space. However, learning the topological structure from the feature space to the multiple noisy label space is difficult. Thus, we further assume that the global topological structure can be simply shared. With the $\boldsymbol{W}$ and the aforementioned assumption, the multiple noisy label space is approximately formulated as:

$$\boldsymbol{\mu}_i \propto \sum_{j:\boldsymbol{x}_j \in \mathcal{N}(\boldsymbol{x}_i)} w_{ij}\boldsymbol{\mu}_j \quad (5)$$

where $\boldsymbol{\mu}_i = \{\mu_i^{y_1}, \mu_i^{y_2}, ..., \mu_i^{y_c}\}$ is the multiple noisy label distribution of $\boldsymbol{x}_i$ in the propagation. $\boldsymbol{\mu}_i$ is different from $\mathcal{P}_i$. $\mathcal{P}_i$ is the initial multiple noisy label distribution. However, $\boldsymbol{\mu}_i$ is the multiple noisy label distribution at a particular propagation state. Although this hypothesis is naive, we believe it is reasonable in most cases because the annotators provide the multiple noisy labels based on the features of the instance.

In the t-th propagation, the propagated multiple noisy label distribution can be divided into two sections: 1) a fraction of the multiple noisy label distributions from its nearest neighbors, and 2) some multiple noisy label information of its initial state (i.e., $\mathcal{P}_i$). Thus, the $\boldsymbol{\mu}_i$ at the t+1-th propagation is:

$$\boldsymbol{\mu}_i^{t+1} = \alpha_i \sum_{j:\boldsymbol{x}_j \in \mathcal{N}(\boldsymbol{x}_i)} w_{ij}\boldsymbol{\mu}_j^t + (1-\alpha_i)\mathcal{P}_i \quad (6)$$

where $\alpha_i \in (0,1)$ is the controlling factor that adjusts the proportions from its nearest neighbors and its own original state (distribution). More specifically, when $\alpha_i < 0.5$, the propagated multiple noisy label distribution relies more on its own original state. The extreme case is $\alpha_i = 0$, and thus our

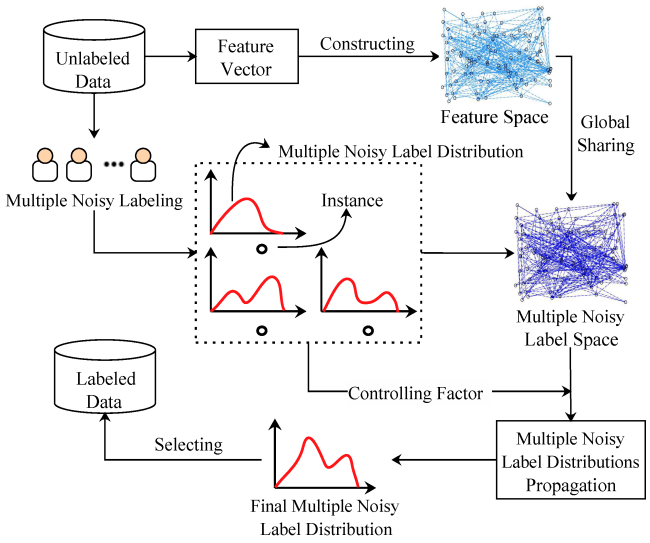

Figure 1: Framework of MNLDP.

MNLDP degenerates into the simplest MV (i.e., $\boldsymbol{\mu}_i = \mathcal{P}_i$). Conversely, when $\alpha_i > 0.5$, the propagated multiple noisy label distribution relies more on its nearest neighbors. When $\alpha_i = 1$, MNLDP completely abandons its own original state $\mathcal{P}_i$ (i.e., $\boldsymbol{\mu}_i^{t+1} = \sum_{j:\boldsymbol{x}_j \in \mathcal{N}(\boldsymbol{x}_i)} w_{ij}\boldsymbol{\mu}_j^t$).

We estimate the controlling factor α_i using the label uncertainty [Sheng *et al.*, 2008]. The detailed equation is:

$$\alpha_i = \frac{\sum_{m=1}^{c} \alpha_i^{y_m}}{c} + \eta \quad (7)$$

where $\eta \in [0, 0.5]$ is a hyper-parameter used to adjust the influence from its nearest neighbors, and

$$\alpha_i^{y_m} = min\{I_d(L_p^{y_m}+1, L_n^{y_m}+1), 1-I_d(L_p^{y_m}+1, L_n^{y_m}+1)\} \quad (8)$$

where

$$I_d(L_p^{y_m}+1, L_n^{y_m}+1) = \sum_{j=L_p^{y_m}+1}^{L_p^{y_m}+L_n^{y_m}+1} \frac{(L_p^{y_m}+L_n^{y_m}+1)!}{j!(L_p^{y_m}+L_n^{y_m}+1-j)!} d^j(1-d)^{L_p^{y_m}+L_n^{y_m}+1-j} \quad (9)$$

where $L_p^{y_m}$ is the number of the class label y_m, $L_n^{y_m}$ is the number of other classes, and the decision threshold $d = 0.5$.

After a certain number of iterations, Equation 6 reaches to a state of convergence. The convergence analysis of Equation 6 is presented in Section 4. Let $\boldsymbol{\mu}_i = \{\mu_i^{y_m}\}_{m=1}^c$ be the final multiple noisy label distribution of $\boldsymbol{x}_i$. Its integration label is then defined as:

$$c(\boldsymbol{x}_i) = \arg\max_{y_m \in \mathcal{Y}} \mu_i^{y_m} \quad (10)$$

Figure 1 graphically shows the basic framework of our proposed MNLDP.

Now, let us take an artificial binary toy classification dataset as an example to illustrate the process and effectiveness of our proposed MNLDP. The dataset is artificially generated from two interleaving half circles, and the standard deviation of the Gaussian noise is 0.1. It contains 200 data points with two-dimensional features. We then employ 20 simulated labelers who possess the same level of reliability to label each point. In our experiments, we set the number

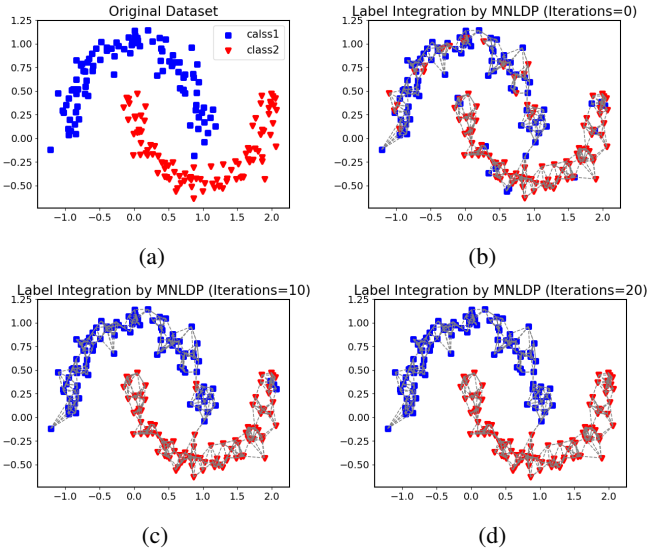

Figure 2: Label integration process and results on the toy dataset.

of nearest neighbors k to 5, and then we observe the label integration process and results when the numbers of iterations are 0, 10, and 20.

Figure 2 graphically shows the detailed process and results, where Figure 2 (a) shows the original data with the true class label distributions. The blue squares and red triangles represent two classes. When the numbers of iterations are 0, 10, and 20, the label integration process and results are shown in Figure 2 (b), Figure 2 (c), and Figure 2 (d), respectively. The grey dashed lines among the different data points connect their five nearest neighbors. From these, the multiple noisy label space is constructed and then the multiple noisy label distributions are propagated through the multiple noisy label space.

4 Convergence Analysis of MNLDP

In this section, we analyze the convergence property of MNLDP theoretically. Let the entire propagative multiple noisy label distribution sequence $\boldsymbol{U}^t = [(\boldsymbol{\mu}_1^t)^T, (\boldsymbol{\mu}_2^t)^T, \cdots, (\boldsymbol{\mu}_n^t)^T]^T$, the entire initial multiple noisy label distribution sequence $\boldsymbol{\mathcal{P}} = [\mathcal{P}_1^T, \mathcal{P}_2^T, \cdots, \mathcal{P}_n^T]^T$, and the diagonal matrix be expressed as:

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 & \cdots & 0 \\ \vdots & \alpha_i & \vdots \\ 0 & \cdots & \alpha_n \end{bmatrix} \tag{11}$$

where α_i is the controlling factor for $\boldsymbol{x}_i$. Equation 6 can then be rewritten as:

$$\boldsymbol{U}^{t+1} = \boldsymbol{\alpha}\boldsymbol{W}\boldsymbol{U}^t + (\boldsymbol{1} - \boldsymbol{\alpha})\boldsymbol{\mathcal{P}} \tag{12}$$

Because the initial condition is $\boldsymbol{U}^0 = \boldsymbol{\mathcal{P}}$, Equation 12 becomes:

$$\boldsymbol{U}^t = (\boldsymbol{\alpha}\boldsymbol{W})^t\boldsymbol{\mathcal{P}} + \sum_{i=0}^{t-1}(\boldsymbol{\alpha}\boldsymbol{W})^i(\boldsymbol{1} - \boldsymbol{\alpha})\boldsymbol{\mathcal{P}} \tag{13}$$

Because of $\alpha_i w_{ij} \in \boldsymbol{\alpha}\boldsymbol{W} \geq 0$ and $\sum_i \alpha_i w_{ij} = \alpha_i$, from the *Perron-Frobenius* theorem, the spectral radius of $\boldsymbol{\alpha}\boldsymbol{W}$ ranges from $r_{min}(\boldsymbol{\alpha}\boldsymbol{W})$ to $r_{max}(\boldsymbol{\alpha}\boldsymbol{W})$ (i.e., $r_{min}(\boldsymbol{\alpha}\boldsymbol{W}) \leq \rho(\boldsymbol{\alpha}\boldsymbol{W}) \leq r_{max}(\boldsymbol{\alpha}\boldsymbol{W})$), where $r_{min}(\boldsymbol{\alpha}\boldsymbol{W}) = \min_i \sum_j \alpha_i w_{ij}$ and $r_{max}(\boldsymbol{\alpha}\boldsymbol{W}) = \max_i \sum_j \alpha_i w_{ij}$. The eigendecomposition of $(\boldsymbol{\alpha}\boldsymbol{W})^t$ can be represented as:

$$(\boldsymbol{\alpha}\boldsymbol{W})^t = \boldsymbol{V}diag(\boldsymbol{\lambda})^t\boldsymbol{V}^{-1} \tag{14}$$

where $diag(\boldsymbol{\lambda})$ is the diagonal matrix composed of all eigenvalues and $\boldsymbol{V}$ is connected by all eigenvectors. With the process of propagation, the eigenvalues, which are less than 1, are reduced to 0. Thus,

$$\lim_{t\to\infty}(\boldsymbol{\alpha}\boldsymbol{W})^t = \lim_{t\to\infty}\boldsymbol{V}diag(\boldsymbol{\lambda})^t\boldsymbol{V}^{-1} = 0 \tag{15}$$

where $\sum_{i=0}^{t-1}(\boldsymbol{\alpha}\boldsymbol{W})^i$ can be regarded as the infinite series. It is rewritten as:

$$\sum_{i=0}^{t-1}(\boldsymbol{\alpha}\boldsymbol{W})^i = \frac{\boldsymbol{1} - (\boldsymbol{\alpha}\boldsymbol{W})^t}{\boldsymbol{1} - \boldsymbol{\alpha}\boldsymbol{W}} \tag{16}$$

Therefore,

$$\lim_{t\to\infty}\sum_{i=0}^{t-1}(\boldsymbol{\alpha}\boldsymbol{W})^i = (\boldsymbol{1} - \boldsymbol{\alpha}\boldsymbol{W})^{-1} \tag{17}$$

Based on this analysis, $\boldsymbol{U}^t$ will converge to:

$$\lim_{t\to\infty}\boldsymbol{U}^t = (\boldsymbol{1} - \boldsymbol{\alpha}\boldsymbol{W})^{-1}(\boldsymbol{1} - \boldsymbol{\alpha})\boldsymbol{\mathcal{P}} \tag{18}$$

5 Experiments and Results

We validated the effectiveness of our proposed MNLDP by comparing it with five baseline methods, namely, MV, DS, ZC, KOS, and GTIC on simulated and real-world datasets. We implemented our proposed MNLDP on the crowd environment and its knowledge analysis (CEKA) platform [Zhang *et al.*, 2016b] and used the existing implementations of MV, DS, ZC, KOS, and GTIC on the CEKA platform [Zhang *et al.*, 2016b]. In MNLDP, we set the number of nearest neighbors k to 5 and the hyper-parameter η to 0.5.

5.1 Experiments on Simulated Datasets

To validate the effectiveness of our proposed MNLDP in a slightly more controlled environment, six popular benchmark datasets were used by simulating multiple labelers with different levels of expertise. Table 1 provides detailed information on these datasets, which came from the University of California at Irvine (UCI) repository and represent a wide range of domains and data characteristics.

To simulate a crowdsourcing process to obtain multiple noisy labels of each instance, the original true labels of all instances were hidden. For each labeler, the original true label was assigned to each instance with the probability p and any one of false labels otherwise. To maintain the robustness of the experiments under different situations, two simulating strategies were considered:

1. In the first series of experiments, the labeling quality of each labeler was fixed at 0.6 and the number of labelers varied from 3 to 50.

Dataset	Features	Instances	Classes
Glass	9	214	7
Ionosphere	35	351	2
Iris	4	150	3
Image Segmentation	19	2310	7
Vehicle	18	846	4
Vote	17	435	2

Table 1: Six UCI datasets used in the experiments.

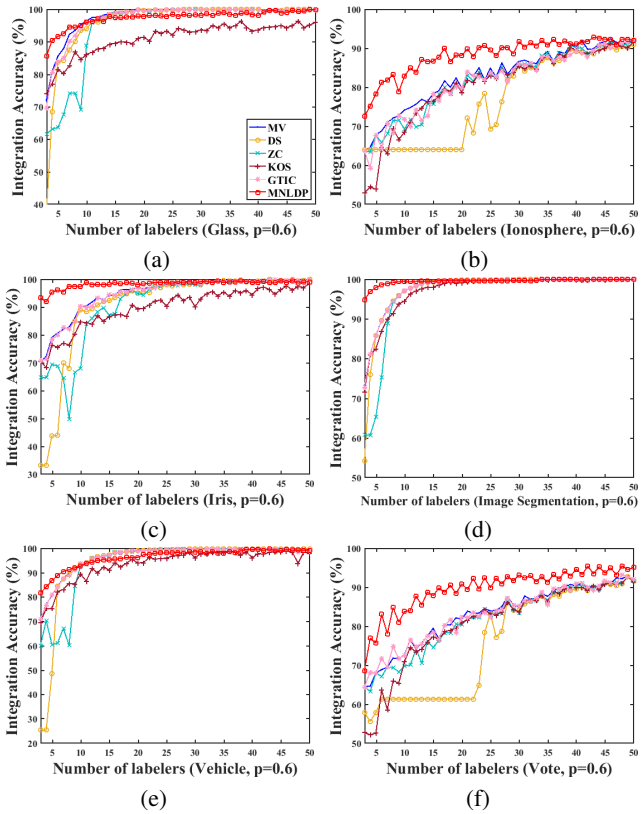

Figure 3: Integration accuracy comparisons in the first series of experiments, where the labeling quality of each labeler was fixed at 0.6.

2. In the second series of experiments, the number of labelers was fixed at 10 and the labeling quality of each labeler was randomly generated from a uniform distribution on the interval $[0.1, 0.95]$.

After obtaining the multiple noisy label set of each instance, we applied label integration (consensus) methods to infer its integration label. We know that each simulation process has a certain degree of randomness that cannot be avoided. To reduce the fluctuations caused by this randomness, each experiment was repeated 20 times independently and the integration accuracies were averaged.

Figure 3 displays the detailed comparison results for the first series of experiments. From these results, we can see that: 1) The integration accuracy of MNLDP was much higher than the five baseline methods, particularly when the number of labelers was less than 10. 2) The differences among the different label integration (consensus) methods gradually

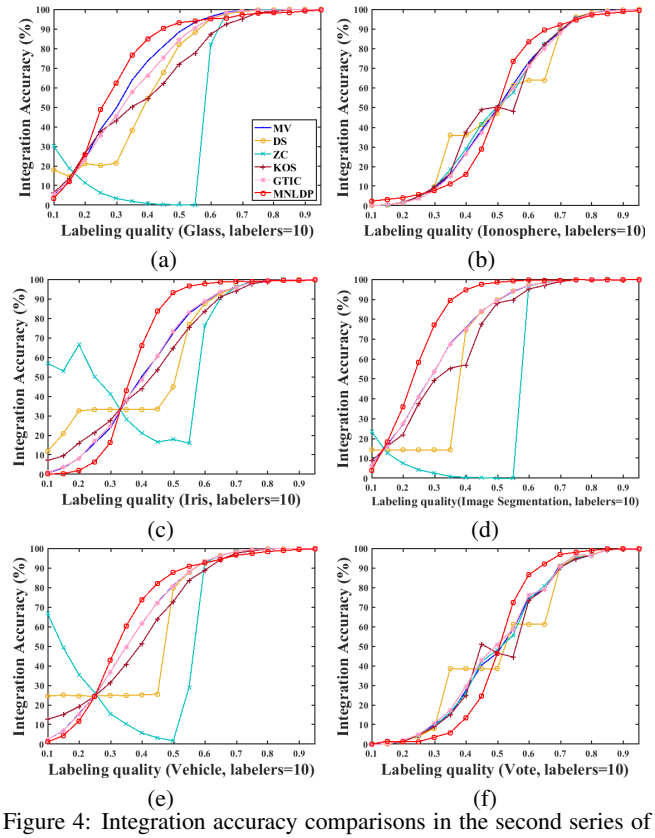

Figure 4: Integration accuracy comparisons in the second series of experiments, where the number of labelers was fixed at 10.

disappeared as the number of labelers increased, particularly when the number of labelers approached 50.

Figure 4 displays the detailed comparison results for the second series of experiments. From these comparison results, we can see that: 1) As expected, the integration accuracy improved as the labeling quality of each labeler increased. 2) Compared to the five baseline methods, an apparent turning point was reached, which mainly concentrated at around 0.3 in the different datasets. When the labeling quality of each labeler was below this turning point, our proposed MNLDP was even worse than a few of the baseline methods. However, when the labeling quality of each labeler surpassed this turning point, our proposed MNLDP significantly outperformed all the baseline methods until the labeling quality approached 0.7. 3) The differences among different label integration (consensus) methods gradually disappeared when the labeling quality reached 0.8.

Based on these comparison results, we can see that the performance of MNLDP was much better overall, that is, more stable and robust than all the other baseline methods. This is because, in MNLDP, each instance absorbed a fraction of the multiple noisy label distributions from its nearest neighbors and yet simultaneously maintained a fraction of its own original multiple noisy label distribution. Therefore, when the labeling quality of each labeler was very low (for example, less than 0.3), MNLDP was even worse than some existing baseline methods because it enlarged the incorrect label distributions in the process of propagation.

Information	Leaves	LabelMe	Music Genre
Classes	6	8	10
Features	64	512	31
Instances	384	1000	700
Labelers	83	59	44
Labels	3840	2547	2946
Mean Labeling Quality	0.638	0.692	0.7328

Table 2: Three real-world crowdsourced datasets used in the experiments.

Method	Leaves	LabelMe	Music Genre
MV	63.8	76.2	70.57
DS	63.54	74.7	52.57
ZC	64.58	77.2	78.29
KOS	64.5	8.9	22.71
GTIC	62.24	76.7	70.71
MNLDP	**65.1**	**82.3**	**79**

Table 3: Integration accuracy (%) comparisons on three real-world crowdsourced datasets.

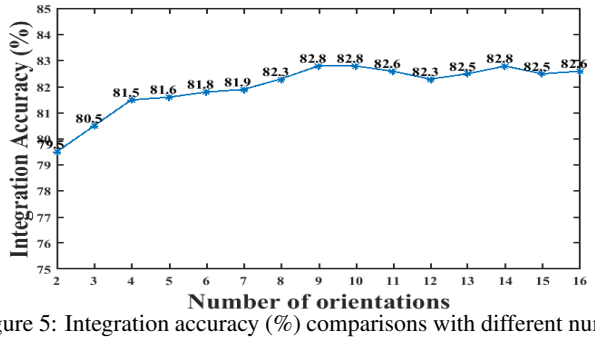

Figure 5: Integration accuracy (%) comparisons with different numbers of orientations per scale.

5.2 Experiments on Real-World Datasets

To evaluate further the performance of MNLDP, we conducted our experiments on three real-world crowdsourced datasets: Leaves, LableMe, and Music Genre, which were collected from Amazon Mechanical Turk (AMT) and are publicly available.

The crowdsourced dataset Leaves [Zhang *et al.*, 2016b] is a traditional classification dataset, which distinguishes six types of leaves depending on their shape and other characteristics. The LabelMe dataset [Rodrigues and Pereira, 2017] is an image classification domain derived from LabelMe data [Russell *et al.*, 2008]. To find neighbor images, MNLDP computes the image similarities based on 512 numerical features determined by the GIST descriptor [3], which was proposed by [Oliva and Torralba, 2001]. The Music Genre dataset [4] is published on AMT and used to collect multiple noisy labels by [Rodrigues *et al.*, 2013]. Because different genres have different extraction criteria and the criteria are non-overlapping [Aucouturier and Pachet, 2003], we utilized Principal Component Analysis to extract 31 pivotal features from the 124-dimensional full combined feature vector. Table 2 summarizes the detailed data characteristics of these crowdsourced datasets.

Table 3 presents detailed comparison results obtained by different label integration (consensus) methods on three real-world crowdsourced datasets. As expected, our proposed MNLDP significantly outperformed all the baseline methods. The integration accuracies of MNLDP on three real-world crowdsourced datasets were 65.1%, 82.3%, and 79%, respectively, which were much higher than those of MV, DS, ZC, KOS, and GTIC. Based on these results, we can draw nearly the same conclusions as those from the simulated benchmark datasets. However, we also should note that the KOS performed very poorly on both the LabelMe and Music Genre datasets. We believe that this was because some labelers only labeled a tiny portion of instances, and KOS failed to estimate the reliabilities of these labelers.

To explore the effect of the global sharing of the topological structure on both the feature and multiple noisy label spaces, we modified the parameters of the GIST descriptor to obtain different features on the LabelMe dataset. Based on these different features, the degree of sharing between the feature and multiple noisy label spaces was approximated as different. In the GIST descriptor, the number of blocks and scale of filters were fixed at 4. In addition, the number of orientations per scale was changed from 2 to 16, which indicates that the number of features varied from $2 * 4 * 4^2 = 128$ to $16 * 4 * 4^2 = 1024$. Figure 5 shows that the integration accuracy (%) improved as the number of different orientations per scale increased. These comparison results reveal that the integration accuracy (%) of our proposed MNLDP steady improved as the number of different orientations per scale increased until the number of generated features approximated $9 * 4 * 4^2 = 576$. These results also reveal the rationality for the global sharing of the topological structure between the feature and multiple noisy label spaces.

[3] http://people.csail.mit.edu/torralba/code/spatialenvelope

[4] http://fprodrigues.com//mturk-datasets.tar.gz

6 Conclusion and Feature Work

This study proposed an MNLDP method to improve the performance of label integration for crowdsourcing. MNLDP first transforms the multiple noisy label set of each instance into its multiple noisy label distribution and then propagates this distribution to its nearest neighbors. The extensive experimental results on simulated and real-world crowdsourced datasets show that, in many cases, our proposed MNLDP significantly outperformed all the other state-of-the-art label integration (consensus) methods used in comparison.

For simplicity, we naively assumed that the multiple noisy label space shared nearly the same topological structure with the feature space. We noticed that this global sharing assumption is quite simple and rough. Therefore, alleviating this assumption more or less to improve further the performance of the current MNLDP and strengthen its advantage is a main direction for our future work.

Acknowledgments

The work was partially supported by the National Natural Science Foundation of China (U1711267), the Open Research Project of Hubei Key Laboratory of Intelligent Geo-Information Processing (KLIGIP201601) and the Fundamental Research Funds for the Central Universities (CUG2018JM18 and CUGGC03).

References

[Aucouturier and Pachet, 2003] Jean-Julien Aucouturier and Francois Pachet. Representing musical genre: A state of the art. *Journal of New Music Research*, 32(1):83–93, 2003.

[Dawid and Skene, 1979] Alexander Philip Dawid and Allan M. Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied statistics*, 28(1):20–28, 1979.

[Demartini *et al.*, 2012] Gianluca Demartini, Djellel Eddine Difallah, and Philippe Cudré-Mauroux. Zencrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In *Proceedings of the 21st international conference on World Wide Web*, pages 469–478, 2012.

[Geng, 2016] Xin Geng. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748, 2016.

[Guan *et al.*, 2017] Melody Y. Guan, Varun Gulshan, Andrew M. Dai, and Geoffrey E. Hinton. Who said what: Modeling individual labelers improves classification. *arXiv preprint arXiv:1703.08774*, 2017.

[Hou *et al.*, 2016] Peng Hou, Xin Geng, and Min-Ling Zhang. Multi-label manifold learning. In *Association for the Advancement of Artificial Intelligence (AAAI)*, pages 1680–1686, 2016.

[Karger *et al.*, 2014] David R. Karger, Sewoong Oh, and Devavrat Shah. Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research*, 62(1):1–24, 2014.

[Li *et al.*, 2016] Chaoqun Li, Victor S. Sheng, Liangxiao Jiang, and Hongwei Li. Noise filtering to improve data and model quality for crowdsourcing. *Knowledge-Based Systems*, 107:96–103, 2016.

[Li *et al.*, 2019] Chaoqun Li, Liangxiao Jiang, and Wenqiang Xu. Noise correction to improve data and model quality for crowdsourcing. *Engineering Applications of Artificial Intelligence*, 82:184–191, 2019.

[Liu *et al.*, 2012] Qiang Liu, Jian Peng, and Alexander Ihler. Variational inference for crowdsourcing. In *Advances in Neural Information Processing Systems 25*, pages 692–700, 2012.

[Oliva and Torralba, 2001] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[Qiu *et al.*, 2018] Chen Qiu, Liangxiao Jiang, and Zhihua Cai. Using differential evolution to estimate labeler quality for crowdsourcing. In *Pacific Rim International Conference on Artificial Intelligence*, pages 165–173, 2018.

[Raykar *et al.*, 2010] Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. Learning from crowds. *Journal of Machine Learning Research*, 11(Apr):1297–1322, 2010.

[Rodrigues and Pereira, 2017] Filipe Rodrigues and Francisco C. Pereira. Deep learning from crowds. *arXiv preprint arXiv:1709.01779*, 2017.

[Rodrigues *et al.*, 2013] Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. Learning from multiple annotators: distinguishing good from random labelers. *Pattern Recognition Letters*, 34(12):1428–1436, 2013.

[Roweis and Saul, 2000] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[Russell *et al.*, 2008] Bryan C. Russell, Antonio Torralba, Kevin P. Murphy, and William T. Freeman. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*, 77(1-3):157–173, 2008.

[Sheng *et al.*, 2008] Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. Get another label? improving data quality and data mining using multiple noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622, 2008.

[Sheshadri and Lease, 2013] Aashish Sheshadri and Matthew Lease. Square: A benchmark for research on computing crowd consensus. In *First AAAI Conference on Human Computation and Crowdsourcing*, pages 156–164, 2013.

[Tian and Zhu, 2015] Tian Tian and Jun Zhu. Max-margin majority voting for learning from crowds. In *Advances in Neural Information Processing Systems*, pages 1612–1620, 2015.

[Wang and Zhang, 2008] Fei Wang and Changshui Zhang. Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, 20(1):55–67, 2008.

[Whitehill *et al.*, 2009] Jacob Whitehill, Ting fan Wu, Jacob Bergsma, Javier R. Movellan, and Paul L. Ruvolo. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in neural information processing systems*, pages 2035–2043, 2009.

[Zhang and Wu, 2018] Jing Zhang and Xindong Wu. Multi-label inference for crowdsourcing. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2738–2747, 2018.

[Zhang *et al.*, 2016a] Jing Zhang, Victor S. Sheng, Jian Wu, and Xindong Wu. Multi-class ground truth inference in crowdsourcing with clustering. *IEEE Transactions on Knowledge and Data Engineering*, 28(4):1080–1085, 2016.

[Zhang *et al.*, 2016b] Jing Zhang, Xindong Wu, and Victor S. Sheng. Learning from crowdsourced labeled data: a survey. *Artificial Intelligence Review*, 46(4):1–34, 2016.

[Zhu *et al.*, 2005] Xiaojin Zhu, John Lafferty, and Ronald Rosenfeld. *Semi-supervised learning with graphs*. PhD thesis, Carnegie Mellon University, language technologies institute, school of computer science, 2005.