Knowledge and Information Systems (2020) 62:2521–2538
https://doi.org/10.1007/s10115-020-01475-y

REGULAR PAPER

Label similarity-based weighted soft majority voting and pairing for crowdsourcing

Fangna Tao[1] · Liangxiao Jiang[1] · Chaoqun Li[2]

Received: 3 March 2020 / Accepted: 23 April 2020 / Published online: 14 May 2020

Abstract
Crowdsourcing services provide an efficient and relatively inexpensive approach to obtain substantial amounts of labeled data by employing crowd workers. It is obvious that the labeling qualities of crowd workers directly affect the quality of the labeled data. However, existing label aggregation strategies seldom consider the differences in the quality of workers labeling different instances. In this paper, we argue that a single worker may even have different labeling qualities on different instances. Based on this premise, we propose four new strategies by assigning different weights to workers when labeling different instances. In our proposed strategies, we first use the similarity among worker labels to estimate the specific quality of the worker on different instances, and then we build a classifier to estimate the overall quality of the worker across all instances. Finally, we combine these two qualities to define the weight of the worker labeling a particular instance. Extensive experimental results show that our proposed strategies significantly outperform other existing state-of-the-art label aggregation strategies.

Keywords Crowdsourcing · Label aggregation · Specific quality · Overall quality · Label similarity

1 Introduction

Supervised learning algorithms require extensive labeled data to train models and then make predictions on new data [7,8,25,30]. Conventional labeling tasks have been typically marked by domain experts or well-trained workers [19]. This kind of method provides high-quality labels, but is inefficient and expensive [10,11]. The social network service has supplied a novel method to resolve the labeling problem. In fact, programs such as the Listen game [21] have proven the feasibility of using public resources to address difficult machine learning problems [22]. Although these methods provide free-labeled data, guaranteeing their quality is difficult. Therefore, a more direct and economical method is to hire online crowd workers

✉ Liangxiao Jiang
ljiang@cug.edu.cn

[1] School of Computer Science, China University of Geosciences, Wuhan 430074, China

[2] School of Mathematics and Physics, China University of Geosciences, Wuhan 430074, China

to label the data. This has become possible thanks to the rapid growth of crowdsourcing platforms such as Amazon Mechanical Turk[1] and Crowdflower.[2]

These crowdsourcing platforms have been widely used to obtain extensive labeled data in applications such as ImageNet [3], computer vision [13], and natural language processing [9]. However, owing to differences in personal preferences and cognitive abilities, the quality of labels collected by a single crowd worker is often poor, which may compromise practical applications that use such data. To solve this problem, multiple labels are frequently requested from different workers for a single instance. Indeed, many existing works [20,29] have revealed the efficiency of repeated labeling. After each instance has been labeled by different crowd workers and, thus, obtains its multiple noisy label set, a label aggregation strategy is needed to infer the unknown true label from its multiple noisy label set, a method known as label aggregation (integration).

In recent years, label aggregation (integration) from multiple noisy labels has attracted much attention, and a large number of label aggregation strategies have been proposed [18, 28]: Dawid and Skene [1] proposed the DS strategy, which uses the maximum likelihood estimation to estimate a confusion matrix for each labeler and a class prior. Raykar et al. [16] proposed the RY strategy, which based on the Bayesian estimation to model the sensitivity and the specificity of labelers. Demartini et al. [2] proposed ZC strategy, which uses a two-element parameter to weight the reliability of a labeler. Karger et al. [9] proposed the KOS strategy based on the reliabilities of labelers to capture the presence of spammers. Zhang et al. [28] proposed the GTIC strategy based on Bayesian statistics for multi-class labeling. Ma et al. [14] proposed the FaitCrowd strategy, which uses a novel probabilistic Bayesian model to address the challenge of inferring fine grained source reliability. Zhang et al. [26] proposed the BLC strategy, which clusters two layers of features (conceptual-level and physical-level) to infer true labels of instances. Zhang et al. [24] proposed the MNLDP strategy, which considers the intercorrelation among multiple noisy label sets of different instances.

Of the numerous strategies, majority voting (MV) is the most straightforward, efficient, and widely used [4,6,19]. However, it discards a lot of useful information, such as the certainty information of the majority class and all the information of the minority class. To solve this problem, Sheng et al. [17] proposed four improved strategies, including two soft MV strategies and two paired soft MV strategies, to avoid this loss of information. Nevertheless, these strategies do not account for the labeling qualities of the crowd workers, especially the differences in the quality of workers labeling different instances. In other words, these strategies assume that different crowd workers have the same labeling quality, which is rarely true in real-world crowdsourcing scenarios.

To relax this assumption, in this paper, we propose four new strategies, including two weighted soft MV strategies and two weighted paired soft MV strategies, by assigning different weights to workers when labeling different instances. Specifically, we first use the similarity among worker labels to estimate the specific quality of the worker on different instances. Then, we build a classifier on the training set with the labels given by the worker and evaluate the classification accuracy on the test set as the overall quality of the worker across all instances. Finally, we combine these two qualities to define the weight of the worker labeling a particular instance. It can be seen that the differences in the quality of workers labeling different instances are considered in our proposed strategies. More importantly, the extensive empirical studies validate the effectiveness of our four newly proposed strategies.

[1] http://www.mturk.com.

[2] http://crowdflower.com.

The remainder of this paper is organized as follows. Our research starts from soft MV and pairing and, thus, we first provide a comprehensive introduction in Sect. 2. Then, we propose four new strategies in Sect. 3. The experiments and results are reported in Sect. 4. Some extensions to multi-class classification are discussed in Sect. 5. Finally, the conclusions are drawn and some main directions for future work are outlined in Sect. 6.

2 Soft MV and pairing

For a crowdsourcing system, a training instance set is defined as $E = \{e_i\}_{i=1}^{n}$, where each instance is $e_i =< x_i, y_i, \mathcal{L}_i >$, x_i is the feature vector, y_i is the unknown true label, and $\mathcal{L}_i = \{l_{ij}\}_{j=1}^{m}$ is the multiple noisy label set provided by m crowd workers for the ith instance x_i. For simplicity, in this paper, we provisionally restrict our discussion to binary classification, and thus both y_i and l_{ij} take values from a finite set $\{+, -\}$ only.

When each instance has only a multiple noisy label set, conventional supervised learning algorithms cannot learn a model from these instances directly. Thus, label aggregation strategies are required to infer the unknown true label from its multiple noisy label set. Of the numerous strategies, MV is the most straightforward, efficient, and widely used. However, it discards a lot of useful information, such as the certainty information of the majority class and all the information of the minority class. For example, there exist two instances with the multiple noisy label sets $\{+, +, +, +, -\}$ and $\{+, +, +, -, -\}$, respectively. According to MV, their aggregated (integrated) labels are of course the majority class $+$. However, the certainty (confidence) information of $+$ is ignored, which means that we cannot express the information regarding how "far off" they are from belonging to $+$. At the same time, all the information of the minority class $-$ is thoroughly discarded. As a result, we cannot distinguish between these two entirely different multiple noisy label sets, although the certainty (confidence) of them belonging to the majority class $+$ are totally different.

2.1 Soft MV

By exploiting the certainty information of the majority class, Sheng et al. [17] proposed two soft MV strategies: MV-Freq and MV-Beta. Similar to MV, MV-Freq and MV-Beta still use the majority class of a multiple noisy label set as the aggregated label, but at the same time assign a weight that represents the certainty of the majority class.

For MV-Freq, the certainty of the majority class is defined as the appearance frequency of the majority class in the multiple noisy label set. The detailed formula is

$$W_{H_i} = \begin{cases} P\left(+|\mathcal{L}_i\right), & P\left(+|\mathcal{L}_i\right) \geq P\left(-|\mathcal{L}_i\right) \\ P\left(-|\mathcal{L}_i\right), & P\left(+|\mathcal{L}_i\right) < P\left(-|\mathcal{L}_i\right) \end{cases}, \tag{1}$$

where $P\left(+|\mathcal{L}_i\right)$ (or $P\left(-|\mathcal{L}_i\right)$) is the certainty of the majority class $+$ (or $-$) of the multiple noisy label set $\mathcal{L}_i$ of the ith instance x_i, which can be estimated by

$$P(+|\mathcal{L}_i) = \frac{\sum_{j=1}^{m} \delta\left(l_{ij}, +\right)}{\sum_{j=1}^{m} \delta\left(l_{ij}, +\right) + \sum_{j=1}^{m} \delta\left(l_{ij}, -\right)}, \tag{2}$$

$$P(-|\mathcal{L}_i) = \frac{\sum_{j=1}^{m} \delta\left(l_{ij}, -\right)}{\sum_{j=1}^{m} \delta\left(l_{ij}, +\right) + \sum_{j=1}^{m} \delta\left(l_{ij}, -\right)}, \tag{3}$$

where l_{ij} is the class label provided by the jth worker for the ith instance, and $\delta(\cdot)$ is an indicator function that outputs 1 if its two parameters are identical, and 0 otherwise.

Please note that Eqs. (2)–(3) are a little different from those of the original paper by [17], which uses Laplace correction to reduce the effect of extreme probability estimations. However, to our knowledge, Laplace correction should be removed from these equations to reflect the true frequency of each class in the multiple noisy label set. More importantly, our experiments show that using Laplace correction reduces the performance of the related strategies to some extent. For saving space, we do not present the detailed experimental results in this paper.

Now, for the above two different multiple noisy label sets $\{+,+,+,+,-\}$ and $\{+,+,+,-,-\}$, their weights are $\frac{4}{5} = 0.8$ and $\frac{3}{5} = 0.6$, respectively. Therefore, they can be represented by $\{(+, 0.8)\}$ and $\{(+, 0.6)\}$, respectively.

For MV-Beta, the certainty of the majority class in the multiple noisy label set is defined as

$$W_{H_i} = \max\left\{I_{0.5}(\alpha_i, \beta_i), 1 - I_{0.5}(\alpha_i, \beta_i)\right\}, \tag{4}$$

where $I_{0.5}(\alpha_i, \beta_i)$ is the value of the cumulative distribution function (CDF) of the Beta distribution at the decision threshold 0.5. The detailed formula is

$$I_{0.5}(\alpha_i, \beta_i) = \sum_{k=\alpha_i}^{\alpha_i+\beta_i-1} \frac{(\alpha_i + \beta_i - 1)!}{k!(\alpha_i + \beta_i - 1 - k)!} 0.5^{\alpha_i+\beta_i-1}, \tag{5}$$

where α_i and β_i are two shape parameters of the Beta distribution, which are calculated by

$$\alpha_i = \sum_{j=1}^{m} \delta\left(l_{ij}, +\right) + 1. \tag{6}$$

$$\beta_i = \sum_{j=1}^{m} \delta\left(l_{ij}, -\right) + 1. \tag{7}$$

2.2 Paired soft MV

Just as shown in Sect. 2.1, MV-Freq and MV-Beta indeed exploit the certainty information of the majority class. However, similarly to the simplest MV, they also discard all the information regarding the minority class. According to the observations by [17], the information regarding the minority class is also very important, especially when there are only a few labels available in the multiple noisy label set.

By further exploiting the information about the minority class, Sheng et al. [17] proposed two paired soft MV strategies: Paired-Freq and Paired-Beta. Different from MV-Freq and MV-Beta, Paired-Freq and Paired-Beta generate a pair of weighted pairwise instances (a majority class instance and a minority class instance) from a single instance with a multiple noisy label set, where the weights of each pair of instances are defined as the certainty of the majority class and the certainty of the minority class, respectively.

For Paired-Freq, the certainty of the majority class is also calculated by Eqs. (1)–(3). After obtaining the certainty of the majority, the certainty of the minority class can be estimated by $1 - W_{H_i}$. Now, the above two different multiple noisy label sets $\{+,+,+,+,-\}$ and $\{+,+,+,-,-\}$ can be represented by $\{(+, 0.8), (-, 0.2)\}$ and $\{(+, 0.6), (-, 0.4)\}$, respectively. For Paired-Beta, the certainty of the majority class is also calculated by Eqs. (4)–(7). In the same way, the certainty of the minority class is $1 - W_{H_i}$.

3 Proposed strategies

Compared with the simplest MV, the four improved strategies [17] MV-Freq, MV-Beta, Paired-Freq, and Paired-Beta indeed avoid the loss of much information, such as the certainty information of the majority class and the certainty information of the minority class. However, none of these methods consider the labeling qualities of the crowd workers, especially the differences in the quality of workers labeling different instances. In other words, all of them assume that different crowd workers have the same labeling quality, which is rarely true in real-world crowdsourcing scenarios.

In many real-world crowdsourcing scenarios, to the best of our knowledge, even a high-quality worker may provide an incorrect label for a particular instance, whereas a low-quality worker may provide a correct label. Assume that the same worker has the same labeling quality on different instances; the influence of incorrect labeling from the high-quality workers will be strengthened, whereas the influence of correct labeling from the low-quality workers will be weakened. We call this phenomenon "quality inversion." For example, for a particular instance with a multiple noisy label set $\{+, +, +, -, -\}$, if we do not account for the labeling qualities of the crowd workers, Paired-Freq represents it as $\{(+, 0.6), (-, 0.4)\}$. However, suppose that these five workers have different labeling qualities, such as $\{0.95, 0.6, 0.94, 0.92, 0.59\}$, on this instance, this can be represented as $\{(+, 0.6225), (-, 0.3775)\}$. By taking the labeling quality into account, we can scale up the certainty of the majority class $+$ and reduce the certainty of the minority class $-$. Again suppose that each of these five workers has the same labeling quality on another instance with a multiple noisy label set $\{-, -, +, +, +\}$, then this instance is represented as $\{(+, 0.6125), (-, 0.3875)\}$. Thus, the certainty of the majority class $+$ decreases slightly, whereas the certainty of the minority class $-$ increases slightly. In other words, for this instance, the influence of incorrect labeling $(-)$ from the high-quality worker (the first worker with the labeling quality 0.95) is strengthened, whereas the influence of correct labeling $(+)$ from the low-quality worker (the last worker with the labeling quality 0.59) is weakened.

To deal with the phenomenon of "quality inversion" discussed previously, in this paper, we argue that the same worker may also have different labeling qualities on different instances. Based on this premise, we propose four new strategies, including two weighted soft MV strategies and two weighted paired soft MV strategies, by assigning different weights to workers when labeling different instances. Specifically, a label similarity-based weighting method that combines the specific quality of the worker on different instances and the overall quality of the worker across all instances is proposed to estimate the weight of each crowd label. We simply denote the resulting strategies by WMV-Freq, WMV-Beta, WPaired-Freq, and WPaired-Beta, respectively.

3.1 Weighted soft MV

Similar to MV, MV-Freq and MV-Beta also assume that all crowd workers have the same labeling quality. To improve MV-Freq and MV-Beta, we propose two weighted soft MV strategies: WMV-Freq and WMV-Beta, respectively.

For WMV-Freq, the weight formula is the same as Eq. (1). We repeat it here for convenience:

$$W_{H_i} = \begin{cases} P(+|\mathcal{L}_i), & P(+|\mathcal{L}_i) \geq P(-|\mathcal{L}_i) \\ P(-|\mathcal{L}_i), & P(+|\mathcal{L}_i) < P(-|\mathcal{L}_i) \end{cases}, \tag{8}$$

where $P(+|\mathcal{L}_i)$ (or $P(-|\mathcal{L}_i)$) is also the certainty of the majority class $+$ (or $-$) of the multiple noisy label set $\mathcal{L}_i$ of the ith instance x_i, but they are estimated using Eqs. (9)–(10) instead of Eqs. (2)–(3), respectively:

$$P(+|\mathcal{L}_i) = \frac{\sum_{j=1}^{m} w_{ij}\delta\left(l_{ij}, +\right)}{\sum_{j=1}^{m} w_{ij}\delta\left(l_{ij}, +\right) + \sum_{j=1}^{m} w_{ij}\delta\left(l_{ij}, -\right)}, \tag{9}$$

$$P(-|\mathcal{L}_i) = \frac{\sum_{j=1}^{m} w_{ij}\delta\left(l_{ij}, -\right)}{\sum_{j=1}^{m} w_{ij}\delta\left(l_{ij}, +\right) + \sum_{j=1}^{m} w_{ij}\delta\left(l_{ij}, -\right)}, \tag{10}$$

where w_{ij} is the weight of l_{ij}.

For WMV-Beta, the weight formula is the same as Eq. (4). We also repeat it here for convenience:

$$W_{H_i} = \max\left\{I_{0.5}(\alpha_i, \beta_i), 1 - I_{0.5}(\alpha_i, \beta_i)\right\}, \tag{11}$$

where

$$I_{0.5}(\alpha_i, \beta_i) = \sum_{k=[\alpha_i]}^{[\alpha_i+\beta_i]-1} \frac{([\alpha_i + \beta_i] - 1)!}{k!([\alpha_i + \beta_i] - 1 - k)!} 0.5^{[\alpha_i+\beta_i]-1}, \tag{12}$$

where $[\cdot]$ is an integer-valued function, α_i and β_i are also two shape parameters of the Beta distribution, but they are estimated using Eqs. (13)–(14) instead of Eqs. (6)–(7), respectively,

$$\alpha_i = \sum_{j=1}^{m} w_{ij}\delta\left(l_{ij}, +\right) + 1, \tag{13}$$

$$\beta_i = \sum_{j=1}^{m} w_{ij}\delta\left(l_{ij}, -\right) + 1. \tag{14}$$

3.2 Weighted paired soft MV

Similar to MV-Freq and MV-Beta, WMV-Freq and WMV-Beta also only use the certainty information of the majority class and discard all the information of the minority class. Consequently, to improve WMV-Freq and WMV-Beta, we also adapt Paired-Freq and Paired-Beta to propose two weighted paired soft MV strategies: WPaired-Freq and WPaired-Beta, respectively.

For WPaired-Freq, the certainty of the majority class is also calculated using Eqs. (8)–(10). After we obtain the certainty of the majority, the certainty of the minority class can be estimated by $1 - W_{H_i}$. For WPaired-Beta, the certainty of the majority class is also calculated using Eqs. (11)–(14). In the same way, the certainty of the minority class can be estimated by $1 - W_{H_i}$.

3.3 Label similarity-based weighting

Now, the only question left to answer is how to define the weight w_{ij} of each crowd worker labeling a particular instance. Generally speaking, there are mainly two kinds of methods to define (learn) such weights. The first is to conduct a sophisticated search process to find the weights that maximize the performance of the resulting model. Usually, this kind of method leads to a good weight assignment, but it requires a significant amount of time and

an appropriate fitness function for the search. The other is to directly compute the weights using the statistical characteristics of the available data, and thus it is often more efficient.

In this paper, we focus our attention on the second method and propose a label similarity-based weighting method, which combines the specific quality of the worker on different instances and the overall quality of the worker across all instances to estimate the weight w_{ij} of each crowd label. We expect that the learned weights could weaken the influence of incorrect labeling on high-quality workers and strengthen the influence of correct labeling on low-quality workers. Inspired by [12], we define the normalized weight w_{ij} of each crowd label as

$$w_{ij} = \frac{1}{Z} w'_{ij}, \tag{15}$$

where Z is a normalization constant, which ensures that the sum of all crowd label weights for the ith instance is still equal to m, the detailed formula is Eq. (16). w'_{ij} is the non-normalized weight of each crowd label defined by Eq. (17).

$$Z = \frac{1}{m} \sum_{j=1}^{m} w'_{ij}, \tag{16}$$

$$w'_{ij} = \frac{1}{1 + e^{-\gamma_{ij}}}, \tag{17}$$

where γ_{ij} is estimated by

$$\gamma_{ij} = \tau_j \left(1 + s_{ij}^2\right), \tag{18}$$

where τ_j is the overall quality of the jth worker across all instances, s_{ij} is the specific quality of the jth worker for the ith instance, s_{ij}^2 is used to strengthen the influence of the specific quality s_{ij}, and $1 + s_{ij}^2$ is used to avoid the effect of the extreme estimation of $s_{ij} = 0$.

Next, we introduce how to estimate s_{ij}. Inspired by the similarity assumption [12], we propose to use the similarity among worker labels to estimate the specific qualities of the same worker for different instances. For a specific instance e_i, if the jth worker uses the same label as most other workers, this indicates that the worker has a high degree of confidence in this instance. That is, the specific quality of the jth worker for the ith instance is very high. Based on this idea, we can define s_{ij} as the label similarity among workers:

$$s_{ij} = \sum_{j'=1 \wedge j' \neq j}^{m} \delta\left(l_{ij}, l_{ij'}\right). \tag{19}$$

We now introduce how to estimate τ_j. Estimating the overall qualities of different workers is not a new research topic in the crowdsourcing learning community. To the best of the authors' knowledge, there exist many state-of-the-art algorithms, such as Dawid–Skene [1], ZenCrowd , KOS [9], and DEW [15,23]. However, none of them exploit feature vectors of instances, which makes it impossible to take full advantage of the statistical characteristics of the available data when evaluating the label qualities. According to the observation by [30], in traditional supervised learning, there exists a schema to exhibit the relationship between data features and the ground-truth labels. For example, suppose there exists a high-quality worker; the data schema will be well-inherited in their labels, because the difference between their labels and ground-truth labels is small. Meanwhile, suppose there exists a low-quality worker, the data schema may be broken because their labels will be very different from the ground-truth labels. Therefore, we can estimate the overall quality of a worker by evaluating

how well the schema is inherited in their labels. Specifically, we can first extract all training instances' feature vectors and the corresponding crowd labels provided by the jth worker to form a new single-label data set. Then, we use tenfold cross-validation to evaluate the classification accuracy of a classifier. In theory, this classifier can be any classifier. Finally, we define the overall quality of the jth worker as the classification accuracy of the built classifier. The detailed formula can be expressed as

$$\tau_j = \frac{\sum_{i=1}^{n} \delta\left(l_{ij}, f_j\left(x_i\right)\right)}{n}, \tag{20}$$

where n is the size of the extracted data set and $f_j(x_i)$ is the class label of the feature vector x_i predicted by the built classifier.

It can be seen that although the existing FaitCrowd strategy [14] also considers the quality of workers on different tasks, our label similarity-based weighting method is totally different from it. The FaitCrowd strategy jointly models question content and source answering behavior to learn latent topics and estimate the topical source expertise. By contrast, our method directly uses the similarity among worker labels to estimate the specific quality of the worker on different instances. Yet at the same time, our method is totally different from the existing BLC strategy [26] that also takes the features of instances into account. The BLC strategy utilizes the conceptual-level features extracted from crowdsourced labels to infer the true labels of instances by clustering. By contrast, our method only uses the original features of instances to build a classifier to estimate the overall quality of the worker across all instances.

4 Experiments and results

The purpose of this section is to validate the effectiveness of our proposed strategies: WMV-Freq, WMV-Beta, WPaired-Freq, and WPaired-Beta. Therefore, we designed four groups of experiments to compare them with the original MV-Freq, MV-Beta, Paired-Freq, and Paired-Beta, respectively. We conducted our experiments on 12 real-world datasets from the University of California at Irvine (UCI) repository [5] listed in Table 1, which includes all nine binary datasets from the website of the CEKA platform [27] and three transformed binary datasets used in [17].

To simulate a crowdsourcing process to obtain multiple noisy labels of each instance, the original true labels of all instances were hidden, and all simulated workers were employed to label each instance. For each worker, the original true label was assigned to each instance with the probability p, and the opposite value was assigned with the probability $1 - p$. In our experiments, the labeling quality p of each worker was generated randomly from a uniform distribution on the interval $(0.3, 0.9)$. In fact, in our experiments, we also tested some other distributions, such as the normal (Gaussian) distribution $N(0.65, 0.35^2)$, to randomly generate the labeling quality of each worker. Owing to virtually the same experimental conclusions and for brevity, we do not present the detailed experimental results here.

After obtaining the multiple noisy label set of each instance, we use label aggregation strategies to infer its aggregation label. Then, the classifier is built on the training set with the aggregation labels and evaluated on the test set with the true labels. Because the simulation process has a certain degree of randomness, we use tenfold cross-validation to evaluate the classification accuracy of the built classifier. In our experiment, we use C4.5, one of the top 10 data mining algorithms, to estimate the overall quality of each worker τ_j $(j = 1, 2, \ldots, m)$ in our proposed strategies and evaluate the performance of all label aggregation strategies.

Table 1 Descriptions of the used datasets

Dataset	#Features	#Instances	#Positives	#Negatives
kr–vs–kp	37	3196	1669	1527
mushroom	22	8124	4208	3916
sick	30	3772	231	3541
spambase	58	4601	1813	2788
tic–tac–toe	10	958	332	626
splice	61	3190	1535	1655
thyroid	30	3772	291	3481
waveform	41	5000	1692	3308
biodeg	42	1055	699	356
horse-colic	23	368	136	232
ionosphere	35	351	126	225
vote	17	435	267	168

Figures 1, 2, 3, and 4 show the detailed classification accuracy (%) comparison results between our proposed four strategies and their original counterparts, respectively. From these comparison results, we can see that assigning different weights to different workers when labeling different instances can largely improve the performance of the existing label aggregation strategies. Now, we summarize some of the highlights.

1. Our proposed two weighted soft MV strategies (WMV-Freq and WMV-Beta) are better overall than the original two soft MV strategies (MV-Freq and MV-Beta). Our proposed two weighted paired soft MV strategies (WPaired-Freq and WPaired-Beta) are also better overall than the original two paired soft MV strategies (Paired-Freq and Paired-Beta). All these results validate our viewpoints: different workers should have different labeling qualities and the same worker should also have different labeling qualities on different instances.
2. The accuracies of our weighted strategies, WMV-Freq, WMV-Beta, WPaired-Freq, and WPaired-Beta, are much higher than those of the original MV-Freq, MV-Beta, Paired-Freq, and Paired-Beta, respectively. However, the advantages between our weighted strategies and the original strategies gradually degraded as the number of workers increased.
3. As expected, the accuracies of our weighted strategies and the original strategies rapidly upgraded as the number of workers increased. However, the same as Paired-Freq [17], we also notice that the performance of WPaired-Freq does not produce an expected increment when more and more labels for each instance are available. Its learning curves are completely flat and even fall back a little over three datasets (i.e., "biodeg", "ionosphere", and "vote"). Why does WPaired-Freq perform so? The fundamental reason is that WPaired-Freq also keeps the noise completely. Suppose there exists an instance with a multiple noisy label set $\{+, +, +, -, -\}$ and the labeling qualities of these five workers are 0.95, 0.6, 0.94, 0.92, and 0.59, respectively. WPaired-Freq represents it as $\{(+, 0.6225), (-, 0.3775)\}$. If these five workers label this instance twice, its multiple noisy label set becomes $\{+, +, +, -, -, +, +, +, -, -\}$. However, WPaired-Freq represents it as $\{(+, 0.6225), (-, 0.3775)\}$ as well. It can be seen that as more labels are acquired for each instance, the certainty of the majority class $+$ and the certainty of the minority class $-$ do not change anymore. By contrast, WPaired-Beta

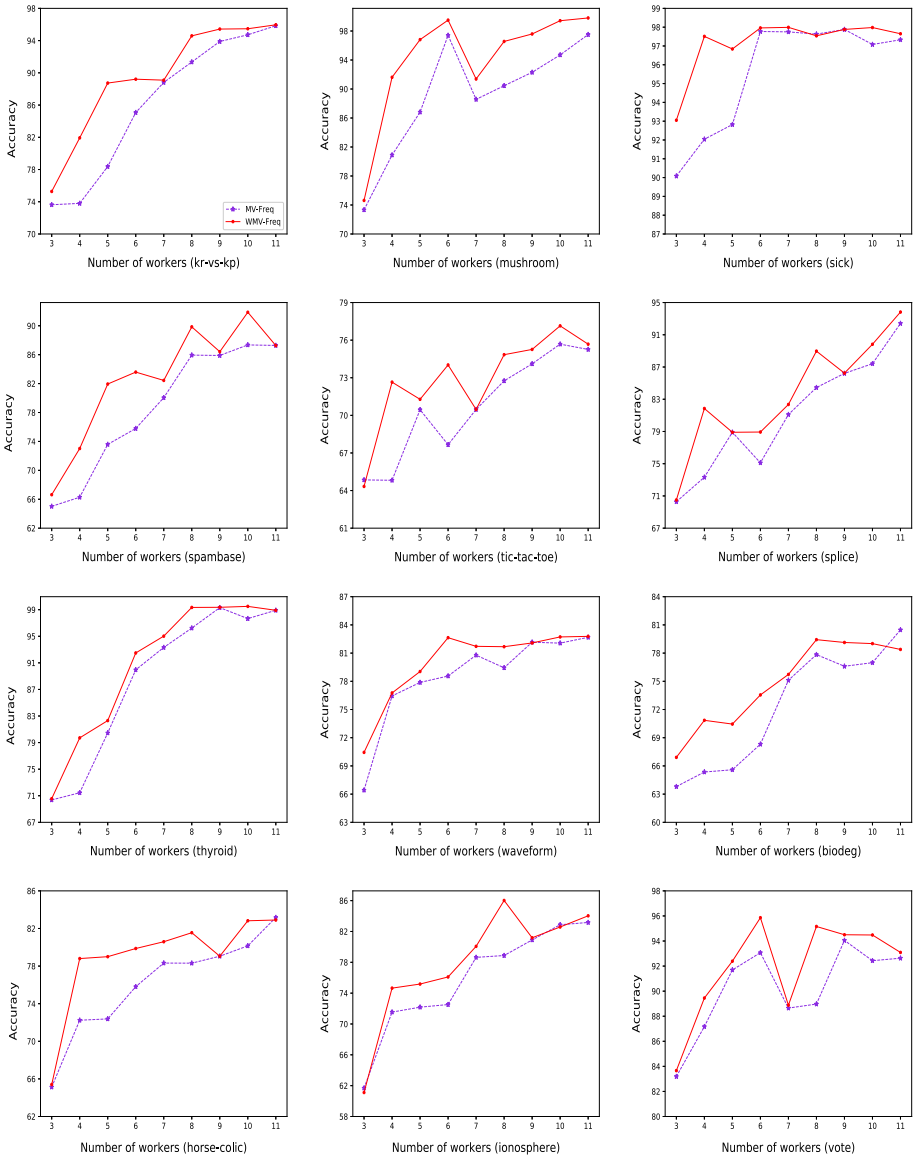


Fig. 1 Classification accuracy (%) comparisons for WMV-Freq versus MV-Freq. The labeling quality $p \in (0.3, 0.9)$

does not incur this issue. For the same example, WPaired-Beta represents them as $\{(+, 0.6563), (-, 0.3437)\}$ and $\{(+, 0.8867), (-, 0.1133)\}$, respectively. That is to say, as more labels are acquired for each instance, the certainty of the majority class $+$ keeps rising and the certainty of the minority class $-$ continues to decrease, which means that the influence of correct labeling $(+)$ is further strengthened and the influence of incorrect labeling $(-)$ is further weakened.

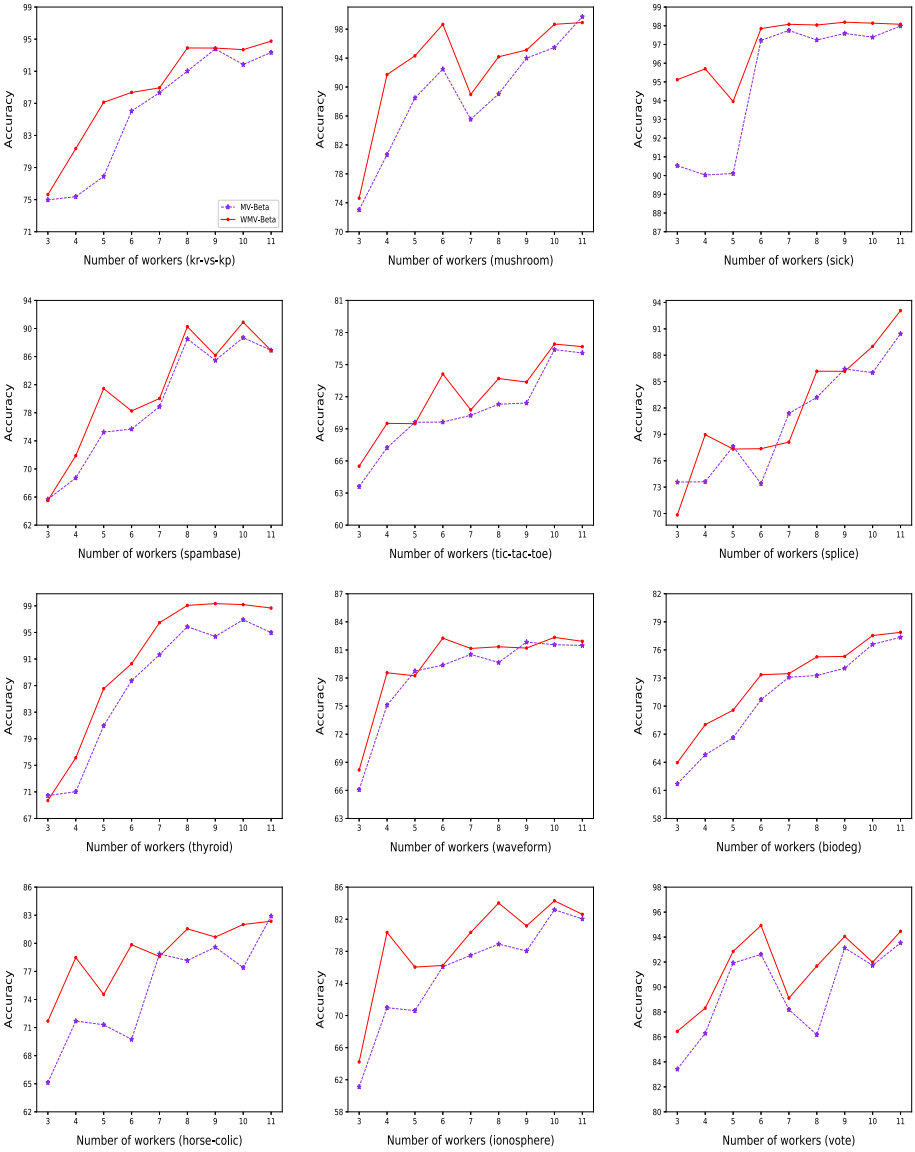


Fig. 2 Classification accuracy (%) comparisons for WMV-Beta versus MV-Beta. The labeling quality $p \in (0.3, 0.9)$

To further validate the effectiveness of our proposed four strategies, we performed another group of experiments to compare them with some other existing state-of-the-art label aggregation strategies such as ZC [2], RY [16], KOS [9], and GTIC [28]. Owing to virtually the same experimental conclusions and for brevity, we only show the detailed comparison results when the number of workers is six. Table 2 shows the detailed comparison results in terms of the classification accuracy of the target classifier. From these comparison results, we can see that the average classification accuracies (79.14%, 79.5%, 80.93%, and 83.66%)

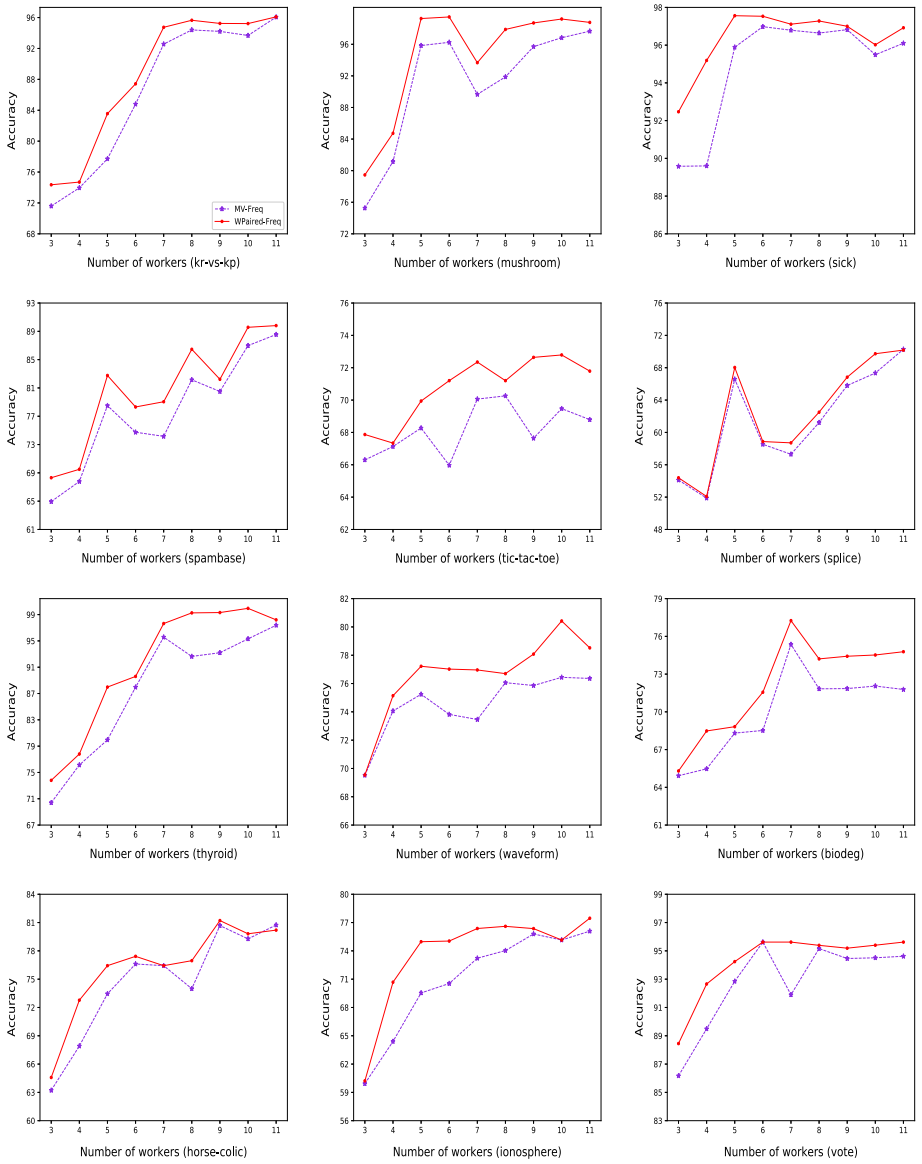


Fig. 3 Classification accuracy (%) comparisons for WPaired-Freq versus Paired-Freq. The labeling quality $p \in (0.3, 0.9)$

of our proposed four strategies are all much higher than those of ZC (77.47%), RY (78.41%), KOS (75.65%), and GTIC (76.44%). Besides, we also observed the performance of our proposed four strategies in terms of the integration accuracy, which is defined as the proportion of instances whose integration labels are the same as their true labels. Table 3 shows the detailed comparison results. From these comparison results, we can see that the average integration accuracies (87.72%, 87.57%, 87.95%, and 88.92%) of our proposed four strategies are all also much higher than those of ZC (83.54%), RY (86.52%), KOS (87.33%), and GTIC (83.23%).

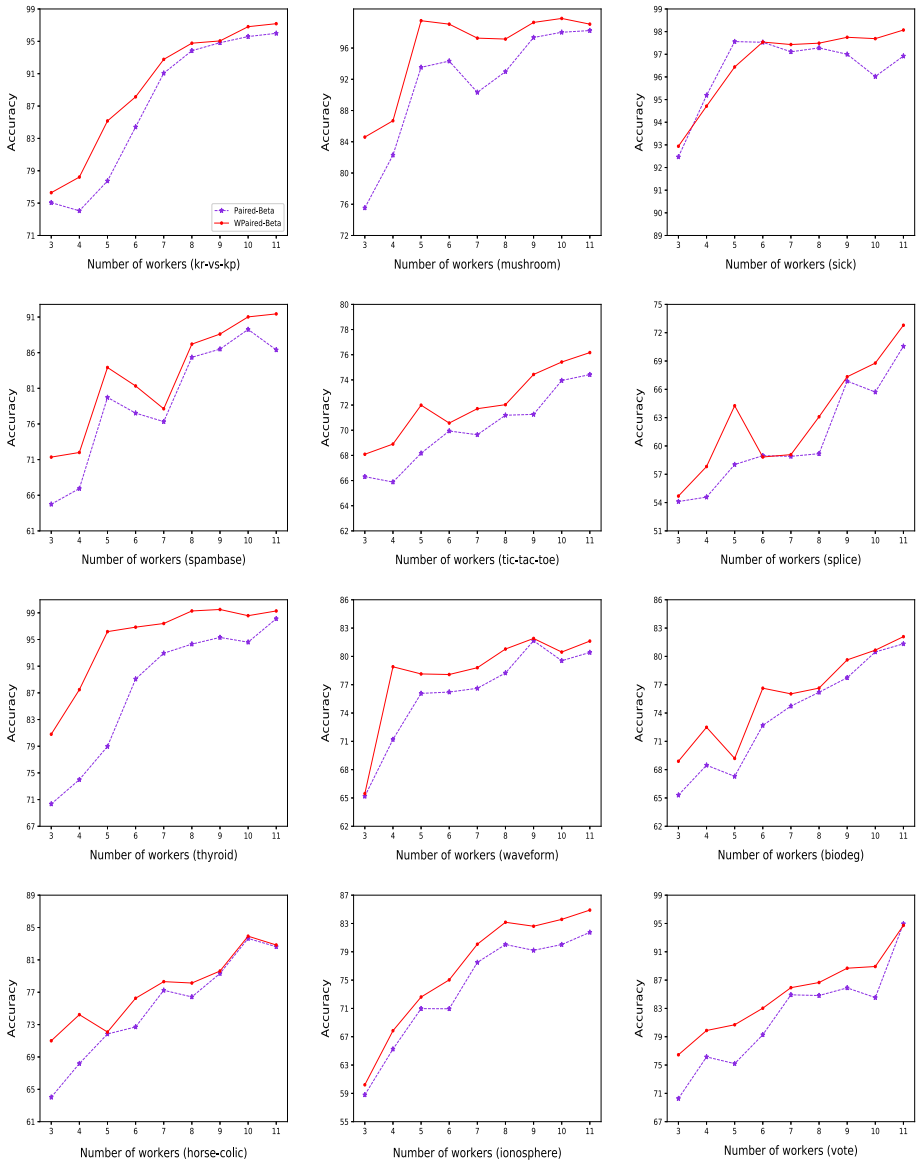


Fig. 4 Classification accuracy (%) comparisons for WPaired-Beta versus Paired-Beta. The labeling quality $p \in (0.3, 0.9)$

5 Discussion

As we have validated the effectiveness of our proposed four strategies for binary classification, we now discuss and extend the proposed new strategies to multi-class classification in this section.

At first, we focus on how to define the uncertainty of the majority class when multi-class classification is need. Given a multiple noisy label set $\mathcal{L}_i$ of the ith instance x_i, we can directly

Table 2 Classification accuracy (%) comparisons for our proposed four strategies versus ZC, RY, KOS, and GTIC

Dataset	ZC	RY	KOS	GTIC	WMV -Freq	WMV -Beta	WPaired -Freq	WPaired -Beta
kr–vs–kp	93.06	98.28	97.93	92.09	94.87	94.68	96.56	96.53
mushroom	84.71	89.89	89.9	84.66	94.07	93.49	98.66	98.58
sick	88.39	90.56	88.22	87.64	89.47	89.15	95.94	97.67
spambase	80.96	74.58	74.87	79.11	80.74	80.74	80.03	81.4
tic–tac–toe	69.02	71.82	69.54	65.97	64.11	65.76	67.34	74.85
splice	50.94	56.52	58.12	50.94	51.22	61.41	50.94	65.08
thyroid	91.99	90.74	89.47	90.27	92.07	92.18	91.91	96.29
waveform	78.1	74.86	74.66	77.76	78.14	79.54	82.1	79.36
biodeg	64.92	58.88	58.59	62.92	68.7	65.3	71.08	73.55
horse-colic	64.51	61.27	58.03	64.21	66.66	63.67	63.7	65.59
ionosphere	79.5	82.33	63.1	80.07	80.09	80.65	80.51	82.32
vote	83.49	91.24	85.38	81.64	89.48	87.39	92.43	92.64
Average	77.47	78.41	75.65	76.44	79.14	79.5	80.93	83.66

Table 3 Integration accuracy (%) comparisons for our proposed four strategies versus ZC, RY, KOS, and GTIC

Dataset	ZC	RY	KOS	GTIC	WMV -Freq	WMV -Beta	WPaired -Freq	WPaired -Beta
kr–vs–kp	88.31	93.54	91.69	82.77	89.54	92.62	92.62	90.77
mushroom	100	100	100	100	100	100	100	100
sick	71.24	91.29	98.68	98.42	81	85.75	93.14	97.63
spambase	91.97	91.97	91.97	81.34	92.19	91.11	90.89	92.84
tic–tac–toe	79.61	75.73	78.64	72.82	79.61	79.61	74.76	79.61
splice	82.45	80.25	82.45	93.73	93.1	92.48	82.45	83.07
thyroid	76.52	92.08	95.42	97.89	95.51	96.83	91.82	99.21
waveform	77.2	74.4	76	77.8	77.8	78.2	78.2	78
biodeg	72.73	80	76.36	78.18	77.27	71.82	77.27	70.91
horse-colic	90.91	90.91	88.64	85.91	90.91	90.91	90.91	90.91
ionosphere	77.78	80.56	80.56	69.44	77.78	77.78	91.67	86.11
vote	93.75	87.5	87.5	60.42	97.92	93.75	91.67	97.92
Average	83.54	86.52	87.33	83.23	87.72	87.57	87.95	88.92

borrow the definitions on the impurity of a given decision-tree node to define its uncertainty of the majority class. In decision-tree learning, *Error*, *Gini* and *Entropy* have been widely used for measure the impurity of a given decision-tree node. The detailed definitions are

$$\mathrm{Error}(\mathcal{L}_i) = 1 - \arg\max_{j=1}^{q} P(c_k|\mathcal{L}_i), \tag{21}$$

$$\mathrm{Gini}(\mathcal{L}_i) = 1 - \sum_{k=1}^{q} P(c_k|\mathcal{L}_i)^2, \tag{22}$$

$$\text{Entropy}(\mathcal{L}_i) = -\sum_{k=1}^{q} P(c_k|\mathcal{L}_i) \log_2 P(c_k|\mathcal{L}_i), \tag{23}$$

where q is the number of classes and $0 \log_2 0 = 0$.

Then, the corresponding certainty of the majority class are defined as

$$W_{H_i} = 1 - \text{Error}(\mathcal{L}_i) = \arg\max_{j=1}^{q} P(c_k|\mathcal{L}_i), \tag{24}$$

$$W_{H_i} = 1 - \text{Gini}(\mathcal{L}_i) = \sum_{k=1}^{q} P(c_k|\mathcal{L}_i)^2, \tag{25}$$

$$W_{H_i} = 1 - \text{Entropy}(\mathcal{L}_i) = 1 + \sum_{k=1}^{q} P(c_k|\mathcal{L}_i) \log_2 P(c_k|\mathcal{L}_i), \tag{26}$$

where $P(c_k|\mathcal{L}_i)$ is the appearance frequency of class c_k in $\mathcal{L}_i$ estimated by

$$P(c_k|\mathcal{L}_i) = \frac{\sum_{j=1}^{m} w_{ij} \delta\left(l_{ij}, c_k\right)}{\sum_{k=1}^{q} \sum_{j=1}^{m} w_{ij} \delta\left(l_{ij}, c_k\right)}, \tag{27}$$

where

$$w_{ij} = \frac{1}{Z} \frac{1}{1 + (q-1)e^{-\gamma_{ij}}}. \tag{28}$$

where Z is a normalization constant.

At last, weighted pairing can also be extended by decomposing each instance with a multiple noisy label set $\mathcal{L}_i$ into q class-specific weighted instances, where the weight of each class-specific instance is defined as the certainty of each class $P(c_k|\mathcal{L}_i)$, respectively.

6 Conclusion and future work

In this paper, we have argued that a single worker may even have different labeling qualities on different instances. Based on this premise, we have proposed two weighted soft MV strategies and two weighted paired soft MV strategies. We have simply denoted the resulting strategies as WMV-Freq, WMV-Beta, WPaired-Freq, and WPaired-Beta, respectively. In addition, we have proposed a label similarity-based weighting method, which combines the specific quality of the worker on different instances and the overall quality of the worker across all instances to estimate the weight of each worker labeling different instances. The experimental results have validated the effectiveness of our proposed four new strategies.

Given a weighted multiple noisy label set, the definition of the certainty of the majority class is a crucial problem in our proposed strategies, and thus exploring some other effective definitions is the main direction for our future work. In addition, exploiting some other sophisticated weight learning method is another interesting topic for future work.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. U1711267) and Fundamental Research Funds for the Central Universities (Grant No. CUGGC03).

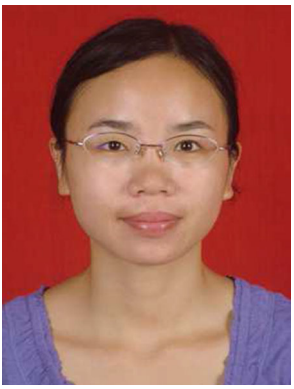
References

1. Dawid AP, Skene AM (1979) Maximum likelihood estimation of observer error-rates using the EM algorithm. J R Stat Soc Ser C (Appl Stat) 28(1):20–28
2. Demartini G, Difallah DE, Cudré-Mauroux P (2012) Zencrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In: Proceedings of the 21st World Wide Web conference 2012, WWW 2012, Lyon, France, pp 469–478
3. Deng J, Dong W, Socher R, Li L-J, Li K, Li F-F (2009) Imagenet: a large-scale hierarchical image database. In Proceedings of conference on computer vision and pattern recognition, (CVPR 2009), Miami, Florida, pp 248–255
4. Donmez P, Carbonell JG, Schneider JG (2009) Efficiently learning the accuracy of labeling sources for selective sampling. In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, Paris, pp 259–268
5. Dua D, Karra TE (2017) UCI machine learning repository
6. Ipeirotis PG, Provost FJ, Sheng VS, Wang J (2014) Repeated labeling using multiple noisy labelers. Data Min Knowl Discov 28(2):402–441
7. Jiang L, Kong G, Li C (2019) Wrapper framework for test-cost-sensitive feature selection. In: IEEE transactions on systems man cybernetics-systems, pp 1–10
8. Jiang L, Zhang L, Liangjun Y, Wang D (2019) Class-specific attribute weighted naive bayes. Pattern Recognit 88:321–330
9. Karger DR, Sewoong O, Shah D (2014) Budget-optimal task allocation for reliable crowdsourcing systems. Oper Res 62(1):1–24
10. Li C, Jiang L, Wenqiang X (2019) Noise correction to improve data and model quality for crowdsourcing. Eng Appl Artif Intell 82:184–191
11. Li C, Sheng VS, Jiang L, Li H (2016) Noise filtering to improve data and model quality for crowdsourcing. Knowl Based Syst 107:96–103
12. Li J, Baba Y, Kashima H (2018) Incorporating worker similarity for label aggregation in crowdsourcing. In: Proceedings of the 27th international conference on artificial neural networks, ICANN 2018, Rhodes, pp 596–606
13. Liu Q, Peng J, Ihler AT (2012) Variational inference for crowdsourcing. In: Proceedings of the 26th annual conference on neural information processing systems 2012, Lake Tahoe, pp 701–709
14. Ma F, Li Y, Li Q, Qiu M, Gao J, Zhi S, Su L, Zhao B, Ji H, Han J (2015) Faitcrowd: fine grained truth discovery for crowdsourced data aggregation. In: Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, Sydney, pp 745–754
15. Qiu C, Jiang L, Cai Z (2018) Using differential evolution to estimate labeler quality for crowdsourcing. In: PRICAI 2018: trends in artificial intelligence 15th pacific rim international conference on artificial intelligence, Proceedings, Part II, Nanjing, China, pp 165–173
16. Raykar VC, Shipeng Y, Zhao LH, Valadez GH, Florin C, Bogoni L, Moy L (2010) Learning from crowds. J Mach Learn Res 11:1297–1322
17. Sheng VS, Zhang J, Bin G, Xindong W (2019) Majority voting and pairing with multiple noisy labeling. IEEE Trans Knowl Data Eng 31(7):1355–1368
18. Sheshadri A, Lease M (2013) SQUARE: a benchmark for research on computing crowd consensus. In: Proceedings of the first AAAI conference on human computation and crowdsourcing, HCOMP 2013 (November), Palm Springs. CA, USA, pp 7–9
19. Tian T, Zhu J, Qiaoben Y (2019) Max-margin majority voting for learning from crowds. IEEE Trans Pattern Anal Mach Intell 41(10):2480–2494
20. Tu J, Yu G, Domeniconi C, Wang J, Xiao G, Guo M (2018) Multi-label answer aggregation based on joint matrix factorization. In: Proceedings of the IEEE international conference on data mining, ICDM 2018, Singapore, pp 517–526
21. Turnbull D, Liu R, Barrington L, Lanckriet GRG (2007) A game-based approach for collecting semantic annotations of music. In: Proceedings of the 8th international conference on music information retrieval, ISMIR 2007, Vienna, Austria, pp 535–538
22. Whitehill J, Ruvolo P, Wu T, Bergsma J, Movellan JR (2009) Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In: Proceedings of the 23rd annual conference on neural information processing systems 2009, Vancouver, pp 2035–2043
23. Zhang H, Jiang L, Xu W (2018) Differential evolution-based weighted majority voting for crowdsourcing. In: Proceedings of the 15th pacific rim international conference on artificial intelligence 2018, Nanjing, pp 228–236

24. Zhang H, Jiang L, Xu W (2019) Multiple noisy label distribution propagation for crowdsourcing. In: Proceedings of the twenty-eighth international joint conference on artificial intelligence, IJCAI 2019, Macao, pp 1473–1479
25. Zhang H, Jiang L, Liangjun Y (2020) Class-specific attribute value weighting for naive bayes. Inf Sci 508:260–274
26. Zhang J, Sheng VS, Li T (2017) Label aggregation for crowdsourcing with bi-layer clustering. In: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval, Shinjuku, Tokyo, Japan, pp 921–924
27. Zhang J, Sheng VS, Nicholson B, Xindong W (2015) CEKA: a tool for mining the wisdom of crowds. J Mach Learn Res 16:2853–2858
28. Zhang J, Sheng VS, Jian W, Xindong W (2016) Multi-class ground truth inference in crowdsourcing with clustering. IEEE Trans Knowl Data Eng 28(4):1080–1085
29. Zhang J, Wu X (2018) Multi-label inference for crowdsourcing. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery and data mining, KDD 2018, London, pp 2738–2747
30. Zhong J, Yang P, Tang K (2017) A quality-sensitive method for learning from crowds. IEEE Trans Knowl Data Eng 29(12):2643–2654

Fangna Tao received her B.Sc. degree in June 2018. She is currently a M.Sc. student in the School of Computer Science, China University of Geosciences (Wuhan). Her research interests include machine learning and data mining.

Liangxiao Jiang received his Ph.D. degree from China University of Geosciences, Wuhan, China, in 2009. He is currently a professor with the School of Computer Science, China University of Geosciences (Wuhan). His current research interests include machine learning and data mining. Since 2005, he has published over 70 refereed journal and conference papers, such as in the IEEE Transactions on Knowledge and Data Engineering, Pattern Recognition, Information Sciences, Knowledge and Information Systems, Engineering Applications of Artificial Intelligence, Knowledge-Based Systems, Expert Systems with Applications, Pattern Recognition Letters, IJCAI, AAAI, ICML, ICDM and DASFAA, in the above areas.



Chaoqun Li received her Ph.D. degree from China University of Geosciences, Wuhan, China, in 2012. She is currently an associate professor with the School of Mathematics and Physics, China University of Geosciences (Wuhan). Her current research interests include data mining and machine learning. Since 2006, he has published over 30 refereed journal and conference papers, such as in the Information Sciences, Knowledge and Information Systems, Engineering Applications of Artificial Intelligence, Knowledge-Based Systems, Expert Systems with Applications, Pattern Recognition Letters, International Journal of Pattern Recognition and Artificial Intelligence, ICANN, ICTAI and PRICAI, in the above areas.