Multimedia Tools and Applications (2022) 81:34401–34416
https://doi.org/10.1007/s11042-022-12793-4

1168: DEEP PATTERN DISCOVERY FOR BIG MULTIMEDIA DATA

Two-step learning for crowdsourcing data classification

Hao Yu[1] · Jiaye Li[1,2] · Zhaojiang Wu[1] · Hang Xu[1] · Lei Zhu[3]

Received: 1 September 2020 / Revised: 29 December 2021 / Accepted: 24 February 2022 /
Published online: 9 July 2022

Abstract

Crowdsourcing learning (Bonald and Combes 2016; Dawid and Skene, J R Stat Soc: Series C (Appl Stat), 28(1):20–28 1979; Karger et al. 2011; Li et al, IEEE Trans Knowl Data Eng, 28(9):2296–2319 2016; Liu et al. 2012; Schlagwein and Bjorn-Andersen, J Assoc Inform Syst, 15(11):3 2014; Zhang et al. 2014) plays an increasingly important role in the era of big data (Liu et al., IEEE Trans Syst Man Cybern: Syst, 48(12): 451–2461, 2017; Zhang et al. 2014) due to its ability to easily solve large-scale data annotations (Musen et al., J Amer Med Informs Assoc, 22(6):1148–1152 2015). However, in the process of crowdsourcing learning, the uneven knowledge level of workers often leads to low accuracy of the label after marking, which brings difficulties to the subsequent processing (Edwards and Teddy 2013) and analysis of crowdsourcing data. In order to solve this problem, this paper proposes a two-step learning crowdsourced data classification algorithm, which optimizes the original label data by simultaneously considering the two issues of different worker abilities and the similarity between crowdsourced data (Kasikci et al. 2013) samples, so as to get more accurate label data. The two-step learning algorithm mainly includes two steps. Firstly, the worker's ability to label different samples is obtained by constructing and training the worker's ability model, and then the similarity between samples is calculated by the cosine measurement method (Muflikhah and Baharudin 2009), and finally the original label data is optimized by combining the above two results. The experimental results also show that the two-step learning classification algorithm proposed in this article has achieved better experimental results than the comparison algorithm.

Keywords Crowdsourcing learning · Similarity learning · Classification · Majority voting

✉ Lei Zhu
leizhu@hunau.edu.cn

Hao Yu
yuhooo@csu.edu.cn

[1] School of Computer Science and Technology of Central South University, Changsha 410083, People's Republic of China

[2] Guangxi Key Lab of Multi-Source Information Mining & Security, Guangxi Normal University, Guilin 541004, China

[3] College of Information and Intelligence, Hunan Agricultural University, Changsha 410128, People's Republic of China

1 Introduction

The rapid development of artificial intelligence [12, 17, 22, 32, 39] technology has brought a new source of power to the progress of human society. Artificial intelligence has also entered every corner of our daily lives. The fast-growing food delivery industry in recent years has solved work problems for nearly tens of millions people and provided convenience for hundreds of millions people to eat. At the same time, in the core architecture of the takeaway delivery system [23, 38], how to plan the delivery staff and takeaway orders, how to plan the optimal delivery route [2] for each delivery staff, and how to dynamically adjust the entire takeaway delivery system so that it will not be down during peak periods machine, any of the problems mentioned above cannot be separated from the deep intervention of artificial intelligence technology. At the beginning of 2020, a sudden epidemic affected the entire world. Due to its high contagiousness and concealment, the new coronavirus forced various [16, 29] countries and regions to gradually restrict population movement and block entry and exit gates. Due to the long incubation period of the new coronavirus, and the symptoms are very uncertain. Therefore, in the work of controlling the spread of the virus, how to investigate the epidemiology of the patients and populations that have been found and trace the source of the contact history has become the top priority of epidemic prevention and control. Technological workers use artificial intelligence technology base on big data to build a population flow model to scientifically judge the possibility of population flow as much as possible, and improve the efficiency of epidemic prevention and control. On the other hand, the development of the epidemic has also brought about changes in traditional industries. Due to the blockade measures introduced by governments of various countries, economic activities in many urban areas have fallen into a state of suspension. In contrast, the short video platform [19] Douyin achieved a 10-fold increase in monthly revenue during the epidemic, and became the world's most downloaded mobile program. The epidemic has caused many traditional industries to expand from offline to online. Selling goods on Douyin has become a transformation path for most companies. Surprisingly, a few hours of live broadcast may bring more revenue to merchants than previous months. The core of Douyin's successful marketing for businesses is to use artificial intelligence algorithms to accurately match a large number of users and intelligently allocate advertising traffic. Through these examples, it can be found that the application of artificial intelligence technology is inseparable from the support of big data. Big data is the cornerstone of the development of artificial intelligence technology. For now, the most popular technology in artificial intelligence should be deep learning technology [9, 27, 35]. The application of deep learning technology has greatly improved the effect of algorithms in various fields. The BP model [10] at the core of deep learning algorithms has actually been proposed as early as 1974. However, the rapid development of deep learning is in recent years. The main reason is that the amount of data that can be trained and processed at that time is too small, so it can not achieve better results. In recent years, due to the generation of large-scale data and the gradual increase in the ability to process large-scale data [37, 49], the development of deep learning technology has become possible. The classification of deep learning in the field of machine learning is supervised learning [3, 50], which mean each sample needs to have a clear label. At the same time, the magnitude of the training data has a great impact on the performance of the algorithm. Generally, the larger the number of samples with correct labels, the better the performance of the deep learning model.

However, in actual situations, the large-scale data we obtain are often unlabeled or incorrectly labeled. Therefore, how to correctly label these large-scale data has become a very

important research topic in the field of machine learning [40]. In order to label the results accurately, we can ask experts in specific fields to label the small-scale data. However, due to timeliness, economy and other factors, it is not practical to ask experts in large-scale data labeling tasks. To solve this problem, the crowdsourcing method [31, 47] distributes the labeling task to people from all walks of life all over the world through the network. Obtain large-scale data annotation results with lower economic cost and shorter time. While crowdsourcing methods bring the possibility of rapid labeling of large-scale data, there are also very obvious problems. The labelers are not professionals, and they may have very limited understanding of the characteristics of the labeled data. At the same time, the compensation provided by the crowdsourcing method is not high. Many labelers may not be serious about the labeling work, or they may be lazy to mark or simply not mark. These various reasons are very likely to cause the accuracy of the final sample label to be low.

In order to solve this problem, researchers have proposed some targeted algorithms [8, 33]. The majority voting algorithm is the most classic one. The main idea of the majority voting algorithm is very simple. Following the principle of minority obeys the majority, the category with the highest number of markings is selected as the predicted category of the sample from the marking results of all labelers. The principle of majority voting algorithm is clear, interpretable, and relatively easy to implement, so it has become a benchmark algorithm for solving crowdsourced data problems. However, the majority voting algorithm has some flaws. Because crowdsourced data is artificial data produced in a short time by different people from all walks of life around the world, these people have different identities, backgrounds, education levels, and different ideas. Therefore, the marking accuracy of each marker must be different, but in the majority voting algorithm idea, it is simply considered that the abilities of each marker are the same, so the final results often have certain defects.

This paper proposes a new two-step learning crowdsourced data classification algorithm for the problems of majority voting algorithm. In the first step, a worker ability model is proposed for the different labeling abilities of different labeled workers. The model first adds the initial ability weights to all workers, and obtains the labeled ability weights of different workers through fitting samples and self-expression reconstruction samples after adding worker ability weights. At the same time, the L12 norm [11] of the worker ability weight matrix is added to the objective function. The L12 norm is an optimized version of the lasso method [30, 48], which considers the similarity between attributes based on the sparseness of the variables in the Lasso method. The specific implementation is to set one of the weights of two workers with similar marking abilities to 0 to reduce redundancy, so that the difference in the abilities of different markers is further reflected, and then a more accurate weight of worker abilities can be obtained. The second step considers the similarity between different samples through the cosine measurement method [36, 41], specifically by calculating the sample similarity between the two, find the most similar sample for each sample. Finally, the specific category of the sample is calculated by combining the worker ability weight obtained in the first step and the similar sample weight obtained in the second step. The main contributions of the algorithm proposed in this paper are as follows:

– The traditional MV algorithm simply assumes that all marking workers have the same ability, but in actual scenarios, this assumption is often unscientific, which leads to poorer final algorithm results. For this problem, the algorithm proposed in this paper assigns different ability weights to all workers based on the MV algorithm, making the majority voting process more reasonable, and the voting results are naturally more accurate.

- The algorithm proposed in this paper adds an l_{12} regularization term to the model for training worker ability weights. l_{12} regularization is a group sparse method, which groups similar samples in attributes and makes the group sparse through restriction methods , The groups become non-sparse. The l_{12} regularization term is applied to the weight of the worker's ability in this paper so that multiple workers with similar abilities only retain the weight of one of them, which sparses the weight of the entire worker's ability, so that the redundant influence of workers with similar abilities on the entire weight is reduced. It distinguishes the different weights corresponding to workers with different abilities, and finally gets better results.
- The algorithm proposed in this paper considers the uneven abilities of marking workers, but also gives weight to the samples. The real data collected in various industries is not randomly generated, but usually has certain industry characteristics. These sample data are not messy, but have some similar or contradictory characteristics. Therefore, this paper uses the cosine measurement method to calculate the similarity between samples, and combines the worker ability weights obtained from the worker ability model training to establish the final classification model and test the data. Therefore, this paper considers the two key factors of worker ability weight and data sample similarity at the same time, so that the algorithm proposed in this paper has better performance.

2 Related work

In this section, we first introduce some methods of similarity measurement in first part, and then introduce the random forest algorithm in the second part.

2.1 Similarity measure

The basic processing of data by machine learning algorithms is generally classification or clustering, *i.e.,* to separate different data or aggregate the same data together [44]. How to judge whether two data are the same or different is very important. Therefore, data similarity measurement is a very important link in machine learning algorithms. How to measure the difference between samples scientifically and how to choose the correct measurement method for data characteristics are the key factors to improve the performance of the algorithm. Commonly used distance measurement methods include Euclidean distance [5], Manhattan distance, Minkowski distance, Hamming distance, cosine distance, etc. Euclidean distance refers to the actual distance between two points. Because of its more obvious difference in high-dimensional data, it is suitable for measuring the similarity of samples in high-dimensional data. The meaning of Manhattan distance [4] is very intuitive, that is, the distance between Manhattan blocks. Since there are often many intersections in the block, the actual driving distance in the block is the Manhattan distance. Minkowski distance [24] is a distance formula containing variables. When the variable is 1, it is Manhattan distance. When the variable value is 2. It is Euclidean distance. Hamming distance [28] mainly operates on character strings. The number of characters that need to be changed to convert character string 1 to character string 2 is used as the distance between the two character strings, which is the Hamming distance. Therefore, the Hamming distance is very suitable for the fields of password and information compression. The law of cosines in geometry can use the cosine of the angle to measure the difference between two vector directions, and the cosine measurement method of machine learning uses this method to measure the difference between samples. The cosine measurement method first needs to

normalize the sample, so the cosine measurement is not sensitive to the length of the sample, but only sensitive to the direction of the sample. Therefore, the cosine measurement method is usually suitable for the similarity discrimination of high-dimensional samples, but not for specific distance calculations.

2.2 Random forest

Obviously, the random forest algorithm [18] mainly includes two parts, one is random and the other is forest. The forest in the random forest algorithm is composed of decision trees [42, 43, 45], and the way of composition is random. Decision tree is a supervised machine learning algorithm. It starts to split from the root node containing all samples, and splits into split nodes one by one until the last layer is all leaf nodes. Each split node represents a split condition. The samples in the previous node are classified into two or more categories through this condition. After the split is completed, each leaf node represents a specific category. To a certain extent, the decision tree algorithm solves the shortcoming that some algorithms can only perform linear segmentation of data. The sample data is classified more accurately through the tree structure of the algorithm. At the same time, it has strong interpretability and the algorithm process is intuitive. Easy to understand. The random forest algorithm first randomly samples samples and attributes, then completely splits the obtained data to obtain a decision tree, and then repeats these steps to finally obtain a random forest. Among them, the sample sampling method is random sampling with replacement. Therefore, the samples collected during the establishment of each tree are not necessarily the same, so that the model of each decision tree is not easy to overfit and can obtain the intermodel Some deviation information. Random forest is widely used in forecasting systems, big data modeling, etc. due to its strong robustness, suitable for processing high-dimensional data, simple implementation and excellent performance.

3 Methodology

In this Section, we first introduce some character definitions used in this article in Section 3.1, then introduce the MV algorithm and explain the specific mathematical representation of the MV algorithm in Section 3.2, and then introduce the Knv algorithm and its Specific process in Section 3.3. In Section 3.4, we introduce the two-step learning crowdsourcing data classification algorithm in detail. Finally, the objective function of the algorithm is optimized in Section 3.5.

3.1 Notations

In this paper, we use uppercase and lowercase letters to represent matrices and vectors, respectively. $\mathbf{X} = \{x_i\}_{i=1}^{n} \in \mathbb{R}^{n\times d}$ represents the sample set, where n and d represent the number of samples and the dimension of each sample, respectively.$\mathbf{X}^i$ represents the i-th row of the matrix $\mathbf{X}$, which is the i-th sample of the sample set.$\mathbf{X}_j$ represents the j-th column of matrix $\mathbf{X}$, which is the j-th attribute of all samples in the sample set. $\mathbf{X}_{i,j}$ represent the j-th attribute of the i-th sample in the sample set. The Frobenius norm of $\mathbf{X}$ is denoted as $||\mathbf{X}||_F = \sqrt{\sum_{ij} |x_{ij}|^2}$. Furthermore, The trace, inverse, and transpose of matrix $\mathbf{X}$ are represented as:$tr(\mathbf{X})$, $\mathbf{X}^{-1}$ and $\mathbf{X}^T$, respectively.We also summarize these notations in Table 1 .

Table 1 The detail of the notations used in this paper

$\mathbf{X}$	a matrix
$\mathbf{x}$	a vector of $\mathbf{X}$
$\mathbf{x}^i$	the i-th row of $\mathbf{X}$
$\mathbf{x}_j$	the j-th column of $\mathbf{X}$
$x_{i,j}$	the element in the i-th row and the j-th column of $\mathbf{X}$
$\|\|\mathbf{X}\|\|_F$	the Frobenius norm of $\mathbf{X}$, *i.e.,* $\|\|\mathbf{X}\|\|_F = \sqrt{\sum_{i,j} \mathbf{x}_{i,j}^2}$
$\mathbf{X}^T$	the transpose of $\mathbf{X}$
$tr(\mathbf{X})$	the trace of $\mathbf{X}$
$\mathbf{X}^{-1}$	the inverse of $\mathbf{X}$

3.2 Majority voting

Table 2 is a crowdsourcing data set. The abscissa represents the sample, while the ordinate represents the worker. The coordinate$L(x, y)$ represents the marking of the y-th worker to the x-th sample. In practice, due to the ability of a single marker worker to understand some problems is relatively limited, the accuracy of labels is not high if only one worker's tagging results are used. In order to solve this problem, the MV method proposes to analyze the labeled tags by the principle of the minority obeying the majority, and uses the result with the highest frequency as the final classification label of the sample. The specific expression is as follows:

$$v(x) = \arg \max_{b \in \Omega} v(b|x) \tag{1}$$

where $v(b|x) = \frac{1}{|S_x|} \sum_{w \in S_x} \mathbf{1}(w = b)$, $|S_x|$ represents the number of workers, S_x represents all the labeling conditions of the x-th sample, and Ω represents the tag collection. The **1.** function analyzes the comparison between all the tag results of the sample and the real tag. If the tag result is correct, return 1. Otherwise it is 0. Therefore, we can find the case of two class label data set , where $v > 0.5$, it shows that MV algorithm gets the real label. Compared with the traditional single label method, the majority voting method can get a more accurate labeling result. In the calculation process of MV algorithm, we give each worker the same weight, but in the actual situation, each worker's ability to understand different problems is often very different. Therefore, it is not optimal to assign the same weight to each worker, which means that there are still some defects in the MV algorithm.

Table 2 Crowdsourcing data set

Samples	x_1	x_2	x_3	x_4	x_5	...,	x_n
y_1	L_{11}	L_{21}	L_{31}	L_{41}	L_{51}	...,	L_{n1}
y_2	L_{12}	L_{22}	L_{32}	L_{42}	L_{52}	...,	L_{n2}
y_3	L_{13}	L_{23}	L_{33}	L_{43}	L_{53}	...,	L_{n3}
...,	...,	...,	...,	...,	...,	...,	...,
y_m	L_{1m}	L_{2m}	L_{3m}	L_{4m}	L_{5m}	...,	L_{nm}

3.3 K nearest voting

In order to solve some defects of MV algorithm, this chapter introduces an improved algorithm Knv method based on MV. Knv method refers to k-nearest neighbor voting algorithm, and its specific mathematical expression is as follows:

$$v_k(x) = \arg \max_{b \in \Omega} v_k(b|x) \tag{2}$$

where $v_k(b|x) = \frac{1}{|S_x|+\bar{\alpha}} \left[|S_x| \, v(b|x) + \alpha_b^x \right]$, $\alpha_b^x = \frac{1}{k} \sum_{i=1}^{k} \alpha_i v(b|x_i)$, $x_i \in N_{K(x)}$ The vector α represents the weight of k nearest neighbors of the sample. In order to reflect the discrimination degree of k samples, initialization $\alpha = [k, k-1, k-2, ..., k]$, $\bar{\alpha}$ is used to represent the mean value of the elements in the vector. The MV algorithm just votes the labeled results of all the workers to get the final algorithm result without considering the relationship between adjacent samples. In practice, the samples with similar distance tend to have similar characteristics. These samples with similar characteristics often have the same category, so it is very necessary and scientific to consider the labeling of adjacent samples when judging the real label of a single sample. In order to solve this problem, Knv algorithm considers the labeling of k nearest neighbors of samples base on the original MV algorithm, so as to judge the real label of the sample more scientifically and accurately. At the same time, the experiment also proves that the Knv algorithm achieves better performance. Although the Knv algorithm improves the performance of MV algorithm to a certain extent by considering the influence of k nearest neighbor sample labeling. However, the Knv algorithm still does not consider the factors of different tagging workers' ability to label different samples, so the Knv algorithm still can be improved by considering the factors of marker workers' understanding of the samples.

3.4 Proposed method

In view of the shortcomings of MV and Knv algorithms introduced above, this paper proposes a two-step learning crowdsourcing data classification algorithm based on the traditional MV algorithm. The first step is to build a worker's tagging ability model . Specifically, firstly assign a labeling ability weight matrix β to all workers, and reconstruct the sample by fitting the original sample and adding the self-expression of the worker labeling ability weight matrix β to obtain the optimal worker labeling ability weight matrix β. The specific expression is as follows:

$$\min_{\beta} \left\| \mathbf{X}^T - \mathbf{X}^T \mathbf{Y}^T \beta \right\|_F^2 \tag{3}$$

Where $\mathbf{X} \in \mathbb{R}^{n \times d}$ represents the original crowdsourcing dataset contains n d-dimensional samples, $\mathbf{Y} \in \mathbb{R}^{m \times n}$ represents all the labeling results of m workers on n samples in the data set,and the matrix $\beta \in \mathbb{R}^{m \times n}$ represents the labeling ability weight of m workers for n samples in the crowdsourcing data set. In view of the fact that there are some workers with similar marking ability, there is a certain degree of redundancy in the marking ability weight matrix β. In order to solve this defect, our method creatively adds the l_{12} norm as the regularization term of the objective function base on (3), and sparses the ability weight matrix β , so that the redundancy of workers with similar abilities is reduced, and workers with different marking abilities are further distinguished. So the performance of the algorithm is better. The l_{12} norm is a group Lasso method. The core idea of the l_{12}

norm is to group all samples so that the samples in the same group become sparse and the groups become as close as possible. The specific expression is as follows:

$$\forall W \in \mathbb{R}^{d\times 1}, \Omega_g^G(W) = \sum_{g\in G} \left\| W_{Gg} \right\|_1^2 \tag{4}$$

Among them, W represents the $1-d$ attributes, G represents the set of all groups, G represents one of the groups, l_1-norm is used to make the attributes within the group more sparse, and the l_2- norm makes the groups not sparse. Because we know that samples in the same group often have great similarities, making them sparse can remove redundant samples in some groups, and samples in different groups are often different, so we need to keep these useful samples. Therefore, after adding the l_{12} norm, we get the final objective function of the worker ability model, the expression is as follows:

$$\min_{\beta} \left\| \mathbf{X}^T - \mathbf{X}^T \mathbf{Y}^T \beta \right\|_F^2 + \lambda \left\| \beta_g \right\|_1^2 \tag{5}$$

The parameters λ are used to adjust the l_{12} regularization term. The main idea of the worker ability model is to reconstruct the original sample, and obtain the optimal ability weight matrix β by fitting and reconstructing the sample. At the same time, the group Lasso regularization term is added. We perform sparse and non-sparse operations on the worker ability weight matrix at the same time, so that while removing redundant workers, it retains useful worker information as much as possible.

Unlike the first step, which considers the weight of the worker's marking ability, the second part of the algorithm considers the similarity between samples. In crowdsourced data, in addition to the similar or opposite relationship between different labeled workers, there is also a certain similar relationship between samples. Similar samples usually have similar labels, so the labels of similar samples also have certain reference significance for the current sample. Therefore, we use this idea to calculate the label for a single sample while considering the labeling of similar samples, which can reduce the adverse effects of random errors by a small number of labeling workers to a certain extent. This paper uses the cosine measurement method to measure the similarity between samples. The mathematical expression is as follows:

$$\alpha = similarity = \cos(\theta) = \frac{\mathbf{AB}}{\|\mathbf{A}\| \, \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} \mathbf{A}_i \times \mathbf{B}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{A}_i)^2} \times \sqrt{\sum_{i=1}^{n} (\mathbf{B}_i)^2}} \tag{6}$$

Where$\mathbf{A}$ and $\mathbf{B}$ represent the two samples in the dataset, $\mathbf{A}_i$ and $\mathbf{B}_i$ represent the i-th attribute value of the samples respectively. Specifically, the above cosine similarity formula is used to calculate all samples, all n samples are traversed, and the nearest neighbor sample that is most similar to each sample is calculated (when n is an odd number, the last sample is kept without calculation). Then we get $n/2$ pairs and $n/2$ similarity values α ($\alpha = similarity$).

Finally, the worker's labeling ability weight matrix β obtained in the first step is combined with the $n/2$ similarity values α obtained in the second step to calculate the final predicted label . Specifically for each sample, the ability weight matrix β is first applied to different markers, and on this basis, the sample weight is assigned 1-0.5 * α , and the similar sample weight of the sample is assigned 0.5 * α for calculation, and finally the predicted label of the sample is obtained . Since this algorithm also considers two key factors, the

similarity between the crowdsourced data samples and the difference in the marking ability of different workers, the predicted label obtained by this algorithm is greatly improved in accuracy compared with the original label.

3.5 Optimization

Because the objective function of the first part of the algorithm can not be solved directly, this paper optimizes the objective function in this section, and the specific steps are as follows. And we also list the pseudo code in Algorithm 1.

$$\min_{\beta} \left\| \mathbf{X}^T - \mathbf{X}^T \mathbf{Y}^T \beta \right\|_F^2 + \lambda \left\| \beta_g \right\|_1^2 \tag{7}$$

Where $\mathbf{X} \in \mathbb{R}^{n \times d}$, $\mathbf{Y} \in \mathbb{R}^{m \times n}$, $\beta \in \mathbb{R}^{m \times n}$. First of all, the above formula is expanded and the results are as follows:

$$\min_{\beta} \left\| \mathbf{X}^T - \mathbf{X}^T \mathbf{Y}^T \beta \right\|_F^2 + \lambda tr(\beta^T \mathbf{F} \beta) \tag{8}$$

Then, the derivation of the above formula can be obtained as follows:

$$-2\mathbf{Y}\mathbf{X}\mathbf{X}^T + 2\mathbf{Y}\mathbf{X}\mathbf{X}^T\mathbf{X}^T\beta + 2\lambda\mathbf{F}\beta \tag{9}$$

We make the above formula equal to 0, the final weight matrix β of worker's ability can be obtained:

$$\beta = (\mathbf{Y}\mathbf{X}\mathbf{X}^T\mathbf{Y}^T + \lambda\mathbf{F})^{-1}\mathbf{Y}\mathbf{X}\mathbf{X}^T \tag{10}$$

Where $\mathbf{F}$ of above formula is a diagonal matrix . Its diagonal elements are:

$$\mathbf{F}_{ii} = \sum_g \frac{(IG_g)_i \left\| \beta_g \right\|_1}{\left\| \beta^i \right\|_1} (i = 1, \cdots, m) \tag{11}$$

Algorithm 1 The pseudo code of crowdsourcing Data Classification.

Input: $\mathbf{X} \in \mathbb{R}^{n \times d}$, $\mathbf{Y} \in \mathbb{R}^{m \times n}$, λ;

Output: $\beta \in \mathbb{R}^{m \times n}$;

1. Initialize $\beta \in \mathbb{R}^{m \times n}$;

2. **repeat:**

2.1 Calculate F_{ii} by (11);

2.2 Update β by (10);

until (7) convergence

Table 3 The summarization of the used data sets

Data Sets	#(Samples)	#(Dimensions)	#(Classes)
Clean	476	167	2
German	1000	20	2
Parkinsons	195	22	2
Sonar	208	60	2
Contro	600	60	6
Drift	1244	129	6
Ecoli	336	343	8
CCUDS	1994	101	10
Movements	360	90	15
Soybean	307	35	19

4 Experiments

In this Section, the algorithm proposed in this paper and comparison method will be tested on 10 data to compare their data classification ability. Specifically, the crowdsourcing dataset used in this paper and how to set the parameters needed in the experiment are introduced in Section 4.1. In Section 4.2, the sensitivity of the parameters to the experimental effect is analyzed, so that we can find the appropriate parameters. Finally, the specific experimental results are analyzed also in the last Section 4.2.

4.1 Data set and parameter settings

In this experiment, 10 data sets of UCI data set are used. They are Clean,German, Parkinsons,Sonar,Contro,Drift,Ecoli,CCUDS,Movements and Soybean.The detail of these data sets is listed in Table 3 . In the experiment process of this article, we set three parameters. The first is the average number of of markers $\left|\overline{S_x}\right|$. The setting of the average number of markers takes into account the different number of markers in each sample, and simulates the real by setting the average number of markers. The second is the parameter con of the beta distribution. In this paper, a simulated crowdsourced data set is constructed on the basis of the original data through the principle of beta distribution. Since the number of tags for each sample is not necessarily the same, it simulates the real crowdsourced data well. The third is the reliability parameter rel. This parameter represents the average labeling ability of all workers, which represents the label accuracy of the original data.In the specific experiment, this paper fixed the average number of markers $\left|\overline{S_x}\right| = 25$ and the beta distribution parameter $con = 1$, and set the reliability parameters as 0.6, 0.7 and 0.8.

The first step of the experiment is to create the crowdsourced data set required for this article on the basis of the original data set. The specific steps are as follows: The first step

Table 4 Average Classification accuracy(rel=0.8)

Datasets	Clean	German	Parkinsons	Sonar	Contro	Drift	Ecoli	CCUDS	Movements	Soybean
MV	97.27	98.31	97.95	96.63	87.67	93.65	86.33	82.32	81.22	71.78
Proposed	**99.27**	**99.50**	**99.49**	**98.56**	**95.33**	**93.99**	**98.51**	**99.60**	**89.44**	**89.58**

Table 5 Average Classification accuracy(rel=0.7)

Datasets	Clean	German	Parkinsons	Sonar	Contro	Drift	Ecoli	CCUDS	Movements	Soybean
MV	94.54	96.70	96.49	96.15	83.17	90.51	83.33	83.14	75.56	69.06
Proposed	**98.95**	**99.50**	**98.46**	**98.08**	**96.50**	**91.72**	**96.13**	**99.61**	**88.06**	**86.32**

is to classify the data through the random forest algorithm in related work, and then use the labels obtained by the classification. And the true label generation matrix M of the data. The second step is to construct the R matrix after the M matrix is generated. Then build the marking of crowdsourced data based on the R matrix.

4.2 Experimental result

In this section, we choose the average number of markers $|\overline{S_x}|$=25, and set the reliability parameters rel = 0.6, 0.7, 0.8, respectively. Simulate the performance of the algorithm in this paper under the environment of different raw data quality, and compare it with the traditional MV algorithm. The experiment selected 4 two-classification data sets and 6 multi-classification data sets, which can test the performance of this algorithm on simple two-classification problems and complex multi-classification problems at the same time. From Tables 4, 5, and 6, we can find that this algorithm has achieved very good results on 10 data sets, and the accuracy of the algorithm is higher than that of MV. Respectively, when the reliability parameterrel =0.8, the algorithm is 7.01% higher than the MV algorithm, when the reliability parameter rel=0.7, the algorithm is 8.47% higher than the MV algorithm, and when the reliability parameter rel =0.6, the algorithm It is 12.36% higher than the MV algorithm. The traditional MV algorithm only does a large number of statistics for all the labeling results of all workers, and selects the category with the highest occurrence probability as the final prediction label of the MV method. The method I put forward first considers the differences in marking abilities of different workers, and assigns different weights to different marked workers. At the same time, the data samples are analyzed, considering that similar samples may have similar labels, combining the above two key points, so the algorithm in this paper has achieved better performance than the traditional MV algorithm.

By analyzing the three tables at the same time, it can also be found that as the accuracy of the original crowdsourced data decreases, the accuracy of the proposed algorithm in this paper decreases more slowly than the accuracy of the traditional MV algorithm, indicating that the quality of the original data of the algorithm in this paper is not good. At the same time, it can still achieve better results and has strong robustness. Finally, in Table 7 we find that the classification accuracy of the algorithm is usually better than the traditional MV algorithm in terms of stability.

Table 6 Average Classification accuracy(rel=0.6)

Datasets	Clean	German	Parkinsons	Sonar	Contro	Drift	Ecoli	CCUDS	Movements	Soybean
MV	79.20	94.20	92.31	94.71	80.50	84.24	80.36	81.44	71.11	62.87
Proposed	**97.48**	**99.50**	**97.95**	**97.60**	**95.83**	**92.12**	**94.35**	**98.81**	**87.50**	**83.39**

Table 7 Standard deviation of Classification accuracy(rel=0.6)

Datasets	Clean	German	Parkinsons	Sonar	Contro	Drift	Ecoli	CCUDS	Movements	Soybean
MV	6.10	0.64	1.54	0.97	3.74	**2.98**	1.90	0.92	2.87	**2.93**
Proposed	**6.10**	**0.16**	**0.65**	**0.90**	**2.02**	3.54	**0.94**	**0.13**	**2.38**	3.35

5 Conclusion

This paper proposes a new two-step learning crowdsourced data classification algorithm. First of all, by assigning different labeling weights to each worker, the negative impact of different workers' abilities in the process of crowdsourced data labeling is reduced to a certain extent. At the same time, the similarity between samples in the crowdsourced data is analyzed by the cosine measurement method, and the most similar samples are found and weighted. Therefore, the algorithm proposed in this paper takes into account the two key factors of the difference in the ability of workers and the similarity between the data samples, and reclassifies the original crowdsourced data. It has achieved higher accuracy rate than the traditional MV algorithm on 10 data sets. On this basis, we added three sets of comparative experiments on raw data with different quality.It was found that the proposed algorithm achieved good performance in the stability of experimental results, and it was less affected by the quality of the original data, indicating that the proposed algorithm has highly accurate and good robustness at the same time.

In the future work, we will further consider how to use the similarity between samples to improve the performance of classification algorithm.

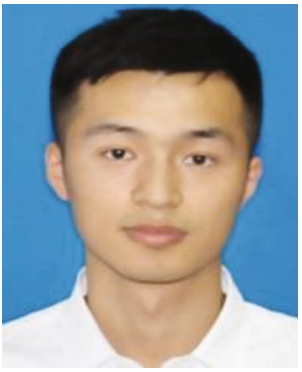
Acknowledgments This work was supported in part by the Key Program of the National Natural Science Foundation of China (Grant No: 61836016), Fundamental Research Funds for the Central Universities (2021zzts0209), the Natural Science Foundation of China (Grants No: 61876046, 61573270, 81701780 and 61672177), Research Fund of Guangxi Key Lab of Multi-source Information Mining & Security (MIMS20-04).

References

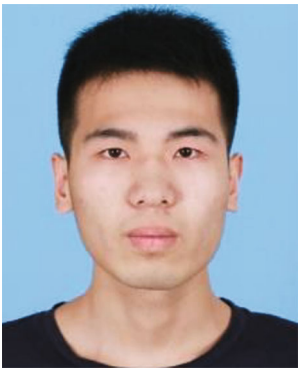
1. Bonald T, Combes R (2016) A minimax optimal algorithm for crowdsourcing
2. Bornstein CF, Canfield TK, Miller GL, Rao SB, Sundaram R (2011) Optimal route selection in a content delivery network. US Patent 7,929,429
3. Caruana R, Niculescu-Mizil A (2006) An empirical comparison of supervised learning algorithms. In: Proceedings of the 23rd international conference on Machine learning, pp 161–168
4. Chang D-J, Desoky AH, Ouyang M, Rouchka EC (2009) Compute pairwise manhattan distance and pearson correlation coefficient of data points with gpu. In: 2009 10th ACIS International conference on software engineering, artificial intelligences, networking and parallel/distributed computing, pp 501–506. IEEE
5. Danielsson P-E (1980) Euclidean distance mapping. Comput Graph Image Process 14(3):227–248
6. Dawid AP, Skene AM (1979) Maximum likelihood estimation of observer error-rates using the em algorithm. J R Stat Soc: Series C (Appl Stat) 28(1):20–28
7. Edwards JL, Teddy JD (2013) Subsequent processing of scanning task utilizing subset of virtual machines predetermined to have scanner process and adjusting amount of subsequest vms processing based on load. US Patent 8,516,478
8. Felsenthal DS, Machover M (2001) The treaty of nice and qualified majority voting. Soc Choice Welf 18(3):431–464
9. Goodfellow I, Bengio Y, Courville A, Bengio Y (2016) Deep learning, vol 1. MIT press Cambridge

10. Hecht-Nielsen R (1992) Theory of the backpropagation neural network. In: Neural networks for perception, pp 65–93. Elsevier
11. Jacob L, Obozinski G, Vert J-P (2009) Group lasso with overlap and graph lasso. In: Proceedings of the 26th annual international conference on machine learning, pp 433–440
12. Johnson KW, Torres Soto J, Glicksberg BS, Shameer K, Miotto R, Ali M, Ashley E, Dudley JT (2018) Artificial intelligence in cardiology. J Am Coll Cardiol 71(23):2668–2679
13. Karger DR, Oh S, Shah D (2011) Iterative learning for reliable crowdsourcing systems. In: Advances in neural information processing systems, pp 1953–1961
14. Kasikci B, Zamfir C, Candea G (2013) Racemob: crowdsourced data race detection. In: Proceedings of the twenty-fourth ACM symposium on operating systems principles, pp 406–422
15. Li G, Wang J, Zheng Y, Franklin MJ (2016) Crowdsourced data management: a survey. IEEE Trans Knowl Data Eng 28(9):2296–2319
16. Li G, Fan Y, Lai Y, Han T, Li Z, Zhou P, Pan P, Wang W, Hu D, Liu X et al (2020) Coronavirus infections and immune responses. J Med Virol 92(4):424–432
17. Li Y, Lei C, Fang Y, Hu R, Li Y, Zhang S (2018) Unsupervised feature selection by combining subspace learning with feature self-representation. Pattern Recognit Lett 109:35–43
18. Liaw A, Wiener M et al (2002) Classification and regression by randomforest. R news 2(3):18–22
19. Link PJ (2012) Hand-held video game platform emulation. US Patent 8,157, 654
20. Liu C, Wang Y-M (2012) Truelabel+ confusions: a spectrum of probabilistic models in analyzing multiple ratings. arXiv:1206.4606
21. Liu H, Li X, Li J, Zhang S (2017) Efficient outlier detection for high-dimensional data. IEEE Trans Syst Man Cybern: Syst 48(12):2451–2461
22. Lu H, Li Y, Chen M, Kim H, Serikawa S (2018) Brain intelligence: go beyond artificial intelligence. Mob Netw Applic 23(2):368–375
23. Martucci J, Bui T, Hitchcock J, DiGianfilippo A, Pierce R (2006) Medication delivery system. US Patent 6,985,870
24. Merigó JM, Casanovas M (2011) A new Minkowski distance based on induced aggregation operators. Int J Comput Intell Syst 4(2):123–133
25. Muflikhah L, Baharudin B (2009) Document clustering using concept space and cosine similarity measurement. In: 2009 International conference on computer technology and development, vol 1, pp 58–62. IEEE
26. Musen MA, Bean CA, Cheung K-H, Dumontier M, Durante KA, Gevaert O, Gonzalez-Beltran A, Khatri P, Kleinstein SH, O'Connor MJ et al (2015) The center for expanded data annotation and retrieval. J Am Med Inform Assoc 22(6):1148–1152
27. Ngiam J, Khosla A, Kim M, Nam J, Lee H, Ng AY (2011) Multimodal deep learning. In: ICML
28. Norouzi M, Fleet DJ, Salakhutdinov RR (2012) Hamming distance metric learning. In: Advances in neural information processing systems, pp 1061–1069
29. Novel Coronavirus Pneumonia Emergency Response Epidemiology et al (2020) The epidemiological characteristics of an outbreak of 2019 novel coronavirus diseases (covid-19) in china, vol 41
30. Park T, Casella G (2008) The bayesian lasso. J Am Stat Assoc 103(482):681–686
31. Poblet M, García-Cuesta E, Casanovas P (2018) Crowdsourcing roles, methods and tools for data-intensive disaster management. Inf Syst Front 20(6):1363–1379
32. Russell S, Norvig P (2002) Artificial intelligence: a modern approach
33. Ruta D, Gabrys B (2005) Classifier selection for majority voting. Inform Fus 6(1):63–81
34. Schlagwein D, Bjorn-Andersen N (2014) Organizational learning with crowdsourcing: the revelatory case of lego. J Assoc Inf Syst 15(11):3
35. Schmidhuber J (2015) Deep learning in neural networks: an overview. Neur Netw 61:85–117
36. Sohangir S, Wang D (2017) Improved sqrt-cosine similarity measurement. J Big Data 4(1):25
37. Vasudevan S, Ramos F, Nettleton E, Durrant-Whyte H (2009) Gaussian process modeling of large-scale terrain. J Field Robot 26(10):812–840
38. Wei G, Behnam K, Forsyth N, Winterbottom J, Beisser J, Boyce TM, Attawia M, Richards CJ, Shimp LA (2016) Delivery system attachment. US Patent 9,333,082
39. Xu C, Sun J, Wang C (2019) A novel image encryption algorithm based on bit-plane matrix rotation and hyper chaotic systems. Multimed Tools Appl, 1–21
40. Zeng Z, Bao H, Wen Z, Zhu W (2019) Object tracking using the particle filter optimised by the improved artificial fish swarm algorithm. Int J Intell Inf Database Syst 12(1/2):6–19
41. Zhang H, Song S, Zhou A, Gao XZ (2015) A multiobjective cellular genetic algorithm based on 3d structure and cosine crowding measurement. Int J Mach Learn Cybern 6(3):487–500
42. Zhang S (2012) Decision tree classifiers sensitive to heterogeneous costs. J Syst Softw 85(4):771–779
43. Zhang S (2018) Multiple-scale cost sensitive decision tree learning. World Wide Web 21(6):1787–1800

44. Zhang S, Li J (2021) Knn classification with one-step computation. IEEE Trans Knowl Data Eng, 1–1
45. Zhang S, Qin Z, Ling CX, Sheng S (2005) "missing is useful": missing values in cost-sensitive decision trees. IEEE Trans Knowl Data Eng 17(12):1689–1693
46. Zhang S, Zhang C, Yu JX (2004) Mining dependent patterns in probabilistic databases. Cybern Syst: Int J 35(4):399–424
47. Zhang Y, Chen X, Zhou D, Jordan MI (2014) Spectral methods meet em: a provably optimal algorithm for crowdsourcing. In: Advances in neural information processing systems, pp 1260–1268
48. P Zhao BYu (2006) On model selection consistency of lasso. J Mach Learn Res 7:2541–2563
49. Zhu J, Ge Z, Song Z (2017) Distributed parallel pca for modeling and monitoring of large-scale plant-wide processes with big data. IEEE Trans Industr Inform 13(4):1877–1885
50. Zhu X, Goldberg AB (2009) Introduction to semi-supervised learning. Synth Lect Artif Intell Mach Learn 3(1):1–130



Hao Yu was born in Shang Rao, China, on December 23, 1994. He received the M.Sc. degree from Guangxi Normal University, China, in 2018. He is currently a Ph.D. candidate in computer science and technology of Central South University. His research interests are in the field of image retrieval, machine learning, computer vision and Crowdsourcing learning.



Jiaye Li is a PhD candidate in computer science and technology of Central South University, Changsha 410083, PR China. His research interests are machine learning, data mining and deep learning.

Zhaojiang Wu was born in An Qing, China, on February 5, 1994. He received his bachelor's degree from Hefei University of Technology, China, in 2016. He is now a postgraduate student in computer science and technology of Central South University. His research interests are in the field of machine learning, cross-modal retrieval and multi-modal learning.

Hang Xu was born in Chang Sha, China, on December 19, 1996. She is currently a M.D. candidate in computer science and technology of Central South University. Her research interests are in the field of machine learning.

Lei Zhu was born in Changsha, China, on June 7, 1988. He received the M.Sc. degree from Central South University, China, in 2014. He is currently a Ph.D. candidate in computer science and technology of Central South University. His research interests are in the field of machine learning, deep learning, computer vision and spatio-temporal data retrieval.