Knowledge-Based Systems 107 (2016) 96–103

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Noise filtering to improve data and model quality for crowdsourcing

Chaoqun Li [a], Victor S. Sheng [b], Liangxiao Jiang [c,*], Hongwei Li [a,*]

[a] Department of Mathematics, China University of Geosciences, Wuhan 430074, China
[b] Department of Computer Science, University of Central Arkansas, Conway, AR, USA
[c] Department of Computer Science, China University of Geosciences, Wuhan 430074, China

ARTICLE INFO

Article history:
Received 18 March 2016
Revised 31 May 2016
Accepted 1 June 2016
Available online 2 June 2016

Keywords:
Crowdsourcing learning
Integrated labels
Label noise
Noise filtering

ABSTRACT

Crowdsourcing services provide an easy means of acquiring labeled training data for supervised learning. However, the labels provided by a single crowd worker are often unreliable. Repeated labeling can be used to solve this problem. After multiple labels have been acquired by repeated labeling for each instance, in general consensus methods are used to obtain the integrated labels of instances. Although consensus methods are effective in practice, it cannot be denied that a level of noise still exists in the set of integrated labels. In this study, an attempt was made to employ noise filters to delete the noise in integrated labels, and consequently, enhance the training data and model quality. In fact, noise handling is a relatively mature field in the machine learning community, and many noise filters for deleting label noise have been presented in the past. However, to the best of our knowledge, in very few studies was noise filtering used to improve crowdsourcing learning. Therefore, in this study we empirically investigated the performance of noise filters in terms of improving crowdsourcing learning. Thus, in this paper some existing noise filters presented in previous papers are reviewed and their experimental application to crowdsourcing learning tasks is described. Experimental results based on 14 benchmark UCI data sets and three real-world data sets show that these noise filters can significantly reduce the noise level in integrated labels and thereby considerably enhance the performance of target classifiers.

1. Introduction

In supervised learning, a training instance is always denoted by a d-dimensional feature vector and a known label. The traditional technique for acquiring the known label from domain experts is expensive and time-consuming in many cases. With the development of the Internet techniques, crowdsourcing services, such as Amazon Mechanical Turk, have become an efficient and low cost means of obtaining a great quantity of labeled data for supervised learning. However, some factors, such as the low payment of crowd workers and their limited abilities, lead to the labels provided by a single crowd worker frequently being unreliable. To solve this problem, multiple labels are frequently requested from different crowd workers for an instance, that is, repeated labeling is performed. [25] proved that, when a single worker's labeling is not perfect, repeated labeling is an effective approach to improving the quality of integrated labels. After acquiring multiple labels of an instance by repeated labeling, consensus methods, such as Majority Voting (MV), RY [21], and ZenCrowd (ZC) [4], can be used to infer a training label (an integrated label, i.e., a best consensus label) of the instance. A relatively comprehensive review of consensus methods can be found in [12,26].

However, it cannot be denied that some level of noise exists in a set of integrated labels inferred by consensus methods. Here, noise refers to the instances, the integrated labels of which are different from their true labels, i.e., the labels given by domain experts. For example, after multiple labels of an instance are acquired from different crowd workers, MV infers the integrated label by majority voting. MV is the simplest consensus method and runs fast, but its simplicity may come at the price of low integrated label quality. More complicated consensus methods may result in better integrated label quality than MV. However, noise is still inevitably present in the integrated labels.

Here, we use an illustrative example to show the noise level in integrated labels. For the discussion below, we give some notations and definitions. $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{M}$ is a training data set containing M instances, where $\mathbf{x}_i$ is the instance described by a d-dimensional feature vector and y_i is the corresponding known true label. $U = \{u_j\}_{j=1}^{R}$ denotes the labelers of a crowdsourcing system. Each instance $\mathbf{x}_i$ has a multiple label set $\mathbf{l}_i = \{l_{ij}\}_{j=1}^{R}$, where l_{ij} is the label of the instance $\mathbf{x}_i$ annotated by the labeler u_j.

* Corresponding author.
E-mail addresses: ljiang@cug.edu.cn (L. Jiang), hwli@cug.edu.cn (H. Li).

http://dx.doi.org/10.1016/j.knosys.2016.06.003
0950-7051/

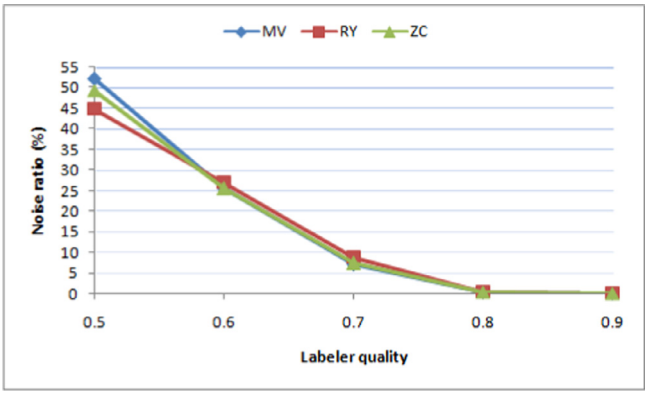

Fig. 1. Noise ratio reduces with the increase in labeling qualities and where the number of labelers is 9.

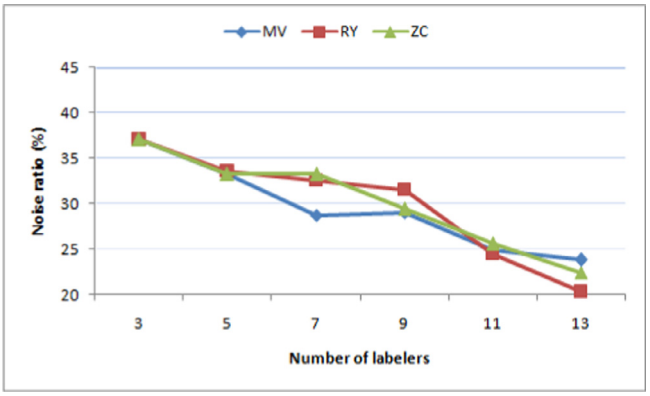

Fig. 2. Noise ratio reduces with the increase in the number of labelers and where the labeling quality of each labeler is $p = 0.6$.

Definition 1 (Overall labeling quality)**.** The overall labeling quality of a labeler u_j is the probability that the label of an instance $\mathbf{x}_i$ annotated by the labeler u_j is the true label, notated as p_j.

Definition 2 (Integrated label quality)**.** After acquiring the multiple label set of an instance $\mathbf{x}_i$, $\mathbf{l}_i = \{l_{ij}\}_{j=1}^{R}$, a certain consensus method is used to induce its integrated label, notated as $\hat{y}_i$; the integrated label quality is the probability that the integrated label is the true label, notated as q_i.

Definition 3 (Noise ratio)**.** The noise ratio (NR) of a set of integrated labels of a data set is the percentage of instances, the integrated labels of which are different from their true labels. The NR can be calculated as $NR = \sum_{i=1}^{M} I(\hat{y}_i \neq y_i)/M$, where $I(\bullet)$ is a binary function, which is 1 when this condition in this bracket is met; otherwise, 0.

In order to show the noise level in integrated labels clearly, we designed two simulation experiments. We randomly select a data set *breast-cancer* (see Section 4), and the true labels of all instances are hidden. The first experiment employs nine labelers, and each labeler generates a label for every instance according to a labeling quality p_j: the true label of every instance is assigned to the instance with probability p_j and the opposite value is assigned with probability $1 - p_j$. Here, we consider a simple case where the labeling quality of all labelers is the same, that is, $p_j = p$ for all j (we relax this assumption in Section 4). After obtaining nine labels for every instance, consensus methods MV, RY, and ZC are used to infer the integrated label of every instance. Notice that, according to a previous study [25], $p > 0.5$ must be satisfied; otherwise, the integrated label quality q_i cannot be improved by repeated labeling. The second experiment employs different numbers of labelers, but the labeling quality p is fixed ($p = 0.6$) and MV, RY, and ZC are used to infer the integrated label of every instance.

Fig. 1 shows the reduction in the NR in the set of integrated labels with the increase in labeling quality. Fig. 2 shows the reduction in NR with the increase in the number of labelers. In Fig. 1, we can see that, when the labeling quality is relatively high, the NR is relatively low, and when the labeling quality p approaches 0.9, the NR approaches 0. This means that when the labeling quality is quite high there is almost no noise in the set of integrated labels. Unfortunately, in real-world applications, the labeling quality is frequently low. In Fig. 2 we can see that, when the labeling quality of each labeler is 0.6, and even when the number of labelers increases to 13, there is still a higher level of noise in the set of integrated labels. Therefore, noise filtering is very necessary for improving crowdsourcing learning.

Noise filtering is not a new technique in the machine learning community, and it can even be stated that noise filtering is a relatively mature field. Many years ago, researchers already noticed the presence of noise in data sets, including feature noise and label noise. In this paper, we discuss label noise. However, to the best of our knowledge, in very few studies has the use of existing noise filters been considered for improving the quality of integrated labels. Since in existing studies many methods to handle label noise have been presented, our objective was to borrow from the results of these studies to enhance the crowdsourced data quality, and consequently, improve the model quality.

The rest of this paper is organized as follows. In Section 2, related work on crowdsourcing is introduced. In Section 3, some noise filters are revisited. In Section 4, experimental validation of the performance of these noise filters for deleting noise in integrated labels is presented. Section 5 gives our conclusions and future work.

2. Related work

In order to address conventional supervised learning problems in the scenario of crowdsourcing, it is very important to study the induction of an integrated label from multiple noisy labels. Many consensus methods have been designed for label integration, among which MV is the simplest. However, MV is a little rough. In order to improve the quality of integrated labels, researchers have presented more complicated consensus methods. These can be categorized into two approaches. The first comprises consensus methods based on Expectation Maximization (EM). The common idea of consensus methods based on EM is to use EM to optimize model parameters and estimate labels simultaneously. Representative methods include RY [21], ZC [4], GLAD [31], and DS [3]. The second approach is the weighted majority voting approach, which includes frequency-based majority voting (MV-Freq), Beta distribution-based majority voting (MV-Beta) [24], iterative weighted majority voting (IWMV), which optimizes the error rate bound and approximates the oracle MAP rule [17], and max-margin majority voting (M^3V) [28].

Although these consensus methods perform well in many real applications, in order to improve crowdsourcing learning, researchers have been attempting to use different techniques to improve the crowdsourced training data quality further, enhance the model performance, and reduce the cost of acquiring labels. Among these techniques, the bridging of crowdsourcing learning and active learning has attracted many researchers' attention. [36] focused on active learning for the multiple labelers scenario, and their work provided a criterion for selecting both the most uncertain instance and the labeler/s from whom to query the labels.