Knowledge-Based Systems 66 (2014) 28–35

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Detecting potential labeling errors for bioinformatics by multiple voting

Donghai Guan [a,b], Weiwei Yuan [a,c,*], Tinghuai Ma [d], Sungyoung Lee [e]

[a] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China
[b] College of Automation, Harbin Engineering University, China
[c] College of Computer Science and Technology, Harbin Engineering University, China
[d] School of Computer & Software, Nanjing University of Information Science & Technology, China
[e] Dept. of Computer Engineering, Kyung Hee University, Republic of Korea

ARTICLE INFO

Article history:
Received 7 June 2013
Received in revised form 31 March 2014
Accepted 7 April 2014
Available online 18 April 2014

Keywords:
Bioinformatics analysis
Mislabeled data detection
Single-voting
Multiple-voting
Classification

ABSTRACT

Classification techniques are important in bioinformatics analysis as they can separate various bioinformatical data into distinct groups. To obtain good classifiers, accurate labeling of the training data is required. However labeling in practical bioinformatics applications might be erroneous due to various reasons. To identify those mislabeled data, an ensemble learning based scheme, single-voting has been widely used. It generates multiple classifiers and makes use of their voting to detect mislabeled data. Single-voting scheme mainly consists of two components: data partitioning component to generate multiple classifiers, and mislabeled detection component to identify mislabeled data. Existing works in this field mainly focus on mislabeled detection part and neglect data partitioning. However, our analysis shows that data partitioning plays an important role in single-voting scheme. This analysis helps us proposing a novel multiple-voting scheme. It is superior to traditional single-voting by reducing the unreliable influence from data partitioning. Empirical and theoretical evaluations on a set of bioinformatics datasets illustrate the utility of our proposed scheme.

1. Introduction

Classification techniques are widely used for bioinformatics data analysis [1–5]. It can separate bioinformatics data with similar features into distinct sets, which can support many applications. In classification, a training set is required to train a classifier, which can be used later to classify new data. To obtain a satisfied classifier, the training data is generally required to be with accurate features and labels.

However, in the field of bioinformatics, mislabeling of training data is usually present mainly due to two reasons including subjective nature of the labeling task and the insufficient information to determine the true label. Subjective mislabeling occurs when experts give the labeling according to their personal judgments. The annotations provided by multiple experts might disagree with the general consensus, which leads to mislabeling errors. For example, in [6], 9 mislabeled samples are detected from 49 breast tumor training data. The other source of mislabeling is from insufficient information. For example, a physician may not be able to make the right diagnosis if certain expensive medical procedures are missing.

Existing study [7] has shown that even a small number of mislabeled data could dramatically degrade the performance of the obtained classifier. This has attracted many researchers to develop various techniques to address this issue [8–22]. Existing methods can be classified into two groups: robust classifier designing [8,9] and mislabeled data detecting [10–22]. Robust classifier designing mainly focuses on developing novel classifiers which are robust to mislabeled data during model training. While, mislabeled data detection is to detect and remove mislabeled data prior to training. Our study focuses on mislabeled data detection techniques, which mainly consists of two types: *k*-nearest neighbor based and ensemble learning based.

The core idea of *k*-nearest neighbor (kNN) based algorithms is to compare the label of one sample with the labels of its surrounding neighbors [10]. If there is strong inconsistency among these labels, this training sample is treated as mislabeled. One problem with this approach is from the limitation of kNN algorithm. Not every data distribution is suitable for kNN based method. There are some data distributions wherein the neighbor samples have different labels. Moreover, this group of algorithms does not propagate the mislabeling information to the detection

* Corresponding author at: College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. Tel.: +82 312012950.
E-mail address: yuanweiwei00@khu.ac.kr (W. Yuan).

http://dx.doi.org/10.1016/j.knosys.2014.04.013
0950-7051/

of other training examples, so each training sample is checked independently.

By contrast, ensemble learning based algorithms are used more widely [11–14,16,18,21] for mislabeling detection. The representative algorithms in this group are majority and consensus filtering [13]. In their algorithms, the training data is firstly randomly partitioned into several subsets. Each subset will be checked for mislabeled data separately. The checking is through the voting of multiple classifiers which are trained based on the remaining subsets. These algorithms mainly consists of two steps: data partitioning and multiple classifier voting. As partitioning and voting are executed only once, they are called single-voting scheme in this work.

As an ensemble learning based algorithm, single-voting can achieve accurate mislabeling detection performance based on the voting of multiple classifiers. For single-voting scheme, various elegant voting policies have been proposed, such as majority voting and consensus voting. However, data partitioning, an actual important part of single-voting, is usually neglected. So far random partitioning (randomly partition training data into several subsets) is widely used as it has various advantages. But on the other hand, our analysis has shown that its randomness property makes single-voting unreliable. Some successful detected mislabeled data under one partitioning case are failed to identify when the partitioning changes.

To address this issue, in this paper, we propose a novel multiple-voting scheme. Multiple-voting consists of several single-voting detectors which are different to each other due to various random partitioning. Multiple-voting is superior to single-voting by alleviating the dependency of mislabeled data detection on data partitioning. We also propose various fusion techniques to combine the decisions from different detectors, including one vote veto, majority voting, and consensus voting. Based on the proposed multiple-voting scheme, new variants of majority filtering and consensus filtering algorithms are proposed.

The comparison of multiple-voting and single-voting is analyzed both theoretically and experimentally. Experimental results indicate that our proposed scheme can effectively improve the performance of single-voting. Straightforwardness is a distinguished advantage of our scheme. It can be easily applied on existing single-voting approaches.

In summary, the main technical contribution is pointing out the limitation of existing single-voting scheme and proposing an efficient multiple-voting scheme with sufficient theoretical proofs for solving it.

2. Related works

Mislabeled training data detection and elimination is crucial to improve the accuracy of classifiers when mislabeling is present in the training set. Various techniques have been proposed, among which, ensemble learning based methods including majority filtering (*MF*) and consensus filtering (*CF*) have been widely used. *MF* utilizes the idea of majority voting, while *CF* utilizes the idea of consensus voting.

The general idea of *MF* and *CF* is as follows: They employ ensemble classifier to detect mislabeled instances by constructing a set of base-level classifiers and then using their classifications to identify mislabeled instances. The general approach is to tag an instance as mislabeled if x of the m base-level classifiers cannot classify it correctly. *MF* tags an instance as mislabeled if more than half of the m base level classifiers classify it incorrectly. *CF* requires that all base-level classifiers must fail to classify an instance as the class given by its training label for it to be eliminated from the training data.

The reason to employ ensemble classifiers in *MF* and *CF* is that ensemble classifier has better performance than each base-level classifier on a dataset if two conditions hold: (1) the probability of a correct classification by each individual classifier is greater than 0.5 and (2) the errors in predictions of the base-level classifiers are independent.

Shown in Table 1, majority filtering begins with n equalsized disjoint subsets of the training set E (step 1) and the empty output set A of detected noisy examples (step 2). The main loop (steps 3–6) is repeated for each training subset E_i . In step 4, subset E_t is formed which includes all examples from E except those in E_i, which then is used as the input an arbitrary inductive learning algorithm that induces a hypothesis (a classifier) H_j (step 6). Those examples from E_i for which majority of the hypotheses does not give the correct classification are added to A as potentially noisy examples (step 14).

Consensus filtering algorithm is shown in Table 2. Its only difference with *MF* is at step 14. In *CF*, the example in E_i is regarded as a noisy example only when all the hypotheses incorrectly classify it. Compared with *MF*, *CF* is more conservative due to the severer condition for noise identification, and which results in fewer instances being eliminated from the training set. The drawback of *CF* is the added risk in retaining bad data.

Majority filtering and consensus filtering are regarded as single-voting detectors. Single-voting detector consists of two steps. The first step is data partitioning. The training data E will be randomly divided into n equal size subsets $(E_1, E_2, \ldots, E_n)$. Then each subset E_i is taken out. Other $n-1$ subsets, $E \setminus E_i$ are used to train k different classifiers based on different classification algorithms. These k classifiers will be used as noise filters to detect the potential mislabeled data in E_i. Each classifier will classify the data in E_i individually. Suppose e is one training data in E_i; its given label is Label_e; its predicted label by classifier C is PLabel_e. If PLabel_e equals to Label_e, then classifier C will treat e as a noise-free data. Otherwise, e will be treated as a mislabeled data. Considering different classifiers (totally num. is k) might have different opinions on e, a voting mechanism is needed to combine their opinions.

3. The proposed multiple voting scheme

In single-voting, the voting of different classifiers can guarantee the reliability for mislabeling detection to some extent. However, it

Table 1
Majority filtering algorithm.

Algorithm 1: Majority Filtering (MF)

Input: E (training set)
Parameter: n (number of subjects), y (number of learning algorithms), $A_1, A_2, \ldots, A_y$ (y kinds of learning algorithms)
Output: A (detected noisy subset of E)
(1) form n disjoint almost equally sized subset of E_i, where $\bigcup_i E_i = E$
(2) $A \leftarrow \emptyset$
(3) **for** $i = 1, \ldots, n$ **do**
(4) form $E_t \leftarrow E \setminus E_i$
(5) **for** $j = 1, \ldots y$ **do**
(6) induce H_j based on examples in E_t and A_j
(7) **end for**
(8) **for** every $e \in E_i$ **do**
(9) *ErrorCounter* $\leftarrow 0$
(10) **for** $j = 1, \ldots, y$ **do**
(11) **if** H_j incorrectly classifies e
(12) **then** *ErrorCounter* $\leftarrow$ *ErrorCounter* $+ 1$
(13) **end for**
(14) **if** *ErrorCounter* $> \frac{y}{2}$, **then** $A \leftarrow A \cup \{e\}$
(15) **end for**
(16) **end for**