Semi-supervised Outlier Detection with only Positive and Unlabeled Data based on Fuzzy Clustering

Armin Daneshpazhouh
School of Electrical and Computer Engineering
Shiraz University
Shiraz, Iran
armin.dph@gmail.com

Ashkan Sami
School of Electrical and Computer Engineering
Shiraz University
Shiraz, Iran
asami@ieee.org

Abstract—The task of semi-supervised outlier detection is to find the instances that are exceptional from other data with the use of some labeled examples. In many applications such as fraud detection and intrusion detection, this issue becomes important. Most existing techniques are unsupervised and the semi-supervised approaches use both negative and positive instances to detect outliers. However, in many real-world applications, very few positive labeled examples are available. This paper proposes an effective method to address this problem. This method is based on two steps. First, extracting reliable negative instances by KNN technique and then using fuzzy clustering with both negative and positive examples for outlier detection. Experimental results on real datasets demonstrate that the proposed method outperforms the previous methods in detecting outliers.

Keywords—data mining; outlier detection; semi-supervised learning;

I. INTRODUCTION

In many real world applications like fraud detection, intrusion detection and medicine, finding such exceptional instances is more interesting than identifying the common cases.

The outlier detection techniques categorize into three groups due to the kind of supervision each method utilized. The first approach is unsupervised. The most of existing outlier detection techniques are unsupervised [1,2]. Unsupervised learning usually suffers from high false alarm rate and low detection rate without labelled information [3]. In the second approach, methods are based on supervised learning [4]. Supervised techniques require the existence of a large number of labelled examples for training set which can be a difficult and time consuming process. In many application domains, obtaining labelled data is usually expensive and require much human effort while gathering unlabeled data is much cheaper. Consequently, the third approach which is semi-supervised learning has attracted much attentions in the recent years [5,6] and has been shown to achieve better generalization performances in many fields.

In binary problems there as two sets of examples. One set is positive examples and the other set is negative examples. Suppose that the available input is incomplete set P of positive instances and set U of unlabeled examples which consists of both positive and negative examples so the task is finding the other positive instances based on the two input sets P and U. for example in fraud detection usually very few fraudulent instances are available and conserving the fact that labeling negative instances is costly, semi-supervised learning from positive and unlabeled data could be very helpful.

To address the problem of detection outliers using just few positive examples and unlabeled data, this paper proposes an accurate method based on fuzzy clustering. At first, enough reliable negative instances are extracted by means of kNN method and then fuzzy rough C-means clustering [3] will be applied to identify the other positive examples as outliers.

The rest of the paper is organized as follows. Section 2 presents a brief survey of existing related works. Section 3, describes in details the two main part of the proposed method. Section 4 illustrates the experimental results followed by conclusion in section 5.

II. RELATED WORK

The previous works on this subject are divided into three categories. Different methods for detecting outliers are presented in the first group. The second group includes different approaches in cases that only positive and unlabeled data are available. The last one contains the semi-supervised outlier detection techniques that recently have been proposed.

Previous studies on outlier mining mainly fall into the following categories.

Distribution-based methods are previously conducted by the statistics community [7,8]. Some standard distribution models have been used so the points that deviate from this models detected as outliers. For example Yamanishi [9] used a Gaussian mixture model to present the normal behavior. In [10] this approach is combined with a Supervised-based learning approach to gain general patterns for outlier.

Depth-based is the second class for outlier mining in statistics [11]. Based on some definition of depth, data objects are organized in convex hull layers in data space according to peeling depth, and outliers are expected to be found from data objects with shallow depth values [12]. These techniques are infeasible for large dataset with high dimensions due to the extreme computational complexity.
Distance-based outlier is originally suggested by Knorr and Ng [13]. In distance-based methods, outlier is a data which has the most distance from its k nearest neighbors [14]. Reference [14] defines outliers by using the point distances in all dimensions. So they have the problem of cost and unexpected performances regarding to the curse of dimensionality.

Deviation-based techniques identify outliers by inspecting the characteristics of objects and consider an object that deviates these features as an outlier [15]. The concept of “local outlier” was proposed by [16]. LOF is local outlier factor for each point which depends on local density of its neighbors. These methods confront with the problem of high dimensions too.

Cluster-based outlier detection methods only regarded small clusters as outliers or identified outliers by removing clusters from original dataset [17]. The concept of “cluster-based local outlier” is proposed in [18].

For the problem of learning from positive and unlabeled data (LUP), there is only the binary classification situation. So one class is considered as positive class while the other class is considered as negative. Traditional semi-supervised classification techniques require labeled training examples for all classes, so they are not suitable for LPU problem. However, there are four kinds of approaches for this problem.

1. Ignoring the unlabeled data and learn just from labeled positive examples. In one-class SVM [19], a suitable area for covering almost all labeled positive examples is constructed and all the other data out of this region considered as negative examples. By discarding useful information in unlabeled data, it easily makes over-fitting.
2. The most popular approach is based on co-training with labeled positive data. It usually uses two-phases strategy: 1) extracting likely negative examples from the unlabeled set; 2) using traditional learning methods for classification. PEBL [20], S-EM, NB [21] and Roc-Clu-SVM [22] use this strategy to classify the unlabeled instances.
3. B-Pr [23] and W-SVM [24] belong to the probabilistic approach. Extracting likely negative examples, is not required in these methods. However, there are several parameters need to be estimated, which directly affect the performance. Additionally, they can’t cope with imbalance datasets.
4. LGN [25] is based on the assumption that the identical distribution of words belonging to the positive class both in labeled positive examples and unlabeled data, generates an artificial negative document for learning. However, in practice, it is very hard or impossible to satisfy this assumption [26].

The last category includes outlier detection algorithms which used the semi-supervised approach as a learning method [5,6]. Gao [5] employs an objective function to detect outliers based on clustering. Method [3] is based on fuzzy rough C-means clustering and uses a different objective function for outlier detection. However, none of the existing methods can cope with the problem of semi-supervised outlier detection based on only positive and unlabeled data.

Different from the above approaches, this paper properly addresses the problem of detecting outliers using semi-supervised learning from positive and unlabeled examples.

III. The Proposed Approach: SSODPU

This section introduces the proposal SSODPU (Semi-supervised Outlier Detection with Positive and Unlabeled data) method to solve the problem where there are only a few positive examples for training. As mentioned before, this approach is based on two main steps: 1) Extracting reliable negative examples from positive and unlabeled data; 2) Detecting outliers based on the new labeled examples.

A. Extracting reliable negative examples using kNN

kNN stands for k-nearest neighbor classification, is a well-known statistical approach that has been intensively studied in pattern recognition. kNN is a type of instance-based learning where the function is only approximated locally and all computation is deferred until classification. The k-nearest neighbor algorithm is amongst the simplest of all machine-learning algorithms. kNN algorithm can use the unprocessed training set directly in classification. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors. The parameter k in kNN is often chosen based on experience or knowledge about the problem. In binary classification problems, it is desirable for k to be odd to make ties less likely.

The kNN algorithm can be adapted to LPU problem by employing a process of ranking [27]. The unlabeled instances are ranked according to their similarity to the positive examples. When the distances of unlabeled examples from k nearest positive examples are computed, the resulting values can be used for sorting the classified examples, nearer unlabeled instances take positions ahead of the ones that are further away [28]. The proposed algorithm for extracting negative examples using kNN is shown below:

Algorithm: Extracting reliable negative examples using kNN

Input: P positive examples set,
U unlabeled examples set,
K the number of nearest neighbors,
N the number of negative examples
Output: NE negative examples set

1. For each unlabeled example \( u_i \)
   Compute \( \text{sim}(u_i, v_j) \)
   Compute \( \text{rank}(u_i, v_j) \)

2. End For

3. For each positive example \( v_j \)
   Compute \( \text{sim}(u_i, v_j) \)

4. Select k nearest neighbors \( v_j \) \( (j=1,...,p) \) and remove them
5. Select top N examples with the most average similarity from $v_j (j=1,...,p)$.

B. Detecting outliers based on the new labeled examples

In this section by using both positive and negative instances, the FRSSOSD [3] can be applied to the task of outlier detection. This algorithm uses a semi-supervised learning approaches for finding outliers. FRCM [29] is a new clustering method and can synchronously handle overlap of clusters and uncertainty involved in class boundary, thereby yielding the best approximation of a given data [3]. The FRSSOSD algorithm is briefly illustrated in the following section.

Let $X=\{x_1, x_2, \ldots, x_n\}$ be a set of data points drawn from $R^d$ and the first $l < n$ points in $X$ be labeled as $y_i \in \{1,0\}$ $i = 1, 2, \ldots, l$, where $y_i = 0$ if the $i$th point is an outlier, and $y_i = 1$ otherwise. The aim is to find a $n \times c$ matrix, $u=\{u_{ik}\}$ $1 \leq i \leq n, 1 \leq k \leq c$ where $0 \leq u_{ik} \leq 1$ is fuzzy membership that the $i$th point belongs to the $k$th class, and $u_{ik} = 0$ otherwise. Thus for $x_i$:

$$
\begin{align*}
0 < \sum_{k=1}^{c} u_{ik} &\leq 1, \quad x_i \text{ is a normal point} \\
\sum_{k=1}^{c} u_{ik} & = 0, \quad x_i \text{ is an outlier point} 
\end{align*}
$$

The optimization problem is:

$$
\min_{m,v} f_m(u,v) = \sum_{i=1}^{n} \sum_{k=1}^{c} (u_{ik})^m d_{ik}^2 + \gamma_1 \left(n - \sum_{i=1}^{n} \left(\sum_{k=1}^{c} u_{ik}\right)^m\right) + \gamma_2 \sum_{i=1}^{l} \left(y_i - \sum_{k=1}^{c} u_{ik}\right)^2
$$

s.t. $0 \leq u_{ik} \leq 1, \sum_{k=1}^{c} u_{ik} \leq 1, \quad i = 1, 2, \ldots, n$, $k = 1, 2, \ldots, c$, $\gamma_1 > 0$, $\gamma_2 > 0$, $m \in (1, +\infty)$

The first term represents the sum squared error. The second term is also to constrain the number of outliers not to be too large. The third term maintains consistency of the labeling with existing labels and punishes mislabeled points. Two weighting parameters $\gamma_1$, $\gamma_2$ are applied to make these three terms compete with each other.

The FRSSOSD is fully described in [3]. 1-norm, 2-norm, $p$-norm, infinite norm and some other measures, such as, Mahalanobis distance can be applied as a distance measure to compute similarity.

IV. EXPERIMENTAL RESULTS

We have conducted a comprehensive performance study to evaluate our SSODPU algorithm. In this section, we illustrate the results. We ran our algorithm on a real-life dataset obtained from the UCI Machine Learning Repository $^1$ to test its performance against five state-of-the-art algorithms in this area. All experiments were performed on a 1.6 GHz Pentium PC with 1G of main memory running on Windows 8 (32-bit). All the algorithms are implemented in MATLAB version 7.12.0.635 (R2011a).

A. Wisconsin breast cancer data

The dataset used in this experiment is the Wisconsin breast cancer. The WDBC is commonly used as a dataset in medical diagnosis. It has 699 instances with 9 attributes. Each record is labeled as benign (458 or 65.5%) or malignant (241 or 34.5%). To form a very unbalanced distribution, we removed some of the benign and malignant records. As a result we built a dataset with 367 instances (357 benign records and 10 malignant records). To determine positive examples, we selected 3 instances among 10 malignant. The class distribution is shown in Table I.

B. Results

The purpose of this experiment is to test the performance of SSODPU algorithm when the number of positive training instances is few. Yet, considering the fact that there is no other method dealing with this problem, we decided to compare the performance of our method against some state-of-the-art unsupervised algorithms (Table I). The methods are as follow: ABOD algorithm [30], Gaussian method, INFLO algorithm [31], kNN algorithm [14], LDOF algorithm [32], LOF algorithm [16] and LoOP algorithm [33].

We altered the parameter $k$ from 1 to 65 for each algorithm and designated the best $k$ based on their results. Note that the parameter $k$ is set to 6 for the SSODPU algorithm. The result did not change significantly when the parameter $k$ is specified to alternative values. As noted in [3] we set $m=2$, $\gamma_1=0.063$, $\gamma_2=0.1$ and $\gamma_3=3$.

Regarding to measure similarity, we used Euclidean distance so when the range of the difference attributes of the data varies considerably, those attributes having large range of values easily dominate the distance and lead to the loss of information hidden in other important attribute. So we performed normalization as in [3]:

<table>
<thead>
<tr>
<th>Class name</th>
<th>Class label</th>
<th>Percentage of instances</th>
<th>Number of known labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benign (Common class)</td>
<td>Negative</td>
<td>97.28% (357)</td>
<td>0</td>
</tr>
<tr>
<td>Malignant (Rare class)</td>
<td>Positive</td>
<td>2.72% (10)</td>
<td>3</td>
</tr>
</tbody>
</table>

We employ five evaluation metrics to compare the results of our SSODPU algorithm. “First” indicates the rank of the first outlier instance detected among the result list. “Last” shows the rank of the last outlier example detected. Median, Mean (average) and Std Dev (standard deviation) are three other statistical metrics utilized to appraise the SSODPU algorithm performance.

Table IV shows the results produced by the mentioned algorithms on the dataset. In this experiment the SSODPU algorithm performed the best in almost all metrics and the results demonstrate the superiority of the proposed SSODPU algorithm over the eight novel works on the subject of outlier detection.

V. CONCLUSION

This paper provides a demonstration of the SSODPU algorithm which different from former works, tackles the problem of detecting outliers with only few labeled positive examples. This method consists of two main steps. At first, some reliable negative example will be extracted and finally by means of FRSSOD, outlier data will be fined. In addition, this paper contributes with a new point of view of learning from only positive data, showing that although there is no labeled negative examples but it is obtainable by using KNN algorithm. The results verifies that this approach surpass the previous works in the literature. The future works can be toward changing the method for the step two. However, larger testing with more real data could also cause more accurate answers.

REFERENCES


