Information-Theoretic Outlier Detection for Large-Scale Categorical Data

Shu Wu, *Member*, *IEEE*, and Shengrui Wang, *Member*, *IEEE*

Abstract—Outlier detection can usually be considered as a pre-processing step for locating, in a data set, those objects that do not conform to well-defined notions of expected behavior. It is very important in data mining for discovering novel or rare events, anomalies, vicious actions, exceptional phenomena, etc. We are investigating outlier detection for categorical data sets. This problem is especially challenging because of the difficulty of defining a meaningful similarity measure for categorical data. In this paper, we propose a formal definition of outliers and an optimization model of outlier detection, via a new concept of holoentropy that takes both entropy and total correlation into consideration. Based on this model, we define a function for the outlier factor of an object which is solely determined by the object itself and can be updated efficiently. We propose two practical 1-parameter outlier detection methods, named ITB-SS and ITB-SP, which require no user-defined parameters for deciding whether an object is an outlier. Users need only provide the number of outliers they want to detect. Experimental results show that ITB-SS and ITB-SP are more effective and efficient than mainstream methods and can be used to deal with both large and high-dimensional data sets where existing algorithms fail.

Index Terms—Outlier detection, holoentropy, total correlation, outlier factor, attribute weighting, greedy algorithms

1 Introduction

Outlier detection, which is an active research area [1], [2], [27], [25], refers to the problem of finding objects in a data set that do not conform to well-defined notions of expected behavior. The objects detected are called outliers, also referred to as anomalies, surprises, aberrants, etc. Outlier detection can be implemented as a preprocessing step prior to the application of an advanced data analysis method. It can also be used as an effective tool to discover interest patterns such as the expense behavior of a to-be-bankrupt credit cardholder. Outlier detection is an essential step in a variety of practical applications including intrusion detection [28], health system monitoring [2], and criminal activity detection in E-commerce [45], and can also be used in scientific research for data analysis and knowledge discovery in biology, chemistry, astronomy, oceanography, and other fields [2].

According to [1], [2], if the existing methods for outlier detection are classified according to the availability of labels in the training data sets, there are three broad categories: supervised, semi-supervised, and unsupervised approaches. In principle, models within the supervised or the semi-supervised approaches all need to be trained before use, while models adopting the unsupervised approach do not include the training phase. Moreover, in a supervised approach a training set should be provided with labels for anomalies as well as labels of normal objects, in contrast with the training set with normal object labels alone required by the semi-supervised approach. On the other hand, the unsupervised approach does not require any object label information. Thus the three approaches have different prerequisites and limitations, and they fit different kinds of data sets with different amounts of label information. The three broad categories of outlier detection techniques are discussed below.

The *supervised anomaly detection approach* learns a classifier using labeled objects belonging to the normal and anomaly classes, and assigns appropriate labels to test objects. The supervised approach has been studied extensively and many methods have been developed. For instance, the group of proximity-based methods includes the cluster-based "K-Means+ID3" algorithm [4], which cascades K-Means clustering and an ID3 decision tree for classifying anomalous and normal objects. The work of Barbará et al. [42] is based on statistical testing and an application of Transduction Confidence Machines, which requires k neighbors. Moreover, one-class SVMs [38], [39] have been applied broadly in this field as they do not have to make a probability density estimation. A variety of methods [40], [41] based on information theory have also been proposed. The work of Filippone and Sanguinetti [40] proposes a method to control the false positive rate in the novelty detection problem. In [41], a formal Bayesian definition of surprise is proposed.

The *semi-supervised anomaly detection approach* primarily learns a model representing normal behavior from a given training data set of normal objects, and then calculates the likelihood of a test object's being generated by the learned model. Zhang et al. [5] propose an adapted hidden Markov model for this approach to anomaly detection, while Gao et al. [46] propose a clustering-based algorithm which punishes deviation from known labels. Methods that assume availability of only the outlier objects for training

- *S. Wu is with the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China. E-mail: shu.wu@nlpr.ia.ac.cn.*
- *S. Wang is with the Department of Computer Science, University of Sherbrooke, Sherbrooke, QC J1K 2R1, Canada. E-mail: shengrui.wang@usherbrooke.ca.*

Manuscript received 27 Mar. 2011; revised 4 Oct. 2011; accepted 30 Nov. 2011; published online 16 Dec. 2011.
Recommended for acceptance by G. Cormode.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2011-03-0161.
Digital Object Identifier no. 10.1109/TKDE.2011.261.

1041-4347/13/$31.00 © 2013 IEEE Published by the IEEE Computer Society

are rare [2], because it is difficult to obtain a training data set which covers all possible abnormal behavior that can occur in the data.

The *unsupervised anomaly detection approach* detects anomalies in an unlabeled data set under the assumption that the majority of the objects in the data set are normal. Angiulli et al. [30] propose a KNN distance-based method. Clustering is another widely implemented method, of which [13] is an example. Moreover, this approach is applied to different kinds of outlier detection tasks and data sets, e.g., conditional anomaly detection [29], context-aware outliers [14], and outliers in semantic graphs [32]. As this approach does not require a labeled training data set and is suitable for different outlier detection tasks, it is the most widely applicable.

To implement supervised and semi-supervised outlier detection methods, one must first label the training data. However, when faced with a large data set with millions of high-dimensional objects and a low anomalous data rate, picking the abnormal and normal objects to compose a good training data set is time-consuming and labor-intensive. The unsupervised approach is more widely used than the other approaches because it does not need labeled information. If one wants to employ a supervised or semi-supervised approach, an unsupervised method can be used as the first step to find a candidate set of outliers, which will help experts to build the training data set. The unsupervised approach is our research focus in this paper.

1.1 Unsupervised Categorical Outlier Detection

In real applications, a large portion or the entirety of the data set is often presented in terms of categorical attributes. Examples of such data sets include transaction data, financial records in commercial banks, demographic data, etc. The problem of outlier detection in this type of data set is more challenging since there is no inherent measurement of distance between the objects. Existing unsupervised outlier detection methods, e.g., LOF [19], LOCI [24], and [13], [31], are effective on data sets with numerical attributes. However, they cannot be easily adapted to deal with categorical data.

Outlier detection methods for categorical data can be characterized by the way outlier candidates are measured w.r.t. other objects in the data set. In general, outlier candidates can be assessed based either on data distribution or on attribute correlation, which provides a more global measure. They can also be assessed using a between-object similarity or local density, which provides a local measure. Various techniques such as proximity-based [11], rule-based [10], and information-theoretic [36] methods have been proposed (Section 2 provides a more detailed discussion) and fall into one of these two categories. The common problem with the existing methods is the lack of a formal definition for the outlier detection problem. Without a formal definition, outlier detection is often designed as an ad-hoc process. In particular, several user-defined parameters are often required to define whether an object possesses properties sufficiently different from others to be qualified as an outlier. The parameter-laden results are heavily dependent on suitable parameter settings, which are very difficult to estimate without background knowledge about the data. Many existing methods also suffer from low effectiveness and low efficiency due to high dimensionality and large size of the data set, high-complexity statistical tests, or inefficient proximity-based measures.

1.2 Objectives

The goal of this paper is twofold. First, we deal with the lack of a formal definition of outliers and modeling of the outlier detection problem; second, we aim to propose effective and efficient methods that can be used to solve the outlier detection problem in real applications. In this paper, these two goals are achieved by exploring the information-theoretic approach [6].

First, in our approach, we adopt the deviation-based strategy which, according to [22], avoids the use of statistical tests and proximity-based measures to identify exceptional objects. We explore information theory [6] to derive several new concepts. In particular, we combine entropy and total correlation with attribute weighting to define the concept of weighted holoentropy, where the entropy measures the global disorder of a data set and the total correlation measures the attribute relationship. Based on this concept, we build a formal model of outlier detection and propose a criterion for estimating the "goodness" of a subset of objects as potential outlier candidates. Then outlier detection is formulated as an optimization problem involving searching for the optimal subset in terms of "goodness" and number of outliers. Finally, to solve the optimization problem, we carry out a deep investigation of the analytical and statistical properties of the proposed criterion and propose two greedy algorithms that effectively bypass probability estimation and the high complexity of exploring the whole outlier candidate space.

1.3 Contributions

The contributions of this work are as follows:

1. We propose a formal optimization-based model of categorical outlier detection, for which a new concept of weighted holoentropy which captures the distribution and correlation information of a data set is proposed.
2. To solve the optimization problem, we derive a new outlier factor function from the weighted holoentropy and show that computation/updating of the outlier factor can be performed without the need to estimate the joint probability distribution. We also estimate an upper bound of outliers to reduce the search space.
3. We propose two effective and efficient algorithms, named the Information-Theory-Based Step-by-Step (**ITB-SS**) and Single-Pass (**ITB-SP**) methods. These algorithms need only the number of outliers as an input parameter and completely dispense with the parameters for characterizing outliers usually required by existing algorithms.

The rest of this paper is organized as follows. Section 2 discusses related work and gives a detailed description of the methods which will be compared. Section 3 presents the concepts of holoentropy and modeling of outlier detection as an optimization problem. Section 4 describes the proposed algorithms for solving the detection problem. Major experimental results, including comparisons with existing methods, are presented in Section 5. Section 6 discusses a potentially interesting avenue for developing a true parameter-free detection algorithm. The conclusion is given in Section 7.

2 Related Work

Mainstream methods/algorithms designed for outlier detection from categorical data can be grouped into four categories. Some of these algorithms are compared with the proposed algorithms in Section 5.

2.1 Proximity-Based Methods

Being intuitively easy to understand, proximity-based outlier detection, which measures the nearness of objects in terms of distance, density, etc., is an important technique adopted by many outlier detection methods. For numerical outlier detection, there are a variety of methods [3], [19], [30], [33] in this category. For instance, LOF [19] is an effective method that utilizes a concept of local density to measure how isolated an object is w.r.t. the surrounding $Minpts$ objects.

For categorical data sets, the proximity-based methods must confront the problems of how to choose the measurement of distance or density and how to avoid high time and space complexity in the distance computing process. For instance, ORCA [33] uses the Hamming distance and CNB [11] employs a common-neighbor-based distance to measure the distance between categorical objects. The CNB algorithm consists of two steps, the neighbor-set generating step and the outlier mining step. The neighbor-set of the k nearest neighbors with similarity threshold θ to all objects is computed in the neighbor-set generation step. Both k and θ are user-defined parameters. In the second step, an outlier factor for each object is computed by summing its distance from its neighbors. The objects with the o (number of outliers) largest values are set to be outliers. The proximity-based approach has many prerequisite parameters, which need repeated trial-and-error to attain the desired result. Proximity-based methods also suffer from the curse of dimensionality when using distance or local density measures on the full dimensions. In general, these methods are time- and space-consuming and consequently are not appropriate for large data sets.

2.2 Rule-Based Methods

Rule-based methods borrow the concept of frequent items from association-rule mining. Such methods consider the frequent or infrequent items the data set. For instance, in the work of [20], [21], objects with few frequent items or many infrequent items are more likely to be considered as anomalous objects than others.

Frequent Pattern Outlier Factor (called the *FIB* method in this paper) [10] and Otey's Algorithm (called the *OA* method in this paper) [7] are two well-known rule-based techniques. The procedure of the FIB algorithm includes an initial computation of the set of frequent patterns, using a predefined minimum support rate. For each object, all support rates of associated frequent patterns are summed up as the outlier factor of this object. The objects with the o smallest factors are considered as the outliers. Contrary to the FIB algorithm, OA begins by collecting the infrequent items from the data set. Based on the infrequent items, the outlier factors of the objects are computed. The objects with the o largest scores are treated as outliers. The time complexity of both algorithms is determined by the frequent-item or infrequent-item generating processes. For instance, the time complexity of the FIB method is exponentially increasing with the number of attributes due to the Apriori algorithm [12]. Therefore, this approach is limited to low-dimensional data sets.

2.3 Information-Theoretic Methods

Several information-theoretic methods have been proposed in the literature. For anomaly detection in audit data sets, Lee and Xiang [36] present a series of information-theoretic measures, i.e., entropy, conditional entropy, relative conditional entropy, and information gain, to identify outliers in the univariate audit data set, where the attribute relationship does not need to be considered. The work of He et al. [23] employs entropy to measure the disorder of a data set with the outliers removed. In these methods, heuristic local search is used to minimize the objective function. The methods proposed in [8] and [9] set a threshold of mutual information and obtain a set of dependent attribute pairs. Based on this set, an outlier factor for each individual object is defined. In general, information-theoretic methods focus either on a single entropy-like measurement or on mutual information, and require expensive estimation of the joint probability distribution when the data set is shrunk following elimination of certain outliers.

2.4 Other Methods

Several other approaches using the Random Walk, Hypergraph theory, or clustering methods have been proposed to deal with the problem of outlier detection in categorical data. For instance, based on hypergraph theory, HOT [18] captures the distribution characteristics of an object in the subspaces and these characteristics are then used to identify outliers. In the random-walk-based method [34], outliers are those objects with a low probability of jumping to neighbors. In other words, they have a high probability of staying in their states. In [35], the relationships among the neighbors are considered and a mutual-reinforcement-based local outlier factor is proposed to identify outliers. This can also be viewed as a random-walk method with a fixed number of walk steps. In [37], a cluster-based local outlier detection method is proposed to identify the physical significance of an object. The outlier factor in this method is measured by both the size of the cluster the object belongs to and the distance between the object and its closest cluster. These methods are not very efficient for large or high-dimensional data sets because they contain some high-complexity procedures, e.g., frequent-item generating processes in HOT [18], similarity computation in the random-walk-based methods [34], [35], and the clustering process in the cluster-based method [37].

3 Measurement for Outlier Detection

In this section, we first look at how entropy and total correlation can be used to capture the likelihood of outlier candidates. We propose the concept of holoentropy and formulate the outlier detection problem.

3.1 Entropy and Total Correlation

Consider a set $\mathcal{X}$ containing n objects $\{x_1, x_2, \dots, x_n\}$, each x_i for $1 \leq i \leq n$ being a vector of categorical attributes $[y_1, y_2, \dots, y_m]^T$, where m is the number of attributes, y_j has a value domain determined by $[y_{1,j}, y_{2,j}, \dots, y_{n_j,j}]$ $(1 \leq j \leq m)$ and n_j indicates the number of distinct values in attribute y_j. Considering each y_j as a random variable, the random vector $[y_1, y_2, \dots, y_m]^T$ is represented by $\mathcal{Y}$. x_i can be denoted as $(x_{i,1}, x_{i,2}, \dots, x_{i,m})^T$. We use $H_{\mathcal{X}}()$, $I_{\mathcal{X}}()$, and $C_{\mathcal{X}}()$, respectively, to represent entropy, mutual information, and total correlation computed on the set $\mathcal{X}$; e.g., $I_{\mathcal{X}}(y_i; y_j)$ represents the mutual information between attributes y_i and y_j. Sometimes, we drop off the index term $\mathcal{X}$ when there is no ambiguity, e.g., using $I(y_i; y_j)$ in place of $I_{\mathcal{X}}(y_i; y_j)$.

Now, based on the chain rule for entropy [6], the entropy of $\mathcal{Y}$, denoted as $H_{\mathcal{X}}(\mathcal{Y})$ can be written as follows:

$$\begin{aligned} H_{\mathcal{X}}(\mathcal{Y}) &= H_{\mathcal{X}}(y_1, y_2, \dots, y_m) = \sum_{i=1}^{m} H_{\mathcal{X}}(y_i | y_{i-1}, \dots, y_1) \\ &= H_{\mathcal{X}}(y_1) + H_{\mathcal{X}}(y_2|y_1) + \cdots + H_{\mathcal{X}}(y_m | y_{m-1}, \dots, y_1) \end{aligned} \tag{1}$$

where

$$\begin{aligned} &H_{\mathcal{X}}(y_m | y_{m-1}, \dots, y_1) \\ &\quad = - \sum_{y_m, y_{m-1}, \dots, y_1} p(y_m, y_{m-1}, \dots, y_1) \log p(y_m | y_{m-1}, \dots, y_1). \end{aligned}$$

The entropy can be used as a global measure in outlier detection. In information theory, entropy means uncertainty relative to a random variable: if the value of an attribute is unknown, the entropy of this attribute indicates how much information we need to predict the correct value. A subset of objects is good outlier candidates if their removal from the data set causes significant decrease of the entropy of the data set. The method proposed in [36] makes use of entropy as a quality measure in outlier detection from unidimensional audio data. He et al. [23] extend this schema to measure the disorder of a multidimensional data set with the outliers removed, where a heuristic local search is employed to minimize the objective function.

Let us look at how total correlation can also be used in outlier detection. The total correlation [16] is defined as the sum of mutual information of multivariate discrete random vectors $\mathcal{Y}$, denoted as $C_{\mathcal{X}}(\mathcal{Y})$.

$$\begin{aligned} C_{\mathcal{X}}(\mathcal{Y}) &= \sum_{i=2}^{m} \sum_{\{r_1 \dots r_i\} \subset \{1, \dots, m\}} I_{\mathcal{X}}(y_{r_1}; \dots; y_{r_i}) \\ &= \sum_{\{r_1, r_2\} \subset \{1, \dots, m\}} I_{\mathcal{X}}(y_{r_1}; y_{r_2}) + \cdots + I_{\mathcal{X}}(y_{r_1}; \dots; y_{r_m}), \end{aligned} \tag{2}$$

where $r_1 \dots r_i$ are attribute numbers chosen from 1 to m. $I_{\mathcal{X}}(y_{r_1}; \dots; y_{r_i}) = I_{\mathcal{X}}(y_{r_1}; \dots; y_{r_{i-1}}) - I_{\mathcal{X}}(y_{r_1}; \dots; y_{r_{i-1}} | y_{r_i})$ [6] is the multivariate mutual information of $y_{r_1} \dots y_{r_i}$, where $I_{\mathcal{X}}(y_{r_1}; \dots; y_{r_{i-1}} | y_{r_i}) = E(I(y_{r_1}; \dots; y_{r_{i-1}}) | y_{r_i})$ is the conditional mutual information. The total correlation is a quantity that measures the mutual dependence or shared information of a data set.

Taking the case of total correlation $C_{\mathcal{X}}(y_1; y_2)$ with two attributes y_1 and y_2 as an example, $C_{\mathcal{X}}(y_1; y_2) = I_{\mathcal{X}}(y_1; y_2)$ denotes the total correlation for a random vector $\mathcal{Y}$ with two attributes y_1 and y_2. Its value corresponds to the reduction in the uncertainty of one attribute value yielded by knowledge of the other. If the value of $C_{\mathcal{X}}(y_1; y_2)$ is large, it means that the number of duplicate pairs of attribute values is small in these two attributes compared with the situation when the value of $C_{\mathcal{X}}(y_1; y_2)$ is small. In general, for the case where there are more than two attributes, larger $C_{\mathcal{X}}(\mathcal{Y})$ means a smaller number of objects sharing common attribute values, which in turn implies fewer number of frequent item sets and worse cluster structure. Thus, similar to entropy, the total correlation can be used to measure the goodness of the outlier candidates in a subset $\mathcal{O}$ by evaluating $C_{\mathcal{X}'}(\mathcal{Y})$ for $\mathcal{X}' = \mathcal{X} \backslash \mathcal{O}$. Again, the smaller the value of $C_{\mathcal{X}'}(\mathcal{Y})$, the better the subset $\mathcal{O}$ as a set of outlier candidates.

TABLE 1
Adjusting Total Correlation

# Object	y_1	y_2	y_3	y_4
x_1	a_1	a_2	a_3	a_4
x_2	a_1	a_2	b_3	b_4
x_3	a_1	a_2	c_3	c_4
x_4	a_1	a_2	d_3	d_4
x_5	a_1	a_2	e_3	e_4
x_6	a_1	a_2	f_3	f_4
x_7	b_1	b_2	g_3	g_4
x_8	c_1	c_2	g_3	g_4
x_9	d_1	d_2	g_3	g_4
x_{10}	e_1	e_2	g_3	g_4
x_{11}	f_1	f_2	g_3	g_4
x_{12}	g_1	g_2	g_3	g_4
x_{13}	b_1	d_2	c_3	a_4
x_{14}	h_1	h_2	h_3	h_4

3.2 Holoentropy

We begin here with an example to show that entropy alone is not a good enough measure for outlier detection and the contribution of the total correlation is necessary. Looking at the example in Table 1, where 14 objects with four attributes are illustrated, we represent the data set by $\mathcal{X}$. $\mathcal{X}$ includes two objects x_{13} and x_{14} which can be identified as the most likely outliers by comparison with the other 12 objects. Moreover, x_{14} is clearly more exceptional than x_{13} since it shares none of its attributes with the rest of objects. Now, $H_{\mathcal{X} \backslash x_{14}}(\mathcal{Y}) = H_{\mathcal{X} \backslash x_{13}}(\mathcal{Y}) = 3.7$ means that, if only the entropy is used, x_{14} and x_{13} are equally exceptional as outlier candidates. On the other hand, if we combine the total correlation and the entropy, we obtain $H_{\mathcal{X} \backslash x_{14}}(\mathcal{Y}) + C_{\mathcal{X} \backslash x_{14}}(\mathcal{Y}) = 9.414$ and $H_{\mathcal{X} \backslash x_{13}}(\mathcal{Y}) + C_{\mathcal{X} \backslash x_{13}}(\mathcal{Y}) = 10.030$, which allows object x_{14} to be distinguished as a more likely outlier than x_{13}. Interestingly, given the distributions of attributes in a data set, there is a complementary relationship that exists between the entropy and total correlation of $\mathcal{Y}$. It is based on Watanabe's proof [17] that the total correlation can be expressed as $C_{\mathcal{X}}(\mathcal{Y}) = \sum_{i=1}^{m} H_{\mathcal{X}}(y_i) - H_{\mathcal{X}}(\mathcal{Y})$. This motivates the following definition of holoentropy as a new measure for outlier detection.

Definition 1 (Holoentropy of a random vector). *The holoentropy $HL_{\mathcal{X}}(\mathcal{Y})$ is defined as the sum of the entropy and the total correlation of the random vector $\mathcal{Y}$, and can be expressed by the sum of the entropies on all attributes*

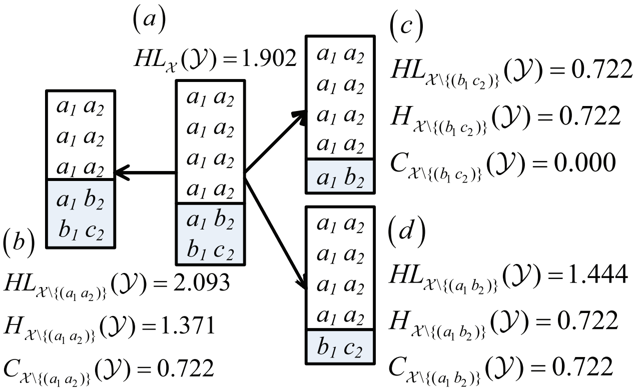

(a)
$HL_{\mathcal{X}}(\mathcal{Y}) = 1.902$
(b)
$HL_{\mathcal{X}\setminus\{(a_1\,a_2)\}}(\mathcal{Y}) = 2.093$
$H_{\mathcal{X}\setminus\{(a_1\,a_2)\}}(\mathcal{Y}) = 1.371$
$C_{\mathcal{X}\setminus\{(a_1\,a_2)\}}(\mathcal{Y}) = 0.722$
(c)
$HL_{\mathcal{X}\setminus\{(b_1\,c_2)\}}(\mathcal{Y}) = 0.722$
$H_{\mathcal{X}\setminus\{(b_1\,c_2)\}}(\mathcal{Y}) = 0.722$
$C_{\mathcal{X}\setminus\{(b_1\,c_2)\}}(\mathcal{Y}) = 0.000$
(d)
$HL_{\mathcal{X}\setminus\{(a_1\,b_2)\}}(\mathcal{Y}) = 1.444$
$H_{\mathcal{X}\setminus\{(a_1\,b_2)\}}(\mathcal{Y}) = 0.722$
$C_{\mathcal{X}\setminus\{(a_1\,b_2)\}}(\mathcal{Y}) = 0.722$
a1 a2
a1 b2
b1 c2

Fig. 1. Entropy, total correlation and holoentropy for outlier detection.

$$HL_{\mathcal{X}}(\mathcal{Y}) = H_{\mathcal{X}}(\mathcal{Y}) + C_{\mathcal{X}}(\mathcal{Y}) = \sum_{i=1}^{m} H_{\mathcal{X}}(y_i). \tag{3}$$

Note that when the components of $\mathcal{Y}$ are independent or $\mathcal{Y}$ has only one component, $HL_{\mathcal{X}}(\mathcal{Y}) = H_{\mathcal{X}}(\mathcal{Y})$, i.e., the holoentropy coincides with the entropy.

The example in Fig. 1 illustrates how holoentropy is more appropriate than entropy or total correlation for describing outliers. Fig. 1a is the original data set containing six objects, in which the object (b_1, c_2) and to a lesser extent the object (a_1, b_2) are most likely to be outliers. Figs. 1b, 1c, and Fig. 1d illustrate three possible data sets which result when one object is removed. Similar to the example in Table 1, Figs. 1c and 1d show that entropy provides no hint as to which one, (b_1, c_2) or (a_1, b_2), is more likely to be an outlier. On the other hand, if only the total correlation is taken into consideration, Fig. 1c indicates the smallest total correlation for $C_{\mathcal{X}\setminus\{(b_1\,c_2)\}}(\mathcal{Y})$ for (b_1, c_2), while Figs. 1b and 1d indicate that (a_1, a_2) and (a_1, b_2) are equally likely to be outliers, which is wrong. The holoentropy allows us to clearly establish appropriate outlier likelihoods among (b_1, c_2), (a_1, b_2), and (a_1, a_2).

Proposition 1. $0 \le HL_{\mathcal{X}}(\mathcal{Y}) \le m\log(n)$.

Proof. For an attribute y_i of $\mathcal{Y}$, if all its values are the same, the minimum entropy of this attribute satisfies $H_{\mathcal{X}}(y_i) = 0$. If all the values of y_i are different, the maximum entropy is noted as $H_{\mathcal{X}}(y_i) = \log(n)$. Since $HL_{\mathcal{X}}(\mathcal{Y}) = \sum_{i=1}^{m} H_{\mathcal{X}}(y_i)$, the inequalities hold. □

3.3 Attribute Weighting

The proposed holoentropy assigns equal importance to all the attributes, whereas in real applications, different attributes often contribute differently to form the overall structure of the data set. In this section, after demonstrating the need for attribute weighting, we will propose a simple method for weighting attributes and then modify the holoentropy by incorporating the attribute weights. The proposed weighting method computes the weights directly from the data and is motivated by increased effectiveness in practical applications rather than by theoretical necessity. In the outlier detection algorithms proposed in Section 4, the attributes are assumed to be weighted. The "unweighted" version of the proposed algorithms can be obtained simply by setting all the weights to one. In Section 5, both weighted and unweighted algorithms are evaluated.

TABLE 2
Weighted Holoentropy in Outlier Detection

# Case	$Degree$	Age	$HL_{\mathcal{X}\setminus\{x_o\}}(\mathcal{Y})$	$\mathcal{W}_{\mathcal{X}\setminus\{x_o\}}(\mathcal{Y})$
1	$Master's$	[30, 40)	3.507	1.050
2	$Master's$	[30, 40)	3.507	1.050
3	$Master's$	[30, 40)	3.507	1.050
4	$High\ School$	[30, 40)	**3.113**	**0.895**
5	$Ph.D.$	[20, 30)	**3.113**	0.967
6	$Ph.D.$	[40, 50)	**3.113**	0.967
7	$Ph.D.$	[50, 60)	**3.113**	0.967
8	$Ph.D.$	[60, 70)	**3.113**	0.967

As an example, let us look at the data from a survey on positive attitude toward science given in Table 2, where the observations (surveyed persons) are described by their education level and age range. We will argue that for outlier detection from this survey data, the attribute *Degree* is more important than the attribute *Age*.

According to the column $HL_{\mathcal{X}\setminus\{x_o\}}(\mathcal{Y})$ in Table 2, the cases 4, 5, 6, 7, and 8 are equally likely to be outliers since the removal of each results in the same decrease in the value of $HL_{\mathcal{X}\setminus\{x_o\}}(\mathcal{Y})$. In fact, each of the cases 4, 5, 6, 7, and 8 is distinguished by its value on either the *Degree* or the *Age* attribute. By looking at the internal structure of the values of each attribute, we see that *High-School* is more outstanding within *Degree* than, for example, [40, 50) is within *Age*, since [40, 50) is one of the four values that are different from the dominating value [30,40), while *High-School* is the only value different from the dominating values *Master* and *PhD*. In other words, it is the good cluster structure of the attribute *Degree*, compared to that of *Age*, that makes *High-School* more outstanding than [40,50). The weighting strategy proposed in this paper aims to give more importance to the attribute *Degree* so that the case (*High-School*, [30,40)) is identified as a more likely outlier candidate than, for example, the case (PhD, [40,50)).

Given that the holoentropy is defined as the sum of entropies of individual attributes and outliers are detected by minimizing the holoentropy through the removal of outlier candidates, our strategy consists in weighting the entropy of each individual attribute in order to give more importance to those attributes with small entropy values, e.g., *Degree* in the example of Table 2. This increases the impact of removing an outlier candidate that is outstanding on those attributes. To weight the entropy of each attribute, we propose to employ a reverse sigmoid function of the entropy, as follows:

$$w_{\mathcal{X}}(y_i) = 2\left(1 - \frac{1}{1 + \exp(-H_{\mathcal{X}}(y_i))}\right). \tag{4}$$

This reverse sigmoid is a decreasing function ranging between (0, 2). In practice, because the entropies are all positive, the weight coefficients range between 0 and 1. The weighted holoentropy is defined as follows.

Definition 2 (Weighted Holoentropy of a Random Vector). *The weighted holoentropy $\mathcal{W}_{\mathcal{X}}(\mathcal{Y})$ is the sum of the weighted entropy on each attribute of the random vector $\mathcal{Y}$*

$$\mathcal{W}_{\mathcal{X}}(\mathcal{Y}) = \sum_{i=1}^{m} w_{\mathcal{X}}(y_i) H_{\mathcal{X}}(y_i). \tag{5}$$

The weighted holoentropy is bounded according to the following proposition.

Proposition 2. $0 \leq \mathcal{W}_{\mathcal{X}}(\mathcal{Y}) \leq \frac{2m}{n+1}\log(n)$.

Proof. Since

$$\frac{\partial[w_{\mathcal{X}}(y_i)H_{\mathcal{X}}(y_i)]}{\partial H_{\mathcal{X}}(y_i)} = \left[\frac{\exp(-H_{\mathcal{X}}(y_i))}{1+\exp(-H_{\mathcal{X}}(y_i))}\right]^2 > 0,$$

$\mathcal{W}_{\mathcal{X}}(\mathcal{Y})$ of each attribute is monotonically increasing with the attribute weight. When $H_{\mathcal{X}}(y_i) = 0$, the minimum $w_{\mathcal{X}}(y_i)H_{\mathcal{X}}(y_i) = 0$. When $H_{\mathcal{X}}(y_i) = \log(n)$, the maximum value is $\frac{2}{n+1}\log(n)$. Since $HL_{\mathcal{X}}(\mathcal{Y}) = \sum_{i=1}^{m} H_{\mathcal{X}}(y_i)$, the inequalities hold. □

To illustrate the effectiveness of weighted holoentropy as an outlier factor, let's look back at the example in Table 2. The $\mathcal{W}_{\mathcal{X}\backslash\{x_o\}}(\mathcal{Y})$ column, which is impacted more by attribute *Degree* than by attribute *Age*, indicates Case 4 is more likely to be an outlier than the Cases from 5 to 8. In Section 5, we provide extensive experimental results that show it is generally more advantageous to use attribute weighting in practical applications. In Section 4, we show that the attribute weighting in (5) can be efficiently handled within the detection process.

3.4 A Formal Definition of the Outlier Detection Problem

To formally define outliers, we need to describe the condition for judging how exceptional a subset of objects is. The following definition of outliers is based on the weighted holoentropy, supposing that the number of the desired outliers o is given. A set of o candidates is the best if its exclusion from the original data set $\mathcal{X}$ causes the greatest decrease in the weighted holoentropy value, compared to all the other subsets of $\mathcal{X}$ of size o.

Definition 3 (Outliers). *Given a data set $\mathcal{X}$ with n objects and the number o, a subset $Out(o)$ is defined as the set of outliers if it minimizes $J_{\mathcal{X}}(\mathcal{Y}, o)$, defined as the weighted holoentropy of $\mathcal{X}$ with o objects removed*

$$J_{\mathcal{X}}(\mathcal{Y}, o) = \mathcal{W}_{\mathcal{X}\backslash Set(o)}(\mathcal{Y}), \tag{6}$$

where $Set(o)$ is any subset of o objects from $\mathcal{X}$. In other words

$$Out(o) = argmin\ J_{\mathcal{X}}(\mathcal{Y}, o). \tag{7}$$

Hence, outlier detection is now formulated be stated as an optimization problem. For a given o, the number of possible candidate sets for the objective function is $C_n^o = \frac{n!}{o!(n-o)!}$, which is very high. Moreover, one might have to determine the optimal value of o, i.e., how many outliers a data set really has. A possible theoretical approach to this problem is to search for a range of values of o and decide on an optimal value of o by optimizing a certain variational property of $J_{\mathcal{X}}(\mathcal{X}, o)$. We leave this as a future research direction. For now, we will focus on developing practical solutions to the optimization problem.

4 New Outlier Detection Algorithms

In this section, we propose two greedy algorithms to solve the above optimization problem for outlier detection. Our algorithms are built upon several important properties of the holoentropy. In the following discussion, we first show how the holoentropy can be efficiently estimated when only one object is removed from the data set. This can be done using the information of the removed object, without the need of estimating the probability distribution of each attribute. In addition, we propose a method to estimate the upper bound number and the candidate set of outliers to further reduce the search space for the optimization problem. Finally, we present the two algorithms accompanied with a complexity analysis.

4.1 A New Concept of the Outlier Factor

In addition to the high computational complexity of searching for the optimal subset, solving (7) also involves the problem of repeatedly estimating the weighted holoentropy, which in turn requires estimation of probability distribution of each attribute. Thus, (7) is considered as a theoretical model of outliers for which approximate solutions need to be found. Interestingly, the difference in weighted holoentropy can be estimated, especially when only one object is removed, without having to estimate attribute probabilities. This opens up the possibility of an efficient heuristic approach to solving optimization problem (7).

Definition 4 (Differential Holoentropy). *Given an object x_o of $\mathcal{X}$, the difference of weighted holoentropy $h_{\mathcal{X}}(x_o)$ between the data set $\mathcal{X}$ and the data set $\mathcal{X}\backslash\{x_o\}$ is defined as the differential holoentropy of the object x_o*

$$\begin{aligned} h_{\mathcal{X}}(x_o) &= \mathcal{W}_{\mathcal{X}}(\mathcal{Y}) - \mathcal{W}_{\mathcal{X}\backslash\{x_o\}}(\mathcal{Y}) \\ &= \sum_{i=1}^{m}\left[w_{\mathcal{X}}(y_i)H_{\mathcal{X}}(y_i) - w_{\mathcal{X}\backslash\{x_o\}}(y_i)H_{\mathcal{X}\backslash\{x_o\}}(y_i)\right]. \end{aligned} \tag{8}$$

Since $w_{\mathcal{X}}(y_i)$ is defined as a reverse sigmoid function of the entropy $H_{\mathcal{X}}(y_i)$, the difference between $w_{\mathcal{X}}(y_i)$ and $w_{\mathcal{X}\backslash\{x_o\}}(y_i)$ is significantly smaller than the entropy $H_{\mathcal{X}}(y_i)$. So we simplify the differential holoentropy using the following expression:

$$\hat{h}_{\mathcal{X}}(x_o) = \sum_{i=1}^{m} w_{\mathcal{X}}(y_i)\left[H_{\mathcal{X}}(y_i) - H_{\mathcal{X}\backslash\{x_o\}}(y_i)\right]. \tag{9}$$

Our preliminary experiment indicates that the performance of exact and approximate outlier factor are very similar. To avoiding the high time complexity of exact factor computation, we use the approximate factor to represent the approximate one in this work. The approximate differential holoentropy $\hat{h}_{\mathcal{X}}(x_o)$ can be directly computed according to the following proposition.

Proposition 3. *The approximate differential holoentropy $\hat{h}_{\mathcal{X}}(x_o)$ can be represented as follows:*

$$\begin{aligned} \hat{h}_{\mathcal{X}}(x_o) &= \sum_{i=1}^{m} w_{\mathcal{X}}(y_i)\Big(\log a - \frac{a}{b}\log b\Big) - aW_{\mathcal{X}}(\mathcal{Y}) \\ &\quad + a\sum_{i=1}^{m}\begin{cases} 0, & if\ n(x_{o,i}) = 1; \\ w_{\mathcal{X}}(y_i)\cdot\delta\big[n(x_{o,i})\big], & else. \end{cases} \end{aligned} \tag{10}$$

where $\delta(x) = (x-1)\log(x-1) - x\log x$, and $x_{o,i}$ means the value appears in the ith attribute of the object x_o. $n(x_{o,i})$ is the simplified form of $n(i, x_{o,i})$, which means the times $x_{o,i}$ appears *in the ith attribute. b and a are reciprocal values of the cardinality of $\mathcal{X}$ and $\mathcal{X}\backslash\{x_o\}$.*

Proof. $\hat{h}_{\mathcal{X}}(x_o) = \sum_{i=1}^{m} w_{\mathcal{X}}(y_i)[H_{\mathcal{X}}(y_i) - H_{\mathcal{X}\backslash\{x_o\}}(y_i)]$; when $n(x_{o,i}) = 1$, $H_{\mathcal{X}}(y_i) - H_{\mathcal{X}\backslash\{x_o\}}(y_i)$ is written as

$$\begin{aligned} &a \sum_{j=1, j\neq o}^{n_i - 1} \big[n(x_{j,i}) \log n(x_{j,i}) + n(x_{j,i}) \log a\big] \\ &\quad - b \sum_{j=1, j\neq o}^{n_i - 1} \big[n(x_{j,i}) \log n(x_{j,i}) + n(x_{j,i}) \log b\big] - b \log b; \end{aligned}$$

when $n(x_{o,i}) > 1$, $H_{\mathcal{X}}(y_i) - H_{\mathcal{X}\backslash\{x_o\}}(y_i)$ is written as

$$\begin{aligned} &a \sum_{j=1, j\neq o}^{n_i - 1} \big[n(x_{j,i}) \log n(x_{j,i}) + n(x_{j,i}) \log a\big] \\ &\quad - b \sum_{j=1, j\neq o}^{n_i - 1} \big[n(x_{j,i}) \log n(x_{j,i}) + n(x_{j,i}) \log b\big] - a \log a \\ &\quad + (a \log a - b \log b) n(x_{o,i}) - b \cdot n(x_{o,i}) \log n(x_{o,i}) \\ &\quad + a\big[n(x_{o,i}) - 1\big] \log \big[n(x_{o,i}) - 1\big]. \end{aligned}$$

Combining these two situations, the deduced form of $\hat{h}_{\mathcal{X}}(x_o)$ is expressed as follows:

$$\begin{aligned} \hat{h}_{\mathcal{X}}(x_o) &= a \sum_{i=1}^{m} w_{\mathcal{X}}(y_i) \cdot \delta\big[n(x_{o,i})\big] \\ &+ \sum_{i=1}^{m} \left[\log \frac{a}{b} + ab \cdot \log \left(\frac{n(x_{1,i})}{n}^{\frac{n(x_{1,i})}{n}} \cdots \frac{n(x_{n_{i,i}})}{n}^{\frac{n(x_{n_{i,i}})}{n}} \right)\right]. \end{aligned}$$

Since $\log(\frac{n(x_{1,i})}{n}^{\frac{n(x_{1,i})}{n}} \cdots \frac{n(x_{n_{i,i}})}{n}^{\frac{n(x_{n_{i,i}})}{n}}) = -\frac{1}{b}(E(y_i) + \log b)$, the simplified deduced form is

$$\begin{aligned} \hat{h}(x_o) &= \sum_{i=1}^{m} w_{\mathcal{X}}(y_i) \Big(\log a - \frac{a}{b} \log b\Big) - a W_{\mathcal{X}}(\mathcal{Y}) \\ &+ a \sum_{i=1}^{m} \begin{cases} 0, & if\ n(x_{o,i}) = 1; \\ w_{\mathcal{X}}(y_i) \cdot \delta\big[n(x_{o,i})\big], & else. \end{cases} \end{aligned}$$

□

If we consider only the unweighted holoentropy, i.e., all the attribute weights are treated as 1, Proposition 3 holds for the differential holoentrepy $h_{\mathcal{X}}(x_o)$. We will use this exact equation to derive the formula for updating entropies and attribute weights in the next section. Also, according to Proposition 3, $\hat{h}(x_o)$ is determined by the data set $\mathcal{X}$, i.e., in the first two terms, $\sum_{i=1}^{m} w_{\mathcal{X}}(y_i)(\log a - \frac{a}{b} \log b) - a W_{\mathcal{X}}(\mathcal{Y})$, and by the object x_o itself in the third terms. Based on these discussions, we define the outlier factor of an object as follows.

Definition 5 (Outlier Factor of an Object). *The outlier factor of an object x_o, denoted as $OF(x_o)$, is defined as*

$$\begin{aligned} OF(x_o) &= \sum_{i=1}^{m} OF(x_{o,i}) \\ &= \sum_{i=1}^{m} \begin{cases} 0, & if\ n(x_{o,i}) = 1; \\ w_{\mathcal{X}}(y_i) \cdot \delta\big[n(x_{o,i})\big], & else. \end{cases} \end{aligned}$$

where $OF(x_{o,i})$ is defined as the outlier factor of x_o on the ith attribute.

$OF(x_o)$ can be considered as a measure of how likely it is that object x_o is an outlier. An object x_o with a large outlier factor value is more likely to be an outlier than an object with a small value. Here are a few other interesting properties of the outlier factor.

Proposition 4. *$OF(x_{u,i}) \geq OF(x_{j,i})$, if $n(x_{u,i}) = 1$ and $n(x_{j,i}) \geq 1$.*

Proof. The outlier factor has a negative or zero value on an attribute; when $x_{u,i}$ is unique, the outlier factor achieves its largest value, zero. So the proposition holds. □

Proposition 5. *$OF(x_{j,i}) \geq OF(x_{k,i})$, if $n(x_{j,i}) \leq n(x_{k,i})$ and $n(x_{j,i}) > 1$.*

Proof. Set $\alpha(x_{j,i}) = [\frac{n(x_{j,i})^{n(x_{j,i})}}{(n(x_{j,i})-1)^{n(x_{j,i})-1}}]^{w_{\mathcal{X}}(y_i)}$, $\varphi(x_{j,i}, x_{k,i}) = \frac{\alpha(x_{k,i})}{\alpha(x_{j,i})}$ and

$$\phi(x_{j,i}, x_{k,i}) = \log(\varphi(x_{j,i}, x_{k,i})) = OF(x_{j,i}) - OF(x_{k,i}).$$

Since

$$\begin{aligned} \alpha'(x_{j,i}) = w_{\mathcal{X}}(y_i) & \frac{{x_{j,i}}^{x_{j,i}} [In x_{j,i} - In(x_{j,i} - 1)]}{(x_{j,i} - 1)^{x_{j,i} - 1}} \\ & \left[\frac{n(x_{j,i})^{n(x_{j,i})}}{(n(x_{j,i}) - 1)^{n(x_{j,i}) - 1}}\right]^{w_{\mathcal{X}}(y_i) - 1} > 0, \end{aligned}$$

$\alpha(x_{j,i}) > 0$, hence $\varphi(x_{j,i}, x_{k,i}) \geq 1$, and thus $\phi(x_{j,i}, x_{k,i}) \geq 0$. When $n(x_{j,i}) = n(x_{k,i})$, the equality holds. □

According to Propositions 4 and 5, for each attribute, the outlier factor is monotonically decreasing w.r.t. the frequency of the object value on that attribute. This corresponds to the following intuitive idea: given an object, regardless of the weight of an attribute, the higher the frequency of the object value on that attribute, the less likely it is that the object is an outlier.

4.2 Updating the Outlier Factor

In this section, we discuss the issue of updating the outlier factor within a constant time in a step-by-step process. To update $OF(x_o)$, according to Definition 5 and the definition of attribute weight in (4), we should first update the entropy of each attribute. Since the attribute entropy is always changing when outliers are detected and removed from the data set, the direct computation of $H_{\mathcal{X}\backslash\{x_o\}}(y_i)$ is very time-consuming. By a line of reasoning similar to the proof of Proposition 3, the unweighted differential holoentropy $HL_{\mathcal{X}}(\mathcal{Y}) - HL_{\mathcal{X}\backslash\{x_o\}}(\mathcal{Y})$ can be deduced as follows:

$$\begin{aligned} & HL_{\mathcal{X}}(\mathcal{Y}) - HL_{\mathcal{X}\backslash\{x_o\}}(\mathcal{Y}) \\ &\quad = m\Big[\Big(\frac{a}{b} - a\Big) \log a - (b+1) \log b\Big] - b HL_{\mathcal{X}}(\mathcal{Y}) \\ &\qquad + a \sum_{i=1}^{m} \begin{cases} 0, \ if\ n(x_{o,i}) = 1; \\ \delta\big[n(x_{o,i})\big], \quad else. \end{cases} \end{aligned} \tag{11}$$

Based on this expression, we can obtain the simple updated form of the holoentropy $HL_{\mathcal{X}\backslash\{x_o\}}(\mathcal{Y})$ as

$$\begin{aligned} HL_{\mathcal{X}\setminus\{x_o\}}(\mathcal{Y}) = {} & (1+b)HL_{\mathcal{X}}(\mathcal{Y}) - m \\ & \left[\left(\frac{a}{b} - a\right)\log a - (b+1)\log b\right] \\ & - a\sum_{i=1}^{m}\begin{cases} 0, & if\ n(x_{o,i}) = 1; \\ \delta\big[n(x_{o,i})\big], & else. \end{cases} \end{aligned}$$

From this, the formula for each individual attribute entropy $H_{\mathcal{X}\setminus\{x_o\}}(y_i)$ is obtained

$$\begin{aligned} H_{\mathcal{X}\setminus\{x_o\}}(y_i) = {} & (1+b)H_{\mathcal{X}}(y_i) \\ & - \left[\left(\frac{a}{b} - a\right)\log a - (b+1)\log b\right] \\ & - a\begin{cases} 0,\ if\ n(x_{o,i}) = 1; \\ \delta\big[n(x_{o,i})\big], \quad else. \end{cases} \end{aligned} \tag{12}$$

This can be efficiently implemented in a step-by-step process. After calculating the entropy by (12), we can easily compute the updated attribute weight using (4). Finally, using Definition 5, the outlier factor can be efficiently updated.

4.3 Upper Bound on Outliers

In unsupervised outlier detection, the majority of objects in a data set are supposed to be normal objects [1]. How can we estimate an upper limit on the number of outliers in a data set? And how can we divide the data set into normal objects and anomaly (outlier) candidates? In this section, we introduce three new concepts: the upper bound on outliers ($\mathbf{UO}$), the anomaly candidate set ($\mathbf{AS}$), and the normal object set ($\mathbf{NS}$).

These concepts are constructed on the assumption that eliminating outliers will improve the purity of the data set and that this process reduces $\mathcal{W}_{\mathcal{X}}(\mathcal{Y})$. When a normal object is removed from the data set, the value of $\mathcal{W}_{\mathcal{X}}(\mathcal{Y})$ should increase. Thus, the objects with positive $\hat{h}(x_i)$ are defined as the anomaly candidate set (AS), $AS = \{x_i, |\hat{h}(x_i) > 0|\}$. The objects with nonpositive $\hat{h}(x_i)$ are defined as elements of the normal object set (NS), $NS = \{x_i, |\hat{h}(x_i) \le 0|\}$. The number of objects in AS is defined as UO

$$\begin{aligned} & AS = \{x_i, |\hat{h}(x_i) > 0|\}, \\ & UO = N(AS) = \sum_{i=1}^{n}\big(\hat{h}(x_i) > 0\big). \end{aligned} \tag{13}$$

AS will be used as the outlier candidate set; i.e., only the UO objects from AS will be examined by our algorithms. For instance, the UO in Fig. 1a is 2, the AS contains two elements $\{a_1$, $b_2\}$ and $\{b_1$, $c_2\}$, and the rest of the objects $\{a_1, a_2\}$ are normal objects. Later in the paper, we will provide extensive evidence on the adequacy of limiting the outlier search to AS. It is worth pointing out that the normal object set NS can be of great interest as the candidate set for frequent-item mining and class-profile building. In this paper, we are focusing only on the use of AS for outlier detection. For the experimental data sets, the UO values are listed in Table 5. Note that the average UO is about $0.21n$.

4.4 ITB-SP and ITB-SS Algorithms

In this section, we make use of the outlier factor defined in section 4.1 to derive two greedy algorithms for outlier detection. One is named **ITB-SS** for Information-Theory-Based Step-by-Step (or **SS** for short), the other one is named **ITB-SP** for Information-Theory-Based Single-Pass (or **SP** for short). Both algorithms detect outliers one by one. At each step of SS, the object with the largest $OF(x_o)$ is identified as an outlier and is removed from the data set. Following this removal, the outlier factor $OF(x)$ is updated for all the remaining objects. The process repeats until o objects have been removed. In SP, the outlier factors are computed only once, and the o objects with the largest $OF(x)$ values are identified as outliers. In both algorithms, search is conducted only within the anomaly candidate set AS, although this does not make any difference for the algorithm ITB-SP since the initialization of AS requires computation of the outlier factors of all the objects. ITB-SS does benefit, however, from the reduced search space. In designing the two algorithms, we assumed that the number of requested outliers o is always smaller than UO. Experimental results in the next section show that AS is indeed large enough to include all the candidate objects that can reasonably be considered as outliers. Nevertheless, only minor modifications need to be made if a user wants to obtain more than UO "outliers."

Let's look at the time complexity of ITB-SP (Algorithm 1). In ITB-SP, the attribute weights $w_{\mathcal{X}}(y_i)(1 \le i \le m)$, the $OF(x_i)$ of all the objects, initialization of AS and the heapsort search to find the top-o outlier candidates are computed. The time complexity of computing $w_{\mathcal{X}}(y_i)$ and $OF(x_i)$, including initialization of AS, is $O(mn)$, and the time cost of top-o searching is $O(nlog(o))$. Since the value of $log(o)$ is always much smaller than the number of attributes m in real applications, the final time complexity of ITB-SP can be written as $\mathbf{O(nm)}$.

Algorithm 1. ITB-SP single pass

```
Input: data set $\mathcal{X}$ and number of outliers requested $o$
Output: outlier set $OS$
Compute $w_{\mathcal{X}}(y_i)$ for $(1 \le i \le m)$ by (4)
Set $OS = \phi$
for $i = 1$ to $n$ do
    Compute $OF(x_i)$ and obtain $AS$ by (13)
end for
if $o > UO$ then
    $o = UO$
else
    Build $OS$ by searching for the $o$ objects with greatest
        $OF(x_i)$ in $AS$ using heapsort
end if
```

For ITB-SS (Algorithm 2), the attribute weights, initial outlier factors including initialization of AS, and the step-by-step top-o outlier selection procedure are computed. The time cost of attribute weights, initial outlier factors, and initialization of AS is $O(mn)$, and the time complexity of step-by-step top-o outlier selection from step 11-15 is $O(om(UO))$. Thus, the overall complexity is $O(nm + om(UO))$. Considering that $o(UO)$ is usually larger than n, it is possible to say that the final complexity of ITB-SS is $\mathbf{O(om(UO))}$. Compared with ITB-SP, the time complexity of the ITB-SS method is a little higher.

Algorithm 2. ITB-SS Step-by-Step

```
1: Input: data set $\mathcal{X}$ and number of outliers requested $o$
2: Output: outlier set $OS$
3: Set $OS = \phi$
4: Compute $w_{\mathcal{X}}(y_i)$ for $(1 \le i \le m)$ by (4)
```

TABLE 3
Comparison among ITB-SP, ITB-SS, and Optimal Solutions on Soybean Data

o	ITB-SP	$J_{\mathcal{X}}(\mathcal{Y}, o)$	ITB-SS	$J_{\mathcal{X}}(\mathcal{Y}, o)$	Optimal	$J_{\mathcal{X}}(\mathcal{Y}, o)$
1:	11	9.686	11	9.686	11	9.686
2:	11,18	9.687	11,18	9.687	11,18	9.687
3:	11,**15**,18	**9.687**	11,**15**,18	**9.687**	11,16,18	9.676
4:	11,15,16,18	9.671	11,15,16,18	9.671	11,15,16,18	9.671
5:	11,15,16,18,20	9.659	11,15,16,18,20	9.659	11,15,16,18,20	9.659
6:	11,15,**16**,18,19,20	**9.646**	11,13,15,18,19,20	9.642	11,13,15,18,19,20	9.642
7:	11,13,15,16,18,19,20	9.585	11,13,15,16,18,19,20	9.585	11,13,15,16,18,19,20	9.585
8:	11,13,**14**,15,16,18,19,20	**9.541**	11,13,15,16,17,18,19,20	9.537	11,13,15,16,17,18,19,20	9.537
9:	11,13,14,15,16,18,19,20,**29**	**9.493**	11,13,14,15,16,17,18,19,20	9.468	11,13,14,15,16,17,18,19,20	9.468
10:	11,12,13,14,15,16,18,19,20,**29**	**9.419**	11,12,13,14,15,16,17,18,19,20	9.334	11,12,13,14,15,16,17,18,19,20	9.334

```
for $i = 1$ to $n$ do
    Compute $OF(x_i)$ and obtain $AS$ by (13)
end for
if $o > UO$ then
    $o = UO$
else
    for $i = 1$ to $o$ do
        Search for the object with greatest $OF(x_o)$ from $AS$
        Add $x_0$ to $OS$ and remove it from $AS$
        Update all the $OF(x)$ of $AS$
    end for
end if
```

5 Experiments

In this section, we conduct effectiveness and efficiency tests to analyze the performance of the proposed methods. To test effectiveness, we compare ITB-SS and ITB-SP with competing methods on synthetic and real data sets. For the efficiency test, we conduct evaluations on synthetic data sets to show how running time increases with the number of objects, the number of attributes and the number of outliers.

5.1 Compared Methods and Experiment Outline

For our experiments, we implement and compare our algorithms with several mainstream methods for categorical outlier detection. These representative methods include CNB from the proximity-based approach and FIB and OA from the rule-based approach. Since the anomaly candidate set (AS) is utilized as a pruning facility to reduce the time complexity of the proposed methods, ITB-SS and ITB-SP can be considered as top-N outlier detection methods [47]. To the best of our knowledge, for categorical outlier detection, there is no other clear claim in the literature of a top-N outlier detection method. Some efficient top-N methods do exist for numerical outlier detection [43], [44], but these methods cannot be easily adapted to deal with categorical data because to reach the top-N they explore properties of their distance measures that are difficult to generalize to categorical data. In a preliminary test, we tried to adapt the LOF method [19] and its efficient top-N variation [43] with a microcluster pruning mechanism [44] to categorical data sets. The adapted methods did not work very well in our experiments. For reasons of fairness, we decided not to include any comparison with an adapted method from numerical outlier detection.

Various experimental results are reported in this section. To evaluate the proposed methods, we begin by comparing the performance of ITB-SS and ITB-SP with the optimal solutions obtained by exhaustive search on a small real data set. Although limited in the size of the test data set, this experiment illustrates that the proposed methods are able to provide very good solutions to the high-complexity optimization problem. Experiments on different synthetic data in this section can be used as evidence to illustrate the effectiveness and stability of the proposed methods for large-scale data sets. Outlier factors of different methods are compared to gain a better understanding of the advantage of the proposed methods. Extensive comparisons on real data sets allow us to judge the effectiveness of the proposed methods in comparison with other methods. Moreover, we include in these comparisons the detection performance of ITB-SS and ITB-SP in both their weighted and unweighted versions. This illustrates the benefit and importance of weighting the attributes. Finally, to evaluate the efficiency of the proposed methods, synthetic data sets are utilized to test the run time w.r.t. increasing numbers of objects, attributes, and outliers.

5.2 Effectiveness Test

5.2.1 *Evaluation of Approximation*

This section reports on experiments conducted to see whether the solutions obtained by ITB-SS and ITB-SP are close to the optimal solutions obtained by optimizing the object function $J_{\mathcal{X}}(\mathcal{Y}, o)$. The data set used is the public, categorical "soybean data" [50], with 47 objects and 35 attributes. This data contains a very small class of 10 objects (numbers 11 to 20 in the original data set). Since the data does not have explicitly identified outliers, it is natural to treat the objects of the smallest class as "outliers." Therefore, we should check whether objects from this class will be detected for $o = 1, \ldots, 10$.

Table 3 shows different sets of "outliers" obtained by ITB-SP, ITB-SS, and the optima for different values of o. The $J_{\mathcal{X}}(\mathcal{Y}, o)$ values in **bold-faced** letters indicate the cases where non-optimal sets were detected by either ITB-SP or ITB-SS, while the subsets of objects 11 to 20, which originally belong to the smallest class, found by strictly optimizing the $J_{\mathcal{X}}(\mathcal{Y}, o)$ are taken as reference sets of optimality. It can be observed that ITB-SS seems to be quite

TABLE 4
Outlier Factors of Different Methods on a Synthetic Data Set

obj.	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	CNB	FIB	OA	ITB
1:	k_1	j_2	k_3	k_4	e_5	c_6	e_7	b_8	2.00	0.44	25.0	-0.51
2:	j_1	i_2	j_3	j_4	b_5	c_6	d_7	e_8	2.00	0.44	25.0	-0.51
3:	i_1	h_2	i_3	i_4	d_5	d_6	b_7	a_8	2.00	0.44	25.0	-0.48
4:	h_1	g_2	h_3	h_4	c_5	b_6	b_7	d_8	2.00	0.44	25.0	-0.51
5:	g_1	f_2	g_3	g_4	b_5	a_6	a_7	b_8	2.00	0.89	23.0	-0.99
6:	f_1	e_2	f_3	f_4	a_5	b_6	a_7	a_8	2.00	0.89	23.0	-0.99
7:	a_1	a_2	c_3	c_4	a_5	a_6	c_7	c_8	2.00	0.89	23.0	-1.01
8:	a_1	b_2	a_3	a_4	f_5	e_6	f_7	f_8	2.00	0.89	23.0	-0.99
9:	b_1	a_2	a_3	b_4	g_5	f_6	g_7	g_8	2.00	0.89	23.0	-0.99
10:	c_1	b_2	b_3	d_4	h_5	g_6	h_7	h_8	2.00	0.44	25.0	-0.51
11:	d_1	d_2	b_3	a_4	i_5	h_6	i_7	i_8	2.00	0.44	25.0	-0.48
12:	b_1	c_2	d_3	e_4	g_5	i_6	j_7	j_8	2.00	0.44	25.0	-0.51
13:	e_1	c_2	e_3	b_4	k_5	g_6	k_7	k_8	2.00	0.44	25.0	-0.51

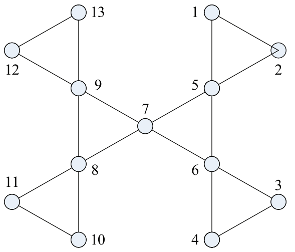


Fig. 2. Graph drawing of the synthetic data set.

effective, since it falsely detects an outlier subset only once in the 10 tries. As can be anticipated, ITB-SP makes more mistakes (5 out of 10 subsets). Nevertheless, the ITB-SP process is able to approximate the optimal solutions quite well when more and more outliers are detected. Also, if we look at the outlier output of each detection step, there is never more than one wrongly detected object. Similar phenomena have been observed with our other evaluations of approximation experiments.

5.2.2 Test of Outlier Factors

The experiments reported in this section help to understand why ITB-SS and ITB-SP are effective in solving the outlier detection problem. Here, we show some important differences between the outlier factors used in different algorithms. For this purpose, we make use of a synthetic data set, illustrated in Table 4 by $y_1, \ldots, y_8$, and compare the outlier factor values, also illustrated in Table 4. The 13 objects are different from each other. In order to visualize the data set, we draw a two-dimensional representation in Fig. 2, using the principle of graph drawing [26]. In this graph, the vertices indicate the objects and the edges represent the similarity between objects, where all the similarities are 1. The columns CNB, FIB, OA, and ITB show the outlier factor values of each object obtained by the compared methods. Note that for OA, CNB, and ITB, an object with a larger outlier factor is more likely to be an outlier, while for FIB the opposite is true. The column ITB represents $OF(x_o)$ defined in this paper. The settings of the parameters for the other methods, are as follows: similarity threshold and number of nearest neighbors in CNB are set to $\theta = 0.1$ and $k = 2$; minimum support rate in OA and FIB is set to $SupRate = 0.1$.

The results indicate that our proposed factor $OF(x_o)$ for ITB better reflects the intuitive understanding of the data set. Specifically, the column CNB shows that all objects obtain the same outlier factor value. So for CNB, all the objects are equally likely to be outliers. FIB and OA make a similar distinction between objects 5-9 and the rest of the objects. They improve on the assessment of CNB by assigning a greater likelihood of being outliers to objects 1-4 and 10-13. It is ITB that provides the most precise assessment. It indicates that object 7 in the middle of the data set is less likely to be an outlier than objects 5, 6, 8, and 9, which are similar to each other but have a common similar object 7. Moreover, objects 5, 6, 8, and 9 are less likely to be outliers than objects 1-4 and 10-13, each of which is similar to only two other objects. These differences are important indices used by ITB-SP and ITB-SS to accurately identify the most likely outlier candidate.

5.2.3 Test on Real Data Sets

A large number of public real data sets, most of them from UCI [50], are used in our experiments, representing a wide range of domains in science and the humanities. Some of them have already been used as benchmarks for intrusion and outlier detection [7], [10], [11]. Some data sets such as web-advertisement [50][1] and sampled KDD Cup 1999 Data [50][2] contain already labeled anomaly objects. The others are categorical or mixed-type data sets with class labels representing many different data distributions in the real world. For these data, we use the same strategy as [10], [11] to choose the objects in the smallest classes as the most likely anomalies.

Numeric attributes in these real data sets are, for the sake of simplicity, discretized by 10-bin discretization [48]. It is possible to adapt ITB-SS and ITB-SP to continuous attributes either through extending the holoentropy, or through a more sophisticated discretization method [48], e.g., equal distance discretization, equal frequency discretization, unsupervised clustering methods and so on. But this may require an extensive effort and will be investigated as part of our future work. For the experiments in this paper, the adopted discretization scheme is fair for all the tested algorithms.

The other general setting of our experiments is as follows: all the missing values are replaced with the modes in the corresponding categorical attributes. The Area Under the Curve (**AUC**) (curve of detection rate and false alarm rate) [1], [2] and significance test are used to measure the performance. The AUC results of different methods and the characteristics of all test data sets, such as the numbers of objects (#n), attributes (#m) and outliers (#o), and the upper bound on outliers (#UO), are summarized in the upper part of Table 5. There is no result for CNB on the KDD data set because the time and space complexities of CNB are too high for this large set. Similarly, there is no result for either FIB or OA on the web advertisement data set, because the

[1] The web-advertisement data represents a snapshot of image advertisements that have appeared on Internet pages. It is composed of major objects of "normal" images and some "bad" images, i.e., advertisements.

[2] The 10-percent KDD Cup 1999 Data has some attacks and "good" normal connections. Since the number of attacks is greater than the number of normal connections, we select a total of 157,663 normal objects and randomly choose 11,213 attacks to make the "bad" objects occupy a small part of the whole data set.

TABLE 5
AUC Results of Tested Algorithms on the Real and Synthetic Data Sets

	Dataset	#n	#m	#o	#UO	CNB	FIB	OA	unweighted ITB-SP	ITB-SP	unweighted ITB-SS	ITB-SS
Real Data-sets	autos	133	26	12	58	0.588	0.753	0.588	**0.786**	0.762	0.776	0.757
	breast-c.	495	11	45	125	0.993	0.909	**0.996**	0.991	0.993	0.993	**0.996**
	breast-w.	699	10	241	281	0.975	0.989	0.989	0.984	0.985	0.990	**0.992**
	credit-a	413	17	30	171	0.844	0.926	0.875	0.888	0.935	0.925	**0.969**
	diabetes	768	9	268	340	0.869	0.885	0.769	0.758	0.797	0.835	**0.907**
	ecoli	336	8	9	144	0.894	0.921	0.965	0.968	0.986	0.974	**0.989**
	glass	187	10	12	83	0.566	0.681	0.681	**0.782**	0.767	0.773	0.748
	heart-h	294	14	106	132	0.650	0.780	0.695	0.727	0.728	**0.842**	0.800
	heart-s.	270	15	120	128	0.707	0.778	0.788	0.705	0.707	0.827	**0.849**
	hepati.	155	21	32	72	0.714	0.870	0.876	0.831	0.854	0.888	**0.903**
	ionosph.	351	45	126	183	0.559	0.492	0.563	0.554	0.614	0.561	**0.681**
	kr-vs-kp	1829	37	160	733	**1.000**	0.955	0.937	0.939	0.935	0.955	0.953
	labor	57	17	20	30	0.453	0.762	0.811	0.568	0.647	0.717	**0.873**
	splice	1795	61	140	1657	0.568	0.878	0.635	0.995	**0.996**	**0.996**	**0.996**
	tic-tac-toe	688	10	62	294	0.996	**1.000**	0.966	**1.000**	0.967	**1.000**	0.955
	voting	293	17	26	101	**0.989**	0.976	**0.989**	0.966	0.974	0.977	0.984
	vowel	750	14	30	306	0.679	0.577	**1.000**	0.834	0.798	0.801	0.781
	zoo	90	18	6	53	0.300	**0.844**	0.597	0.784	0.746	0.831	0.816
	kdd	168876	42	11213	32923	*-	0.930	0.940	0.937	0.945	0.953	**0.954**
	police	122	3	7	18	0.882	0.988	0.977	0.981	**0.993**	**0.993**	**0.993**
	web-ad.	3279	1558	458	736	0.719	*-	*-	0.705	0.701	0.735	**0.735**
real data results average						0.747	0.845	0.832	0.842	**0.852**	0.873	**0.890**
Synth. Data-sets	Data1	1000	10	50	50	0.718	0.821	0.816	1.000	1.000	1.000	1.000
	Data2	5000	10	250	1438	0.773	0.793	0.771	1.000	1.000	1.000	1.000
	Data3	1000	100	50	172	0.638	0.781	0.653	0.998	1.000	1.000	1.000
	Data4	5000	100	250	1424	0.545	0.543	0.668	0.999	1.000	1.000	1.000

dimensionality of this set is too large for FIB and OA. The **bold-faced** AUC indicates the best method(s) for a particular data set. The parameters in the compared algorithms are set as suggested, i.e., $\theta = 0.3$, $k = 5$ in CNB and $SupRate = 0.3$, $MaxItem = 5$ in FIB and OA.

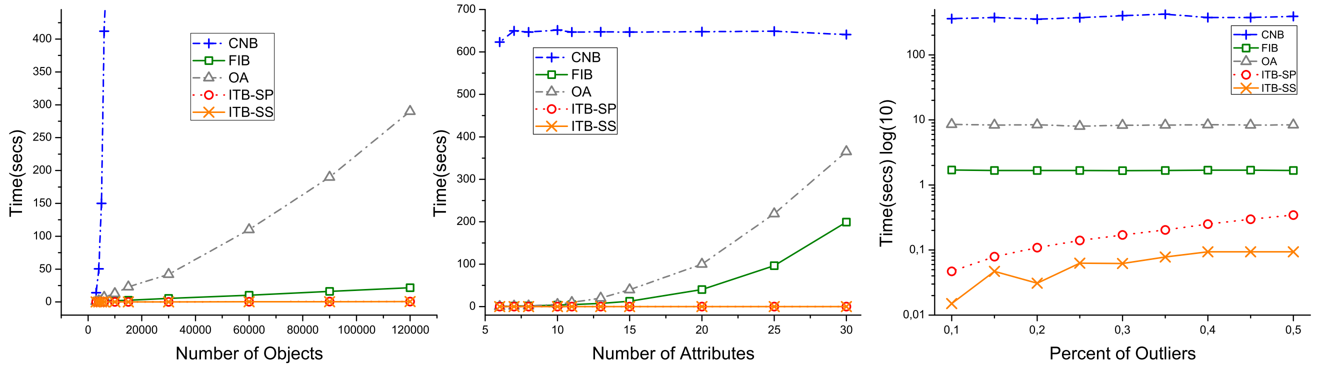
The results reported in Table 5 warrant a number of comments. First, between the weighted and unweighted versions of the proposed methods, the results in the last four columns of Table 5 show that the performance of the weighted version generally surpasses that of the unweighted version. These results are evidence of the importance of capturing attribute weights. Moreover, the Average line indicates that the improvement of ITB-SS over unweighted ITB-SS is much more significant than the improvement of ITB-SP over unweighted ITB-SP. This difference can be explained by the repeated weight updating in the ITB-SS method each time an outlier is detected and removed, whereas ITB-SP does not involve weight updating. We remark that the unweighted ITB-SP and the unweighted ITB-SS do outperform their weighted counterparts occasionally. This may be caused by the way "outliers" are determined and by nonrepresentative objects that do not allow reliable estimation of attribute weights.

Now, let us look at the comparison between our proposed methods and the compared methods. The results in Table 5 reveal that our proposed methods are more effective than CNB, FIB, and OA. The table shows that ITB-SS outperforms these methods on more than 70 percent of all data sets. The Average row of the AUC value also indicates that ITB-SS performs much better overall than the other methods, followed by ITB-SP, FIB, and OA. More importantly, ITB-SS is effective on the large data set *KDD* and on the high-dimensional data set *web-ad*.

In order to determine whether the differences in outlier detection accuracy are statistically significant, we perform a pairwise comparison. The results are presented in Table 6. Each cell in the table contains the number of data sets for which the method in the row, i.e., ITB-SP or ITB-SS, wins, loses, or ties relative to the corresponding method in the column, over the selected 21 data sets. For detecting ties (statistically similar results), we use a two-tailed T-Test [15] with a significance level of 0.005. The pairwise comparison shows that ITB-SP and ITB-SS are more accurate than the other methods on these data sets. ITB-SS outperforms every other method in at least 13 data sets, and underperforms in at most 4 of them. ITB-SP, although not as effective as ITB-SS, outperforms the other compared methods on at least 11 data sets and loses on at most 5 data sets.

TABLE 6
Results of Significance Test (Win/Lose/Tie)

	CNB	FIB	OA	ITB-SP	ITB-SS
ITB-SS	**18**/1/2	**16**/2/3	**17**/1/3	**13**/4/4	
ITB-SP	**14**/4/3	**11**/7/3	**11**/5/5		4/**13**/4



(a) Objects Increasing

(b) Attributes Increasing

(c) Percent of Outliers Increasing

Fig. 3. Results of efficiency test on synthetic data sets.

5.2.4 Test on Synthetic Data Sets

We also compare the effectiveness of different methods on synthetic data sets in a relatively ideal setting, since the generated outliers are usually more distinctive than those in real data and the outliers “truth” can be used to verify whether an outlier algorithm is able to find them. Four experiments are reported in the bottom part of Table 5,[3] where the outliers take up 5 percent of the corresponding data set. In fact, to generate each test set, the data generator [51] is first used to generate rule-based categorical data sets with 10 clusters. Then 95 percent of the objects of the test set are obtained by randomly choosing from three of the ten generated clusters. These are considered to be normal objects. On the other hand, 5 percent of objects are randomly chosen from the remaining clusters and are considered to be outliers.

The results in Table 5 and in our other nonreported experiments show that synthetic data sets are in general too easy for ITB-SS and ITB-SP, as they often achieve near-perfect results. In general, these experiments confirm that the performance of CNB, FIB, and OA is acceptable when the dimensionality of the data is not too high. Their performance declines quickly with an increasing number of dimensions. Increasing data size seems to hurt the performance of these methods too, but more extensive experiments are needed to draw a definitive conclusion.

5.3 Efficiency Test

To measure the time consumption with increasing numbers of objects, attributes and outliers, we employ GAClust [49] to generate synthetic data sets for these experiments. In the “objects increasing” test, the number of objects is increased from 3,000 to 120,000. In the “attributes increasing” test, the number of attributes increases from 6 to 30.[4] In the “percentage of outliers increasing” test, we assume the percentage of outliers in a data set is increased from 10 to 50 percent. The results are shown in Fig. 3. All of the compared methods were implemented with C++, and run on a desktop with Intel Core 2 Quad processor (clocked at 2.4 GHz) and 4 G memory.

As Fig. 3a indicates, the run times of ITB-SP, ITB-SS, and FIB are almost linear functions of the number of objects. FIB has a higher increase rate than ITB-SP and ITB-SS. From the theoretical analysis, we know that the time complexity of CNB [11] increases quadratically with the number of objects, which is confirmed by the experimental data of Fig. 3a. For the attributes increasing test, Fig. 3b shows that the run times of the FIB and OA increase rapidly with the number of attributes, which closely matches the theory that the time complexities of FIB [10] and OA [7] increase quadratically with the number of attributes. Compared with the time increase of FIB and OA, the increases for the other methods are too small to be noticeable on the figure. Fig. 3c illustrates the run time as a function of the percentage of “outliers” in the data set each method is asked to search for. The time axis is in the log(10) scale. The run times of CNB, OA, and FIB remain almost fixed with the “outlier percentage.” Those of ITB-SP and ITB-SS methods increase linearly, but remain much lower than those of other methods even for very high “outlier percentages.”

The three efficiency tests suggest ITB-SP and ITB-SS are efficient. They are particularly appropriate for large data sets with high dimensionality, and are also suitable for data sets with a high percentage of outliers. The CNB algorithm is not suitable for large data sets. The FIB and OA algorithms are not suitable for high-dimensional data sets, due to their high time complexities.

6 Conclusion

In this paper, we have formulated outlier detection as an optimization problem and proposed two practical, unsupervised, 1-parameter algorithms for detecting outliers in large-scale categorical data sets. The effectiveness of our algorithms results from a new concept of weighted holoentropy that considers both the data distribution and attribute correlation to measure the likelihood of outlier candidates, while the efficiency of our algorithms results from the outlier factor function derived from the holoentropy. The outlier factor of an object is solely determined by the object and its updating does not require estimating the data distribution. Based on this property, we apply the greedy approach to develop two efficient algorithms, ITB-SS

[3] Since FIB and OA have high time complexities with attributes and CNB is not able to deal with large data sets, we have set relatively small upper limits for the numbers of attributes and of objects, i.e., 100 and 5,000, respectively. Our algorithm is effective to deal with large-scale data sets, e.g., the KDD data set with 168,876 objects and the web advertisement data set with 1,558 attributes.

[4] To avoid the high time costs of FIB and OA, we set a relatively small upper limit on the number of attributes, i.e., 30 in this test.

and ITB-SP, that provide practical solutions to the optimization problem for outlier detection. We also estimate an upper bound for the number of outliers and an anomaly candidate set. This bound, obtained under a very reasonable hypothesis on the number of possible outliers, allows us to further reduce the search cost.

The proposed algorithms have been evaluated on real and synthetic data sets, and compared with different mainstream algorithms. First, our evaluations on a small real data set and a bundle of synthetic data sets show that the proposed algorithms do tend to optimize the selection of candidates as outliers. Moreover, our experiments on real and synthetic data sets in comparison with other algorithms confirm the effectiveness and efficiency of the proposed algorithms in practice. In particular, we show that both of our algorithms can deal with data sets with a large number of objects and attributes.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions allowing to improve the quality of this paper. The work has been funded by Discovery and Discovery Accelerator Supplements programs of Natural Sciences and Engineering Research Council of Canada (NSERC) granted to Prof. Shengrui Wang. Shu Wu has been partly supported by the PhD Scholarship program of the China Scholarship Council.

References

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM Computing Surveys,* vol. 41, no. 3, pp. 1-58, 2009.

[2] V.J. Hodge and J. Austin, "A Survey of Outlier Detection Methodologies," *Artificial Intelligence Rev.,* vol. 22, no. 2, pp. 85-126, 2004.

[3] E.M. Knorr and R.T. Ng, "Algorithms for Mining Distance-Based Outliers in Large Data Sets," *Proc. 24rd Int'l Conf. Very Large Data Bases (VLDB '98),* 1998.

[4] S.R. Gaddam, V.V. Phoha, and K.S. Balagani, "K-Means+ID3: A Novel Method for Supervised Anomaly Detection by Cascading K-Means Clustering and ID3 Decision Tree Learning Methods," *IEEE Trans. Knowledge and Data Eng.,* vol. 19, no. 3, pp. 345-354, Mar. 2007.

[5] D. Zhang, D. Gatica-Perez, S. Bengio, I. McCowan, "Semi-Supervised Adapted HMMs for Unusual Event Detection," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition (CVPR '05),* 2005.

[6] T. Cover and J. Thomas, *Elements of Information Theory.* John Wiley & Sons, 1991.

[7] M.E. Otey, A. Ghoting, and S. Parthasarathy, "Fast Distributed Outlier Detection in Mixed-Attribute Data Sets," *Data Mining and Knowledge Discovery,* vol. 12, pp. 203-228, 2006.

[8] K. Das and J. Schneider, "Detecting Anomalous Records in Categorical Data Sets," *Proc. 13th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '07),* 2007.

[9] K. Das, J. Schneider, and D.B. Neill, "Anomaly Pattern Detection in Categorical Data Sets," *Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '08),* 2008.

[10] Z. He, X. Xu, Z.J. Huang, and S. Deng, "FP-Outlier: Frequent Pattern Based Outlier Detection," *Computer Science and Information Systems,* vol. 2, pp. 103-118, 2005.

[11] S. Li, R. Lee, and S. Lang, "Mining Distance-Based Outliers from Categorical Data," *Proc. IEEE Seventh Int'l Conf. Data Mining Workshops (ICDM '07),* 2007.

[12] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules Between Sets of Items in Large Databases," *Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '93),* 1993.

[13] C.C. Aggarwal and P.S. Yu, "Outlier Detection for High Dimensional Data," *Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '01),* 2001.

[14] X. Wang and I. Davidson, "Discovering Contexts and Contextual Outliers Using Random Walks in Graphs," *Proc. IEEE Ninth Int'l Conf. Data Mining (ICDM '09),* 2009.

[15] T.G. Dietterich, "Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms," *Neural Computation,* vol. 10, no. 7, pp. 1895-1923, 1998.

[16] S. Srinivasa, "A Review on Multivariate Mutual Information," Univ. of Notre Dame, Notre Dame, Indiana, vol. 2, pp. 1-6, 2005.

[17] S. Watanabe, "Information Theoretical Analysis of Multivariate Correlation," *IBM J. Research and Development,* vol. 4, pp. 66-82, 1960.

[18] L. Wei, W. Qian, A. Zhou, W. Jin, and J.X. Yu, "HOT: Hypergraph-Based Outlier Test for Categorical Data," *Proc. Seventh Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD '03),* 2003.

[19] M. Breunig, H-P. Kriegel, R. Ng, and J. Sander, "LOF: Identifying Density-Based Local Outliers," *Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '00),* 2000.

[20] P.K. Chan, M.V. Mahoney, and M.H. Arshad, "A Machine Learning Approach to Anomaly Detection," technical report, Florida Inst. of Technology, 2003.

[21] M. Fox, G. Gramajo, A. Koufakou, and M. Georgiopoulos, "Detecting Outliers in Categorical Data Sets Using Non-Derivable Itemsets," Technical Report, The AMALTHEA REU Program, 2008.

[22] J. Han and M. Kamber, *Data Mining—Concepts and Techniques.* Elsevier, 2006.

[23] Z. He, X. Xu, and S. Deng, "An Optimization Model for Outlier Detection in Categorical Data," *Proc. Int'l Conf. Advances in Intelligent Computing (ICIC '05),* 2005.

[24] S. Papadimitriou, H. Kitagawa, P.B. Gibbons, and C. Faloutsos, "Loci: Fast Outlier Detection Using Thelocal Correlation Integral," *Proc. 19th Int'l Conf. Data Eng. (ICDE '03),* 2003.

[25] J. Takeuchi and K. Yamanishi, "A Unifying Framework for Detecting Outliers and Change Points from Time Series," *IEEE Trans. Knowledge and Data Eng.,* vol. 18, no. 4, pp. 482-492, Apr. 2006.

[26] G.D. Battista, P. Eades, R. Tamassia, and I.G. Tollis, "Algorithms for Drawing Graphs: An Annotated Bibliography," *Computational Geometry: Theory and Applications,* vol. 4, pp. 235 282, 1994.

[27] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection for Discrete Sequences: A Survey," *IEEE Trans. Knowledge and Data Eng.,* vol. 24, no. 5, pp. 823-839, May 2012.

[28] T. Leckie and A. Yasinsac, "Metadata for Anomaly-Based Security Protocol Attack Deduction," *IEEE Trans. Knowledge and Data Eng.,* vol. 16, no. 9, pp. 1157-1168, Sept. 2004.

[29] X. Song, M. Wu, C. Jermaine, and S. Ranka, "Conditional Anomaly Detection," *IEEE Trans. Knowledge and Data Eng.,* vol. 19, no. 5, pp. 631-645, May 2007.

[30] F. Angiulli, S. Basta, and C. Pizzuti, "Distance-Based Detection and Prediction of Outliers," *IEEE Trans. Knowledge and Data Eng.,* vol. 18, no. 2, pp. 145-160, Feb. 2006.

[31] F. Angiulli and C. Pizzuti, "Outlier Mining in Large High-Dimensional Data Sets," *IEEE Trans. Knowledge and Data Eng.,* vol. 17, no. 2, pp. 203-215, Feb. 2005.

[32] S.-d. Lin and H. Chalupsky, "Discovering and Explaining Abnormal Nodes in Semantic Graphs," *IEEE Trans. Knowledge and Data Eng.,* vol. 20, no. 8, pp. 1039-1052, Aug. 2008.

[33] S.D. Bay and M. Schwabacher, "Mining Distance-Based Outliers in Near Linear Time with Randomization and a Simple Pruning Rule," *Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '03),* 2003.

[34] H.D.K. Moonesignhe and P. Tan, "Outlier Detection Using Random Walks," *Proc. IEEE 18th Int'l Conf. Tools with Artificial Intelligence (ICTAI '06),* 2006.

[35] J.X. Yu, W. Qian, H. Lu, and A. Zhou, "Finding Centric Local Outliers in Categorical/Numerical Spaces," *Knowledge and Information Systems,* vol. 9, no. 3, pp 309-338, 2006.

[36] W. Lee and D. Xiang, "Information-Theoretic Measures for Anomaly Detection," *Proc. IEEE Symp. Security and Privacy,* 2001.

[37] Z. He, X. Xu, and S. Deng, "Discovering Cluster-Based Local Outliers," *Pattern Recognition Letters,* vol. 24, pp. 1641-1650, 2003.

[38] D.M.J. Tax and R.P.W. Duin, "Support Vector Domain Description," *Pattern Recognition Letters,* vol. 20, nos. 11-13, pp. 1191-1199, 1999.

[39] B. Scholkopf, J.C. Platt, J.S. Taylor, A.J. Smola, and R.C. Williamson, "Estimating the Support of a High-Dimensional Distribution," *Neural Computation,* vol. 13, no. 7, pp. 1443-1471, 2001.

[40] M. Filippone and G. Sanguinetti, "Information Theoretic Novelty Detection," *Pattern Recognition,* vol. 43, pp. 805-814, 2010.

[41] L. Itti and P. Baldi, "Bayesian Surprise Attracts Human Attention," *Proc. Neural Information Processing Systems Conf. (NIPS '05),* 2005.

[42] D. Barbará, C. Domeniconi, and J.P. Rogers, "Detecting Outliers Using Transduction and Statistical Testing," *Proc. 12th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '06),* 2006.

[43] W. Jin, A.K.T. Tung, and J. Han, "Mining Top-n Local Outliers in Large Databases," *Proc. Seventh ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '01),* 2001.

[44] W. Jin, A.K.T. Tung, J. Han, and W. Wang, "Ranking Outlier Using Symmetric Neighborhood Relationship," *Proc. 10th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD '06),* 2006.

[45] E. Aleskerov, B. Freisleben, and B. Rao Cardwatch, "A Neural Network Based Database Mining System for Credit Card Fraud Detection," *Proc. IEEE/IAFE Computational Intelligence for Financial Eng. Conf. (CIFEr '97),* 1997.

[46] J. Gao, H. Cheng, and P.N. Tan, "Semi-Supervised Outlier Detection," *Proc. ACM Symp. Applied Computing (SAC '06),* 2006.

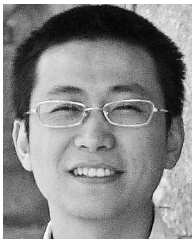
[47] H.P. Kriegel, P. Kroger, and A. Zimek, "Outlier Detection Techniques," *Proc. ACM Symp. Applied Computing (SDM '10),* 2010.

[48] J. Dougherty, R. Kohavi, and M. Sahami, "Supervised and Unsupervised Discretization of Continuous Features," *Proc. Int'l Conf. Machine Learning (ICML '05),* 2005.

[49] http://www.cs.umb.edu/dana/GAClust/index.html, 2012.

[50] UCI Machine Learning Repository, http://www.ics.uci.edu/mlearn/MLRepository.html, 2011.

[51] http://www.data setgenerator.com/, 2011.



Shu Wu received the BS degree from Hunan University, China, in 2004, the MS degree from Xiamen University, China, in 2007, and the PhD degree from the University of Sherbrooke, Canada, in 2012, all in computer science. He is an assistant professor in the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences. His research interests include data mining, recommendation systems, and pervasive computing. He is a member of the IEEE.

Shengrui Wang received the BS degree in mathematics from Hebei University, China, in 1982, the MS degree in applied mathematics from the Université de Grenoble in 1986, and the PhD degree from the Institut National Polytechnique de Grenoble, France, in 1989. He is a professor in the Department of Computer Science at the University of Sherbrooke, Canada. In 1990, he worked as a postdoctoral fellow in the Department of Electrical Engineering at Laval University, Canada. His research interests include data mining, pattern recognition, machine learning, neural networks, bioinformatics, social networks and Web, recommendation systems, image processing, image databases, geographical information systems, and navigation systems. He is a member of the IEEE.