Efficient Outlier Detection Using Graph Based Semi Supervised Clustering with BAT Algorithm

J. Rajeswari and Dr.R. Gunasundari

Abstract--- *Outlier detection is a fundamental issue in data mining, specifically it has been used to detect and remove anomalous objects from data. It is an extremely important task in a wide variety of application domains. The existing research method named as Expectation Maximization Particle Swarm Optimization Weighted Clustering (EMPWC) outlier detection technique is used for detecting the outliers more efficiently. However it has issue with handling the unbalanced dataset and time complexity. To avoid the above mentioned issues, in this research, BAT optimization based semi supervised algorithm is proposed. It has three phases such as pre-processing, outlier estimation using BAT algorithm and clustering using semi-supervised algorithm. The pre-processing is done by using min-max normalization approach which is sued to increase the outlier detection accuracy. The outlier detection is improved by using BAT optimization algorithm which is achieved by best objective function. The clustering is done by Graph based Semi Supervised (GSS) algorithm. The GSS based BAT (GAABAT) approach is used to improve the performance metrics such as execution time and false alarm rate compare than existing methods. The experiment results on large scale categorical datasets have shown that the IBAT with semi supervised based outlier detection ensures a better trade-off between Detection Rate (DR), False Alarm Rate (FAR) than the existing outlier detection schemes.*

Index Terms--- *Clustering, Outlier Data, BAT Algorithm and GSS Algorithm.*

I. INTRODUCTION

The problem of outlier detection has been receiving a lot of attention in recent years due to its significance in dealing with various real life problems. Outliers are frequently treated as noise that needs to be removed from a dataset in order for a specific model or algorithm to succeed. Outliers are patterns in data that do not conform to a normal behavior, or conform to an outlying behavior. Outliers exist in almost every real data set. Outliers can be directly related to the real life behavior, this is the key feature of outlier detection. The identification of outlier can lead to the discovery of useful and meaningful knowledge. The applications of outlier detections are intrusion detection, fraud detection, insurance claim fraud detection, insider training detection, medicine outlier detection, public health outlier detection, industrial damage detection [1].

To handle the outlier data, clustering approaches are introduced. Clustering is a prominent task in mining data, which group related objects into a cluster. The clustering based techniques involve a clustering step which partitions the data into groups which contain similar objects. The use of this particular type of clustering methods is motivated by the unbalanced distribution of outliers versus “normal” cases in these data sets. By definition, outliers are rare occurrences, thus representing a very low fraction of the total examples. Outlier detection over data stream is active research area from data mining that aims to detect object which have different behaviour, exceptional than normal object [2] [3].

The density-based approach of classic outlier finds the density allocation of the information and identifies outliers as those present in low-density region. Breunig et al. [4] allocate a local outlier factor (LOF) to every point based on

J. Rajeswari, Research Scholar, Karpagam University, Karpagam Academy of Higher Education, Coimbatore. E-mail: rajeswarikrishna82@gmail.com
Dr.R. Gunasundari, Head, Dept of Information Technology, Karpagam University, Karpagam Academy of Higher Education, Coimbatore.

the neighboring density of its environs, which is firm by a user-given least amount of points (MinPts). Papadimitriou et al. [5] present Local Correlation Integral (LOCI) which uses numerical values based on the data itself to deal with the problem of selecting values for minimum points. Density based methods can understand with a solitary, universal principle. However it has issue with sparse data. Cluster based outlier detection methods are very useful in result analysis.

Rodrigues, et.al used a new incremental algorithm for clustering streaming of time series. The Online Divisive Agglomerative Clustering (ODAC) system is continuously maintains a tree-like hierarchy of clusters that evolves with data. The system is designed and planed to process number of data streams that flow at high-rate. The system main features include update memory and time consumption that does not depend on the examples of number of stream. Moreover, the time and memory required to process in lower, whenever the cluster structure expands. It proves on real and artificial data assess the system processing qualities, signifying the competitive performance on data clustering time series and also its deal with concept drift.

In [6] introduce a new approach of semi supervised anomaly detection that deals with categorical data. Given a training set of instances (all belonging to the normal class) analyze the relationship among features for the extraction of a discriminative characterization of the anomalous instances. The key idea is to build a model that characterizes the features of the normal instances and then use a set of distance based techniques for the discrimination between the normal and the anomalous instances. In [7] semi supervised method which uses normal instances to build an ensemble of feature classification models and identifies instances that disagree with those models as anomalous. It is not specifically tailored on categorical data, but it can adopt any clustering algorithms that work well on each specific feature type. But, these methods are not effective for the case of reasonably very large dimensional dataset and imbalanced datasets. Hence the overall system performance is reduced significantly.

To overcome the above mentioned issues, this research presents a new outlier factor function is derived from the weighted entropy and shows that calculation/updating of the outlier factor can be carried out without the necessity for estimating the joint probability distribution. Here the BAT optimization is inclusive of n number of data samples N which move around a D-dimensional search space for the optimization of a particular variational property. Moreover, an upper bound of the outliers for reducing the search space is also estimated. Based on the best objective function values, the outliers are estimated for the given dataset. The proposed performance of GSSBAT is measured with regard to the DR, FAR, time comparison between the numbers of attributes, number of data objects, Normalized Mean Square Error (NMSE) for the comparison of error results, Area Under the Curve (AUC). It indicates that the proposed GSSBAT have lesser NMSE error, FAR, and more Detection Rate (DR) with lesser time consumed for completing the process.

II. RELATED WORK

In [8] presented a novel mutual reinforcement based local outlier detection approach. Instead of detecting local outliers as noise, it attempt to identify local outliers in center, which are similar with some clusters of objects on the one hand, and are unique on the other hand. This technique can be used for bank investment to identify a unique body, similar with many good competitors, to invest. This research attempts to detect local outliers in categorical, ordinal as well as numerical data. In categorical data, the challenging is that there are many similar but different ways to specify relationships among data items. The mutual-reinforcement based approach is stable with similar but different user-defined relationships. The technique can reduce the burden for users to determine the relationships among data items. However it has issue with handling the high dimensional data.

In [9] suggested that most approaches to date have focused on detecting outliers in a continuous attribute space. However, almost all real-world data sets contain a mixture of categorical and continuous attributes. Categorical attributes are typically ignored or incorrectly modeled by existing approaches, resulting in a significant loss of information. Second, there have not been any general-purpose distributed outlier detection algorithms. Most distributed detection algorithms are designed with a specific domain (e.g. sensor networks) in mind. Third, the data sets being analyzed may be streaming or otherwise dynamic in nature. Such data sets are prone to concept drift and models of the data must be dynamic as well. To address these challenges, presents a tunable algorithm for distributed outlier detection in dynamic mixed-attribute data sets. It has higher detection accuracy.

Optimization model [10] is introduced for outlier detection that contains a formal definition of the outliers, through a concept of holoentropy which considers both entropy and total correlation. From this model a function for the outlier factor of object is decided by the object itself and can be updated in an efficient manner. In this technical work, two practical 1-parameter outlier detection techniques, which are the ITB ITB-SS (Information-Theory-Based Step-by-Step) and ITB-SP (Single-Pass) techniques are presented that need no user-defined parameters for determining if an object is an outlier. The proposed ITB-SS and ITB-SP methods have more effectiveness when compared to mainstream techniques. But these methods dealing with both large and high-dimensional data sets become difficult task.

For many data mining and machine learning applications predicting minority class samples from skewed unbalanced data sets is a crucial problem. To address this problem, it introduced a majority filter-based minority prediction (MFMP) approach for unbalanced datasets. The MFMP adopts an unsupervised learning technique for selecting samples for supervised learning. The approach is based on two steps. In the first-step, minority samples are clustered and majority class samples that are out of minority classification regions are identified. This improves minority prediction rate. In the second step majority samples are randomly selected in individual clusters and this enhances majority prediction rate [11].

In [12] introduced a generalization of the k-means problem with the aim of simultaneously clustering data and discovering outliers. A naive approach is to apply the k-means algorithm and list as outliers the top points that are the furthest away from their nearest cluster centers. In [13] weighted k-mean algorithm is used for weighting attributes based on their relevance, which helps in reduce the effect of the irrelevant and noisy attributes. For outlier detection it is hard to collect labeled data and also data is coming increasingly and data points may be asynchronous. So it is better to use unsupervised outlier detection where no need of class labels of data objects.

In [14] Particle Swarm Optimization (PSO) based approach to outlier detection can then be applied, which expands the scope of PSO and enables new insights into outlier detection. PSO is used to automatically optimize the key distance measures instead of manually setting the distance parameters via trial and error, which is inefficient and often ineffective. The novel PSO approach is examined and compared with a commonly used detection method, Local Outlier Factor (LOF), on five real data sets. The results show that the new PSO method significantly outperforms the LOF methods for correctly detecting the outliers on the majority of the datasets and that the new PSO method is more efficient than the LOF method on the datasets tested.

As a novel feature, bat algorithm is based on the echolocation features of microbats [15], and bat algorithm uses a frequency-tuning technique to increase the diversity of the solutions in the population, while at the same, it uses the automatic zooming to try to balance exploration and exploitation during the search process by mimicking the variations of pulse emission rates and loudness of bats when searching for prey. It proves to be very efficient with a typical quick start. It is used to increase the optimal results for outlier dataset in this research.

III. PROPOSED METHODOLOGY

In the proposed system, the categorical, numerical and mixed data are evaluated by using the GSSBAT algorithm more effectively. In this section, the means by which entropy, Shannon, Jensen-Shannon Divergence (JSD) and total correlation could be exploited for capturing the likelihood of outlier candidates is looked at. In this research, the unbalanced dataset is handled and the outliers are detected optimally.

A. Preprocessing

The preprocessing is a very important step since it can improve the result of a clustering algorithm. This module calculates tuples with missing values using different options like maximum, minimum, constant, average and standard deviation for the treatment of missing values tuples before applying normalization approach on the dataset. This process gives the treatment of missing value data and then it applies to data normalization of data preparation. Normalization is used to standardize all the features of the dataset into a specified predefined criterion so that redundant or noisy objects can be eliminated and use made of valid and reliable data which can improve accuracy of the result. Data normalization is an essential step to prevent larger features from randomized value to the specific range. The importance of normalization is that it enhances the accuracy of the results that are obtained during clustering [16]. For the unbalanced dataset, the min-max normalization method is used for identifying the missing values effectively. Min-Max normalization is a simple technique where the technique can specifically fit the data in a pre-defined boundary.

It performs a linear transformation on the original data. Min-max normalization maps a value $d\ of\ P\ to\ d'$ in the range $[new\_min(p), new\_max(p)]$. The min-max normalization is calculated by the following formula:

$$d' = \frac{[d - \min(p)] * [new_{\max(p)} - new_{\min(p)}]}{[\max(p) - \min(p)]} \quad (1)$$

Where $\min(p) = minimum\ value\ of\ attribute$

$\max(p) = maximum\ value\ of\ attribute$

By using the above formula, the missing values are identified efficiently. it is used to list out the outliers from the given dataset.

This work considered outlier detection as highly overlapped unbalanced data clustering problem, where original samples heavily outnumber the outlier samples. Usually, the clustering algorithms display poor performance while dealing with unbalanced datasets and results bias towards majority class. For this type of problems, the time, error and cost associated with outlier sample is predicted. An unbalanced data set is defined as a data set, in which one class of data severely outnumbers the other class of data. For the prediction of the class of a data record in data set, classification method can be used. It learns the model from already labeled historical data. It uses learned models for predicting the class of unseen or unknown data.

This research Synthetic Minority Over-sampling Technique (SMOTE) on the entire class samples, outlier regions may not be well defined, because of sparsely located outlier samples. The blind generation of synthetic samples along the sparse samples, resulting a greater chance of class mix. Hence the minority samples may not recognize well. To avoid this class mix in training data distribution, this research use extreme outlier elimination from the minority class by using k Nearest Neighbor (kNN) concept as a data cleaning method. The kNNs cardinality value represents whether the point located in a sparse region or in a dense region. It is called the points that are very sparsely located are extreme outliers. By eliminating the extreme outliers, it is ignoring the points that are far from the minority decision boundary for doing SMOTE [17].

B. Measurement for Outlier Detection

Assume the data be referred to as the Xcomprising the number of the data objects as $\mathrm{n}(x_1, \dots. x_n)$every$x_i$ for

$1 < i < n$ being a vector of the categorical attributes $[y_1, y_2, \ldots., y_m]^T$, where $m$refers to the number of categorical and discrete data attributes,y_j indicates the value of the attribute which belongs to either a categorical and discrete value given by $(y_{1,j}, y_{2,j}, \ldots. y_{n,j})(1 < j < m)$and n_j refers to the number of unique values in attribute y_j. For the purpose of measuring the attribute value importance, the Shannon, JSD is applied and the holoentropy of the attribute is indicated as$H_x()$, mutual information [18] $I_x()$, and total correlation $C_x()$ is calculated on the set X; e.g., $I_x(y_i, y_j)$stands for the mutual information between the attributes y_i and y_j. The holoentropy $H_X(Y)$ can be expressed as below:

$$\begin{aligned} H_x(y) &= H_x(y_1, y_2, \ldots y_m) \\ &= \sum_{i=1}^{m} H_x(y_i | y_{i-1}, \ldots. y_1) \\ &= H_x(y_1) \\ &+ \cdots. + H_x(y_m | y_{m-1}, \ldots y_1) \end{aligned} \quad (2)$$

$$\begin{aligned} H_x(y_m | y_{m-1}, \ldots. y_1) \\ = -\sum_{y_m, \ldots. y_1} p(y_m, y_{m-1}, \ldots y_1) \\ logp(y_m | y_{m-1}, \ldots y_1) \end{aligned} \quad (3)$$

The total correlation is defined to be the summation of mutual information of multivariate discrete random vectors Y, represented as $C_x(Y)$

$$C_x(y) = \sum_{i=2}^{m} \sum_{\{r_{-1}, \ldots r_i\} \subset \{1, \ldots m\}} I_x(y_{r_1}, \ldots. y_{r_i})$$

$$= \sum_{\{r_1, \ldots r_i\} \subset \{1, \ldots m\}} I_x(y_{r_1}, ; y_{r_2}) + \cdots + I_x(y_{r_1}, \ldots y_{r_m}) \quad (4)$$

Where $r_1 \ldots r_i$ refer to attribute numbers selected from 1 $tom.$ $I_x(y_{r_{1,}\ldots.} y_{r_i}) = I_x(y_{r_{1,}\ldots.} y_{r_{i-1}}) - I_x(y_{r_{1,}\ldots.} y_{r_i})$ indicates the multivariate mutual information of $y_{r_1} \ldots y_{r_i}$, where $I_x(y_{r_1}, \ldots y_{r_{i-1}} | y_i) = E(I(y_{r_1}, \ldots. y_{r_{i-1}}) | y_{r_i})$stands for the conditional mutual information. The holoentropy$HL_x(Y)$ is defined to be the summation of the entropy and the total correlation of the random vector Y, and can be formulated by the summation of the entropies on all the attributes

$$HL_x(Y) = H_x(Y) + C_x(Y) = \sum_{i=1}^{m} H_x(y_i) \quad (5)$$

Holoentropy allocates same importance to every attribute, whereas in real applications. This problem is resolved by the proposed weighting technique which calculates the weights from the data directly and is influence by the rise in the efficiency in practical applications compared to theoretical requirements.

$$w_x(y_i) = 2\left(1 - \frac{1}{1 + \exp(-H_x(y_i))}\right) \quad (6)$$

Even though in the holoentropy function thus sets a minimum value for each attributes and the maximum anticipated number of attributes value are detected in the Shannon and JSD.

Shannon entropy: Shannon entropy is one among the most essential metrics in information theory. Entropy does the measurement of the uncertainty which is associated with a random variable.

$$H(X) = \sum_{i=1}^{n} p(x_i)\, I(x_i) = \sum_{i=1}^{n} p(x_i)\, log_b \frac{1}{p(x_i)} \quad (7)$$

Jensen-Shannon Divergence (JSD) gives the mean relative entropy between two distributions and the distribution mean [20].

$$\begin{aligned} JS(y_i | y_j) &= \frac{1}{2}\sum_i P(y_i) \ln \frac{P(y_i)}{\frac{1}{2}\left(P(y_i) + P(y_j)\right)} \\ &+ \frac{1}{2}\sum_i P(y_j) \ln \frac{P(y_j)}{\frac{1}{2}\left(P(y_i) + P(y_j)\right)} \end{aligned} \quad (8)$$

$$\begin{aligned} &= \frac{1}{2} D(y_i || M) + \frac{1}{2} D(y_j || M) \\ &= S(M) - \frac{1}{2} S(y_i) - \frac{1}{2} S(y_j) \end{aligned} \quad (9)$$

The equation (9) provides the probability computation formula of every firefly for a set of data given.

$$p(y_j) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{V_n} \varphi\left(\frac{y_i - y_m}{h_n}\right) \quad (10)$$

Where $\varphi(x)$stands for the window function and n refers to the total number of data objects, V_n and $h_n$are the respective volume and edge length of a hypercube. When the JSD is computed, then the weights are computed from the data directly and is influenced by the increase in the effectiveness

$$w_x(y_i) = 2\left(1 - \frac{1}{1 + exp\left(\mathrm{JS}\left(y_i \middle| y_j\right)\right)}\right) \quad (11)$$

The weighted holoentropy of random vector $W_X\ (Y)$ is defined to be the summation of the weighted entropy on every attribute of the random vector Y.

$$W_X(y) = \sum_{i=1}^{m} w_x(y_i) H_X(y_i) \quad (12)$$

Provided a data set X with n objects and the number o, a subset Out(o) is defined to be the set of outliers in case it reduces J_X (Y; o), defined as the weighted holoentropy of X with o objects eliminated

$$J_X(Y, O) = W_{X \backslash set\,(O)}(Y) \quad (13)$$

Where set (O) refers to any subset of o objects from X.

$$Out(O) = argminJ_X(Y, O) \quad (14)$$

Therefore, the formulation of the outlier detection is now expressed to be an optimization problem. For a provided o, the number of probable candidate sets for the objective function is $C_n^O = \frac{n!}{O!(n-O)!}$, that is very huge. In addition, one may have to decide the optimal value of O, i.e., the number of outliers a data set actually has. A probable theoretical approach to this issue is searching for a range of values of $O$and then deciding over an optimal value of Othrough the optimization of a particular variational property of $J_X(Y, O)$. Assume this as a direction proposed in this research work. At present, focus will be on the development of practical solutions for the optimization issue.

C. BAT Optimization Algorithm

Bat Algorithm is inspired by echolocation characteristic of bats. Echolocation is typical sonar which bats use to detect prey and to avoid obstacles. These bats emit very loud sound and listen for the echo that bounces back from the surrounding objects [20]. Thus a bat can compute how far they are from an object. Furthermore bats can distinguish the difference between an obstacle and a prey even in complete darkness [21].

Any bat flies randomly with velocity $V_i = [v_{i1} \ldots \ldots v_{id}]$ and pulse frequency $freq_i \in [freq_{min}, freq_{max}]$ at $P_i = [p_{i1} \ldots\ldots p_{id}]$, varying the rate of pulse emission $rate_i \in [rate_{min}, rate_{max}]$ and loudness $L_i \in [L_{min}, L_{max}]$

If the food is closer, the $rate_i$ will be bigger and the L_i will be lower.

The rules of the updating of a bat are shown as follows

$$freq_i = freq_{min} + (freq_{max} - freq_{min})\rho_i \quad (15)$$
$$V_i = V_i + (P_i - P_*) freq_i \quad (16)$$
$$P_i = P_i + V_i \quad (17)$$

Where $\rho_i \in [0,1]$ is a random value according to a uniform distribution. P_* is the current global best solution among all the n bats, which is chosen through the fitness function $F(P_i)$. The pseudo code of BA can be summarized as follows

```
Input: The fitness function $F$
Output: The best solution
Method
Initialize the bat population $P_i (i = 1, 2, \ldots n)$ and $V_i$
Define pulse frequency $rate_i$ at $P_i$
Initialize pulse rates $rate_i$ and the loudness $L_i$
For 1→$iter$
 Generate new solutions by adjusting frequency, and updating velocities and positions by using (15) (16) (17)
If $rand < rate_i$ then
Select a solution among the best solutions
```

```
Generate a position around the selected best one
End if
Generate a new solution by flying randomly
If $rand < L_i \& F(P_i) > F(P_*)$
Accept the new solutions
Increase $rate_i$ and reduce $L_i$
End if
Rank the bats and find the current best $P_*$
End for
Return $P_*$
```

BAT consist of n number of data samples N which move around a d-dimensional search space for the optimization of a particular variational property of $J_X(Y, O)$. The process of BAT starts with a population that contains a number of the data objects as n$(x_1, \ldots. x_n)$with every$x_i$with $r_1 \ldots r_i$refer to the attribute numbers selected from $1\ to\ m$ for each data sample and the optimization appropriately next searches for the best range of values for O through continuously updating the generations. The location of the ith data samples of cluster bats can be referred to by $\mathrm{l} = (l_1, , .. \mathrm{l}_\mathrm{j})$. The velocity with respect to the ith cluster of data points can be represented as $\mathrm{v_i} = (\mathrm{v_{i1}}, \mathrm{v_{i2}}, \ldots, \mathrm{v_{iD}})$. The velocities corresponding to the data points in the cluster are restricted within$[\mathrm{V_{min}}, \mathrm{V_{max}}]^{\mathrm{D}}$ respectively. The frequency rules are updated and global best solution is selected among several bats. By using the best fitness function value, the best solution is ranked.

At each generation, the position and the velocity of each i^{th} data points in the cluster gets revised by current best flies in bats. It occurs in the space which data point's discrete problem, with the intent of resolving this issue, BAT which is used for discrete binary variables. In binary space, a bat which is a data point in the cluster probably will move to the nearly corners of a hypercube through the flipping of multiple numbers of bits; as a result, the bat velocity on the whole may be defined by means of the number of bits modified according to the number of processes. In each step, the pulse rate and loudness is updated using best bat values. Hence the outlier in the given dataset is recognized more optimally.

In this research ρ refers to the random value for the optimization of a particular dataset samples and random numbers within$(0, 1)$. Velocities V_i and $F(P_i)$ refer to the current best and old velocities for outliers. P_* indicates the global current best position, and Increase $rate_i$ and reduce L_i, updated outlier detection position. In Equation (15) (16), outlier detection position velocities and frequency of each dataset sample are tried to be at a maximum velocity and maximum frequency. Thus it is more suitable for the outlier dataset compare than preceding algorithms.

D. *Graph Based Semi-Supervised Clustering with BAT Algorithm (GSSBAT)*

In this research, to improve the clustering result, use the graph-based semi-supervised clustering algorithm which uses the Gaussian random field to do semi supervised learning. In which the mean of the field is characterized in terms of harmonic functions. The method has two main steps: construct the graph and classification. It describes the detail of the two steps of the algorithm:

Let $\chi = \{x_1, \ldots\ldots x_l, x_{l+1}, \ldots x_n\}$ represents a set of n microarray data objects. The first l points $x_i \in X(i \leq l)$ are labeled and the remaining points $x_u \in X(l + 1 \leq u \leq n)$ are unlabeled. And y is the class label set.

In the graph-based semi-supervised learning algorithms described in [12], the method first defines an undirected graph W on the whole data set. In the graph W, the nodes are data instances in the graph and the edges are the strength of two data instances in the graph. Then the method constructs a k nearest neighbors graph with the Gaussian function of Euclidean distance to weight the edges [23].

$$W_{ij} = \begin{cases} \exp\ \left(-\frac{(x_i - x_j)^T (x_i + x_j)}{\sigma^2}\right) & \text{if } i \sim j \\ 0 & otherwise \end{cases} \quad (18)$$

Where $i{\sim}j$ denotes that node i and j has an edge between them. Then we can denote the graph as the following

$$W = \left(\begin{cases} W_{u} & W_{lu} \\ W_{ul} & W_{uu} \end{cases}\right)$$

Where W_{ll} denotes the weight of the edges between two labeled microarray data instances and Wuu denotes the weight of the edges between two unlabeled microarray data instances in the graph. W_{uu} denotes the weights of the edges from the labeled microarray points to the unlabeled microarray points in the graph and W_{ul} denotes the weights of the edges from the unlabeled microarray points to the microarray labeled points in the graph. And all the weights in W are weighted by w_{ij}.

With the graph constructed above, the graph-based methods can be viewed as estimating a function f on the graph W. f is a real-value class assign matrix which assign the class labels. In graph-based semi supervised learning, the f should satisfy two things: (1) the value of f should be close with the class labels of the labeled data samples then the regularizer in the graph can be expressed as the following equation

$$\textstyle\sum_{i\in L}(f_i - y_i)^2 \qquad (19)$$

and then the f should satisfy the second condition (20) and the f should be smooth enough on the whole graph. Then the smooth enough graph with the regularizer can be denote as the following

$$\tfrac{1}{2}\textstyle\sum_{ij} w_{ij}\ (f_i - y_i)^2 \qquad (20)$$

Then the clustering problem of the graph-based semi-supervised learning can be viewed as the combination of the above two regularizer

$$\textstyle\sum_{i\in L}(f_i - y_i)^2 + \tfrac{1}{2}\sum_{ij} w_{ij}\ (f_i - y_i)^2 = (f - y)^T(f - y) + \tfrac{1}{2} f^T \Delta f \qquad (21)$$

In which Δ is called the graph which is used to indicate the outliers more effectively

$$f = \begin{pmatrix} f_l \\ f_u \end{pmatrix} \qquad (22)$$

and then the f are expressed as

$$f = Pf \qquad (23)$$

Where the $P = D^{-1}W$, f_u defines the label of the unlabeled microarray data examples and f_l defines the values of the labeled examples.

$$f_u = (I - P_{uu})^{-1} P_{ul} f_l \qquad (24)$$

Where I the identity matrix and the clustering results is are get from the f_u. The outliers are optimally detected by using GSSBAT approach. For the mixed dataset, the outliers are detected which is sued to increase the clustering result than the previous approaches.

IV. Experimental Results

This section carries out the efficiency and effectiveness tests for analyzing the performance of the novel GSSBAT technique. In order to test effectiveness, the result is compared with the available techniques such as Information-Theory-Based Step-by-Step(ITB-SS) and Information-Theory-Based Single-Pass (ITB-SP) for artificial data sets. For the test of efficiency, evaluations on are conducted over artificial data sets to indicate how execution time sees an increase with the number of objects, attributes and the outliers. A huge number of public actual data sets, many of them obtained from UCI [23], are utilized in these experiments, indicating an extensive range of fields in science and the humanities. The data set utilized is the public, categorical “soybean data”, having 47 objects and 35 attributes. This data has a very smaller class of 10 objects. As the data does not have outliers that are explicitly identified, it is obvious to have the objects of the smallest class treated as “outliers”. The Area Under the Curve (AUC) results of various techniques and the characteristics of all the test data sets, like the numbers of objects (#n), attributes (#m) and outliers (#o), and the upper bound on outliers (#UO), are tabulated in the upper portion of Table 1.The results that are reported in Table 1 suggest a number of comments. These results are proof of the significance of acquiring attribute weights; it is then compared with the available techniques EMPWC, AMCEM, ITB-SS, ITB-SP with and without weighting. Frequent Pattern Outlier Factor (FIB), Common-neighbor-based distance (CNB)

Table 1: AUC Results of Tested Algorithms on the Real Dataset

Dataset	CNB	FIB	ITB-SP	ITB-SS	AMCEM	EMPWC	GSSBAT
Breast-c	0.99	0.90	0.991	0.993	0.996	0.997	0.998
Credit-a	0.84	0.92	0.985	0.992	0.995	0.996	0.997
Diabetes	0.86	0.88	0.75	0.912	0.945	0.945	0.957
Ecoli	0.89	0.92	0.96	0.99	0.996	0.998	0.999

The time consumption with rising numbers of objects, attributes and outliers is measured.

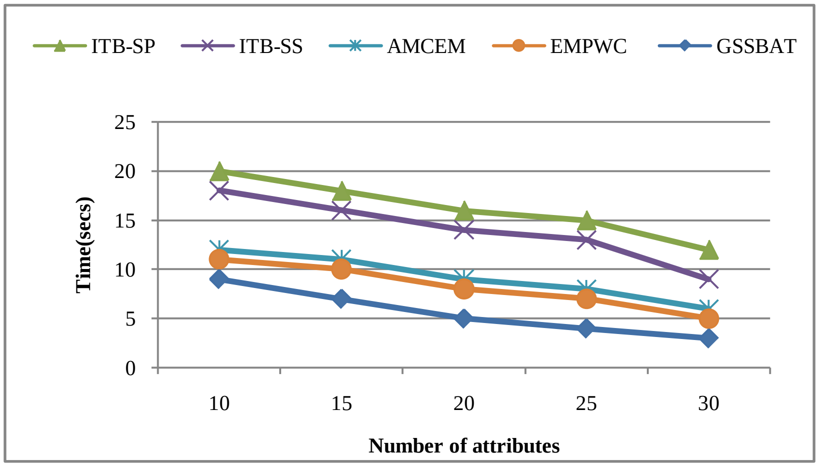

Figure 1: Results of Efficiency Real Data Sets for Data Attributes vs Methods

Like the Figure. 1 show, the execution times of GSSBAT, EMPWC, AMCEM, ITB-SP, ITB-SS, and FIB are nearly linear functions of the number of objects. The proposed GSSBAT has lesser rate in comparison with the other existing system. The result concludes that the proposed GSSBAT, increase quickly with the number of attributes that is average of 5.6 milliseconds. In comparison with the increase in time of FIB, CNB, ITB-SS, ITB-SP, AMCEM and EMPWC the increase in time for the other techniques are barely detectable

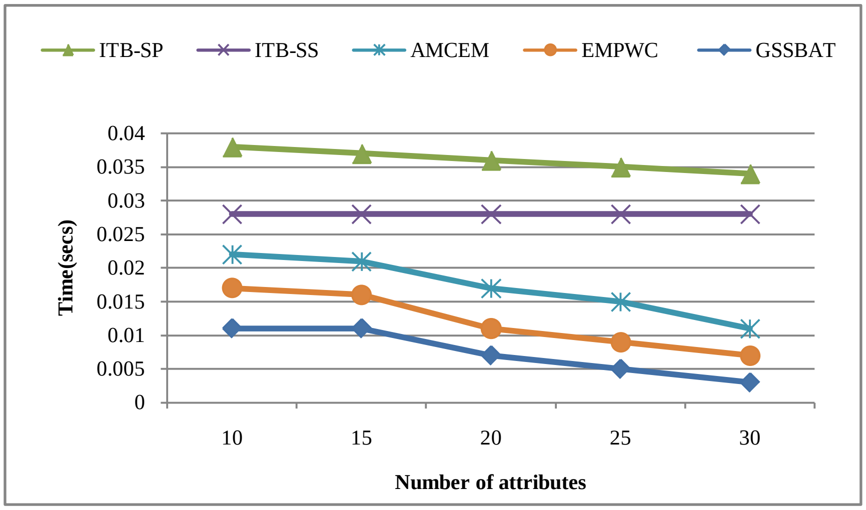

Figure 2: Results of Efficiency Real Data Sets for Percentage of the Outliers vs Methods

Figure 2 shows the execution time in the form of a function of the percentage of “outliers” in the data set every method is supposed to find. The time axis is in terms of the log (10) scale. For the percentage of the outliers test, Figure 2 illustrates that the execution time comparison of all clustering methods, from the results it concludes that the proposed GSSBAT, increase quickly performs with the percentage of the outliers that is average of 0.0074 milliseconds , when compared with the other techniques ITB-SP, ITB-SS, AMCEM and EMPWC. However the execution time comparison results of other clustering algorithms are 0.0526 ms, 0.0206 ms, 0.0098 ms and 0.0046 ms higher for ITB-SP, ITB-SS, AMCEM and EMPWC methods respectively.

The proposed GSSBAT stays much lower when compared to other clustering techniques even in the case of very huge “outlier percentages.”. The Normalized Root Mean Square Error (NRMSE) is defined as

$$\mathrm{NRMSE} = \frac{\sqrt{\mathrm{Mean}[(\mathrm{y}_{\mathrm{guess}} - \mathrm{y}_{\mathrm{ans}})^2]}}{\mathrm{std}[\mathrm{y}_{\mathrm{ans}}]} \qquad (25)$$

where y_{guess} and y_{ans} refer to the vectors whose elements indicate the estimated values and the known answer values correspondingly, for all the data objects in the cluster s. The mean and the standard deviation are computed over the outlier data in the whole matrix.

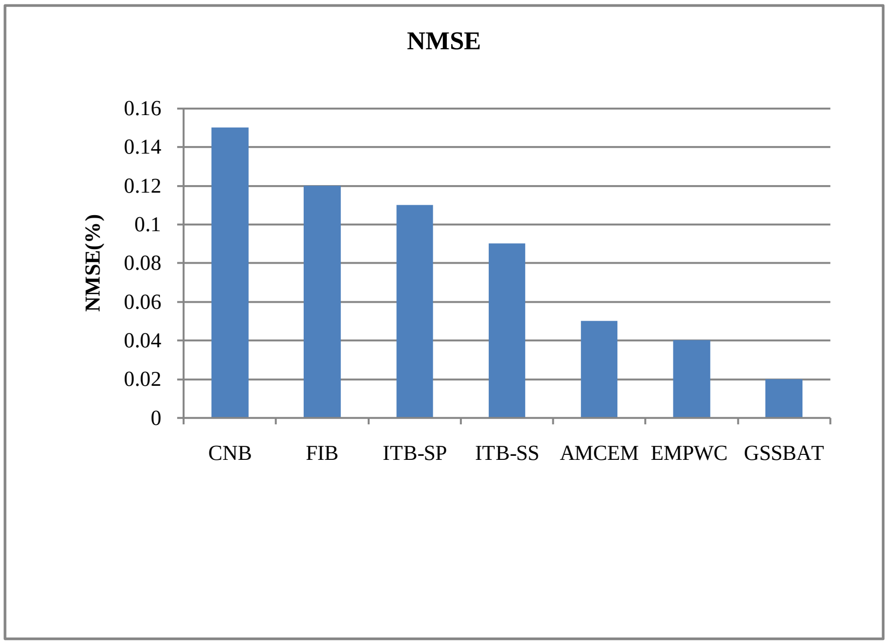

Figure 3: NMSE for Real Datasets vs Methods

In Figure 3 illustrates the results of the performance comparison of the NMSE for the clustering techniques like CNB, FIB, ITB-SP, ITB-SS, EMPWC and the proposed GSSBAT algorithm. The proposed GSSBAT algorithm have obtains lesser NMSE value of 0.02, which is 0.04,0.05,0.09,0.11, 0.12 and 0.15 for EMPWC, AMCEM, ITB-SS,ITB-SP, FIB, and CNB methods respectively.

Correct Detection Rate (CDR), that is the number of outliers identified correctly by every approach as outliers:

$$\mathrm{CDR} = \frac{\text{No of outliers correctly detected as outlie}}{Totalnoofoutlierindataset} \qquad (26)$$

False Alarm (FA) rate, that reflects the number of normal points identified erroneously to be outliers

$$\mathrm{FA} = \frac{\text{No of outliers incorrectly detected as outl}}{\text{Total no of normal points in dataset}} \qquad (27)$$

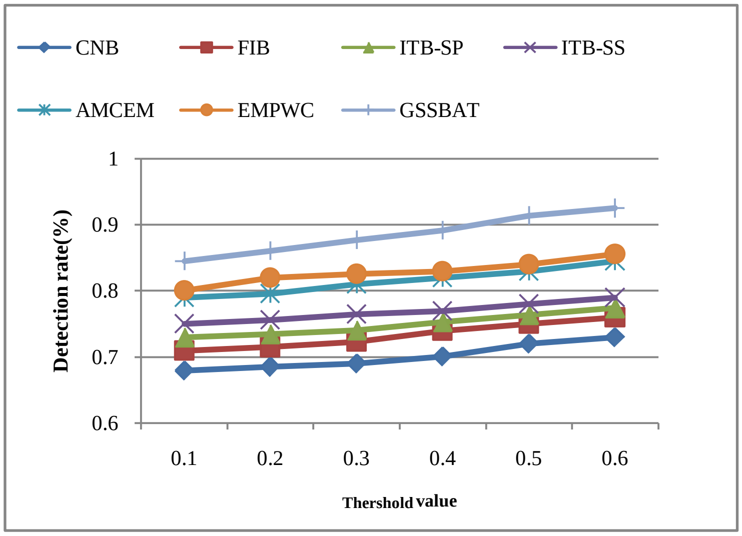

Figure 4: Detection Rate(DR) for Real Data Sets vs. Methods

In Figure 4 is shown the results of the performance comparison of the outlier Detection Rate (DR) for the available techniques like CNB, FIB, ITB-SP, ITB-SS, EMPWC and proposed GSSBAT algorithm. The proposed GSSBAT algorithm have obtains higher DR value of 0.8856667, which is 0.0568333, 0.0705, 0.1171667, 0.1361667, 0.1526667 and 0.1846667 higher when compared to other EMPWC, AMCEM, ITB-SS,ITB-SP, FIB, and CNB methods respectively.

DR value of the proposed GSSBAT algorithm has more DR in comparison with the other existing techniques.

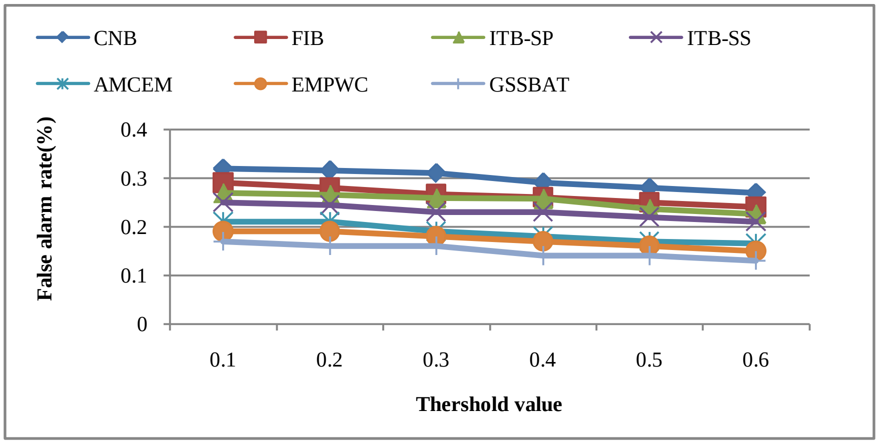

Figure 5: False Alarm Rate(FAR) for Real Datasets vs Methods

Figure 5 illustrates the results of performance comparison in terms of the False Alarm Rate (FAR) for the traditional techniques like CNB, FIB, ITB-SP, ITB-SS, EMPWC and proposed GSSBAT algorithm. FAR value of the SAVF algorithm proposed have less FAR compared to available techniques.

V. Conclusion and Future Work

In this research, the efficient outlier detection methods are proposed to improve the dataset accuracy. The efficiency of the proposed GSSBAT outlier detection technique needs attribute frequency based results from a novel concept of weighted entropy optimization which takes both the data Shannon and Jensen-Shannon Divergence (JSD) into consideration for measuring the likelihood of outlier candidates, whereas the effectiveness of algorithms proposed is a result from the outlier factor function obtained from the entropy. In this research, the important phases are preprocessing, outlier detection and clustering. The preprocessing is done with the help of min-max normalization approach. It is more useful for dealing with the missing values and hence the clustering performance is progressed than the previous method. Then the unbalanced dataset problem is handled by using SMOTE with kNN approach which is focused to increase the dataset efficiency. Apply the BAT optimization algorithm for optimizing the outlier attributes. By computing the optimal fitness function, the outlier attributes are identified more effectively. In this research work, the range values of outliers (o) get optimized employing the BAT algorithm. An upper bound for the number of outliers along with an anomaly candidate set is estimated. This bound, derived under a moderate hypothesis on the number of probable outliers, allows further reducing the cost of search. It is also used to reduce the computational complexity rather than the existing method. On the basis of this BAT technique, the data clustering results are seen to increase and therefore the algorithm proves to be greatly efficient. Then use the GSS based clustering algorithm for accurate clustering. It is focused to carry out on a small actual data set and a bundle of artificial data sets indicate that the algorithms do proposed attempt to have the optimization of the selection of candidates to be outliers. The result proves that the proposed GSSBAT approach has superior performance in terms execution, NMSE time and false alarm rate than the previous approaches. In future, the kernel based outlier detection method can be developed for distributed mixed arbitrary-type data sets.

REFERENCES

[1] A. Koufakou, E.G. Ortiz, M. Georgiopoulos, G.C. Anagnostopoulos and K.M. Reynolds, “A scalable and efficient outlier detection strategy for categorical data”, IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007), Vol. 2, Pp. 210-217, 2007.

[2] A. Loureiro, L. Torgo and C. Soares, “Outlier detection using clustering methods: a data cleaning application”, Proceedings of KDNet Symposium on Knowledge-based Systems for the Public Sector, Bonn, Germany, 2004.

[3] P.P. Rodrigues, J. Gama and J. Pedroso, “Hierarchical clustering of time-series data streams”, IEEE transactions on knowledge and data engineering, Vol. 20, No. 5, Pp. 615-627, 2008.

[4] M.M. Breunig, H.P. Kriegel, and R.T. Ng, “LOF: Identifying density- based local outliers”, ACM Conference Proceedings, Pp. 93-104, 2000.

[5] S. Papadimitriou, H. Kitagawa, P.B. Gibbons and C. Faloutsos, “Loci: Fast outlier detection using the local correlation integral”, Proceedings of 19th International Conference on Data Engineering, Pp. 315-326, 2003.

[6] D. Ienco, R.G. Pensa and R. Meo, “A Semisupervised Approach to the Detection and Characterization of Outliers in Categorical Data”, IEEE Transactions on Neural Networks and Learning Systems, Pp. 1-13, 2016.

[7] K. Noto, C. Brodley and D. Slonim, “FRaC: A feature-modeling approach for semi-supervised and unsupervised anomaly detection”, Data Mining Knowl. Discovery, Vol. 25, No. 1, Pp. 109–133, 2012.

[8] 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.

[9] M.E. Otey, A. Ghoting and S. Parthasarathy, “Fast distributed outlier detection in mixed-attribute data sets”, Data Mining and Knowledge Discovery, Vol. 12, No. 2-3, Pp. 203-228, 2006.

[10] S. Wu and S. Wang, “Information-theoretic outlier detection for large-scale categorical data,” IEEE Transactions on Knowledge and Data Engineering, Vol. 25, No. 3, Pp. 589-602, 2013.

[11] Padmaja, T. Maruthi, P. Radha Krishna and Raju S. Bapi, “Majority filter-based minority prediction (MFMP): An approach for unbalanced datasets”, TENCON 2008-2008 IEEE Region 10 Conference, Pp. 1-6, 2008.

[12] S. Chawla, and A. Gionis, “k-means-: A Unified Approach to Clustering and Outlier Detection”, SDM, Pp. 189-197, 2013.

[13] Y. Thakran and D. Toshniwal, “Unsupervised Outlier Detection in Streaming Data Using Weighted Clustering”, 12th International Conference on Intelligent Systems Design and Applications (ISDA), Vol. 6, Pp. 947-952, 2012.

[14] Mohemmed, W. Ammar, M. Zhang, and W.N. Browne, “Particle swarm optimisation for outlier detection”, Proceedings of the 12th annual conference on Genetic and evolutionary computation, 2010.

[15] X.S. Yang, “A New Metaheuristic Bat-Inspired Algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NISCO 2010) (Eds. Cruz, C.; Gonz´alez, J. R.; Pelta, D. A.; Terrazas, G)”, Studies in Computational Intelligence, Vol. 284, Springer Berlin, Pp. 65–74, 2010.

[16] Y.K. Jain and S.K. Bhandare, “Min max normalization based data perturbation method for privacy protection”, International Journal of Computer & Communication Technology (IJCCT), Vol. 2, No. 8, Pp.45-50, 2011.

[17] T.M. Padmaja, N. Dhulipalla, R.S. Bapi and P.R. Krishna, “Unbalanced data classification using extreme outlier elimination and sampling techniques for fraud detection”, International Conference on Advanced Computing and Communications, Pp. 511-516, 2007.

[18] S. Srinivasa, “A Review on Multivariate Mutual Information”, Univ. of Notre Dame, Notre Dame, Indiana, Vol. 2, Pp. 1-6, 2005.

[19] S. Bano and K. Rao, “Partial context similarity of gene/proteins in leukemia using context rank based hierarchical clustering algorithm”, International Journal of Electrical and Computer Engineering, Vol.5, No.3, Pp. 483-490, 2015.

[20] X.S. Yang, “A new metaheuristic bat-inspired algorithm”, Nature inspired cooperative strategies for optimization (NICSO 2010), Pp. 65-74, 2010.

[21] R. Nakamura, L. Pereira, K. Costa, D. Rodrigues, J. Papa and X.S. Yang, “BBA: A Binary Bat Algorithm for Feature Selection”, Proc. 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Pp. 291-297, 2012.

[22] X. Zhu, Z. Ghahramani and J. Lafferty, “Semisupervised learning using gaussian fields and harmonic functions”, ICML, Vol. 3, Pp. 912-919, 2003.

[23] The datasets are provided from website Available from: http://www.ics.uci.edu/learn/MLRepository.html, 2011.