International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 01 | Jan -2017 www.irjet.net p-ISSN: 2395-0072

Unsupervised Distance Based Detection of Outliers by using Anti-hubs

Mr. Kiran V. Markad, Mr. Kiran M. Moholkar , Mr. Sopan N. Abdal *Department of Information Technology D Y PATIL College of Engineering, Ambi Talegaon , Pune, India*

Prof. Rajani Thite*AssistantProfessor*

Department of Information Technology D Y PATIL College of Engineering, Ambi Talegaon , Pune, India

---

Abstract - Distance based outlier detection methods fails as the dimensionality of the data increases due to all point becomes good outlier. Reason of these issues is irrelevant and redundant features; nearest neighbor of the point P is K points whose distance to point P is less than all other points. Reverse nearest neighbors (RNN) of Point P is the points for which P is in their k nearest neighbor list. Some points are frequently comes in k-nearest neighbor list of another points and some points are infrequently comes in k nearest neighbor list of different points are called as Anti-hubs. There are many researches of ant hub based unsupervised outlier detection but there is one issue is occurring that high computation cost for finding anti hubs. If the data that having better dimensionality, high computation cost, high computation complexity, time requirement to find anti hubs. To remove the unwanted feature and high dimensionality data to make an efficient system results. Feature selection is the process proposed to remove unwanted feature and make a system more efficient. By applying feature selection this paper extends anti-hub based outlier detection method for high dimensional data.

Key Words **:- High-Dimensional, Detection of Data Outliers, Reverse nearest Neighbours.**

1.INTRODUCTION

There are three main types of outlier detection methods namely, unsupervised, semi-supervised and supervised. There is need to find outlier in many application for that we have to study outlier detection analysis. There is Need of availability of correct labels of the instances for Supervised and Semi Supervised outlier detection. Currently unsupervised technique is used widely which does not need label to the instances for outlier detection.

Currently the best efficient method for outlier detection is unsupervised distance base outlier detection method. The normal instances have small amount of distances among them and outliers have large amount of distances among them in distance base outlier detection. As the increase in dimensions of data distances are not useful to find outliers causes every point become an outlier. Therefore, high dimensionality of data is the biggest problem or challenge for unsupervised distance outlier detection. The base paper can show that unsupervised distance base outlier detection system can handle high dimensional data, it can detect outlier under specific condition i.e. data should be useful and attributes are meaningful means data should not be noisy then it can be success to handle high dimensional data. K-nearest neighbor (KNN) of the point P is K points whose distance to point P is less than all other points. Reverse nearest neighbors (RNN) of Point P is the points for which P is in their k nearest neighbor list. Some points are frequently comes in k-nearest neighbor list of other points and some points are infrequently comes in k nearest neighbor list of some other points are called as Anti-hubs.

For outlier detection RNN concept is used in literature, but there is no theoretical proof which explores the relation between the Outlier natures of the points and reverses nearest neighbors. The reverse nearest count is get affected as the dimensionality of the data increases, so there is need to investigate how outlier detection methods bases on RNN get affected by the dimensionality of the data.

For outlier detection RNN concept is used in literature [2] [4], but there is no theoretical proof which explores the relation between the Outlier natures of the points and reverse nearest neighbors. Paper [6] states that reverse nearest count is get affected as the dimensionality of the data increases, so there is need to investigate how outlier detection methods bases on RNN get affected by the dimensionality of the data.

In existing system it takes large computation cost, time to calculate the reverse nearest neighbors of the all points. Use of Antihubs for outlier detection is of high computational task. Computation complexity increases with the data dimensionality. For this there is scope to removal of irrelevant features before application of Reverse Nearest Neighbor. So to overcome this problem, feature selection is applied on the data. In this step, all features are rank according to their importance and required features are selected for finding reverse nearest neighbors. To find reverse nearest neighbor using Euclidean distance and outlier score is calculated by using technique from existing system. According to studies, if system does not know about the distribution of the data then Euclidean distance is the best choice. Proposed scheme deals with curse of dimensionality efficiently. We discussed existing system, problem statement and proposed scheme with detailed structure and algorithms.

2. Literature Survey

Hawkins et al [1] the quality of data has problem observed in OBIS data binding processes is discussed. DBSCAN a density based clustering algorithm for large spatial database is employed to identify outliers. The algorithm is to be effective and efficient for this purpose. The relationship between outliers and erroneous data points are discussed and the future scope to develop an operational data quality checking tool based on this algorithm is discussed.

V. Hautamaki et al [2] used reverse nearest neighbor count is to score outlier nature of the point. User defined threshold was used to take decision about outlier nature of the point. Method proposed in this paper is named as Outlier Detection using in degree Number (ODIN). If score is less than threshold then the point is said to be an outlier otherwise it is normal point. The link between the reverse nearest neighbor count and outlier nature of the point investigate by this paper.

J. Lin et al [3] were special case of ODIN [2] where point was considered outlier if reverse nearest neighbor count of the point is zero. In this paper does not provide any mathematical explanation or proof why point which has reverses nearest neighbor count is outlier. They mainly focused on the speed and scalability.

The method to find reverses nearest neighbor of the point in metric spaces described by Y. Tao et al [4]. Proposed algorithms do not necessitate representation of the instances i.e. objects. Proposed technique uses metric index therefore it affirms by recurring to the insertion/deletion operations of the index.

C. Lijun et al [5] explored the relation between outlier and RNN but there was no re-search study how high dimensionality was connected with reverse nearest neighbors. They focused on data stream application and reducing execution time for finding reverse nearest neighbor of point.

Outlier detection was the process of discovering observations which noticeably deviates from other observations and also it was a fundamental approach in data analysis task described by Gustavo H. Orair et al [6]. Applications range from financial fraud detection to clinical diagnosis of diseases and network intrusion detection. They described and evaluated several distance based outlier detection approaches. They presented the study to understand the impact of optimizations strategies and tried to consolidate them.

K. S. Beyer et al [7] tried to finding the effective answers for the problem of nearest neighbor. This problem is specified [7] as, finding the data point that was closest to the query point by giving an aggregation of points of data and a query point in a multidimensional metric space. They analyzed the effects of dimensionality on Nearest Neighbor queries. They observed that as there is increase in the dimensionality, the distance to the neighbor advances to the distance to the farthest neighbor. Conducted the experiments to find out the proportion at which the NN breaks down and also explored the situations where even on dimensionality NN queries do not break down.

Many current systems uses data mining based methods or the methods that are based on signature which depends on labeled data i.e. supervised training data for the purpose of intrusion detection described by E. Eskin et al [8]. As such systems can only detects the intrusions based on previously identified intrusions, there was a risk of attack until new type of attack has been manually revised. To train the model which will detect the attacks, anomaly detection algorithms based on supervised learning needs purely normal data, might contain some intrusions. Algorithm may not be able to identify and predict the future instances of such intrusions as it was considered as normal. To address the problem, they proposed a geometric model for anomaly detection founded on unsupervised method. For the detection of the outlier, paper proposed three algorithms. First algorithm was based on cluster based technique, second was based on k-nearest neighbor and third was based on support vector machine based algorithm.

The context of outlier detection, in this approach assigned the each object with the level of being an outlier and this assigned level i.e. degree of the object was called as Local Outlier Factor (LOF) explored by H.P.Kriegel et al [9]. In this approach, they used density based clustering for finding outliers in multidimensional datasets. Local Outlier factor means instead of considering an outlier as a dual dimension, assign object a level to which it was kept apart from the around neighbors.

Hans-Peter Kriegel et al [10] describe distance based approaches. This distance based approaches degrades in performance due to high dimensionality of the data [10]. As they rely on the distances curse of dimensionality arises the performance issues. Identification of Density

Hermine N. Akouemo [11] this paper proposed the combination of two statistical techniques for the detection and imputation of outliers in time series data. An autoregressive integrated moving average with exogenous inputs (ARIMAX) model is used to extract the characteristics of the time series and to find the residuals. The outliers are detected by performing hypothesis testing on the residuals and the anomalous data are imputed using another ARIMAX model. This paper tests the algorithm using both synthetic and real data sets and we present the analysis and comments on those results.

Discoursed issues in outlier detection in the case of eminent data dimensionality and showed the way outlier detection in high dimensional data can be made using unsupervised methods describe In paper [12] Milos Radovanovic et al. It also enquires how Anti-hubs are associated to the point's outlier nature.

Milos Radovanovic et al [13] discusses problems in outlier detection in high dimensionality and shows that how unsupervised methods can be used for outlier detection in high dimensional data 2) investigates how Anti-hubs are related to outlier nature of the point. Based on the relation anti-hubs and outlier two methods are proposed for outlier detection for high and low dimensional data.

2.1 Proposed System

1. **Feature Selection**

To deal with the Curse of dimensionality proposed system is designed. It takes high computation cost, time to calculate the reverse nearest neighbors of the all points in existing system. Feature selection is applied on the data to overcome this problem. In this step, all features are rank according to their importance and required features are selected for finding reverse nearest neighbors. Importance of the feature is calculated using the Mutual Information (MI) measure. Mutual Information is one most important feature which calculates the mutual dependence between two features.

$$MI(A, B) = \sum_a \sum_b P_{AB}(a, b) \log \frac{P_{AB}(a,b)}{P_A(a)P_B(b)} \quad \ldots\ldots\ldots\ldots (1)$$

The mutual information between feature A and feature B calculated by Equation 1 where P_B (b).P_A (a) is marginal probability distribution and P_{AB} (a, b) is joint probability distribution. To calculate the MI of A, sum of MI of A with all other features is taken,

$$MI(A) = \sum_{i=0}^{N} (MI(A, i)) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

After calculation of MI values of all features, features with MI values less than threshold values are discarded from further process.

2. Find Reverse nearest Neighbor

In this step, data of selected features will be considered for finding the reverse nearest neighbor. To determine the reverse nearest neighbor, first k-nearest neighbors of each point is evaluated. Existing system used Euclidian measure for calculating the distance between two instances. Euclidian distance measure

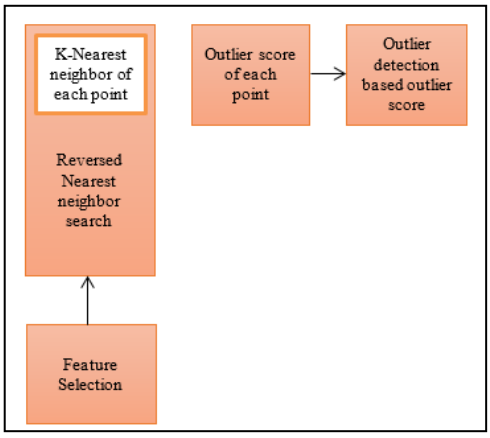

Fig. 1. System Architecture

Works fine for two and three dimensional data but is gets negatively affected with high dimensionality. According to studies, if system doesn't know about the distribution of the data then Euclidean distance is the best choice. Number of occurrences of point P in the k nearest neighbor list of the all other points is called as k-occurrence. Points in the dataset for which point's P is k-nearest neighbor are reverse nearest neighbor for point P. From the k-nearest neighbor list of each point, reverse nearest neighbor list of each point is calculated.

3. Outlier Score of Each Point

Previous methods than existing system considered k-occurrence of the point as an outlier score. Less k-occurrence indicates more outlier score of the point. Proposed system will follow existing system to calculate the outlier score of the point. Sum of k-occurrence score of k-nearest neighbors of the point P is outlier score of the point P.

Outlier Score (P) = $P_{i=0}^{K}$(koccurence(pi))

Where pi is the i^{th} nearest point of point P If Outlier scores (P) is larger than the threshold then Point P is considered as outlier.

3 Implementation details

A. Algorithm

Algorithm AntiHub 2 with feature selection

It works under the following stages

1: Select features

2: Computation of mutual dependence of two random variable using equation

$$MI(A, B) = \sum_a \sum_b P_{AB}(a, b) \log \frac{P_{AB}(a,b)}{P_A(a)P_B(b)} \quad \ldots\ldots (1)$$

Equation 1 calculates the mutual information between feature A and feature B. where P_B(b).P_A(a) is marginal probability distribution and P_{AB}(a,b) is joint probability distribution

3: Then MI of one feature with all other features is computed using the relation:

MIft = MIftj..... (2)

Where i,j=1,2,.....ft with i!=j and ft is total number of features

4: Then MI of each feature is use to rank the feature

5: a = AntiHubdist (D,k)

6: For each i (1,2, ...,n)

7: anni =$\sum_j$_ NN_{dist}(k, i) a_j where NN_{dist} (k, i) is the set of indices of k nearest neighbors of x_i

8: disc=0

9: For each α (0, step, 2* step ....1)

10: For each i _ (1, 2 ... n)

11: ct_i = (1-α).a_i + α.ann_i

12: cdisc = discScore (ct, p)

13: If cdisc > disc

14: t = ct, disc = cdisc

15. For each i _ (1, 2... n)

16. s_i= f(t_i) where f : R⟶ R is a monotone function

B. Mathematical Model

Let, S be Anti hub based fast unsupervised outlier detection scheme having Input, Processes and Output it can be represented as,

S = (I, P, O)

Where, I, is a set of inputs given to the System, O is a set of outputs given by the System,

P is a set of processes in the System.

I = (I1, I2, I3, I4)

I1- is set of input data D with m number of features with n number of instances.

I2- k for knn

I3- Mutual Information threshold

I4- Outlier score threshold

P= (P1, P2, P3, P4, P5, P6, P7)

P1- Find the Mutual Information between two random variables A and B

MI (A, B) = $\sum_a \sum_b P_{AB}(a,b) \log \frac{P_{AB}(a,b)}{P_{AB}(a)P_B(b)}$(1)

Where

PA (a) is marginal probability distribution and

PAB (a; b) is joint probability distribution

Output will be O1

P2 - Find Mutual Information of Feature

MI (A) = $\sum_{i=0}^{N}$ (MI (A, i)) (2)

Features with high MI than threshold MI is selected for farther process

If MI (Ai) ≥Threshold MI

Then Select Ai

Else discard Ai

P3 - To find the distance between two instances Euclidean distance is used

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_i - q_i)^2 + \cdots + (p_n - q_n)^2}.$$

Where d (p,q) is Euclidean distance between p and q points , both points has n dimensions

Output will be O3

P4 - Find k-nearest neighbor of each point

Knn (P) = (p1, p2, p3, p4, .pk)

List of k nearest neighbor points is calculated.

P5 - Find RNN list of each point

RNN (P) = Set of points for which P is in their knn list

P6 - Outlier score of each point

Outlier Score (P) = $sum_{i=0}^{k}$ (koccurrence (pi))

Where k indicates k nearest neighbors of point p

P7 - Outlier detection

If Outlier Score (P) > threshold then P is outlier

O1 - List of MI of among all features in D

O2 - List of selected features

O3 - Euclidean distance

O4 - List of list of knn points for each point

O5 - List of RNN of each point is calculated

O6 - List of outlier score of each point

Let, S be Anti hub based fast unsupervised outlier detection scheme having Input, Processes and Output it can be represented as,
S = (I, P, O)
Where, I, is a set of inputs given to the System, O is a set of outputs given by the System,
P is a set of processes in the System.
I = (I1, I2, I3, I4)
I1- is set of input data D with m number of features with n number of instances.
I2- k for knn
I3- Mutual Information threshold
I4- Outlier score threshold
P= (P1, P2, P3, P4, P5, P6, P7)
P1- Find the Mutual Information between two random variables A and B
MI (A, B) = $\sum_a \sum_b P_{AB}(a,b) \log \frac{P_{AB}(a,b)}{P_{AB}(a)P_B(b)}$(1)
Where
PA (a) is marginal probability distribution and
PAB (a; b) is joint probability distribution
Output will be O1
P2 - Find Mutual Information of Feature
MI (A) = $\sum_{i=0}^{N}$ (MI (A, i)) (2)

Features with high MI than threshold MI is selected for farther process
If MI (Ai) ≥Threshold MI
Then Select Ai
Else discard Ai
P3 - To find the distance between two instances Euclidean distance is used

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_i - q_i)^2 + \cdots + (p_n - q_n)^2}.$$

Where d (p,q) is Euclidean distance between p and q points , both points has n dimensions
Output will be O3
P4 - Find k-nearest neighbor of each point
Knn (P) = (p1, p2, p3, p4, .pk)
List of k nearest neighbor points is calculated.
P5 - Find RNN list of each point
RNN (P) = Set of points for which P is in their knn list
P6 - Outlier score of each point
Outlier Score (P) = $sum_{i=0}^{k}$ (koccurrence (pi))
Where k indicates k nearest neighbors of point p
P7 - Outlier detection
If Outlier Score (P) > threshold then P is outlier
O1 - List of MI of among all features in D
O2 - List of selected features
O3 - Euclidean distance
O4 - List of list of knn points for each point
O5 - List of RNN of each point is calculated
O6 - List of outlier score of each point

C Experimental Setup

The scheme is implemented using Java framework (version jdk 1.8) on Windows platform. The Net bean IDE (version 8.0.2) is applied as a development tool. The scheme doesn't need any particular hardware to run; any standard machine can be able to run the application.

4. Experimental Results

The reason of the conducting experiments is to check the effect of feature selection before anti-hub based outlier detection on high dimensional data. To see the effectiveness accuracy, memory and time requirement of Antihub based outlier detection i.e. Antihub2 [13] and Proposed method is compared. For experiment purpose, we used KDD dataset. Dataset contains 1050 instances, 42 attributes and 1.456% outliers. Minor class category considered as outlier class. Table 1 shows the actual results.

TABLE1.ACCURACY COMPARISON WITH K VARIATION

K	Simple Anithub 2	Antihub 2 with feature selection
10	84.79 %	93.40%
100	84.70 %	90.34 %
500	83.37 %	88.04 %

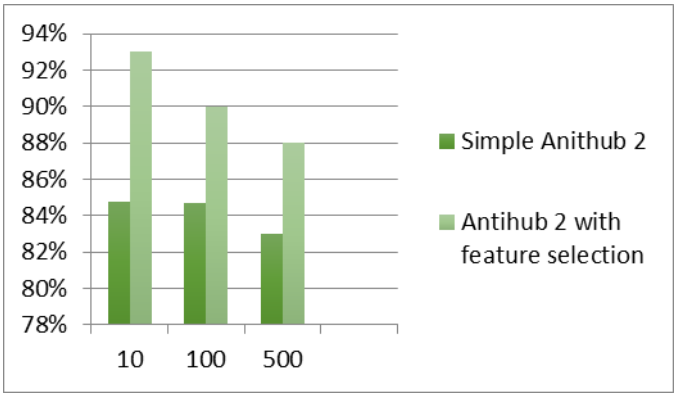

Fig. 2. Accuracy comparison with k variation

Table2.time comparison with k variation

K	Simple Anithub 2 time in sec.	Antihub 2 with feature selection time in sec.
10	322	310
100	324	314
500	335	324

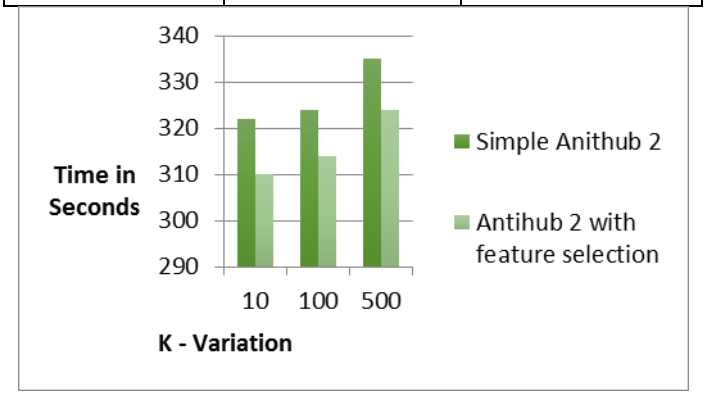

Fig. 3. time comparison with k variation

Table3.memory comparison with k variation

K	Simple Anithub 2 memory in MB	Antihub 2 with feature selection memory in MB
10	25	19
100	37	26
500	48	45

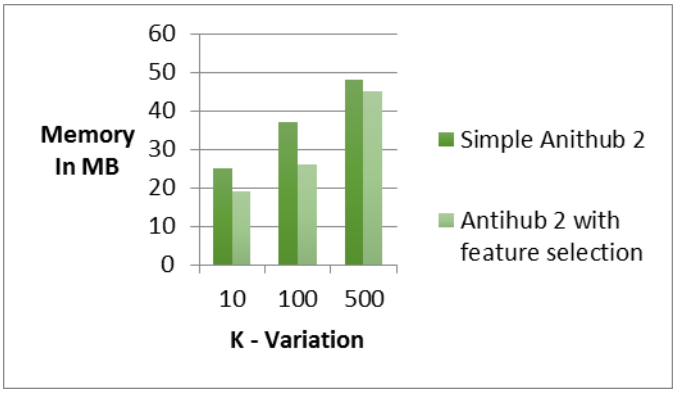

Fig. 4. Memory comparison with k variation

onsider Feature selection selects 25 dimensions from 38 dimensions. If existing system needs 1 unit time to process all 28 features then proposed system will required 0.65 unit time. Same as time, memory requirement will be less than existing system.

5. Conclusion

Exiting method proposed reverse nearest neighbor outlier detection using anti-hub. But using anti hub for outlier detection is of large computational task. Computational complexity increases with the data dimensionality to avoid this discarding of unwanted features before application of reverse nearest neighbor is proceed. Reduces computational task and improves the efficiency of finding anti-hub and also enhances the anti-hub based unsupervised outlier detection. From actual results it is clear that proposed system maintains the accuracy and also reduces the time and memory requirement for outlier detection.

REFERENCES

[1] Hawkins, D.: "Identification of Outliers", Chapman and Hall, London, 1980..

[2] V. Hautamaki, I. Karkkainen, and P. Franti,"Outlier detection using k-nearest neigh-bour graph," in Proc 17th Int. Conf. Pattern Recognit., vol. 3, 2004, pp. 430-433.

[3] J. Lin, D. Etter, and D. DeBarr,"Exact and approximate reverse nearest neighbor search for multimedia data," in Proc 8th SIAM Int. Conf. Data Mining, 2008, pp. 656-667.

[4] Y. Tao, M. L. Yiu, and N. Mamoulis, "Reverse nearest neighbor search in metric spaces,"IEEE Trans. Knowl. Data Eng.,vol. 18, no. 9, pp. 1239-1252, Sep. 2006.

[5] C. Lijun, L. Xiyin, Z. Tiejun, Z. Zhongping, and L. Aiyong, "A data stream outlier delection algorithm based on reverse k nearest neighbors," in Proc. 3rd Int. Symp. Comput.Intell.Des., 2010, pp. 236-239.

[6] Gustavo H. Orair, Carlos H. C. Teixeira, Wagner Meira Jr., Ye Wang and Srinivasan-Parthasarathy , "Distance-Based Outlier Detection: Consolidation and Renewed Bear-ing,"2010.

[7] K. S. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When is nearest neighbor meaningful?,"in Proc. 7th Int. Conf. Database Theory, 1999, pp. 217-235.

[8] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, "A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data,"in Proc. Conf. Appl. Data Mining Comput. Security, 2002, pp. 78-100.

[9] M. Breunig, H.P.Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," SIGMOD Rec., vol. 29, no. 2, pp. 93-104, 2000.

[10] Hans-Peter Kriegel,Matthias Schubert and Arthur Zimek, "Angle-Based Outlier De-tection in High-dimensional Data," 2008.

[11] Hermine N. Akouemo and Richard J. Povinelli "Time series outlier detection and imputation" Milwaukee, Wisconsin 53233,july 2014

[12] Milos Radovanovic, AlexandrosNanopoulos, and MirjanaIvanov , "Reverse nearest Neighbors in Unsupervised Distance-Based Outlier Detection," 2015.

[13] Milos Radovanovic, AlexandrosNanopoulos, and MirjanaIvanov , "Reverse nearest Neighbors in Unsupervised Distance-Based Outlier Detection," 2015.