ATLANTIS PRESS

Advances in Computer Science Research, volume 83

8th International Conference on Social Network, Communication and Education (SNCE 2018)

Outlier Detection Based on Group Behavior in Trajectory Data

Junling Liu[1,a *], Owaraye Ademola Elijah[2,b],Dacheng Ye[3,c]

[2]School of Information and Control Engineering, Shenyang Jianzhu University, Shenyang 110015, China

[a] liujl@sjzu.edu.cn, [b]adeadura5@gmail.com, [c]571404720@qq.com

* The corresponding author

Keywords: Trajectory data; Exception object detection; Similarity calculation; Sliding Window; Hash table

Abstract. In recent years, with the rising popularity of GPS, and sensor network, the behavior of the trajectory data is collected and stored in the application server. There has been lots of research achievements at present in the application scenario. The uncertainty of the eventis not completely suitable for outlier detection algorithm of various kinds of application scenarios. In this paper, we study the behavior of the trajectory data, which is based on the behavior of the event and the group relation between objects, which based on the above two points. This paper introduce the object trajectory similarity computing the exception object detection algorithm.The experimental results on real datasets show the effectiveness and efficiency of proposed algorithms.

Introduction

With the increasing popularity of applications such as GPS system and wireless communication, a lot of behavior trajectory data are collected and stored on application servers, such as campus card data, bus card data and employee card data. Through these object behaviors, data can be recorded, which can be analyzed including trajectory clustering, trajectory classification and location recommendation.The trajectory of the same group is similar, for example, in a class, students trajectories are similar in class time.To find outlier of objects in a group such as the campus card data of a university, which labels students with academic performances by findingtheir behavioral differences.

We propose a problem that detects outliersby their behavior trajectory. The abnormal objects are defined as obvious objects on behavior trajectories with the same group.The existing outlier detection problems are mostly based on continuous time series trajectory data. The general method is to extract sequence features and detect abnormal objects by pattern matching. In this paper, the time span between the trajectory data points is relatively large. Therefore, the traditional outlier detection method is not suitable for this problem.

This paper studies the behavior of trajectory data. The traditional outlier detection method for trajectory data cannot be used directly, so the design of an effective method and measure of outlier an efficient algorithm for outlier detection is necessary.In order to solve these problems, we design a new similarity measure method, to detect abnormal objects by calculating similarity between objects at the same time. Also, we preset the double index detection algorithm based on hash table to improve the detection efficiency.The main contributions of this paper are as follows:(1) A new similarity measure for trajectory data is designed; (2) An outlier detection algorithm based on double hash index is designed; (3) This paper uses a real data set to verify the effectiveness of the algorithm with a perfect experimental scheme.

This paper is organized as follows. Section 2 discusses the related work. Section 3 gives the problem defintion; Section 4 describes the outlier detection algorithm; Section 5 presents the experimental results and analysis; Section 6 concludes this paper.

Related Works

Outlier detection is an important task in data mining, including a variety of detection methods, mainly based on statistical methods, such as [1], distance-based method [2], density-based method [3], clustering-based method, etc. Abnormal behavior detection algorithm based on statistical assumptions on the data set is common. The serious deviation from the distribution curve of the data points is defined as abnormal behavior. Abnormal data mining based on distance was first proposed by Knorr et al, the abnormal behavior is defined as the concentration of most data and data is greater than the distance from a threshold point. Outlier data mining method based on density was first proposed by Breunig et al, this method gives each data object to a local outlier factor (LOF), the greater the value is, the higher the possibility of abnormal data mining is. Theoutliers are the objects that are not in any clusters.

Sequencial data can be divided into three categories according to different application fields: time series, event sequence and biological sequence. Research works mainly include classification and prediction, clustering [4], anomaly detection etc. In sequencial data anomaly detection, a small amount of data that does not conform to a certain rule in a large number of data is defined as an exception. The trajectory data in this paper is similar to time series and event sequences, but time series emphasize on the continuity on time. Therefore, these three data are different. The similarity of these features is measured by calculating the Euclidean distance and dynamic time warping distance [5]. This method is mainly from the distribution of data according to the similarity of distribution to describe the original similarity. Indexing technology is an important means to improve the efficiency of the algorithm, since trajectory data is a multi-dimensional data structure.The existing multidimensional data index is mainly divided into: points division index such as R-tree, R+-tree and R*-tree[6]. Space partition index such as k-d tree, and k-d-B tree[7] is based on quad-tree, which is the index of space for repeat segmentation. However, these index structures can not be applied to the trajectory data, the spatial data of this study cannot track data segmentation, therefore we proposed the index structure of the hash table key.

Problem Definitions

Given a set of objects $P=\{p_1,p_2,\ldots,p_n\}$, and a hierarchal structure $C\{c_0<c_1<,\ldots,<c_n\}$, where$c_0$is the highest level including one object. For example, level c_1 is $(c_{10},c_{11},\ldots,c_{1n})$. Object $p(ID,c_p,TR)$, where $c_p(c_{0p},c_{1p},\ldots,c_{np})$, $TR=\{b_1,b_2,\ldots,b_n\}$, any $b(l_i,t_i,e_i)$ is an activity, wherel_i is a location, t_i is the time of activity, e_i is an event.

Definition 1:Similarity Of Behavior Trajectories. Given a time threshold δ, if object p and q attend the same event, then p and q have similar activity. The similarity on timep.b and q.b can be defined as activity similarity of p and q, as shown in formula 1:

$$Sim(p.b_1, q.b_2) = \begin{cases} 1 - \Delta_t/\delta, \Delta t \le \delta \\ 0, \Delta t > \delta \end{cases} \quad (1)$$

Where $\Delta t = |b_1.t - b_2.t|$ is the time difference of p.b and q.b. For object p and q, their behavior similarity is accumulated by activity similarity, which is shown in formula 2:

$$TSim(p,q) = \sum_{i=1,j=1}^{m,n} sim(p.b_i, q.b_j) \quad (2)$$

Where m and n are the number of p and p activities respectively.We discovered that if we only use definition (2) to calculate the similarity between two objects, in some cases it is not reasonable. For example, when the number of an event of an object is far greater than the other types of events. We present the definition of the similarity based on the grouping.

Definition 2:Object Similarity with Groups. The sum of the behavior trajectories of all objects that belong to the same group GSim(p) is used to represent the group, based similarity of object p, which is shown in formula 3.

$$GSim(p) = \sum_{i=1}^{n-1} TSim(p, q_i) \quad (3)$$

Wheren represents the total number of objects in the group c_i.

Definition 3:Total Similarity. The total similarity of the object p is the weight value of the group based on the similarity of the group that the object p belongs to. The total similarity calculation formula is given as equation 4:

$$ASim(p) = \sum_{i=0}^{m} \beta_i \cdot (GSim(p) / |c_i|) \quad (4)$$

Where |ci | indicates the number of objects in the c_i group, where m is the group number of p, and β_i is the weight by in group c_i.

With regard to β_i, the smaller the group of the object is, the greater the β_i. That is, in each group, other objects interact with object p frequently, and the impact on p is the largest.

Definition 4Top-k Outlier Trajectories. In a given dataset, suppose that there are n objects. If the total similarity of the former k objects is smaller than the total similarity of the other n-k objects, the k objects are called top-k abnormal objects.

Algorithms

In this paper, we detect the abnormal objects by calculating the similarity between objects and the grouping relations of objects, and propose several algorithms for effective computation of the similarity between objects.

Similarity Computation Of Trajectories. To calculate the object trajectory similarity by definition 3, the first step is to find similar behavior. Behavior trajectory object is a collection of behavior. To find similar behavior that requires full scan to match, we use a sliding window algorithm to improve the efficiency. The time threshold is a set reference pointer.

Sliding Window Algorithm. In group C, we can calculate the similarity of an object with other objects by definition 4. It is necessary to calculate the behavior trajectory similarity with all other objects, assuming that there are three objects p_1, p_2, p_3 in group C, i.e.

$$GSim(p_1) = TSim(p_1, p_2) + TSim(p_1, p_3). \quad (5)$$

According to the formula (5), the similarity between the behavior trajectory of p_1 and the other two objects is calculated respectively. Suppose there are n objects to the group, according to the sliding window algorithm, it is necessary to calculate $C_n^2/2$ times. In order to reduce computation times, we improve the sliding window algorithm, in which a group of objects is merged as a whole *P*, so the algorithm only needs to compute each object with the similarity of the whole shown in formula 6.

$$GSim(p_1) = TSim(p_1, P) \quad (6)$$

Algorithm 1*CGSim(L_1,L_2)*
Input：Objects in a group
Output：each object p_i

1. *For i from 1 to n step 1*
2. *L_2.Insert(p_i)*
3. *For j from 1 to n step 1*
4. *L_1.Insert(p_j)*
5. *TSim(L_1,L_2)*

Hash Table Algorithm. The sliding window algorithm improves the matching speed and reduce the number of calculations. However, these two algorithms have a drawback, which cannot find the results once. we propose hash table algorithm that indexes the whole object with hash table data.We propose the indexing by time as the key to create a hash table index.

The idea of the algorithm 2 is as follows:(1) For each data of the merged trajectory data object *P*, the time string is used as the key to establish hash table index.(2) To get object p_i, a hash value is computed according to the time of the string, and then find the hash table.(line 3-5);calculatesthe

behavior similarity of object p_i (line 3-5);(3) until all objects have been calculated. Algorithm 3 is a Pseudocode.

Algorithm 2*HGSim(P)*
Input: A group objects *P*
Output: the similarity of each p_i
1. *For i 1 to n step 1*
2. *If Hash_Table.has_key(hash(P_i.t) then*
3. *Hash_Table[pos].insert(P_i)*
4. *Else then*
5. *Hash_Table .insert(Hash(P_i.t))*
6. *For j 1 to n step 1*
7. *If hash(p_j.t) in Hash_Table then*
8. *If (p_j.pl &&Δt>δ)*
9. *Sim +=(1-Δt/δ)*

Experiments

Effectivness. We analyzedstudents trajectories of three class from campus card data by definitions given in this paper. We found that an outlier object has small similarity with other students. From these students, we can infer that they may not eat regularly, and stay at school for a relatively short time. At the same time, they like to play games. if a student's performance is described above, such student has a poor academic performance. In order to verify the idea of this paper, we observedthese students' academic rankings, and are ranked behind. The results of the experiment verify that the algorithm proposed in this paper is effective.

Sliding Window And Hash Algorithms. The experiment parameters settings: The time threshold is 1 hour, five datasets are selected, and the hash barrel size is set to 1 hour. The hash table size is 2×10^7. The results of the experiment is as shown in Figure 1.

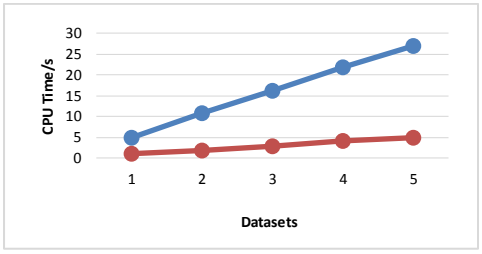

Figure 1. Finite Comparison between sliding window and hash table algorithms

From Fig. 1, the hash table algorithm takes longer time than the sliding window algorithm. The sliding window algorithm growth rate, is greater than the hash table.

The Effect of Time Threshold Size. The experimental time threshold is 30,60,90, 120, 150 minutes respectively.

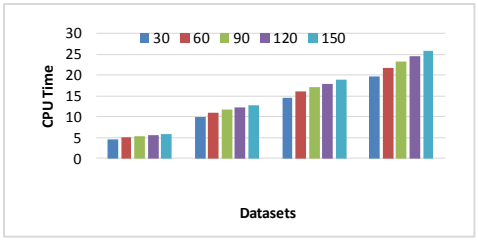

Figure 2. Finite Time threshold changes impact on the efficiency of the sliding window algorithm

The experimental results are shown in Fig. 2, we can see that while the time threshold increases, the running time also increases.Choosing a suitable time threshold is very important, according to the actual application.

Fig. 3shows the experimental results of the hash table algorithm, where the hash barrel size is set to 1 hour. It can be seen from the graph that when the data size is relatively small, the time threshold change, has little effect on the running time. When the data size is large, the running time is also increasing with time threshold.

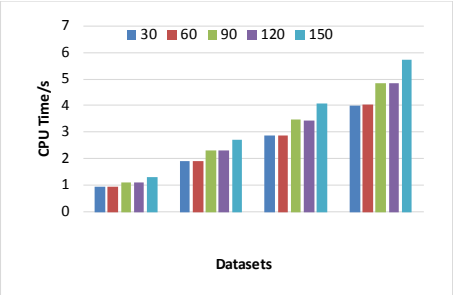

Figure 3. Finite Time threshold changes impact on the efficiency of the hash table algorithm

Conclusions

In this paper, we studied the behavior trajectory data of object grouping relations, proposed a similarity measurement method for the object, to detect abnormal objects by calculating the similarity between objects. To improve the efficiency of query, we present an anomaly detection algorithm based on double hash, validity and correctness and through real the data set to verify the algorithm.

Acknowledgements

This work was supported by Chinese Society of Academic Degrees and Graduate Education under Grant No.B-2017Y0904-161, Department of Education of Liaoning Province Foundation under Grant No.LJZ2016008.

References

[1] Rousseeuw P J, Leroy A M. Robust regression and outlier detection[M]. John Wiley & Sons, 2005.
[2] Knorr E M, Ng R T, Tucakov V. Distance-based outliers: algorithms and applications[J]. The VLDB Journal—The International Journal on Very Large Data Bases, 2000, 8(3-4): 237-253.
[3] Breunig M M, Kriegel H P, Ng R T, et al. LOF: identifying density-based local outliers[C]//ACM sigmod record. ACM, 2000, 29(2): 93-104.
[4] Wang J, Zhang Y, Zhou L, et al. Discriminating Subsequence Discovery for Sequence Clustering[C]//SDM. 2007: 605-610.
[5] Kim S W, Park S, Chu W W. An index-based approach for similarity search supporting time warping in large sequence databases[C]//Proceedings. 17th International Conference onData Engineering. IEEE, 2001: 607-614.
[6] Beckmann N, Kriegel H P, Schneider R, et al. The R*-tree: an efficient and robust access method for points and rectangles[M]. ACM, 1990.
[7] Peng L, Shizhao N, Zheng W, Ziwei J, Jianwu Y, Zhongxiang Q, Wangmo P. Predicting durations of online collective actions based on Peaks' heights [J]. Communications in Nonlinear Science and Numerical Simulation. 2018, 55: 338-354.
[8] Robinson J T. The KDB-tree: a search structure for large multidimensional dynamic indexes[C]//Proceedings of the 1981 ACM SIGMOD international conference on Management of data. ACM, 1981: 10-18.