IEICE TRANS. INF. & SYST., VOL.E97–D, NO.8 AUGUST 2014

LETTER

Trajectory Outlier Detection Based on Multi-Factors

Lei ZHANG[†a)], ***Member***, **Zimu HU**[†b)], ***and*** **Guang YANG**[†c)], ***Nonmembers***

SUMMARY Most existing outlier detection algorithms only utilized location of trajectory points and neglected some important factors such as speed, acceleration, and corner. To address this problem, we present a Trajectory Outlier Detection algorithm based on Multi-Factors (TODMF). TODMF is improved in terms of distance-based outlier detection algorithms. It combines multi-factors into outlier detection to find more meaningful trajectory outliers. We resort to Canonical Correlation Analysis (CCA) to optimize the number of factors when determining what factors will be considered. Finally, the experiments with real trajectory data sets show that TODMF performs efficiently and effectively when applied to the problem of trajectory outlier detection.

key words: *trajectory, outlier detection, multi-factors, canonical correlation analysis*

1. Introduction

As the localization technology of moving objects such as GPS, RFID and wireless sensors develops, people constantly obtain more and more mobile data. In recent years, the research on how to handle mobile data has attracted much interest and outlier detection has become a very important research direction. Outlier detection and analysis is one of important branches of data mining, and it is widely used in many fields like weather forecasting, animal migration, and network intrusion detection analysis. It is mainly used to find data that is significantly different from other data in the dataset. Knorr et al. [1] proposed a distance-based outlier detection algorithm in terms of the concept of distance-based outliers. According to this concept, an object would be considered as an outlier if the distances between other objects in the dataset and it were greater than a given distance. Breuning et al. [2] proposed a density-based outlier detection algorithm (LOF). In their method, if the LOF of an object was greater than a given threshold, it would be considered as an outlier. Sarawagi et al. [3] provided a discovery-driven migration-based outlier detection algorithm based on OLAP data cube. Strugy et al. [4] put forward a depth-based outlier detection algorithm called DEEPLOC. In DEEPLOC, they assigned each object to a depth value. If the depth value of an object was small, it would be considered as an outlier. However, most of the algorithms above just consider the location information. Thus they could not make full use of the trajectory data containing some factors such as speed, acceleration, corner.

In this paper, we propose a trajectory outlier detection algorithm based on multi-factors, called TODMF. TODMF improves the traditional distance-based algorithm by combining multi-factors into outlier detection to discover more meaningful trajectory outliers. In order to select multi-factors used in outlier detection, we adopt CCA as done in [5], [6]. If the correlation between two factors is high, the number of factors will be optimized by reducing the algorithm complexity of TODMF.

2. Trajectory Outlier Detection Based on Multi-Factors

Note that most existing outlier detection algorithms only used the location of trajectory points, and ignored the other important factors of trajectory points such as speed, acceleration and corner. Since these factors can reflect the movement status of moving objects, we can obtain more meaningful information if we make use of them in real applications. To this end, we put forward a trajectory outlier detection algorithm based on multi-factors which consider multi-factors in outlier detection. For point P_i of trajectory T_i, its speed, acceleration and corner can be described as follows, respectively.

$$\begin{aligned} v_i &= \frac{\sqrt{(x_{i+1}-x_{i-1})^2+(y_{i+1}-y_{i-1})^2}}{t_{i+1}-t_{i-1}}, \\ a_i &= \frac{v_{i+1}-v_i}{t_{i+1}-t_i}, \\ \theta_i &= \pi - \cos^{-1} \\ &\quad \times \frac{\begin{array}{c}(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2+(x_i-x_{i-1})^2 \\ +(y_i-y_{i-1})^2-(x_{i+1}-x_{i-1})^2-(y_{i+1}-y_{i-1})^2\end{array}}{\begin{array}{c}2\times\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2} \\ \times\sqrt{(x_i-x_{i-1})^2+(y_i-y_{i-1})^2}\end{array}} \end{aligned} \quad (1)$$

After these multi-factors are considered, the trajectory point P_i and the trajectory T_i can be described as $P_i = \langle x_i, y_i, t_i, v_i, a_i, \theta_i \rangle$ and $T_i = \{P_1, P_2, \ldots, P_i, \ldots, P_n\}$.

In such a case, a reasonable method to measure the distance of multi-factors between trajectory points is needed to be given before multi-factors are taken into outlier detection. Traditional distance-based outlier detection algorithms

Manuscript received April 9, 2013.
Manuscript revised December 29, 2013.

[†]The authors are with School of Computer Science and Technology, China University of Mining and Technology, 1 University Road, Xuzhou, 221116, Jiangsu, China.
a) E-mail: zhanglei@cumt.edu.cn
b) E-mail: huzimu@cumt.edu.cn
c) E-mail: yangguanglm@cumt.edu.cn

DOI: 10.1587/transinf.E97.D.2170

only took location of trajectory points into consideration, so they used the Euclidean distance to measure the distance between points. When multi-factors are taken into outlier detection, the distance for each factor has to be defined before outlier detection. For the sake of computational simplicity, the absolute values of the difference between them are used as distances. Before calculation, the values of multi-factors are normalized in order to be combined together. After the distances of multi-factors are obtained, the multi-factors distance can be defined in terms of the weighted average.

TODMF is improved in terms of distance-based algorithms due to the fact that it can effectively combine multi-factors to find outliers of a trajectory with the definition of the multi-factor distance. Before a trajectory point is detected, a percentage pct and a distance d_{min} should be set. If there exist more than pct percentage of the total points of a trajectory whose distances between the detected point and them are larger than d_{min}, the detected point would be considered as an outlier. The pseudo code of TODMF can be described as follows:

Algorithm : TODMF
Input: T_i={P_1,P_2,...,P_i,...,P_n}, d_{min}, pct
Output: O_T

```
01: Inialize d[m] to store distance of each factor, m is the number of the defined
distance
02:   For i=1 to n
03:     For j=1 to n
04:       For each factor
05:         compute dk
06:       d(pi,pj)=∑dk/m
07:   if d(pi,pj)>dmin
08:     Numi++;
09:   if(Numi>n*pct)
10:     add Pi to OT
```

From the above algorithm, we can know that the time complexity of TODMF is O(N^2).

3. Trajectory Multi-Factors Correlations Based on Canonical Correlation Analysis

In real applications, due to the characteristics of the moving objects' movement, there are always some relationships among trajectory points' multi-factors. In traditional methods, in order to study correlations between two factors, all the correlation coefficients among them had to be calculated. This makes the problem become complicated and it cannot well embody the essence of the problem. While in canonical correlation analysis (CCA), two factors are treated as a whole and the correlation between them is explored. The wholes can be described as linear combinations between their features. So the problem can be converted to study the correlations between linear combinations, which is simpler than traditional methods.

3.1 Mathematical Description of CCA

For two groups of variables $X = (x_1, x_2)$ and $Y = (y_1, y_2)$, the covariance matrix of variables $Z = (x_1, x_2, y_1, y_2)$ can be described as Eq. (2).

$$\sum = \begin{bmatrix} \sum_{11} & \sum_{12} \\ \sum_{21} & \sum_{22} \end{bmatrix}. \tag{2}$$

In Eq. (2), $\sum_{11}$ is the covariance matrix of the variable $\langle x_1, x_2 \rangle$, $\sum_{22}$ is the covariance matrix of the variable $\langle y_1, y_2 \rangle$, and $\sum_{12}$ is the covariance matrix of X and Y.

The linear combination of the first variable can be described as Eq. (3).

$$\begin{aligned} u_1 &= \alpha_1^T X. \\ v_1 &= \beta_1^T Y. \end{aligned} \tag{3}$$

From Eq. (3), one has

$$\begin{aligned} Var(u_1) &= \alpha_1^T Var(X)\alpha_1 = \alpha_1^T \sum_{11} \alpha_1 = 1, \\ Var(v_1) &= \beta_1^T Var(Y)\beta_1 = \beta_1^T \sum_{22} \beta_1 = 1, \\ \rho_{u_1,v_1} &= Cov(u_1, v_1) = \alpha_1^T Cov(X, Y)\beta_1 = \alpha_1^T \sum_{12} \beta_1. \end{aligned} \tag{4}$$

So the aim of CCA is to work out α_1 and β_1, and to make the correlation coefficient ρ between them reach the maximum value. The detailed procedure is described in [5]. If this part cannot be able to explain the original variables, we can work out the second pair of canonical variables and their correlation coefficients from the rest of the correlations.

3.2 Correlation and Type of Multi-Factors

In order to reflect the correlation between two variables, correlation coefficients are used. It is an indicator which is used as the measure of correlation and it is usually denoted by λ. In CCA, the correlation coefficient refers to the maximal value ρ. In statistic, the value of $|\lambda|$ and the sign of λ are usually used to express correlation and correlation type, respectively. For the correlation type, if $\lambda > 0$, it is positive correlation. If $\lambda < 0$, it is negative correlation, and if $\lambda = 0$, it is zero correlation. As to the correlation, the closer to 1 the value of $|\lambda|$ is, the closer the correlation between them. Table 1 lists the meaning of the value of $|\lambda|$.

In real applications, if the correlation between two variables exceeds low correlation, the correlation between them can be considered strong. In order to select the factors used for outlier detection with CCA, we can use either of the features for outlier detection instead of using them both if there exist two factors which have the correlation exceeding low correlation. With this scheme, the complexity of TODMF can be reduced and its efficiency can be significantly improved.

Table 1 Relationships between value of $|\lambda|$ and correlation.

Value of $\|\lambda\|$	Correlation
0.00-0.30	Low Correlation
0.30-0.50	True Correlation
0.50-0.80	Remarkable Correlation
0.80-1.00	High Correlation

4. Experiments and Analysis

In order to validate TODMF, the trajectory outlier analysis system TraOAS is developed, in which we use Visual C++ 2008, and store the trajectory data in Access2003. The real dataset of the Starkey Project [7] is used in our experiments.

4.1 Impact of the Number of Factors on Outlier Detection Time

In this experiment, we set $d_{min} = 0.1$ and $pct = 0.9$. As shown in Fig. 1, the time of TODMF with three factors including speed, acceleration and corner is represented as the line marked with rectangle. The time with two factors containing speed and acceleration is represented as the line marked with diamond, and the time with just one factor (speed) is represented as the line marked with triangle. It can be seen from Fig. 1 that as the number of factors increases, the running time grows.

4.2 Impact of Parameters on Outlier Detection

In TODMF, the parameters used for outlier detection are d_{min} and pct. In this experiment, trajectory of NO.311 is taken, which contains 1018 points. Then we can validate the impact of parameters on TODMF.

Figure 2 shows how the number of outliers changes with different d_{min} when we set $pct = 0.9$, From Fig. 2, we can see that as the value of d_{min} increases, the number of points which satisfy the constraint conditions increases for fixed pct. The result is consistent with the expectations of TODMF. In addition, we can also find that when d_{min} is located in 0.05 and 0.15, the result varies sharply. So for this area, the increment is reduced and many values d_{min} are taken into the experiment. Figure 3 shows changes of the number of outliers with different pct when we set $d_{min} = 0.1$. It can be seen from Fig. 3 that as the value of pct increases, the number of outliers decreases.

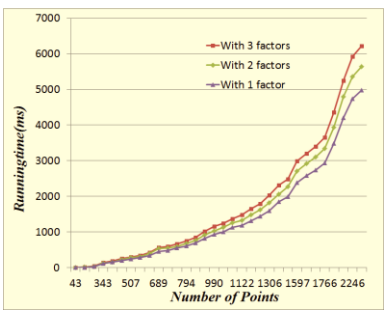

Fig. 1 Impact of number of factors on detection time.

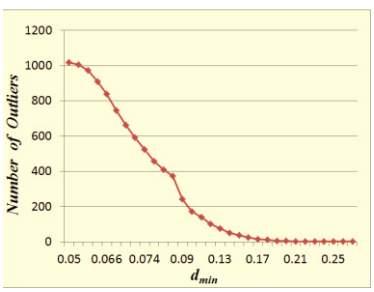

Fig. 2 Impact of d_{min} on outlier detection.

4.3 Impact of CCA on Outlier Detection

In order to reduce the time, CCA on multi-factors can be used. If there are two factors that have high correlation, we can use either of them for outlier detection instead of both of them to optimize the number of factors. In order to compare the results under two different conditions above, the concept of set similarity is used. If the set similarity between them is high, it can be considered that they have the same effect.

Set similarity. Given two sets A and B, the set similarity of them can be described as follows.

$$Sim(A, B) = \frac{num(A \cap B)}{num(A \cup B)} \tag{5}$$

In order to validate the impact of CCA between multi-factors on outlier detection, the following experiments are done. First, the correlation coefficient and correlation between speed and acceleration are analyzed with CCA. Second, we show how the set similarity changes as the parameters change. In this step, the set MO is the result of outlier detection with speed, acceleration and corner, while the set SCO is the result with speed and corner. Finally, we can validate the impact of CCA on outlier detection.

In this experiment, trajectory No.311 which contains 1018 points is taken. In order to conform to real applications, speed and acceleration are described as $V = (v_x, v_y)$ and $A = (a_x, a_y)$. According to the conclusions in Table 1, we work out that the largest correlation coefficient between them is 0.39 and so the correlation between them is true correlation.

Then we compute the value of $Sim(MO, SCO)$ with different parameters d_{min} and pct. In Fig. 4, we set pct to the fixed value 0.9, and compute the value of $Sim(MO, SCO)$ with different d_{min}. In Fig. 5, we set d_{min} to a fixed value 0.1, and compute the value of $Sim(MO, SCO)$ with different pct. It can be clearly seen from Fig. 5 that in the effective range the values of $Sim(MO, SCO)$ are always greater than 0.35.

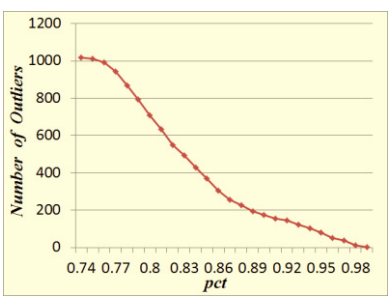

Fig. 3 Impact of pct on outlier detection.

Fig. 4 Impact of d_{min} on set similarity.

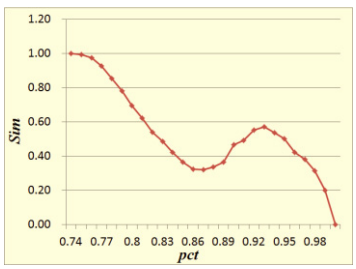

Fig. 5 Impact of *pct* on set similarity.

On the other hand, it means that the set similarity between set *MO* and set *SCO* is quite high.

In summary, when the correlation between speed and acceleration is true correlation, there is a high set similarity between the set which comes from the outlier detection with speed, acceleration, and corner and the set with speed and corner.

5. Conclusion

Existing outlier detection algorithms only focused on location information of trajectory points and neglected some important factors such as speed, acceleration and corner. In this paper, we propose an outlier detection algorithm based on multi-factors. It improves the traditional distance-based algorithms since it combines multi-factors into outlier detection with the concept of multi-factors distance to find more meaningful trajectory outliers. In order to select multi-factors, we can optimize the number of factors with CCA. The experiment on real dataset shows when the correlation between speed and acceleration is true correlation; we can use speed, corner for outlier detection instead of using speed, acceleration and corner. By this method we can reduce the running time and remain the same effect.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities (2013XK10).

References

[1] E.M. Knorr and R.T. Ng, "Algorithms for mining distanced-based outliers in large datasets," Proc. 24th Int'l Conf. Very Large Data Bases, 1998, New York, US, 1998.
[2] M. Breunig, H.P. Kriegel, and R.T. Ng, "LOF: Identifying density-based local outliers," Proc. 2000 ACM SIGMOD International Conference on Management of Data, 2000, NewYork, US, 2000.
[3] S. Sarawagi, R. Agrawal, and N. Megiddo, "Discovery-driven Exploration of OLAP Data Cubes," Proc. Int. Conf. Extending Database Technology, 1998, Valencia, 1998.
[4] A. Struy and P.J. Rousseeuw, "High-dimensional Computation of the deepest location," Computational Statistics and Data Analysis, vol.3 no.4, pp.415–426, March 2000.
[5] H. Asoh and O. Takechi, "An approximation of nonlinear canonical correlation analysis by multilayer perceptrons," Proc. Int. Conf. Artificial Neural Networks, 1994, Sorrento, Italy, 1994.
[6] P.L. Lai and C. Fyfe, "Kernel and nonlinear canonical correlation analysis," Int. J. Neural System, pp.365–377, Oct. 2000.
[7] http://www.fs.fed.us/pnw/starkey/data/tables/