2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)

Enhanced Minimum Description Length Preprocessing of Time Series Trajectories

Gajanan Gawde
Computer Engineering Department
Goa College of Engineering
Ponda, India
gg@gec.ac.in

Jyoti Pawar
Department of Computer Science and Technology
Goa University
Panaji, India
jyotidpawar@gmail.com

***Abstract*—Mining of moving objects trajectories is becoming more and more popular due to its wide range of real time applications. Discovering knowledge from the time series trajectories is challenging and there is needs to improve the quality of the output generated. There are many effort being made to improve the quality of mining process and also to improve the accuracy of the techniques. Preprocessing of the input raw trajectories play an important role in the mining of trajectories. The objective of preprocessing of input trajectories is to reduce the number of segments of the original trajectories without losing major data from original trajectories. The Minimum Description Length (MDL) principle is used to preprocess the input trajectories and reduce the length of trajectories. There are two major issue with the MDL principle. The first issue faced by MDL, the points which are very close to each other are remained unprocessed. Second issue is, the loss of information using MDL principle is considerably more. There is need to tackle both the problem ,so that performance of the preprocessing is improved. In this paper, we have proposed RegressionLine technique to process points of trajectories which MDL principle cannot process. We have proposed RelativeLoss algorithm to preprocess the input trajectories by reducing the loss of data. Proposed techniques were tested on real time and synthetic datasets and showing improvement over existing MDL technique.**

***Index Terms*— Enhanced MDL, Regression Line, Relative Loss.**

I. Introduction

Moving objects are generating large number of moving objects trajectories and these trajectories contain hidden information which are vital for personal or commercial purpose. Extracting knowledge from huge amount of moving objects trajectories is very challenging task. There is need to design accurate algorithm and at the same time, we need to reduce the loss of data or vital information during preprocessing.

The preprocessing of time series trajectories is carried out using MDL principle. The core idea of preprocessing is to reduce the length of the trajectories and this is achieved by identifying segments or points which are redundant. Using MDL principle, angular and perpendicular distance of segments under consideration are computed and this distance is marked as mdl distance. Using Euclidean distance, again distance is computed between the segments under consideration and this distance is marked as euclidean distance. The Trajectory is partitioned if mdl distance value is greater than euclidean distance value and this is called as MDL condition to partition the trajectory. Trajectory is scanned from starting to end , to identify partition points and accordingly segments are replaced by new segments.

The existing MDL technique has few issues , which we would like to discuss. If the points of the trajectory are very close to each other, then MDL principle is not able to process such points of the trajectory. In other words , MDL principle is unable to detect the partitioning points of the trajectory and hence the same points are displayed without processing.

The second major issue with the MDL technique is that there is loss of data while preprocessing of trajectory. This loss of data is very vital while extracting interesting pattern from the moving objects trajectories. The amount of information loss is at higher side and there is need to reduce such loss of information in order to get good accurate result. Let us try to understand this issue with the help of example. Consider the diagram shown in Figure. The diagram consists of one original trajectory and second one is preprocessed trajectory using mdl principle. The number of partition points identified are N and for each partition points we have calculated loss of data , which is shown in the table. By looking at the table , we can make out that, the amount of loss of data is higher side and there is need to reduce this loss , so that accuracy of trajectories mining is improved.

In order to tackle above two issue, we have proposed RegressionLine and RelativeLoss techniques. The RegressionLine Technique is to process all the points which are close to each other and replace such points by a segment which represents a set of points. The RelativeLoss technique identify a partition point which minimizes the loss of data relative compared to the MDL principle.

Contribution of our work in this paper is as follows:

1. We have proposed RegressionLine Technique to process noisy points of the trajectory.
2. We have proposed RelativeLoss Technique to reduce the loss of data which result from using MDL principle.

The rest of paper is organized as follows. In section 2 we present related work. In section 3, we propose Regression-

978-1-5090-3294-5/17/$31.00 ©2017 IEEE

Line and RelativeLoss Technique to improve the performance of time series trajectories mining. Section 4, provides the experimental results and finally, section 5 concludes the paper.

Symbols	Meaning
R	Time Series Trajectory R
S	Time Series Trajectory S
M	Length of trajectory R
N	Length of trajectory S
MDL	Minimum Description Length

Table 1: Symbols and Meaning

II. Related Work

Dynamic Time Warping(DTW) (1) distance measure is used to compares two trajectories for similarity. DTW distance measure stretches the trajectories to check if there is any similarity in the neighbourhood of points. DTW is invariant of translation and scaling since trajectories are normalized using mean and standard deviation. DTW is not invariant of rotation and cannot compares trajectories if one of the trajectories is rotated with some angle. Longest Common subsequence(LCSS) is proposed in (2) to compares the trajectories for similarity by identifying longest continuous sequence match. When the two points are in close proximity of each other with threshold value , similarity distance is incremented by one, thus showing maximum similarity. In LCSS, the longest common subsequence is identified recursively and if the distance is maximum which shows maximum similarity between two trajectories under consideration.

Edit distance on real penalty(ERP) is proposed in (3). The two trajectories are compared using ERP distance measure for similarity and if there is mismatch, then penalty is set. ERP distance measure is invariant of scaling and rotation but not invariant of rotation. ERP distance measure supports metric property and this metric property is useful to prune the search space of searching. Edit distance on Real Sequence(EDR) is proposed in the (4). EDR distance measure compares the two trajectories by checking if two points are in close proximity of each other with some threshold value. If two points are close to each other, then this is considered as match and distance is set to zero else there is mismatch and distance is set to one. ERD distance measure is invariant of scaling and translation but not invariant of rotation.

Das (5) and Vlachos (2) applied the LCSS measure to time series trajectory similarity. LCSS allow a variable length gap to be inserted during matching of trajectories and hence robust to the noise. In (6) extended LCSS measure to compare trajectories of objects from video. In previous paper , first 3D trajectories are extracted from video and then LCSS distance measure is used to compare trajectories. In (2) explored searching of similar multi dimensional trajectories. LCSS distance measure is used to compares muti dimensional trajectories. LCSS measure is used in conjection with sigmoidal function is proposed by (7) to compare trajectories for similarity. In (8) index multi dimensional trajectories supporting multiple distance measures such as LCSS , DTW and the index structure was design in a such a way that there was no need to rebuild index again and again. In (9) various dimensionality reduction methods were investigated and contributed novel PAA technique to reduce the dimensionality. In (10) highlights the extra computation done by LCSS and same is enhanced by fine tuning the threshold value.

Various indexing techniques (11),(12) , (13),(14) , (15), (16) were proposed to improve the performance of distance measures. Cai and Ng (15) proposes an effective lower bound technique for indexing. However , Euclidean distance are used as similarity measures and Euclidean distance is sensitive towards noise and local time shifting. In (17) have enhanced the indexing method of DTW by modeling exact indexing. Human motion were efficiently indexed using bounding rectangle by (18).

III. Enhanced MDL Technique

Input trajectories need to be processed using MDL principle, so that redundant segments are replaced. There are two main issues with MDL such as the points which are very close to each other are remain unprocessed and there is loss of data due to MDL principle. We have proposed two technique to deal with the issues faced by the MDL. RegressionLine technique is proposed to delete with closely related points. RelativeLoss technique is proposed to minimize the loss of data.

3.1. RegressionLine Technique

Trajectories are processed to identify points which are very close to each other. Such data points are stored in the separate buffer. Lowest and highest points on x or y axis are identified from the buffer. Using these lowest and highest points of buffer, buffer is sampled into equal parts. This sampling is required to process the points which are close to sampled points. For each sampled point, identify all the points which are in the neighborhood of the sample point. Once all the neighbors are identified, calculate the average of these neighbors in order to get midpoint of the sampled point of buffer. Average is computed for all the sampled points. Draw a line segment passing through the average point computed at each sampled point. Calculate the error distance of regression line with the data points of the buffer. Save the information of the error distance computed. Draw all possible regression line passing through midpoint computed and compute error distance for each regression line drawn. The information computed of error distance of each regression line is saved. The regression line which has minimum error distance is selected as the representative line of the data points of the buffer.

$$\text{Error\_distance} = \text{RegLine}_i - \sum \text{Pj} \qquad (1)$$

3.2. RelativeLoss Technique

RelativeLoss Technique reduces the loss of data while identifying partition points. This reduction in the data loss is required to improve the accuracy of mining of time series trajectories.

The method works on simple procedure of computing area under curve such as f(x1) and f(x2). The f(x1) and f(x2) represent the two curve while identifying partition points. The relativeloss of data is computed using following formula :-

$$\text{Relative\_loss} = \int f(x1)\,dx1 - \int f(x2)\,dx2 \qquad (2)$$

The relative loss is computed using equation 1. If the relative loss value is greater or equal to some threshold value, then the current point is marked as partition point. The best threshold value can be set by user as per his requirement to reduce the loss of data. The threshold value would indicate the relative loss that is acceptable by the user.

```
-------------------------------------------------------------------
 Algorithm 1: RegressionLine(D) Algorithm
-------------------------------------------------------------------
 Input: D : Input Trajectories
 Output: Regression Line : RL
for each point t € D do
    for each point p € t do
        if p[i]-p[i-1]<= Threshold and count >= 3 then
            buffer[j]=p[i-1]
            count++
        Find lowest and highest point of the buffer
        Sampled the buffer in equal parts
        for each midpoint mp € MP do
            Draw regression line passing through midpoints
            compute error distance for each regression
            if error distance is minimum then
                RL = regression line
return RL
```

```
-------------------------------------------------------------------
 Algorithm 2: RelativeLoss(D) Algorithm
-------------------------------------------------------------------
 Input: D : Input Trajectories
 Output: D' : Processed Trajectories
for each trajectory t € D do
    prevArea = area of first three points
    for each point p € t do
        Add three point in sequence to set S
        if prevArea- Area(S) <= Threshold value then
            remove the last point appended
            represent set S by a segment joining first and last point of S set.
            prevArea = next thresh points in sequence
        else
            append next point to the set S
```

IV. Experimental Study

4.1. System Configuration Experimental study was carried out on Pentium V processor with 4GB of RAM and 500GB of harddisk memory. All the programs were successfully implemented using C++ language. The g++ compiler was used to compile the C++ programs. Ubuntu 12.04 operating system was used to carry out experimental study. The programs were debugged thoroughly and correct output was obtained. Experimental study was carried out with different datasets such as Character Trajectories, UJI Pen character, optical recognition of handwritten digits and MouseTracking character Trajectories datasets.

4.2. Characteristic of Time Series Datasets

Character Trajectories Dataset:- This dataset is generated by Lichman (2013), it consist of 2858 character samples, contained in the Three Dimensional Matrix. Each character sample is a 3-dimensional pen tip velocity trajectory. This is contained in matrix format, with 3 rows and T columns where T is the length of the character sample.

UJI Pen Character Dataset:- This dataset is generated by Lichman (2013), a character database by collecting samples from 11 writers. Each writer contributed with letters (lower and uppercase), digits, and other characters. Two samples have been collected for each pair writer/character, so the total number of samples in this database version is 1364.

Optical Recognition Character Dataset:- This dataset is generated by UCI Lichman (2013), who have used preprocessing programs made available by NIST to extract normalized bitmaps of handwritten digits from a preprinted form. From a total of 43 people, 30 contributed to the training set and different 13 to the test set. 32x32 bitmaps are divided into non overlapping blocks of 4x4 and the number of on pixels are counted in each block. This generates an input matrix of 8x8 where each element is an integer in the range 0 to 16. This reduces dimensionality and gives invariance to small distortions.

MouseTracking Dataset :- We generated this dataset in real time with the help of 100 user. This dataset is generated using VB.net platform. Each user was asked to move the mouse and generate different set of characters. Each person had generated 50 number of different character trajectories.

4.3. Results and Interpretation

Our proposed RegressionLine and RelativeLoss algorithms were tested on different datasets. MDL algorithm is not able to process data points which are very close to each other. Our proposed algorithm is able to process all the points of the trajectories and generate the output. We have compared MDL and RegessionLine algorithm with different datasets. Experimental results is shown in the Table 2. Experimental results revealed that our proposed technique able to process all the data points in rrespective of distance between the data points. On other hand MDL is not able to process all the points. The points which are very close to each other are not able to get detected and it remains unprocessed. Experimental results show that there few points which remain unprocessed using MDL technique whereas RegressionLine is able to process all the points.

Dataset	MDL	RegressionLine
Character	10	0
UJI Pen Character	8	0
Optical Recognition Character	12	0

Table 2: Percentage of Unprocessed points

Dataset	MDL	RegressionLine
Character	6	1.5
UJI Pen Character	9	2
Optical Recognition Character	8	2.1

Table 3: Percentage of Data loss

MDL algorithm lead to loss of the data of original trajectory. This loss of data is very critical is some applications. Thus, loss of data reduces the accuracy of the algorithm. Our proposed RelativeLoss algorithm prevent the loss of data. User has been given option to select the threshold value for the amount of loss which is acceptable for the given application. If the threshold value is very small, then loss of data is very minimal and hence does not affect the accuracy of the final results. When threshold value is small , its compression ratio or reduction ratio is small. This is due to fact that, less number of replacement are carried out during preprocessing. When the threshold value is at higher side , the compression ratio or reduction ratio is higher , since segments are getting replace easily. so , it is up to the user to select threshold value. If the application is very critical and there should not be any loss of data , then user might select very small threshold value. If the application is not critical and loss of data may not degrade the performance of application, higher threshold value may be selected. Experimental results of MDL and RelativeLoss algorithms are shown in Table 3. There is very less amount of data loss in case of relativeloss method compared to MDL method.

V. CONCLUSION

MDL technique is prune to two main issues such as it cannot process close points of the trajectory and there is considerable amount of data loss happening during preprocessing. In order to eliminates these two shortfall of MDL technique we have proposed RegressionLine and RelativeLoss technique. In RegressionLine technique, the points which are very close to each other are identified and then this points are represented using regression line. The regression line which returns minimum error distance is selected as the representative line. In RelativeLoss technique we have minimized the loss of data due to replacement. User is given option to select threshold value , so that user can control the amount of loss in the preprocessing technique. We have tested RegressionLine and RelativeLoss technique theoretically and experimentally for correctness and accuracy. Our proposed technique is able to eliminates both the issues faced by MDL technique.

REFERENCES

[1] D. J. Berndt, J. Clifford, Finding patterns in time series: A dynamic programming approach, in: Advances in Knowledge Discovery and Data Mining, 1996, pp. 229–248.

[2] M. Vlachos, G. Kollios, D. Gunopulos, Discovering similar multidimensional trajectories, in: In ICDE, 2002, pp. 673–684.

[3] L. Chen, R. T. Ng, On the marriage of lp-norms and edit distance., in: M. A. Nascimento, M. T. zsu, D. Kossmann, R. J. Miller, J. A. Blakeley, K. B. Schiefer (Eds.), VLDB, Morgan Kaufmann, 2004, pp. 792–803.

[4] L. Chen, M. T. zsu, Robust and fast similarity search for moving object trajectories, in: In SIGMOD, 2005, pp. 491–502.

[5] G. Das, D. Gunopulos, H. Mannila, Time-series similarity problems and well-separated geometric sets, Nord. J. Comput. 8 (4) (2001) 409–423.

[6] D. Buzan, S. Sclaro, G. Kollios, Extraction and clustering of motion trajectories in video, in: 17th International Conference on Pattern Recognition, ICPR 2004, Cambridge, UK, August 23-26, 2004., 2004, pp. 521–524.

[7] M. Vlachos, D. Gunopulos, G. Kollios, Robust similarity measures for mobile object trajectories., in: DEXA Workshops, IEEE Computer Society, 2002, pp. 721–728.

[8] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, E. J. Keogh, Indexing multidimensional timeseries with support for multiple distance measures, VLDB J. 15 (1) (2006) 1–20.

[9] E. Keogh, K. Chakrabarti, M. Pazzani, S. Mehrotra, Dimensionality reduction for fast similarity search in large time series databases, JOURNAL OF KNOWLEDGE AND INFORMATION SYSTEMS 3 (2000) 263–286.

[10] M. D. Morse, J. M. Patel, An ecient and accurate method for evaluating time series similarity., in: C. Y. Chan, B. C. Ooi, A. Zhou (Eds.), SIGMOD Conference, ACM, 2007, pp. 569–580.

[11] C. Faloutsos, M. Ranganathan, Y. Manolopoulos, Fast subsequence matching in time-series databases (1994).

[12] B.-K. Yi, H. V. Jagadish, C. Faloutsos, Ecient retrieval of similar time sequences under time warping, in: S. D. Urban, E. Bertino (Eds.), Proceedings of the Fourteenth International Conference on Data Engineering, Orlando, Florida, USA, February 23-27, 1998, IEEE Computer Society, 1998, pp. 201–208.

[13] K.-P. Chan, A.W. chee Fu, Ecient time series matching by wavelets, in: In ICDE, 1999, pp. 126–133.

[14] B.-K. Yi, C. Faloutsos, Fast time sequence indexing for arbitrary lp norms., in: A. E. Abbadi, M. L. Brodie, S. Chakravarthy, U. Dayal, N. Kamel, G. Schlageter, K.-Y. Whang (Eds.), VLDB, Morgan Kaufmann, 2000, pp. 385–394.

[15] Y. Cai, R. Ng, Indexing spatio-temporal trajectories with chebyshev polynomials, in: Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data,

SIGMOD ’04, ACM, New York, NY, USA, 2004, pp. 599–610. doi:10.1145/1007568.1007636.

[16] E. J. Keogh, T. Palpanas, V. B. Zordan, D. Gunopulos, M. Cardle, Indexing large human-motion databases., in: M. A. Nascimento, M. T zsu, D. Kossmann, R. J. Miller, J. A. Blakeley, K. B. Schiefer (Eds.), VLDB, Morgan Kaufmann, 2004, pp. 780–791.

[17] E. Keogh, Exact indexing of dynamic time warping (2002).

[18] E. Keogh, T. Palpanas, V. B. Zordan, D. Gunopulos, M. Cardle, Indexing large human-motion databases, in: Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30, VLDB ’04, VLDB Endowment, 2004, pp. 780–791.

[19] M. Lichman, Uci machine learning repository (2013).