Noise Resilient Periodicity Mining In Time Series Data bases

K.Prasad Rao
Sr. Asst. Professor,Dept. of C.S.E
AITAM,Tekkali A.P., India

M.Gayathri
Dept. of Computer Science & Engineering
AITAM,Tekkali, AP, India

***Abstract*: One important aspect of the data mining is found out the interesting periodic patterns from time series databases. One area of research in time series databases is periodicity detection. Time series prognosticating is the use of a model to presage future values based on previously observed values. Discovering periodic patterns from an entire time series has proven to be inadequate in applications where the periodic patterns occur only with in small segments of the time series. In this paper we proposed a couple of algorithms to extract interesting periodicities (Symbol, Sequence, and Segment) of whole time series or part of the time series. One algorithm utilizes Dynamic warping Technique and other one is suffix tree as a data structure. Couple of these algorithms has been successfully incontestable to work with replacement, Insertion, deletion, or a mixture of these types of noise. The algorithm is compared with DTW (Dynamic Time Warping) algorithm on different types of data size and periodicity on both real and synthetic data. The worst case complexity of the suffix tree noise resilient algorithm is O ($n.k^2$) where n is the maximum length of the periodic pattern. K is size of the portion (whole or subsection) of the time series.**

Keywords–Time Series, suffix tree, periodicity detection, noise resilient, DTW

I. INTRODUCTION

A time-series database consists of sequences of values or events obtained over repeated measurements of time. The values are typically measured at equal time intervals (e.g., hourly, daily, weekly). Time-series databases are popular in many applications, such as stock market analysis, economic and sales forecasting, utility studies, inventory studies, yield projections, workload projections, process and quality control, Observation of natural phenomena (such as atmosphere, temperature, wind, earthquake), Time series prognosticating is the use of a model to presage future values based on previous observed values. Identifying periodic Patterns could divulge that grievous observations about the behavior and future trends of the Case Represented by the time series and hence would lead to more efficacious decision making.

In General, three types of periodic patterns can be espied in a time series:

1) **Symbol periodicity:** A time series is said to have symbol periodicity if at least one symbol is repeated periodically.

2) **Sequence or partial periodicity**: only a portion of the time series data is essential to the mining results.

3) **Segment or full-cycle periodicity**: In full-cycle periodicity, every point in the time series contributes part of the cycle.

A time-series database consists of sequences of values or events obtained over repeated measurements of time. The values are typically measured at equal time intervals (e.g., hourly, daily, weekly). Time-series databases are popular in many applications, such as stock market analysis, economic and sales forecasting, utility studies, inventory studies, yield projections, workload projections, process and quality control, observation of natural phenomena (such as atmosphere, temperature, wind, earthquake), scientific and engineering experiments, and medical treatments. Time series forecasting is the use of a model to predict future values based on previously observed values. Identifying periodic patterns could reveal that important observation s about the behavior and future trends of the case represented by the time series and hence would lead to more effective decision making.

General, three types of periodic patterns can be detected in a time series:

1) **Symbol periodicity:** A time series is said to have symbol periodicity if at least one symbol is repeated periodically.

2) **Sequence or partial periodicity**: only a portion of the time series data is essential to the mining results.

3) **Segment or full-cycle periodicity**: In full-cycle periodicity, every point in the time series contributes to part of the cycle

Contributions of our work can be summarized as follows:

1) The development of DTW (Dynamic Time Warping) algorithm that can detect only segment periodicity.

2) The development of suffix tree based comprehensive algorithm that can simultaneously detect Symbol, sequence and segment periodicity.

3) Finding periodicity within subsection of the series

4) Compare STNR with WARP.

5) Various experiments have been conducted to demonstrate the time efficiency, robustness, scalability, and accuracy of the reported results by comparing the proposed algorithm with existing state-of-the-art algorithms like WARP.

II. Existing System

Existing literature on time series algorithms requires user has to specify the period value or find out the candidate period value. Finding the candidate period value requires many database scans. Another way to classify existing algorithms is based on the type of periodicity they detect. Some detect symbol periodicity, some detect segment periodicity, and some detect only sequence periodicity. Segment periodicity currently used in many areas: handwriting and online signature matching, sign language recognition and gestures recognition, data mining and time series clustering (time series databases search)

- DTW algorithm detects only segment periodicity. It also Works well in the presence of insertion and deletion noise.
- Noise resilient algorithm detects all the three types of periodicity.

It requires only single scan of the data base. It also detect where the periodic occurrences can be shifted in an allowable range along the data base .It also detect where the periodic occurrences can be shifted in an allowable range along the time axis. This is somehow similar to how our algorithm deals noisy data by utilizing the time tolerance window for periodic occurrences.

III. Periodicity Detection Algorithms

Dynamic Time Warping (DTW) is a pattern matching technique used to compare time series, not necessarily of the same length, based on their characteristic shapes. It is common practice to constrain the warping path in a global sense by limiting how far it may stray from the diagonal of the warping matrix. The matrix subset is the warping band.

Noise resilient algorithm involves two phases. First build the suffix tree for the time series and in the second phase, we use the suffix tree to calculate the periodicity of various patterns in the time series. One important aspect of our algorithm is redundant period pruning.

A. DTW Representation

The DTW Algorithm calculates an optimal warping path between two time series. The algorithm starts with local distances Calculation between the elements of the two sequences using different types of distances. The most frequent used method for distance calculation is the absolute distance between the values of the two elements (Euclidian distance). DTW Algorithm is very useful for isolated words recognition in a limited dictionary. It is currently used in many areas: handwriting and online signature matching, sign language recognition and gestures recognition, data mining and time series clustering (time series databases search).

The Dynamic Programming part of DTW algorithm uses the DTW distance function.

$DTW(X, Y) = cp^*(X, Y) = \min \{cp(X, Y), p \in P^{N*M}$

Where P^{N*M} is the set of all possible warp paths. The WARP Algorithm builds the accumulated cost matrix for calculating the warping paths. To reduce the computational cost associated to the mining task, we consider the lower bounding technique introduced by Keogh et al. This technique aims at reducing the required number of DTW by the adoption of a lower-bounding measure.

B. Suffix Tree Based Representation

Suffix tree (also called PAT tree) is a data structure that presents the suffixes of a given string in a way that allows for a particularly fast implementation of a string in processing. Suffix tree is a famous data structure that has been proven to be very useful in string processing. The most popular use of the time series analysis is fore casting and control.

The main idea of this algorithm is:

1. Time series is encoded as string.
2. Find all the three types of periodicity in the whole time series or sub section of the time series.

There are algorithms for handling the disk based suffix tree. Suffix tree can be extended as new symbols are added to the string. Depth of the suffix tree is of order log (n).

C. Periodicity Detection in Presence of Noise

Real world data set obfuscate to find out the periodicities because the data contains noise. Three types of noise generally considered in time series data are replacement, insertion, and deletion noise. In replacement noise, some symbols in the time series are replaced at random with other symbols. In case of insertion and deletion of noise, some symbols are inserted or deleted, respectively.

T = abcde abdcec abcdca abdd abbca
01234 567890 123456 7890 12345;

The occurrence vector for pattern X = ab is (0, 5, 11, 17, 21)

Conf (p, stPos,X)=Actual Periodicity(P,Stpos,X)
Perfect Periodicity (P,stpos,X.)

Where Perfect Periodicity (p,stops,X)=|T|-stPos+1

Actual Periodicity (p,stops,X) is computed by counting (Starting at stP os and repeatedly jumping by p positions) the number of occurrences of symbol in time series(T).Conf (ab, p = 5, 0, 21) = 2/5. But, Conf (ab, 5, 0, 21, tt = 1) = 5/5. By considering the time tolerance is 1. The idea is that periodic occurrences can be drifted (shifted ahead or back) within a specified limit is time tolerance denoted as tt in the algorithm.

Algorithm: /* Noise Resilient Periodicity Detection*/

Begin

Step 1: Initialize occurrence vector.

Step 2: *Repeat*

Calculate difference vector (P).

Initialize the staring, ending position.

Step 3: *Repeat*

Calculate A, B, C value.

Verify the C value.

If C lies between ±1 increment P values.

Calculate periodicity values until end of the time series.

Step 4: *End.*

Step 5: calculate mean value.

Step 6: calculate confidence value.

Step 7: *End*

Algorithm 2 contains three new variables, namely A, B, and C. A represents the distance between the current occurrence and the current reference starting position; B represents the number of periodic values that must be passed from the current reference starting position to reach the current occurrence. Variable C represents the distance between the current occurrence and the expected occurrence. Suppose the occurrence vector is (0, 5, 11, 17, 23, 29, 35, 42) based on Algorithm 2, the period value would be p = 5, but the average period value is P = 5.85 ~ 6, which reflects more realistic information.

This section reports the results of the experiments that measure the time performance of STNR compared to WARP. We test the time behavior of the compared algorithms by considering two perspectives of varying data size, data distribution .we compare the performance of STNR against the WARP. This finds the periodic patterns for a specific period with uniform distribution containing 10 unique symbols and embedded period size of 25 and 32.The results of STNR are compared with those reported by WARP the curves are plotted in. Please see below page of this document for time performance of different data distribution. Where it can be seen that STNR gives better performance than WARP. We compare the performance of STNR against the WARP with different data distribution. STNR performs better than WARP because STNR applies various optimization strategies, most notably the adopted redundant period pruning techniques.

IV. Results

A. Time Performance

Table 1.1 time performance on different data size

SIZE	*STNR*	*WARP*
10,000	1995	6202
20,000	3895	9329
30,000	6795	9741
40,000	7695	9818
50,000	8795	9879

Fig.1 shows the comparison of time performance of STNR, DTW under different data size. As the data size increases the execution time of WARP also increases. But the time performance of STNR is doesn't increase. STNR performs better than WARP because STNR apply various optimization strategies for achieving the better performance.

Table 1.2 time performance(warp) on different data distribution

size	*time(25)*	*time(32)*
0.1	*1050*	*1050*
0.2	*2587*	*2589*
0.3	*5061*	*5061*
0.4	*8697*	*8957*
0.5	*10050*	*10062*

Table 1.3 time performance(STNR) on different data distribution

data size	*time(25)*	*time(32)*
0.1	30	32
0.2	37	39
0.3	57	65
0.4	71	81
0.5	90	110

Figure1:Time Performance of Different Data Distribution

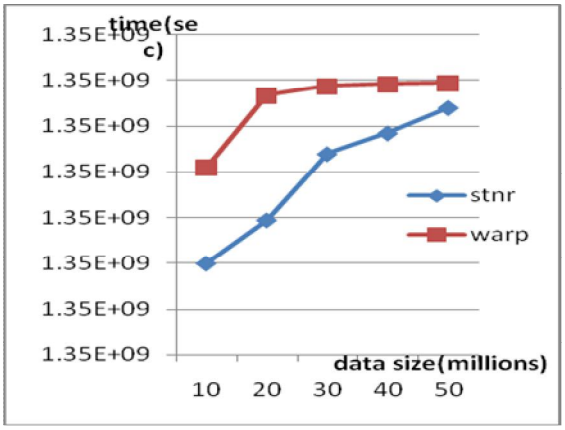
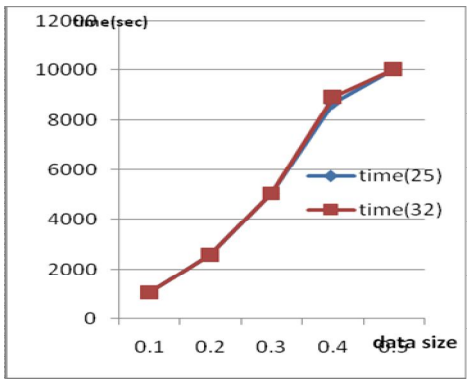


Fig.2 WARP under uniform distribution

Figure 3: STNR under uniform distribution



V. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented DTW Algorithm. Using Fourier transform, auto correlation functions to reduce the time complexity of the distance functions and STNR algorithm is a suffix-tree-based algorithm for periodicity detection in time series data. Our algorithm is noise-resilient. The single algorithm can find symbol, sequence (partial periodic), and segment (full cycle) periodicity in the time series. It can also find the periodicity within a subsection of the time series. We are also implementing the disk-based solution of the suffix tree to extend.

capabilities beyond the virtual memory into the available disk space. Using Fourier transform to reduce the complexity and wavelet transform is used to solve the problem in linear time.

REFERENCES

[1]. M. Ahdesma¨ki, H. La¨hdesma¨ki, R. Pearson, H. Huttunen, and O. Yli-Harja, “Robust Detection of Periodic Time Series Measured from Biological Systems,” BMC Bioinformatics, vol. 6, no. 117, 2005

[2]. C. Berberidis, W. Aref, M. Atallah, I. Vlahavas, and A. Elmagarmid, “Multiple and Partial Periodicity Mining in Time Series Databases,” Proc. European Conf. Artificial Intelligence, July 2002.

[3]. C.-F. Cheung, J.X. Yu, and H. Lu, “Constructing Suffix Tree for Gigabyte Sequences with Megabyte Memory,” IEEE

[4]. Trans. Knowledge and Data Eng., vol. 17, no. 1, pp. 90-105, Jan. 2005.

[5]. M. Dubliner “Faster Tree Pattern Matching,” J. ACM, vol. 14, pp. 205-213, 1994.

[6]. M.G. Elfeky, W.G. Aref, and A.K. Elmagarmid, “Periodicity Detection in Time Series Databases,” IEEE Trans. Knowledge and Data Eng., vol. 17, no. 7, pp. 875-887, July 2005.

[7]. M.G. Elfeky, W.G. Aref, and A.K. Elmagarmid, “WARP: Time Warping for Periodicity Detection,” Proc. Fifth IEEE Int’l Conf. Data Mining, Nov. 2005.

[8]. D. Gusfield, Algorithms on Strings, Trees, and Sequences. Cambridge Univ. Press, 1997.