International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 10, October- 2014

Study of Temporal Data Mining Techniques

Amit Doshi, Kashyap Bhansali, Prof. Lynette D'Mello
Dept. of Computer Engineering
D. J. Sanghvi College of Engineering
Mumbai, India

***Abstract*—Temporal data is data that includes time. Data mining problems can be classified into two categories: Data and Mining Operations. The main issue involved in data mining is processing data that encompasses temporal information. Temporal data mining has gained large momentum in the last decade. Various techniques of temporal data mining have been proposed. These techniques have been proved to be useful, to extract important information. In order to understand this phenomenon completely, we need to view temporal data as a sequence of events. Techniques from fields like machine learning, databases, statistics etc. are required when dealing with temporal data mining. In this paper, we provide a brief overview of temporal data mining techniques which have been developed in the last ten years.**

Keywords—TDM, Temporal Data, Temporal Data Mining, TDM techniques, SPADE, GSP.

I. INTRODUCTION

Owing to the increase in amount of stored data, a lot of interest in the discovery of hidden information has evolved in the last ten years. This discovery has been focused mainly on data clustering, data classification. Treating data having temporal attributes, is an important problem that arises during mining. These attributes need to be handled separately and differently from other attributes.

The goal of this paper is to give an overview of the techniques proposed within the last decade, to deal with temporal data mining. In addition to that, we aim to classify and organize them in such a way, that it helps to solve real life problems.

Temporal data mining is an important part of data mining. It is extraction of implicit, potentially useful and previously unspecified information, from large amount of data.

Temporal data mining deals with data mining of large sequential data sets. Sequential data is data that is ordered with respect to an index.

II. TEMPORAL DATA TYPES.

A. *Fully Temporal*

It is time dependent. Data and information derived from it are completely dependent on time. Ex: Transactional data in databases.

B. *Time Series*

This is a special case of time stamped data. It is similar to a number line. The events are uniformly separated in time dimension. Time series and temporal sequences, are seen in a variety of domains like engineering, research, medicine and finance.

C. *Time Stamped*

It has explicit information related to time. Temporal distance between data elements can be found. Inferences made can be temporal or non-temporal. Ex: data from stock exchange, inventory management.

D. *Sequences*

Sequences are ordered events with or without a concrete notion of time. Ex: customer shopping sequences, biological sequences. If an event appears before another, it means that the former event has occurred before the latter.

III. TEMPORAL DATA MINING GROUPS

Data mining has a wide range of applications. Tasks of data mining can be classified into some broad groups. In case of temporal data mining, these groups are Prediction, Classification, Clustering, Search and retrieval, Association. This categorization is not unique. Also, it is not exhaustive and does not cover all of the categories. In traditional time series analysis and pattern recognition, the first four categories have been extensively studied, understood and developed. In this section, a small overview of TDM techniques mentioned above is provided.

A. *Clustering*

Clustering groups the data on the basis of a similarity measure, like Manhattan distance, Euclidian distance. K-means, K-medoids are well-known clustering techniques.

Manhattan Distance:

$$d = \sum_{i=1}^{n} |x_i - y_i|$$

Euclidian Distance:

$$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Clustering of sequences, groups sequences based on their similarity measure. Clustering provides a mechanism to automatically find patterns in large data sets that would be otherwise difficult to find. In web activity logs, time series clustering proves quite useful where clusters can depict navigation patterns of users. There are various methods for clustering of sequences. We have model based sequence clustering methods like learning mixture models, a big class of model based clustering methods. For time series clustering,

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

variations and hybrid models of ARIMA (Autoregressive Integrated Moving Average) models and Hidden Markov Models used. Another broad class in clustering of sequences uses pattern alignment-based scoring. Some techniques use mixture of both model based and alignment based methods.

K-means clustering algorithm:

Input:

$D = \{t_1, t_2, \ldots, t_n\}$ //Set of elements

k //Number of desired clusters

Output:

K //Set of clusters

Algorithm:

Assign initial values for means $m_1, m_2, \ldots, m_k$;

Repeat

Assign each item t_i to the cluster which has the closest mean;

Calculate new mean for each cluster;

Until convergence criteria is met;

B. Classification

Classification is supervised learning. In classification, there are predefined classes into which the unknown set of attributes is classified. In temporal classification, given temporal sequence is assumed to belong to one of the predefined classes. This can be called as a training database. Using this training data, we can determine the class to which the given input belongs. Examples of sequence classification include gesture recognition, speech recognition, handwriting recognition, signature verification, etc. Speech recognition system converts audio/speech into text by recognizing the speech. In speech recognition, commonly used method is to divide the speech pattern and apply a feature extraction method on the divided parts. One application of pattern recognition is Optical Character Recognition (OCR) in which images are considered as sequences.

In signature verification, the input is a sequence of pixel coordinates drawn by the user. The task here is to classify each sequence according to its pattern. In gesture recognition, trajectories of motion and other features of objects are collected from the video/frames. Feature extraction step generates a sequence of vectors, for each pattern, that is then classified. Sequence classification methods can be pattern based or model based. In pattern based methods, training database is maintained for each pattern and class. The classifier then searches entire database, for the one that is most similar to pattern of the input. Sequence aligning methods like Dynamic Time Warping are used as similarity measures. There is another popular class of sequence recognition techniques which uses Hidden Markov Models (HMMs). It is a model based technique.

C. Association Rules

Apriori is an effective algorithm to mine association rules from a data set. In association rule mining we extract the relation among the attribute using parameters called support and confidence. Association rule mining can be applied to temporal association as well. In order to use Apriori algorithm to temporal data, some changes need to be made to the original algorithm. Support is the fraction of entities now, which has consumed the itemsets in the transactions. An entity can contribute one time to the support of each itemset. Large itemsets are identified. The itemsets with support greater than the minimum support are translated to an integer. Every sequence is changed to a new sequence having elements from large itemsets of the earlier sequence. In the next step, large sequences are found.

The algorithm first generates the candidate sequences. It then chooses the large sequences from the candidate sequences, until no candidate sequences are left. In Apriori algorithm, candidate generation is the most costly operation, as it suffers from combinatorial explosion. The general association rule X =>Y i.e. if X occurs then Y occurs, can be extended to a new rule X => tY i.e. if X occurs then Y will occur within time t. This new rule enables us to control the occurrence of an event to another event, within a time interval.

The problem of discovering association rules arises from the need to unearth patterns in transactional data. Transactional data is temporal as the time of the purchase is stored in the transaction when products are purchased. This is called transaction time. In large data volumes, required for data mining purposes, there may be some information of products that did not exist throughout the data gathering period. We can find some products that have been discontinued. There may also be new products which were introduced after the beginning of the gathering process. Some of these products must be included in the associations, but may not be included in any rule because of support restrictions. Hence, these new products would not appear in interesting and potentially useful association rules. To solve this problem, we integrate time in the model of discovery of association rules. These rules are called Temporal Association Rules.

D. Prediction

Prediction is an important problem in data mining. Prediction problems have some specific traits that distinguish them from other problems. There has been previous work in algorithms which can be used to predict time series evolution.

In prediction we forecast the future based on the data gathered in the past. In time-series prediction we predict future output of the time series using past data. An autoregressive family of models can predict a future value as a linear combination of sample values, given the condition that the time series is stationary. Models like ARIMA, which is a linear stationary model, have been found to be useful in various industrial and economic applications, where some suitable variant of the process is assumed to be stationary.

For non-stationary processes, the time series is assumed to be piece-wise stationary. This series is broken down into smaller parts called frames, within which, the stationary condition is assumed to hold and then separate models are learnt for every frame. In addition to the ARIMA family of models, there are many other nonlinear models for time series prediction like neural networks which are used for nonlinear modeling. The prediction problem is a part of Artificial Intelligence. Based on various rule models such as disjunctive

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

normal form model, periodic rule model etc. sequence-generating rules are found out that state some properties about the symbol which can appear next in the sequence.

Prediction has huge importance in fields like medicine, finance, environment & engineering.

E. Search and Retrieval

In searching, we aim to locate subsequences in large database of sequences in a single sequence, efficiently. To locate exact matches of substrings is a trivial problem, however to handle efficiency when looking for approximate matches is a difficult task. In data mining, we are more interested in approximate matching rather than exact string matching. When a query is given by the user, similar results must be given because the user might not be looking for exact results. We define similarity measures by considering distances between two corresponding sequences. Similarity measures like Euclidian distance or Manhattan distance can be used. Similarity measures based on DFT (Discrete Fourier Transform) and DWT (Discrete Wavelet Transform) have been discovered as well. Choice of similarity or dissimilarity measures is just one part of the sequence matching problem.

When we are determining similarity between two sequences, the sequences can be of different size. So it is not possible to calculate distances between corresponding sequences. Hence we use sequence alignment. We insert gaps in the two sequences or decide which should be corresponding elements in the given pair of sequences. For sequence classification and matching, time warping methods have been used. In speech applications, Dynamic Time Warping (DTW) is an efficient method that uses dynamic programming to identify correspondence among the vectors of two sequences to determine similarity between them. Symbolic sequence matching problems find applications in biological sequences such as proteins, genes, etc.

IV. TEMPORAL DATA MINING ALGORITHMS

The goal of temporal data mining is to find hidden relations between given sequence of events. An efficient approach to mining such relations is sequence mining. It involves three steps:

1) Transformation: converting given data into suitable form.

2) Similarity Measure: defining the similarity measure to be used.

3) Mining Operation: applying mining operation to get desired results.

We discuss some of the Sequence Mining algorithms.

A. GSP Algorithm

GSP stands for Generalized Sequential Pattern. It is used for sequence mining. It is based on the Apriori algorithm. We first discover all the frequent items level-wise by counting the occurrences of all singleton elements in the data set. The transactions are then filtered. Non frequent items are removed. After this step, each transaction consists of only the frequent elements. This is the input to the algorithm.

GSP Algorithm makes multiple passes. In the 1st pass, all single items are counted. A set of candidate 2-sequences are formed from the frequent items, and one more pass is made to find out their frequency. Candidate 3-sequences are generated from frequent 2-sequences. This process is repeated until no more frequent sequences are found.

Two main steps in the algorithm are:

Candidate Generation: The candidates for the next pass are generated by joining $F_{(k-1)}$ with itself. Pruning is done in order to eliminate any sequence at least one of whose subsequences is not frequent.

Support Counting: A hash-tree based search is used for counting support efficiently. Non-maximal frequent sequences are removed.

Algorithm:

F_1 = the set of frequent 1-sequence
k=2,
do while $F_{(k-1)}$!= Null;
Generate candidate sets C_k (set of candidate k-sequences);
For all input sequences s in the database D
do
Increment count of all a in C_k if s supports a
F_k = {a Є C_k such that its frequency exceeds the threshold}
k= k+1;
Result = Set of all frequent sequences is the union of all F_ks
End do
End do.

B. SPADE

SPADE stands for Sequential Pattern Discovery using Equivalence Classes. SPADE is based GSP. SPADE uses a vertically structured database. SPADE initiates from the bottom-most element of the lattice and works in a bottom-up fashion to generate all frequent sequences. It maintains the vertical structure as it proceeds from the less elements to more elements.

Algorithm:

```
SPADE (min_sup, D):
F1 = {frequent items or 1-sequences};
F2 = {frequent 2 sequences};
E = {Equivalence Class [X]θ1};
for all [X] Є E do Enumerate-Frequent-Seq([X]);

Enumerate-Frequent-Seq(S):
for all atoms Ai Є S do
        Ti = φ ;
        for all atoms Aj Є S with j>=i do
                R = Ai v Aj;
                if (Prune(R) == FALSE) then
                    L(R) =L(Ai ) ∩ L(Aj);
                if σ(R) >= min_sup then
                    Ti = Ti U {R}; F|R| = F|R| U {R};
        end
if(Depth-First-Search) then Enumerate-Frequent-Seq(Ti );
```


(This work is licensed under a Creative Commons Attribution 4.0 International License.)

```
end
if(Breadth-First-Search) then
        for all T_i  !=  φ do Enumerate-Frequent-Seq(T_i);

Prune (β):
for all (k-1)-subsequences, α <  β do
        if([ α_1] has been processed, and  α not Є F_(k-1) then
                return true;
        return false;
```

C. *Comparison between GSP and SPADE*

1) *GSP and SPADE Comparison.*

TABLE I. GSP AND SPADE COMPARISON

	GSP	**SPADE**
Purpose	It is used for extracting frequently occurring sequences.	It is used for fast discovery of sequential pattern.
Approach	Apriori Based	Apriori Based
Candidate Sequence	Candidate Sequence Are required to be generated.	Candidate Sequences are required to be generated.
Database Format	Uses Horizontal Format Database	Uses Vertical Format Database
Performance	1. Iterative algorithm 2. Makes multiple passes over the database depending on the length of the longest frequent sequences in database. 3. I/O cost is high if database has very long frequent Sequences.	1. Outperforms the GSP. 2. Excellent Scale up properties w.r.t parameters like number of input sequences, event size, size of potential maximal frequent sequences etc.
Speed	It is slower than SPADE	It is faster than GSP
Total Time	~143 ms	~81 ms
Frequent Sequence Count	~3920	~3920
Max Memory	8.194585571	8.40794158

CONCLUSION

We have studied and given an overview of techniques used for temporal data mining. TDM has an important use in many other areas of knowledge. There are a lot of TDM techniques, classified considering the similarity measures they used, their applications. A significant number of algorithms are present for TDM.

REFERENCES

[1] Han J, Kamber (2001), Data mining: Concepts and techniques (San Fransisco, CA: Morgan Kauffmann)

[2] Srivatsan Laxman and P S Sastry (2006), A survey of temporal data mining, Sadhana, Vol. 31, Part 2, pp. 173–198.

[3] Claudia M. Antunes and Arlindo L. Oliveira, Temporal Data Mining: an overview. Lecture Notes in Computer Science.

[4] Kanak Saxena (2009), Efficient Mining Weighted Temporal Association Rules. 2009 World Congress on Computer Science and Information Engineering, pp 421-425 IEEE Computer Society.

[5] A.K. Pujari (2007), Data Mining Techniques, University Press ISBN 8173713804.

[6] Wu Y-L, Agrawal D, Abbadi A E (2000), A comparison of DFT and DWT based similarity search in time series databases. Ninth Int. Conf. on Information and Knowledge Management, McLean,VA, pp 488–495.

[7] Han, J., Pei, J., Yin (2000), Y.: Mining Frequent Patterns without Candidate Generation. ACM SIGMOD Int. Conf. on Management of Data, pp 1-12.

[8] Mohammed J. Zaki (2001) : SPADE: An Efficient Algorithm for Mining Frequent Sequences. Kluwer Academic Publishers, Machine Learning 42, pp 31-60 2001

[9] Manika Verma, Dr Devarshi Verma (2014) : Sequential Pattern Mining: A Comparison between GSP, SPADE and Prefix SPAN. International Journal of Engineering Development and Research , Vol 2 Issue 3.

[10] Margaret H Dunham, Data Mining: Introductory And Advanced Topics Pearson Education ISBN 9788177587852.

[11] Wikipedia: http://en.wikipedia.org/wiki/GSP_Algorithm

IJERT

(This work is licensed under a Creative Commons Attribution 4.0 International License.)