© 2018 IJSRST | Volume 4 | Issue 5 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

Cluster Analysis of Temporal Data using Maximum Likelihood Estimation

Prof. Sweta C. Morajkar*, Durga Karapurkar
Computer Department, Don Bosco College of Engineering, Fatorda, Margao, Goa, India

ABSTRACT

Due to rapid growth of technologies, a large amount of data gets generated. The need arises to handle this data for retrieving and analyzing useful information. Clustering of temporal data has been explored using evolutionary clustering. However the time dimension associated with the record has not been considered. Traditional clustering algorithms usually focus on grouping data objects based on similarity function. However, if temporal dimension is incorporated, it allows to perform cluster analysis for evolving patterns. Temporal data clustering extends traditional clustering mechanisms and provides underpinning solutions for discovering the condensing information over the period of time. This paper proposes a methodology for clustering records based on time frame. The proposed methodology first clusters the records based on time frame. When a new query record comes, using maximum likelihood estimation we try to identify its true representative cluster. The assignment of query record to a particular cluster is based on the distance measure.

Keywords: Clustering, Temporal Data, Maximum Likelihood

I. INTRODUCTION

Temporal data plays an important role to store data at different time granularities. Temporal data stored in a temporal database is different from the data stored in non-temporal database in that a time period attached to the data expresses when it was valid or stored in the database. A first step towards a temporal database is to incorporate timestamp.

With the help of time stamping, database states can be obtained. One approach is that a temporal database may timestamp entities with time periods. Another approach is to identify entity values that change over time period.

Clustering is the unsupervised classification of observations into groups [1]. This problem has been identified in different contexts and disciplines. This reflects the usefulness of clustering in analyzing the data. A cluster is therefore a collection of objects which are similar between them and dissimilar to the objects belonging to other clusters. Clustering algorithms may be classified as Hierarchical methods, Partitioning methods and Density based methods. Cluster analysis allows us to find longitudinal patterns over a period of time. Many data sets contain temporal records [2]. Each record entry is associated with a timestamp and describes some aspect of entity. Analysis over such data is an important task since number of records belonging to a particular entity is huge. Temporal data mining provides constructive mechanism for representing time evolving data. The representation for temporal data changes over time therefore, appropriate techniques have to be provided to cope up with the problems of handling such data. Unlike static data, there is high level of dependency among data elements. Analysis over such data provides an effective and efficient way to discover the intrinsic structure and condense information over time. Usually two problems are considered in clustering analysis mainly model selection and grouping. In this paper, a temporal dimension for clustering is considered. A set of timestamped records are considered. Clustering on temporal dimension is

difficult task due to its dimensionality. Firstly, a set of clusters are obtained based on timestamp. Using maximum likelihood estimation, a set of probability values is calculated. For each query record, probability value for each of the cluster is calculated.

In the following sections this paper lists out the Motivation and the literature survey carried out. Later the design of proposed methodology is elaborated and discussed.

II. MOTIVATION

Temporal data plays an important role in representing past, present and future data. The amount of data storage for temporal data is very huge. The need arises how to efficiently cluster and search over this time evolving data. Given a timestamped collection of records that are associated with some events, mining such complex data is an important task. The challenge lies in handling research problems involving temporal data. Temporal data clustering is a difficult task because of its high dimensionality. As temporal data sets grow larger, there should be some efficient mechanisms to retrieve the data. The need arises to cluster time evolving data and querying the clusters to find the appropriate cluster representative. Most of the clustering algorithms work only on static data. When a new instance arrives, full clustering process has to be repeated. So for evolving data, a clustering mechanism should be such that it is performed in just one scan.

III. LITERATURE SURVEY

Temporal databases provide support for time evolving data. Data clustering is one of the most effective mechanisms to improve performance of database system. In [3], a new measure called "temporal Affinity" is incorporated. The functionality is based on selecting two objects based on query patterns. Certain canonical operators have been considered such as overlap, contains, precede, follow, as-of and within. The query patterns are categorized into reference queries, version scanning and historical queries. Clustering mechanism used in this is Clara clustering. The methodology is divided into 4 steps:

1) Data generation: Given a set of parameters, the temporal data objects are generated.

ii) Clustering of data objects: Using a clustering measure, similar data objects are clustered together.

iii) Query processing: Benchmark queries are processed based on temporal affinity measure and the temporal references are identified.

In [4], a Bayesian approach to temporal data clustering has been discussed. Using unsupervised techniques, this paper aims at discovering structure for temporal data. Supervised learning assumes labelled data with predefined classes whereas unsupervised learning assumes no information on class labels. It tries to deduce structure by partitioning the data into groups. Analysis over such groups is important to identify the longitudinal patterns. Representations for temporal data are usually classified into two types: piecewise and global. In piecewise, representation is obtained by partitioning temporal data into segments based on certain criteria. The obtained set of partitions is modelled into specific representation.

Some algorithms directly work on temporal data. Using temporal similarity measure, cluster analysis is done. Using dynamic time warping or dynamic models, clustering analysis provides finding similarity between two time series. Representation based algorithms convert temporal data clustering into static data clustering and tries to capture dependency among data elements. Any traditional algorithm can be further applied on this low dimensional data and clusters can be obtained.

Streaming algorithms for dynamic data also plays an important role in clustering evolving data. A stream can be considered as a sequence of timestamped record. Stream mining is a way to extract knowledge and find evolution of data over time. A number of methods have been used in order to transform this data. The transformed form can further be used for

data analysis. Variations of normal clustering algorithms such as classification, clustering can be used on stream data. Traditional clustering algorithms such as K-means [4], hierarchical clustering and DBSCAN only work on static data. These methods cannot work well for stream data in view of its indefinite data size. Evolutionary clustering frameworks for K-means and agglomerative clustering have been proposed [5].

In [6], a method for identifying and mining complex timestamp events has been proposed. Two algorithms have been proposed.T3 algorithm coverts a given problem into graph analysis problem. It automatically groups timestamps into meaningful clusters. After clustering, anomalies can be detected .Second one is MT3 algorithm which is used for multiresolution analysis. It considers time granularities such as month, day, year etc.

Discovery of moving clusters in case of temporal databases plays an important role. At each timestep, data point from one cluster might shift to other cluster. [7] Deals with identifying areas that remain dense in long period of time. Firstly it assumes data to be static and performs clustering on static data. It then tries to identify changes in cluster.

IV. BASIC CONCEPTS

One of the most important issues in cluster analysis is to evaluate the clustering results. Analysis performed over the data set should clearly reflect the original structure of data. Visualization technique is crucial verification of clustering results. For high dimensional space, it is difficult to visualize the clusters obtained after clustering.

A. K-medoid algorithm

This algorithm is applied on timestamp data points which are mapped onto domain. In this case, times stamped records are considered, e.g. Web log data.

B. Algorithm

1. Initialize: select k for n points in random order.
2. Assign each data point to closest medoid.
3. For each medoid w
   a. For each data point o which is non medoid
      i. Interchange w and o and compute the total cost of the configuration.
4. Select the one with lowest cost.
5. repeat steps 2 to 4 until there is no change in the medoid.

C. Maximum likelihood estimation

Maximum likelihood estimation is a parameter estimation method mostly applied in statistics. Given a set of independently identically distributed points, Maximum likelihood finds the real parameters by maximizing the probability.

We use concept of latent variables. Latent variable is the one which is not directly observable. Given n samples, label for each sample is a latent variable. Using probabilistic approach we try to maximize the likelihood.

V. PROPOSED METHODOLOGY

A sample data set used is web log data. Each record contains a timestamp when a particular event occurred.

A. Pre-processing data

Data pre-processing is an important step in data mining process. Pre-processing is mainly done in order to remove unwanted data. The data in the log files about the actions of the users can not be used for mining purposes in the form as it is stored. For this reason a pre-processing step must be performed before the pattern discovering phase.This is mostly done in

order to improve the efficiency. The timestamp attribute is normalized so that it becomes easier for cluster operation. After pre-processing, data is now ready for performing clustering.

B. Clustering

Before clustering the data set, timestamp attribute for each record will be mapped onto a particular space. After obtaining the time points, k-medoid clustering is applied in order to obtain the clusters.

C. Calculating probabilities

Two assumptions are made. The first assumption is that clusters do not overlap. The second assumption is that covariances within clusters are the same. We combine clustering method with maximum likelihood estimation in order to determine the cluster assignment.

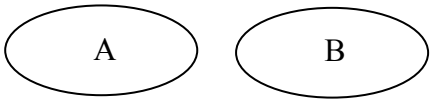


Figure 1:Non Overlapping Clusters

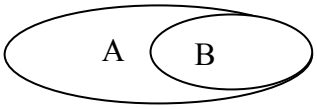


Figure 2 : Overlapping Clusters

Let Aj and S denote entities where aj represent an attribute of Aj:

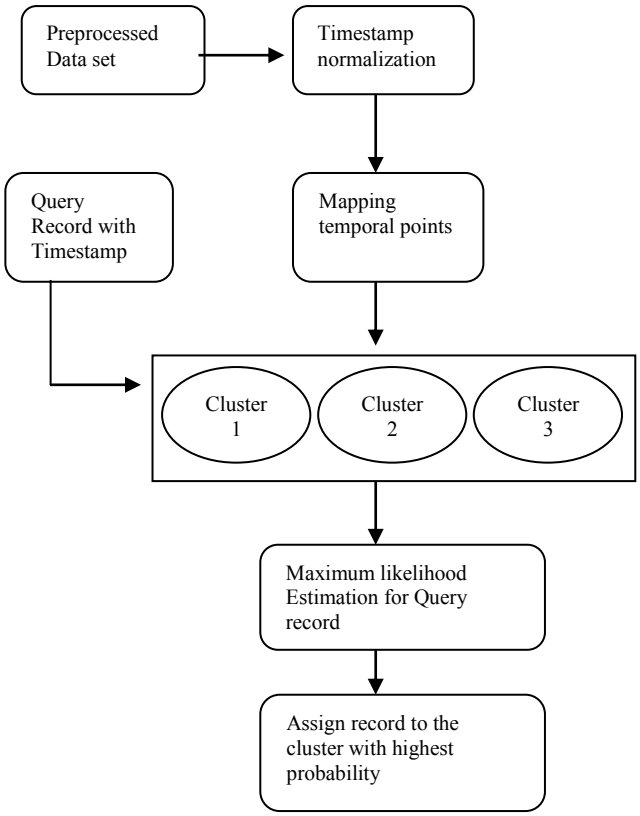
a. P(Aj, t| S) Probability that a query and a database record that actually refer to the same entity have an agreeing value in attribute aj over t.

b. P(¬Aj, t| ¬S) Probability that a query and a database record that actually refer to different entities have disagreeing (different) values in attribute aj over t.

The calculated set of probabilities is saved so that these can be used to adjust values when new record comes.

D. Searching cluster

The prior probabilities are stored. When a new query record comes, based on calculated set of probabilities, we calculate the distance of query record with each of the cluster.



VI. IMPLEMENTATION

Current clustering methods considered static data clustering. The number of clusters needs to be specified in advance. The Clustering technique is based on considering time frames. Existing clustering methods do not provide proper cluster. The proposed system is under implementation to obtain better clusters with maximum likelihood estimation.

VII. CONCLUSION

Temporal data provides promising techniques to cluster data based on temporal dimension. Using

maximum likelihood estimation technique increases the probability of a given record in a cluster. The paper proposes an approach to cluster time stamped data in an efficient manner. The approach first clusters the data based on time stamps, calculates probabilities and assigns the new record to the respective cluster.

VIII. REFERENCES

[1] A.K. Jain, M.N Murty, P.J. Flynn The Ohio State University ,"Data Clustering: A Review ",ACM Computing Surveys, Vol. 31, No. 3, September 1999

[2] Pei Li, Xin Luna Dong, Andrea Maurino, Divesh Srivastava,"Linking Temporal Records",Proceedings of the VLDB Endowment, Vol. 4, No. 11 Copyright 2011 VLDB Endowment 2150 8097/11/08

[3] Jong Soo Kim, Myoung Ho Kim, "On Effective Data Clustering in Bitemporal Databases" Temporal Representation and Reasoning, 1997. (TIME '97), Proceedings., Fourth International Workshop on Temporal data.

[4] Cen Li,Gautam Biswas A Bayesian Approach to Temporal Data Clustering using Hidden Markov Models International Conference on Machine Learning (ICML 2000)

[5] C. C. Aggarwal, J. Han, J. Wang, P. S. Yu. A framework for clustering evolving data streams. In Proc. of VLDB, 2003

[6] Hanghang Tong, Yasushi Sakurai, Tina Eliassi-Rad, Christos Faloutsos. Fast mining and forecasting of complex time-stamped events, CIKM'08, October 26–30, 2008, Napa Valley, California, USA. Copyright 2008 ACM 978-1-59593-991-3/08/10

[7] Marios Hadjieleftheriou, George Kollios, Dimitrios Gunopulos, Vassilis J. Tsotras, On-line discovery of dense areas in spatio-temporal databases. In: Proc. of SSTD. (2003)

[8] Clustering Data Stream: A survey of Algorithms, International Journal of Knowledge-based and Intelligent Engineering Systems, Volume 13 Issue 2, April 2009