A Novel Approach of Temporal Data Clustering via Weighted Clustering Ensemble with Different Representations

ALLA NAGA SRAVYA[#1],M.NALINI SRI[#2]

[#1]*B.TECH(1V/1V),E.C.M,K.L.University*
VADDESWARAM, INDIA.

[#2]*ASSISTANT PROFESSOR*

E.C.M, K.L.University
VADDESWARAM, INDIA

Abstract : **Temporal data clustering provides underpinning techniques for discovering the intrinsic structure and condensing information over temporal data. In this paper, we present a temporal data clustering framework via a weighted clustering ensemble of multiple partitions produced by initial clustering analysis on different temporal data representations. In our approach, we propose a novel weighted consensus function guided by clustering validation criteria to reconcile initial partitions to candidate consensus partitions from different perspectives, and then, introduce an agreement function to further reconcile those candidate consensus partitions to a final partition. As a result, the proposed weighted clustering ensemble algorithm provides an effective enabling technique for the joint use of different representations, which cuts the information loss in a single representation and exploits various information sources underlying temporal data. In addition, our approach tends to capture the intrinsic structure of a data set, e.g., the number of clusters. Our approach has been evaluated with benchmark time series, motion trajectory, and time-series data stream clustering tasks. Simulation results demonstrate that our approach yields favourite results for a variety of temporal data clustering tasks. As our weighted cluster ensemble algorithm can combine any input partitions to generate a clustering ensemble, we also investigate its limitation by formal analysis and empirical studies.**

Keywords: **Temporal data clustering, clustering ensemble, different representations, weighted consensus function, model.**

I. INTRODUCTION

The supervised classification or clustering provides an effective way to condensing and summarizing information conveyed in data, which is demanded by a number of application areas for organizing or discovering structures in data. The objective of clustering analysis is to partition a set of unlabeled objects into groups or clusters where all the objects grouped in the same cluster should be homogeneous. There are two core problems in clustering analysis; i.e., model selection and proper grouping. The former is seeking a solution that estimates the intrinsic number of clusters underlying a data set, while the latter demands a rule to group coherent objects together to form a cluster. From the perspective of machine learning, clustering analysis is an extremely difficult supervised learning task since it is inherently an ill-posed problem and its solution often violates some common assumptions. There have been many researches in clustering analysis, which leads to various clustering algorithms categorized as partitioning, hierarchical, density-based, and model-based clustering algorithms. Actually temporal data is a collection of observations associated with information such as the time at which data has been captured and the time interval during which a data value is valid. Temporal data is composed of a sequence of nominal symbols from the alphabet known as a temporal sequence, The use of temporal data has become widespread in recent years and temporal data mining continues to be a rapidly evolving area of inter-related disciplines including statistics, temporal pattern recognition, temporal databases, optimization, visualization, high-performance computing and parallel computing.

However, the recent empirical studies in temporal data analysis reveal that most of the existing clustering algorithms do not work well for temporal data due to their special structure and data depend, which presents a big challenge in clustering temporal data of a various and high dimensionality, large volume, very high feature correlation and a substantial amount of noise. Recently, several studies have attempted to improve clustering by combining multiple clustering solutions into a single consolidated clustering ensemble for better average performance among given clustering solutions. This has led to many real world applications, including gene classification, image segmentation. Clustering ensembles usually involve two stages. First, multiple partitions are obtained through several runs of initial clustering analysis. Subsequently, the specific consensus function is used in order to find a final consensus partition from multiple input

partitions. In this thesis, we are going to concentrate on weighted clustering ensembles and its application for temporal data clustering tasks based on three methodologies: the model-based approach and the feature-based approach.

The model-based approach aims to construct statistical models to describe the characteristics of each group of temporal data, providing more intuitive ways to capture dynamic behaviours and a more flexible means for dealing with the variable lengths of temporal data. In general, the entire temporal dataset is modeled by a mixture of these statistical models, whilst an individual statistical model such as Gaussian distribution is used to model a specific cluster of temporal data. Model-based approaches for temporal data clustering.

The feature-based approach is indirect temporal data clustering, which begins with the extraction of a set of features from raw temporal data, so that all temporal data can be transformed into a static feature space. Then, classical vector-based clustering algorithms can be implemented within the feature space.

Obviously, feature extraction is the essential factor that decides the performance of clustering. Generally, feature-based clustering reduces the computational complexities for higher dimensional temporal data.

A. PROBLEM STATEMENT

Although the clustering algorithms have been intensively developing for last decades, due to the natural complexity of temporal data, we still face many challenges for temporal data clustering tasks. How to select an intrinsic number of cluster is still a critical model selection problem existed in many clustering algorithms.. In other words, the model selection problem is solved by optimizing the pre-defined criterion. For common model selection criterion, Akaike's information criterion, balances the good fit of a statistical model based on maximum log-likelihood and model complexity based on the number of model parameters. The optimal number of clusters is selected with a minimum value of AIC. In each run, the training set is used to estimate the best-fitting parameters while the testing set computes the model's error. The optimal number of clusters is selected by a posteriori probabilities or criterion function. Recently. It treats the supervised learning problem as a problem of estimating the joint distribution between the observable pattern in the observable space and its representation pattern in the representation space. In theory, the optimal number of clusters is given by the minimum value of cost function.

Performance of these different criterions depends on the structure of the target Cancer.dataset dataset and no single criterion emerges as outstanding when measured against the others. Moreover, a major problem associated with these model selection criterions also remains: the computation procedures involved are extremely complex and time consuming.

How to significantly reduce the computational cost is another importance issue for temporal data clustering task due to the fact of that temporal data are often collected in a dataset with large volume, high and various dimensionality, and complex clustered structure. From the perspective of model-based temporal data clustering, and proposed a model-based hybrid partitioning-hierarchical clustering. and its variance such as hierarchical meta-clustering. In the first approach, one is a improved version of model-based agglomerative clustering, which keeps some hierarchical structure. However associating with K-models clustering, the complexity of input data to the agglomerative clustering

II.OBJECTIVE OF THE RESEARCH

The work presented in this thesis concentrates on supervised classification/clustering tasks for temporal clustering via weighted clustering using k-means algorithm, temporal data from the model-based approach, & the feature-based approach respectively. The problems summarized above are addressed in association with supervised ensemble learning techniques.

Firstly, K-Means is an important model-based approach for temporal data clustering, has been studied. We propose a novel approach based on the ensemble of partitioning clustering associated with hierarchical clustering refinement in order to solve problems in finding the intrinsic number of clusters and model initialization problems which exist in most model-based clustering algorithms.

Secondly, a feature-based approach to temporal data clustering is proposed through a weighted ensemble of a simple clustering algorithm with minimum user-dependent parameters, such as k-means with different representations, in order to address both proper grouping with minimum computational cost and selecting an intrinsic number of clusters as model selection problems in clustering analysis as a whole. This proposed approach takes into account the diversity of groupings generated by certain clustering algorithm , initialization to reconcile them in an optimal way. Furthermore, the proposed weighted consensus function not only enables automatic model selection for clustering analysis, but also provides a generic technique for the optimal solution of combining multiple partitions

III.PARTITIONAL CLUSTERING

Partitioning clustering directly divides the data sets into several subsets, where each subset represents a cluster containing at least one data. In general, the partition is hard or crisp if each data belongs to exactly one cluster, or soft or fuzzy if one data is allowed to be in more than one cluster at a different degree, where each cluster is represented by a prototype and assigns the patterns to clusters according to most similar prototype. As well known K-means algorithm and its modified version of K-medoids algorithm are quite popular partitional clustering algorithms, where each cluster is represented by either the mean value of the data in the cluster or the most centrally located data in a cluster. Two counterparts for fuzzy partitions are the fuzzy c-means algorithm and the fuzzy c-medoids algorithm Actually there are many possible

outputs obtained by partitioning the data sets into several groups, partitional clustering algorithms always attempt to achieve a desire result by optimizing a criterion function such as square-error, which is defined either globally or locally. These heuristic algorithms work well for finding spherical-shaped clusters and small to medium data sets, but they always reveal the weakness of analyzing the complex structured data such as temporal data.

IV.K-MEANS ALGORITHM

It is one of the simplest partitional clustering algorithms, and commonly used for solving temporal data clustering problem, directly applied K-means as a proximity-based approach to multivariate battle simulation temporal data with the objective to form a discrete number of battle states, indirectly applied K-means as feature-based approach for analyzing time series based on the its wavelet-based representation. The procedure of K-means follows a simple way to classify a given temporal dataset through a certain number of clusters (assume K clusters) fixed a priori, which consists of the following steps:

1. Place K seed points into the representation space obtained from the data sets that are being clustered. These points represent initial groups.
2. Assign each data point to the group that has the closest seed point.
3. When all data points have been assigned, recalculate the positions of the K seed points.
4. Repeat Steps 2 and 3 until the seed points no longer move.

This produces a separation of the entire data sets into groups known as clusters The entire process can be formulated by minimizing an objective function

$$\sum_{k=1}^{K}\sum_{x\in C_k} D(x, C_k)$$

where D is distance metric based on the meaningful objective to compute the distance between a data point x belonging to cluster k and representative point of the cluster C_K such as center of clustered data points.

Although the K-means can be proved that the procedure will always terminate for temporal data clustering task, this algorithm does not necessarily find the most optimal solution, corresponding to the local minimum of objective function, and sensitivity to the initialization and selected number of seed points as number of clusters. Moreover, by directly applying k-means it requires the temporal data with equal length because the concept of cluster centers would be ill-defined when the individual one is represented in arbitrary length in the target dataset.

It accepts the number of clusters to group data into, and the dataset to cluster as input values.

It then creates the first K initial clusters (K= number of clusters needed) from the dataset by choosing K rows of data randomly from the dataset. For Example, if there are 10,000 rows of data in the dataset and 3 clusters need to be formed, then the first K=3 initial clusters will be created by selecting 3 records randomly from the dataset as the initial clusters. Each of the 3 initial clusters formed will have just one row of data.

The K-Means algorithm calculates the Arithmetic Mean of each cluster formed in the dataset. The Arithmetic Mean of a cluster is the mean of all the individual records in the cluster. In each of the first K initial clusters, there is only one record. The Arithmetic Mean of a cluster with one record is the set of values that make up that record. For Example if the dataset we are discussing is a set of Height, Weight and Age measurements for students in a University, where a record P in the dataset S is represented by a Height, Weight and Age measurement, then P = {Age, Height, Weight). Then a record containing the measurements of a student Prasanna for example , would be represented as Prasanna = {20, 170, 80} where Prasanna's Age = 23 years, Height = 1.70 meters and Weight = 60 Pounds. Since there is only one record in each initial cluster then the Arithmetic Mean of a cluster with only the record for prasanna as a member = {20, 170, 80}.

It Next, K-Means assigns each record in the dataset to only one of the initial clusters. Each record is assigned to the nearest cluster (the cluster which it is most similar to) using a measure of distance or similarity like the Euclidean Distance Measure or Manhattan/City-Block Distance Measure.

It K-Means re-assigns each record in the dataset to only one of the new clusters formed. A record or data point is assigned to the nearest cluster (the cluster which it is most similar to) using a measure of distance or similarity like the Euclidean Distance Measure or Manhattan/City-Block Distance Measure.

The preceding steps are repeated until stable clusters are formed and the K-Means clustering procedure is completed. Stable clusters are formed when new iterations or repetitions of the K-Means clustering algorithm does not create new clusters as the cluster center or Arithmetic Mean of each cluster formed is the same as the old cluster center. There are different techniques for determining when a stable cluster is formed or when the k-means clustering algorithm procedure is completed.

V.ENHANCED K-MEANS ALGORITHM

1.Spatial databases has a huge amount of data collected and stored.(increases the need for effective analysis methods).

1. **Cluster analysis** is a primary data analysis tasks .
2. **Goal of enhancement**: Improve the computational speed of the *k*-means algorithm.

3. Using simple **data structure(e.g. Arrays)** to keep some information in each iteration to be used in the next iteration.

4. **K-Means**: Computes the distances between data point and all centers(computationally very expensive).

5. **Function *distance_new*()**

//assign each point to its nearest cluster
1 For *i*=1 to *n*
Compute squared Euclidean distance
***d*2(*xi*, *Clustered*[*i*]);**
If (*d*2(*xi*, *Clustered*[*i*])<=*Pointdis*[*i*])
Point stay in its cluster;
2 Else
3 For *j*=1 to *k*
4 Compute squared Euclidean distance
***d*2(*xi*, *mj*);**
5 end for
6 Find the closest centroid *mj* to *xi*;
7 *mj*=*mj*+*xi*; *nj*=*nj*+1;
8 *MSE*=*MSE*+*d*2(*xi*, *mj*);
9 *Clustered*[*i*]=number of the closest centroid;
10 *Pointdis*[*i*]=Euclidean distance to the closest centroid;
11 end for
12 For *j*=1 to *k*
13 *mj*=*mj* /*nj*;
14 end for

So K-Means Complexity : ***O*(*nkl*)**.
where ***n* is the number of points**, ***k* is the number of clusters** and ***l* is the number of iterations**.
If the point stays in its cluster this require *O*(1), otherwise require *O*(*k*).
If we suppose that half points move from their clusters, this
requires *O*(*nk*/2),
since the algorithm converges to local minimum, the number of points moved from their clusters decreases in each iteration.
So we expect the total cost is *nk* Σi=1toL 1/ *i* . Even for large number of iterations, *nk* Σi=1toL 1/ *i* is much less than *nkl*.
Enhanced *k*-means algorithm Complexity : ***O*(*nk*)**.

A. Hierarchical clustering

Also known as a tree of clusters or a dendrogram, hierarchical clustering builds a cluster hierarchy in which every cluster node contains child clusters. These sibling clusters partition the points covered by their common parent allowing for the exploration of temporal data on different levels of granularity. In the early work, they just performed an agglomerative hierarchical clustering of daily power consumption data based on the root mean square distance. Hierarchical clustering methods are generally categorized as either agglomerative (bottom-up) or divisive (top-down). Agglomerative clustering begins with one-point (singleton) clusters and recursively merges two or more appropriate clusters. Divisive clustering begins with one cluster of all data points and recursively splits the most appropriate cluster. The process continues until a stopping criterion (frequently, the requested number K of clusters) is achieved. Theoretically, divisive hierarchical clustering is unfeasible because the possible divisions of data into two clusters at the first step of the algorithm are quite various. Therefore, in most applications, divisive hierarchical clustering is rarely applied which generally restricts attention to agglomerative hierarchical clustering.
So finally K-Means algorithm introduced by J.B. MacQueen, is one of the most common clustering algorithms and it is considered as one of the simplest unsupervised learning algorithms that partition feature vectors into k clusters so that the within group sum of squares is minimized.

B. Density-based clustering

Density-based clustering algorithms are designed to find the arbitrary-shaped clusters in data sets, a cluster is defined as a high-density region, which exceeds a threshold, separated by low-density regions in data space. Density-Based Spatial Clustering of Applications with Noise, DBSCAN is a typical density-based clustering algorithm. The basic idea of DBSCAN is to iteratively grow a cluster of data points as long as the density in the "neighborhood" exceeds some threshold. Rather than producing a clustering explicitly, Ordering Points To Identify the Clustering Structure, OPTICS, computes an augmented cluster ordering for automatic and interactive cluster analysis. The ordering contains information that is equivalent to density-based clustering obtained from a wide range of parameter settings, thus overcoming the difficulty of selecting parameter values. For analyzing time series contained significant noise, Density-based clustering has been typically applied by Denton , to identifies and removes this noise by only considering clusters rising above a preset threshold in the density landscape. proposed a density-based hierarchical clustering (DHC) to tackle the problem of effectively clustering time series gene expression data, where all objects in a data set are organized into an attraction tree according to the density based connectivity. Then, clusters are identified by dense areas.

C. Model-based clustering

In model-based clustering, we use certain models for clusters, and each cluster can be mathematically represented by a parametric model, such as HMM or ARMA. The entire dataset is therefore modeled by a mixture of these component models. An individual model used to represent a specific cluster that is often referred to as a probability distribution. A large amount of literatures . have shown that the model-based approach has been widely used for time series clustering analysis.

VI.SOFTWARE AND HARDWARE REQUIREMENTS

A.SOFTWARE REQUIREMENTS:

Operatingsystem :WindowsXP sp2,WindowsProfessional

Front End :Oracle

Back End Tool :Myeclipse10

B.HARDWARE REQUIREMENTS:

SYSTEM : Pentium IV 2.4 GHz

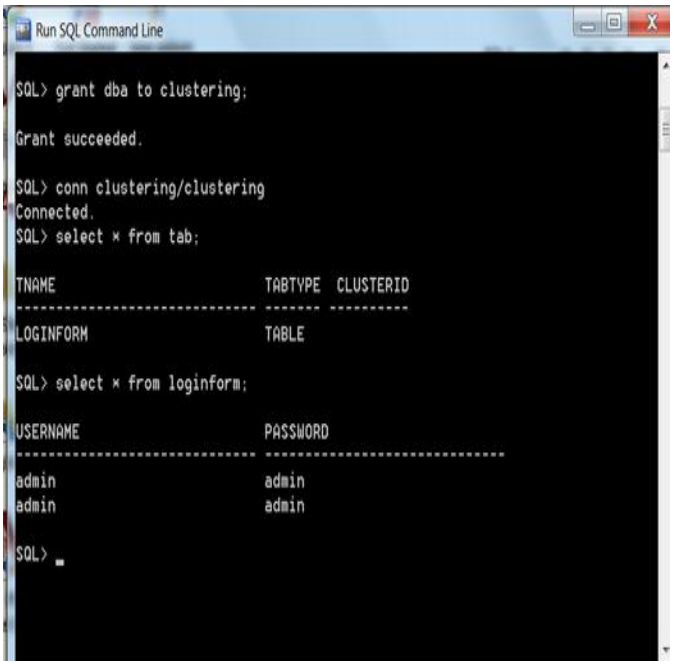
HARD DISK : 40 GB

RAM : 256 MB

KEYBOARD : 110 keys enhanced

MONITOR : 15 VGA colour

MOUSE : Logitech



This Oracle/SQL provides a detailed introduction to the SQL query language and the Oracle Relational Database Management System. Further information about Oracle and SQL.In relational database systems (DBS) data are represented using tables (relations). A query issued against the DBS also results in a table. A table has the following structure:
Column 1 Column 2 . . . Column n
A table is uniquely identified by its name and consists of rows that contain the stored information, each row containing exactly one (or record). A table can have one or more columns. A column is made up of a column name and a data type, and it describes an attribute of the tuples. The structure of a table, also called relation schema, thus is dened by its attributes.
The type of information to be stored in a table is dened by the data types of the attributes at table creation time.
SQL uses the terms table, row, and column for relation, tuple, and attribute, respectively. In this we will use the terms interchangeably.
A table can have up to 254 columns which may have dierent or same data types and sets of values (domains), respectively. Possible domains are alphanumeric data (strings), numbers and date formats. Oracles the following basic data types:
char(n): Fixed-length character data (string), n characters long. The maximum size for n is 255 bytes (2000 in Oracle8). Note that a string of type char is always padded on right with blanks to full length of n. (can be memory consuming).

A.Weighted Clustering

The temporal data In our approach, we propose a weighted consensus function guided by clustering validation criteria to reconcile initial partitions to candidate consensus partitions from different perspectives and then introduce an agreement function to further reconcile those candidate consensus

partitions to a final partition. As a result, the proposed weighted clustering ensemble algorithm provides an effective enabling technique for the joint use of different representations, which cuts the information loss in a single representation and exploits various information sources underlying temporal data. In addition, our approach tends to capture the intrinsic structure of a data set, e.g., the number of clusters in cancer.data file.

In this section, we first describe our motivation to propose our temporal data clustering model. Then, we present our temporal data clustering model working on Single representations via weighted clustering ensemble learning.

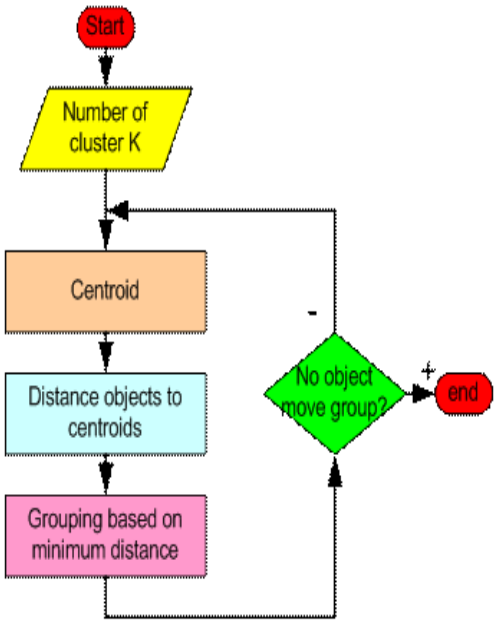


Figure : Flow of proposed work

VIII.CONCLUSION,FUTURE WORK

This thesis has principally focused on K-Means algorithm the fundamental problems described introduction chapter for temporal data clustering via Weighted Clustering data tasks in close association with clustering ensemble learning techniques. As described earlier, there are three methodologies for temporal data clustering; model-based clustering & feature-based clustering. Each approach favors differently structured temporal data or types of temporal data with certain assumptions. There is nothing universal that can solve all problems and it is important to understand the characteristics of both clustering algorithms and the target temporal data, so that the right approach can be selected for a given clustering problem. However, there are very limited amounts of prior information for most clustering problems, making the selection of a proper clustering algorithm for certain characteristics of temporal data extremely difficult.

1. No parameter re-estimation is required for the new merged pair of clusters, significantly reducing computation costs, which has been typically justified in a similar model-based hybrid clustering approach proposed by k-means algorithm .
2. In comparison of single model such as hybrid partitional-hierarchical clustering, the composite model is better equipped to characterize complexly structured clusters in order to produce a robust and accurate clustering results, which has been demonstrated on a various temporal datasets including K-Means generated dataset of a general synthetic dataset.
3. The model initialization problem is solved by implementing the ensemble technique, which has been typically investigated by a experimental study , the higher averaged classification accuracy with smaller standard deviation obtained by our proposed approach just demonstrated its insensitivity to model initialization

IX.REFERENCES

[1] Bagnall, A., Ratanamahatana, C. A., et al. (2009). "A bit level representation for temporal data mining with shape based similarity." Data Mining and Knowledge Discovery 13(1): 11-40.

[2] Cheng, H., Hua, K. A., et al. (2008). "Constrained locally weighted clustering." Proceedings of the VLDB Endowment 1(1): 90-101.

[3] Ghaemi, R., Sulaiman, M. N., et al. (2009). A Survey: Clustering Ensembles Techniques. World Academy of Science, Engineering and Technology.

[4] Yang, Y. and Chen, K. (2010). supervised Learning via Iteratively Constructed Clustering Ensemble. Proceedings of International Joint Conference on Neural Networks Barcelona, Spain.

[5] Azimi, J., Abdoos, M., et al. (2007). A new efficient approach in clustering ensembles.