A NOVEL APPROACH FOR PATTERN ANALYSIS FROM HUGE DATAWAREHOUSE

BABITA[1]
Department of Computer Science
Mewar University.

PARAMJEET RAWAT[2]
Department of Computer Science
UP Technical University

PARVEEN KUMAR[3]
Department of Computer Science
UP Technical University

***Abstract* -Due to the tremendous growth of data and large databases, efficient extraction of required data has become a challenging task. This paper propose a novel approach for knowledge discovery from huge unlabeled temporal databases by employing a combination of HMM and K-means technique. We propose to recursively divide the entire database into clusters having similar characteristics, this process is repeated until we get the cluster's where no further diversification is possible. Thereafter, the clusters are labeled for knowledge extraction for various purposes.**

Keywords : Temporal database, knowledge discover, pattern analysis, HMM, K-means.

I. INTRODUCTION

In recent years, data-mining (DM) has become one of the most valuable tools for extracting and manipulating data and for establishing patterns in order to produce useful information for decision-making. Breakthroughs in data-collection technology, such as bar-code scanners in commercial domains and sensors in scientific and industrial sectors, getting larger storage devices at cheaper cost has led to the generation of huge amounts of data [1]. This tremendous growth in data and databases has spawned a pressing need for new techniques and tools that can intelligently and automatically transform data into useful information and knowledge. As a consequence, the discovery or extraction of information in a data store is becoming an increasingly challenging task. For example, finding information on the World Wide Web, the largest known source of digital information storage, has become possible through efficient search engine designs. The scalability of web search engines is achieved through the distribution and parallelization of the search task, and through algorithms which scale at most linearly with the size of a dataset.

Much more challenging is the task of finding patterns in large sets of data. This exercise is commonly referred to as Data Mining. Data Mining is actually a process that consists of applying data analysis and discovery algorithms that, under acceptable computational efficiency limitations, produce a particular enumeration of patterns (or models) over the data" [2]. Another, sort of pseudo definition; "The induction of understandable

models and patterns from databases". In other words, we initially have a large (possibly infinite) collection of possible models (patterns) and (finite) data. Data Mining should result in those models that describe the data best, the models that fit (part of the data).

For analyzing the data we keep a record of events of past activities in a system. For example, sales by a salesmen are logged electronically at branch office. The records stored normally include a name or description of the item, the sales price, the location of the sale, the time of sale and name, ID number, of the salesperson, the identity of the buyer, etc. The analysis of such logged data can be utilized to analyze a variety of aspects concerning the system. For example, the analysis of data logs can help to draw a conclusion about the performance of a salesperson, peak season of sales etc. It can also serve as a marketing tool to help identify successful marketing strategies, or to draw conclusions about user behaviors. In other words, data mining often involves the task of discovering knowledge from unlabeled data.

This paper addresses knowledge discovery in a data mining environment. In this work, we present a new method in discovering patterns from a large set of unlabeled temporal data. The k-means and hidden Markov model (HMM) form the core of our proposed approach.

II. LITERATURE REVIEW

Partitioning clustering algorithms, such as K-means [3] and CLARA [4] assign objects into k (predefined cluster number) clusters, and iteratively reallocate objects to improve the quality of clustering results. K-means is the most popular and easy-to-understand clustering algorithm [3]. However, K-means algorithm is very sensitive to the selection of the initial centroids and there is no general theoretical solution to find the optimal number of clusters for any given data set.

An approach given by CURE[6] uses a set of representative points to describe the boundary of a cluster in its hierarchical algorithm. But with the increase of the complexity of cluster shapes, the number of representative points increases dramatically in order to maintain the precision. CHAMELEON [5] employs a multilevel graph partitioning algorithm on the k-Nearest Neighbour graph, which may produce better results than CURE on complex cluster shapes for spatial datasets. But the high complexity of the algorithm prevents its application on higher dimensional datasets.

Many grid-based methods have been proposed, such as STING (Statistical Information Grid Approach) [WYM97], CLIQUE [11], and the combination of grid-density based technique WaveCluster [12]. The grid-based methods are efficient on clustering data with the complexity of O(N). However the primary issue of grid-based techniques is how to decide the size of grids. This quite depends on the user's experience.

DENCLUE (DENsity-based CLUstEring) is a distribution-based algorithm [9], which performs well on clustering large datasets with high noise. Also, it is significantly faster than existing density-based algorithms, but DENCLUE needs a large number of parameters.

Then we have Model-based clustering methods, that are based on the assumption, that data are generated by a mixture of underlying probability distributions, and they optimize the fit between the data and some mathematical model, for example statistical approach, neural network approach and other AI approaches. The typical techniques in this category are Autoclas [8], DENCLUE [9] and COBWEB [10]. When facing an unknown data distribution, choosing a suitable one from the model based candidates is still a major challenge. On the other hand, clustering based on probability suffers from high computational cost, especially when the scale of data is very large. Another most commonly used model is HMM model, the Hidden Markov Model (HMM) is a tool based on statistical modeling. The proven capabilities of the HMM to encode temporal patterns have made this approach a very popular to pattern discovery applications. A number of variants of HMMs exist. These include discrete HMM, continuous observation HMM, and input-output HMM, to name a few. The most commonly used model is the Hidden Markov model (HMM), e.g., [13,14,15]. Other models used include Autoregressive Moving Average model (ARMA) [16], Latent Variables Model [17], and others.

III. PROPOSED ALGORITHM

In this work, we present a new method in discovering patterns from a large set of unlabeled temporal data. K-means (KM) and hidden Markov model (HMM) form the core of our proposed approach. These methods are used to cluster temporal data by using a recursive KM-HMM model. This novel recursive KM-HMM model works as follows:

It will start with the entire dataset, and will partition the dataset into clusters on the basis of their similarity to one another using the KM technique. These clusters are initially unlabelled clusters, containing patterns which are similar to one another. Then, each cluster is considered one by one, and on the basis of the context of each pattern, HMM is used to label them into different labelled sets. Then each labelled set is considered to find sub clusters of similar patterns thereof. This process of applying KM followed by HMM will be repeated for each cluster, until the clusters contain all similar patterns with same label. The result of this recursive process is a hierarchical model[18], where the number of levels depends upon the data. It is a data driven approach, with little a priori assumption about the number of levels in the hierarchical model, or the unlabelled data itself. It is conceivable that the method can be parallelized in that an independent machine can take care of a cluster, and can use the KM and HMM repeatedly to further partition the data, thus overcoming one of the major issues involved in using HMM in such applications, as the HMM will only operate on a partition of the data, rather than the entire dataset.

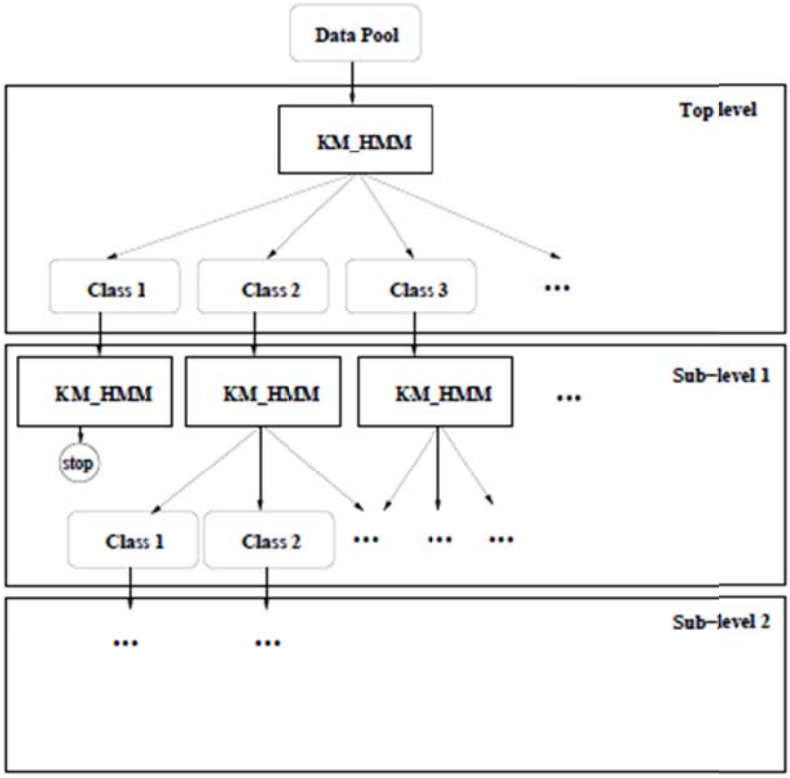

Fig 1. Recursive Refinement

A main contribution of our work is the introduction of a hierarchal HMM for the purpose of pattern discovery in temporal data. This has been achieved through a recursive application of K-means(KM) clustering and HMMs which effectively combines known positive aspects of popular K-means and hidden Markov models in clustering a large set of temporal data. HMM is employed because it is a very successful approach for temporal pattern discovery tasks. The challenge is to use HMM in an efficient way since HMMs are expensive models to train and generally not suitable for mining large set of data. The efforts taken to reduce the impact of the data size on the training of HMMs include:

1. a suitable segmentation of the dataset into cohorts, and
2. the limitation of the size of the training dataset. An iterative procedure assures that over time, all data in a given dataset is eventually considered by the model.

A main benefit of our proposed approach is that an automated procedure is introduced which detects redundancies in the dataset. These redundancies are not explicitly removed but cause an early termination of the algorithm, and hence, reduce the computational demand. For each cohort, a hierarchical clustering is initialized by applying KM to group profiles into cluster, as shown in the figure 1.

The number of clusters can be fixed a priori, i.e. to allow control by an experienced user. These clusters are then considered as annotated automatically by KM thus serve as a basis for training a HMM on each cluster. This can be performed in parallel. This training would be expensive if all the profiles in the cluster are to be considered by

HMM at the same time. This is addressed by selecting only a portion of the profiles in a cluster for processing by a HMM.

This can obviously weaken the model since the HMMs receive only partial information about the problem domain. An iterative training procedure is proposed which re-shuffles the training data, and re-trains the HMMs until convergence is observed. This iterative procedure allows an exposure of the HMMs to as many data profiles as necessary to achieve convergence. Hence, if a dataset contains redundant data then convergence will be observed sooner, reducing the overall computational demand by avoiding the processing of all data items in a cohort.

In our algorithm, the training of HMMs starts only when there are more than one valid clusters remaining. The first round training of HMMs is then guided by these R0- Clusters producing a set of trained HMMs: the R1-HMMs. These are then used to conduct data classification yielding a set of classes: the R1-Classes. The classification is assessed to determine if next iteration training of HMMs is needed. Obviously if the classification has converged or the maximum number of iteration is met, the iterative mining stops, otherwise, it continues into the next iteration.

ALGORITHM

1. Perform Data Clustering
2. If valid_number_of_cluster>1 then
    a. Perform Data Modeling: HMM training
    b. Data Classification
    c. If Converged or Max reached, then
        Goto Step 4
      Else
        Goto Step 2
      Endif
3. Endif
4. End

IV. BENEFIT OF ALGORITHM

A hierarchical clustering is obtained by refining the number of clusters after the previous iterative mining step. This causes clusters to be diversified into sub clusters until no further diversification is possible (or until a given threshold is reached). This produces a number of clusters as decided from the data rather than as from user specification. A recursive approach is taken by inferring this number from data through iterative levels. KM is applied to each of the classes obtained at the end of the iterative process at a previous iteration, which are now referred to as classification results from the Top level. If KM fails in re-grouping the profiles in the class into clusters, this implies that all the profiles in a given cluster are considered structurally similar, thus no further refinement is needed. If KM succeeds, it is clear that some dissimilarity still exists among profiles in the same

class. These new clusters become the basis for training another sets of HMMs, thus commences another round of iterative training process. When all the classes have been processed at a top level, then the process is applied recursively to each of the clusters. Each iteration is referred to a sub level n, where n is the depth of a given iteration (the iteration number). This recursive process refines the clustering until either the algorithm converges or when the data clusters become smaller than a prescribed threshold.

In summary, the contributions of this paper are:

A hierarchical approach in clustering temporal data.
Iterative training process results in robust models (HMMs) and at the same time reduces the expense of training HMM in a data mining application.
Recursive refinement of the clustering results determine the number of clusters required for the application domain in an automated fashion.

V. CONCLUSION

A novel recursive hierarchical approach has been introduced to tackle the two research issues of our temporal data mining task:big data and the lack of labeling. The core of our methodology is a model based clustering which is enabled by a data clustering (KM) and a data modeling (HMM). The novelty of is, the modelling has been implemented iteratively to overcome the weakness of employing HMMs on large datasets. Instead of modeling the data in one go by learning all the profiles in the training set at once, iterative mining is employed to model patterns in a controlled manner so that the training times of HMMs at each iteration is kept within acceptable limits while the models are more and more refined through the iterations. In future we will prove the benefit of our proposed model with suitable simulation model.

REFERENCES

[1] R. Chaiken, B. Jenkins, P. Larson, B. Ramsey, D. Shakib, S. Weaver, and Zhou. SCOPE: Easy and Efficient Parallel Processing of Massive Data Sets.PVLDB, 1(2):1265–1276, 2008.
[2] Abdelmelek, S.B. Saidane, S. Trabelsi, M. Base oils Biodegradability Prediction with Data Mining Techniques, Algorithms 3:92-99, 2010.
[3] J. B. MacQueen, “Some Methods for classification and Analysis of Multivariate Observations”, Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, pp.281-297 (1967)
[4] L. Kaufman and P. J. Rousseeuw, “Finding Groups in Data: an Introduction to Cluster Analysis”, John While & Sons. (1990)
[5] G. Karypis, E.-H. S Han, and V. Kumar, “Chameleon: hierarchical clustering using dynamic modeling,” IEEE Computer, vol. 32(8), pp.68–75 (1999)
[6] S. Guha, R. Rastogh and K. Shim, “CURE: An efficient clustering algorithm for large databases,” Proceedings of ACM SIGMOD Conference 98, pp.73–84 (1998)
[7] D. DeWitt, E. Robinson, S. Shankar, E. Paulson, J. Naughton, A. Krioukov, and J. Royalty. Clustera: An Integrated Computation and Data Management System. PVLDB, 1(1):28–41, 2008.
[8] P. Cheeseman, J. Kelly, M. Self, J. Stutz, W. Taylor, and D. Freeman, “AutoClass: A bayesian classification system”, Proceedings of 5th International Conference on Machine Learning, Morgan Kaufmann, pp. 54-64 (1988)
[9] A. Hinneburg and D. Keim, "An Efficient Approach to Clustering in Large Multimedia Databases with Noise”, Proceedings of KDD-98 (1998)
[10] D. Fisher, “Improving Inference through Conceptual Clustering”, Proceedings of 1987 AAAI Conferences, Seattle Washington, pp.461-465 (1987)
[11] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, “Automatic subspace clustering of high dimensional data for data mining applications”, Proceedings of the ACM SIGMOD Conference, Seattle, WA., pp.94-105 (1998)
[12] G. Sheikholeslami, S. Chatterjee, A. Zhang, “Wavecluster: A multi-resolution clustering approach for very large spatial databases”, Proceedings of Very Large Databases Conference (VLDB98), pp.428-439 (1998)
[13] A. Panuccio, M. Bicego, and V. Murino. A hidden markov model-based approach to sequential data clustering. In *Proceedings of Joint IAPR International Workshops SSPR 2002 and SPR 2002*, pages 734–743, 2002.

[14] M. P. Perrone and S. D. Connell. K-means clustering for hidden markov models. In *Proceedings of the 7th InternationalWorkshop on Frontiers in Handwriting Recognition*, pages 229–238, 2000.
[15] P. Smyth. Clustering sequences with hidden markov models. *Advances in Neural Information Processing Systems*, Vol.9:648–654, 1997.
[16] Y. Xiong and D. Y. Yeung. Mixtures of arma models for model-based time series clustering. In *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM'02)*, pages 717–720, 2002.
[17] Z. X. Ying and J. H. Chiang. Pattern discovery on complex diagnosis and biological data using fuzzy latent variables. In *IEEE 23rd International Conference on Data Engineering, 2007. ICDE 2007*, pages 576–585, 2007.
[18] M. Athanassoulis, S. Chen, A. Ailamaki, P. B. Gibbons and R. Stoica. MaSM: Efficient Online Updates in Data Warehouses. In Proc. of SIGMOD, 2011.
[19] D. Jiang, A. K. H. Tung, and G. Chen. Map-join-reduce: Towards Scalable and Efficient Data Analysis on Large Clusters. TKDE, 23(9):1299–1311, 2010.
[20] A.R. Post and J.H. Harrison. Temporal data mining. *Clinics in Laboratory Medicine*, 28(1):83–100, 2008. S. Han, D. Chen, M. Xiong, and A. K Mok. Online Scheduling Switch for Maintaining Data Freshness in Flexible Real-Time Systems. In Proc. of RTSS, pp. 115–124, 2009.