Efficient Mining of Association Rules Using Closed High Utility Item Set Lattice

R.Radhika[1], K.Sudhakar2

[1]R.Radhika, M.E/CSE, Ganadipathy Tulsi's Jain Engineering College,Vellore, India -632102.
[1]radhika24.r@gmail.com
[2]K,Sudhakar,Asst Prof/CSE, Ganadipathy Tulsi's Jain Engineering College,Vellore, India – 632102.
[2]vksudhakar@gmail.com

ABSTRACT

Mining high utility item sets from a transactional database refers to the discovery ofitem sets with high utility like profits. Although a number of relevant algorithms have been proposed in recent years, they incur the problem of producing a large number of candidate itemsets for high utility itemsets. Such a large number of candidate item sets degrades the mining performance in terms of execution time and space requirement. The situation may become worse when the database contains lots of long transactions or long high utility itemsets. In this paper, we propose two algorithms, namely utility pattern growth (UP-Growth) and UP-Growth+, for mining high utility itemsets with a set of effective strategies for pruning candidate itemsets.The information of high utility itemsets is maintained in a tree-based data structure named utility pattern tree (UP-Tree) such that candidate itemsets can be generated efficiently with only two scans of database. The performance of UP-Growth and UP-Growth+ is compared with the state-of-the-art algorithms on many types of both real and synthetic data sets. Experimental results show that the proposed algorithms, especially UP-Growth+, not only reduce the number of candidates effectively but also outperform other algorithms substantially in terms of runtime, especially when databases contain lots of long transactions.

***Keywords*—High utility itemset ,UP-Tree**

1.INTRODUCTION

It is widely recognized that a large number of features can adversely affect the performance of inductive learning algorithms, and clustering is not an exception. However, while there exists a large body of literature devoted to this problem for supervised learning task, feature selection for clustering has been rarely addressed. The problem appears to be a difficult one given that it inherits all the uncertainties that surround this type of inductive learning. Particularly, that there is not a single performance measure widely accepted for this task and the lack of supervision available.

1.1 Feature Selection

In machine learning and statistics, feature selection, also known as variable selection, attribute selection or variable subset selection, is the process of selecting a subset of relevant features for use in model construction. The central assumption.

When using a feature selection technique is that the data contains many redundant or irrelevant features. Redundant features are those which provide no more information than the currently selected features, and irrelevant features provide no useful information in any context. Feature selection techniques are a subset of the more general field of feature extraction.

Feature extraction creates new features from functions of the original features, whereas feature selection returns a subset of the features. Feature selection techniques are often used in domains where there are many features and comparatively few samples (or data points). The archetypal case is the use of feature selection in analyzing DNA microarrays, where there are many thousands of features, and a few tens to hundreds of samples. Feature selection techniques provide three main benefits when constructing predictive models.

With such an aim of choosing a subset of good features with respect to the target concepts, feature subset selection is an effective way for reducing dimensionality, removing irrelevant data, increasing learning accuracy, and improving result comprehensibility. Irrelevant features, along with redundant features, severely affect the accuracy of the learning machines. Thus, feature subset selection should be able to identify and remove as much of the irrelevant and redundant information as possible. Moreover, "good feature subsets contain features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other."

Many feature subset selection methods have been proposed and studied for machine learning applications. They can be divided into four broad categories: the Embedded, Wrapper, Filter, and Hybrid approaches.

1.2 APPROACH

1.2.1 Embedded approach

The embedded methods incorporate feature selection as a part of the training process and are usually specific to given learning algorithms, and therefore may be more efficient than the other three categories. Traditional machine learning algorithms like decision trees or artificial neural networks are examples of embedded approaches.

1.2.2 Wrapper Method

Wrapper methods are widely recognized as a superior alternative in super- vised learning problems, since by employing the inductive algorithm to evaluate alternatives they have into account the particular biases of the algorithm. How- ever, even for algorithms that exhibit a moderate complexity, the number of executions that the search process requires results in a high computational cost, especially as we shift to more exhaustive search strategies. The wrapper methods use the predictive accuracy of a predetermined learning algorithm to determine the goodness of the selected subsets, the accuracy of the learning algorithms is usually high. However, the generality of the selected features is limited and the computational complexity is large.

.

1.2.3 Filter Method

The filter methods are independent of learning algorithms, with good generality. Their computational complexity is low, but the accuracy of the learning algorithms is not guaranteed.

1.2.4 Hybrid Approach

The hybrid methods are a combination of filter and wrapper methods by using a filter method to reduce search space that will be considered by the subsequent wrapper. They mainly focus on combining filter and wrapper methods to achieve the best possible performance with a particular learning algorithm with similar time complexity of the filter methods.

In cluster analysis, graph-theoretic methods have been well studied and used in many applications. Their results have, sometimes, the best agreement with human performance. The general graph-theoretic clustering is simple: compute a neighborhood graph of instances, then delete any edge in the graph that is much longer/shorter (according to some criterion) than its neighbors. The result is a forest and each tree in the forest represents a cluster. In our study, we apply graph-theoretic clustering methods to features. In particular, we adopt the minimum spanningtree (MST)-based clustering algorithms, because they do not assume that data points are grouped around centers or separated by a regular geometric curve and have been widely used in practice.

Based on the MST method, we propose a Fast clustering based feature Selection algorithm (FAST). The FAST algorithm works in two steps. In the first step, features are divided into clusters by using graph-theoretic clustering methods. In the second step, the most representative feature that is strongly related to target classes is selected from each cluster to form the final subset of features.

Features in different clusters are relatively independent; the clustering based strategy of FAST has a high probability of producing a subset of useful and independent features. The proposed feature subset selection algorithm FAST was tested various numerical data sets. The experimental results show that, compared with other five different types of feature subset selection algorithms, the proposed algorithm not only reduces the number of features, but also improves the classification accuracy.

1.3APPLICATIONS OF FAST CLUSTERING DATASET

1.3.1 Statistics Applications

Modern statistics deals with large and complex data sets, and consequently with models containing a large number of parameters. This book presents a detailed account of recently developed approaches, including the Lasso and versions of it for various models, boosting methods, undirected graphical modeling, and procedures controlling false positive selections.

1.3.2 Scalability and usability

The clustering technique should be fast and scale with the number of dimensions and the size of input. It should be insensitive to the order in which the data records are presented. Finally, it should not presume some canonical form for data distribution. Current clustering techniques do not address all these points adequately, although considerable work has been done in addressing each point separately.

1.3.3 Subspace Clustering Application

Empirical evaluation shows that CLIQUE scales linearly with the size of input and has good scalability as the number of dimensions in the data or the highest dimension in which clusters are embedded is increased. CLIQUE was able toaccurately discover clusters embedded in lower dimensional subspaces, although there were no clusters in the original data space.

2.OVERVIEW OF EXISTING SYSTEM

Feature subset selection can be viewed as the process of identifying and removing as many irrelevant and redundant features as possible. This is because,

1) Irrelevant features do not contribute to the predictive accuracy, and

2) Redundant features do not redound to getting a better predictor for that they provide mostly information which is already present in other feature(s).

Of the many feature subset selection algorithms, some can effectively eliminate irrelevant features but fail to handle redundant features, yet some of others can eliminate the irrelevant while taking care of the redundant features. Our proposed FAST algorithm falls into the second group.

3.PROPOSED SYSTEM

Quite different from these hierarchical clustering-based algorithms, our proposed FAST algorithm uses minimum spanning tree-based method to cluster features. Meanwhile, it does not assume that data points are grouped around centers or separated by a regular geometric curve. Moreover, our proposed FAST does not limit to some specific types of data.
Irrelevant features, along with redundant features, severely affect the accuracy of the learning machines. Thus, feature subset selection should be able to identify and remove as much of the irrelevant and redundant information as possible. Moreover, “good feature subsets contain features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other.” Keeping these in mind, we develop a novel algorithm which can efficiently and effectively deal with both irrelevantand redundant features, and obtain a good feature subset. We achieve this through a new feature selection framework which composed of the two connected components of irrelevant feature removal and redundant feature elimination. The former obtains features relevant to the target concept by eliminating irrelevant ones, and the latter removes redundant features from relevant ones via choosing representatives from different feature clusters, and thus produces the final subset.

4.DESIGN

The design which improves efficiency and effectiveness of features to form clusters. Select a subset of relevant features by using feature selection technique. Eliminate irrelevant and redundant features from relative ones via choosing representatives from different feature clusters and thus produces the final feature subset in which dimensionality is drastically reduc The high dimensional datas are taken as an input from the cancer data set. Perform entropy calculation in which we have to compute entropy and conditional entropy using some mathematical formulas. Perform F-correlation method. Construct a minimum spanning tree (MST) by using the F-Correlation values which we obtain Eliminate irrelevant and redundant features from relative ones via choosing representatives from different feature clusters.

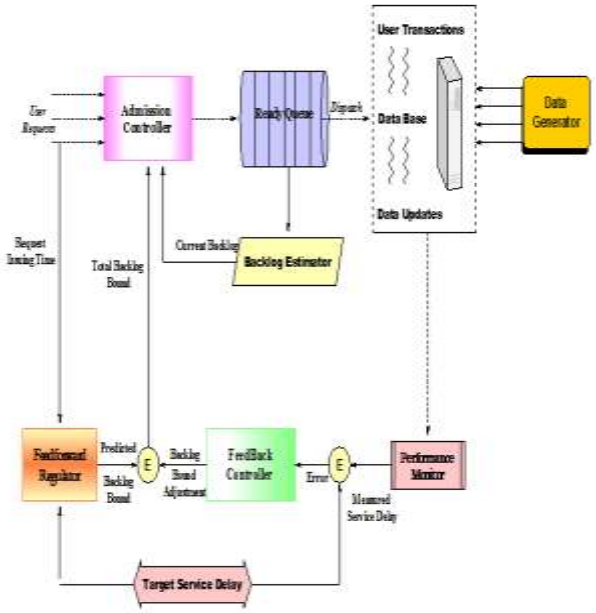

Fig:Overall Process Of System Design

Thus produces the final feature subset in which dimensionality is drastically reduced.

5.EXPERIMENT

5.1 Load Data and Classify

Load the data into the process. The data has to be preprocessed for removing missing values, noise and outliers. Then the given dataset must be converted into the arff format which is the standard format for WEKA toolkit. From the arff format, only the attributes and the values are extracted and stored into the database. By considering the last column of the dataset as the class attribute and select the distinct class labels from that and classify the entire dataset with respect to class labels.

Fig:Load And Classify

5.2 Information Gain Computation

Relevant features have strong correlation with target concept so are always necessary for a best subset, while redundant features are not because their values are completely correlated with each other. Thus, notions of feature redundancy and feature relevance are normally in terms of feature correlation and feature-target concept correlation.

Fig2:Information Gain Computation

To find the relevance of each attribute with the class label, Information gain is computed in this module. This is also said to be Mutual Information measure. Mutual information measures how much the distribution of the feature values and target classes differ from statistical independence. This is a nonlinear estimation of correlation between feature values or feature values and target classes. The symmetric uncertainty (SU) is derived from the mutual information by normalizing it to the entropies of feature values or feature values and target classes, and has been used to evaluate the goodness of features for classification

The symmetric uncertainty is defined as follows:

$$Gain(X|Y) = H(X) - H(X|Y)$$
$$= H(Y) - H(Y|X).$$

To calculate gain, we need to find the entropy and conditional entropy values. The equations for that are given below:

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x).$$

$$H(X|Y) = -\sum_{y \in Y} p(y) \sum_{x \in X} p(x|y) \log_2 p(x|y).$$

Where p(x) is the probability density function and p (x|y) is the conditional probability density function.

5.3 T-Relevance Calculation

The relevance between the feature Fi € and the target concept C is referred to as the T-Relevance of Fi and C, and denoted by SU (Fi,C). If SU (Fi,C) is greater than a predetermined threshold , we say that Fi is a strong T-Relevance feature.

$$SU(X,Y) = \frac{2 \times Gain(X|Y)}{H(X) + H(Y)}.$$

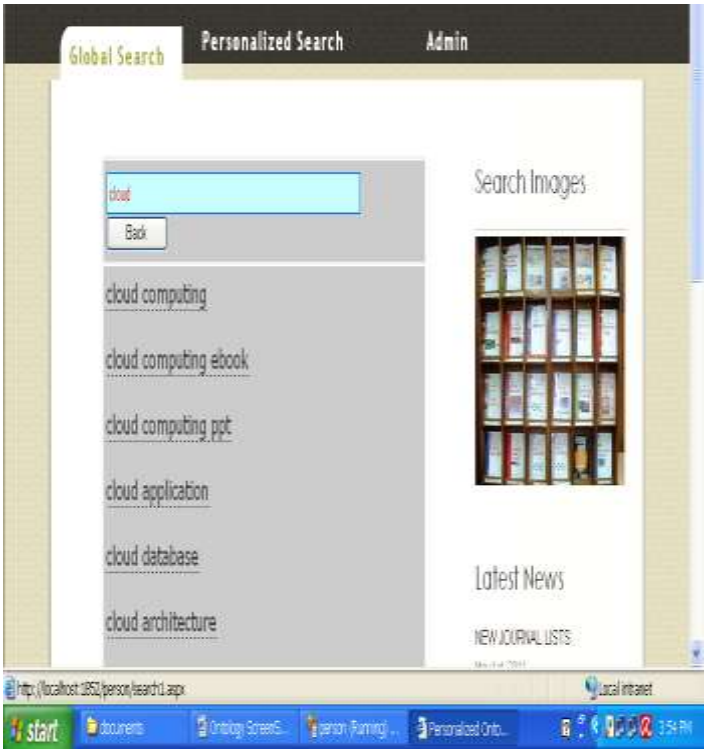

Fig-3 T-Relevance Calculation

After finding the relevance value, the redundant attributes will be removed with respect to the threshold value.

6.CONCLUSION

In this paper, we have presented a novel clustering-based feature subset selection algorithm for high dimensional data. The algorithm involves 1) removing irrelevant features 2) redundant attributes will be removed based on the threshold value we get from T-Relevance computation. In the proposed algorithm, a cluster consists of features. Each cluster is treated as a single feature and thus dimensionality is drastically reduced. In future work, we are going to compute F-Correlation and construct a Minimum Spanning Tree (MST) by using the correlated value we obtain.Partition MST and select representative patterns. It is more effective and improves speed and accuracy of learning algorithms.

7.FUTURE WORK

A search query interface allows a user to search the desired data by selecting the options that describes the items of his/her interest. This phase of proposed prototype automatically detects the domain specific search interfaces by looking at domain ontology. To support keyword based searching, query processing method has been proposed in this phase that splits the user query into keywords then searches these keywords in inverted index for attribute name. As soon as attribute name corresponding to keyword is found, SQL query is automatically generated using template for that attribute and respective keyword (value).

Security Implementation using one-time password. Email notification regarding on topic knowledge. User knowledge analysis and ontology mining using big data.

Finally the Graphical user interface is designed for user interaction where the user can fill the query in the form of keywords and find the desired result in integrated form.

REFERENCES

[1] H. Almuallim and T.G. Dietterich(2005), “Algorithms for Identifying Relevant Features,” Proc. Ninth Canadian Conf. Artificial Intelligence, pp. 38-45.

[2] H. Almuallim and T.G. Dietterich(2004), “Learning Boolean Concepts in the Presence of Many Irrelevant Features,” Artificial Intelligence, vol. 69, nos. 1/2, pp. 279-305.

[3] A. Arauzo-Azofra, J.M. Benitez, and J.L. Castro(2004), “A Feature Set Measure Based on Relief,” Proc. Fifth Int’l Conf. Recent Advances in Soft Computing, pp. 104-109.

[4] S. Bandyopadhyay, A. Mukhopadhyay, and U. Maulik(2007), “An improvedalgorithmfor clustering gene

expression data," *Bioinformatics*, vol. 23,no. 21, pp. 2859–2865.

[5] S. Bandyopadhyay, R. Mitra, and U. Maulik (2010), "Development of the human cancer microRNA network," *BMC Silence*, vol. 1, no. 6.

[6] L.D. Baker and A.K. McCallum(2000), "Distributional Clustering of Words for Text Classification," Proc. 21st Ann. Int'l ACM SIGIR Conf. Research and Development in information Retrieval, pp. 96-103.

[7] R. Battiti(2002), "Using Mutual Information for Selecting Features in Supervised Neural Net Learning," IEEE Trans. Neural Networks, vol. 5, no. 4, pp. 537-550.

[8] D.A. Bell and H. Wang(2000), "A Formalism for Relevance and Its Application in Feature Subset Selection," Machine Learning, vol. 41, no. 2, pp. 175-195.

[9] L. Bruzzone and M. Marconcini(2006), "An advanced sem-isupervised SVM classifier for the analysis of hyperspectral remote sensing data," in *Proc. Image Signal Process. Remote Sens. XII*,, pp. 636.

[10] A. Dupuy and R. M. Simon(2007), "Critical review of public microarray studies in cancer outcome and guidelines on statistical analysis and reporting," *J. Natl. Cancer 1*, vol. 99, pp. 147–157.

[11] J. Pei, J. Han, H. Lu, S. Nishio, S. Tang, and D. Yang, "H-mine: Fast and space-preserving frequent pattern mining in large databases," IIE Trans., vol. 39, no. 6, pp. 593–605, Jun. 2007.

[12] B.-E. Shie, H.-F. Hsiao, V. S. Tseng, and P. S. Yu, "Mining high utility mobile sequential patterns in mobile commerce environ- ments," in Proc. Int. Conf. Database Syst. Adv. Appl., 2011, vol. 6587, pp. 224–238.

[13] B.-E. Shie, V. S. Tseng, and P. S. Yu, "Online mining of temporal maximal utility itemsets from data streams," in Proc. Annu. ACM Symp. Appl. Comput., 2010, pp. 1622–1626.

[14] V. S. Tseng, C.-W. Wu, B.-E. Shie, and P. S. Yu, "UP-Growth: An efficient algorithm for high utility itemset mining," in Proc. ACM SIGKDDInt.Conf.Knowl.Discov.DataMining,2010,pp.253–262.

[15] B. Vo, H. Nguyen, T. B. Ho, and B. Le, "Parallel method for min- ing high utility itemsets from vertically partitioned distributed databases," in Proc. Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst., 2009, pp. 251–260.

[16] C.-W. Wu, B.-E. Shie, V. S. Tseng, and P. S. Yu, "Mining top-k high utility itemsets," in Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2012, pp. 78–86.

[17] J. Wang, J. Han, and J. Pei, "Closetþ: Searching for the best strate- gies for mining frequent closed itemsets," in Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2003, pp. 236–245.

