PATTERN GENERATION FOR COMPLEX DATA USING HYBRID MINING

Manish Kumar[1], Sumit Kumar[2], Sweety[1]

[1]IIIT-Allahabad, India, [2]IVY Comptech Pvt. Ltd.-Hyderabad
[manish.singh76,sumit78kumar90,sweety.bhagat89]@gmail.com

ABSTRACT

Combined mining is a hybrid mining approach for mining informative patterns from single or multiple data-sources, multiple-features extraction and applying multiple-methods as per the requirements. Data mining applications often involve complex data like multiple heterogeneous data sources, different user preference and create decision-making actions. The complete useful information may not be obtained by using single data mining method in the form of informative patterns as that would consume more time and space. This paper implements hybrid or combined mining approach that applies Lossy-counting algorithm on each data-source to get the frequent data item-sets and then generates the combined association rules. Applying multi-feature approach, we generate incremental pair patterns and incremental cluster patterns. In multi-method combined mining approach, FP-growth and Bayesian Belief Network are combined to generate classifier to get more informative knowledge. This paper uses two different data-sets to get more useful knowledge and compare the results.

KEYWORDS

Association Rule Mining, Lossy-Counting Algorithm, Incremental Pair-Patterns, Incremental Cluster-Patterns, Bayesian Belief Network.

1. INTRODUCTION

In distributed data mining algorithms [Kargupta and Park (2002)], data sampling is generally not accepted because it may miss useful data that may be filtered out during sampling. Distributed data sets need to join into one large data set but process may be more time and space consuming. More often such approach of handling multiple data sources can only be developed for specific cases and cannot be applied for all domain problems. Combined mining is a two-to-multistep data mining approach: In first step, it involves mining the atomic patterns from each individual data source and then next step combines atomic patterns into combined-patterns, which is more suitable for a particular problem. In multi-source combined mining approach, it generates informative patterns from individual data source and then generates the combined patterns. In multi-feature combined mining approach, we consider features from multiple data sets while generating the informative patterns, where it is necessary in order to make the patterns more actionable. In case of cluster patterns, we made the cluster of patterns with same prefix but the remaining data items in the pattern reflect results to be different. The advantage of our approach, it does not apply any pruning method or any clustering method separately to get the more informative patterns. In Lossy-counting algorithm's implementation, data sources have already been pruned at the boundary that directs most similar data items in the same bucket itself.

2. RELATED WORKS

Kargupta and Park (2002) provide an overview of distributed data mining algorithms, systems and applications. The paper pointed out a mismatch between the architecture of most off-the-shelf

DOI : 10.5121/ijdkp.2013.3409

data mining systems and the needs of mining systems for distributed applications. It also claims that such a mismatch may cause a fundamental bottleneck in many distributed applications. Kargupta et al (1999) presented a framework of collective data mining to conduct distributed data mining from heterogeneous sites. Authors observed that in a heterogeneous environment, naïve approaches to distributed data analysis may lead to incorrect data-model. Chattratichat et al (1999) designed Kensington software architecture for distributed enterprise data mining, which addresses the problem of data mining on logical and physical distribution of data and heterogeneous computational resources. Karypis and Wang (2005) present a new classifier, HARMONY, which is an example of direct mining for informative patterns as it directly mines the resultant set of rules required for classification. G. Dong and J. Li (1999) introduce a new type of patterns i.e. *emerging patterns* (EPs), for discovering knowledge from databases. Authors define EPs as data item-sets whose support increases more significantly from one to other data-set. They have used EPs to build very powerful classifiers. W. Fan et al (2008) builds a model based search tree, which partitions the data onto different nodes and at each node, it directly find out a discriminative pattern, which further divide its examples into more purer subsets. A novel technique was proposed by B. Liu et al.(1999), which first prunes the discovered association-rules to remove the insignificant association-rules from the entire set of association-rules, and then finds a subset of the un-pruned association-rules by which a summary of the discovered association-rules can be formed. The paper refers it as subset of association-rules as the *direction setting* (DS) *rules* because they can be used to set the directions, which are followed by the rest of the association-rules. By the help of the summary, the user can have more focus on the important aspects of the particular domain and also can view the relevant details. They suggest that their approach is effective as their experimental result shows that the set of DS rules is quite very small. Lent et al. (1997) proposed a method for clustering two-dimensional associations in large data-bases. This paper presents a geometric-based algorithm called BitOp, for clustering, embedded within ARCS (Association Rule Clustering System). It also measures the quality of the segmentation generated by ARCS. J. Han et al (2006) proposed a new approach called CrossMine, which mainly includes a set of novel and powerful methods for multi-relational classification including 1) tuple ID propagation, 2) new definitions for predicates and decision-tree nodes and 3) a selective sampling method. This paper also proposed two accurate and scalable methods for multi-relational classification i.e. CrossMine-Rule and CrossMine-Tree. C. Zhang et al (2008) proposed a novel approach of combined patterns to extract important, actionable and impact oriented information from a large amount of association rules. Authors also proposed definitions of combined patterns and also designed novel matrices to measure their interestingness and analyzed the redundancy in combined patterns. Combined mining as a general approach is proposed by C. Zhang et al (2011) to mine the informative patterns. This paper summarizes general framework, paradigms and basic processes for various types of combined mining. Authors also generate novel types of combined patterns from their proposed frameworks. H. Yu et al. (2003) proposed a new method called as *Clustering-Based SVM* (CB-SVM), in which, the whole data set can be scanned only once to have an SVM with samples that carry the statistical information of the data by applying a hierarchical micro-clustering algorithm. Authors also show that CB-SVM is also highly scalable for very large data sets and also generating very high classification accuracy. Longbing Cao (2012) proposed combined mining as an approach from the perspective of object and pattern relation analysis. This paper also discusses the fundamental aspects of combined pattern mining like feature interaction, pattern dynamics, pattern interaction, pattern impact, pattern structure etc. K. Kavitha et al. (2012) proposed a generic framework that uses utility in decision making to drive the data mining process. Their study proposes a novel approach to discover actionable combined patterns with composite items.

3. PROBLEM DEFINITION

Complex data may contain incredible information, which may not be mined directly by using a single method and it is also tough to deal with such information using different perspective such as client's perspective, business analyst's perspective and decision-makers perspective etc. Any service provider wants to predict the client's behavior to design the services according to client's perspective and also to reduce the traffic load. In our approach, we try to get patterns to retrieve useful information from complex data. This information may be used in different places, for example in e-commerce, stock market, market campaigns, measuring the success of marketing efforts and client-company behavior etc.

4. PROPOSED SOLUTION

The effectiveness and quality of the patterns which have to be discovered highly depends on the effectiveness and type of the data used during the pattern discovery process. We considered two data-sets from *UCI-Machine Learning Repository* named as "Adult" and "Pen-Digits". "Adult" data-set have 14 attributes, some of them are discrete attributes, while others are continuous attributes while "Pen-Digits" data-set have 17 attributes and all attributes are of type continuous attributes. Our proposed solution consists of following steps:

4.1 Preprocessing & Data mining approach

Data-sets contains unknown value (present as '?' in our data-set) for any of the attribute, then that tuple should be removed first, as such tuples are the source for noise and errors. Removing all such tuples from data-sets first, non-overlapping partitions of "Adult" dataset are generated so that each of such partition can behave as a sub-dataset. The main idea behind generating sub-datasets is that each of the sub-dataset can be used as a source of data for multi-source combined mining approach. We have generated 210 sub-datasets for "Adult" dataset as try to have maximum number of distinct attributes from all the sub-dataset on the basis of information gain value and gathered maximum 8 distinct attributes (from *Figure* 1), when we generated 210 partitions of "Adult" dataset. This paper considered another data set "Pen-Based Recognition of Handwritten Digits" as un-partitioned for analysis. Figure 1 shows the partitions of "Adult" data-set.

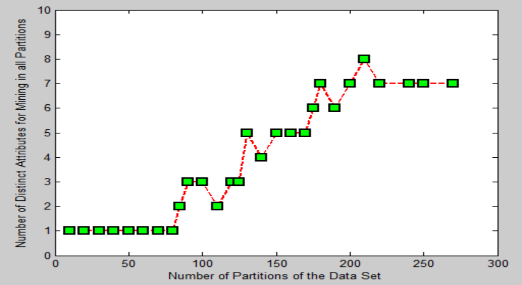

Figure 1. Number of partitions vs. Number of distinct attributes for Adult Data-Set

As stated earlier that input "Adult" dataset contains discrete as well as continuous attributes. Information gain for discrete attributes in a partition P have been computed as given below:

Expected information needed for the classification of a tuple in a partition P, if the class label attribute has n distinct values, each of which defines a distinct class [2], as follows:

$$\mathrm{I(P)} = -\sum_{i=1}^{n} \mathrm{p_i \log_2 p_i} \qquad (1)$$

Where, p_i is the probability that a particular tuple in partition P belongs to class C_i and we can compute p_i as $p_i = |C_{i,p}| \,/\, |P|$ and where $C_{i,p}$ defines the set of tuples of class C_i in P while $|C_{i,p}|$ and $|P|$ denotes the number of tuples in $C_{i,p}$ and P respectively.

Then we have to classify the tuples in P on some attribute A having m distinct values, as observed from the training dataset. The amount of information would be required for an exact classification [2] is measured by:

$$\mathrm{I_A}(P) = \sum_{j=1}^{m} (|\mathrm{P_j}|/|\mathrm{P}|) \times \mathrm{I}(\mathrm{P_j}) \tag{2}$$

Then the information Gain $G(A)$ for attribute A is measured as:

$$G(A) = I(P) – I_A(P) \tag{3}$$

For computing the information gain for continuous attributes B in a partition P, compute the information gain for every possible split-point for B and then choosing the best split-point. We considered split-point as a threshold on B. First, we sorted the values of B in increasing order and then typically the mid-point between each pair of adjacent values considered as a possible split-point. If there are y values of B, then $(y - 1)$ possible splits has to be computed. The reason for sorting the values of B is that if the values are already sorted then for determining the best split for B requires only one pass through the values. Then, for each possible split-point for B, by using $equation\ (2)$, we measured $I_A(P)$, where the number of value of $m = 2$. The point with the minimum expected information requirement for B will be selected as the best split-point for B. After finding out the information gain for each attribute, an attribute with maximum information gain from each partition P_i are selected and then by concatenating the attribute values from all partitions, a data-stream is formed, which serves as an input for lossy-counting algorithm. As state earlier “Pen-Digits” data-set is having continuous attributes only, the information gain for all continuous attributes are generated as stated above.

A. Lossy-counting Algorithm

Lossy-counting is a deterministic algorithm [2], which computes frequency counts over a stream of data-items. It approximates the frequency of items or item-sets within a user-specified error bound ε. If N is the current length of the data-stream then this algorithm takes $1/\varepsilon log(\varepsilon N)$ space in worst-case for computing the frequency counts of a single data-item. The steps are as follows:

Input: Support s, error bound ε and input data-stream

Output: Set of data-items with frequency counts at least equals to $(s - \varepsilon)N$

Step 1: The input data-stream logically divided into the buckets of width $w = ceil(1/\ \varepsilon)$ and each bucket is labeled with bucket id, initially starting from 1 for the first bucket, and the current bucket id is denoted by $B_{current}$, which is equal to $ceil(N/\ \varepsilon)$.

Step 2: Then maintain a data-structure DS, which is a set of values of the form (E, F_E, δ), where, E is an element from the input data-stream and F_E is the true frequency of the element E and δ denotes the maximum number of times E could have occurred in first $B_{current} - 1$ buckets. Initially DS will be empty.

Step 3: For an element from the data-stream, if E already exists in DS then increase its F_E by 1 else we have to create a new entry in DS such as $(E, 1, B_{current}\ - 1)$.

Step 4: If it is the bucket boundary then we have to prune DS as follows:
if $F_E\ +\ \delta\ \leq\ B_{current}$, then the entry (E, F_E, δ) has to be deleted from DS.

Step 5: When a user wants a final list of frequent data-items with support s, then output all those entries in DS with $F_E \geq (s - \varepsilon)N$.

Then, we generate the combined association rules [C. Zhang et al, 2011] of the frequent data-items computed by Lossy-counting algorithm.

4.2 Multi-feature mining approach

Generating combined association rules, we consider the heterogeneous features of different data types as well as of different data categories. If combined association rule is of the form "$IF\ X\ THEN\ Y$", where X is the antecedent and Y is the consequent part of the rule, then we have some traditional definitions for support, confidence and lift of the rule as given below in the Table 1.

Table 1. Support, Confidence and Lift for the Rule $X \rightarrow Y$

$SUPPORT$	$Prob(X\ U\ Y)$
$CONFIDENCE$	$Prob(X\ U\ Y)\ /\ Prob(X)$
$LIFT$	$Prob(X\ U\ Y)\ /\ (Prob(X)\ \times\ Prob(Y))$

On the basis of these traditional definitions of support, confidence and lift, we can compute the Contribution and Interestingness [1] $I_{RULE}$of the rule $X_p U X_R \rightarrow Y$ as follows:

$$Contribution\ (X_P U X_R \rightarrow Y) = LIFT\ (X_P\ U\ X_R \rightarrow Y)\ /\ LIFT\ (X_P \rightarrow Y)$$

$$= CONFIDENCE\ (X_P\ U\ X_R\ \rightarrow Y)/CONFIDENCE\ (X_P\ \rightarrow Y) \quad (4)$$

$$I_{RULE}\ (X_P U\ X_R \rightarrow\ Y) = Contribution\ (X_P\ U\ X_R \rightarrow Y)\ /\ LIFT\ (X_R \rightarrow\ Y) \quad (5)$$

Where, I_{RULE} indicates whether the Contribution of X_P (or X_R) to the occurrence of Y increases, while considering X_R (or X_P) as a precondition to the rule. To get more information, we also generate pair patterns, cluster pattern, incremental pair-pattern and incremental cluster-patterns [C. Zhang et al, 2011], and their respective contribution and interestingness matrices. In case of pair pattern, two atomic rules are taken to form a pair-pattern if and only if the two atomic rules have at least one common data-item in their antecedent parts and after removing those common data-item/data-items from the atomic rules the antecedent parts of none of the atomic rules should be null. In case of incremental pair-pattern, removing the common data-item/data-items from the pair-patterns and considered the common data-items as a pre-condition. In case of cluster pattern formation, we have included as much as rules in a single cluster on the basis of the common data-item/data-items in their antecedent parts and also have taken care of the fact that after removing the common data-item/data-items from the antecedent parts of the respective rules, the antecedent part of the rules should not be empty or null. In case of incremental cluster-patterns, we removed the common data-item/data-items from the respective rules in a cluster and considered the common data-item/data-items as a precondition for that particular cluster of rules.

4.3 Multi-method mining approach

In this approach, we first generate the association rules by using FP-growth algorithm [Han et al. (2006)] and then create the Bayesian belief network [J. Cheng et al. (1997)]. The testing data-set are classified by Bayesian belief network during testing phase.

5. RESULT AND ANLAYSIS

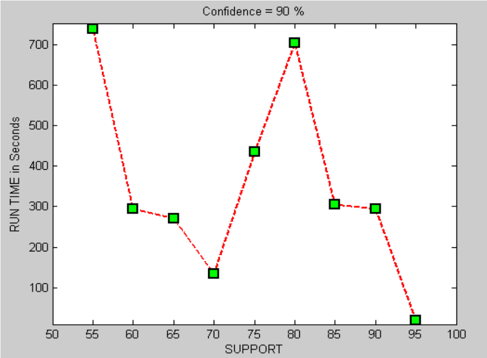

Figure 2. Support vs. Run-Time

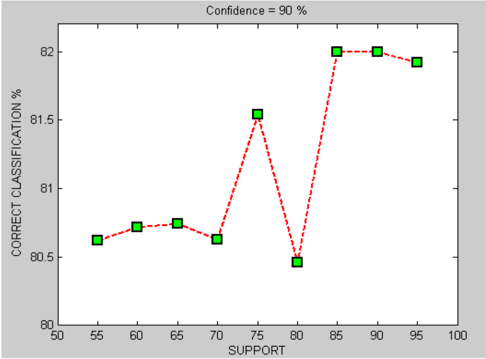

Figure 3. Support vs. Correct Classification %

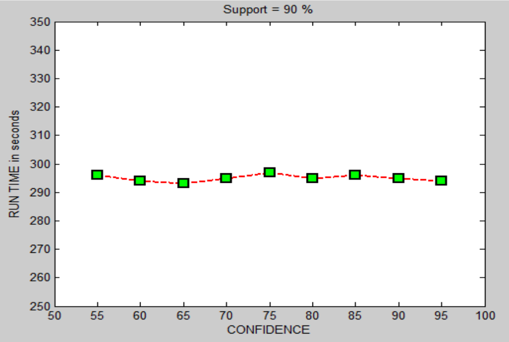

Figure 4. Confidence vs. Run-Time

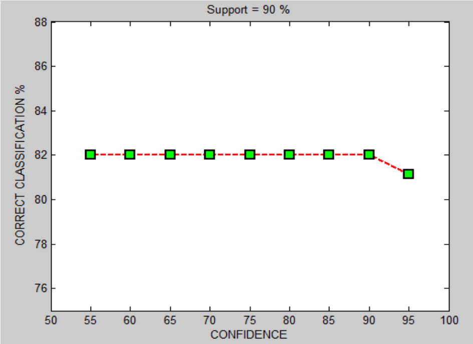

Figure 5. Confidence vs. Correct Classification %

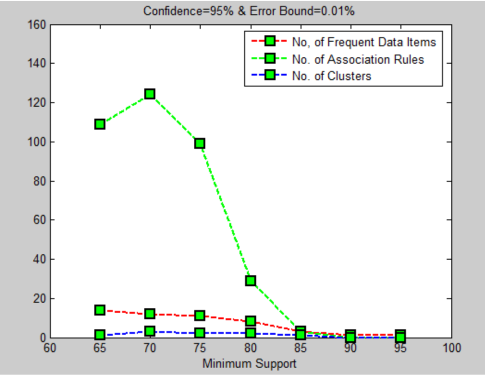

Figure 6. Support vs. Frequent Data Items, No. of Association Rules & No. of Clusters for "Adult" data-set

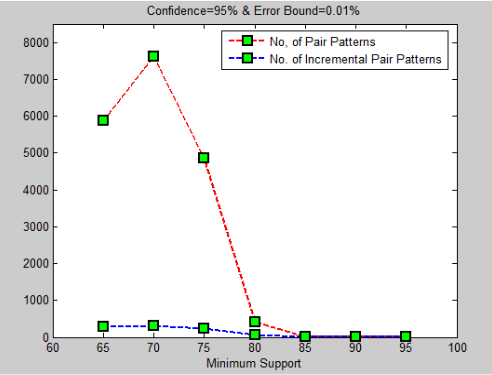

Figure 7. Support vs. No. of Pair Patterns & No. of Incremental Pair Patterns for "Adult" data-set

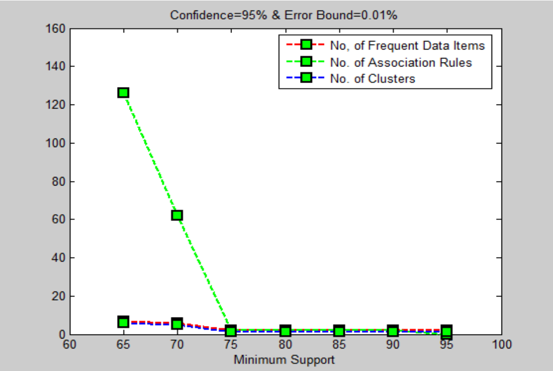

Figure 8. Support vs. Frequent Data Items, No. of Association Rules & No. of Clusters for "Pen-Digits" Data-set

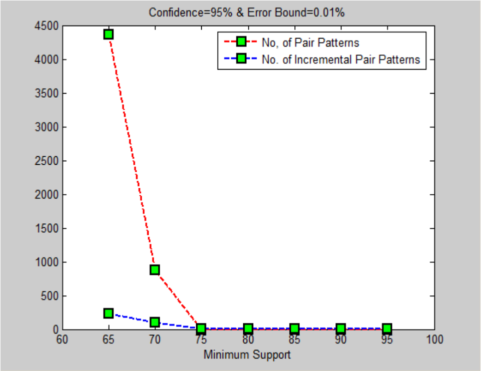

Figure 9. Support vs. No. of Pair Patterns & No. of Incremental Pair Patterns for "Pen-Digits" Data-set

"Adult" data-set with $32{,}561$ entries is considered and after preprocessing step, $30{,}162$ entries get selected for mining the useful information. Then considering "Pen-Digits" data-set with 7,494 entries and after preprocessing step, all entries get selected as "Pen-Digits" data-set doesn't have any noisy data or missing value for any attribute. We performed mining task and generated the information by including other perspective like client's perspective etc. In $fig. 2 \,\&\, 3$, we have shown the variation of run-time (in seconds) and correct classification percentage by Bayesian belief network vs. support by keeping $confidence = 90\%$ fixed for FP-growth implementation and in $fig. 4 \,\&\, 5$, we have shown the variation of run-time (in seconds) and correct classification percentage by Bayesian belief network vs. confidence by keeping $support = 90\,\%$ fixed for FP-growth implementation for "Adult" data-set. In $fig. 6$, we have shown the variation of the minimum support vs. the number of frequent data items obtained by Lossy-counting algorithm, number of association rules and number of total clusters obtained by the implementation of the multi-feature mining approach by keeping the $confidence = 95\%$ and $error\ bounds = 0.01\%$ for Lossy Counting Algorithm for "Adult" data-set. In $fig. 7$, we have shown the variation of the minimum support vs. the number of pair patterns along with the variation of number of incremental pair patterns for "Adult" data-set by keeping the $confidence = 95\%$ and $error\ bounds = 0.01\%$. In $fig. 8$, we have shown the variation of the minimum support vs. the number of frequent data items obtained by Lossy Counting Algorithm, number of association rules and number of total clusters obtained by the implementation of the multi-feature mining approach by keeping the $confidence = 95\%$ and $error\ bounds = 0.01\%$ for Lossy Counting Algorithm for "Pen-Digits" data-set. In $fig. 7$, we have shown the variation of the minimum support vs. the number of pair patterns along with the variation of number of incremental pair patterns for "Pen-Digits" data-set by keeping the $confidence = 95\%$ and $error\ bounds = 0.01\%$. From $fig. 6, 7{,}8 \,\&\, 9$, we can conclude that more the number of frequent data items more will the number of association rules and more will be the number of pair-patterns. The obtained numbers of clusters also have the dependency on the number of association rules obtained and the similarity between the association rules in-terms of the common data-items in the antecedent part of the association rules.

6. CONCLUSION AND FUTURE WORK

The support has impact on run-time as well as correct classification percentage by Bayesian belief network. Our approach identified combined patterns, which are more informative, actionable and impact-oriented as compared to any single patterns identified by traditional methods. There may be such frameworks, which are flexible and customizable for handling a large amount of complex data, for which data sampling and table joining may not be required. We further may develop some effective paradigms, for handling large and multiple sources of data available in industry, insurance, stock market, e-commerce and banking etc. in real-time.

REFERENCES

[1] C. Zhang, D. Luo, H. Zhang, L. Cao and Y. Zhao (2011), 'Combined Mining: Discovering Informative Knowledge in Complex Data', *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS*, VOL. 41, NO. 3, pp. 699-712.

[2] Han and Kamber (2006), 'Data Mining Concepts and Techniques', 2^{nd} ed., United State of America.

[3] C. Zhang, F. Figueiredo, H. Zhang, L. Cao and Y. Zhao (2007), 'Mining for combined association rules on multiple datasets', in *Proc. DDDM*, pp. 18–23.

[4] C. Zhang, H. Bohlscheid, H. Zhang, L. Cao and Y. Zhao (2008), 'Combined pattern mining: From learned rules to actionable knowledge', in *Proc. AI*, pp. 393–403.

[5] B. Park and H. Kargupta (2002), 'Distributed Data Mining: Algorithms, Systems, and Applications'. *Data Mining Handbook*, N. Ye, Ed 2002.

[6] Jaturon Chattratichat, John Darlington, et al (1999). 'An Architecture for Distributed Enterprise Data Mining'. Proceedings of the 7th International Conference on High-Performance Computing and Networking.

[7] B. Park, D. Hershbereger, E. Johnson and H. Kargupta (1999), 'Collective data mining: A new perspective toward distributed data mining'. *Accepted in the Advances in Distributed Data Mining*, Eds: Hillol Kargupta and Philip Chan, AAAI/MIT Press (1999).

[8] G. Dong and J. Li (1999), 'Efficient mining of emerging patterns: Discovering trends and differences,' in *Proc. KDD*, pp. 43–52.

[9] H. Cheng, J. Han, J. Gao, K. Zhang, O. Verscheure, P. Yu, W. Fan and X. Yan (2008), 'Direct mining of discriminative and essential graphical and item-set features via model-based search tree,' in *Proc. KDD*, pp. 230–238.

[10] B. Liu, W. Hsu and Y. Ma (1999), 'Pruning and summarizing the discovered associations,' in *Proc. KDD*, pp. 125–134.

[11] A. N. Swami, B. Lent and J. Widom (1997), 'Clustering association rules,' in *Proc. ICDE*, pp. 220–231.

[12] J. Han, J. Yang, P. S. Yu and X. Yin (2006), 'Efficient classification across multiple database relations: A CrossMine approach,' *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 6, pp. 770–783.

[13] C. Zhang, H. Bohlscheid, H. Zhang, L. Cao and Y. Zhao (2008), 'Combined pattern mining: From learned rules to actionable knowledge,' in *Proc. AI*, pp. 393–403.

[14] H. Yu, J. Han and J. Yang (2003), 'Classifying large data sets using SVM with hierarchical clusters,' in *Proc. KDD*, pp. 306–315.

[15] Frank, A. & Asuncion, A. (2010). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[16] David A. Bell, Jie Cheng and Weiru Liu (1997), 'An Algorithm for Bayesian Belief Network Construction from Data,' *In Proceedings of AI & STAT'97*, pp. 83-90.

[17] Dr. E. kumar and A. Solanki (2010), 'A Combined Mining Approach and Application in Tax Administration,' International Journal of Engineering and Technology Vol.2(2), 2010, 38-44.

[18] Longbing Cao (2012), 'Combined Mining: Analyzing Object and Pattern Relations for Discovering Actionable Complex Patterns,' sponsored by Australian Research Council Discovery Grants (DP1096218 and DP130102691) and an ARC Linkage Grant(LP100200774).

[19] Kavitha, K. and Ramaraj, E. (2012), 'Mining Actionable Patterns using Combined Association Rules,' International Journal of Current Research, Vol. 4, Issue, 03, pp.117-120, March, 2012.

Biographical Notes:

Sumit Kumar received his B.Tech (Information Technology) from Indian Institute of Information Technology, Allahabad, India. He is Software Engineer at IVY Comptech Pvt. Ltd., Hyderabad, India. His research interest areas are distributed systems, data mining, databases and computer networks.

Manish Kumar received his PhD from Indian Institute of Information Technology, Allahabad, India in Data Management in Wireless Sensor Networks. He is an Assistant Professor at Indian Institute of Information Technology, Allahabad, India. His research interest areas are databases, data management in sensor networks, and data mining.

Sweety received her B.Tech (Information Technology) from Indian Institute of Information Technology, Allahabad, India. She is a software developer at Kony India Pvt. Ltd., Hyderabad, India. Her research interest areas are data mining, databases and operating systems.