Mining articulate association rules from closed item sets: A Counter Support Measurement approach

Anurag Choubey
Dean Academic, Technocrats Institute of Technology, Bhopal
Rajiv Gandhi Technological University
Bhopal, India.

Ravindra Patel
Associate Professor and Head
Department of Computer Applications at Rajiv Gandhi Technological University
Bhopal, India

J.L. Rana
Former Professor and Head,
Department of Computer Science and Engineering & Information Technology at MANIT, Bhopal

ABSTRACT
In the previous works it has been observed that a frequent item set mining algorithm are supposed to mine the closed ones as the finish results in a compact and a complete progress set and enhanced potency. However, the latest closed item set mining algorithms works with both candidate maintenance and check paradigm hand in hand, which proves to be friendlier in runtime, as in case of area usage when support threshold is a reduced entity or the item sets gets long. In this paper, we have shown, CEG&REP with CSM (Counter Support Measurement) that is supposed to be a more efficient approach which can be utilized for mining articulate association rules from closed sequences. This approach outfits a exclusive rule coherency checking format with CSM, further that is laid mostly on another approach termed as Sequence Graph protruding which is termed as "Concurrent Edge Prevision and Rear Edge Pruning", hereby referred as CEG&REP. Moreover, we have pronounced a novel CSM methodology to crop rules which in turn seems to formulate articulate rules. The performance of CEG&REP with CSM (Counter Support Measurement) is tested on a whole observation having scrubby and dense real-life information, the tests have shown that approach of CEG&REP performs in a more efficient manner as compared to the previous versions as the CEG&REP approach takes less memory space and is swifter than the algorithms which were used in past works.

General Terms
Data Mining, Closed Itemset mining, Association Rule Mining, pattern discovery.

Key Terms
Counter Support Measurement, CSM, Concurrent Edge Prevision and Rear Edge Pruning, CEG&REP, GAZELLE

1. INTRODUCTION
The most significant tasks in knowledge Discovery in Databases [29] is Association rule mining, introduced in [28], aims at finding implicative tendencies from the sets of items in transaction database, which can be precious information for the decision-maker. The association rule can be best defined as the implication $X \rightarrow Y$, a function based on two interestingness parameters support and confidence, where X and Y are the sets of items and $X \cap Y = \phi$.Most of the algorithms are derived from Apriori [28] which is the first algorithm proposed in the association rule mining field. This algorithm tends to extract all association rules sustaining the condition of minimum thresholds of support and confidence, where the extraction initiates from the database. It is a widely accepted fact that the mining algorithms can determine excessive quantity of association rules; for example, thousands of rules are extracted from a database of several dozens of attributes and several hundreds of transactions. Additionally, as suggested by Silbershatz and Tuzilin [30], sometimes the less found support and unpredicted association rules which are even astonishing to the user, represents the valuable information. Hereby it can be learned that the efficiency of the algorithms is directly proportional to the support threshold which results in the increased familiarity of the discovered rules and hence, the user founds them less interesting. Hence, keeping the support threshold as low as possible in order to extract valuable information becomes a primary task. Regrettably, the consequence of lowering the support is the huge amount of rules, resulting in a source of confusion and tedious to analyze mining result for the decision-maker. It has been proven by the experiments that the rules are terrible to use as soon as the number of rules exceeds 100. In this light, the advent of an efficient technique for the reduction of number of rules will be a vital help to the decision maker.

This flaw has been discussed in past many works, which have suggested several solutions to overcome this issue of exceeding rules. In Contrast to the different algorithms which were introduced to reduce the number of itemsets by generating closed [31], maximal [32] or optimal itemsets [33], and several algorithms to reduce the number of rules, using non redundant rules [34], [35], or pruning techniques [36]; the selection of these discovered rules can be upgraded by post processing methods. Pruning, summarizing, grouping, or visualization [37] can be used as a post processing method. Elimination of irrelevant and redundant rules is included in Pruning. The post processing method, summarizing, generates brief set of rules. Grouping process, as the name indicates generates a group of rules. The visualization on the other hand enhances the readability of huge amount of rules with the help of graphical aids.

However, these post processing methods do not provide any definite assurance of producing set of rules, the user may find interesting as the interestingness of a set of rule totally depends on the taste and goal of the user whereas these methods are based on the statistical information in the database.

Through this paper, a novel framework is proposed by us aiming at identification of closed itemsets. CSM is used in discovering the associations. The basic concept behind the approach is that an association rule should only be reported when there is enough interest gain claimed during CSM in the data. To achieve this, presences as well as absence of items during mining are considered by us. An association such as beer $\rightarrow$ nappies will only be reported if we can also find that there are fewer occurrences of $\neg$beer $\rightarrow$nappies and beer $\rightarrow \neg$nappies but more of $\neg$beer$\rightarrow \neg$nappies. This approach will ensure that when a rule such as beer $\rightarrow$ nappies is reported, it indeed has the strongest interest in the data as comparison was made on both presence and absence of items during the mining process.

2. RELATED WORK

The problem of sequential item set mining was first observed by Agarwal and Srikant, and they reported a filtered algorithm as the solution to this problem. The algorithm, GSP [2] was based on Apriri property [19].Followed by this a numerous sequential item mining set algorithms are being developed for maximizing the efficiency. Some of the algorithms are, SPADE [4], PrefixSpan [5], and SPAM [6]. The concept of SPADE is laid on principle of vertical id-list format and it utilizes a lattice-theoretic method to fester the search space into many tiny spaces. In contrast to this PrefixSpan implements a horizontal format dataset representation and mines the sequential item sets with the pattern-growth paradigm: grow a prefix item set to attain longer sequential item sets on building and scanning its database. The SPADE and the PrefixSPan highly perform GSP. The recent algorithm is SPAM which is used for mining lengthy sequential item sets and implements a vertical bitmap representation. It has been found that the SPAM is a better performer than SPADE or PrefixSpan in case of mining long item sets, but the problem is that the SPADE occupies more space than SPADE &PrefixSpan. Since the frequent closed item set mining [15], many capable frequent closed item set mining algorithms are introduced, like A-Close [15], CLOSET [20], CHARM [16], and CLOSET+ [18]. Most of these algorithms use pre-defined ready mined frequent closed item sets to attain item set closure checking. The problem of memory space occupation is solved in two algorithms, TFP[21] and CLOSSET+2, both of which decrease the memory usage and search space for item set closure checking by implementing a compact 2-level hash indexed result tree structure to preserve the freely mined frequent closed item set candidates. Some pruning methods and item set closure verifying methods, initiated the can be extended for optimizing the mining of closed sequential item sets also. Another recent algorithm used for mining frequent closed sequence [17] is CloSpan, it follows the candidate maintenance-and-test method: first, generate a set of closed sequence candidates being stored in a hash indexed in a result-tree structure and then perform pruning on it. Moreover, it requires some pruning techniques like *Common Prefix* and *Backward Sub-Item set pruning* to prune the search space as CloSpan requires maintaining the set of closed sequence candidates but it results in heavy search space for item set checking when the frequency of closed sequences is high as it consumes much memory. Further it results in inappropriate scaling of the number if frequent closed sequences. Another algorithm which is a high-ranked algorithm in terms of performance and efficiency as compared to previously discussed is BIDE [26], which projects the sequences after which is then pruned iff the selected subsets for pruning contains the same support requires as the superset. This approach is more costly when the sequence length is considerably high, as this model chooses to project and pruning in a sequential manner. Our earlier literature [27] discussed some other interesting works published in recent literature.

3. DATASET ADOPTION AND FORMULATION

Set of Attributes I : A set of individual items that are unique in identity, which are together more than one in count representing transactions and/or sequences.

$$\mathrm{I} = \{i : \exists! i\}$$

Note: ‘I’ is set of unique attributes

Sequence set ‘S’: A set of sequences, where each sequence contains elements each element ‘e’ belongs to ‘I’ and true for a function p(e). Sequence set can formulate as

$$S = \{s \mid s = \{i : \exists! i \in \mathrm{I}\}\}$$

Here s represents the sequence of attributes that are diverged in properties and representation and each attribute belongs to attribute set I and

Sequence subset: If sequence $s_{ss} \in S$, $s_s \in S$ and

If $\quad (s_p \subseteq s_q)$

Then $\quad for_{i=1}^{n}(s_{pi}) < \bullet\, for_{j=1}^{m}(s_{qj})$ where

$$s_p \in S \; and \; s_q \in S$$

Total Support ts : occurrence count of a sequence as an ordered list in all sequences in sequence set ‘S’ can adopt as total support ts of that sequence. Total support ts of a sequence can determine by fallowing formulation.

$$f_{ts}(s_t) = | s_t < \bullet s_p \;\; (for\, each\; p = 1..| DB_S |) |$$

DB_S Is set of sequences

$f_{ts}(s_t)$: Represents the total support ts of sequence s_t is the number of super sequences of s_t

Qualified support q_S : The resultant coefficient of total support divides by size of sequence database adopt as qualified support q_S . Qualified support can be found by using fallowing formulation.

$$f_{qs}(s_t) = \frac{f_{ts}(s_t)}{| DB_S |}$$

Sub-sequence and Super-sequence: A sequence is sub sequence for its next projected sequence if both sequences having same total support.

Super-sequence: A sequence is a super sequence for a sequence from which that projected, if both having same total support.

Sub-sequence and super-sequence can be formulated as

If $f_{ts}(s_t) \geq r_S$ where r_s is required support threshold given by user

And $s_t <: s_p$ *for any p value* where $f_{ts}(s_t) \cong f_{ts}(s_p)$

4. CLOSED ITEMSET DISCOVERY

4.1 CEG&REP: Concurrent Edge Prevision and Rear Edge Pruning [39]

4.1.1 Preprocess

Dataset preprocessing and itemsets Database initialization is performed by us as the first stage of proposal. As we find itemsets with single element, we in parallel prune it with the itemsets of single elements if the support in the selected itemsets is less than the required support.

4.1.2 Concurrent Edge Prevision:

In this phase, we select all itemsets from given itemset database as input in parallel. Then we start projecting edges from each selected itemset to all possible elements. The first iteration includes the pruning process in parallel, from second iteration onwards this pruning is not required, which we claimed as an efficient process compared to other similar techniques like BIDE. In first iteration, we project an itemset s_p that spawned from selected itemset s_i from DB_S and an element e_i considered from 'I'. If the $f_{ts}(s_p)$ is greater or equal to rs, then an edge will be defined between s_i and e_i. If $f_{ts}(s_i) \cong f_{ts}(s_p)$ then we prune s_i from DB_S. This pruning process required and limited to first iteration only.

From second iteration onwards project the itemset S_p that spawned from $S_{p'}$ to each element e_i of 'I'. An edge can be defined between $S_{p'}$ and e_i if $f_{ts}(s_p)$ is greater or equal to rs. In this description $S_{p'}$ is a projected itemset in previous iteration and eligible as a sequence. Then apply the fallowing validation to find closed sequence.

4.1.3 Edge pruning:

If any of $f_{ts}(s_{p'}) \cong f_{ts}(s_p)$ that edge will be pruned and all disjoint graphs except s_p will be considered as closed sequence and moves it into DB_S and remove all disjoint graphs from memory.

The termination of above process do not take place till the graph becomes empty, i.e. till the elements which are connected through transitive edges and projecting itemsets are available in the memory.

4.1.4 CEG&REP [39] Algorithm:

This section describes algorithms for initializing sequence database with single elements sequences, spawning itemset projections and pruning edges from Sequence Graph SG.

Input: DB_S and 'I';
L1: For each sequence s_i in DB_S
Begin:
L2: For each element e_i of 'I'
Begin:
C1: if edgeWeight$(s_i, e_i) \geq rs$
Begin:
Create projected itemset s_p from (s_i, e_i)
If $f_{ts}(s_i) \cong f_{ts}(s_p)$ then prune s_i from DB_S
End: C1.
End: L2.
End: L1.
L3: For each projected Itemset s_p in memory
Begin:
$s_{p'} = s_p$
L4: For each e_i of 'I'
Begin:
Project s_p from $(s_{p'}, e_i)$
C2: If $f_{ts}(s_p) \geq rs$
Begin
Spawn SG by adding edge between $s_{p'}$ *and* e_i
End: C2
End: L4
C3: If $s_{p'}$ not spawned and no new projections added for $s_{p'}$
Begin:
Remove all duplicate edges for each edge weight from $s_{p'}$ and keep edges unique by not deleting most recent edges for each edge weight.
Select elements from each disjoint graph as closed sequence and add it to DB_S and remove disjoint graphs from SG.
End C3
End: L3
If $SG \neq \phi$ go to L3

5. COUNTER SUPPORT MEASUREMENT (CSM)

Let I be the universe of items composed of m different attributes such that I is set of items $\{i_1, i_2, \ldots. i_m\}$. Let D be the set of transactions $\{t_1, t_2, ..., t_n\}$ and for $k = 1..n$, $t_k = (t_{id}, T)$. Here t_{id} is transaction id and T is a tuple of items. The count of an transaction itemset T in D, denoted by count $< T >$, is the coverage of T, which is number of transactions in D containing T. Hence the support of T $S_{(T)}$ can be measured as fallow

$$S(T) = \frac{<T>}{|D|}$$

The confidence of the false positive rule $(T_i \in D) \Rightarrow (\neg T_j \in D)$ can be measured as

$$C(T_i \Rightarrow \neg T_j) = \frac{S(T_i \cup \neg T_j)}{S_{T_i}}$$

In Transaction database, each transaction is a collection of items involved sequences. The purpose of mining association rules is to find all the association rules for which the support and confidence is respectively greater than the minimum criteria provided by the user.

Further, the issue of mining association rules can be classified into two sub-sections, which are as follows:-

- Find frequent itemsets, Generate all itemsets that support is greater than the minimum support;
- Generation of association rules from frequent itemsets.

In logical analysis, the direct calculation of support logical analysis is not convenient, to calculate the support and confidence of negative associations using the support and confidence of positive association that is known: set A, B⊂I, A∩B=Φ, then:

$$\sup(\neg A) = 1 - \sup(A);$$
$$\sup(A \cup \neg B) = \sup(A) - Sup(A \cup B);$$
$$\sup(\neg A \cup B) = \sup(B) - \sup(A \cup B)$$
$$\sup(\neg A \cup \neg B) = 1 - \sup(A) - \sup(B) + \sup(A \cup B);$$

The logical analysis is carried out by us based on the above formulae, which results as the derivation of the actual support of the patterns which make the rule more articulate.

5.1 Hypothesis of Counter Support Measurement:

True Positive coverage (TPC): The set of transactions that contains all the attributes of the itemsets generating a rule R. Let i_A, i_B are the itemsets, which generating a rule $R(i_A \Rightarrow i_B)$ then

$$tpc(R_{(i_A \Rightarrow i_B)}) = S_{(i_A \cup i_B)}$$

True negative coverage (TNS): The set of transactions with no coverage of any attribute that belongs to the itemsets, which are generating a rule R. Let i_A, i_B are the itemsets, which generating a rule $R(i_A \Rightarrow i_B)$ then

$$tns(R(i_A \Rightarrow i_B)) = 1 - S(i_A) - S(i_B) + S(i_A \cup i_B)$$

False Positive Coverage (FPC): The coverage of set of attributes in the absence of one or more other attributes of the itemsets that are generating a rule R. Let i_A, i_B are the itemsets, which generating a rule $R_{(i_A \Rightarrow i_B)}$ then

$$FPC(R_{(i_A \Rightarrow i_B)}) = S_{(i_A)} - S_{(i_A \cup i_B)}$$

False Negative Coverage (FNC): The set transactions with no coverage of an attribute in the presence of other attributes of the itemsets that are generating rule R.

$$FNC(R_{(i_A \Rightarrow i_B)}) = S_{(i_B)} - S_{(i_A \cup i_B)}$$

The coverage of itemsets in rule R and positive score is sum of no of transactions in which all items in the rule R exist, no of transactions in which all items in the rule does not exist

a. Rule negative score is false positive count.
b. Rule positive score is sum of true positive count and true negative count of the rule.
c. Rule actual score is the difference of rule positive score and rule negative score, that is
   $(TPC(R) + TNC(R)) - (FPC(R) + FNC(R))$
d. Pattern Scope: Actual coverage of the pattern involved in rule R
e. Articulate rule: A rule R that generated with patterns, which are having pattern scope more than given support threshold and pattern scope of the LHS_P must be greater than pattern scope of the RHS_P.
   Here LHS_P is pattern on left hand side of the rule R
   Here RHS_P is pattern on right hand side of the rule R

6. HYPOTHESIS OF RULE PRUNING STRATEGY

For each rule R, if any of the pattern with less than pattern scope then that rule R will be pruned.

For each rule R, if pattern scope of the LHS_P is lesser than the pattern scope of the RHS_P then the rule R will be pruned

7. COMPARATIVE STUDY

This portion main concentrates on providing evidence in order to verify the claimed assumptions that:

The CEG&REP is actually a sealed series mining algorithm that is capable enough to critically surpass results, in comparison to algorithms like Closet+ and Charm.

The memory and momentum consumption is swifter than that of Closet+ algorithm.

3) With the effective use of the trait equivalent prognosis and also rim snipping of the CEG&REP with CSM for no articulate pattern pruning, we found that there is a notification of an enhanced occurrence and an expected reduction in the memory rate.

From the surveillance results we conclude that CEG&REP's implementation is far more noteworthy and significant as compared to other algorithms such as Closet+ and bide.

The system employed for the study of the algorithms was a workstation equipped with core2duo processor, 2GB RAM and operating system as Windows XP. The parallel replica was set up to attain the thread concept in JAVA.

7.1 Dataset Characteristics:

GAZELLE[38] is supposedly found to be a very opaque dataset, which assists in excavating enormous quantity of recurring clogged series with a profitably high threshold somewhere close to 90%. It also has a unique element of being equipped with large volume around 59602 of transaction series and 497 divergent objects. The job of review of serviceable legacy's consistency has been accomplished by this dataset.

7.2 Performance Analysis

In assessment with all the other regularly quoted forms, taking in view the detailed study of the factors mainly, memory consumption and runtime, judging with CEG&REP with CSM.

In contrast to CEG&REP and other regularly quoted forms, a very intense dataset GAZELLE is used which has petite recurrent closed series whose end to end distance is less than 10, even in the instance of high support amounting to around 90%. The diagrammatic representation displayed in Fig 1 explains that the CEG&REP and other regularly quoted forms execute in a similar fashion in case of support being 90% and above. But in situations when the support case is 88% and less, then the act of CEG&REP with CSM surpasses other regularly quoted forms. The disparity in memory exploitation of CEG&REP and other regularly quoted forms can be clearly observed because of the consumption level of CEG&REP being lower than that of others. The concept CSM we introduced here played a vital role in articulate rule detection. The significant improvement in articulate rules detection from closed itemsets can be observable in our results (see fig 2 and 3).

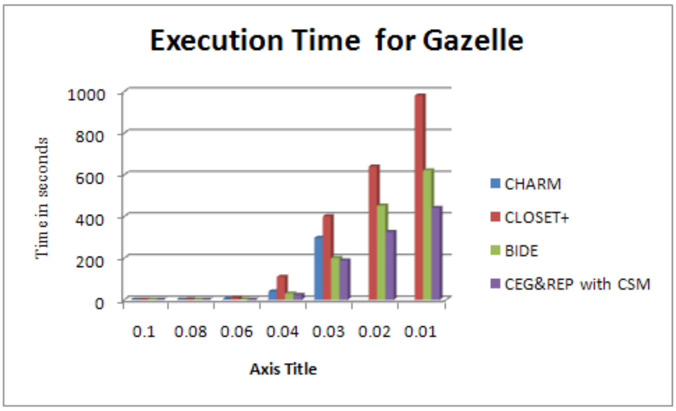

Fig 1: A comparison report for Runtime

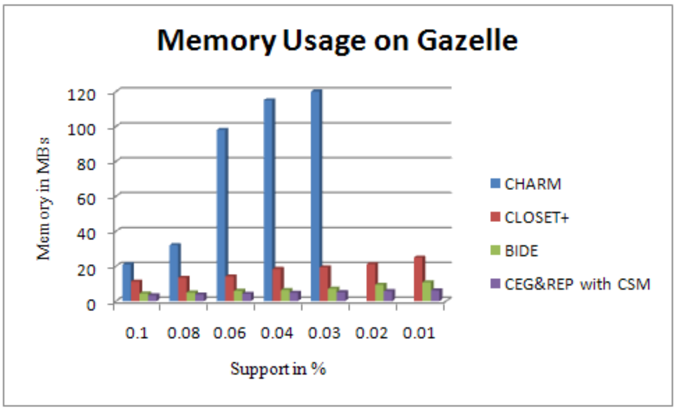

Fig: 2: A comparison report for memory usage

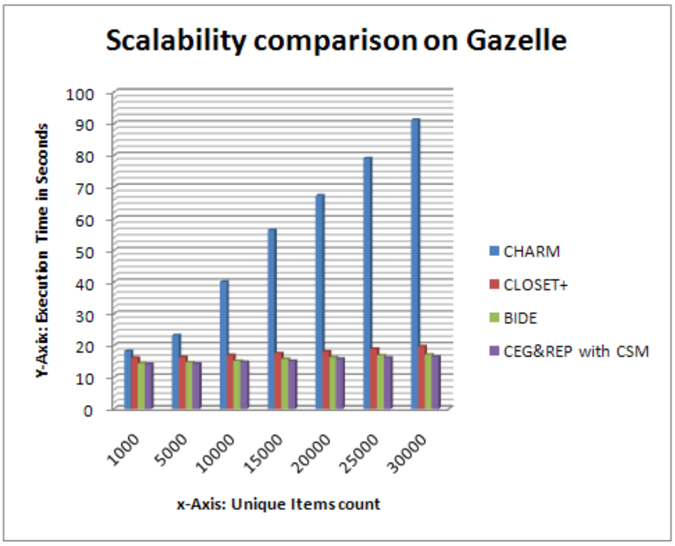

Fig 3: A comparison report of scalability on gazelle dataset

8. CONCLUSION

The clogged prototype mining drives dense product set and it have fairly advanced competency in contrast to the recurrent prototype mining even though both of these types represent similar animated power, this fact has been proved scientifically as well as experimentally. Further, the in-depth study suggests that the fact is generally true when the count of recurrent models is sufficiently large and moreover with the same recurrent bordered models as well. However, in an informal observation it has been found that previously formed clogged mining algorithms are a function of chronological set of recurrent mining outlines. This is used to check the testimony of a recurrent outline that whether an innovative outline is blocked or otherwise if it can outlast few previously mined blocked patterns. These results to a condition where the memory utilization is considerably extraordinary but consequences insufficiency of increasing seek out space for outline closure inspection. Through this paper, we predicted a rare algorithm for retreating recurring closed series with the aid of a graph. The functions performed by the algorithm are: It avoids the disfigurement of contender's maintenance and test paradigm, allocates the memory space more precisely and guarantees recurrent closure of clogging in an efficient manner and apart from it, consuming less amount of memory plot as compared with the previously developed mining algorithms. Further, the inevitability of conserving per-defined set of blocked recurrences is no more required; hence it maintains an appreciable coherence among the range of the count of frequent clogged models. A Sequence graph is incorporated by CEG&REP and has the proficiency of collecting the recurrent clogged pattern in an online approach. The effectiveness of dataset drafts can be pasteurized by a wide-spread range of experimentation on a number of authentic datasets amassing varied allocation attributes. Moreover, the CEG&REP approach is much advanced in terms of velocity and memory allocation as compared to like CHARM, CLOSET+, CHARM and BIDE algorithms. From the amount of progressions, linear scalability is provided. Through the CSM it is also proven that CEG&REP is efficient in discovering the closed itemsets. Many scientific researches prove and verify that the restrictions are crucial for a number of synchronized outlined mining algorithms. In addition we improved articulate rule mining performance by introducing CSM as an extension to CEG&REP. Future studies include proposing of post processing and pruning of the rules based on ontological weights of the attribute relations.

9. REFERENCES

[1]F. Masseglia, F. Cathala, and P. Poncelet, The psp approach for mining sequential patterns. In PKDD'98, Nantes, France, Sept. 1995.

[2]R. Srikant, and R. Agrawal, Mining sequential patterns: Generalizations and performance improvements. In EDBT'96, Avignon, France, Mar. 1996.

[3]J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M.C. Hsu, FreeSpan: Frequent pattern-projected sequential pattern mining . In SIGKDD'00, Boston, MA, Aug. 2000.

[4]M. Zaki, SPADE: An Efficient Algorithm for Mining Frequent Sequences. Machine Learning, 42:31-60, Kluwer Academic Pulishers, 2001.

[5]J. Pei, J. Han, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M.C. Hsu, PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth. In ICDE'01, Heidelberg, Germany, April 2001.

[6]J. Ayres, J. Gehrke, T. Yiu, and J. Flannick, Sequential PAttern Mining using a Bitmap Representation. In SIGKDD'02, Edmonton, Canada, July 2002.

[7]M. Garofalakis, R. Rastogi, and K. Shim, SPIRIT: Sequential PAttern Mining with regular expression constraints. In VLDB'99, San Francisco, CA, Sept. 1999.

[8]J. Pei, J. Han, and W. Wang, Constraint-based sequential pattern mining in large databases. In CIKM'02, McLean, VA, Nov. 2002.

[9]M. Seno, G. Karypis, SLPMiner: An algorithm for finding frequent sequential patterns using lengthdecreasing support constraint. In ICDM'02,, Maebashi, Japan, Dec. 2002.

[10]H. Mannila, H. Toivonen, and A.I. Verkamo, Discovering frequent episodes in sequences . In SIGKDD'95, Montreal, Canada, Aug. 1995.

[11]B. Ozden, S. Ramaswamy, and A. Silberschatz, Cyclic association rules. In ICDE'98, Olando, FL, Feb. 1998.

[12]C. Bettini, X. Wang, and S. Jajodia, Mining temporal relationals with multiple granularities in time sequences. Data Engineering Bulletin, 21(1):32-38, 1998.

[13]J. Han, G. Dong, and Y. Yin, Efficient mining of partial periodic patterns in time series database. In ICDE'99, Sydney, Australia, Mar. 1999.

[14]J. Yang, P.S. Yu, W. Wang and J. Han, Mining long sequential patterns in a noisy environment. In SIGMOD' 02, Madison, WI, June 2002.

[15]N. Pasquier, Y. Bastide, R. Taouil and L. Lakhal, Discoving frequent closed itemsets for association rules. In ICDT'99, Jerusalem, Israel, Jan. 1999.

[16]M. Zaki, and C. Hsiao, CHARM: An efficient algorithm for closed itemset mining. In SDM'02, Arlington, VA, April 2002.

[17]X. Yan, J. Han, and R. Afshar, CloSpan: Mining Closed Sequential Patterns in Large Databases. In SDM'03, San Francisco, CA, May 2003.

[18]J. Wang, J. Han, and J. Pei, CLOSET+: Searching for the Best Strategies for Mining Frequent Closed Itemsets. In KDD'03, Washington, DC, Aug. 2003.

[19]R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In VLDB'94, Santiago, Chile, Sept. 1994.

[20]J. Pei, J. Han, and R. Mao, CLOSET: An efficient algorithm for mining frequent closed itemsets .In DMKD'01 workshop, Dallas, TX, May 2001.

[21]J. Han, J. Wang, Y. Lu, and P. Tzvetkov, Mining Top- K Frequent Closed Patterns without Minimum Support. In ICDM'02, Maebashi, Japan, Dec. 2002.

[22]P. Aloy, E. Querol, F.X. Aviles and M.J.E. Sternberg, Automated Structure-based Prediction of Functional Sites in Proteins: Applications to Assessing the Validity of Inheriting Protein Function From Homology in Genome Annotation and to Protein Docking. Journal of Molecular Biology, 311, 2002.

[23]R. Agrawal, and R. Srikant, Mining sequential patterns. In ICDE'95, Taipei, Taiwan, Mar. 1995.

[24]I. Jonassen, J.F. Collins, and D.G. Higgins, Finding flexible patterns in unaligned protein sequences. Protein Science, 4(8), 1995.

[25]R. Kohavi, C. Brodley, B. Frasca, L.Mason, and Z. Zheng, KDD-cup 2000 organizers' report: Peeling the Onion. SIGKDD Explorations, 2, 2000.

[26]Jianyong Wang, Jiawei Han: BIDE: Efficient Mining of Frequent Closed Sequences. ICDE 2004: 79-90

[27] Anurag Choubey, Dr. Ravindra Patel and,Dr. J.L. Rana. Article: Frequent Pattern Mining With Closeness Considerations: Current State Of The Art. GJCST Issue 11, Volume 17, August 2011. Published by Global Journals, 25200 Carlos Bee Blvd. #495, Hayward, CA 94542, USA Published by Foundation of Computer Science, New York

[28] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proc. ACM SIGMOD, pp. 207-216, 1993.

[29] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, 1996.

[30] A. Silberschatz and A. Tuzhilin, "What Makes Patterns Interesting in Knowledge Discovery Systems," IEEE Trans. Knowledge and Data Eng. vol. 8, no. 6, pp. 970-974, Dec. 1996.

[31] M.J. Zaki and M. Ogihara, "Theoretical Foundations of Association Rules," Proc. Workshop Research Issues in Data Mining and Knowledge Discovery (DMKD '98), pp. 1-8, June 1998.

[32] D. Burdick, M. Calimlim, J. Flannick, J. Gehrke, and T. Yiu, "Mafia: A Maximal Frequent Itemset Algorithm," IEEE Trans. Knowledge and Data Eng., vol. 17, no. 11, pp. 1490-1504, Nov. 2005.

[33] J. Li, "On Optimal Rule Discovery," IEEE Trans. Knowledge and Data Eng., vol. 18, no. 4, pp. 460-471, Apr. 2006.

[34] M.J. Zaki, "Generating Non-Redundant Association Rules," Proc. Int'l Conf. Knowledge Discovery and Data Mining, pp. 34-43, 2000.

[35] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal, "Efficient Mining of Association Rules Using Closed Itemset Lattices," Information Systems, vol. 24, pp. 25-46, 1999.

[36] H. Toivonen, M. Klemettinen, P. Ronkainen, K. Hatonen, and H. Mannila, "Pruning and Grouping of Discovered Association Rules," Proc. ECML-95 Workshop Statistics, Machine Learning, and Knowledge Discovery in Databases, pp. 47-52, 1995.

[37] B. Baesens, S. Viaene, and J. Vanthienen, "Post-Processing of Association Rules," Proc. Workshop Post-Processing in Machine Learning and Data Mining: Interpretation, Visualization, Integration, and Related Topics with Sixth ACM SIGKDD, pp. 20-23, 2000.

[38] http://archive.ics.uci.edu/ml/datasets/

[39] Anurag Choubey, Dr. Ravindra Patel, Dr. J.L. Rana "Concurrent Edge Prevision and Rear Edge Pruning Approach for Frequent Closed Itemset Mining", IJACSA, Volume 2 No. 11, November 2011

10. AUTHORS PROFILE

Anurag Choubey[1]: He has completed B.Sc. (Electronics) in 1990 from Dr. Hari Singh Gaur Vishwavidyalaya, Sagar, M.P., India and M.Sc. (Applied Physics) in 1993 from Govt. Engineering College, Jabalpur, M.P., India. He has completed MCA in 2008 from Guru Ghasidas Vishwavidyalaya (Presently central University), Bilaspur C.G., India. He has worked as a Lecturer at Govt. Engg. College, Jabalpur and Hitkarni College of Engineering & Technology, Jabalpur from 1998 to 2000. From October 2000 onwards he has been working in Technocrats Institute of Technology, Bhopal. Currently working as Dean Academic, Technocrats Institute of Technology, Bhopal M.P., India. He posses more than 13 years of experience in teaching and has worked in different capacities like controller of exam, admission in-charge and other administrative post.

Dr. Ravindra Patel[2]: Associate Professor and Head, Department of Computer Applications at Rajiv Gandhi Technological University, Bhopal, India. He has awarded Ph.D. degree in Computer Science. He posses more than 10 years of experience in teaching post-graduate classes. He has published more than 15 papers in international and national journals and conference proceedings. He is member of International Association of Computer Science and Information Technology (IACSIT).
Email: ravindra@rgtu.net

Dr. J.L. Rana[3]: Former Professor and Head, Department of Computer Science and Engineering & Information Technology at MANIT, Bhopal, India with 32 years of vast experience of teaching, out of which 19 years as Professor of Computer Science & Engg. and Information Technology. Currently working as Group Director, Radha Raman Group of Institute, Bhopal, he has completed B.E. (Hons) in 1968 from GEC, Jabalpur and M.S. Computer Control from University of Hawaii (UH), USA in 1972. He has awarded Ph.D. degree in Computer from Indian Institute of Technology, Bombay in 1987. He also posses more than 25 years of experience in post graduate teaching He has published more than 30 papers in international and national journals and 65 international and national conference proceedings. He has guided 12 nos. of Ph.D. and 5 no. in progress He is a senior life member of CSI and Chairman CSI Bhopal Chapter for two terms.
Email: jl_rana@yahoo.com