

Pattern Discovery using Fuzzy FP-growth Algorithm from Gene Expression Data

Sabita Barik¹, Debahuti Mishra², Shruti Mishra³, Sandeep Ku. Satapathy⁴, Amiya Ku. Rath⁵ and Milu Acharya⁶

^{2,3,4,6}Institute of Technical Education and Research, Siksha O Anusandhan University, Bhubaneswar, Odisha, INDIA

¹Department of Computer Science and Engineering, Gandhi Engineering College, Bhubaneswar, Odisha, INDIA

⁵Department of Computer Science and Engineering, College of Engineering Bhubaneswar, Odisha, INDIA

sabita.barik10@gmail.com ,debahuti@iter.ac.in,shruti_m2129@yahoo.co.in, sandeepkumar04@gmail.com, amiyamaiya@rediffmail.com and milu_acharya@yahoo.com

Abstract- The goal of microarray experiments is to identify genes that are differentially transcribed with respect to different biological conditions of cell cultures and samples. Hence, method of data analysis needs to be carefully evaluated such as clustering, classification, prediction etc. In this paper, we have proposed an efficient frequent pattern based clustering to find the gene which forms frequent patterns showing similar phenotypes leading to specific symptoms for specific disease. In past, most of the approaches for finding frequent patterns were based on Apriori algorithm, which generates and tests candidate itemsets (gene sets) level by level. This processing causes iterative database (dataset) scans and high computational costs. Apriori algorithm also suffers from mapping the support and confidence framework to a crisp boundary. Our hybridized Fuzzy FP-growth approach not only outperforms the Apriori with respect to computational costs, but also it builds a tight tree structure to keep the membership values of fuzzy region to overcome the sharp boundary problem and it also takes care of scalability issues as the number of genes and condition increases.

Keywords: Gene Expression Data; Association Rule mining; Apriori Algorithm, Frequent Pattern Mining, FP-growth Algorithm

I. INTRODUCTION

Gene expression data is arranged in a data matrix, where each gene corresponds to one row and each condition to one column. Each element of this matrix represents the expression level of a gene under a specific condition and it is represented by real value [2]. Main objective of the gene expression are.: first, grouping of genes according to their expressions under the multiple conditions and second, grouping of conditions based on the expression data.

It helps the molecular biologist in many aspects, like gathering information about different cell states, functioning genes, identifying gene that reflects biological process of interest etc. The importance of gene expression data is to identify genes whose expression levels that reflects biological process of interest (Such as development of cancer). The analysis can provide clues and guesses for the functioning of genes.

Frequent patterns are pattern (such as item sets, subsequences or substructures) that appear in a data set frequently. In this paper we are interested in using association rule mining for discovering the gene expression patterns those exhibit similar expression levels and by discovering such patterns it can be helpful for medical diagnosis. One of the oldest Apriori algorithms is based upon the Apriori property that for an item set to be frequent; certainly any

subset must be frequent. The Apriori algorithm employs an iterative approach known as a level wise search, where k -items are used to explore $(k+1)$ item sets. Apriori algorithm suffers by scanning the database while finding the k -item sets in each and every step. Due to which the processing overhead is drastically increased. FP-growth algorithm which is enhanced version of apriori algorithm gives better performance while scanning the data base.

For finding the large set of data items we need association rule mining. Association rule mining finds interesting association and correlation relationship among a large set of data items. Frequent itemset mining leads to the discovery of associations and correlation relationship among genes and conditions in large data bases or data sets with massive amounts of data. The discovery of interesting correlation relationship among gene and conditions can help in medical diagnosis, gene regulatory network etc. FP-growth algorithm is that takes less time to search the each level of the tree. The FP-growth method transforms the problem of finding long frequent patterns to searching for shorter ones recursively and then concatenating the suffix. It uses the least frequent items as a suffix, offering good selectivity. The method substantially reduces search cost.

In this paper, we have used FP-growth mining algorithm hybridized with Fuzzy logic to find interesting association and correlation relationship among a large set of data items. Fuzzy logic is used to find out the support and confidence using the membership function $\mu = [0, 1]$. Fuzzy sets can generally be viewed as an extension of the classical crisp sets. They have been first introduced by Lofti A. Zadeh in 1965 [14]. Fuzzy sets are generalized sets which allow for a graded membership of their elements. Usually the real unit interval $[0;1]$ is chosen as the membership degree structure. Crisp sets are discriminating between members and nonmembers of a set by assigning 0 or 1 to each object of the universal set. Fuzzy sets generalize this function by assigning values that fall in a specified range, typically 0 to 1, to the elements. This evolved out of the attempt to build a mathematical model which can display the vague colloquial language. Fuzzy sets have proofed to be useful in many areas where the colloquial language is of influence. The support and confidence framework can be generalized using fuzzy membership values for not considering the crisp nature of clustering.

A. Proposed Model



Figure 1: Our Proposed Model

Our proposed work is to find the frequent patterns from gene expression data using FP-growth algorithm which is the enhanced version of Apriori. FP-growth algorithm constructs the conditional frequent pattern (FP) tree and performs the mining on this tree. FP-tree is extended prefix tree structure, storing crucial and quantitative information about frequent sets. FP-growth method transforms the problem of finding long frequent patterns to search for shorter once recursively and then concentrating the suffix. Then, to hybridize the fuzzy logic to map the support and confidence measures to the membership value to $\mu = [0, 1]$. Finally, we validate our hybridized model by comparing our model with existing apriori models by considering various parameters. Our model (See figure 1) outperforms the apriori model on the basis of run time for finding number of patterns and also the scalability issues have been found to be improved significantly considering both the attributes and objects as they increases.

B. Paper Layout

This paper is arranged in the following manner, section I gives the introduction as well as our proposed model is outlined, section II deals with related work on frequent pattern mining. In section III the preliminary information about gene expression data, apriori algorithm, FP-growth approach and algorithms are described. Section VI describes our proposed algorithm. Section V gives the analysis of our work and shows its significance over the Apriori algorithm. Finally, section VI gives the conclusion and future directions of our work.

II. RELATED WORK

Association rule mining finds interesting association and correlation relationship between genes [1]. A key step in the analysis of gene expression data is to find association and correlation relationship between gene expression data. DNA microarrays allow the measurement of expression levels for a large number of genes, perhaps all genes of an organism within a number of different experimental samples. It is very much important to extract biologically meaning the information from this huge amount of expression data to know the current state of the cell because most cellular processes are regulated by changes in gene expression. Gene expression data refers to a distinct class of clustering algorithms that performs simultaneous row-column clustering [2]. An imposed version of depth first search, the depth fast implementation of Apriori as devised in [8] is presented. Given a database of (e.g. supermarket) transactions, the DF algorithm builds a so called trie that contains all frequent item sets i.e. is all item sets that are contained in at least minimum support transactions with minimum support a given threshold value. There is a one to one correspondence between the paths and the frequent item sets. The new version, so called Depth Fast search, differs from Depth fast in that its data structure representing the data base is borrowed from the FP-growth algorithm. Frequent item sets play an essential role in many data mining tasks that try to

find interesting patterns from databases, such as association rules, correlations, subsequences, episodes, classifiers, cluster and many more of which the mining association rule is one of the most popular problems[4].

The original motivation for searching association rules came from the need to analyze so called supermarket transaction data that is to examine customers' behavior in terms of the purchased products. Such rules can be useful for decisions concerning product pricing, promotions, store layout many others. With the rapid progress of bioinformatics of post genomic era, more and more bio information needs to be analyzed [6]. Many researchers have been also focused on using association rule mining method to construct the gene regulatory network. The existing approach of mining frequent pattern in microarray data set are item are enumeration and sample enumeration. Association rules, used widely in the area of market basket analysis, can be applied to the analysis of expression data as well [7]. Association rules can reveal biologically relevant associations between different genes or between environmental effects and gene expression. Item in gene expression data can include genes are highly expressed or repressed, as well as relevant facts describing the cellular environment of the genes e.g. the diagnosis of the tumor sample from which a profile was obtained.

III. PRELIMINARIES

A. Gene Expression Data

One of the reasons to carry out a microarray experiment is to monitor the expression level of genes at a genome scale. Patterns could be derived from analyzing the change in expression of the genes, and new insights could be gained into the underlying biology. The processed data, after the normalization procedure, can then be represented in the form of a matrix, often called gene expression matrix (See table 1). Each row in the matrix corresponds to a particular gene and each column could either correspond to an experimental condition or a specific time point at which expression of the genes have been measured. The expression levels for a gene across different experimental conditions are cumulatively called the gene expression profile, and the expression levels for all genes under an experimental condition are cumulatively called the sample expression profile. Once we have obtained the gene expression matrix, additional levels of annotation can be added either to the gene or to the sample. For example, the function of the genes can be provided, or the additional details on the biology of the sample may be provided, such as disease state or normal state.

Table 1 Gene Expression Data Set

	C ₁	C ₂	C ₃	C ₄
Gene ₁	10	80	40	20
Gene ₂	100	200	400	200
Gene ₃	30	240	40	60
Gene ₄	20	160	60	80

B. Association Rule mining for frequent pattern

Association rule mining [12] finds interesting association and correlation relationship among a large set of data items.

The rules are considered interesting if they satisfy both a minimum support threshold and minimum confidence threshold. In general, association rule mining can be viewed as a two-step process:

- Find all the frequent item sets: by definition, each of these item sets will occur at least as frequently as a predetermined minimum support count, minimum support.
- Generate strong association rules from the frequent item sets; by definition these rules must satisfy minimum support and minimum confidence.

An association rule is an implication of the form $A \Rightarrow B$, where A sub set of I , B sub set of I , $A \cap B = \Phi$. The rule $A \Rightarrow b$ holds in the transactions set D with support s , where s is the percentage of transaction in D that contain $A \cup B$. this is taken to be probability, $P(A \cup B)$

$$\text{Support } (A \Rightarrow B) = P(A \cup B) \quad \text{Equation (1)}$$

The rule $A \Rightarrow B$ has confidence C in the transaction set D , where C is the percentage of transaction in D containing A that also contain B . This is taken to be conditional probability $P(B/A)$.

$$\text{Confidence } (A \Rightarrow B) = P(B/A) \quad \text{Equation (2)}$$

C. Apriori Algorithm

Apriori algorithm [5] is proposed by Rakesh Agrawal and Ramakrishnan Srikant from IBM Almaden Research Center. It is regarded as one of the breakthroughs in the progress of frequent pattern mining algorithms. Apriori employs an iterative approach known as level wise search, where k -item sets are used to explore $k + 1$ -itemsets. First, the set of frequent 1-itemsets is found. This is denoted as $1L$. $1L$ is used to find $2L$, the frequent 2-itemsets, which is used to find $3L$, and so on, until no more frequent k -item sets can be found. The finding of each kL requires one full scan of the database. Throughout the level-wise generation of frequent item sets, an important anti-monotone heuristic is being used to reduce the search space. This heuristic is termed as Apriori heuristic.

The Apriori heuristic: if any length k pattern is not frequent in the database, its length $k + 1$ super-pattern can never be frequent.

D. FP-growth Approach

It is noticed that the bottleneck of the Apriori method rests on the candidate set generation and test. In [11], a novel algorithm called FP-growth is proposed by Jiawei Han, Jian Pei and Yiwen Yin. This algorithm is reported to be an order of magnitude faster than the Apriori algorithm. The high efficiency of FP-growth is achieved in the following three aspects. They form the distinct features of FP-growth meanwhile. First, an extended prefix tree structure, called frequent pattern tree or FP-tree for short is used to compress the relevant database information. Only frequent length-1 items will have nodes in the tree, and the tree nodes are arranged in such a way that more frequently occurring nodes will have better chances of sharing than less frequently occurring nodes. Second, an FP-tree-based pattern fragmentation growth mining method FP-growth is developed. Starting from a frequent length-1 pattern (as an

initial suffix pattern), FP-growth examines only its conditional pattern base (a “sub-database” which consists of the set of frequent items co-occurring with the suffix pattern), constructs its conditional FP-tree and performs mining recursively on such a tree. The pattern growth is achieved via concatenation of the suffix pattern with the new ones generated from a conditional FP-tree. Since the frequent pattern in any transaction is always encoded in the corresponding path of the frequent pattern trees, pattern growth ensures the completeness of the result.

In this context, FP-growth is not Apriori-like restricted generation-and-test but restricted test only. The major operations of mining are count accumulation and prefix path count adjustment, which are usually much less costly than candidate generation and pattern matching operations performed in most Apriori-like algorithms. Third, the search technique employed in mining is a partition-based, divide-and conquers method rather than Apriori-like bottom-up generation of frequent patterns combinations. This dramatically reduces the size of conditional pattern base generated at the subsequent level of search as well as the size of its corresponding conditional FP-tree. Inherently, it transforms the problem of finding long frequent patterns to looking for shorter ones and then concatenating with the suffix. The novelty of FP-growth is that to generate all frequent patterns, no matter how long they will be; only 2 database scans are needed. One is used to find out frequent 1-item sets, the other is used to construct a FP-tree. The remaining operation is recursively mine the FP-tree using FP-growth. FP-tree resides in main-memory and therefore FP-growth avoids the costly database scans.

E. Mining the FP-Tree using FP-Growth

The FP-Tree provides an efficient structure for mining, although the combinatorial problem of mining frequent patterns still has to be solved. For discovering all frequent item sets, the FP-Growth algorithm takes a look at each level of depth of the tree starting from the bottom and generating all possible item sets that include nodes in that specific level. After having mined the frequent patterns for every level, they are stored in the complete set of frequent patterns. FP-Growth takes place at each of these levels. To find all the item sets involving a level of depth, the tree is first checked for the number of paths it has. If it is a single path tree, all possible combinations of the items in it will be generated and added to the frequent item sets if they meet the minimum support. If the tree contains more than one path, the conditional pattern base for the specific depth is constructed. Looking at depth a in the FP-Tree of figure 2, the conditional pattern base will consist of the following item sets: $\langle b, e : 1 \rangle$, $\langle b, d : 1 \rangle$ and $\langle d, e : 1 \rangle$. The item set is obtained by simply following each path of a upwards.

Table 2: Conditional pattern base

Item	Conditional pattern
a	$\{\langle b, e : 1 \rangle, \langle d, e : 1 \rangle\}$
e	$\{\langle b, 2 \rangle, \langle b, d : 1 \rangle \langle d : 1 \rangle\}$
d	$\{\langle b : 3 \rangle\}$
b	ϕ

F. FP-Tree Construction

Constructing an FP-Tree from a fuzzy database is rather straightforward. The only difference to the original algorithm is that it is not enough to count the occurrences of an item, but the value of membership has to be considered as well. This membership value is then simply added to the overall count of the node. The database of fuzzy values in Table 3 is used for FP-Tree construction. Generating the FP-Tree from this database will lead to a tree containing the sums of the fuzzy values in the nodes. (See figure 2)

Table 3: Fuzzy Database

A	B	C	D	D
2	0	0.6	0	1
0	0.3	1	0	0
0.1	1	1	1	0
0	0.2	0	0.4	1
0	0	0.2	0	0.2
0	0	0.1	1	0.1
1	1	0	0	0.1

It is now easy to calculate the support of a path in the tree because it is simply the minimum of the path that is controlled. That tree can be used for conducting the FP-Growth algorithm.

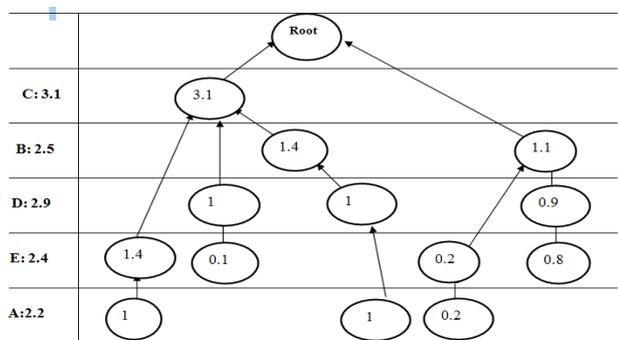


Figure 2: Fuzzy FP-Tree

F. Hybridizing FP-growth algorithm with Fuzzy Logic

Traditional quantitative association rule mining methods partition continuous domains into crisp intervals. When dividing an attribute into intervals covering certain ranges of values, the *sharp boundary problem* arises. Elements near the boundaries of a crisp set (interval) may be either ignored or overemphasized [13]. Fuzzy association rules overcome this problem by defining fuzzy partitions over the continuous domains. Moreover, fuzzy logic is proved to be a superior technology to enhance the interpretability of these intervals. Hence, fuzzy association rules are also expressions of the form $X \rightarrow Y$, but in this case, X and Y are sets of fuzzy attribute-value pairs and fuzzy confidence and support measure the significance of the rule.

FP-Growth algorithm does not produce the candidate item-sets, but it uses growth models to mine frequent pattern [13]. It compresses the database into a frequent pattern tree, but still retains itemsets associated information. Then it will divide such a compressed database into a group of conditional databases, each one of which is related to a frequent item-set

and mined. FP-growth algorithm is made up for these deficiencies. FP-growth algorithm does not produce frequent item-sets, replaced by an FP-tree. At the same time, FP-growth algorithm need transaction database only twice, which makes in the proceedings as soon as possible to close out database, which occupies a large number of storage, can be achieved.

IV. OUR PROPOSED ALGORITHM

In this paper, we are interested in finding the frequent patterns of genes and conditions which leads to discover the symptoms of diseases. Though, FP-growth algorithm shows it's potential over the Apriori algorithm to find the frequent patterns, we have used this FP-growth algorithm along with the support and confidence is being measured using the membership function of the fuzzy logic to find the frequent patterns which are not only using the crisp nature but also uses the fuzziness to measure the support and confidence to infer the association rules which ill dive us direction to discover the diseased symptoms. Therefore, we have hybridized the FP-growth algorithm with fuzzy logic. Our proposed algorithm is given step wise below.

Fuzzy FP-Growth Algorithm

Input: A gene expression dataset consisting of G no. of genes and C no. of conditions or samples, a set of membership functions, and a predefined minimum support threshold f .

OUTPUT: A Fuzzy FP-growth tree.

STEP 1: Transform the gene expression value g_{ij} of each gene g_j in the i^{th} row into a fuzzy set f_{ij} represented as $(f_{ij1}/R_{j1} + f_{ij2}/R_{j2} + \dots + f_{ijn}/R_{jn})$ using the given membership functions, where h_j is the number of fuzzy regions for I_j , R_{jl} is the l -th fuzzy region of I_j , $1 \leq l \leq h_j$, and f_{ijl} is v_{ij} 's fuzzy membership value in region R_{jl} .

STEP 2: Calculate the scalar cardinality $count_{jl}$ of each fuzzy region R_{jl} in all the genes as:

$$Count_{j_l} = \sum_{i=1}^n f_{ijl} \quad \text{Equation (3)}$$

STEP 3: Find $max-count_j = \text{Max}(count_{jl})$ for $j = 1$ to m , where h_j is the number of fuzzy regions for gene g_j and c is the number of conditions. Let $max-R_j$ be the region with $max-count_j$ for gene g_j . It will then be used to represent the fuzzy characteristic of conditions c_j in the later mining process.

STEP 4: Check whether the value $max-count_j$ of a fuzzy region $max-R_j$, $j = 1$ to C , is larger than or equal to the predefined minimum count $G * f$. If a fuzzy region $max-R_j$ satisfies the above condition, put the fuzzy region with its count in L_l . That is:

$$L_l = \{max-R_j \mid max-count_j \geq G * f, 1 \leq j \leq C\} \quad \text{Equation (4)}$$

STEP 5: While executing the above steps, also find the occurrence number $o(max-R_j)$ of each fuzzy region in the gene expression dataset.

STEP 6: Build the Header Table by keeping the fuzzy regions in L_I in the descending order of their occurrence numbers.

STEP 7: Remove the fuzzy regions of the items not existing in L_I from the transactions of the transformed database. Sort the remaining fuzzy regions in each transaction according to the order of the fuzzy regions in the Header Table.

STEP 8: Initially set the root node of the Fuzzy FP-growth tree as *root*.

STEP 9: Insert the genes in the transformed dataset into the Fuzzy FP-growth tree tuple by tuple. The following two cases may exist.

- If a fuzzy region $max-R_j$ in the currently processed i -th gene has appeared at the corresponding path of the Fuzzy FP-growth tree, add the membership value f_{ijl} of the region $max-R_j$ in the transaction to the node with $max-R_j$. Besides, calculate the membership values of the super-gene sets of $max-R_j$ in the path by the intersection operator and add the values to the corresponding elements of the array in the node.
- Otherwise, add a new node with $max-R_j$ to the end of the corresponding path and set the membership value f_{ijl} of the region $max-R_j$ in the currently processed i -th gene as the value in the node. Besides, calculate the membership values of the super-gene sets of $max-R_j$ in the path by the intersection operator and set the values to the corresponding elements of the array in the node. At last, insert a link from the node of $max-R_j$ in the last branch to the current node. If there is no such a branch, insert a link from the entry of $max-R_j$ in the Header Table to the current node.

After this step, the final Fuzzy FP-growth tree is built. In this step, a corresponding path is a path in the tree which corresponds to the fuzzy regions to be processed in objects (genes) according to the order of fuzzy regions appearing in the Header Table.

V. RESULT ANALYSIS

In this paper, a hybridized tree structure called the Fuzzy FP-growth tree has been designed. The tree can keep related mining information such that the database scans can be greatly reduced. A tree construction algorithm is also proposed to build the tree from a gene expression dataset. The construction process is similar to the FP-tree-like processing for building a tight tree structure except each node in the tree has to keep the membership value of the fuzzy region in the node as well as the membership values of its super-gene sets in the path.

We have tested both the approaches in Intel Dual Core machine with 2GB HDD. The OS used is Microsoft XP and all programs are written in MATLAB 8.0. We have observed the similar trends on runtime versus number of frequent patterns found in both Apriori and our proposed Fuzzy FP-growth approaches, the figure 3 shows the running time is significantly less as compared to Apriori approach. We have

tested the scalability of both the algorithms on the number of objects as well as number of columns, the figure 4 and figure 5. Our pattern based approach performs substantially better than Apriori based approach.

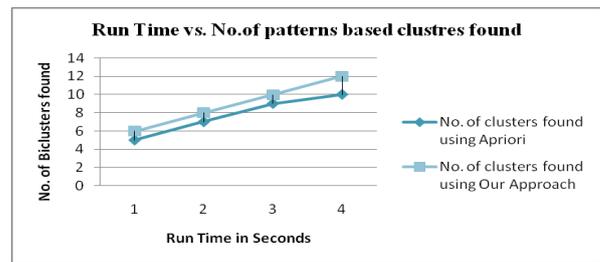


Figure 3: Run time VS No. of frequent pattern based clusters discovered

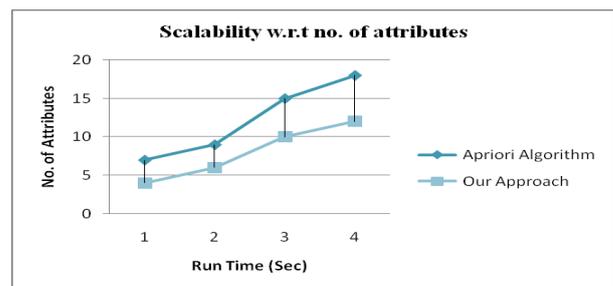


Figure 4: Scalability w.r.t No. of attributes

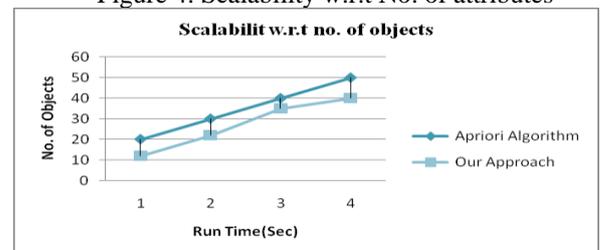


Figure 5: Scalability w.r.t No. of objects

VI. CONCLUSION

An advantage of FP-Growth is that the constructing FP-Tree has great performance of compression, and its process of mining can reduce the cost of rescanning data. In addition, it applies conditional FP-Tree on avoiding generating candidate item and testing examining process. About its disadvantage, mining process needs extra processing time and space to store that it continuously generates large conditional bases and conditional FP-Trees. In this paper, we combine the concept of fuzzy weight, fuzzy partition methods in data mining, and use FP-Growth to propose FP-Growth tree mining algorithm with all association rules. The heterogeneity and noisy nature of the data along with its high dimensionality decided us to use of a fuzzy association rule mining algorithm for the analysis.

A number of interesting associations have been found in this first approach. A deep study and empirical evaluation of the rules is however needed to confirm such associations. Our future work will be oriented towards modelling this algorithm

to find biclusters from gene expression dataset. Our work can be extended to find the overlapping frequent patterns using Rough Set method and also vertical data format method can be used instead of FP-Tree to find the frequent patterns from gene expression dataset.

REFERENCES

- [1] M.Anandhavalli Gauthaman, "Analysis of DNA Microarray Data using Association Rules: A selective study", *Ward Academy of Science, Engineering & Technology*, Volume-42, 2008, pp.12 to 15.
- [2] Sara C. Madeira and Arlindo L. Oliveira, "Biclustering Algorithms for Biological Data Analysis: A Survey", *IEEE Transaction on Computational on Biology & Bioinformatics*, Volume-1.No.1 January – March 2004, PP.24 – 25.
- [3] C. Gyorodi, R. Gyorodi "Minimum association rule in large Database", *Proc. of Oradea EMES'02*:45/50, Oradea, Romania, 2002, pp.12 – 16.
- [4] Y.chengand G.M church,"Biclustering of Gene Expression data",*Proc.Eight'1 Conf.Intelligent systems for molecular Biology(ISMB '00)*,pp.93-103,2000.
- [5] Wim pijis and Walter A. Kostens+,"How to find frequent patterns",*Econometric Institute Report EI 2005-24* june 1,2005
- [6] Bart Gothals,"survey on frequent pattern mining",*HIIT basic Research unit Dept of Computerscience University of Helsinki p.o.box 26,FIN-00014 Helsinki finland.*
- [7] Miao Wang, Xuequnshag, Zhanhuki, Li: "Research on microarray dataset mining", *VLDB2010 ph workshop*, September 13, 2010, Singapore
- [8] M. Anandhavalli ,M K Ghose, K Gautaman," Association Rule mining in Genomics", Member, *IACSIT, IAENG International Journal of computer theory and Engineering*
- [9] J.han, J.Pai, Y. Yin,"Mining Frequent patterns without candidate generation", *Proc .of Oradea EMES'02* :45-50. *Oradea, Romania,2002*
- [10] Pijls W., Bioch J.C. (1999)," Mining Frequent Itemsets in Memory-Resident Databases" In: *Postma E., Gyssens M. (Eds.) Proceedings of the Eleventh Belgium-Netherlands, Conference on Artificial Intelligence (BNAIC1999)*, pages 75–82.
- [11] Jiwei Han, Jian Pei, Yiwen Yen and Runying Mao: "Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach, *Data Mining and Knowledge Discovery*, 8, 53–87, 2004, 2004 Kluwer Academic Publishers
- [12] Liqiang Geng and Howard J. Hamilton. Interestingness measures for data mining: A survey. *ACM Computing Surveys*, 38(3):9, (2006)
- [13] F. Javier Lopez, Marta Cuadros, Armando Blanco and Angel Concha:"Unveiling Fuzzy Associations Between Breast Cancer Prognostic Factors and Gene Expression Data", 20th International Workshop on Database and Expert Systems Application, pp.338-342,(2009).
- [14] D. Dubois, E. Hüllermeier, and H. Prade, "A systematic approach to the assessment of fuzzy association rules," *Data Mining and Knowledge Discovery*, Vol. 13, pp. 167-192, 2006.
- [15] CHRISTIAN BORGELT, AN IMPLEMENTATION OF THE FP-GROWTH ALGORITHM, OSDM '05 PROCEEDINGS OF THE 1ST INTERNATIONAL WORKSHOP ON OPEN SOURCE DATA MINING: FREQUENT PATTERN MINING IMPLEMENTATIONS, (2005).
- [16] R. Das,D.K. Bhattacharyya and J.K. Kalita, Clustering Gene Expression Data Using An Effective Dissimilarity Measure, *International Journal of Computational Bioscience*, Vol. 1, No. 1, 2010,pp 55-68,(2010)

AUTHORS PROFILE

Shruti Mishra is a scholar of M.Tech(CSE) at Kaustuv Institute for Self Domain, Biju Pattanaik University, Bhubaneswar, Odisha, INDIA. Her research includes Data mining, Soft Computing Techniques etc.

Debahuti Mishra is an Assistant Professor and research scholar in the department of Computer Sc. & Engg, Institute of Technical Education & Research (ITER) under Siksha O Anusandhan University, Bhubaneswar, Odisha, INDIA. She received her Masters degree from KIIT University,

Bhubaneswar. Her research areas include Datamining, Bio-informatics, Software Engineering, Soft computing. Many publications are there to her credit in many International and National level journal and proceedings.She is member of OITS, IAENG and AICSIT. She is an author of a book Aotumata Theory and Computation by Sun India Publication (2008).

Shruti Mishra is a scholar of M.Tech(CSE) Institute of Technical Education & Research (ITER) under Siksha O Anusandhan University, Bhubaneswar, Odisha, INDIA. Her research areas include Data mining, Parallel Algorithms etc.

Sandeep Kumar Satapathy is Asst.Prof. in the department of Computer Sc. & Engg, Institute of Technical Education & Research (ITER) under Siksha O Anusandhan University, Bhubaneswar . He has received his Masters degree from Siksha O Anusandhan University, Bhubaneswar, Odisha, INDIA. His research areas include Web Mining, Data mining etc.

Dr.Amiya Kumar Rath obtained Ph.D in Computer Science in the year 2005 from Utkal University for the work in the field of Embedded system. Presently working with College of Engineering Bhubaneswar (CEB) as Professor of Computer Science & Engg. Cum Director (A&R) and is actively engaged in conducting Academic, Research and development programs in the field of Computer Science and IT Engg. Contributed more than 30 research level papers to many national and International journals. and conferences Besides this, published 4 books by reputed publishers. Having research interests include Embedded System, Adhoc Network, Sensor Network ,Power Minimization, Biclustering, Evolutionary Computation and Data Mining.

Dr. Milu Acharya obtained her Ph.D at Utkal University. She is a Professor in Department of Mathematics at Institute of Technical Education and Research (ITER), Bhubaneswar. She has contributed more than 20 research level papers to many national and International journals and conferences Besides this, published 3 books by reputed publishers. Her research interests include Biclustering, Data Mining , Evaluation of Integrals of analytic Functions , Numerical Analysis , Complex Analysis , Simulation and Decision Theory.