Frequent Itemset Matching for Real Time Applications using Reconfigurable Hardware Architecture

J. Samson Immanuel, G. Manoj, A. Amir Anton Jone, P. Esther Jebarani

Abstract: ***Data mining methods remain a quickly developing class of claims that are popular basic usage in numerous fields. An accumulative quantity of data increases the claim for calculating power. Usually human being utilizes enormous size of data and understands that the data and information are widely spread at particular pointer. The algorithms and techniques are known as data mining, remain developed to channel the breach. To utilize the demand for microprocessor systems and use of graphics processing units (GPU) there are numerous methods can be obtained. The added method operates on the hardware accelerators termed as Field programmable gate array (FPGA). Three data mining algorithms nominated for this review: In this apriori algorithm is best to mine the frequent itemsets from the extensive database, and the frequent itemsets are very useful to get the association rule for the discovery of knowledge. In this paper apriori algorithm is modified which reduces the large frequent itemsets and it has implemented in Xilinx Virtex-5 FPGA platform provides up to 5.58 × performance improvement over an equivalent software implementation. Evaluation and investigation are performed for the three selected algorithms using FPGA implementations. To precede with conclusion the investigations executed on common complications, restrictions and resources of various algorithms.***

I. INTRODUCTION

Data mining process gives useful information and provides interesting hidden information about the database since the data mining is also called knowledge data discovery. There are many techniques and several algorithms to find that useful information — data mining used for many purposes such as Sales promotion, Fraud detection, Intrusion Detection. Nowadays the researches are being performed to implementing data mining techniques in FPGA to reduces the computation power. There are many algorithms implemented in FPGA to achieve high performance such k-means algorithm increases the possible parallelism when implemented in reconfigurable hardware and results from approximately 200 times speedup over a software implementation [1].

An algorithm can be simply to a digital architecture and calculate the real performance of each the algorithm and also the design and implement the response on a field programmable gate array (FPGA) [2].FPGA implementation provides the orders of magnitude faster than the progressive software implementations[3] and parallel hardware to achieve higher rates of cluster than the algorithm implemented in software[4]. Hardware implementation of Decision Tree Classification yields high accuracy whereas handling giant datasets[5]. Productivity is increased significantly via design reusability, accomplished by hastening machine learning on FPGA's using pattern based disintegration[6].In the area of microarray data processing, computational performance to explore through the FPGA implementation[7]. FPGA is used to accelerate certain terribly C.P.U. intensive data-mining, and the results that run on the hardware is to display to the real world [8]. FPGA implementation shows the higher performance than the graphical processing unit (GPU) and general purpose processor (GPP)[9]. A reconfigurable platform executed more rapidly than software algorithm to acquire more amounts of common patterns and also reduces the mining time. FPGA computing platform consumes low power and the performance level is very high when accelerating with parallel data processing. FPGA is to use as a hardware accelerator which achieves high speed than the software implementation[11]. FPGA computing platform consumes low power and the performance level is very high when accelerating with parallel data processing[12].

The algorithm used in this paper is the best technique to analysis the market basket analysis. This modified algorithm from the Apriori algorithm where the conjunctive and dis-conjunctive method has introduced. This method used to evaluate a maximum possible limit of the number of items in the dataset and the frequent itemset reduced according to user specified. This paper proposes the hardware implementation for mining frequent itemset and finding the association rules from a market basket database or the database of glossary shop. It has done by using Xilinx 9.2i software and FPGA (Xilinx Virtex-5) with the clock rate of 100MHZ. Denoting time t as a data stream input through domain N, where the three performance measures such as storage, per-item processing time and the computation time are measured by preferably log(N,t).

Revised Manuscript Received on March 08, 2019.

J. Samson Immanuel, Assistant Professor, Department of Electronics and Communication Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India

G. Manoj, Assistant Professor, Department of Electronics and Communication Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India

A. Amir Anton Jone, Assistant Professor, Department of Electronics and Communication Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India

P. Esther Jebarani, Assistant Professor, Kovai Kalaimagal College of Arts and Science.

1) FP-Growth Algorithm

The algorithm [13] the Frequent Pattern Growth (FP-Growth) stays is an algorithm that mines usual item units disadvantaged of an high-priced applicant technology method. The divide-and-conquer method combined the numerous items that are inserted in FP-Tree which in turn holds the information of normal items. The Frequent Pattern Tree and Qualified Frequent Tree shares a set of objects as a result the object can individually mine new objects. An instance of Frequent Pattern Tree that signifies the established items has given in Fig 1.

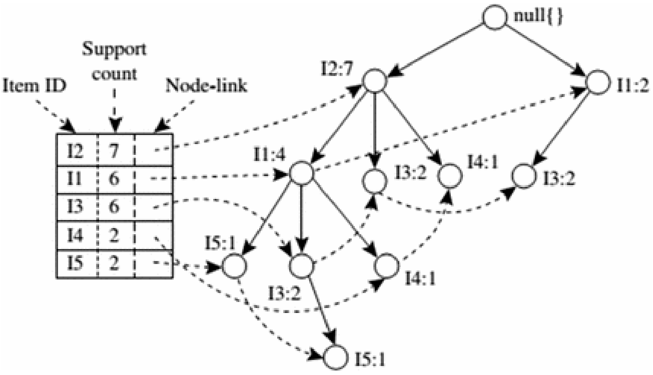
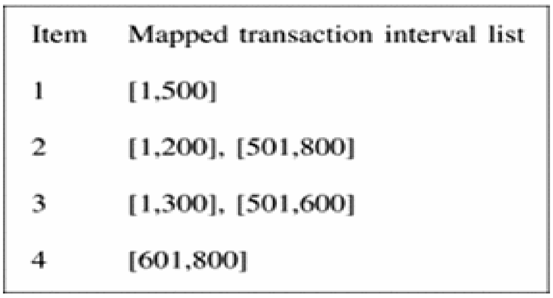


Fig. 1 Transition model of frequent pattern tree (FP-Tree) [13]

The algorithm FP-Growth solves the downside of distinguishing extended, commonplace pattern throughout looking out via slighter restrictive FP-Trees repetitively. An instance of the restrictive FP-Tree related to node 13 and important points of conditional FP-Tree are observed in Fig.1. The restrictive pattern supports “sub-database” consist of prefix course inside the Frequent Pattern Tree that are incorporated with each and every length of one item. Conditional FP-Tree generates all the ordinary patterns which are related with each and every regular length-1 item. In the specified way, while checking the time-honored patterns, the cost is reduced. Expanding the FP-Tree is time consuming when the information set is huge [14].

2) TM algorithm

The rule of Transaction Mapping (TM) [15] is like the EClaT rule that mines explicated frequent itemsets in the vertical know-how illustration. In this TM algorithm, the IDs of each rectangular measure itemset remodeled and mapped into subsequent transactions at another location in inventory. The intersection performance of the transaction are evaluated using depth-first search order that depends on the itemsets. An example of the TM approach is shown in table1.



Item	Mapped transaction interval list
1	[1,500]
2	[1,200], [501,800]
3	[1,300], [501,600]
4	[601,800]

Table. 1 Example of transaction mapping. Reproduced with permission from [15]

When the quantity of minimal support is elevated, the TM technique will pack together the TM IDs hooked on to the continuous series at certain intervals. After compressing itemset into inventory, the intersection time has significantly improved. When comparing TM algorithmic and FP-Growth, the TM algorithm attains higher performance over FP-Growth. TM algorithms acknowledge that it consumes lesser unit’s much less well-known patterns. Even although it's therefore, the TM algorithmic rule is nevertheless lower regarding system velocity compared to the FP-Growth* algorithmic rule.

i. Apriori Algorithm

The objective of the regular pattern mining is to examine which itemset holds the transaction information exceedingly. Association rule mining is one of the critical practicality in data processing [17], and the core of information mining is extracting association rules. The vital field of research is mining for association rules from the database of the sales transaction between products or items. The purpose of these rule is to identify the unknown relationship between the items in the transactional database and to make the decision. The association rule evaluates from frequent itemsets. Frequent itemsets which are purchased together frequently called frequent itemsets. The occurrence of the items together should be higher than the minimum support which is defined by the user.

After generating the frequent itemsets according to the minimum support, an association rule is applied to those frequent itemsets for example if the frequent itemsets are $\{I_1, I_2, I_3\}$ its rules are $\{I_1 \rightarrow I_2,I_3\},\{I_2 \rightarrow I_1,I_3\},\{I_3 \rightarrow I_2,I_1\},\{I_1,I_2 \rightarrow I_3\},\{I_1,I_3 \rightarrow I_2\},\{I_2,I_3 \rightarrow I_1\}$.

Confidence threshold should have calculated for each rule from this we can obtain the percentage the items which have frequently purchased together.

For example $\{X \rightarrow Y\}$ where X and Y are the frequent items in the transaction [18]. Confidence threshold determines how many percentages the items X and Y purchased together in the transaction. The formula for the confidence threshold as given below.

$$\text{Confidence}(A,B \Rightarrow C) = \frac{\text{Number of transaction containing C}}{\text{Number of transaction containing both A and B}} \quad (1)$$

The hidden information about the database and this information is used to adjusting the store layouts and for cross-selling where profits and promotions increased.

II. ALGORITHM TO GENERATE FREQUENT ITEMSETS

Input:
Database D
Minimum Support ε
Output:
Large Frequent Itemsets
Method:
Scan the transactional database
T = {A,B,C,D,E,F,G,H}

```
Find out the conjunctive patterns from the transaction
If E, F, G, H is user specified
Conjunctive = ( E,F,G,H )
Disjunctive  = {A,B,C,D}
K= attributes from the conjunctive sets
Iteration I=0
For all combinations of (k-1) number of attributes
Do
generate candidate K-1 itemsets
generate frequent itemsets from candidate K-1 itemsets
where support count of generated itemsets >= ε
If successful, then go to step 13
Else i=i+1 and go to step 6
Return sets of frequent itemsets
End
```

Frequent pattern mining algorithm is straightforward to execute and extremely straightforward, is used to mine all frequent itemsets in an exceedingly database. This algorithm searches indepthly for the frequently used itemsets wherever k-itemsets square measure would generate the successive k+1-itemsets. Each of the k-itemset with same frequency must be greater than or equal to the minimum threshold level[19] if the case is vice-versa then it is called as candidate itemsets. Initially the algorithm inspects the information and eliminates the conditions that the user doesn't specify in the itemsets. The conjunctive set find the frequency of 1-itemsets that includes only one item from the numerable h item with certain information. The frequency of 1-itemsets is employed to search out the itemsets in 2-itemsets which successively is used to search out 3-itemsets until there exist without any additional k-itemsets. If the subset of a itemset is frequently used even though the itemset is very huge then, it is called as the frequently used itemset— the sample transactional database of market basket analysis as given below.

Table. 2 Sample Transactions

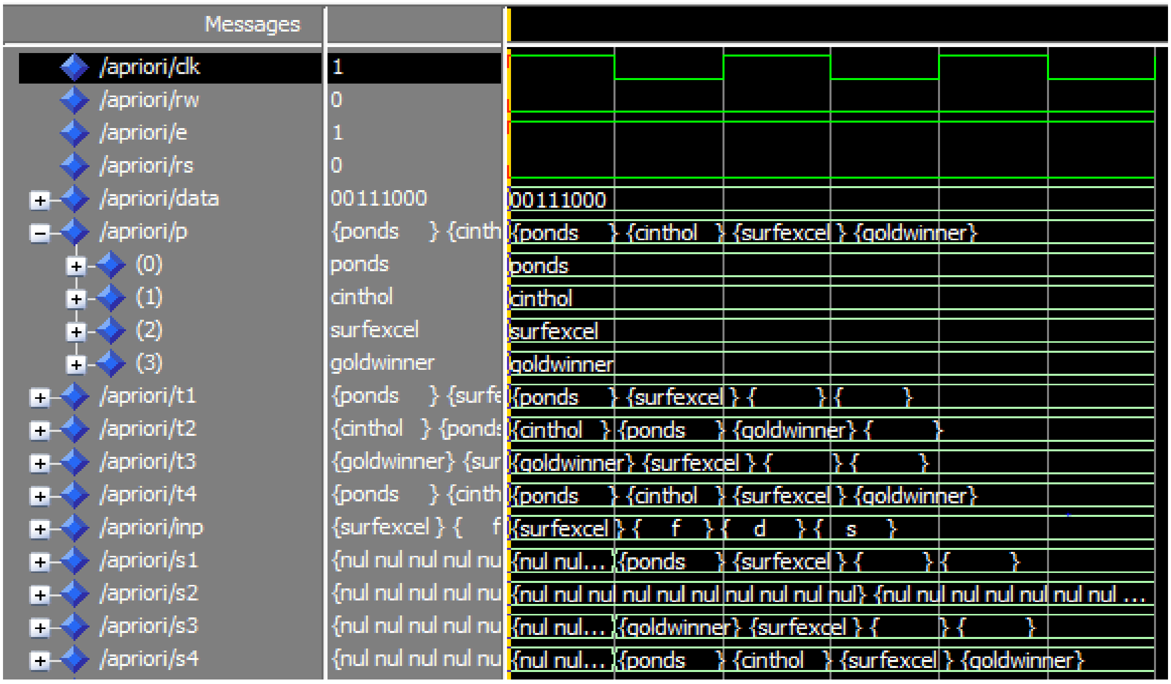
Transc.ID	Items purchased
T1	Ponds,surfexcel
T2	cinthol,ponds, Goldwinner
T3	Goldwinner, surfexcel
T4	ponds ,cinthol, surfexcel ,goldwinner,

In the above transaction, if the user specifies surfexcel, the transaction T2 will be eliminated because transaction T2 does not contain surfexcel since the database filtered according to the user specified and reduce the frequent itemsets.

III. RESULTS AND SIMULATIONS

1. Simulation of the Design

The design is simulated and synthesized using Xilinx 9.2, and the outcomes of the design as given in Figure2 and 3. Fig.2 shows the implementation of conjunctive and disconjunctive patterns in the sample database according to the given input.



```
Messages
/apriori/clk     1
/apriori/rw      0
/apriori/e       1
/apriori/rs      0
/apriori/data    00111000                 00111000
/apriori/p       {ponds    } {cinth      {ponds    } {cinthol  } {surfexcel } {goldwinner}
  (0)            ponds                    ponds
  (1)            cinthol                  cinthol
  (2)            surfexcel                surfexcel
  (3)            goldwinner               goldwinner
/apriori/t1      {ponds    } {surfe      {ponds    } {surfexcel } {        } {        }
/apriori/t2      {cinthol  } {ponds      {cinthol  } {ponds    } {goldwinner} {        }
/apriori/t3      {goldwinner} {sur       {goldwinner} {surfexcel } {        } {        }
/apriori/t4      {ponds    } {cinth      {ponds    } {cinthol  } {surfexcel } {goldwinner}
/apriori/inp     {surfexcel } {    f     {surfexcel } {   f    } {   d    } {   s    }
/apriori/s1      {nul nul nul nul nu     {nul nul... {ponds    } {surfexcel } {        } {        }
/apriori/s2      {nul nul nul nul nu     {nul nul nul nul nul nul nul nul nul nul} {nul nul nul nul nul nul nul ...
/apriori/s3      {nul nul nul nul nu     {nul nul... {goldwinner} {surfexcel } {        } {        }
/apriori/s4      {nul nul nul nul nu     {nul nul... {ponds    } {cinthol  } {surfexcel } {goldwinner}
```

Fig. 2 Output Wave Form for Conjunctive and Disconjuvctive Patterns

Fig 3 shows the product output which gives the 100% confidence calculated from the equation(1). The number of the transaction should be calculated using the algorithm to find frequent itemsets.

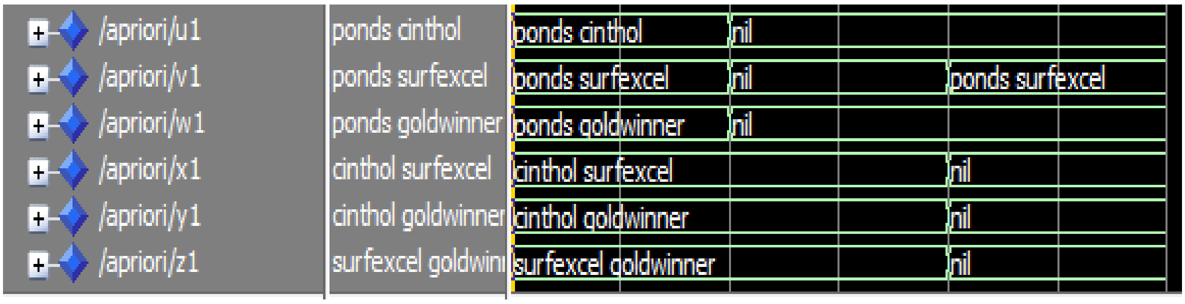
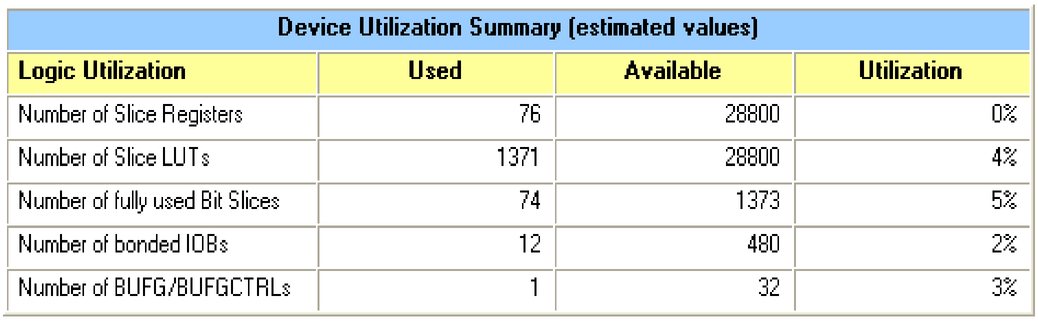


Fig. 3 Output waveform For Confidence of the Item sets

2. Hardware Implementation of System

The hardware implementation [20] of the system is done using the Xilinx software, the device utilization summary in Table 2. The devices utilization of the slices register is 76 registers, Lookup table(LUT) is 1371 and so forth as given in the table.

Table. 2 Device Utilization Summary



Device Utilization Summary (estimated values)			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	76	28800	0%
Number of Slice LUTs	1371	28800	4%
Number of fully used Bit Slices	74	1373	5%
Number of bonded IOBs	12	480	2%
Number of BUFG/BUFGCTRLs	1	32	3%

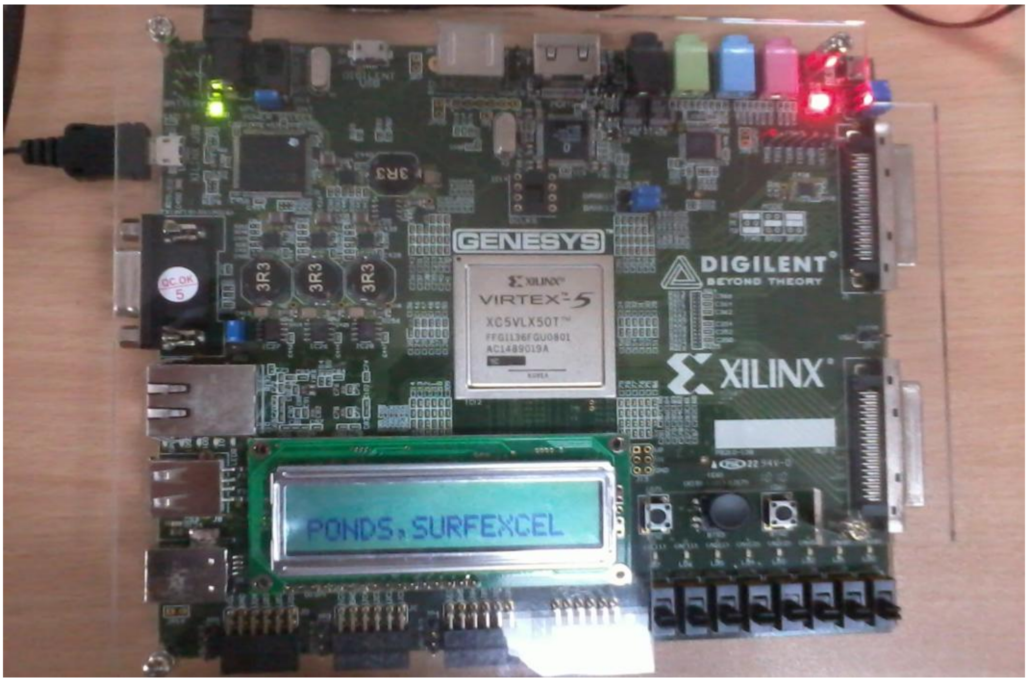


Figure 4. FPGA output

The FPGA implementation of the XILINX Virtex 5 as shown in figure 4, the performance of the system has dined and then analyzed.

IV. CONCLUSION

The analysis of data mining algorithm has done in FPGA. The algorithm provided the accurate, frequent itemsets where conjunctive and disconjunctive methods reduced the number of frequent itemsets from which absolute output is obtained. The process has done in VHDL coding using Xilinx software, and the output is simulated using ModelSim simulator. Simulating a design assures that the algorithm works perfectly, but it does not mean that the algorithm to be mapped to real digital circuits.

The program of the data mining algorithm has loaded in FPGA(Xilinx Virtex-5 XC5VLX50T)with the frequency of 100MHz then the output is displayed in LCD in the FPGA kit.

REFERENCES

1. S. Che, J. Li, J. W. Sheaffer, K. Skadron, and J. Lach, "Accelerating compute-intensive applications with GPUs and FPGAs," in Proc. SASP, 2008, pp. 101-107.
2. M. Estlick, M. Leeser, J. Theiler, and J. J. Szymanski, "Algorithmic transformations in the implementation of k-means clustering on reconfigurable hardware," in Proc. ACM FPGA, 2001, pp. 103-110.
3. D. Anguita, A. Boni, and S. Ridella, "A digital architecture for support vector machines: theory, algorithm, and FPGA implementation," IEEE Tras. Neural Netw., vol. 14, no. 5, Sept.2003.
4. Z. Baker and V. Prasanna. Efficient hardware data mining with the Apriori algorithm on FPGAs[C]. In Proceedings of the Thirteenth Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM '05),2005.
5. G. A. Covington, C. L. G. Comstock, A. A. Levine, J. W. Lockwood, and Y. H. Cho, "High-speed document clustering in reconfigurable hardware," in Proc. FPL, 2006, pp. 411-417.
6. R. Narayanan, D. Honbo, G. Memik, A. Choudhary, and J. Zambreno, “An FPGA implementation of decision tree classification,” in Proc. DATE, 2007.
7. Karthik Nagarajan & Brian Holland & Alan D. George & K. Clint Slatton & Herman Lam," Accelerating Machine-Learning Algorithms on FPGAs using Pattern-Based Decomposition" in J Sign Process Syst (2011) 62:43–63 DOI 10.1007/s11265-008-0337-9
8. H. M. Hussain, K. Benkrid, H. Seker, and A. T. Erdogan, “FPGA of K-means algorithm for bioinformatics application: An accelerated approach to clustering Microarray data,” in Proc. AHS, 2011, pp. 248-255.
9. Hanaa M. Hussain, Khaled Benkrid, Ahmet T. Erdogan, Huseyin Seker,” Highly Parameterized K-means Clustering on FPGAs: Comparative Results with GPPs and GPUs" in 2011 International Conference on Reconfigurable Computing and FPGAs
10. S. Sun and J. Zambreno. Design and Analysis of a Reconfigurable Platform for Frequent Pattern Mining[J]. IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 9, pp. 1497-1505,2011.
11. Grigorios Chrysos, Panagiotis Dagritzikos, Ioannis Papaefstathiou, Apostolos Dollas,” Novel and Highly Efficient Reconfigurable Implementation of Data Mining Classification Tree" in 2011 21st International Conference on Field Programmable Logic and Applications.
12. P. Skoda, B. Medved Rogina, V. Sruk,” FPGA implementations of data mining algorithms”, in MIPRO 2012, May 21-25,2012, Opatija, Croatia.
13. Shaobo Shi, Yue Qi, Qin Wang,” Acceleration Intersection Computation in Frequent Itemset Mining with FPGA" in high-performance computing and communication & 2013 IEEE international conference on embedded and ubiquitous computing(HPCC_EUC).
14. Han J, Pei J, Yin Y (2000) Mining frequent patterns without candidate generation. ACM SIGMOD Rec 29(2):1–12
15. Meenakshi A (2015) Survey of Frequent Pattern Mining algorithms in horizontal and vertical data layouts. Int J Adv Comput Sci Technol 4(4):48–58
16. Song M, Rajasekaran S (2006) A transaction mapping algorithm for frequent itemsets mining. IEEE Trans Knowl Data Eng 18(4):472–481
17. M. Zaki, S. Parthasarathy, M. Ogihara, and W. Li. New Algorithms for Fast Discovery of Association Rules.Proc. 3rd Int. Conf. on Knowledge Discovery and Data Mining (KDD’97, Newport Beach, CA), 283–296. AAAI Press, Menlo Park, CA, USA 1997.
18. R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules.Proc. 20th

Int. Conf. on Very Large Databases (VLDB 1994, Santiago de Chile), 487–499. Morgan Kaufmann, San Mateo, CA, USA 1994

19. C.Borgelt and R.Kruse. Induction of Association Rules: Apriori Implementation. In Proceedings of the 15th Conference on Computational Statistics, 2002
20. M.Estlick, M.Leeser, J. Szymanski, and J. Theiler. Algorithmic Transformations in the Implementation of K-means Clustering on Reconfigurable Hardware. In Proceedings of the Ninth Annual IEEE Symposium on Field-Programmable Custom Computing Machines 2001 (FCCM '01), 2001

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication