ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

Performance Evaluation of Multicore System through Mining Techniques

Ramdas Jare[1], Prof. Vrushali Desale[2]

M. E Scholar, Dept. of Computer Engineering, Dr.D.Y.Patil College of Engineering, Ambi, Maharashtra, India[1]

Assistant Professor, Dept. of Computer Engineering, Dr.D.Y.Patil College of Engineering, Ambi, Maharashtra, India[2]

ABSTRACT: Discovering the frequent patterns in transactional databases Apriori algorithm considered as one of the most important. Apriori algorithm is a masterstroke algorithm of association rule mining. Increased possibility of the Multicore processors is impose us to upgrade the algorithm and applications so as to accomplishment the computational power from multiple cores finding frequent item sets is more upscale in terms of computing resources utilization and CPU power.Apriori Algorithms are used on very big data sets with high dimensionality. Therefore, parallel computing can be applied for mining using association rules. The process of association rule mining consists of finding frequent item sets and generating rules from the frequent item data sets. Finding frequent itemsets is more expensive in terms of CPU power consumption and computing resources utilization. Thus, majority of parallel apriori algorithms focus on parallelizing the process of discovering frequent item set. The computation of frequent item sets mainly consist of creating the candidates and counting them. In parallel frequent itemsets mining algorithms addresses the issue of distributing the candidates among processors such that their creation and counting is effectively parallelized. This paper presents comparative study of these algorithms.

KEYWORDS: Parallel data mining, frequent itemsets, association rules, apriori algorithm,

I. INTRODUCTION

Accumulation of plentiful data from various sources of the society but a little knowledge situation has lead to knowledge discovery from databases which is also called data mining. Data mining techniques use the existing data and retrieve the helpful information from it which is not directly visible in the original data. As data mining algorithms deal with large amount of data, the primary concerns are how to store the data in the main memory at run time and how to increase the run time performance. Sequential algorithms cannot supply scalability, in terms of the data dimension, size, or runtime performance, for such large number of databases. Because the data sizes are increasing to a large quantity, high-performance parallel and distributed computing is used to get the advantage of more than one processor to handle these huge quantities of data. Data mining deals with huge volumes of data to extract the useful knowledge. Association Rule Mining (ARM) or frequent item set mining is an important functionality of data mining. The apriori algorithm is one of the best algorithms for discover frequent itemset from a transaction database. As data mining mainly deals with large volumes of data, the main issue is how to improve the performance of the algorithm. One way of improving the performance of apriori is parallelizing the process of generate frequent itemsets. The rest of the paper is organized as follows. In Section 2 related work is overviewed. In Section 3 concepts of association rule mining are discussed and apriori algorithm is described. In Section 4 comparative analysis of parallel apriori algorithms is given.

II. RELATED WORK

Many parallel ARM algorithms have been given which represent transactions using either horizontal data format or vertical data format [4, 7]. In horizontal data format, data is presented as transaction ID versus items sold in each transaction whereas in vertical data format, data is presented as each item versus transaction ids in which the item is sold. There are several parallel association rule mining algorithms based on data set partitioning like Count Distribution, Data Distribution, Candidate Distribution, Common Candidate Partition,Parallel Partition [1, 5, 9, 10].Apriori algorithm is a primary algorithm for association rule mining. A supermarket wants to implement a bundling sale. They need to find

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

the items buy or asset together frequently. This procedure evaluates customer buying habits by recommending associations between the different items that customers put in their "shipping baskets". The result can help retailers establish or expand marketing strategies by getting to know which items are frequently bought together by customers. Apriori is a positive solution to this Association rules mining problem.Traditional methods waste lot of time to resolve the problems or decision making for profitable business. Data mining formulate databases for finding unknown or hidden patterns, finding anticipating information that experts may omit. Hence, this paper reviews the various trends of data mining and its relative applications from past to present and discusses how adequately can be used for targeting profitable customers in campaigns and utilize the multiple cores of the processor for faster execution. Apriori algorithm [1] was recommended by R Agarwal and R Srikant in 1994 for exploring frequent item set for Boolean association rule, It deliver a frequent item set in transactional database as an output. It's an efficient algorithm for finding frequent items. Disadvantage is it generates massive number of candidate item set. Repeatedly scanning the transaction databases. Record filter approach [4] only those transactions are considered to determine the support count of candidate set whose length is greater than the length of candidate item set. If length of candidate item set is k, only transaction whose length is at least k is considered as k length candidate set cannot exist in the transaction record whose length is is then this approach takes less time as compared to classical apriori algorithm this it improves the efficiency of apriori algorithm and memory management. It removes the complexity of process. Disadvantage is memory optimization. AVI algorithm [7] Transaction database is vertical, item set union and identification intersection is used. For item set X, t (X) = {tid | tid is transaction id, t belongs to D and t supports X}; for the identification set Y, i(Y) = y belongs to Y item set(y), item set (y) that y corresponding to the transaction item set [5]. Sampling method [6] chooses the arbitrary sample S for given database D, and then investigate frequent item sets in the S rather in D. As we have to scan only sample S instead of whole database D, it recover time. This approach sacrifices some efficiency.

III. PROPOSED SYSTEM

A. *Proposed WorkAnd Architecture:*

as mentioned in figure 1, proposed system firstly divides the input stream into chunks with the help of bsw algorithm for the parallel implementation of association rule mining. further these chunks will be assign to independent parts of processor i.e. core using set affinity methodology. with our proposed system, we can easily utilize available processor computational power very efficiently for frequent item mining procedure.

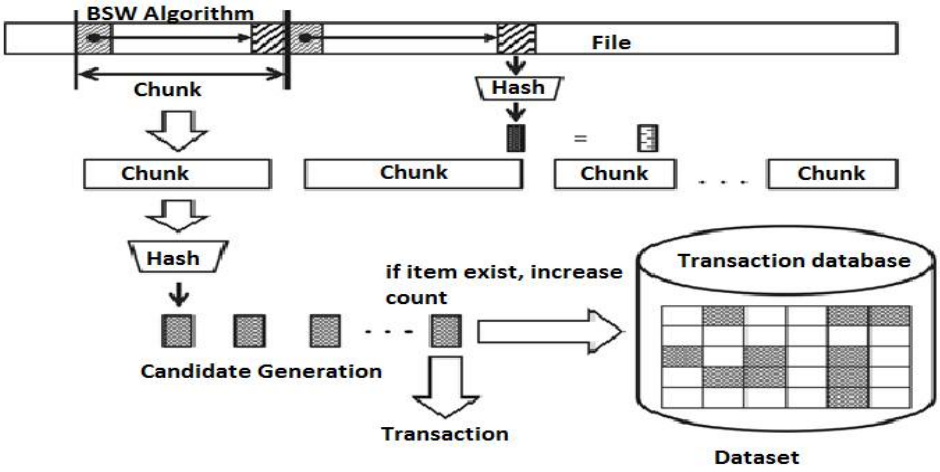


Figure: 1. Proposed System Architecture.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

B. *Algorithms & Mathematical Model*

Algorithm:
Input: D, Database of transactions; min_sup, minimum support threshold
Output: L, frequent item sets in D.
Method:
(1) L1=find_frequent_1-itemsets(D);
(2) for(k=2; Lk-1≠Φ; k++){
(3) Ck=apriori_gen(Lk-1, min_sup);
(4) for each transaction t∈D{
(5) Ct=subset(Ck,t);
(6) for each candidate c∈Ct
(7) c.count++;
(8) }
(9) Lk={ c∈Ck |c.count≥min_sup }
(10) }
(11) return L=UkLk ;
Procedure apriori_gen(Lk-1:frequent(k-1)-itemsets)
(1) for each itemset l1∈ Lk-1{
(2) for each itemset l2∈ Lk-1{
(3) if(l1 [1]= l2 [1])∧ (l1 [2]= l2 [2]) ∧…∧(l1 [k-2]= l2 [k-2]) ∧(l1 [k-1]< l2 [k-1]) then {
(4) c=l1∞l2;
(5) if has_infrequent_subset(c, Lk-1) then
(6) delete c;
(7) else add c to Ck ;
(8) }}}
(9) return Ck;
Procedure has_infrequent_subset(c: candidate k-itemset;
Lk-1:frequent(k-1)-itemsets)
(1) for each(k-1)-subset s of c {
(2) if s ∉ Lk-1 then
(3) return true; }
(4) return false;

C. *Mathematical Model:*

Set of input – $ti \in T$ ....(1)

ti : Each transaction.

T : Transaction dataset.

Processing **–**

1.Support Count – $\sigma(x) = \{t_i/x \subseteq t_i,\ t_i \in T\}$ ....(2)

ti/x : Frequency of x item.

$\sigma(x)$:Support count of x item

2.Support – $S(x \rightarrow y) = \sigma(x \cup y)/N$ ....(3)

3.Confidence – $C(x \rightarrow y) = \sigma(x \cup y)/\sigma(x)$ ....(4)

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

4.BSW Chunking – $BSW_{chunk} = \frac{M.N}{\max\{T_{chunking}, T_{disk}\} + T_{marshal}}$ ...(5)

M: Size of segment.
N: Number of chunker thread.
$T_{chunking}$: Chunking overhead.
T_{disk} : Segment set time duration.
$T_{marshal}$: Marshaling overhead.

Set of output -

Frequent item set – $tx \epsilon T$ ....(6)

tx: Frequent item set.

Mining time factor - ts : Mining time using serial.
tp: Mining time using parallel.

D. *Apriori Algorithm*

Apriori algorithm is the most established association rule mining algorithm. It is based on the apriori principle that all the nonempty (at least one) subsets of a frequent itemset must be frequent. It is a two step process.

Step 1: The prune step

It scans the entire database to preceive the count of each candidate in Ck where Ck represents candidate k- itemset. The count of each itemset in Ck is match up with a predefined minimum support count to find whether that itemset can be arranged in frequent k-itemsetLk.

Step 2: The join step

Lk is natural joined with itself to generate the next candidate k+1-itemset Ck+1. The main step here is the prune step which requires scanning the whole 1database for finding the count of each itemset in whole candidate k-itemset. If thedatabase is enormous then it requires more time to find all the frequent itemsets in the DB.

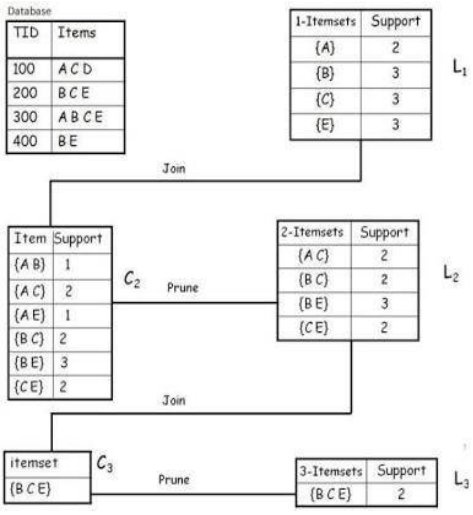


Fig. 5 : Example for apriori algorithm

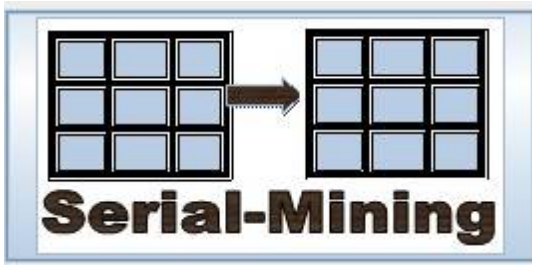


Fig:6 Serial Platform

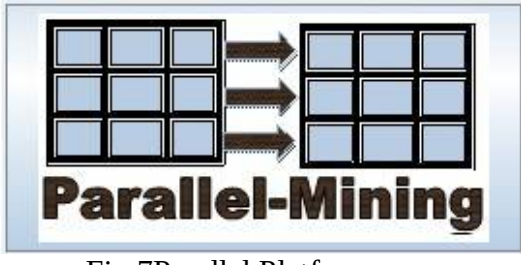


Fig:7Parallel Platform

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

Database Design
Transaction :-
-A transaction is a unit of work that is performed against a database.
-It is a very small unit of a program and it may contain several low level tasks.
Item :-
-unit of data containing in a record describing a particular attributs.
-A data item describes an atomic sate of a particular object concerning when looking at databases.
Database Example

Database:-
-it is an organized collection of data (transactions),queries, reports, views, other object.
-A database is a collection of information that is organized so that it can easily be accessed ,managed and updated.

Ex1:- Database 1(Fruits):- This database contains 10 itemset and 20 transaction .(items: mango,banana, apple …etc)

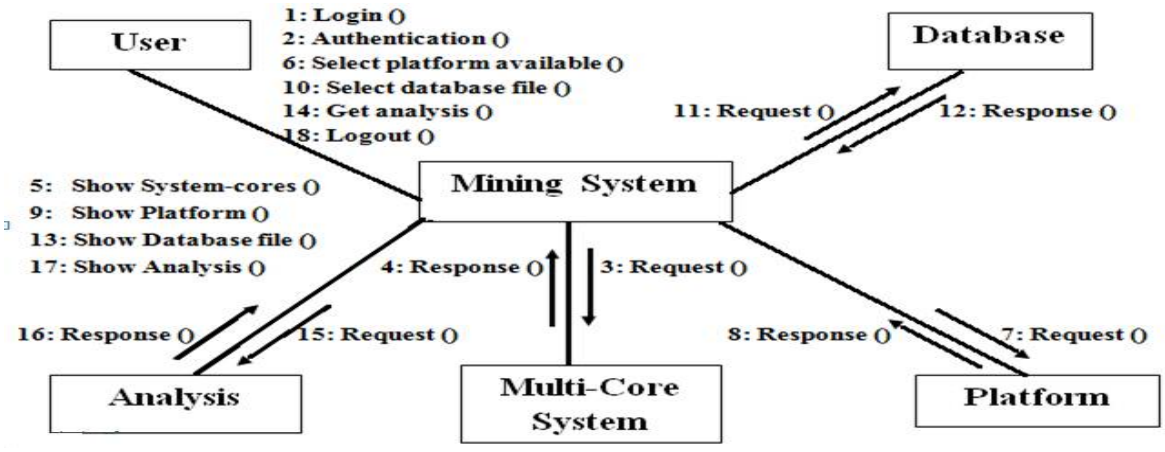
Ex2:-Database 2(Medicines):- this database contains 20 itemsets and 25 transactions (items :Lomofem, asprin, Decold, crocin,Codlever tabs…etc)

Ex3:- Database 3(Book stall):- this database contains 25 itemset and 100 transaction (items: applied science,TOC,MIT,BAI,SSDA,SDMT…etc)

Serial And Parallel Mining:-
Candidate Generation:- The Apriori Algorithm identifies item set which are sub set of at least transaction in the database. Apriori uses bottom up approach where frequent subset are extended one item at a time a step known as candidate generation.

IV. MODULE

In this Input output processing user first login system and authenticate with password. It sends a request to multicore system and get response.



1. User Interaction Module –

This module provides various interfaces for accessing system features as per user activity. User interaction module handles the all functions of user also this module specially designed for interaction between user and system.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

2. Authentication Module –

Establishing the authenticity of a person or other entity. Not to be confused with authorization- defining access rights to resources. This module designed for security purpose which generates unique username and password. User can login by using password and username system check the username and password and give the response. After enter the login mining systems check the username and password if correct it gives the response to the multicore system. Also user can change the password by using old password. The following frames show the how to authenticate the user.

3. Serial Mining module –

Mining system provides two platforms to the user serial mining and parallel mining. In serial mining the items are calculating one by one so serial mining give the response after long time and also it works slowly the performance of serial mining is low so this is the main disadvantages of serial mining.

4. Parallel Mining Module –

There are two options available to the user if he wants to select the parallel mining mining system send request to the platform. Platform give the response and system show the platform which is requested. User selects the database file and calculates the items on parallel platform. After calculating the frequent items set on two platforms system show the analysis.

5. Performance analyzer –

One of the main concepts of the project is comparison between serial and parallel mining. How they work on different processor we have to check on different processor and we have observe that the performance of serial mining and parallel mining.

V. RESULTS

To compare the performance of proposed work, various experiments are executed using proposed Apriori algorithm with external load balancing and BSW. We are applying single threaded and multi threaded on Apriori with the help of multi core system by using HPC.The experiments cover candidate generation and frequent item set mining. There were 3 to 4 system used to test and observe comparison between serial mining parallel mining. The average results for both the execution time and the CPU usage are different after checking on different system.

System Details: -

A. Processor: - Intel® Core™ i3-2330M CPUSpeed: - 2.20 GHz RAM: - 3GBO.S/ Java version: - 64 Bit

1. Minimum Support – 30

Database		Serial		Parallel	
Tns.	**Item**	**Freq_ Item**	**Time**	**Freq_ Item**	**Time**
10	20	119	109 m/s	119	78 m/s
20	25	2003	2043 m/s	2003	223 m/s
20	100	21151	197596 m/s	21151	2031 m/s

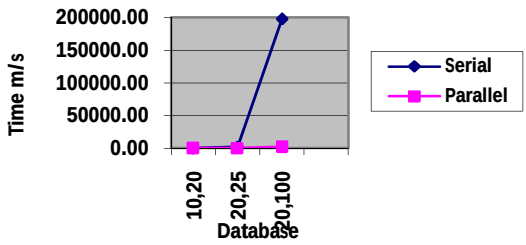


Fig: 3. Comparison graph for Serial &Parallel Mining

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

Database	Serial		Parallel	
Tns. Item	**Freq_Item**	**Time**	**Freq_Item**	**Time**
10 20	70	63 m/s	70	78 m/s
20 25	372	243 m/s	372	94 m/s
20 100	3689	6324 m/s	3689	449 m/s

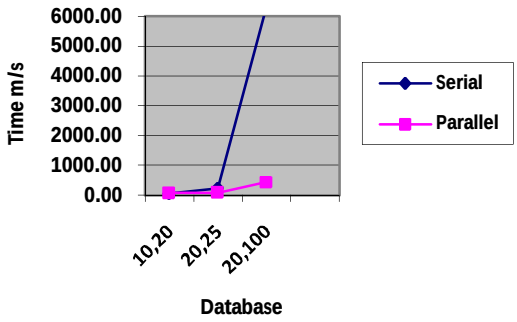


Figure: 4. Comparison graph for Serial and Parallel Mining (Processor: - Intel® Core™ i3-2330M CPU)

3. Minimum Support – 60

Database	Serial		Parallel	
Tns. Item	**Freq_Item**	**Time**	**Freq_Item**	**Time**
10 20	46	31 m/s	46	31 m/s
20 25	206	141 m/s	206	78 /s
20 100	2168	2318 m/s	2168	271m/s

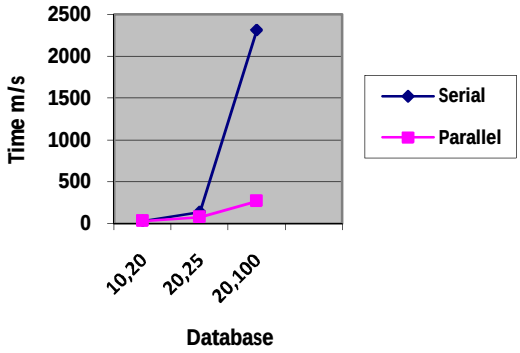


Figure: 5. Comparison graph for Serial and Parallel Mining (Processor: Intel® Core™ i3-2330M CPU)

VI. CONCLUSION AND FUTURE WORK

We are going to use real time database for frequent items calculation**.** Multi-core processors represent an important new trend in computer architecture. To utilize their full potential, applications will need to move from a single to a multi-threaded model. For the improvement of our system we can use graphics processor in future. We can also distribute mining processing load in network.The performance of the parallel apriori algorithms depends on the processing time and the data communication cost. The data communication cost can be reduced by using client- server architecture like Parallel Partitioning Algorithm and exchanging only the counts as in Count Distribution Algorithm. The processing time depends on the database layout, number of times the database is scanned and the size of the candidates generated. Vertical database layout speeds up the searching process as demonstrated in the Apriori Algorithm and reduces the database scanning time. Thus a parallel apriori algorithm using client-server architecture with only counts exchanged and using vertical database layout can achieve balanced trade-off between the processing time and the data communication cost and using multicore processing power we can easily reduce overhead of mining process.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

REFERENCES

1. KhadidjaBelbachir, HafidaBelbachir, "The Parallelization of Algorithm Based on Partition Principle for Association Rules Discovery", In Proceedings of International Conference on Multimedia Computing and Systems(ICMCS), IEEE, May 2012.
2. RuowuZhong, Huiping Wang, "Research of Commonly Used Association Rules Mining Algorithm in Data Mining", In Proceedings of International Conference on Internet Computing and Information Services(ICICIS), IEEE, September 2011.
3. Aziz Ginwala, Priyankakonde, PriyankaBhalekar, MayuriJambhulkar ,Prof.SunilYadav "Performance Enhancement Scheme For Multithreading Application Using Chunking Mechanism" .
4. V.Umarani, Dr.M.Punithavalli, "A Study On Effective Mining Of Association Rules From Huge Databases", International Journal of Computer Science and Research (IJCR), Vol. 1 Issue 1, 2010.
5. Xindong Wu , Vipin Kumar, J. Ross Quinlan, JoydeepGhosh, Qiang Yang ,Hiroshi Motoda, "Top 10 Algorithms in Data Mining", Knowledge and Information Systems, Volume 14, Issue 1, pp 1-37, Springer, January 2008
6. Mohammed J. Zaki, SrinivasanParthasarathy, MitsunoriOgihara, Wei Li, "Parallel Data Mining for Association Rules on Shared-Memory Systems", Data Mining and Knowledge Discovery, Springer, 2001.
7. Eui-Hong (Sam) Han, George Karypis, Vipin Kumar, "Scalable Parallel Data Mining for Association Rules", IEEE Transactions on Knowledge and Data Engineering, Volume:12 , Issue: 3, May/June 2000.
8. Mohammed J. Zaki, "Parallel and Distributed Association Mining: A Survey", IEEE Concurrency, Vol 7, Issue 4, pp 14-25, October 1999.
9. Mohammed J. Zaki, SrinivasanParthasarathy, MitsunoriOgihara, Wei Li, "Parallel Algorithms for Discovery of Association Rules", Data Mining and Knowledge Discovery, Vol 1, Issue 4, pp 343-373, Springer, December 1997.
10. Mohammed J. Zaki, SrinivasanParthasarathy, MitsunoriOgihara, Wei Li, "A Localized Algorithm for Parallel Association Mining", Proceedings of the ninth annual ACM symposium on Parallel algorithms and architectures, ACM 1997.
11. RakeshAgrawal, John C. Shafer, "Parallel Mining of Association Rules", IEEE Transactions on Knowledge and Data Engineering, December 1996.
12.

BIOGRAPHY

Ramdas Popat Jare received the BE degree in Information Technology from Government collage of engineering Karad, Kolhapur in 2013.he is currently lecturer in Dr.D.Y.Patil Polytechnic, Akurdi in the department of computer engineering. His current research interest includes Data Mining, Cloud computing, Data retrieval, mobile grid computer.