SPECIAL ISSUE

# THE IIOAB JOURNAL

VOLUME 7 : NO 2 : JULY 2016 : ISSN 0976-3104

Institute of Integrative Omics and
Applied Biotechnology Journal

*Dear Readers,*

*It is with immense pleasure that I extend a warm welcome to each of you to our distinguished scientific journal dedicated to the exploration of Frontiers in Intelligent Systems Development.*

*As the Editor of this IIOAB Journal issue, I am honored to witness the unfolding of unprecedented advancements and pioneering endeavors in the domain of intelligent systems. Our journal serves as a platform for the dissemination of transformative research, innovative methodologies, and cutting-edge applications that propel the development of intelligent systems across diverse domains.*

*Your relentless pursuit of excellence in crafting intelligent systems stands at the forefront of technological innovation. Your contributions in algorithmic design, machine learning, artificial intelligence, and their interdisciplinary applications inspire breakthroughs that redefine the boundaries of what intelligent systems can achieve.*

*The impact of intelligent systems permeates various facets of our lives, from healthcare and autonomous vehicles to finance and industry. Your dedication to pushing the boundaries of these systems not only drives technological progress but also holds the potential to revolutionize industries and enhance the quality of human existence.*

*I encourage each of you to share your groundbreaking research, submit your visionary insights, and engage in vibrant discussions within the pages of our journal. Let us collectively nurture an environment where ideas flourish, collaborations thrive, and knowledge transcends boundaries.*

*Thank you for your invaluable contributions to the evolution of intelligent systems. Your commitment to advancing the frontiers of technology through innovative developments is truly commendable, and I eagerly anticipate the wealth of transformative insights that will emerge from your contributions.*

*Warm regards,*

*Prof. Steven Lawrence Fernandes*
*Editor-in-Chief*

ISSN: 0976-3104
SPECIAL ISSUE (FISD)
*Fernandes*

THE IIOAB JOURNAL

EDITORIAL OPEN ACCESS

# FRONTIERS IN INTELLIGENT SYSTEMS DEVELOPMENT

**Steven Lawrence Fernandes**[*]

*Sahyadri College of Engineering & Management, ECE Dept, Mangalore, INDIA*

## ABSTRACT

*Science and technology has been growing rapidly and there is great need for developed on intelligent applications which would be useful to society. This special issue is intends to bring out intelligent systems in the area of science and technology. Key objective is to provide a guide to the rapidly developing resources in science and technology and their impact on human lives. Papers of this special issue include intelligent systems developed in the area of Statistical Analysis, Machine Learning, Soft Computing, Next Generation Computing, and Medical Image Processing. In the following section, we start by providing the readers of this special issue a brief overview of these research papers.*

**Published on:** 4th Feb-2016

**KEY WORDS**

Machine Learning, Soft Computing, Medical Image Processing

***Corresponding author:** **Email:** steven.ec@sahyadri.edu.in **Tel:** +91-9844760875

It’s my great pleasure to publish the special issue entitled FRONTIERS IN INTELLIGENT SYSTEMS DEVELOPMENT (FISD) and I thank all the contributors, editors, and reviewers for their all contributions and cooperation.

In the he first paper in this special issue, ‘‘Implementation of area efficient multiplier and adder architecture in digital fir filter’’, Srividya presents two architectures for multiplier design, they are modified booth algorithm and Vedic algorithm. Vedic algorithm is used for implementation as it consumes less area than modified booth algorithm.

The second paper by Prakash kumar et al. intend in analyzing the effect of network failure on QoS and reliability of the system in the presence of high request rate and network traffic. Performance of existing load balancing algorithm is investigated and compared in faulty environment.

The next paper by Prasenjit Mukherjee Sasi and Baisakhi Chakraborty propose a novel Hybrid Knowledge Provider System (HKPS) where permutation-combination (PC) based parsing technique and Grammatical Rules (GR) based parsing technique have been applied on a single system. HKPS is an automated system that shall be able to extract text and image based knowledge data from database.

Two-fold techniques for optimizing system performance using Trigger based VM Migration technique is presented by Prakash kumar et al. This gets activated when CPU temperature increases beyond an upper threshold value, called Hotspot. A Network File System (NFS) based dynamic load balancing strategy is proposed for better system resource utilization.

The analytical approach towards Classification of text documents using integer representation and regression: an integrated approach is the focus of the next paper. Ajit Danti and SN Bharath Bhushan describe integer representation using ASCII values of the each integer and linear regression for classification of text documents. Extensive experimentation is carried out on four publically available databases show the effectiveness of the proposed mode.

The subsequent paper deals with a student evaluation model using Bayesian network in an intelligent e-learning system. Baisakhi Chakraborty and Meghamala Sinha focuses on an evaluation model to correctly detect the knowledge level of each student based on their response to questions. The uncertainty factor has been defined by terms guess and slips parameters.

An in-depth review is performed in the next paper on Inter of Things (IoT) enabled in smart store by Ramesh S Nayak et al. The ideology of the smart store is to notify the store owner about the stock and various other requirements through an application in their phone. This enables the Store owner to notify his supplier to refill the stock.

A novel approach for human activity recognition is presented in the next by Kishor H Walse et.al. Benchmark dataset is considered from the WISDM laboratory, which is available in public domain. Author has performed experiment using AdaBoost.M1 algorithm with Decision Stump, Hoeffding Tree, and Random.

Final two papers are related to Computer Aided Diagnosis (CAD) of early cancer detection. Medical studies show that cancer can be easily cured if it is detected at an early stage. The authors of the last two papers are Steven Lawrence Fernandes et.al who presented two novel techniques for early cancer detection. The developed techniques are analyzed using the images obtained from cancer hospitals and validated with after seeking the advice of a cancer medical practitioner.

## ABOUT THE GUEST EDITOR

***Prof .Steven Lawrence Fernandes** is a member of Core Research Group, Karnataka Government Research Centre of Sahyadri College of Engineering and Management, Mangalore, Karnataka. He has received Young Scientist Award by Vision Group on Science and Technology, Government of Karnataka. He also received grant from The Institution of Engineers (India), Kolkata for his Research work. He completed his B.E (Electronics and Communication Engineering) with Distinction from Visvesvaraya Technological University, Belagavi, Karnataka and M.Tech (Microelectronics) with Distinction from Manipal University, Manipal, Karnataka. Currently he is perusing his Ph.D. in the area of Pattern Recognition. His Ph.D work "Match Composite Sketch with Drone Images" has received patent notification (Patent Application Number: 2983/CHE/2015) from Government of India, Controller General of Patents, Designs & Trade Marks. He has 5 years of industry experience working at STMicroelectronics Pvt. Ltd and Perform Group Pvt. Ltd. He has published several papers in peer-reviewed International Journals having Thomson Reuters Web of Science Impact Factor and IEEE, Springer, Elsevier International Conferences. He is also serving has reviewer and guest editor for several Science Citation Indexed and Scopus Indexed International Journals.*

ISSN: 0976-3104
SPECIAL ISSUE
Srividya.

ARTICLE OPEN ACCESS

# IMPLEMENTATION OF AREA EFFICIENT MULTIPLIER AND ADDER ARCHITECTURE IN DIGITAL FIR FILTER

**Srividya**

*Sahyadri College of Engineering & Management, ECE Dept, Mangalore, INDIA*

## ABSTRACT

*The title of this paper include an implementation of low area efficient multiplier and adder architecture in digital FIR filter design.FIR filter are mainly used in digital signal processing application in which multiplier and adder are the basic fundamental blocks. Efficiency of filter design depends on the architecture used for multiplier and adder block which differ in the delay, area and power. In this paper two architectures are used for multiplier design, they are modified booth algorithm and Vedic algorithm, in which Vedic algorithm is used for implementation as it consumes less area than modified booth algorithm.And for design of adder block efficient architecture is used which include carry save adder. Hence the output responses of FIR filter is obtained using Vedic multiplier and carry save adder. The multiplier and adder architecture are simulated and synthesized using Xilinx ISE tool. The above two multiplier architecture are also simulated, synthesized and physical design of it is done using Cadence tool in order to obtain the GDSII file*

**Received on:** 29th-May-2015
**Revised on:** 30th-July-2015
**Accepted on:** 01st-Aug-2015
**Published on:** 4th-Jan-2016

KEY WORDS

*FIR filter; Modified booth algorithm; Vedic algorithm; Carry save adder; GDSII*

***Corresponding author:** Email: svsvidya82@yahoo.co.in **Tel:** +91-9480503355

## INTRODUCTION

In the recent trends, VLSI technology has brought a significant development in the area of chip design which mainly depends on the factors like area, speed and power. Considering all these factors, much architecture are developed to implement Digital FIR filter which is used in digital signal processing applications. Filter is implemented using multipliers and adder blocks. Selection of efficient architecture for multiplier and adder is a major factor to be considered which in turn improves the efficiency of FIR filter. Digital FIR filter is the linear time invariant filter. A linear time invariant is the filter which does not vary with time and it interacts with the input signal and filter coefficients through a process called linear convolution and hence produces the output response. The synthesized architectures are shown in the following sections. In this paper two architecture for multiplier are simulated and synthesized and the physical design of it is performed using cadence tool and the adder is simulated and synthesized using Xilinx tool.

## MATERIALS AND METHODS

### Existing Algorithm

Array multiplier is the existing algorithm and one of the most widely used multiplier technique and it is a long multiplication process. It has a regular structure and used for unsigned multiplication. The computation cost and time taken by array multiplier is more. From the simulation and synthesis result obtained for array multiplier using Xilinx tool, the delay of array multiplier is found to be 17.522ns.So booth multiplier is used for multiplication process. Booth multiplier multiplies two signed numbers and it is faster than array multiplier. The operation of booth algorithm includes arithmetic shift operation after addition and subtraction operation. The drawback of this type of multiplier is that this uses a large number of add and subtraction operation which becomes inconvenient to design parallel multiplier. Another drawback is that when there is isolated ones the algorithm becomes inefficient. From the simulation and synthesis result for booth multiplier, the delay is found to be 6.712ns. These drawbacks are overcome by modified booth algorithm and Vedic algorithm.

## Modified booth algorithm

Modified booth algorithm is used to multiply signed as well as unsigned numbers and the partial product is reduced to N/2 from N, if N is the multiplier. Modified booth algorithms are so called as radix-2, radix-4, and radix-8 depending on the number which is taken as a base. Modified booth algorithm is faster than array multiplier and booth multiplier. Advantage of modified booth algorithm is that its computation is faster than conventional multiplier as it halves the partial products and it is used for longer operands whereas booth algorithm has partial products same as that of the multiplier bits and can be used only for smaller operands. Disadvantage of this is that it requires and encoder circuit to encode the multiplier bits which consumes more area. This encoding of multiplier is done by grouping the bits in blocks of two(radix-2), blocks of three (radix-4), blocks of four(radix-8),such that one bit overlapping of each block is performed. And the encoding of the bits is performed based on the booth recording table which is shown.**[Table–1, Table–2, Table–3].**The halving of partial product is obtained by shifting and adding for every second column [1].

**Table: 1. Radix-2 booth recoding table**

| Block A(Multiplier) | Re-coded digits | Operations on B(Multiplicand) |
|---|---|---|
| 00 | 0 | 0*B |
| 01 | +1 | +1*B |
| 10 | -1 | -1*B |
| 11 | 0 | 0*B |

**Table: 2. Radix-4 booth recoding table**

| Block A(Multiplier) | Re-coded digits | Operations on B(Multiplicand) |
|---|---|---|
| 000 | 0 | 0*B |
| 001 | +1 | +1*B |
| 010 | +1 | +1*B |
| 011 | +2 | +2*B |
| 100 | -2 | -2*B |
| 101 | -1 | -1*B |
| 110 | -1 | -1*B |
| 111 | 0 | 0*B |

**Table: 3. Radix-8 booth recoding table**

| Block A(Multiplier) | Re-coded digits | Operations on B(Multiplicand) |
|---|---|---|
| 0000 | 0 | 0*B |
| 0001 | +1 | +1*B |
| 0010 | +1 | +1*B |
| 0011 | +2 | +2*B |
| 0100 | +2 | +2*B |
| 0101 | +3 | +3*B |
| 0110 | +3 | +3*B |
| 0111 | +4 | +4*B |
| 1000 | -4 | -4*B |
| 1001 | -3 | -3*B |
| 1010 | -3 | -3*B |
| 1011 | -2 | -2*B |
| 1100 | -2 | -2*B |
| 1101 | -1 | -1*B |
| 1110 | -1 | -1*B |
| 1111 | 0 | 0*B |

## Vedic algorithm

Vedic algorithm is one of the ancient algorithms used for fast multiplication process. The Vedic algorithm is implemented based on Urdhva Tiryakbhyam (vertical and crosswise) Sutra. Out of 16 sutras; this is one such sutra which is used for all multiplication cases. Here partial product can be generated by concurrent addition of partial product.The multiplier using Vedic algorithm is independent of clock frequency of the processor because of parallel calculation of partial product and addition. One such feature of Vedic algorithm is reducing multi-bit multiplication into single bit multiplication. The carry propagation is reduced from LSB to MSB, as the partial product is generated in single step. This is well known algorithm which consumes less area compared to modified booth algorithm. The 2x2 Vedic multiplier blocks multiplies two bit binary numbers and this type of multiplication uses Urdhva Triyagbhyam algorithm. Consider two numbers ‘a’ and ‘b’,each of two bits then a[0]&b[0] represents LSB and a[1]&b[1] represents MSB.The output is 4 bit with y[0] forms the LSB,y[1] forms the second bit from LSB,y[2] forms the third bit from LSB and y [3] forms the carry bit.

### Carry save adder

Carry save adder is one such adder with lesser delay in comparison with different types of adders like ripple carry adder, carry look ahead adder, carry select adder and so on. Operation of Carry save adder is same as that of full adder with three inputs. In carry save adder, sum and carry is calculated separately and then both are added to give the final sum. Carry save adder also consumes less area compared to other types of adder and it has a delay of 9.043ns.

## RESULTS

### Modified booth algorithm

**Figure–1** shows the RTL view of modified booth algorithm using cadence tool.The RTL view represents the encoder circuit which encodes the multiplier and then multiplies with the multiplicand to generate the partial products and these partial products are then added using carry save adder to get the final result. **Figure–2** shows physical design of modified booth algorithm using cadence tool. The physical design shows the layout of the design which are obtained by process such as partitioning,floorplanning and placement, clock tree synthesis, routing,compaction and finally the physical verification process.

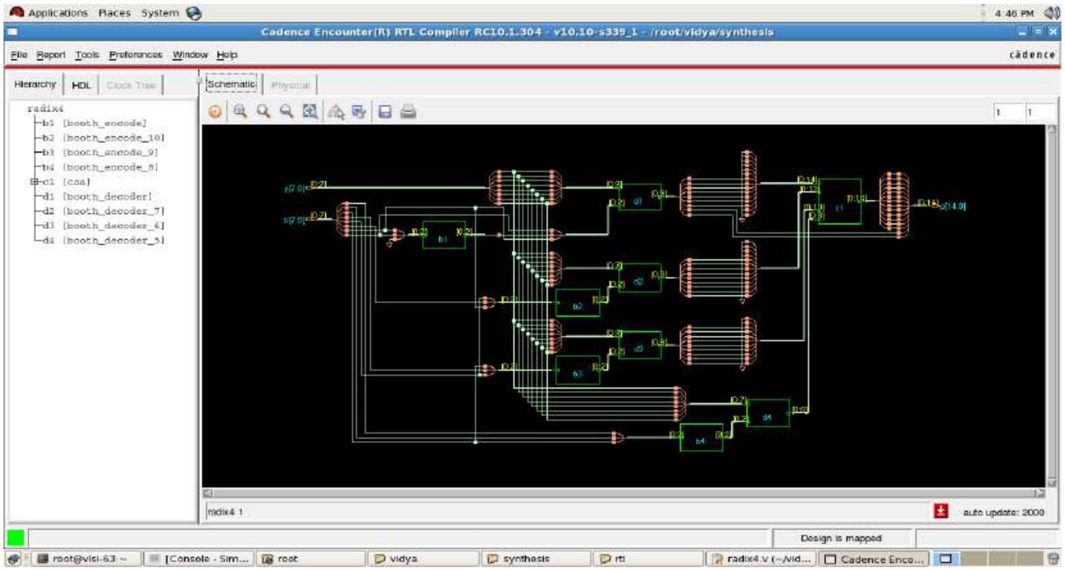
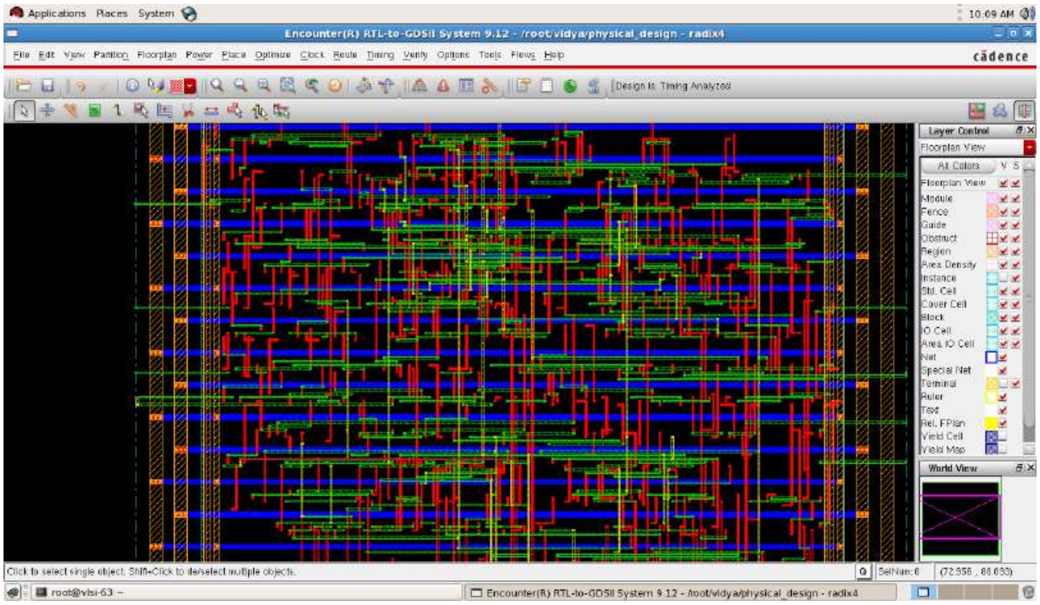


**Fig: 1. RTL view of modified booth algorithm using cadence tool**

### Vedic algorithm

The 4x4 multiplier block can be easily constructed using four 2x2 bit multiplier blocks. And the partial product generated is fed to the addition tree and lower 2 bit of partial product are taken as least two bits to the result. Again 8x8 multiplier block is constructed using four 4x4 multiplier block and the partial product are then added to get the final result.8x8 multiplier blocks are better than 4x4 multiplier block because computations are performed which are of 8 bit in many kind of processors. Here addition is performed using carry save adder which has less delay compared to all other adder. Carry save adder also consumes less area compared to other adders such as carry look ahead adder, carry select adder. **Figure– 3** shows the RTL view of 8x8 multiplier using cadence tool.This figure indicates that using the parallel techniques the computation can be easily performed. **Figure– 4** represents the physical design of 8x8 Vedic multiplier which include the process of generating the layout of the design. The final step of physical design process is the GDSII file generation which are further used for fabrication process. **Figure–5** shows the simulation results of 8x8 Vedic multiplier which explains that multiplication of two 8 bit number obtains a 16 bit result.



Encounter(R) RTL-to-GDSII System 9.12 - /root/vidya/physical_design - radix4

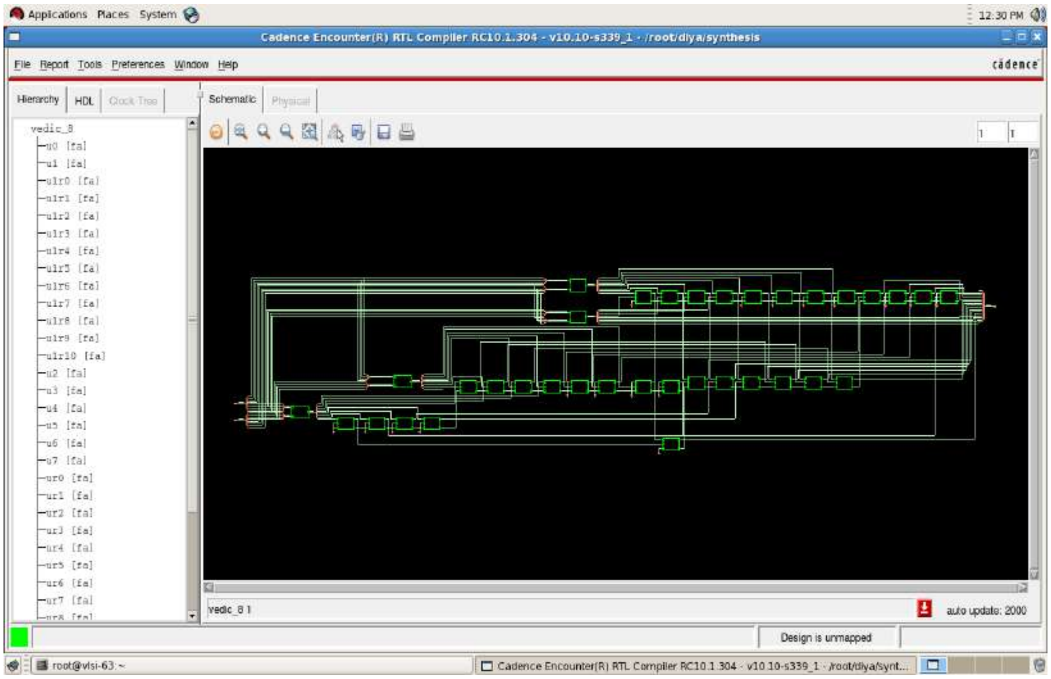
Design is: Timing Analyzed

Click to select single object. Shift+Click to de/select multiple objects.

root@vlsi-63:~

Encounter(R) RTL-to-GDSII System 9.12 - /root/vidya/physical_design - radix4

**Fig: 2. Physical design of modified booth algorithm using cadence tool**



Cadence Encounter(R) RTL Compiler RC10.1.304 - v10.10-s339_1 - /root/diya/synthesis

```
vedic_8
 ├u0 [fa]
 ├u1 [fa]
 ├u1r0 [fa]
 ├u1r1 [fa]
 ├u1r2 [fa]
 ├u1r3 [fa]
 ├u1r4 [fa]
 ├u1r5 [fa]
 ├u1r6 [fa]
 ├u1r7 [fa]
 ├u1r8 [fa]
 ├u1r9 [fa]
 ├u1r10 [fa]
 ├u2 [fa]
 ├u3 [fa]
 ├u4 [fa]
 ├u5 [fa]
 ├u6 [fa]
 ├u7 [fa]
 ├ur0 [fa]
 ├ur1 [fa]
 ├ur2 [fa]
 ├ur3 [fa]
 ├ur4 [fa]
 ├ur5 [fa]
 ├ur6 [fa]
 ├ur7 [fa]
 └ur8 [fa]
```

vedic_8 1

Design is unmapped

root@vlsi-63:~

Cadence Encounter(R) RTL Compiler RC10.1.304 - v10.10-s339_1 - /root/diya/synt...

**Fig: 3. RTL view of 8x8 Vedic multiplier using cadence tool**

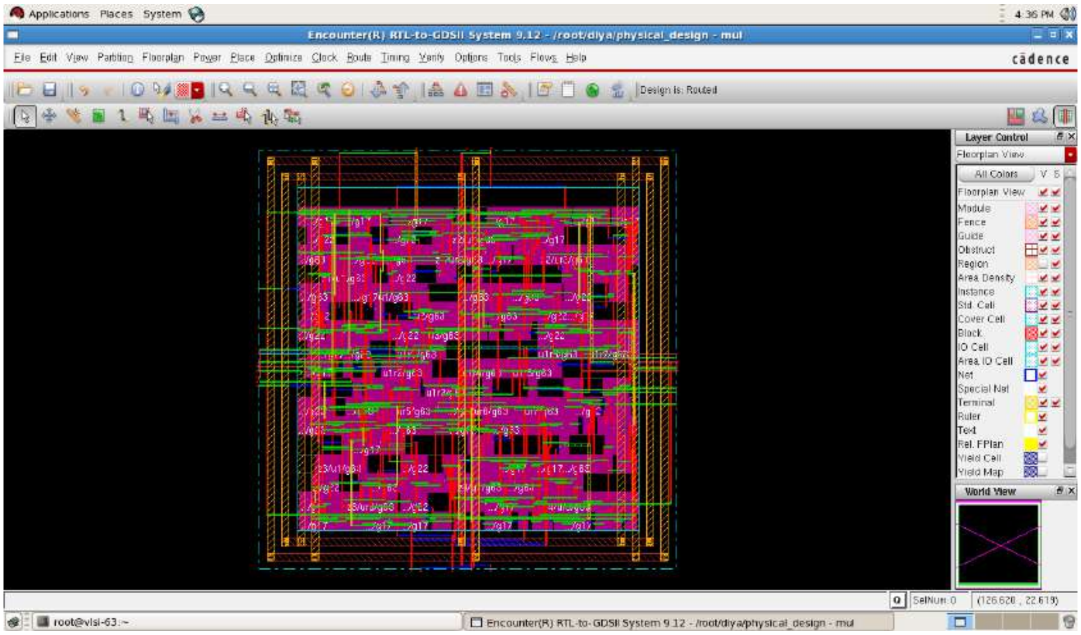
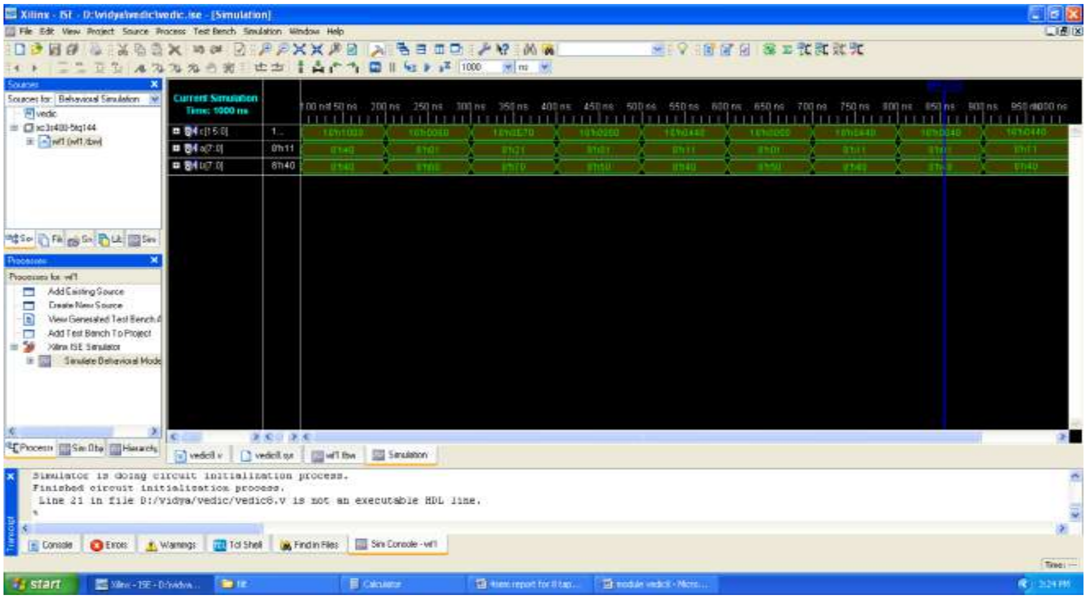


**Fig: 4. Physical design of 8x8 Vedic multiplier using cadence tool**



```
Simulator is doing circuit initialization process.
Finished circuit initialization process.
 Line 21 in file D:/vidya/vedic/vedic8.v is not an executable HDL line.
```

**Fig: 5. Simulation result of 8x8 Vedic multiplier using Xilinx tool**

## DISCUSSION

From the point of implementation of efficient multiplier and adder architecture, It was found that modified booth algorithm is better since it has lesser delay and consumes low power than Vedic multiplier. But in terms of cell area usage, it has seen that Vedic algorithm is far more efficient multiplier. While considering the efficient adder structure, carry save adder was found to be one such adder with lesser delay and low power consumption. **Table– 4** shows the Comparison of modified booth algorithm and Vedic algorithm used for multiplication operation.

**Table: 4. Comparison table of multiplier**

| Multiplier | Cell area | Leakage power(nW) | Dynamic power(nW) | Total power(nW) |
|---|---|---|---|---|
| Radix4 Modified booth algorithm (8x8) | 6912 | 210.760 | 136531.5 | 136742.348 |
| Vedic algorithm (8x8) | 6044 | 280.756 | 357930.524 | 358211.280 |

## CONCLUSION

This paper presents an efficient architecture for multiplier and adder and from the analysis result, modified booth algorithm is found to be efficient in terms of power consumption and Vedic algorithm is efficient in terms of area consumption. So it is better to consider a hybrid architecture which involves the combination of modified booth algorithm and Vedic algorithm for design of FIR filter.

## ACKNOWLEDGEMENT

I would like to thank Mr. Praveen Kumar, Assistant Professor of Sahyadri college of Engineering & Management for helping me in completion of this project.

.

## CONFLICT OF INTEREST

Author declares no conflict of interest.

## FINANCIAL DISCLOSURE

No financial support was received to carry out this project.

## REFERENCES

[1] Narasimha AR, Rajasekhar K, Rani S. [2012] Implementation of low area and power efficient architecture for Digital FIR filter. *IJARCSSE*, 2(8): 238-244.

[2] Chandel D, Kumawat G, Lahoty P, Chandrodaya VV, Sharma S. [2013] Booth Multiplier: Ease of multiplication. *IJETAE,* 3(3): 326-330.

[3] Shukla P, Gahlan NK, Kaur J. [2012] Techniques on FPGA Implementation of 8-bit Multiplier. *IJCST,* 3(2): 455-463.

[4] Kumar KSG, Prasannam JD, Christy MA. [2014] Analysis of Low Power, Area and High Speed Multipliers for DSP Applications. *IJETAE*, 4(3): 278-282.

[5] Choubey R, Arif M. [2014] Low Power and Area Optimized Using Modified Booth Algorithm Radix-2 and Ra-dix-4. *IJETAE*, 4(4): 641-646.

[6] Chaudhary M, Narula MS. [2013] FPGA Implementations of Booth's and Baugh-Wooley Multipliers. *IJITEE*, 3(1): 221-224.

[7] Ganeshkumar G, Charisma V. [2012] Design of high speed Vedic multiplier with Vedic mathematics tech-niques. IJSRP, 2(3): 1-5.

[8] Babulu K, Parasuram G. [2011] High speed Arithmetic Logic using booth multiplication. *IJCSIT*, 2 (3): 4-9.

[9] Ashenden JP. [2008] Digital design-An embedded system approach using verilog. *Elsevier, USA*

[10] Warren HS. [2012] Montgomery Multiplication. http://www.hackersdelight.org/MontgomeryMultiplication.pdf

[11] Kumar H. [2013] Implementation and analysis of power, area and delay of array, Urdhva, Nikilam Vedic multiplier. IJSRP, 3(1): 1-5.

[12] Lamberti F, Andrikos N, Antelo E, Montuschi P. [2011] Reducing the Computation Time in (Short Bit-Width) Two's Complement Multipliers. *IEEE Transactions on Computers*. 60 (2):148-156.

[13] Seo YH, Kim DW. [2010] A New VLSI Architecture of Parallel Multiplier-Accumulator Based on Radix-2 Modified Booth Algorithm. *Very Large Scale Integration*, 18(2):201-208.

ISSN: 0976-3104
SPECIAL ISSUE
Kumar et al.

THE IIOAB JOURNAL

ARTICLE OPEN ACCESS

# FAULT AWARE LOAD BALANCING ALGORITHM FOR CONTENT DELIVERY NETWORK

**Prakash kumar**[1*], **Krishna Gopal**[1], **JP Gupta**[2]
[1]*Department of Computer Science Engineering & IT, JIIT, Noida, INDIA*
[2]*Hydrocarbons Education and Research Society, New Delhi, INDIA*

## ABSTRACT

*With the increasing use of data sharing, traffic over the internet has increased significantly. There is a need to effectively to manage the load over the servers and maintain the overall system performance with better Quality of Service (QoS). To maintain better QoS, Content Delivery Network (CDN) is used. CDNs offer services that improve network performance in terms of utilizing the bandwidth, improving accessibility, maintaining correctness through content replication and reducing load on servers. The limitation of existing CDN load balancing algorithms is that it considers servers and the systems as non faulty which increases the probability of request been allocated at faulty server. With the increase in number of requests, the failure probability increases due to long waiting queue which increases the network load and processing time. To overcome this problem, a fault aware load balancing algorithm for CDNs is proposed that improves the QoS and reliability of the system. In this paper, the effect of network failure on QoS and reliability of the system is studied in the presence of high request rate and network traffic. Performance of existing load balancing algorithm is investigated and compared in faulty environment. Moreover, the performance of the proposed algorithm is compared with reported techniques. The experimental results demonstrate that proposed algorithm provides better robustness and resilience to fault without affecting the QoS. Further, a dynamic fault model is proposed and implemented which takes care of changing failure probability with load and provided better result as compared to static fault models.*

**Received on:** 18th-June-2015
**Revised on:** 20th-July-2015
**Accepted on:** 03rd– August-2015
**Published on:** 4th-Jan-2016

KEY WORDS

*CDN; QoS; Reliability; Load balancing; Fault rate; Network load; System load; resiliency*

***Corresponding author: Email:** kprakash91@yahoo.com, **Tel:** +91-9810292083

## INTRODUCTION

Recently, the network traffic is exploding with rapid development of internet. Therefore to provide uninterrupted services to users and maintain QoS, CDN is required. CDN is a popular solution to balance the load over a distributed system which acts as a single system for users. It is among one of the best methods to cope up with the increasing demand and an effective solution to support the load of fast growing web applications by adopting a distributed network of servers.

CDN has been widely accepted as a method for circulating large amount of content to the users by making several redundant copies of content on multiple servers. CDN can solve even high congestion issues occurred due to unexpectedly high request rate from clients. There are many issues and parameters which restricts the performance of CDN such as issue of load balancing, cost, request traffic, response time. Many proposals [1-3] have been proposed to balance load based on Cost, Response time and load on server [4 - 6]. CDN has also been designed on the basis of Energy consumption and data transfer rate [3, 7, 9]. These proposals take into consideration energy consumed by the server and data transfer rate in the server. So the primary issue that persists in CDN is load balancing of request. In this paper, we have proposed a scalable and reliable architecture for CDN along with fault aware load balancing algorithm. Although many existing approaches address the issue of load balancing in CDN but they do not take into consideration failures at servers which increased with increase in load. The proposed algorithm takes into consideration both load and failure over a server and scalability of CDNs. To summarize, the proposed algorithm tried to solve the problem of scalability and load balancing in CDN and to overcome the drawbacks of existing techniques.

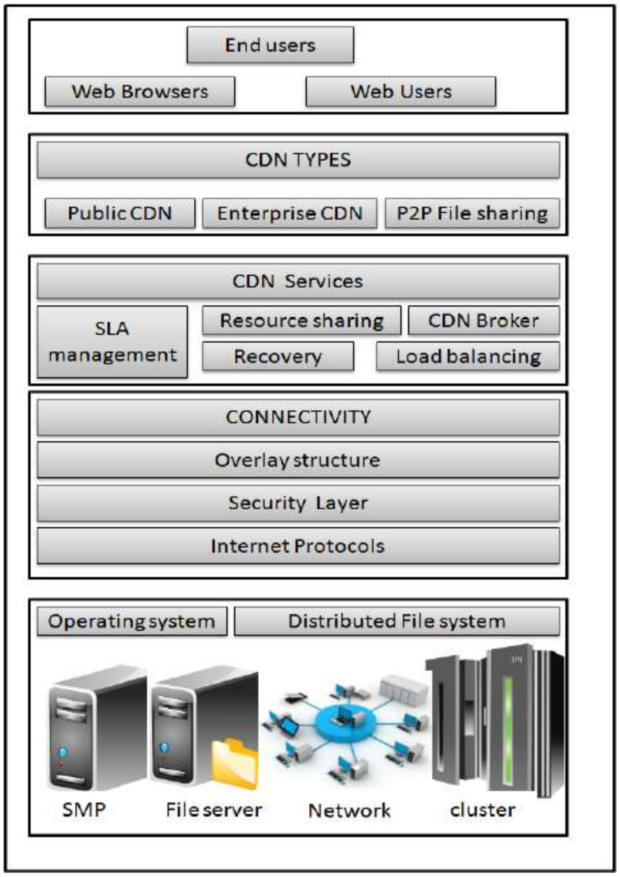


**Fig: 1. Layered Architecture**

**Figure- 1** shows the layered architecture of CDN system with application layer at the top with client end, Then the type of CDN services offered. Third layer is management layer which is responsible for resource allocation security and load balancing and all other lower layers are standard network protocols and network connectivity. Servers are the last entity which remain connected using the networks.

This paper is divided into 5 sections: Introduction, Related Work, Proposed Model, Experimental result and Conclusion. Section II discusses the survey of proposed model for load balancing and fault in CDN and their drawbacks. In section III, We present proposed architecture and load balancing algorithm. Section IV presents Simulation environment and experiment results. Section V finally summarizes the findings and future prospective of the proposed architecture.

## RELATED WORK

This section focus on the survey of existing load balancing algorithm for CDNs and cloud computing. These algorithm aims to reduce the load over the overall system and maintain the overall utilization of system. Cardellini et al.[16] proposed a survey on load balancing algorithms and classified them into two categories static and dynamic load balancing algorithms. This classification over load balancing algorithm helps in better understanding of difference between static and adaptive load balancing algorithms.

Dahlin et al.[17] proposed an least-loaded (LL) load balancing algorithm beast example for dynamic load balancing. In this requests are distributed to a server which is least loaded in term of queue length. Here requests are distributed to lead loaded server until it is completely saturated. To overcome this response time based algorithm was proposed to overcome the drawback of LL. Carter et al.[18] proposed a response time based load balancing algorithm which diverts the load to the server with fastest response time.

Manfredi et al.[19] proposed an load balancing taking care of load over the system and the capability of server to process the request. In this proposed algorithm each server is assumed to have a fixed queue size and if the queue length increases load balancing is initiated and a least loaded server with empty queue is selected to balance the load.

Javadi et al. [20] discoursed load balancing which takes care of hardware fault based on Byzantine fault, i.e. an error in the system may lead to subsequent failure in the system. On the other hand, software failure, which covers request because of resource unavailability or high queue length also leads to request failure. In distributed system, failure can be correlated with a workload using spatial and temporal correlation between workload type and intensity of failure at different servers in short interval of time. Spatial correlation refers to multiple failures occurring on different servers in short interval of time. Temporal correlation means skewness of the failure spread over time. Where correlation between failure is the time between two consecutive failure .Let Ts( Fi ), Ts( Fj) be the start time of failure i , j. Temporal correlation can be calculated as:

$$L_{i\,j} = \| F_i + F_j \| = | T_s(F_i) - T_s(F_j) | \qquad (1)$$

$$Ct(L) = 1 - \alpha \frac{L}{\theta} + \beta \left(\frac{L}{\theta}\right)^3 \qquad (2)$$

Where ϴ is an adjustable time scale parameter for determining the temporal correlation between two failure events, and α and β are positive constants where

$$\alpha = \beta + 1 \qquad (3)$$

A hybrid approach based on random and LL was proposed by Mitzenmacher et al..[15] two random choice algorithm (2RC). In this 2 servers are randomly chosen and least loaded among those is selected. This approach is beneficial is there are a large number of servers and random choice algorithm help to provide an equal probability of a server been selected.

Papagianni en al. [1] proposed similar load balancing algorithms based on cost in which a hierarchical framework is proposed which is further evaluated towards an efficient and scalable content distribution over a multi provider networked cloud environment, where inter and intra cloud communication resources are simultaneously considered along with traditional cloud computing resources. The performance of this proposed framework is accessed via simulation and modeling, while appropriate metrics are defined to associate with and reflecting the interests of different key players.

Maki en al. [3] proposed a periodic combined-content distribution mechanism to increase the gain in traffic localization. This Proposed mechanism automatically optimizes the distribution period by using how long we can expect the previous downloaded combined-content to localize traffic. A pictorial view of this model in shown in figure 2 below.

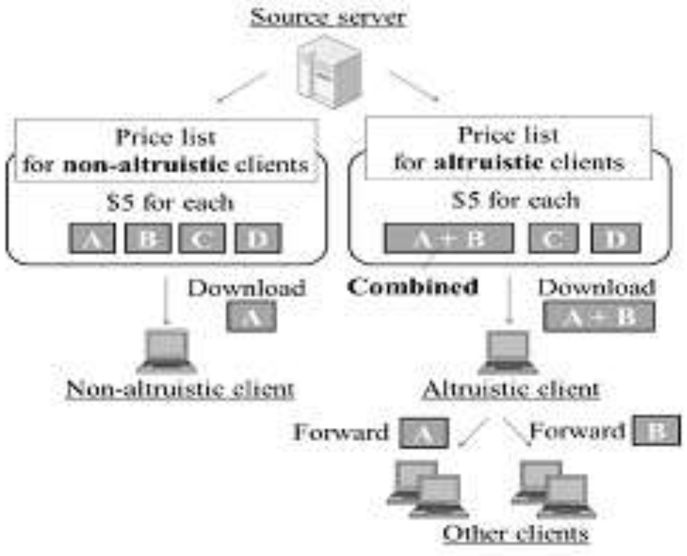


**Fig: 2. Cost based Distribution**

Mathew en al. [7] provided a new dimension to CDN proposed an energy aware load balancing algorithm which is an optimal offline and online algorithm and can be used to extract energy savings both at the level of local load balancing at the data center and global load balancing across data centers.

Mahajan en al.[19] proposed and affinity based round robin load balancing algorithm for cloud to balance the requests in cloud infrastructure . This algorithm takes into consideration the load over the data center and then if the datacenter is over loaded initiates load balancing in round robin fashion. Kansal en al.[25] have discussed an survey on various existing load balancing algorithm in distributed environment. They have identified various parameters used to de

However all the existing techniques considers the system non faulty and do not take into consideration the reliability due to request failure and failure in system. So to overcome all these issues a fault and reliability aware distributed load balancing is proposed to overcome the request failures due to faults.

## RELATED BACKGROUND

In this section we have proposed a fault model for CDN. Fluid models are being proposed for TCP flow control and in many MANET routing protocols [12-14]. CDN proposed framework consists of servers with independent queues of self-determining queue length and service rate. Our proposal uses a fluid model for dynamic queue and real time behavior of the system. We have assumed a CDN with 'n' number of servers with service queue and high rate of request traffic over the system, which cannot be fulfilled by single system and the system remains in critical situation. In such situation load balancing plays an important role to resolve the critical condition of servers by diverting the request to the server with lower request rate and empty queue which can full fill the requests. For a server with a fluid flow model we need to introduce few notations.

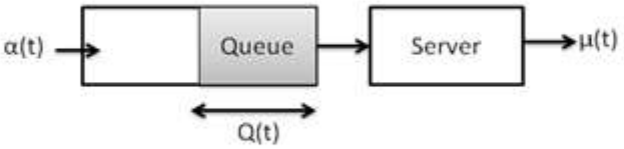


**Fig: 3. Fluid model**

Qi(t) : Queue length of server 'i' at time t.
αi (t) :Request arrival rate of server 'i'
μi (t) : Service rate of the server 'i'

In the fluid model can defined by

$$\frac{dQ_i(t)}{dt} = Q_i^{'}(t) = \alpha_i(t) - \mu_i(t) \quad (4)$$

For i=1 2….N

Where Q' (t) are the extra requests to be fulfilled .In a fluid model with an increase in request arrival rate αi (t) over a node or server queue size increases if service rate is less than the request arrival rate and server cannot handle the requests. As a result of which requests in the queue have to wait for a long time. Also with increase in queue length, load over the servers increases resulting in an increase in response time and computation time. On the other hand, all this result in higher probability of request failure. To provide best QoS (Quality of service) we need to maintain the relation between the average incoming rate and average service rate.

$$\bar{\alpha} \leq \bar{\mu} \quad (5)$$

Average incoming rate = $\bar{\alpha}$

Average service rate =$\bar{\mu}$

Next parameter we need to find is the Probability of failure of request in server over a time t. We have assumed that request rate αi (t) is distributed randomly over the time which follows a Poisson distribution.

$$p(x,\lambda) = \begin{cases} \dfrac{\Lambda^x e^{-\Lambda}}{!x} & ,x \geq 0 \\ 0 & ,x < 0 \end{cases} \qquad (6)$$

Where e is the natural logarithm and k is the possible number of occurrences of the event (positive integer values). X (The number of events in a given interval), λ (mean number of events per interval) is a positive number representing the expected number of occurrences within a specified interval. For example, if 6 requests arrive every 10 minutes, then for 1 hour λ will be 36. The Poisson distribution models the occurrence of an event without knowing the total number of possible occurrences. We have used the Poisson distribution for calculating fault rates and reliability of a system. So probability distribution for failure in a system can be given by

$$f(x,\lambda) = \frac{\Lambda^x e^{-\Lambda}}{!x} \qquad ,x \geq 0 \qquad (7)$$

Equation 7 shows the failure probability distribution over a time t, x (number of failures), $\Lambda$ (failure rate) .Where F (T) is defined as the probability of failure over the time t. To define failure in a system over a time t and t+ Δ T is given as:

$$\begin{aligned} F(t \leq T \leq t + \Delta T \mid T > t) &= \frac{\exp(-\lambda t) - \exp(\lambda(t+\Delta T))}{\exp(-\lambda t)} \\ &= 1 - \exp(-\lambda \Delta t) \end{aligned} \qquad (8)$$

$$F(t) = 1 - \exp(-\lambda t) \quad ,\text{for interval } [0,t]$$

The Reliability of a system can be defined in term of many parameters such as durability, failure and QoS over a time t .In general probability, reliability can be defined as the probability of an item to perform a required function under stated conditions for a specified period of time. In terms of failure reliability of a system can be defined as resistance to failure of an item over time. The Probability that a system is reliable over a time t can be given as:

$$\begin{aligned} R(t) &= 1 - F(t) \\ &= 1 - 1 - \exp(-\lambda t) \\ &= \exp(-\lambda t) \end{aligned} \qquad (9)$$

For interval [0, t]

And for time interval [t, t+ Δ T] reliability R (t) is given as

$$R(t) = \exp(-\lambda t) \qquad (10)$$

## PROPOSED ALGORITHM

Proposed load balancing is an improved algorithm over existing algorithm discoursed in section 2. Proposed algorithm takes into fault over a server over a period of time t as discoursed in section 3. Therefore the factors on which our algorithm is based on are Network Load, Fault Rate, Queue length, and Response Time. These parameters can be defined as:

System Load: The percentage of serve request queue filled.
Fault rate: Number of faults over a period of time.
Queue Size: Maximum size of request queue length a server can maintain and fulfill.
Response time: Time taken to start fulfilling a request.
Network load: Total bandwidth of server out of total under utilization.

Since each server is assumed to have its service rate, request rate, response time, queue size and failure rate which changes dynamically with time. For load balancing we need to find overloaded server or as we say the hot spots. A Server is said to be overloaded if:

QSize (i): queue size of server i;
Qi( t) : queue length of server i at time ‘t’ from equation 4
$\Delta$Qi : Extra Queue to be balanced

$$\text{Queue Size } (i) < Qi(t) \qquad (11)$$

The Proposed algorithm is divided into three phases:
a) Initialization
b) Load balancing
c) Updating

## a. Initialization

In this phase fitness value for a server is initialized with default values of all the parameters discoursed. All the parameters are checked and updated periodically. Initially fault rate and network load are zero, where as Queue size and response time are based on the server properties. Based on these values, fitness values are calculated. When a new server is introduced in CDN it is initialized with default values and fitness value is calculated and updated with equal intervals of time. Initial parameters are defined as:

Fault_Ini : Initial fault rate.
QSize (i) : Initial Queue length based on server.
Resp_Ini : Initial response time based on server.
N_load_Ini : Initial network load.
S_load_Ini :Initial system load.

## a. Load balancing

In this phase when the original server queue is full and no more requests can be queued, in order to save request from waiting in queue of original server and fail due to deadline because they cannot be processed. So to overcome this replica of the data being requested is made on another server to balance the request load over the original server. To balance load we require finding a server which can fulfill the request with highest fitness value and same quality of service as promised by original server. Here we can classify the servers into two categories as a hot spot and a cold spot.

Hot spots are those servers which are overloaded with requests and have most of the MIPS and network bandwidth utilized and long request waiting queue. Cold spots are those servers which have low request rate and can accommodate more requests .In other words servers with low MIPS and network bandwidth under utilization i.e. load network and processing load. Load balancing is required to stop server becoming hotspot and find a cold spot to balance the request load. Whenever a server is found over loaded based on equation 11 we need to find the server which can fulfill extra request defined as :

$$\Delta Qi = Qi(t) - QSize(i) \qquad (12)$$

$\Delta$Qi is the extra queue size to be balanced on server *i* where $i \in \{1,2,3 \ldots . n\}$ . If $\Delta$Qi is positive we will call load balancing function. To balance the load we need to find a server with empty queue length and highest fitness value from a list of all such servers maintained. This list used by load balancing algorithm along with other parameters to find the best server over which request can be diverted.

Whenever load balancing is called we need to find a best fit server based on following parameters.

1) Fault rate: It is directly propositional to the load on the server that can be network load which leads to network failure and system load which leads to system failure which is due to high request rate, increasing the queue length. If the size of the queue is too large beyond the processing rate, the requests waiting time increases which lead to request failure. On other hand system load also increases the probability of system

failure in the form of hard disk and machine failure. All the above discoursed reason leads to degradation in QoS (quality of service) provided by the server.

$$\text{ʎ}(t) = F(N\_Load, S\_load) \tag{13}$$

ʎ(t) : fault over a time t.

Above equation defines that fault rate at a particular instance of time is functionally and directly proportional to system load and network load.

ʎ: fault rate

$$\text{ʎ} = \sum \text{total number of fault / per hour;} \tag{14}$$

2) Response time: This can be defined as the time taken to start processing a request, i.e. the difference between the time request was submitted and the time server started processing the request. It is directly propositional to system load. As the CPU utilization of server increases response time increases. So the server which needs to be selected should have least average response time and can complete the request in least time.

   Resp : Response time

3) Queue length: Every server has a fixed request queue length, which can be fulfilled without request failure. So we need to select a server for load balancing which have a sufficient largest free queue size to accommodate new requests without failure.

To balance the load we need to take all the above parameters into consideration and calculate a fitness value for each server over which load can be balanced to provide better QoS and increase the reliability of the overall system by balancing the load and reducing failures.

The Fitness value for a server can be determined as:
Fval (s): Fitness value of server s

$$Fval(s) = \left(\alpha 1 * \frac{1}{\text{ʎ}}\right) + \left(\alpha 2 * \frac{1}{Resp}\right) + \left(\alpha 3 * \frac{1}{N\_load}\right) + \left(\alpha 4 * \frac{1}{S\_Load}\right) + free(Queue\_Length) \tag{15}$$

$$\alpha 1 + \alpha 2 + \alpha 3 + \alpha 4 = 1$$

$$\alpha 3 = \alpha 4$$
$$\alpha 1 > \alpha 2 > \alpha 3 > \alpha 4$$

$$server\_id= \max(fval1, fval2 \ldots, fvaln); \tag{16}$$

Load balancing is divided into 3 steps 1) Find the list of all the servers which have empty queue grater then ∆Qi is created. 2) From the created list find a server for load balancing with highest fitness value but at the same time the new server should have less or equal fault rate than the searching server to provide same or higher quality of service as promised by the original server, i.e. least fault rate, lease network load, least system load, and largest free queue length. 3) Transfer the set of extra requests to the selected server. This approach helps in maintaining skewness and increase reliability and decrease fault rate.

## b. Fitness updating

This phase includes updating the value of current network load, system load, fault rate, queue length of server. This phase is repeated after an equal interval of time to get the updated current status of the servers. Initially, all the parameters are initialized with default values in which fault rate ʎ (t) is initially zero, network load in also zero and system load is also taken as zero. The Queue length of a server in always initially zero because there is no request made to that server.

ʎ (t)_Initial = 0     \\ Initial fault rate

N_load_ini = 0  \\Initial network load.

S_load_ini = 0   \\Initial system load.

Q_len_ini = 0    \\Initial queue length

Res_Ini = Not zero      \\Initial server response time

For calculating new fitness value we need to find changes in the parameters. Let Si be the server, ʎ (t)_new, N_load_new, S_load_new, Q_len_new , Res_new are new fault rate over a time 't' ,new network load, system load, queue length and response time correspondingly. Let new fitness value be fvalt_new (Si) of server i.

$$\text{ʎ}\,(\mathrm{t}) = \mathrm{F}\left(\mathrm{N}_{\mathrm{Load}_{\mathrm{new}}}, \mathrm{S}_{\mathrm{load}_{\mathrm{new}}}\right)$$

$$\mathrm{Fval}_{\mathrm{new}}\,(s) = \left(\alpha 1 * \frac{1}{\text{ʎ}_{\mathrm{new}}}\right) + \left(\alpha 2 * \frac{1}{\mathrm{Resp}_{\mathrm{new}}}\right) + \left(\alpha 3 * \frac{1}{\mathrm{N\_Load}_{\mathrm{new}}}\right) + \left(\alpha 4 * \frac{1}{\mathrm{S\_Load}_{\mathrm{new}}}\right) + \mathrm{free}(\mathrm{Queue\_Length}) \qquad (17)$$

server_id = min (fval1new, fval2new…., fvalnnew)     (18)

$$\alpha 1 + \alpha 2 + \alpha 3 + \alpha 4 = 1$$

$$\alpha 3 = \alpha 4$$

$$\alpha 1 > \alpha 2 > \alpha 3 > \alpha 4$$

Whenever a fitness value is upgraded next request is always diverted to server with largest fitness value based on updated fitness values give in equation 17.

Load balancing Algorithm

```
Initialize servers
Start sending requests
push request in queue.
if (queue length > server queue length)
        s = find_server()        // find server with empty queue and highest fitness value
        if (( s !=  searching server ) & (fault rate < searching server))
            Migrate request  =>"s"
        else
         keep searching free server.
else
pop request from queue process it
```

```
Update fitness value algorithm

1: Find updated values of parameters
2: Find network load
3: Find system load
4: Find fitness value using equation 14
5: update the new fitness value
```

In this section we have described the performance of proposed fault aware load balancing algorithm with round robin (RR), random (Rand), least loaded (LL), two random choice (2RC) and queue length based load balancing (QLBLB) algorithms. In this for simulation GridSim API [10] is used. GridSim API basically supports scheduling and load balancing in parallel and distributed environment. Load balancing, fault in server and server request queue feature of GridSim are used to simulate CDN.

Initially GridSim do not support failure in servers. In this implementation we have introduced fault aware scheduling in GridSim to study the performance of CDN in the fault aware environment. In simulation to create a network architecture as one shown in figure 1 we have used Brite file. Brite file helps to define network properties and the interconnection between the nodes which are the servers in our case. To compare performance based on the number of faults occurring by using each of the previous algorithms and proposed algorithm. We have considered 3 servers S1, S2, S3 each of them having their independent failure rate ʎ (t), request arrival rate, processing rate and queue length. Table 1 shows the specification of each server.

**Table: 1. Servers Parameters**

| Server Name | Queue length | Fault rate | Service rate |
|---|---|---|---|
| Server1 | 20 | 0.143 | 7 |
| Server2 | 50 | 0.125 | 8 |
| Server3 | 50 | 0.5 | 2 |

Queue length defines that after the specific queue is full extra requests will be balanced using proposed algorithm, to save the request to fail due to large waiting time. **Table– 2** and **figure– 4** shows the number of requests failed when the algorithms are tested for 60,100, 200,500,600 and 700 requests count with all algorithms. Workload traces are achieved from load traces of DAS-2multi-cluster system obtained from the Parallel Workload Archive are used to generate requests [24].

**Table: 2.Request Failure Count**

| Algorithm | Request count | | | | | |
|---|---|---|---|---|---|---|
| | 60 | 100 | 200 | 500 | 600 | 700 |
| Proposed | 8 | 15 | 28 | 85 | 102 | 121 |
| QLBLB | 13 | 23 | 43 | 110 | 131 | 167 |
| RAND | 12 | 25 | 36 | 91 | 109 | 132 |
| 2RC | 10 | 19 | 36 | 98 | 117 | 142 |
| LL | 10 | 23 | 39 | 102 | 124 | 151 |
| RR | 10 | 18 | 37 | 93 | 112 | 131 |

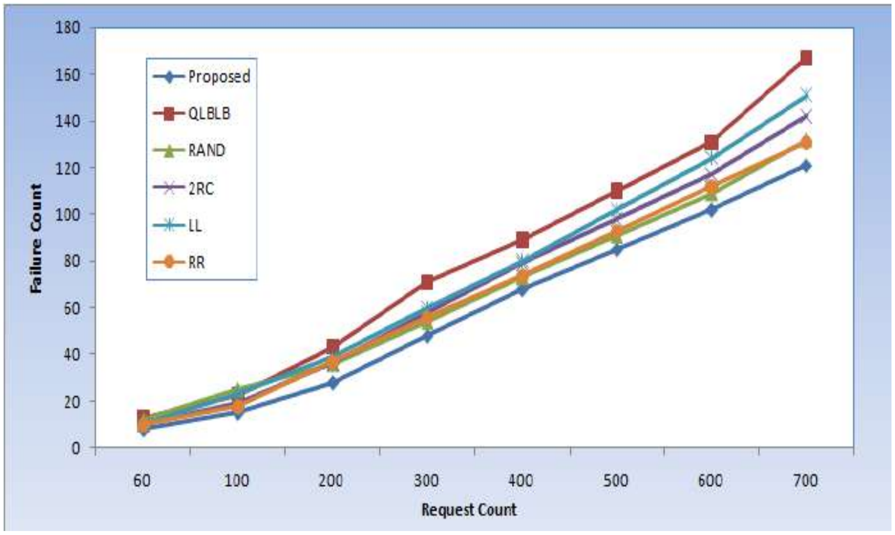


**Fig: 4. Failure count of proposed algorithm against other algorithms**

To compare performance based on probability of failure occurring by using each of the previous algorithms and proposed algorithm using scenario given in **Table– 1**.

**Table: 3. Failure probability**

| Algorithm | Request count | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 60 | 100 | 200 | 300 | 400 | 500 | 600 | 700 |
| Proposed | 0.133 | 0.15 | 0.14 | 0.16 | 0.17 | 0.17 | 0.17 | 0.172857 |
| QLBLB | 0.216 | 0.23 | 0.23 | 0.233 | 0.224 | 0.22 | 0.218333 | 0.238571 |
| RAND | 0.75 | 0.25 | 0.18 | 0.18 | 0.182 | 0.18 | 0.181667 | 0.188571 |
| 2RC | 0.166 | 0.19 | 0.18 | 0.199 | 0.197 | 0.2 | 0.195 | 0.202857 |
| LL | 0.166 | 0.23 | 0.195 | 0.2 | 0.2 | 0.204 | 0.206667 | 0.215714 |
| RR | 0.166 | 0.18 | 0.185 | 0.1866 | 0.185 | 0.186 | 0.186667 | 0.187143 |

**Table– 3 and figure–5** show the probability of request failure when each of the algorithm is tested over 60,100,200,300,400,500,600 and 700 requests. **Table– 3** shows that with increase in request failure probability increases for QLBLB, 2RC and LL. On the other hand the probability of failure is stable for RR and Rand but grater then proposed algorithm. This shows that the proposed algorithm proves to have a lower failure count and failure probability as compared to other algorithms.

**Table: 4. Reliability**

| Algorithm | Request count | | | | | |
|---|---|---|---|---|---|---|
| | 60 | 100 | 200 | 500 | 600 | 700 |
| Proposed | 0.867 | 0.85 | 0.86 | 0.83 | 0.83 | 0.827 |
| QLBLB | 0.784 | 0.77 | 0.77 | 0.78 | 0.781 | 0.761 |
| RAND | 0.25 | 0.75 | 0.82 | 0.82 | 0.818 | 0.811 |
| 2RC | 0.834 | 0.81 | 0.82 | 0.8 | 0.805 | 0.797 |
| LL | 0.834 | 0.77 | 0.805 | 0.796 | 0.793 | 0.784 |
| RR | 0.834 | 0.82 | 0.815 | 0.814 | 0.813 | 0.81 |

THE IIOAB JOURNAL
ISSN: 0976-3104

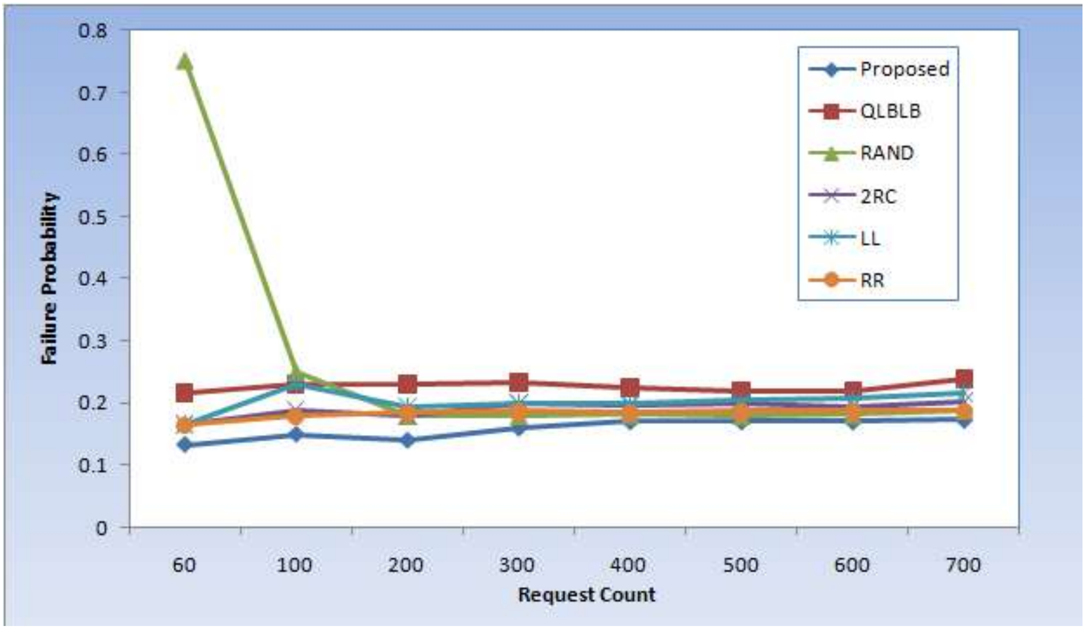


Fig: 5. Failure probability of proposed algorithm against other algorithms

The Proposed algorithm can also be compared with other algorithm based on one more parameter, i.e. reliability which is defined in equation 9. Reliability defines the algorithm to be more dependent and probability that the request will be completed. So, higher the reliability lowers the chance of request failure.

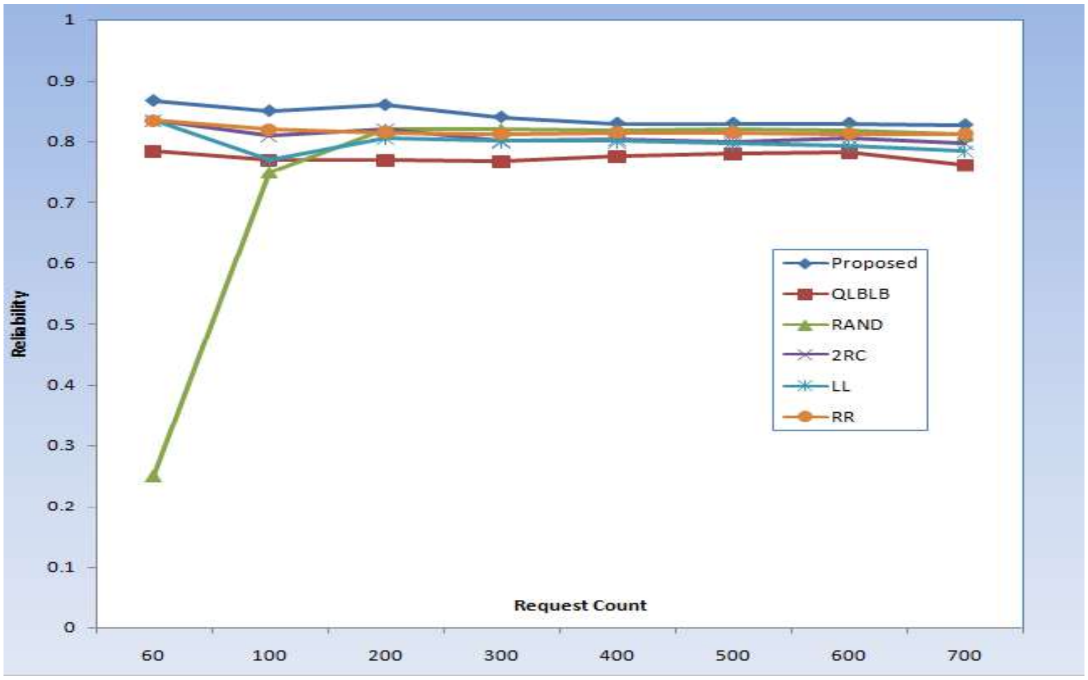


Fig: 6. Reliability of proposed algorithm against other algorithms

Table: 5.Completed request count

| Algorithms | Request count | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 60 | 100 | 200 | 300 | 400 | 500 | 600 | 700 |
| Proposed | 52 | 85 | 172 | 252 | 332 | 415 | 498 | 579 |
| QLBLB | 47 | 77 | 157 | 229 | 311 | 390 | 469 | 533 |
| RAND | 48 | 75 | 164 | 246 | 327 | 409 | 491 | 568 |
| 2RC | 50 | 81 | 164 | 242 | 321 | 402 | 483 | 558 |
| LL | 50 | 77 | 161 | 240 | 320 | 398 | 476 | 549 |
| RR | 50 | 82 | 163 | 244 | 326 | 407 | 488 | 569 |

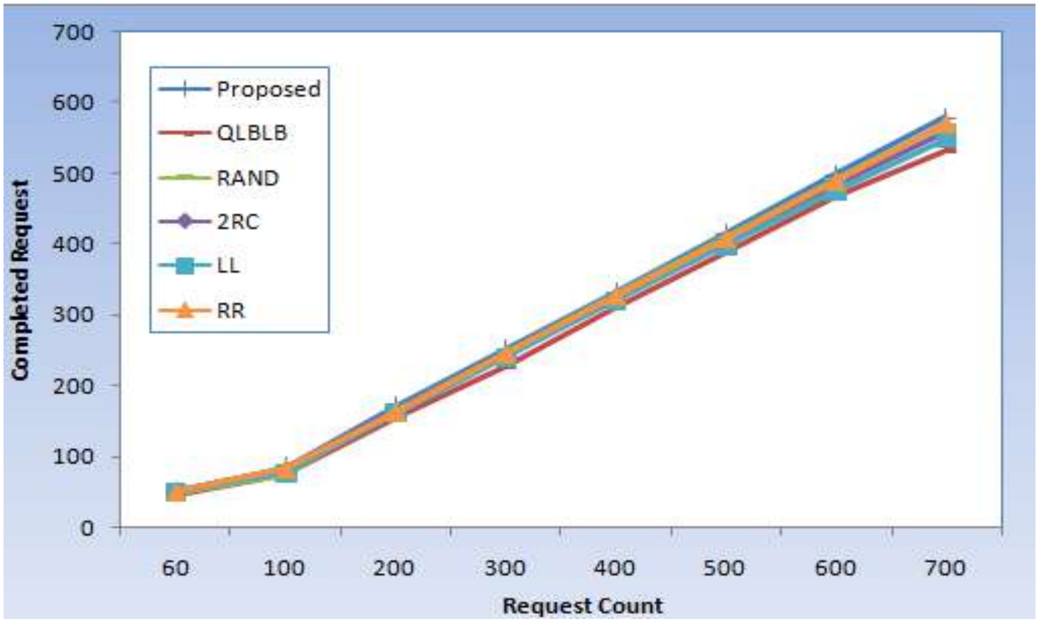


**Fig: 7. Completed request count of proposed algorithm against other algorithms**

**Table– 4 and Figure– 6** shows the increase in reliability using proposed algorithm and improvement over other algorithm. Other advantages of the proposed algorithm over other algorithms that can be derived from **Table– 3 and Table–4** is that the algorithm which has higher reliability has shown to have high request failure, on the other hand proposed algorithm have a lower request failure and higher reliability.

**Table– 5** and **Figure– 7** shows the improvement in Count of completed request using proposed algorithm over other proposed algorithm I faulty environment.

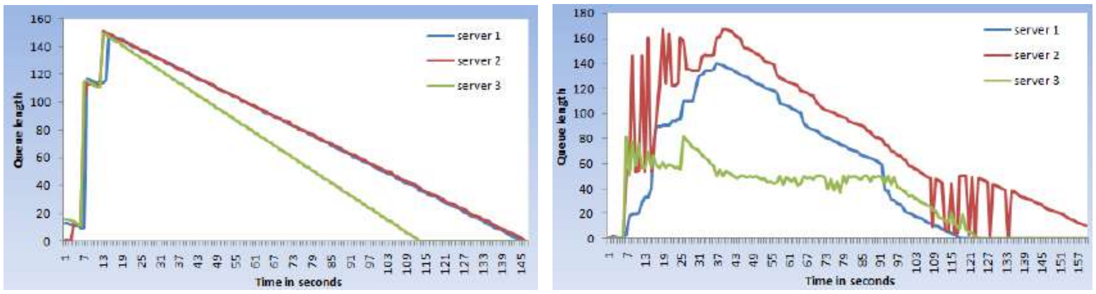
$$\text{Average Queue length} = \frac{\sum_{i=0}^{n} \mathrm{Max\_len}}{n} \qquad (20)$$

N= number of servers.

Max_length_i =Maximum queue length of server i

Proposed can also be compared based on the maximum queue length, because higher the queue size more the request waiting time. This increases the probability of request to fail over the period of time. So by comparing the maximum queue length achieved by each algorithm we can find the best algorithm.

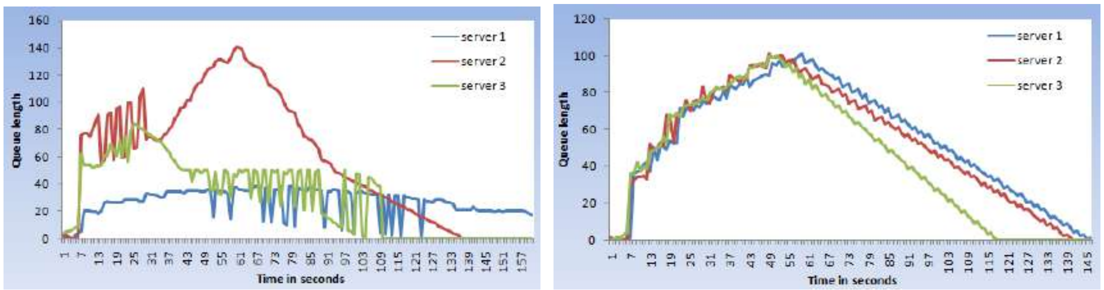
THE IIOAB JOURNAL ISSN: 0976-3104



(a) 2RC

(b) QLBLB

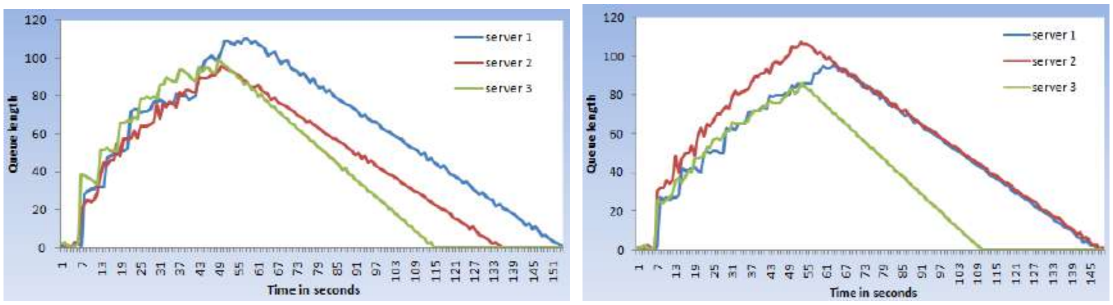
Fig: 8. Queue length of 2RC and QLBLB algorithm



(a) Fault

(b) LL

Fig: 9. Queue length of Proposed Fault and LL algorithm



(a) RAND

(b) RR

Fig: 10. Queue length of RABD and RR algorithm

Table: 6.Average Queue Length

| Algorithms | RR | LL | RAND | 2RC | QLBLB | |
|---|---|---|---|---|---|---|
| Average Queue length | 96 | 101 | 101 | 150 | 129 | 87 |
| Failure count | 97 | 95 | 96 | 94 | 97 | 93 |

**Figure– 8, 9 and 10** shows the behavior for queue length due to proposed and all other algorithms. An observation that comes out from above figures is that the algorithm which lower average queue length but has a higher fault rate. **Figure– 8 (a)** of 2RC algorithm has 150 average queue length and so on for other algorithm as shown in table 5 correspondingly. **Table– 6** clearly shows that an algorithm which has a lower average queue length, but has a higher failure count like RR algorithm, but proposed algorithm prove to have better performance in term of average queue length and failure count at the same time compare to other algorithm. Output for **table–**

6 is tested in the scenario shown in table−1 with 3 servers and corresponding failure rate, service rate and queue length. The service rate for server 1is 7 requests can be processed at the same time, similarly 6 requests for server 2 and server 3.

Table: 7. Throughput

| Algorithms | Request count | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 60 | 100 | 200 | 300 | 400 | 500 | 600 | 700 |
| Proposed | 86.66667 | 85 | 86 | 84 | 83 | 83 | 83 | 82.71429 |
| QLBLB | 78.33333 | 77 | 78.5 | 76.333333 | 77.75 | 78 | 78.16667 | 76.14286 |
| RAND | 80 | 75 | 82 | 82 | 81.75 | 81.8 | 81.83333 | 81.14286 |
| 2RC | 83.33333 | 81 | 82 | 80.666667 | 80.25 | 80.4 | 80.5 | 79.71429 |
| LL | 83.33333 | 77 | 80.5 | 80 | 80 | 79.6 | 79.33333 | 78.42857 |
| RR | 83.33333 | 82 | 81.5 | 81.333333 | 81.5 | 81.4 | 81.33333 | 81.28571 |

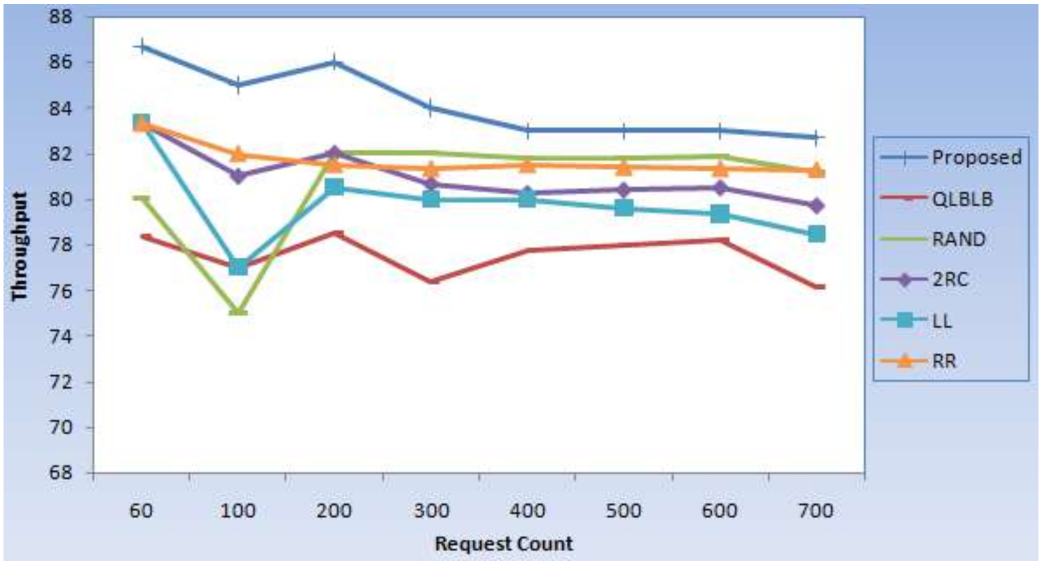


Fig: 11. Throughput Comparison of proposed algorithm against other algorithms

Table− 7 compares the throughput of proposed algorithm and other proposed algorithms for 60, 100, 200, 300, 500, 700 request over the servers. Figure− 11 compares the throughput of proposed algorithm graphically and shows the improvement of proposed algorithm over other algorithms.

Taking into consideration all the performance parameters we can suggest that fault and reliability based proposed algorithm prove to have better performance and QoS over other algorithms.

## CONCLUSION

In this paper, different types of Load balancing algorithm have been discussed with their drawbacks in CDN. To overcome the drawbacks, an efficient fault aware load balancing algorithm is proposed which performs better than other existing load balancing algorithms proposed for CDN in the fault aware environment. For future work, this algorithm may be compared with other proposals and study may be done for further improvements in the QoS.

## CONFLICT OF INTEREST

Authors declare no conflict of interest.

## ACKNOWLEDGEMENT

None

## FINANCIAL DISCLOSURE

No financial support was received to carry out this project.

## REFERENCES

[1] Akyildiz Papagianni, Chrysa, Aris Leivadeas, and Symeon Papavassiliou.[2013] A cloud-oriented content delivery network paradigm: modeling and assessment. Dependable and Secure Computing, *IEEE Transactions* 10( 5): 287−300.

[2] Leong, Derek, Tracey Ho, and Rebecca Cathey.[ 2009] Optimal content delivery with network coding." In Information Sciences and Systems, CISS 2009. 43rd Annual Conference on 2009, IEEE, 414−419.

[3] Maki, Naoya, Ryoichi Shinkuma, Tatsuya Mori, Noriaki Kamiyama, and Ryoichi Kawahara. [2013] A periodic combined-content distribution mechanism in peer-assisted content delivery networks. In ITU Kaleidoscope: Building Sustainable Communities (K-2013), *IEEE* 2013 Proceedings of, 1−8.

[4] Jiang, Xueying, Shiyao Li, and Yang Yang.[2013] Research of load balance algorithm based on resource status for streaming media transmission network. In Consumer Electronics, Communications and Networks (CECNet), 2013 3rd International Conference on, *IEEE*, 503−507.

[5] Ling Li, Ma Xiaozhen, and Huang Yulan.[2013] CDN cloud: A novel scheme for combining CDN and cloud computing. In Measurement, Information and Control (ICMIC), 2013 International Conference on, *IEEE*, 1:.687−690.

[6] Kim TaeYeon, and HoYoung Song.[2012] Hierarchical Load Balancing for Distributed Content Delivery Network. In *Advanced Communication Technology (ICACT)*, 2012 14th International Conference on, *IEEE*.810−813.

[7] Mathew Vimal, Ramesh K Sitaraman, Prashant Shenoy.[2012] Energy-aware load balancing in content delivery networks. In INFOCOM, 2012 Proceedings *IEEE*, 954−962.

[8] Maki Naoya, Takayuki Nishio, Ryoichi Shinkuma, et al.[2013] Expected traffic reduction by content-oriented incentive in peer-assisted content delivery networks. In Information Networking (ICOIN), 2013 International Conference on, *IEEE*, 450−455.

[9] Manfredi Sabato, Francesco Oliviero, and S Pietro Romano.[2012] Optimised balancing algorithm for content delivery networks. *IET communications* 6(7):733−739.

[10] Buyya Rajkumar, and Manzur Murshed. [2002] Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing. *Concurrency and computation: practice and experience* 14(13−15):1175−1220.

[11] Manfredi Sabato, Francesco Oliviero, and Simon Pietro Romano. [2013]A distributed control law for load balancing in content delivery networks. *IEEE/ACM Transactions on Networking (TON)* 21(1): 55−68.

[12] Misra Vishal, Wei-Bo Gong, and Don Towsley.[ 2000] Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED. In ACM SIGCOMM Computer Communication Review, 30(4):151−160.

[13] Hollot Christopher V, Vishal Misra, Donald Towsley, and Weibo Gong.[2002]Analysis and design of controllers for AQM routers supporting TCP flows. Automatic Control, *IEEE Transaction* 47(6): 945−959.

[14] ]V.Misra, W Gong, W boGong, and DTowsley,[2000 ] Fluid-based anal-ysis of a network of AQ Mrouter ssupporting TCP flows with an applicationtored, *Proc.ACM SIG COMM,*:151–160.

[15] Mitzenmacher, Michael.[2001] The power of two choices in randomized load balancing. Parallel and Distributed Systems, *IEEE* Transactions 12(10): 1094−1104.

[16] Cardellini Valeria, Emiliano Casalicchio, Michele Colajanni, and Philip S Yu. [2002]The state of the art in locally distributed Web-server systems. *ACM Computing Surveys* (CSUR) 34(2): 263−311.

[17] Dahlin, Michael.[2000] Interpreting stale load information. Parallel and Distributed Systems, *IEEE Transactions* ,11(10):1033−1047.

[18] Carter Robert L, Mark E Crovella.[1997] Server selection using dynamic path characterization in wide-area networks. In INFOCOM'97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution., *Proceedings IEEE*, 3: 1014−1021.

[19] Mahajan Komal, Ansuyia Makroo, and Deepak Dahiya. [2013] Round Robin with server affinity: a VM load balancing algorithm for cloud based infrastructure. *Journal of information processing* system,s 9(3): 379−394.

[20] Javadi, Bahman, Jemal Abawajy, and Rajkumar Buyya.[2012] Failure-aware resource provisioning for hybrid Cloud infrastructure.*Journal of parallel and distributed computing,* 72(10): 1318−1331.

[21] Fu, Song, and Cheng-Zhong Xu. [2010] Quantifying event correlations for proactive failure management in networked computing systems. *Journal of parallel and distributed computing,* 70(11):1100−1109.

*[22]* Gallet Matthieu, Nezih Yigitbasi, Bahman Javadi, Derrick Kondo, Alexandru Iosup, and Dick Epema.[2010] A model for space-correlated failures in large-scale distributed systems. In Euro-Par 2010-Parallel Processing, 88−100. *Springer Berlin Heidelberg*

[23] Yigitbasi Nezih, Matthieu Gallet, Derrick Kondo, Alexandru Iosup, Dick Epema.[2010] Analysis and modeling of time-correlated failures in large-scale distributed systems. In Grid Computing (GRID), 2010 11th *IEEE/ACM International Conference,* 65−72.

[24] ParallelWorkloadArchive. http://www.cs.huji.ac.il/labs/parallel/workload/.

[25] Kansal, Nidhi Jain, and Inderveer Chana.[2012] Existing load balancing techniques in cloud computing: a systematic review. *Journal of Information Systems and Communication* ,3(1): 87−91.

## ABOUT AUTHORS

***Prakash Kumar*** *has received his B. Tech. in Electronics and Communication Engineering, and M. Tech in Computer Science and Technology from University of Roorkee (Now Indian Institute of Technology, Roorkee), India. He is currently an Assistant Professor (Senior Grade) in Jaypee Institute of Information Technology,(Deemed University), Noida, India. His area of interest is in Computer Networks and Communications, Distributed Computing, Cloud Computing and Virtualization. He is currently pursuing his Ph D in the field of virtualization of resources viz. Systems and network resources. His main focus is on Trust, Reliability and Fault Tolerant networks and systems for distributed and cloud environments.*

***Prof. Krishna Gopal*** *is currently Dean Academic and Research at JIIT, Noida, India since 2011. He is Ph. D. from REC KurukshetraKurukshetra University Kurukshetra, India. He is having 45 years of teaching and R&D experience. He received his Bachelor, Master and PhD in Electronics engineering from IIT, Madras, RECKurukshetra in 1966, 1972, 1979 respectively. He published more than 100 papers in different journals, conferences, patents etc. He is member of various professional bodies like: Life Member System Society of India, Indian Society for Technical Education, senior member of IEEE etc. His area of interest is Reliability and Fault Tolerant Networks and communication Systems.*

***Prof. J P Gupta*** *has received his Ph D degree from University of Westminster, UK. He is currently the Director Emeritus (QA) at Hydrocarbons Education and Research Society, New Delhi, India. He is an academician having more than 35 years experience including Professor at IIT Roorkee, India, Vice Chancellor at JIIT Noida,India, Galgotia University, and Sharda University, India. He is author of more than 70 research papers published in International journals and conferences. He has received Commonwealth Fellowship and many more awards and memberships to his credit.*

ISSN: 0976-3104
SPECIAL ISSUE

ARTICLE OPEN ACCESS

# NLP BASED HYBRID KNOWLEDGE PROVIDER SYSTEM FOR TEXT AND IMAGE DATA EXTRACTION FROM A USER QUERY

**Prasenjit Mukherjee and Baisakhi Chakraborty***

*National Institute of Technology, Durgapur, INDIA*

## ABSTRACT

*Syntactic analysis (Parsing) is a main method of analyzing a sentence in natural language. There are several techniques of parsing. This paper proposes a Hybrid Knowledge Provider System (HKPS) where permutation-combination (PC) based parsing technique and Grammatical Rules (GR) based parsing technique have been applied on a single system. HKPS is an automated system that shall be able to extract text and image based knowledge data from database.*

**Received on:** 06th-May-2015
**Revised on:** 17th-July-2015
**Accepted on:** 21st–Aug-2015
**Published on:** 4th-Jan-2016

KEY WORDS

*Knowledge Provider; Natural language query; Query-Response model; E-R diagram; HKPS*

***Corresponding author: Email:** baisakhichak@yahoo.co.in, **Tel:** +91-9434788065 **Fax:** +91-343-2546406

## INTRODUCTION

Computer linguistic refers to analysis of sentence formation in natural language by the computer. The process of analysis of sentence formation is called parsing. There are several techniques of parsing. The importance of Syntactic parsing is mediating between linguistic expressions and their meanings. Many works have been done on usefulness of syntactic representations for subsequent tasks such as relation extraction, semantic role labeling and paraphrase detection as in [1]. We have proposed two techniques of parsing on a Knowledge Management System (KMS) termed as Knowledge Provider System (KPS) in this paper which are, Permutation-Combination (PC) based parsing and Grammatical Rules (GR) based parsing. In the PC based parsing system, KPS reads natural language query and creates a conceptual form of database using the Permutation-Combination (PC) technique. The semi-automated system follows the client-server architecture to handle the queries. Some individuals termed Knowledge Workers (KWs) are associated with the KPS. KPS has a default database with certain queries and responses stored in it. If KPS is not able to generate the response of a NLP query or if data is not present in the database then situation is handled by the KWs. The PC based KPS has been proposed by the authors in [2]. GR based parsing technique is another approach in Knowledge Provider System (KPS) to create the conceptual form of database. GR based KPS follows English grammar rules. KWs are not associated with GR based AKPS. The system itself generates response to the client side. The natural language queries may be in assertive or interrogative sentences. GR based AKPS refers the grammar rules at runtime to extract nouns as an entities and verb as a relationship. The system uses these entities and initializes them in to the semantic table for creating conceptual form of database. Each response is generated from the default database. The default database needs to be updated from reliable sources. The GR based AKPS has been proposed and discussed in [3]. This paper proposes a Hybrid KPS (HKPS) where PC based parsing technique and GR based parsing technique have been applied in a single system. It is an automated system that able to extract text and image based knowledge data from database.

## PREVIOUS RELATED WORKS

There are many natural language processing based models. A study of several of them has revealed a wide variety of application based models. A common sense filter system has been evaluated in [4] for the ReVerb Open IE system, applied as a method for answer validation in a Question Answering task applicable to a large database of facts. This system queries a database to find the presence of arbitrary facts. A model of NLP considering the problem of learning commonsense knowledge has been discussed in [5]. The commonsense knowledge is in the form of first-order rules and noisy natural-language extractions. The noisy natural-language extractions are produced by an off-the-shelf information extraction system as in [5]. In complex inference problems involving long, complicated formulae, the Markov logic is used to integrate logical and distributional information in natural-language semantics results. The system proposed a new inference algorithm based on Sample Search. The algorithm computes probabilities of complete formulae rather than ground atoms in [6]. Generating sentences from images is an idea of combining visual and linguistic information that has been gaining traction in the Computer Vision and Natural Language Processing communities over the past several years. The motivation of the model for a combined system is to generate richer linguistic descriptions of images and investigate the performance of several integrated information from language and vision systems as in [7].

Script signifies the sequence of knowledge of stereotypical event and it can aid text understanding. The Initial statistical methods have been developed in this model to learn probabilistic scripts from raw text corpora in [8]. The distributional models do not succeed to distinguish between semantic relations and the distributional models cannot be a valid model of conceptual representation. Then the approach is to use the Distributional Inclusion Hypothesis, which states that hyponyms tend to occur in a superset of contexts in which their hyponyms are found and thus propose a robust supervised approach that achieves accuracies of .84 and .85 on two accessible datasets as in [9]. An interactive text to 3D scene generation system which learns the expected spatial layout of objects from data has been discussed in [10]. As per the system, a user provides input in natural language text from which the system can extract explicit constraints on the objects and it should appear in the scene as in [10]. A natural language model has been discussed in [11] that integrate NLP with computer vision. The model has proposed a strategy for generating textual descriptions of videos and has used a factor graph to combine visual detections with language statistics which has improved recognition and description of entities in real-world videos [11]. TOKENSREGEX is a NLP based system which follows a framework for defining cascaded regular expressions over token sequences. TOKENSREGEX is a part of the Stanford CoreNLP software package and it is used for various tasks which require reasoning over tokenized text as in [12]. Two structured prediction models for joint parsing and multiword expression identification have been developed and proposed in [13] wherein earlier, syntactic analysis and multiword expression identification had been proposed as alternative methods of NLP. The experiments on prediction models in [13] show that both models can identify multiword expressions with much higher accuracy than a state-of-the art system based on word co-occurrence statistics.

This paper proposes an application of PC based Parsing technique and Grammatical Rules based Parsing technique in to a single system. Section 2 discusses on previous related works, section 3 stated the principle of the architecture where PC based Parsing algorithm is associated into GR based Parsing technique and generates text based and image based responses by the algorithm. Section 4 describes Algorithm and its Rules, Section 5 describes Methodology and Section 6 winds up with conclusion and future works.

## THE PRINCIPLE OF HYBRID KNOWLEDGE PROVIDER SYSTEM (HKPS)

Knowledge Management System is domain specific and needs an organized knowledge database for extraction of knowledge. The KMS is meant to work for e-service oriented organizations providing regular services to clients. The authors have proposed a PC based KPS model in [2] and GR based AKPS model in [3] to handle Natural language client queries. Both systems follow the request-response model and extract knowledge data from knowledge database. Both of the systems use queries in natural language which may be assertive or interrogative. The role of the KWs in the KPS has been discussed in [2]. PC based KPS may not able to manage complex queries whereas GR based AKPS system may able to manage these complex queries where complexity refers to handling of auxiliary verb as main verb. The parts of speech table (POS) has been used by the KPS to identify the unique words from the query sentences. KPS applies the PC based parsing technique to select the right combination of two entities and relationship of these entities generating a conceptual form of database which creates the sql-query for the extraction of data from the default database as a response. If records are not present into the database, then Knowledge Workers (KWs) handles the requested query and updates the database. The parsing technique of KPS has been described greatly in the paper [2].

### Process of PC based Parsing Algorithm

i.User posts Query in Natural Language.
ii. Query is checked for validity. If valid then go for further Processing, if not, then prompt again.

iii. The Query in Natural Language is tokenized. Query (String) = {S1, S2, S3, S4,………… Sn.} Where, S1, S2 are tokens.

iv. Each token is checked with English Grammar, placed in a Grammar Table (except for unique Tokens) and removed from the Token List after its placement in the grammar Table.

```
For i = 0……n
{
If( Si== wh[])
             // Insert s1 into the grammar table in its position.
                   Else
If(Si == Pre[])
            // Insert S3 into the grammar table in its position.
                   Else
If(Si == Aux[])
// Insert S3 into the grammar table in its position.
.
.
If(Sn== Con[])
// Insert S3 into the grammar table in its position.
}
```

v.Check again for all unique Tokens (Words) and if the Unique Token matches with the content of Grammar table, then it is removed from the Token List. The Algorithm is applied to the rest of the Token List in a similar manner.
If unique Token does not match with the grammar table, then apply the Algorithm on Token List.
vi. After applying the Algorithm will get the Conceptual form of Database (E-R representation) from the Natural Language Query.
vii. If Database and Tables already created response the Query else create the Database and Tables and send query to the Knowledge Workers (KWs). Knowledge Workers will update the Database from where the response will be generated.
viii. Resultant Database stores the response from where the user will get the Query Response.

### Architecture of Hybrid Knowledge Provider System

The proposed HKPS model is based on the client server architecture where requests are sent to the HKPS which generates corresponding responses. The modular architecture of HKPS is shown in **Figure– 1**. Clients access the HKPS through a user interface where queries are posted in natural language. The Parsing Module runs two processes that generate the conceptual form of database from the user's queries in natural language. The database module defines three types of Databases, namely, Temporary database, Default database and Result database. The Temporary database stores the unmanaged queries where request data is not present in Default database. Default database is a main database from where the response is generated to the client side. The Result database temporarily stores the response for the clients. The Context diagram in **Figure– 2** indicates the working principle of HKPS which is self explanatory.

## ALGORITHM AND ITS RULES

In The HKPS reads the queries in natural language. The query sentence may be of assertive or interrogative. The HKPS creates rules at runtime using the phrases of assertive and interrogative sentences.

THE IIOAB JOURNAL ISSN: 0976-3104

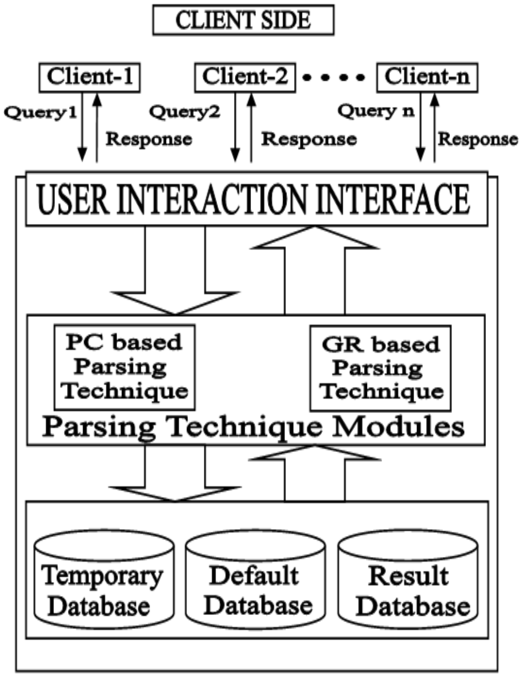


Fig: 1. Architecture of HKPS

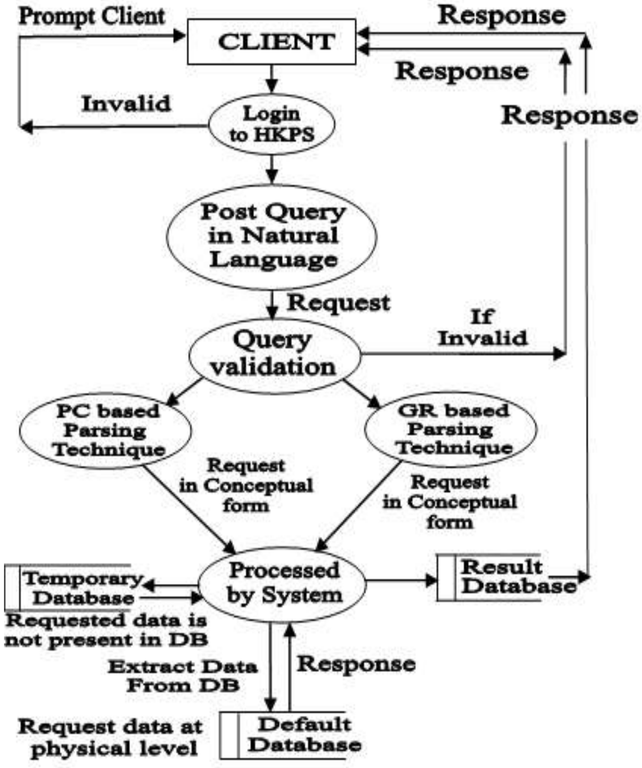

Prompt Client
CLIENT
Response
Response
Response
Invalid
Login to HKPS
Post Query in Natural Language
Request
Query validation
If Invalid
PC based Parsing Technique
GR based Parsing Technique
Request in Conceptual form
Request in Conceptual form
Temporary Database
Processed by System
Result Database
Requested data is not present in DB
Extract Data From DB
Response
Request data at physical level
Default Database

Fig: 2. Context diagram of HKPS

## Assertive Sentence formation

Assertive Sentence - Noun phrase + Verb phrase + Complement.
The formation of rules of Assertive Sentence at runtime of the HKPS system will be
"Noun + Verb + Noun", "Determiners + Noun + Aux + Adverb + Verb + Preposition + Noun "or
"Adjective + Noun + Preposition + Verb + Preposition + Determiners + Noun".

## Interrogative Sentence formation

The syntactic treatment of Interrogative sentence is different from Assertive sentence in English language. In a sentence, the Noun, Verb, Adjective, Adverb represents unique word(s) and verb does not have any fixed position. Verb may appear before Noun or after Noun. HKPS system cannot maintain any database for these unique words identification from a sentence. HKPS utilizes the phrases of interrogative sentence to create rules at runtime.
Interrogative Sentence - Wh Phrase (attached Auxiliary Verb) + Noun Phrase +Verb Phrase + Complement.
Wh phrase contain the Which, Who, What, etc. with am, are, is, was, were etc.
The Rules created at runtime by the HKPS are-
"Wh + Aux + Determiners + Noun + Verb + Preposition + Noun" or
"Wh + Determiners + Noun + Verb + Preposition + Determiners + Noun + Noun".
The HKPS system may able to increase the rules when phrases of interrogative sentence are increased. Rules may be modified and updated from time to time providing an advantage to the system.

This paper combines PC based parsing technique of KPS proposed and discussed in [2] with GR based parsing technique of AKPS as discussed in [3] to generate the HKPS. This system is fully automated and Knowledge Workers are not involved or associated as had been in PC based KPS discussed in [2]. The HPKS system has ability to retrieve text data as well as image data. In this application the image retrieval algorithm has been used. The HKPS model is domain specific and follows the query response model to extract knowledge from default database. The default database stores the knowledge data either in text mode or image mode. The HKPS generates response from this default database. If the HKPS fails to generate response for any query then it will be stored in a temporary database. Whenever the default database is updated from reliable sources like Database Administrator, System Administrator or any other Administrator, system will generate the response of the pending request. Two effective parsing algorithms work in tandem in HKPS to handle user's request and generate response after selecting the appropriate parsing technique. The selection process of parsing method has been done on the basis of number of unique tokens. If number of unique tokens equal to three (3) then PC based parsing algorithm will be applied otherwise GR based parsing algorithm will be applied but number of unique tokens should be greater than one (1) and less than three (3) or greater than three (3).

## Process

i. User posts Query in Natural Language.
ii. Query is checked for validity. If valid then go for further Processing, if not, then prompt again.
iii. The Query in Natural Language is tokenized. Query (String) = {S1, S2, S3, S4,………… Sn.} Where, S1, S2 are tokens.
iv. Now apply the Algorithm on Token List.

## Algorithm

1. Read Input Statement S.
2. Each token is checked with English Grammar, and identify that how many unique tokens are there in the query sentence.
3. IF (UniqueTokens == 3) THEN apply the KPS algorithm
3.1. The Query in Natural Language is tokenized. Query (String) = {S1, S2, S3, S4,………… Sn.} Where, S1, S2 are tokens.
3.2. Each token is checked with English Grammar, placed in a Grammar Table (except for unique Tokens) and removed from the Token List after its placement in the grammar Table.
3.3. Check again for all unique Tokens (Words) and if the Unique Token matches with the content of Grammar table, then it is removed from the Token List. The Algorithm is applied to the rest of the Token List in a similar manner.
3.4. If unique Token does not match with the grammar table, then apply the permutation function on the Token List.

```
3.5. After permutation function being applied, the algorithm will show the noun-verb-noun combination insert them into the semantic table.
4. ELSE
IF ((UniqueTokens >3) || (UniqueTokens < 3 && UniqueTokens >1)) THEN apply the AKPS Algorithm
4.1.    Split Input Statement S into the String Array called STR[n].
4.2.    First check the STR[0] equal to the words- "who","which","what"….etc.
IF STR[0]== "who"|| IF STR[0]== "which"|| IF STR[0]== "what" ….. THEN
CALL Interrogative_Sentence(STR[n]) method
ELSE
CALL Assertive_Sentence(STR[n]) method
END IF
5.      Interrogative_Sentence(STR[n]) method:
5.1     Create the each Rule and store it on Rules Array.

5.2.    Check the STR. length with the each rule length from Rules[n]. If the rule match from Rules[n] then send the selected rule and STR[n] to Parsing method which will either return 0 or 1 to a variable.If variable value is 1 then  Insert STR[n] in Sentence Table in Database and Insert selected Rules in Rules Table in Database.

6.      Assertive_Sentence(STR[n]) method

6.1.    Create the each Rule and store it on Rules Array.
6.2.    Check the STR. length with the each rule length from Rules[n]. If the rule match from Rules[n]
then send the selected rule and STR[n] to Parsing method which will either return 0 or 1 to a variable. If variable value is 1 then  Insert STR[n] in Sentence Table in Database and Insert selected Rules in Rules Table in Database.
7.   int Parsing(string Rules, string STR)
7.1.  Declare Parts of Speech List: WH[n]={"what","which"…}
                                   AUX[n]={"is","am","are"…}
                                   PRE[n]={"of","to","in","for"…}
                                   DET[n]={"a","an","the"…}
                                   PRO[n]={"I","you",he"….}
                                   UNK[n]={"Noun","Adjective","Verb"….}
7.2. split Rules to R[n] and STR to S[n].
     Declare Result[0....n] = NULL. Check each word from S[n] with each word from WH[n],  AUX[n], DET[n],PRO[n] and UNK[n] and store result in Result Array.
7.3. Initialize the variable value=0
FOR i=0,j=0, TO Result.length -1 AND R.length-1 STEP 1
IF Result[i] != R[j] THEN
Value INCREMENTED BY 1
END IF
END FOR
7.4. If Value > 0 then the parsing method will return 0 else 1.
8.  Select Row one by one from Rules Table and Sentence Table in Database.
8.1. Each Row of Rules Table and Sentence Table initialize  in SS1 [n] and SS2 [n].
8.2. Each word from SS1 array inserts it into the grammar table as per SS2 array.
 9. From Grammar Table select the values from Noun, Verb columns and properly initialize in Semantic Table.
10. Select values from semantic table and check each word in the synonym database where the actual word is stored with the synonyms.
10.1. Initialize the Entity1, Entity2, Relationship values from Semantic Table to string variables entity1, entity2 and verb.
10.2. Value of entity1 and entity2 variables matches with synonyms words in Relation, Attributes Table and value of verb variable matches with synonyms words in Relationship Table.
10.3. If value matched then reinitialize the values from Tables to entity1,entity2 and verb variables.
10.4. Initialize the Filter Table through the values of entity1, entity2 and verb variables.
10.5. In next step the system further processes the Filter table and create the SQL Query from the Filter table using the database conceptual form.
10.6. The SQL query fetches the knowledge data from the default database as a response and store it into the result database. If the knowledge data not present in the default database then User/Clients query will be stored in temporary database termed as pending request. When default database will update as per the pending request then System will generate the response to the User/Clients.
```

11. The knowledge data can be the text data or image data. If the data is an image into the database then the system runs this algorithm below to retrieve the images from the default knowledge database.

11.1. Declare the byte array to store image bytes

```
byte[] photo_array;
```

11.2. SQL-query fetch the text as well as image data from the default database and stored into a dataset call ds.

11.3. In the next step initialize the another bytes array through the dataset variable ds.

```
byte[] bytes = new byte[ds.Tables[0].Rows[0][3].ToString().Length * sizeof(char)];
```

11.4. Copies a specified number of bytes from a source array to destination array.

```
System.Buffer.BlockCopy(ds.Tables[0].Rows[0][3].ToString().ToCharArray(), 0, bytes, 0, bytes.Length);
```

11.5. Initialize the data of bytes array variable to photo array variable.

```
photo_array = bytes;
```

11.6. To represents the bytes data from the variable photo_array, the memory stream instance created.

```
MemoryStream ms = new MemoryStream(photo_aray);
```

11.7. Initialize instance of memory stream to Picture Box control or Data Grid control to show the image like-

```
        pictureBox1.Image = Image.FromStream(ms);
        pictureBox1.SizeMode = Picture Box Size Mode. Stretch Image;
```

## METHODOLOGY AND TOOLS

The HKPS has been designed using C# at front end and MS-Access at back end as the designed tools. The HKPS has three events. The First event is User Interface event, second is PC based parsing process event, third is Grammatical Rule based parsing process event. Sequence of these events is as follows.

### User Interface event

i. The User will Login to the HKPS and post a Query string *"University has opened many Departments"* in natural language into the HKPS.

ii. After clicking the "QUERY SUBMIT" Button, the Query in natural language will be tokenized. Each token will be checked by the Parts of Speech Table to identify the unique tokens. If the number of unique tokens is equal to 3 (three) then PC based parsing process event will be called else, if the number of unique tokens is greater than 3(three) then GR based Parsing process event will be called. GR based Parsing process event can be called if number of unique tokens is less than 3(three) but greater than 2(two).

### PC based parsing process event

i. The unique tokens will be extracted from the Query string in natural language. The tokenized query string will be checked by The Parts of Speech Table. Each word is a token and tokens corresponding to the words "has" and "many" in the Parts of Speech Table will be removed. Only three unique tokens "University", "opened" and "Departments" will remain for further processing.

ii. The remaining tokens will be permuted and there will be many combinations from which the system will select the right combination. As per the combinations, the tokens will be inserted into the **Table– 1** (Semantic Table) of the database of the HKPS.

**Table: 1. Semantic Table in MS Access Database**

| Entity1 | Relationship | Entity2 |
|---|---|---|
| University | opened | Departments |

iii. The Conceptual Form (E-R Diagram) of Database is created from the semantic table number **Table– 1** where Entity1, Entity2 are the Relations of the Database and Relationship constructs the connection between two Relations.

### Grammatical Rules Based parsing process event

i. Assume the Query string from the user is *"What are the Computer courses offered by the Burdwan University".*

ii. Split the string into tokens which correspond to each word of Query sentence. Check the first token if "Wh" word or not. If true, then "Interrogative Sentence" Method will be invoked.

iii. Create each Rule at runtime and store it on Rules Array in Interrogative Sentence Method. Compare Query string length with the each rule length from Rules Array and Insert each token in **Table– 2** (Sentence Table in Database) and Insert each word from matched rule in **Table– 3** (Rules Table in Database).

Table: 2. Sentence Table in MS Access Database

| S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | S10 |
|---|---|---|---|---|---|---|---|---|---|
| What | are | the | Computer | courses | offered | by | the | Burdwan | University |

Table: 3. Rules Table in MS Access Database

| W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 |
|---|---|---|---|---|---|---|---|---|---|
| Wh | Aux | Det. | Noun | Noun | Verb | Prep. | Det. | Noun | Noun |

iv**.** Parsing Method will be called and as per the algorithm and the method would return either 0 or 1.
v. Apply step 8 of the algorithm after selecting row from **Table– 2** (Sentence table) and **Table– 3** (Rules Table) in Database. After completion of step 8 it creates two arrays that will be inserted into **Table– 4** (The Grammar Table).

Table: 4. Grammar Table in MS Access Database

| Wh | Det. | Noun | Pronoun | Aux | Adj | Adv | Verb | Preposition |
|---|---|---|---|---|---|---|---|---|
| What | the<br>the | Computer<br>courses<br>Burdwan<br>University | | are | | | offered | by |

vi. Select the Noun, Aux, Verb, and Preposition from **Table– 4** (Grammar Table) and do the same that is mentioned in STEP 9 of the main algorithm. A Semantic Table is formed with entities and their relationship as shown in **Table– 5**.

Table: 5. Semantic Table in MS Access Database.

| Entity1 | Relationship | Entity2 |
|---|---|---|
| Burdwan University | Offered | Computer courses |

vii. Using Entity1, Entity2 and Relationship values from **Table– 5** (Semantic Table) will be produced the conceptual form (E-R Diagram) of Database.
viii. The system will construct the SQL Query using Semantic Table. After creating the SQL Query, the system will run this query and generate response to the client side.

## CONCLUSION AND FUTURE WORKS

This The HKPS model works on the basis of two main processes, the PC based parsing process and GR based parsing process. Any one of the processes may be chosen depending on the nature of the client query. The model is domain specific. The HKPS is modeled to be used for e-government services. The e- government services may handle several different domains and there might be wide variety of queries. Since this HKPS is dedicated to one domain, future work lies on developing e-government service models to cater simultaneously to different services pertaining to multiple domains on same platform.

## CONFLICT OF INTEREST

Authors declare no conflict of interest.

## ACKNOWLEDGEMENT

This Research work has been done at Project Research Lab, Dept. of Information Technology, National Institute of Technology, Durgapur, under Visvesvaraya PhD Scheme, Deity, Govt. of India.

## FINANCIAL DISCLOSURE

Financial support was received from Visvesvaraya PhD Scheme, Deity, Govt. of India (Order Number: PHD-MLA/4(29)/2014-2015 Dated- 27/4/2015) to carry out this research work.

## REFERENCES

[1] Berant Socher Richard, Bauer John, Manning Christopher D. and Ng Andrew Y. [2013] Parsing With Compositional Vector Grammars. *In Proceedings of the Association for Computational Linguistics (ACL)*.

[2] Mukherjee P, and Chakraborty B. Knowledge Provider System with natural language data modeling. *In Proceedings of ISCA 29th. International Conference on Computers and their Applications, Las Vegas, USA*: 229−236.

[3] Mukherjee P, Chakraborty B, Debnath CH Narayan. [2015] A Natural Language Processing based Automated Knowledge Provider System with Speech Recognition. Accepted in *28th International Conference on Computer Applications in Industry and Engineering*, Hilton San Diego, San Diego, California, USA.

[4] Angeli Gabor, Manning Christopher. [2013] Philosophers are Mortal: Inferring the Truth of Unseen Facts. *In Proceedings of Computational Natural Language Learning (CoNLL).*

[5] Raghavan Sindhu, Mooney Raymond J. [2013] Online Inference-Rule Learning from Natural-Language Extractions. *In Proceedings of the 3rd Statistical Relational AI (StaRAI-13) workshop at AAAI.*

[6] Beltagy Islam, Mooney Raymond J. [2014] Efficient Markov Logic Inference for Natural Language Semantics. *In Proceedings of the Fourth International Workshop on Statistical Relational AI at AAAI (StarAI-2014), Quebec City, Canada*: 9−14.

[7] MacKenzie Calvin. [2014] Integrating Visual and Linguistic Information to Describe Properties of Objects. *Undergraduate Honors Thesis, Computer Science Department, University of Texas at Austin*.

[8] Pichotta Karl, Mooney Raymond J. [2014] Statistical Script Learning with Multi-Argument Events. *In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014), Gothenburg, Sweden*: 220-229.

[9] Roller Stephen, Erk Katrin and Boleda Gemma. [2014] Inclusive yet Selective: Supervised Distributional Hypernym Detection. *In Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014), Dublin, Ireland*: 1025−1036.

[10] Chang Angel X. , Savva Manolis , Manning Christopher D. [2014] Interactive Learning of Spatial Knowledge for Text to 3D Scene Generation. *In Proceedings of the Association for Computational Linguistics (ACL, 2014), Workshop on Interactive Language Learning, Visualization, and Interfaces (ACL-ILLVI)*.

[11] Thomason Jesse, Venugopalan Subhashini, Guadarrama Sergio, Saenko Kate, Mooney Raymond. [2014] Integrating Language and Vision to Generate Natural Language Descriptions of Videos in the Wild. *In Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014), Dublin, Ireland*.

[12] Chang Angel X. , Manning Christopher D. [2014] TokensRegex: Defining cascaded regular expressions over tokens. *Department of Computer Science, Stanford University, CSTR 2014−02*.

[13] Green Spence , Marneffe Marie-Catherine de , Manning Christopher D. [2013] Parsing Models for Identifying Multiword Expressions. *In the Computational Linguistics journal* 39(1):195−227.

## ABOUT AUTHORS

***Prasenjit Mukherjee** received the M.Sc. degree in Information Technology In 2005 from Annamalai University, Tamil Nadu, India. In 2011, he received M. Tech in Information Technology at Karnataka State Open University (KSOU), Karnataka, India. Presently, a fulltime Research Scholar in Information Technology in the area of Natural Language Processing from National Institute of Technology (NIT), Durgapur, India under the Visvesvaraya PhD. Scheme.*

***Dr. Baisakhi Chakraborty** received the PhD. degree in 2011 from National Institute of Technology, Durgapur, India in Computer Science and Engineering. Her research interest includes knowledge systems, knowledge engineering and management, database systems, data mining, natural language processing and software engineering. She has several research scholars under her guidance. She has more than 30 international publications. She has a decade of industrial and 14 years of academic experience.*

ISSN: 0976-3104
SPECIAL ISSUE

THE IIOAB JOURNAL

ARTICLE OPEN ACCESS

# SYSTEM PARAMETER BASED APPROACHES FOR VIRTUAL MACHINE MIGRATION AND DYNAMIC LOAD BALANCING USING OPEN SOURCE XEN VMM

**Prakash kumar[1*], Krishna Gopal[1], JP Gupta[2]**

[1]*Department of Computer Science Engineering & IT, JIIT, Noida, INDIA*
[2]*Hydrocarbons Education and Research Society, New Delhi, INDIA*

## ABSTRACT

*Virtualization Technology (VT), re-invented to address most of the computer systems resource utilization challenges especially for Cloud Environment. An important feature of VT is live migration of the Virtual Machine (VM) that consists of Guest Operating Systems and applications running on it.VM migration optimizes the system performance by dynamically balancing the load. This paper proposes two-fold techniques for optimizing system performance: First, A Trigger based VM Migration technique that gets activated when CPU temperature increases beyond an upper threshold value, called Hotspot. Temperature increases due to high computational loads on the physical machine running multiple VMs. Based on the Hotspot threshold, a VM can be live migrated to another best threshold based identified physical machine available. Second, a Network File System (NFS) based dynamic load balancing strategy is proposed for better system resource utilization. This is achieved by selecting the most suitable VM for load allocation.*

**Received on:** 18th-June-2015
**Revised on:** 20th-July-2015
**Accepted on:** 03rd– Aug-2015
**Published on:** 05th –Jan-2016

**KEY WORDS**

*System performance; Virtual Machine Monitor; Hotspot; Coldspot; Migration; Load Balancing; Network File System; Cloud Computing*

***Corresponding author: Email:** kprakash91@yahoo.com **Tel:** +91-9810292083

## INTRODUCTION

The re-invented Virtualization Technology consists of the system software layer known as the Virtual Machine Monitor (VMM) which controls and facilitates the creation, adoption, implementation and running of separate instances that consists of multiple emulated and separate environments named as Virtual Machines (VMs) on the same underlying physical hardware. This VM has emerged as a need for today's datacenter blocks and system clusters. VMs facilitate the datacenter to handle multi-tenant characteristics in a very reliable, secure, flexible and efficient way, which is the need of exponentially emerging Cloud Paradigm.

The principal characteristic feature of virtualization enabled architecture is its dual state hardware- Privileged and Non-privileged access. The former makes all the instructions available to the user, whereas the latter has to make supervisory calls to the operating system nucleus, in order to have privileged access. The main characteristics of a VM is that it runs and uses only the resources allocated to it and does not go beyond that. Virtualization of the instances of operating systems are highly useful as this feature facilitates the datacenter managers to provide isolated software and hardware environments to balance the user loads in a secure, reliable and fault tolerant way. Multiple VMs own the portion of the underlying hardware resources and each VM run their own separate operating systems which are handled and managed by the Virtual Machine Monitor (VMM). Many privileged and critical instructions are executed by these VMMs on behalf of the VMs running on it [1, 2].

A hypervisor actively encapsulates each and all volatile activities and requests of a VM. So, virtualization architecture can essentially be visualized as a sandbox with various virtual environments each with user defined attributes. One of such attribute is the operating system a VM is carrying called "Guest OS". Similarly, the OS of the system over which para-virtualization is done, is called "Host OS". A VMM maps each of the VMs with a separate file onto its local file system. Every change is reflected in the file on local file system. Such file is generally an image file of the VM. If such a file is carefully copied from a specified path to a similar path of

| Guest Editor | Prof. Steven Fernandes|

another VMM with similar hardware architecture, then the new hardware can resume the VM running on previous hardware. This process of shifting a VM from one hardware to another is termed as VM migration [1].

The need to migrate the VMs arises because of the overloading of one physical machine that runs multiple VMs onto it. Consequently, there is overheating which may degrade the performance and may also lead to faulty operations and system crashes. Hence a conditional load balancing is required for better manageability of the cluster of servers [2].

There are few advantages of migration of a whole of VM. The narrow interface between a virtualized OS and the virtual machine monitor (VMM) avoids the problem of residual dependencies. There are various ways to migrate a VM from one physical node to another. “Pure stop-and-copy” or “Cold” migration technique halts the VM and copies all its associated memory pages to another pre-identified destination node and then resumes the VM on it. On the contrary, few selected hypervisors like Xen and VMWare do a “Live” or “Hot” migration. The advantage of the latter method is that even the concerned applications and processes are unaware of the VM migration [3].

A new technique to trigger VM migration or any other desired causal effects of temperature and CPU usage variation on VMM Xen 4.2 as a whole and VMs with host OS Ubuntu 13.04 are discussed in following sections.

## BACKGROUND AND RELATED WORK

The cloud services providers while focusing on amazing user experience of their services also stress on the need of optimized usage of their resources and data durability of their users. So they developed various algorithms and implementation to migrate a VM in various cases like excessive CPU requirement, memory constraints, Stagnant/idle, etc.

There are few techniques to resolve such issues which include VM migration and Process Migration. On Xen hypervisor, the VM migration feature can easily be run and effects analysed with xm-migrate command. And this command has been highly used and emphasized on since it can be modified with variety of attributes that goes with the command such as whether we want the migration to be live or cold. But the real problem arises on detecting when to fire the VM migration mechanism and detecting which VM is the causing the trouble.

Process migration demonstrates a functionality of transferring a process running on one machine to the other. But, there is an inherent difference in the operating concepts of virtual machine migration and process migration [4]. Though in practice, migrating the process is difficult and quite complex as it should take care of legacy applications and at the same time it should also leverage the currently installed and related large databases of operating systems and maintain independence on different machines. These can be overcome by using a VMM based migration. VMMs such as VMware use Hardware abstraction to encapsulate the complete OS environment in such way that it can be suspended from one machine and resumed onto the other one provided there are inherent similarities in the system architectures of the operating systems.

But VM migration supersedes Process Migration except in some cases that occur due to the narrow interface between a virtualized OS and the virtual machine monitor (VMM) where itavoids the problem of residual dependencies. VM migration has the advantage of transferring internal memory states in a very consistent and efficient way [4].

Another part is that the System Virtual Machines (VMs) [5, 6] are widely used from personnel computers to large organizations. System virtualization acts as powerful means of abstraction for upcoming applications. On the cloud computing platform resources are provided according to need on the principle of pay per use. To accommodate specific requirements of subscribers and how the balance is maintained between Cost, Quality and Resources is mentioned in the Service Level Agreement (SLA). The Cloud Service Providers guarantee a level and quality of service to the users as per the terms and conditions of the SLAs.  A lot of challenges are faced while catering the need of the users and at the same time making efficient use of underlying heterogeneous resources in a dynamic and efficient way, which is inherently expected from Cloud Services in terms of Infrastructures, platforms and software.

By mapping the services onto the Virtual Machines (VMs), where multiple VMs can run onto a single physical server, the problems related to heterogeneity in hardware, software and platforms could easily be solved.

Consequently the terms and conditions for the SLAs can be agreed upon between the Cloud Service Providers and the Service Users.

VM Load Balancing is crucial characteristics of system virtualization, allocate and shift the running applications dynamically to other physical machines as and when the load increases on any particular machine [7]. Because of the complexity aroused due to the vast heterogeneity in terms of underlying hardware, operating systems environment, platforms and communication technologies, and that too at the run time, it is inevitably important to address the performance and issues related to smooth delivery of services to the end users. Consequently, addressing the resource allocations and related issues, viz. VM Load balancing, VM Scheduling techniques, VM migration [8], VM performance optimization and cross platform operations issues are the need of the hour for Cloud environments, and that too with a guaranteed level of services [9].

This paper addresses two-fold techniques for optimizing system performance: First, A Trigger based VM Migration technique that gets activated when CPU temperature increases beyond an upper threshold value, called Hotspot. Temperature increases due to high computational loads on the physical machine running multiple VMs. Based on the Hotspot threshold, a VM can be live migrated to another best threshold based identified physical machine available. Second, a Network File System (NFS) based dynamic load balancing strategy is proposed for better system resource utilization. This is achieved by creating performance models for VM load balancing. Many experiments were conducted on Xen for Virtual Machines [10] for Network File Systems. Load balancing is done by scheduling the VMs on a particular physical machine that is comparatively having less load. This method computes the load on various virtual machines and then finds the virtual machine which is most suitable for the upcoming load.

Load balancing is the capability of the system which allows the VM hosted applications to be transparently allocated a VM which has the least load dynamically so as the maximum resource utilization of the whole system is achieved.

There have been many approaches to load balancing viz. Static and dynamic. In addition, some hybrid approaches are also adopted. Major difference being in Dynamic Load Balancing, decision is taken at runtime according to the existing situations, whereas in static it is not. Neither of them is superior or inferior but the selection of algorithm depends on the application requirements.

### Static load balancing

Static Load Balancing (SLB) refers to the load balancing algorithm that distributes the load strictly on the basis of certain predefined rules relating to the nature of input loads. It does not consider which node is receiving more or less load. In all static algorithms final selection of the virtual machine is done immediately after creation of application. Further it cannot be changed while in execution. These static load balancing techniques are suitable for a system in which load is limited and request of the clients is also limited. But nowadays load on cloud servers is also increasing and also the load is not static hence we need more efficient algorithms then static load balancing algorithms.

Subsequently we describe some of the basic algorithms for static load balancing as follows:

### Round Robin Algorithms

Whenever a new application comes it is assigned a virtual machine in a round robin fashion. In general, basic idea for Round Robin [11] is to reduce message passing between various virtual machines and reduce communication delay. Thus it is independent of the state of the system. When coming applications are of similar load then Round Robin works very well as it reduces the communication delay due to inter-process communication. Thus Round Robin has best performance for this special purpose application of similar load, but does not give a good performance for general cases.

### Randomised Algorithm

Random numbers are distributed on a basis of a statistical distribution and assigned to virtual machines. Incoming applications are distributed according to these randomly generated numbers. This algorithm is applicable when we have many virtual machines as compared to processes

### Central Manager Algorithm

In this algorithm (Huang, 2012), there is a central virtual machine and others are slave virtual machines which are assigned applications to be executed. Central virtual machine's task is to gather load information of all the slave virtual machines and assign the coming application to the least loaded slave virtual machine as shown in **Figure-1**.

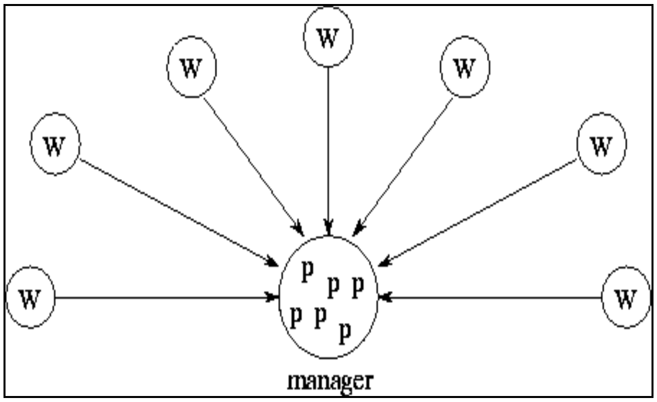


**Fig: 1. Manager collecting information of slaves**

### Threshold Algorithm

In this algorithm the current virtual machine is decided on the basis of two values of upper (t_upper) and lower (t_lower) threshold. Virtual machine is assigned a state depending on its current load compared to these threshold values. If current load is less than the t_lower virtual machine is assigned a Under Loaded State, if its greater than t_upper the state is overloaded, if its between the two threshold values then the state is Medium.

Initially all virtual machines have under loaded state. But as the system advances the load level limit of a virtual machine may change and its state may change. If the state changes then it send message to all other virtual machines notifying the change, so that they can maintain the load state of entire system. When a process arrives and the local virtual machine is under loaded then it executes the application else calls for the remote virtual machine. If no under loaded virtual machine exists then application is executed locally only.

### Dynamic load balancing

Dynamic Load Balancing (DLB) techniques provide a method to dynamically allocate load based on self-adapting distribution and intelligent distribution. Here, it is distributed at runtime based on the new information collected. Mainly these techniques are based on greedy algorithmic approaches. Basic algorithms for dynamic load balancing are:

#### *Central Queue Algorithm*

The host virtual machine maintains a central queue of all the applications. This queue is shared by all the processes. New applications are added and pending applications are maintained in a cyclic FIFO order in the queue. When a virtual machine is free it will request for application for executing to the host and host assigns the application next in queue to the virtual machine which is demanding the request. If there is no application for execution in the queue then the request is buffered in queue form. And request is answered when new application arrives.

#### *Load Queue Algorithm*

Here the applications are assigned virtual machines similar to the static algorithm but a virtual machine here can initiate application migration. Initially all under loaded virtual machines are assigned the applications. The applications are assigned following some static algorithm. Then if a virtual machine's load goes below the lower threshold value then it asks for load from other machines and initiates migration process.

THE IIOAB JOURNAL ISSN: 0976-3104

A lot of scheduling algorithms have been re-invented and tested for cloud environment, namely intelligent scheduling algorithms, autonomous scheduling algorithms, agent-based negotiable scheduling algorithm, centralized scheduling algorithms. These algorithms are used by many popular Cloud Service Providers for balancing the loads on their virtual machines; a popular example could be VM Ware Distributed Resource Scheduler (VMDRS).

Analysing all these algorithms our paper presents a balanced techniques for balancing load on Xen Hypervisor using the technique of Network File System for file Sharing.

## Our Contribution

The technique of Virtual Machine migration has been very helpful in maintaining VMs in a way more resource optimized but the cause for migration is very significant that we have been ignoring for a long time. In This paper as already discussed, focus will be on the importance of various trigger options that initiate the migration procedure. It has been focused on the thermal and CPU usage of the data server as the trigger for VM migration. This leads to a scenario where the user can choose what to do in case of various faults. The `"Cold Spot", "Hotspot" as thermal issue of data server hardware have been looked into. Similarly, "Overload" and "Under load" are the issues alarming CPU usage issues. Shell scripting was used as a tool for determining the boundary conditions for each case and then the triggering part comes into play. Python socket connection was also used so as to facilitate the communication between two PCs acting as data servers independently.

The connection so set up will ask the data server 2 to get ready to receive an image file. On data server 1, an image of VM to be migrated will be selected from the Xen VM image repository and sharing the file with data server 2 over the network. The server will now install this image over itself.

The VM migration in general can be classified into three phases:

1. Recognizing trigger: This phase implies the detection of a trigger that may cause the VM to crash or harm the memory space. This detection can be based on the various parameters like CPU utilization, Process throughputs etc.
2. Image packing: Creating the image file of the memory space and packing the image with headers.
3. File Sharing: Sharing this image file over the hosts/data server network using Network File Sharing.
4. VM replication: Extracting the VM over the data server from the image.
5. Stability test of newly formed VM
6. Instant transfer of workload to new VM.
7. Deletion of old existing VM.

A hotspot/ cold spot is an undesirable hardware temperature fluctuation which generally occurs when data centre is improperly cooled.Hotspot is dangerous problem since it can cause serious trouble to the hardware. A typical data server contain significant amount of power consuming components this producing so much heat like a furnace if not properly cooled.Cold spot on the other hand occurs when the equipments installed in a data server receive too much of the cooling thus posing the moisture trouble and causing disruptions in electronic circuits.

The ASHRAE (American Society for Heating, Refrigerating and Air-Conditioning Engineers) standard cold spot for a typical data server is 64.4 degree Fahrenheit or 18 degree Celsius. And hotspot depends on the quality of hardware use but generally 85+ degrees Celsius is considered highly critical.

## Functional Requirements

Functional requirements define the fundamental actions that must take place in the software in accepting and processing the inputs and in processing and generating the outputs as shown in **Figure-2**.

**Table: 1. Compatible Paring** (source: Linux Foundation, 2014)

| Ubuntu Version | Xen Release |
|---|---|
| Ubuntu 12.10 | Xen 4.1 |
| Ubuntu 13.04 | Xen 4.2 |

*Compatibilities Issues:*

1. Enough Memory space so as to create an image of data to migrated.
2. Smooth Internet connection to eliminate any data losses or interruption
3. All the data servers must be in the same subnet network.
4. Python 2.7+ packages must be installed.

5. Libvirt and libvirsh must configured
6.Live migration settings must be configured.

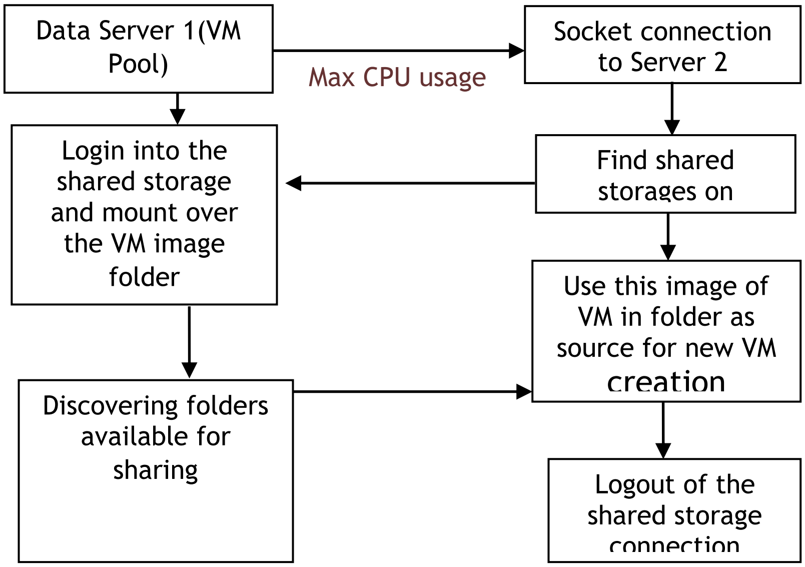

Data Server 1(VM Pool)
Max CPU usage
Socket connection to Server 2
Login into the shared storage and mount over the VM image folder
Find shared storages on
Use this image of VM in folder as source for new VM creation
Discovering folders available for sharing
Logout of the shared storage connection

**Fig: 2: Steps of Execution**

### Non-Functional Requirements

1. *Error handling*
   - Expected and non-expected errors shall be handled in such a way that it prevents loss of information with minimum downtime.

2. *Performance Requirements*
   - Depends on network connection.
   - Processor cycles to execute commands.

3. *Security Requirements*
   - Security of the data involved in migration depends on the network security .
   - Do not apply network security on a private network because it impacts the performance a lot. However, security measures should be implemented when the migration traffic needs to be encrypted

4. *Reliability*
   - The procedure is reliable as long as the data selected is accessible and the migration is nt interrupted.

5. *Correctness*
   - The procedure implemented in the system should be correct which means that they should be performed as required. The testing Phase insures correctness of the software by trying all possible Case and matching their output with the documentation.

## IMPLEMENTATION OF TRIGGER BASED MIGRATION

Virtualization can be implemented using any of the readily available hypervisors but choosing the right one and feasible and good support is important. Here open source VMM namely Xen 4.2 over Ubuntu 13.04 is chosen as this is highly in use in most of the commercial and open source freeware VMMs. Xen is a very popular hypervisor among all cloud communities. It has full compatibility and support by x86, IA-32, ITANIUM, VT$_{x,}$ VT$_i$ and most of the PIC, ARM architectures. It is supported by many operating systems like Solaris, Windows, Linux, etc. as guest operating systems on their CPU architectures. Xen can do full virtualization on systems that support virtualization extensions, but can also work as a hypervisor on machines that don't have the virtualization extensions. Citrix has collaboratively launched a Xen Citrix server.

Performing the objective so discussed is crucially dependent on the installation and configuration of the VMM used.

Xen virtualization is ready to deploy but now a trigger is needed to initiate the VM migration. The host Os doesn't know about the Virtual machines running over it except it knows the existence of VMM over it and it is same for VMs as well, they know Xen only not the presence of any host Os. So for host Os to calculate any over loading CPU usage on account of virtual machines, it has to be the xen process on host Os that should be monitored. Hence, the following command will help in retrieving all processes responsible for CPU cycle usage and can be monitored accordingly.

```
>> top -bn1 | grep "Cpu" | sed "s/.*, *\([0-9.]*\)%* id.*/\1/"
```

Above command can be used to write a shell script that could calculate cumulative CPU usage and hence deal with "Over loading" and "Under use "in order to migrate VM thus optimizing resource use. One shouldn't let a data server host VMs that could be accommodated on other data server thus reducing the expense on a data server. In case of Cold spot/ Hotspot, one needs to continuously monitor the thermals of the hardware involved, thus,
>> *install acpi* (to handle power and thermal related issues and logs one needs to install this on the machine first).
Then only can one monitor using the command:
>>acpi –t| sed "s/.*ok,\([0-9]*)*\).*/\1/"(Command for monitoring CPU temperature)

1. *sudo vim /etc/default/iscsitarget:* (To allow discovery of target machine change the value to true.It normally resides in the default directory of iscsi).
2. *Sudo vim /etc/iet/ietd.conf*(In this set the name, path and type of the iscsi storage).
3. *Sudo /etc/init.d/iscsitarget restart* (To make changes take into effect).
4. *Cat /proc/net/iet/session* (Session is used to start the session used to allow the discovery of the storage to be available to the client system).
5. *Iscsiadm –mode discovery –type sendtargets –portal ipaddress*(To discover the list of available storage device on which can be target to migrate our machine).
6. *Iscsiadm –m node –t NameOfTarget –p ipaddress login*(Login to the storage device. In case it is secured then requie password otherwise it will login automatically).(Commands for migration)
7. *Dmesg*(To detach from iscsitarget)
8. *Iscsiadm –m session –logout*(To logout from the target it is required to log out from the target from all the address).

The above procedure should result in an image of VM to be migrated in the destination data server.

## VM SELECTION FOR MIGRATION

The tricky part is deciding to choose which VM must be migrated from one data server to another in case a trigger has been fired and VM migration is inevitable. In such critical conditions, it is logical and feasible to propose that such a dilemma can be tackled by listing every VM that particular hardware hosts and then running a script to find the combination of VMs whose CPU usage can add up together to maximise CPU usage thus avoiding CPU overloading, In addition, this also helps in deciding as to which VM should be migrated. Above discussed method can be compared to best fit allotment analogy. However, this algorithm can only be applied if the trigger was CPU overload. The algorithm and the environment is still in studying phase to improve and broaden the parameters to design a better algorithm. The ***"Xm"*** command set's ***"Xmtop"*** instruction is quite useful in the algorithm mentioned.

To understand the algorithm in a better way, an example is shown: Assume there are three VMs with CPU usage 25%, 25%, 40%, thus leading the machine to CPU overload. This scenario will cause the machine to fire the trigger and hence migration. Now comes the part when it decides which VM to migrate. Here the maximised combination of CPU usage is 40% and 25% together, if add the other 25% to it, it'll shoot over 85%, which is theupper threshold decided by the standards. Hence, either of the VMs with 25% CPU usage can be migrated to stabilize this host.

### Setup for VM Load Balancing

#### *VM Model*

Initially, the physical machines are mapped to the VMs as shown in fig :3: Three physical machines have been taken; where two VMs are installed on two of them and three VMs on one of them i.e. in total, there are three physical machines and seven VMs. To generalise the abovementioned, there are N physical machines and M Virtual Machines. Then the active Set of existing machines can be given as, PM= {$PM_1$,$PM_2$,…,$PM_N$}, where, $PM_i$(1<=i<=N) denotes the No.i for the physical machines. Physical Machine $PM_i$ has $m_i$ VMs on it represented as $V_i$= {$V_{i1}$,$V_{i2}$,…,$V_{imi}$} and $m_1$+$m_2$+….+$m_N$= M

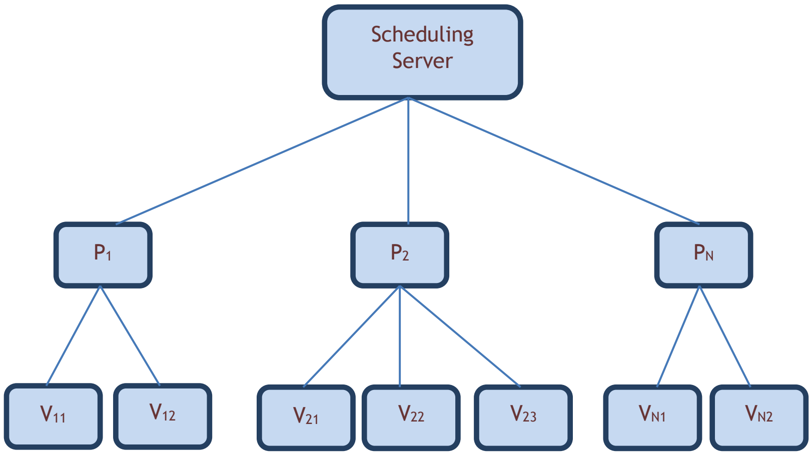


**Fig: 3: System Structure**

#### *Expression of Load*

The load on the physical machine is calculated by adding the total loads of all the Virtual Machines (VMs) running on this physical machine. The load of VMs is measured after a particular period of time say j, where j is the time period between ($t_j$– $t_{j-1}$). Assuming that the load of VMs is relatively constant in every time period, load of the VM No.i can be defined as V(i, j) for the time period p. Supposed that there are n VMs whose loads needs to be measured in the time period k. Then the loads of each one can be measured accordingly e.g. load of $VM_1$ at (1*p/n), $VM_2$ at (2*p/n) and so on. Thus, the load value of a particular VM gets updated once in each time period of duration k.

Therefore, it can be concluded that in cycle T, the average load of VM $VM_i$ on physical machine $PM_i$ [31] can be given as

$$\overline{VMi(i,T)} = \frac{1}{T}\sum_{j=1}^{n} VM(i,j)\times(t_j \text{-} t_{j\text{-}1})$$

Obviously, the total load of any physical machine is the sum of all running VM loads. Hence, load of physical machine $PM_i$ is given by:

$$PM(i,T) = \sum_{j=1}^{mi}\overline{VMi(j,T)}$$

## IMPLEMENTATION

For implementing a load balancer on Xen the specification of the system used were:

- 64bit x 86 computers with 3GB of RAM, 320GB of storage space.

- Processor with $VT_D$ , $VT_X$ enabled
- Ubuntu 12.04 as Operating System

Initially Ubuntu12.04 was installed as host operating system. A hypervisor was created using Xen project, which enabled executions of multiple guest operating system simultaneously on a single physical machine. In particular there are two types of hypervisors, these are type1(native or bare metal) and type2(hosted).Our project is using bare metal hypervisor meaning the hypervisor layer is created directly on the host hardware, which allows the hypervisor to control the hardware. There is a concept of domains in Xen Hypervisor. There are two type of domains, domo-which controls the functioning of the hypervisor and starting the operating guest operating system.. For our work we have chosen Ubuntu 12.04 as dom0 machine. other guest operating systems are called domUs, this is because these domains are “unprivileged” in the sense they cannot control the hypervisor or start/stop other domains For our work we have created 3 machines as domUs in two of them Ubuntu 12.04 is installed and in one of the DSL in installed. For the purpose of communication between different domains, and also intra domain communication, a NFS is configured. NFS is a distributed file system protocol which allows sharing and transfer of files across various nodes on a particular network so to share files across domains we have to use NFS.

### *Algorithm*

This algorithm repeats itself after an interval of time ‘k’.

```
1) After time k the dom0 as updated load status of each VM
2) Dom0 machine will read the load
3) A:= load of vm1
4) B:=load of vm2
5) C:=load of vm3
6) If(A<B && A<C)
   then
       allocate task to vm1
   else if(B<A && B<C)
   then
       allocate task to vm2
   else
       allocate task to vm3
   end if
   end if
```

## ANALYSIS

Our experimental work mainly analyzes the effect of implemented load balancing strategy and compares this method with the performance of a system without load balancing. Here we draw a comparison between three kinds of systems. Firstly, a system running on a single operating system. Second, a system in which Xen hypervisor is implemented without using the technique of load balancing thirdly, a system with Xen hypervisor installed and also load balancing techniques implemented.
Figure 4 shows the measured comparison of response time when number of request increases. The improvement done by our work is clearly visible through this comparison.

Figure 5 shows that when there is only one operating system installed then the performance of system decreases when load increases, while if a hypervisor is installed then initially the performance is lower than native performance due to certain overheads, but as load increases performance becomes better. Performance is best when a load balancer is also implemented. In these plots, Y-axis depicts the performance of the system whereas X-axis depicts the number of tasks.

ISSN: 0976-3104

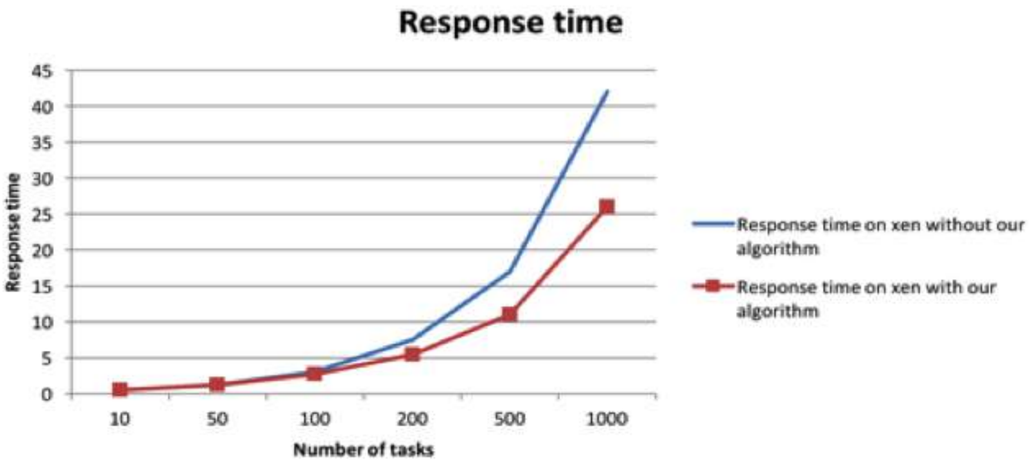


Fig: 4. Comparison of response times

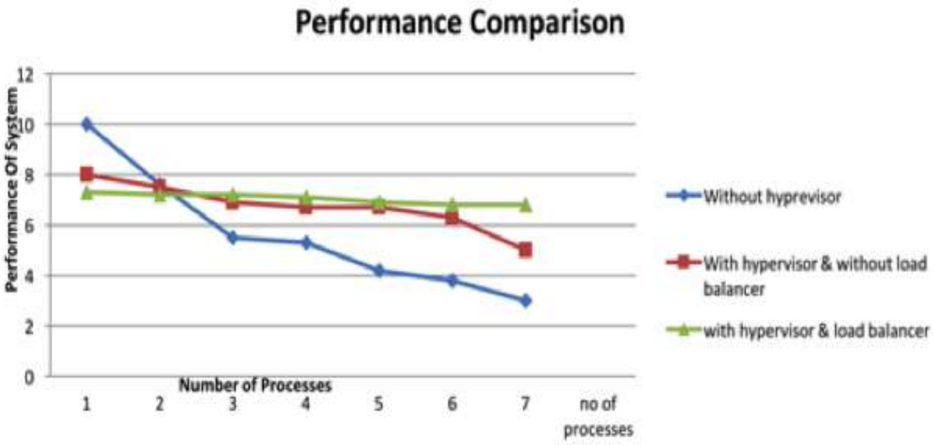


Fig: 5. Comparison of System performance (CPU usage) with and without load balancer in systems with and without hypervisor

## CONCLUSION

The migration of a VM is a tricky and complex process. It may lead to loss of VM or corruption of the same if not handled properly. This innovative technique of trigger based VM migration will surely provide users to take more control over its VM and help them to better manage their VMs,consequently, a better and efficient way of managing the underlying system resources, hence bolstering the core idea of virtualization, which is the technical backbone of cloud computing.

A dynamic load balancing strategy and algorithm is developed and implemented on Xen VMM for VM load balancing based on Network File System (NFS). According to the current states of VMs, it computes and identifies in advance, the most suitable VM where the upcoming application can be allocated. The overhead involved for identifying suitable VMs for successive allocations is drastically reduced as the previous and current states are already available. Additionally, this strategy is quite efficient and requires less computational overhead as compared to normally used Migration strategies and other traditionally adopted load balancing techniques and hence better resource utilization comparatively.

## CONFLICT OF INTEREST
Authors declare no conflict of interest.

## ACKNOWLEDGEMENT
None.

## FINANCIAL DISCLOSURE
No financial support was received to carry out this project.

## REFERENCES

[1] Abels T, Dhawan P. [2005] An Overview of Xen Virtualization. *Dell Power Solutions.*

[2] Anderson AV, Benett S M, Kagi A, et al. [2005] Intel Virtualization Technology. *IEEE*. http://doi.ieeecomputersociety.org/10.1109/MC.2005.163.

[3] Shenoy P, Venkataramani A, Wood T, Yousif M. [2007]Black-box and Gray-box Strategies for Virtual Machine Migration, 4th USENIX Symposium on Networked Systems Design & Implementation, 229–242.

[4] Sandhu S,Venkatesh, S. [2009] Survey of VM migration Techniques. *IEEE.*

[5] Carl A, Waldspurger. [2002] Memory Resource Management in VMware ESX Server. Proceedings of the 5th Symposium on Operating Systems Design and Implementation , 181–194.

[6] Dragovic B, FraserD, Hand S, et al. [2003] Xen and The Art of Virtualization. Proceedings of the 19th ACM Symposium on Operating Systems Primciples. 164–177.

[7] Freeman T, Foster I, Keahey, K, Sotomayor, B. [2007] Enabling cost-effective resource leases with virtual machines. *International Symposium on High Performance Distributed Computing*.

[8] Clark C, Hand S. [2005] Live Migration of virtual machines. 2nd Symposium on Networked Systems Design & Implementation.

[9] Cherkasova L, Gupta D, Vahdat A. [2007] When virtual is harder than real: Resource allocation challenges in virtual machine based it environments. *Technical Report HP,* 1–10.

[10] Ballani H, Costa P, Karagiannis T, Rowstron A. [2011] Towards predictable datacenter networks. Proceedings of ACM SIGCOMM, 242–253.

[11] Huang R, Xu Z. [2012] Performance Study of Load Balancing Algorithms in Distributed Web Server Systems. CS213 Parallel and Distributed Processing Project Report, 3679-3683.

## ABOUT AUTHORS

***Prof. Prakash Kumar** has received his B. Tech. in Electronics and Communication Engineering, and M. Tech in Computer Science and Technology from University of Roorkee (Now Indian Institute of Technology, Roorkee), India. He is currently an Assistant Professor (Senior Grade) in Jaypee Institute of Information Technology,(Deemed University), Noida, India. His area of interest is in Computer Networks and Communications, Distributed Computing, Cloud Computing and Virtualization,. He is currently pursuing his Ph D in the field of virtualization of resources viz. Systems and network resources. His main focus is on Trust, Reliability and Fault Tolerant networks and systems for distributed and cloud environments.*

***Prof. Krishna Gopal** is currently Dean Academic and Research at JIIT, Noida, India since 2011. He is Ph. D. from REC Kurukshetra, Kurukshetra, India. He is having 45 years of teaching and R&D experience. He received his Bachelor, Master and PhD in Electronics engineering from IIT, Madras, NIT Kurukshetra in 1966, 1972, 1979 respectively. He published more than 100 papers in different journals, conferences, patents etc. He is member of various professional bodies like: Life Member System Society of India, Indian Society for Technical Education, senior member of IEEE etc. His area of interest is Reliabilitv and Fault Tolerant Networks and communication Svstems.*

***Prof. J P Gupta** has received his Ph D degree from University of Westminster, UK. He is currently the Director Emeritus (QA) at Hydrocarbons Education and Research Society, New Delhi, India. He is an academician having more than 35 years experience including Professor at IIT Roorkee, India, Vice Chancellor at JIIT Noida,India, Galgotia University, and Sharda University, India. He is author of more than 70 research papers published in International journals and conferences. He has received Commonwealth Fellowship and many more awards and memberships to his credit.*

ISSN: 0976-3104
SPECIAL ISSUE
Danti and Bhushan.

ARTICLE OPEN ACCESS

# CLASSIFICATION OF TEXT DOCUMENTS USING INTEGER REPRESENTATION AND REGRESSION: AN INTEGRATED APPROACH

**Ajit Danti[1] and SN Bharath Bhushan[2*]**
*[1]Department of Computer Science, College of Computer Science, King Khalid University, SAUDI ARABIA*
*[2]Karnataka Government Research Center, Sahyadri College of Engineering and Management, Mangalore, KA, INDIA*

## ABSTRACT

*Text Classification approaches is receiving more and more attention due to the exponential growth of the electronic media. Text representation and classification issues are usually treated as independent problems, but this paper illustrates combined approaches for text classification system. Integer Representation is achieved using ASCII values of the each integer and later linear regression is applied for efficient classification of text documents. An extensive experimentation on four publically available corpuses are carried out to show the effectiveness of the proposed model which provide better result as compared to static fault models. This paper provides a study and detailed analysis of result and experiment performed.*

**Received on:** 18th-June-2015
**Revised on:** 20th-July-2015
**Accepted on:** 03rd– August-2015
**Published on:** 8th-Jan-2016

KEY WORDS

*Integer Representation, Text Classification, Regression based searching.*

***Corresponding author: Email:** ajitdanti@yahoo.com, sn.bharath@gmail.com; **Tel:** +91-9480766063

## INTRODUCTION

Text data, especially the increased popularity of the Internet and the World Wide Web became the most common types of information to store house. Most common sources are web pages, emails, newsgroup messages, internet news feeds etc,[1]. Many real time text mining applications have gained a lot of attention due to large production of textual data. Many applications of text classification are spam filtering, document retrieval, routing, filtering, directory maintenance, ontology mapping, etc.

The goal of the text classification algorithm is to identify text documents with the ontology of domains defined by the subject experts. In text classification a boolean value will be assigned to each pair where is a set of predefined categories and is the theme of documents. The task is to approximate the true function by means of a function such that coincide as much as possible. The function is called a classifier. A classifier can be built by training it systematically using a set of training documents [1]. Generally, textual data being unstructured in nature, pose a number of challenges such as desired representation model, high dimensionality, semanticity, volume and sparsity. Some solutions for these challenges can be found in [2].

In this paper an integer representation for text document which minimize the amount of memory required to store a word which in turn reduces the processing cost is proposed. Text representation algorithm works on the principle that, an integer number requires minimum memory when compared to store a word. Integer representation based classification of text documents is an unconventional approach for classification of text documents.

The rest of the paper is organized as follows. In section 2 a brief literature survey on the text classification is presented. In section 3, a proposed model for the compression based classification of text document. Section 4 discusses about experimentation and comparative analysis performed on the proposed models. Paper will be concluded in section 5.

## LITERATURE SURVEY

In literature few works of compression based text classification can be seen. Generally, text compression involves context modeling which assigns a probability value to new data based on the frequencies of the data that appeared before. But, context modeling algorithms suffers from slow running time and require large amount of main memory. Mortan [3] proposed a modeling based method for classification of text documents. A cross entropy based approach for text categorization is presented in [4]. It is based on the fact that the entropy is the measure of information content. Compression models are derived from information theory, based on this theoretical fact, language models are constructed for text categorization problem. Authors have also illustrated that, character based prediction by partial matching (PPM) compression schemes have considerable advantages over word based approaches. Frank [5] considered the task of text classification as a two class problem. Different language models such as Ma and Mb are constructed for each class using PPM methods. Test document will be compressed according to different models and gain per model is calculated. Finally class label will be assigned based on the positive and negative gain. Modeling based compression for low complexity devices is presented in [6]. This method is based on the fact that, PPM based approaches require high computational effort which is not practically advisable for low complexity devices such as mobile phones. Algorithm makes use of static context models which efficiently reduce storage space. Similar type of work for low complexity devices are found in [7]. This approach split the data into 16 bit followed by the application of Quine-McCluskey Boolean minimization function to find the minimized expression. Further static Huffman encoding method is used for text compression. Dvorski proposed an indexed based compression scheme for text retrieval systems. A document is considered as a combination of words and non words. Similarly, Khurana and Koul [8] considered English text as a dictionary where each word is identified by a unique number in which a novel algorithm is proposed which consists of four phases, where each phase is for different type of input conditions along with a technique to search for a word in the compressed dictionary based on the index value of the word. Word based semi static huffman compression technique is presented [9] in which algorithm captures the words features to construct byte oriented huffman tree. It is based on the fact that byte processing is faster than bit processing. Automaton is constructed based on the length of search space. End tagged Dense code compression is proposed in [10]. Though the proposed method looks similar to tagged huffman technique, the algorithm has the capacity in producing better compression ratio, constructing a simple vocabulary representation in less computation time. Compressed strings are 8% shorter than tagged huffman and 3% over conventional huffman. Another word based compression approach is found in [11]. This method compresses the text data in two levels. First level is the reduction which is through word look up table. Since word look up table is operated by operating system, the reduction is done by operation system only. According to this method each word will be replaced by an address index. Next stage is the compression stage. Deflatge compression algorithm is used for compression. Four different huffman, W-LZW, word based first order and first order context modeling methods for text compression are presented in [12]. All the word based compression techniques discussed above, maintain two different frequency tables for words and non- words.

Many approaches for classification of text documents can be found in literature. These approaches include naïve bayes [13, 14], nearest neighbor [15-17], decision trees [18], support vector machines [19] and neural network [20] approaches.

## PROPOSED METHOD

In this paper, proposed model can be categorized into two stages, such as text representation stage and regression based searching stage.

### Text Representation

It is theoretically verified that a sequence of characters requires more memory than an integer number. Based on this, a novel text representation algorithm is proposed, which has the facility of representing character string (a word) by a unique integer number. It is known that, words are the collection of alphabets, which represent a specific meaning. Similarly a text document is collection of such strings which represent a specific domain. Words from the text documents are extracted. Each word is then subjected for compression algorithm and then it is represented by an integer number. The whole procedure is algorithmically represented in the algorithm 1 and pictorially represented in **Figure– 1** and the proposed method is explained in illustration-2.

**Cumulative sum of ASCII value is determined for given textual word using the equation (1)**

$$\textbf{Cw}=\sum_{k=1}^{n} a_k\, b^k \qquad \ldots(1)$$

Where,

**$C_w$ = Cumulative sum of ASCII values**

w = length of the document. (Number of words in the documents).

k = number of alphabets in the word.

a = ASCII value of alphabet.

b = base.

**Illustration : 1**

Input word: **heart**.

ASCII values: 104,101,97,114,116.

= 104 x $2^4$+101 x $2^3$ + 97 x $2^2$ + 114 x $2^1$ + 116 x $2^0$ = **3204.**

| Data | Before Compression | After Compression |
|---|---|---|
| heart | 5 bytes | 2 bytes |

**heart** will be represented by an integer number **3204**.

**Fig: 1. Pictorial representation of Compression algorithm**

## Regression Based Searching Stage

Now, the text document can be viewed as a collection of integer vales. As a result text classification problem got reduced into integer searching problem. Once the data is represented by an integer value, it is sorted and linear regression is applied using the equation 2.

$$a_0 = \frac{\sum x_i y_i \sum x_i - \sum x_i^2 \sum y_i}{(\sum x_i^2 - n\sum x_i^2)}$$

$$a_1 = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - (n\sum x_i^2)} \qquad \ldots(2)$$

Where, x = Positional Value, and y = word.

Once the regression algorithm is applied, the data will represented by a straight line, as shown in **Figure– 2** using the equation (3)

$$y = a_1 x + a_0. \qquad \ldots(3)$$

where *x* gives the appropriate position of the search key element as shown in **Figure– 2**.

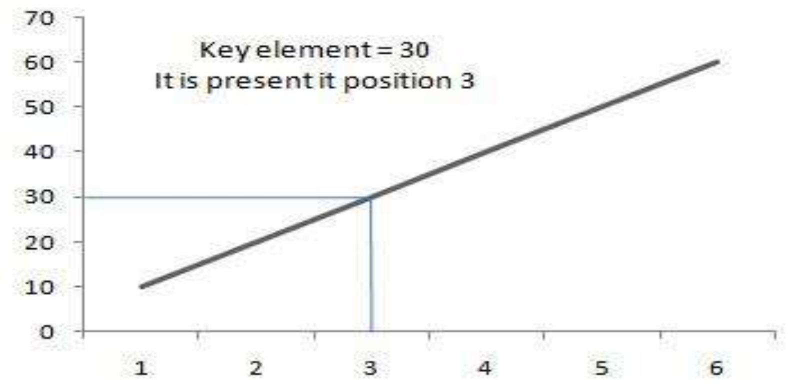


Fig: 2. Regression based searching

Classification of text documents will be accomplished by subjection training documents to representation algorithm. Once the data representation stage is accomplished linear regression is calculated for each class. Effect of this process, training phase will be seen as collection linear regression values. Then, test documents will be subjected to the compression algorithm. As a result, test documents will be collection of integer number. Each integer number from testing document is considered and is fed to regression based searching algorithm. The main advantage of the regression is that, it takes minimum computational unit to search an integer number from the database. Similarly the procedure is carried out and class label will be given to all the integer values. Test document will be assigned a class label based on the maximum class label assigned each integer. **Figure– 3** present the block diagram of the proposed method.

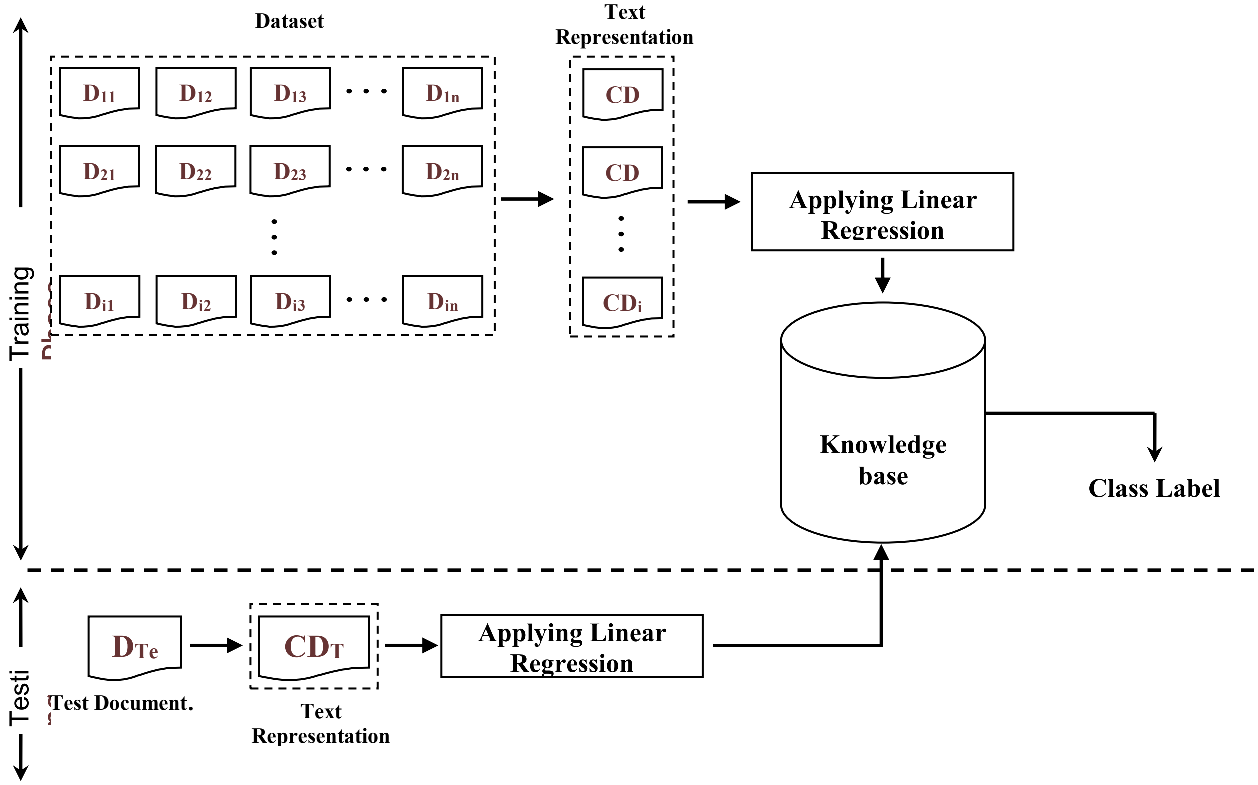


Fig: 3. Block diagram of the proposed approach

## EXPERIMENTATION

The proposed method is evaluated for its effectiveness and efficiency on the publicly available datasets. The first data is from Wikipedia pages which include the characteristics of the vehicle that the vehicle is in the data. The second dataset includes 10 different classes for 1000 documents are from Google newsgroup data. The third and fourth is from 20 Mini newsgroup and 20 newsgroup datasets. To check the efficiency of the proposed model, two sets of experiments are conducted. First set of experimentation consists of 40% training and 60% testing on all the four datasets. Second set of experimentation consists of 60% training and 40% testing on all the four datasets. The details of the first and second set of experiments are shown in **Table– 1**.

**Table: 1. Classification result of proposed compression and regression based technique**

| Datasets | 40% : 60% | | | | 60% : 40% | | | |
|---|---|---|---|---|---|---|---|---|
| | Num of Training | Num of Testing | f Measure | Class Accuracy | Num of Training | Num of Testing | f Measure | Class Accuracy |
| Vehicle Wikipedia | 44 | 66 | 0.9273 | 92.61 | 66 | 44 | 0.9476 | 94.70 |
| Google Newsgroup | 40 | 60 | 0.9205 | 92.00 | 60 | 40 | 0.9321 | 93.16 |
| 20 Mini Newsgroup | 40 | 60 | 0.9003 | 90.00 | 60 | 40 | 0.9184 | 92.25 |
| 20 Newsgroup | 400 | 600 | 0.8873 | 88.62 | 600 | 400 | 0.8960 | 89.47 |

F measure is considered for the evaluation of the proposed methods, precision, recall and class accuracy (CA) for each set of experiments using the equation (4),(5),(6) and (7). Let a,b,c and d respectively denote the number of correct positives, false negatives, false positives and correct negatives.

$$fMeasure = \frac{2PR}{P+R} \qquad \ldots (4)$$

Where,

P(Precession) = a / (a + c) . . . (5)
R(Recall) = a /(a+d) . . . (6)
CA(Class Accuracy ) = (a + d) / N . . . (7)

## CONCLUSION

This paper illustrates a method of providing an integer representation of text and regression for classification of text documents is presented. An extensive experimentation is carried on four publically available datasets to show the efficiency of the proposed models. The performance evaluation of the proposed method is carried out by performance measures such as f-measure and class accuracy (CA). The proposed model is very simple and computationally less expensive. One can think of exploring the proposed model further for other applications of text mining. This can be one of the potential directions which might unfold new problems.

## CONFLICT OF INTEREST

Authors declare no conflict of interest.

## FINANCIAL DISCLOSURE

No financial support received to carry out this research.

ISSN: 0976-3104

## REFERENCES

[1] Rigutini L. [2004] Automatic Text Processing: Machine Learning Techniques.Ph.D. Thesis, University of Siena.

[2] Sebastiani F.[ 2002] Machine learning in automated text categorization. *ACM Computing Surveys* 34:1 – 47.

[3] Mortan Y, Nign Wu, and Lisa Hellerstein. [2005] On compression based text classification. Advances in information retrieval in Advances in information retrieval, pages 300–314.

[4] Teahan W, and Harper D. [1998] Using compression based language models for text categorization. Proceedings of 2001 workshop on language modeling and information retrieval.

[5] Frank E, Cai C, Witten H. [2000] Text Categorization using compression models. In proceedings of DCC-00, *IEEE Data compression conference*.

[6] Clemens S and Frank P. [2006] Low complexity compression of short messages. *In proceedings of IEEE Data Compression Conference*, 123–132.

[7] Snel V, Plato J., and Qawasmeh E. [2008] Compression of small text files. *Journal of Advanced Engineering Informatics Information Achieve*, 20: 410–417.

[8] Khurana U and Koul A. [2005] Text compression and superfast searching. Proceedings of the CoRR, 2005.

[9] Moura E, Ziviani N and Navarro and Yates RB. [1998] Fast searching on compressed text allowing errors. Proceedings of the 21st annual international ACM Sigir conference on Research and Development in Information retrieval, pages 298– 306.

[10] Nieves G, Brisaboa, Eva L, and Param J.[ 2003] An efficient compression code for text databases. Proceedings of the 25th European conference on IR research, pages 468–481.

[11] Azad AK, Ahmad S, Sharmeen R, Kamruzzana SM. [2005] An efficient technique for text compression. In proceedings of *International Conference on Information Management and Business*, 467–473.

[12] Horspool RN and Cormack GV. [1992] Constructing word based text compression of short messages. *In Proceedings of the IEEE Data compression conference*, 62–71.

[13] Hava O, Skrbek M, and Kordík P.[ 2013] Supervised two-step feature extraction for structured representation of text data. *Journal of Simulation Modelling Practice and Theory*, 33: 132–143.

[14] Rocha L, Mourao F, Mota H, T.Salles, MA Gonc-alves, and W.Meira. [2013] Temporal contexts:Effective text classification in evolving document collections. *Journal of Information Systems*, 38: 388–409.

[15] Meiling Wu, Shengyi J, Guansong and Limin Kuang. [2012] An improved k-nearest neighbour algorithm for text categorization. Expert Systems with Applications, 39:1503–1509.

[16] Zhao Y and Wang Y. [2012] Text Categorization Based on Emergency domain Words : A System Engineering View. *Journal of Systems Engineering Procedia*, 5: 8–14.

[17] Ajit Danti and SN Bharath Bhushan. [2013] Document Vector Space Representation Model for Automatic Text Classification. *In Proceedings of International Conference on Multimedia Processing, Communication and Information Technology*, Shimoga. pp. 338–344.

[18] Lewis DD and M Ringuette. [1998] A comparison of two learning algorithms for text classification. Proceedings of the 3rd Annual symposium on Document Analysis and Information Retrieval, pp. 81 – 93.

[19] Wanga S, D Li, L.Zhao and J Zhang. [2013] Sample cutting method for imbalanced text sentiment classification based on BRC. *Journal of Knowledge-Based System* 37: 451–461.

[20] Patra A, Singh D. [ 2013] Neural Network Approach for Text Classification using Relevance Factor as Term Weighing Method. *International Journal of Computer Applications*, 68(17): 37-41.

## ABOUT AUTHORS

***Prof. Ajit Danti** is currently working as professor in Kingdom of Saudi Arabia. He is currently faculty of Department of Computer Science, College of Computer Science, King Khalid University; He received his PhD degree from Gulbarga University, Gulbarga, India. His research interest includes Digital Image Processing, Face Recognition.*

***Mr. SN Bharath Bhushan** is currently working as assistant professor in Sahyadri Engineering of College and Management Adyar, Mangaluru, Karnataka, India. He is currently faculty in the department of Computer Applications. He received his MS degree from University of Mysore, Mysore, India. He is pursuing PhD in Text Mining. His research interest includes Text Data Mining and Image processing.*

THE IIOAB JOURNAL

ARTICLE OPEN ACCESS

# STUDENT EVALUATION MODEL USING BAYESIAN NETWORK IN AN INTELLIGENT E-LEARNING SYSTEM

**Baisakhi Chakraborty and Meghamala Sinha***

*Department of Information Technology, National Institute of Technology, Durgapur, INDIA*

## ABSTRACT

*An Intelligent Tutoring System (ITS) is a type of knowledge based system whose main agenda is to efficiently substitute a human tutor by a machine. Unlike the traditional classroom teaching ways, an ITS has the ability to fit according to the necessity of each student. More and more emphasis is laid on different types of e-learning systems now days. In this paper a probability based ITS system is proposed consisting of four models namely student model, tutor model, domain model and a student evaluation model. The emphasis has been given on the student evaluation model where an element of uncertainty has been introduced and handled by Bayesian network. The purpose of the student evaluation model is to correctly detect the knowledge level of each student based on their response to questions. The uncertainty factor has been defined by terms guess and slip parameters. The two parameters are defined as follows: (a) Guess is the probability that a student of low intelligence gives a correct response to a difficult question whereas (b) Slip is the probability that a student of high intelligence gives an incorrect response to an easy question. During evaluation of the knowledge level of a student, we have incorporated the uncertainty factors of guess and slip with the help of Bayes' rule and have found desirable results that take into account the possibility of slippage or guess.*

**Received on:** 1st -Dec-2015
**Revised on:** 6th-Dec-2015
**Accepted on:** 20th-Dec-2015
**Published on:** 10th –Jan- 2016

KEY WORDS

*E-learning; Bayesian Network; intelligent tutoring system (ITS); personalized learning; student modeling*

***Corresponding author: Email:** meghamala.sinha@gmail.com

## INTRODUCTION

An Intelligent Tutoring System (ITS) is a special type of knowledge based system that replaces a human tutor by a machine which provides personalized tailored instructions and feedbacks to the user [1]. The major difference between the ITS and the traditional classroom is that ITS can fit according to the necessity of each students. It is impossible for a human tutor to cater to the needs for every student in a classroom. Other advantage of the intelligent tutoring systems is removal of time and space complexity of the real world unlike a regular classroom. It aims to provide a reformed education system.

### Framework of an ITS system

A typical intelligent tutoring system consists of four components:

1. The Domain Model
2. The Student Model
3. The Tutoring Model
4. The User Interface Model

The Domain Model or Expert Model contains the detailed description of what the expert user's knowledge consists of. This model contains a superset of all concepts, strategies, rules etc. That is, given a particular problem, the domain model contains all possible steps for its solution [2]. It acts as a reference to evaluate a student's knowledge level as he/she solves a problem.

The Student Model contains the description of the knowledge level of the student along with their misconception and knowledge gaps. It can be represented as an overlay on the domain model. This means that as a student solves a problem, his/her activity is traced according to the domain model in order to correctly identify the presence of the required knowledge in the student. Student Modeling is the most crucial task of the ITS.

The Tutor Model acts as a support to the students to help with their learning process. It takes input from both student model and domain model in order to provide recommendations and instructions to the student [3].
The User Interface Model acts as an interface between the ITS and the student logged in.

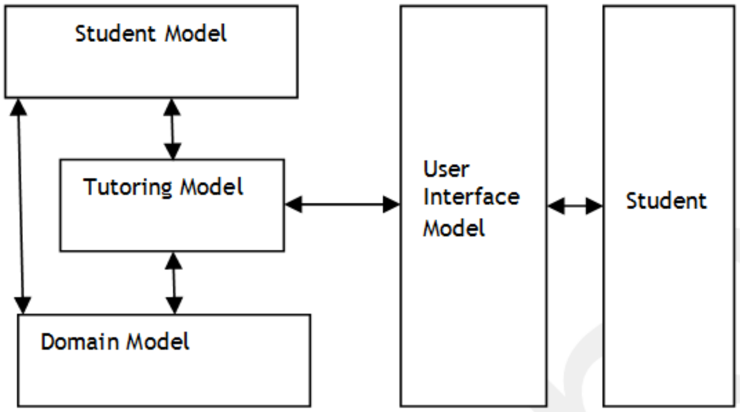


**Fig: 1. Typical ITS framework**

## Challenges in Existing ITS

The major challenge faced by e-learning systems, whose goal is to determine a students' knowledge about a particular domain, is that how much precisely and correctly it evaluates a student based on his/her responses. Even though a student provides a correct (or incorrect) response, it cannot be correctly concluded by the system what the true state of knowledge of the particular student really is. If a student whose past academic history is excellent and continuous assessment is good, but there are few mistakes is his response not expected from him, there is a high possibility that the student did a silly mistake in his/her answer. On the contrary, if a student with low continuous performance suddenly receives excellent grades, there is a possibility of guess which is a combination of random answering and adopting unfair means. We have taken into account such possibilities and have added a student evaluation model component that inputs the student current knowledge level and their response into a Bayesian Network to evaluate probabilities of guess or slippage. Then these probabilities are used to calculate the updated knowledge level of the student.

## Our Approach

Our ITS model allows a student logged in the system gain knowledge on concept/domains and particular subjects with the help of an e-tutor. The level of knowledge of a student at an instant is measured with the help of the student evaluation model. This model makes it possible for students with a pre-requisite knowledge level to access questions and answers them. Based on the responses to the question, this module provides a grade to the student. If the student achieves a grade above a particular threshold, then he/she is entitled to questions of higher knowledge level. Likewise his/her knowledge level is enhanced as he/she goes on solving the question in this ITS model. Based on his/her knowledge level, the student evaluation model also suggests study material which are accessible via the tutoring module. The underlying technology used is Bayesian Networks [4].

Bayesian Network is a compact and theoretically sound probabilistic graphical structures which are used as a tool for building a model to represent probability distribution over a given problem domain. It is a mathematically solid and efficient mechanism to provide insights on imprecise and uncertain information. In a Bayesian Network, any observable state or feature is represented by a node and any interdependencies among the features are represented by directed arcs. Each node is associated with a conditional probability table (CPT) representing probability of their occurrence conditions by their parent nodes. We have designed a Bayesian Network shown in **Figure– 4** to evaluate the probability of slip or guess for each student answering questions.

## RELATED WORK

Array As mentioned earlier e-learning systems have lots of advantages over a traditional tutor in a classroom [5]. Firstly, they are easily available over the internet hence overcoming time complexity as well as geographical complexity. Secondly E-learning systems are lot more attractive due to incorporation of rich multimedia and interactive features, such as, video, audio, animation etc. But there is a major scope of improvement in these static learning systems. It is the necessity of introduction of personalization by making the study materials more adaptive and interactive according to the requirement of the students [6].

Although there are several practical applications of Artificial Intelligence in the development of such systems, the best solution to this problem is a "one-size-fits-all" approach proposed by Brusilovsky and Maybury [7]. Such systems come with the advantage to adapt according to the users current knowledge level and skills. Using this idea Intelligent tutoring systems were build in order to assist students while solving problem [8]. It was also necessary to address the problem of handling uncertain or incomplete information while evaluation of users knowledge. Bayesian Network is one of the strongest probabilistic graphical models which can handle uncertain or imprecise data [9, 10, 11]. It has been applied in various applications like health, e-commerce, tutoring systems etc.
We will briefly review some of the approaches used in the past to develop student models using probabilistic methods [12].

OLAE (Martin and VanLehn.1995) which stands for "On-Line Assessment of Expertise or Off-Line Assessment of Expertise" is an assessment system that collects information about 'what a student knows' during problem-solving in introductory college physics [13]. This information is used by the teacher to make decisions. For each given problem, OLAE builds a Bayesian network that relates knowledge to actions. For example, what rules should be known in order to solve a particular equation. By using this network, the system knows whether a student knows and appropriately uses each of the rules.

POLA (Conati and VanLehn, 1996) stands for Probabilistic On-Line Assessment [14]. It provides a framework for modeling and assessment of student knowledge while they solve problem in introductory physics class. It differs from OLAE by applying probability reasoning to execute both knowledge and model tracing. In order to provide a compact representation of all available solution to a problem, an AND/OR graph is created. Then the Bayesian network is build incrementally. Finally the network provides the student's knowledge assessment regarding the physics problem.

HYDRIVE (Mislevy and Gitomer, 1996) has been developed to help simulate the important cognitive and relative features of troubleshooting on the flight line [15].The problem is given in a video where the pilot is seen describing some aircraft malfunction to the technician. The interface then offers the student several options for performing troubleshooting procedures with the help of online technical support material. The student's performance is thus evaluated based on how he utilizes the given information provided in the material. The students understanding are characterized in terms of dimensional variables which is represented and updated by a Bayesian network. Rule based inference also is an important part in this system.

ANDES (Conati et al.,1997) is an Intelligent Tutoring System for solving Newtonian physics problems via coached problem solving(VanLehn,1996) [16].This model takes help of Bayesian network for assessment of student knowledge, students' action prediction and plan recognition. An approximate algorithm is used to trace and update the network. Here the tutor proceeds along with the student, as he gives the correct response to a problem. Whenever there is the student come across a doubt, the tutor provides tailored hints that help the student to overcome his doubt. Each problem is associated with a solution graph. The Bayesian network is constructed automatically from the respectable solution graph and the corresponding conditional probably are updated. These networks are too large to be solved and the worst case of propagation in a Bayesian network is NP-hard.

The last decade there has been a growing interest among researchers to utilize the theoretically sound approach of Bayesian Network in user modeling systems, especially in the field of education [17]. The old Microsoft Office Assistant used the concept of Bayesian Network to provide users with suggestions and predictions. A survey of systems using Bayesian network paradigm to use or student modeling is given in [18] We have extended the ITS framework to introduce an evaluation model which uses Bayesian Network for mixing prior statistics and user inputs in order to for user knowledge evaluation.

## PROPOSED SYSTEM

The necessity of any ITS model to be modular is its ease to update the system when necessary. A simple change in the system does not require the need to reconstruct the whole model. This is the reason why a typical ITS system is modeled into different components.

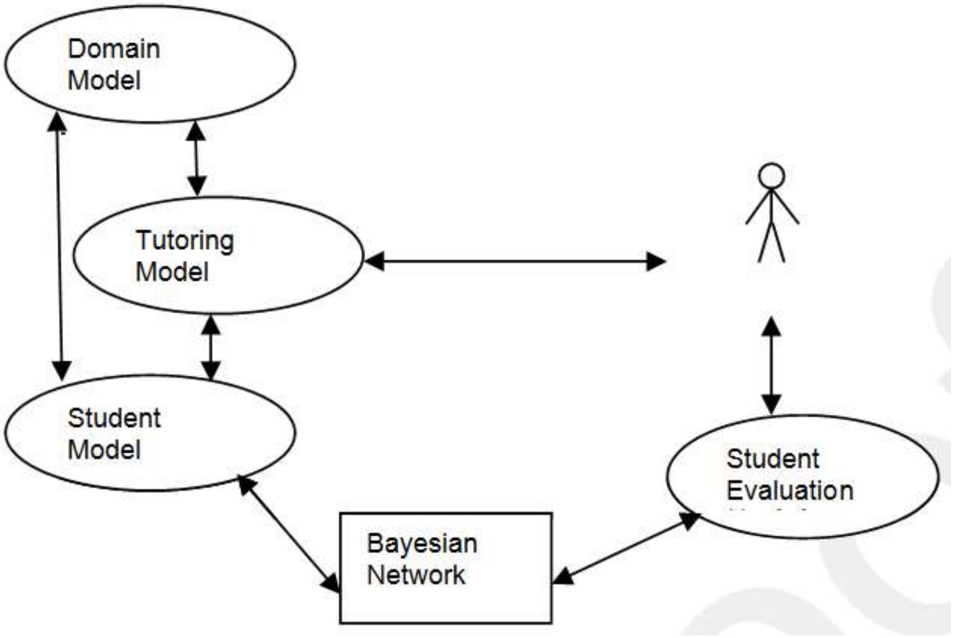


**Fig: 2. Proposed ITS framework**

……………………………………………………………………………………………………………..

Our proposed domain module contains the detailed structure and information for each course .It consists of the learning material of all the sub-topics of the course. Our domain model is designed in the form of a tree like structure. The particular course 'Database Management System' is used as for the domain content case study in the proposed system. It is divided into all the fundamental sub-topics, which are again sub-divided. Each of the leaf nodes represents a particular concept. Corresponding to each concept there is a set of question that will be used to evaluate student knowledge about that concept.

## Student Model

In our proposed system we need to initialize the knowledge level for each student before he/she starts the learning process [19]. We are able to determine the initial knowledge level of each student for each concept from the following flowchart [**Figure-3**].

Firstly a student logs in the system. Then he/she chooses from the available courses. A pre-evaluation test is performed which is used to determine the knowledge levels for each concepts. This is used to initialize the prior student database. The pre-evaluation test consists of a set of 20 questions. Each question is made up of a combination of several concepts. Here each of the concepts in the question has certain weight associated with it. Hence the response to each question is used to calculate the concept weights. This procedure gives us a certain idea about the initial knowledge level for each student.

Each student will be having a definite knowledge level for each concept. Once we finish the initial evaluation, all the calculated concepts are categorized into three levels poor, good and excellent. This is helpful to keep a well defined understanding of the current knowledge level for each concept so as to provide the right guidance to the student. The required study material corresponding to each concept level is provided to the students by an e-tutor so that he/she can improve themselves by learning.

## Study-Material Adaptation

After the pre-test a adaptation process combines the student model with the domain model to deliver appropriate course content to the student. Different kinds of adaptive technology are available. The proposed system uses the link hiding technique.

Link hiding is implemented by showing only relevant links that are suitable to the current knowledge level of the student and hiding irrelevant links from her/him. Link hiding protects the student from the complexity of all section links in a course and reduces her/his cognitive overload in hyperspace. The proposed system hides section links that have section knowledge levels lower than the student's overall knowledge level.

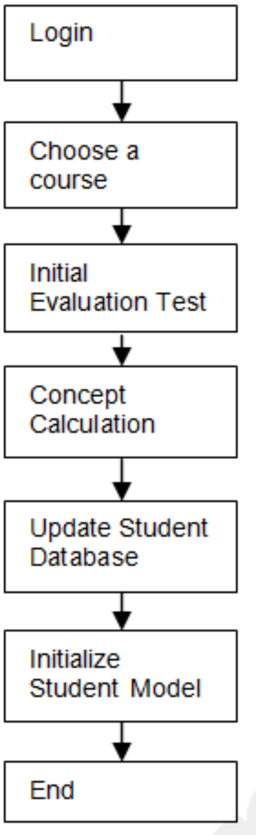


**Fig:3. Flowchart for initialization of student model**

## Student Evaluation Model

After the students have completed all the learning exercises, he/she has to undergo the final evaluation test which will recalculate the concept levels efficiently. Questions from different concepts are randomly picked and given to the students and their responses along with the previous knowledge levels are calculated using the concept of Bayesian network based on the Corbett and Anderson's Bayesian Knowledge Tracing model [20]. This is the key functionality of our Student Evaluation Model.

To explain this, we start by modeling a simple case, where a single question of a certain difficulty level is given by the system. In order to obtain information about the student's current knowledge about a domain, we use the student's answers or responses to given questions. But it is not necessary that a student gives a correct answer only if he/she has the knowledge related to the domain. Student can make random guess. On other hand given a student has knowledge about a domain he/she make mistakenly make a slip. This is known as unreliable information. So the system should consider all the evidences it has, including the student's response, to decide what is the current student knowledge. This requires reasoning under uncertainty which is handled by a belief network where the Bayesian estimate has been applied.

We compute the probability of a student's current knowledge about a particular domain using the fundamentals of Corbett and Anderson's knowledge tracing model.

i. Here we differentiate each question given to a student into particular levels of difficulty (QL) (from 0-1).
ii. All the students assumed to be already graded to a certain intelligent level (IL) (ranging from 0-1).
iii. In our proposed model a certain student appearing for the question either response correctly (C) or incorrectly.
iv. A student with lower knowledge level is more likely to give an incorrect response, but there is a slight chance he/she might answer correctly by making either a guess or cheating (called G).
v. On other hand a student with a higher knowledge level can also answer incorrectly, by making a silly mistake or slip (called S)

vi. Let Ki-1 be the initial knowledge probability of a student before answering a question, either correctly or incorrectly. This parameter will be dynamically updated for each response.
vii. The student also has a probability of learning a skill (called L), while answering a question.

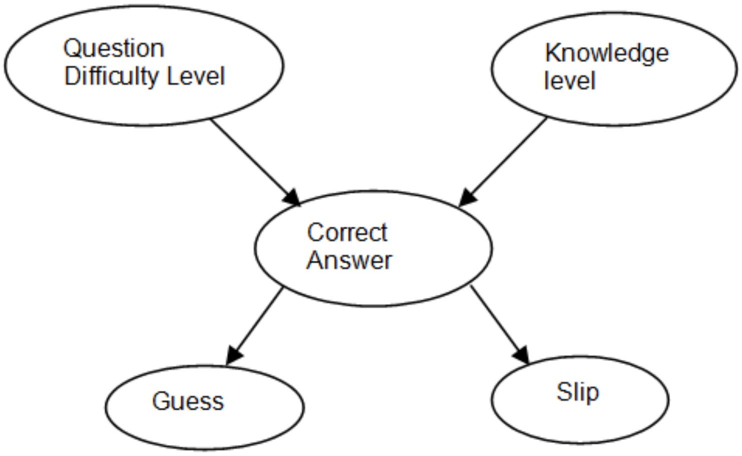


**Fig:4. Bayesian Network Construction**

………………………………………………………………………………………………………...

Hence we compute equations to compute student's knowledge from Bayesian Knowledge Tracing [21]

$$P(K_{i-1}|C_N) = \frac{P(K_{i-1})(1 - P(S))}{P(K_{i-1})(1 - P(S)) + (1 - P(K_{i-1}) P(G)}$$

Equation 1.1

$$P(K_{i-1}|\sim C_N) = \frac{P(K_{i-1})P(S)}{P(K_{i-1})( P(S)) + (1 - P(K_{i-1}) (1 - P(G))}$$

Equation 1.2

As we already mentioned the all the students are already has an initial knowledge probability which is to be re-evaluated and updated, taking into account the response given to a question by help of Bayesian rule. The updating is done as follows:

After we receive the output post the nth interaction the appropriate changes in the student knowledge at nth time can be incorporated for this we have to determine whether a student is in the learned state or not. We take into account two possible cases for this:

$P\ (K_i \mid K_{i-1}, answer)$

This probability denotes being in learned state given the student was already in the learned state.

$P\ (K_i \mid \sim K_{i-1}, answer)$

This probability denotes being in learned state given the student was not in the learned state.
Adding the two above mentioned cases we get the probability of learning of each student:

$$P\ (K_i \mid answer) = P\ (K_i \mid K_{i-1}, answer) * P\ (K_{i-1} \mid answer) + P\ (K_i \mid \sim K_{i-1}, answer) * P\ (\sim K_{i-1} \mid answer)$$

Equation 1.3

## EXPERIMENTATION AND RESULTS

We evaluated 15 students, taking 5 students each from 3 levels of a certain concept from the system. Next they are tested on question of different difficulty levels and each time their results are recorded.

Table: 1. Response record from experiment

| Student id | Knowledge level of a concept | Question level | Response |
|---|---|---|---|
| IT/01 | 0.10 | High | Correct |
| IT/02 | 0.37 | Average | Incorrect |
| IT/03 | 0.23 | Low | Correct |
| IT/04 | 0.19 | Low | Correct |
| IT/05 | 0.31 | High | Incorrect |
| IT/06 | 0.43 | High | Incorrect |
| IT/07 | 0.56 | Average | Incorrect |
| IT/08 | 0.67 | Low | Correct |
| IT/09 | 0.49 | Low | Correct |
| IT/10 | 0.51 | High | Incorrect |
| IT/11 | 0.78 | High | Correct |
| IT/12 | 0.88 | Average | Correct |
| IT/13 | 0.90 | Low | Incorrect |
| IT/14 | 0.70 | Average | Correct |
| IT/15 | 0.95 | High | Incorrect |

### Anomalies and Correction

We can consider some anomalies from above data. For example by considering an extreme case of student with id IT/01.Here the student is of low intelligent level and is given a high difficulty level question which he answers correctly. So there is chance that he might have either guessed the answer or have taken some unfair means to do so. In case of a traditional case we have to wrongly assign a high marks to this student. But in our system we have incorporated some probability of guessing with the help of Bayesian network to impute some degree of correction.

Again, by considering another extreme case of student with id IT/13.Here the student is of high intelligent level and is given a low difficulty level question which he answers wrongly. So there is chance that he might have either made a silly error or have wrongly imputed the answer. In case of a traditional case we have to wrongly assign a low marks to this student. But in our system we have incorporated some probability of slip with the help of Bayesian network to impute some degree of correction.

The calculation for the above two cases have been shown in the following section.

## Calculation of results based on Bayesian Network

**Table: 2.Conditional Probability Table for Guess parameter.**

| Question Difficulty Level | Intelligence Level | Correct Response | Probability<br>(P(G=T\|IL,C,QL) |
|---|---|---|---|
| High | 1 | True | 0.88 |
| High | 1 | False | - |
| High | 2 | True | 0.76 |
| High | 2 | False | - |
| High | 3 | True | 0.35 |
| High | 3 | False | - |
| Low | 1 | True | 0.43 |
| Low | 1 | False | - |
| Low | 2 | True | 0.23 |
| Low | 2 | False | - |
| Low | 3 | True | 0.12 |
| Low | 3 | False | - |
| Average | 1 | True | 0.73 |
| Average | 1 | False | - |
| Average | 2 | True | 0.67 |
| Average | 2 | False | - |
| Average | 3 | True | 0.29 |
| Average | 3 | False | - |

**Table: 3.Conditional Probability Table for Slip parameter.**

| Question Difficulty Level | Intelligence Level | Correct Response | Probability<br>(P(S=T\|IL,C, QL) |
|---|---|---|---|
| High | 1 | True | - |
| High | 1 | False | 0.008 |
| High | 2 | True | - |
| High | 2 | False | 0.05 |
| High | 3 | True | - |
| High | 3 | False | 0.22 |
| Low | 1 | True | - |
| Low | 1 | False | 0.08 |
| Low | 2 | True | - |
| Low | 2 | False | 0.76 |
| Low | 3 | True | - |
| Low | 3 | False | 0.89 |
| Average | 1 | True | - |
| Average | 1 | False | 0.23 |
| Average | 2 | True | - |
| Average | 2 | False | 0.42 |
| Average | 3 | True | - |
| Average | 3 | False | 0.6 |

Showing calculation of student knowledge for students with id (IT/01) with intelligence level 0.01(Low Intelligence Level) who is given a question with high difficulty level given that he/she answers it correctly:

$$P(K_{i-1}|C_n) = \frac{P(K_{i-1})(1-P(S))}{P(K_{i-1})\big(1- P(SLIP)\big)+ (1-P(K_{i-1})) P(G)}$$

$$= \frac{0.1}{0.1+0.9*0.73}$$

$$= \mathbf{0.13} \quad \textbf{(Low Intelligence Level)}$$

In a traditional system this student will be graded a high score without taking into any account of whether he/she has performed any guess or unfair mean.

Showing calculation of student knowledge for students with id (IT/13) with intelligence level 0.09 (High Intelligence Level) who is given a question with low difficulty level given that he/she answers it incorrectly:

$$P(K_{i-1}|{\sim}C_n) = \frac{P(K_{i-1})P(S)}{P(K_{i-1})\big( P(SLIP)\big)+ (1-P(K_{i-1})) (1-P(G))}$$

$$= \frac{0.9*0.89}{0.9*0.89+0.1}$$

$$= \mathbf{0.889} \quad \textbf{(High Intelligence Level)}$$

In a traditional system this student will be graded a low score without taking into any account of whether he/she has performed any slippage or silly mistake.

## CONCLUSION AND FUTURE WORK

In this paper we have proposed an Intelligent Tutoring System which has the potential to enhance the traditional e-Learning systems by incorporating reasoning based probabilistic approach namely Bayesian Network to evaluate a student's knowledge. The aim of this system is to provide a student the correct feedback at the correct time.

We have seen from our experiment results that evaluation of current knowledge level of each student is corrected by incorporating both the Guess and Slip parameter in the Bayesian network. Hence in case of a highly intelligent person performing a slippage, their knowledge level is reduced by a small percent. Also in case of a low intelligent person performing a guess, their knowledge level is increased by a small percent. Since experiment was conducted on 15 students, our aim is to increase the number of students from various backgrounds in order to obtain a wide range of probabilities. A large set of data will give us a more convincing result and will also enhance the validity of our proposed system. But such an attempt will also give rise to the challenge of managing a large number of data. We need to device a mechanism to handle the computation complexity of Bayesian Network for large data input.

The other major directions for future work regarding our system include the following:

- Enhancing the study material recommendation feature of the proposed system with advanced adaptive hypermedia technology (AHS).
- Evaluate individual students learning style from their performance and usage of study materials.
- Creating a more adaptive tutor model which will maintain an estimate of all the probabilities and the learning styles and present more individualized exercises and feedback for improvement of the student.
- Extending the proposed Bayesian network to a Dynamic Bayesian Network (DBN) which can update student's knowledge over long time spans.

## ACKNOWLEDGEMENT

None

## CONFLICT OF INTEREST

Authors declare no conflict of interest.

## FINANCIAL DISCLOSURE

No financial support was received to carry out this project

## REFERENCES

[1] Joseph Psotka, Sharon A. Mutter. [1988] Intelligent Tutoring Systems: Lessons Learned. Lawrence Erlbaum Associates. ISBN 0-8058-0192–8.

[2] Nkambou R, Mizoguchi R, Bourdeau J. [2010] Advances in intelligent tutoring systems. Heidelberg: *Springer*.

[3] Anderson H, Koedinger M. [1997] Intelligent tutoring goes to school in the Big City. *International Journal of Artificial Intelligence in Education*, 8: 30–43.

[4] 4 Millán, Eva, and José Luis Pérez-De-La-Cruz.[ 2002] A Bayesian diagnostic algorithm for student modeling and its evaluation. User Modeling and User-Adapted Interaction 12.2-3: 281–330.

[5] Butz, Cory J, Shan Hua, and R Brien Maguire. [2006]A web-based bayesian intelligent tutoring system for computer programming. Web Intelligence and Agent Systems 4.1: 77–97.

[6] P Brusilovsky.[1999] Adaptive and intelligent technologies for Web-based education, *Special Issue on Intelligent Systems and Teleteaching*, 4: 19–25.

[7] P Brusilovsky and MT Maybury. [2002] From adaptive hypermedia to adaptive Web, *Communications of the ACM, Special Issue on the Adaptive Web*, 45(5): 31–33.

[8] C Liu, L Zheng, J Ji, C Yang and W Yang. [2001]Electronic homework on the WWW, in: Proceedings of First Asia-Pacific Conference on Web Intelligence, Maebashi City, Japan, , pp. 540–547.

[9] J Pearl. [1988] Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann.

[10] SKM Wong and CJ Butz.[2001] Constructing the dependency structure of a multi-agent probabilistic network, *IEEE Transactions on Knowledge and Data Engineering*, 13(3) :395–415.

[11] SKM Wong, CJ Butz and D Wu. [2000]On the implication problem for probabilistic conditional independency, *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 30(6) :785–805.

[12] JM Agosta E Millan and JL Perez de la Cruz.[ 2001] Bayesian student modelling and the problem of parameter specification. British *Journal of Educational Technology*, 32 (2)

[13] Kurt VanLehn and pages=179-221 year=1998-22 Joel Martin title=Evaluation of an assessment system based on Bayesian student modeling, journal=*International Journal of Artificial Intelligence in Education*

[14] Cristina Conati and Kurt VanLehn Pola.[ 1996] A student modeling framework for probabilistic on-line assessment of problem solving performance. Proceedings of the Fifth International Conference on User Modeling.

[15] Linda S Steinberg and Drew H Gitomer.[ 1996] Intelligent tutoring and assessment built on an understanding of a technical problem-solving task. *Instructional Science*, 24.3:223{258 }.

[16] Cristina Conati.[ 1997] On-line student modeling for coached problem solving using bayesian networks. *User Modeling. Springer Vienna*.

[17] Chrysafiadi K and Virvou M.[ 2013]] Student modelling approaches: A literature review for the last decade. *Expert Systems with Application*s, 40(11): 4715–4722

[18] Jameson Anthony. [1995]Numerical uncertainty management in user and student modeling: An overview of systems and issues.*User Modeling and User-Adapted Interaction* 5.3-4: 193–251.

[19] Esichaikul, Vatcharaporn, Supaporn Lamnoi, and Clemens Bechter. [2011]Student modelling in adaptive e-learning systems. Knowledge Management & E-Learning: *An International Journal (KM&EL)* 3.3: 342–355.

[20] ATCorbett and JR Anderson.[ 1995] Knowledge tracing: modelling the acquisition of procedural knowledge. *User modelling and user-adapted interaction* 4:253–278.

[21] D Baker, Ryan SJ, Albert T Corbett, and Vincent Aleven.[ 2008] More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. *Intelligent Tutoring Systems*. Springer Berlin Heidelberg.

THE IIOAB JOURNAL

ARTICLE OPEN ACCESS

# A STUDY ON IOT ENABLED SMART STORE

**Ramesh S Nayak[1], Shreenivas Pai N[2], Akshay Nayak[3], Akhil Simha N[4]***

*[1]IS&E Dept, Canara engineering College, Benjanapadavu, Mangalore, INDIA*
*[2]ECE Dept, M. S. Ramaiah Institute of Technology, Bangalore, INDIA*
*[3]ISE Dept, M. S. Ramaiah Institute of Technology, Bangalore, INDIA*
*[4]EIE Dept, M. S. Ramaiah Institute of Technology, Bangalore, INDIA*

## ABSTRACT

*Most often, the store owners face the problems of refilling the stock in the store sporadically. In this paper, we present a feasibility study that leverages the Internet of Things (IoT) technology to make a store "smart". The ideology of the smart store is to notify the Store Owner about the stock and various other requirements through an application in their phone, which are explained in detail. This enables the Store owner to notify his supplier to refill the stock. From the supplier end, the supplier will get a message regarding the stock fulfillment. Customers can search their required products in the mobile application, select the product, quantity, delivery option, location etc. The app shows the user about the current status of the store (store is open or closed), the quantity available in the store and other details. Further development of this approach could lead to a complete change in our day-to-day shopping experience. The manuscript not only aims the idea of e-commerce but also includes the idea that can be implemented if the customer visits the store and the idea of smart trolley. The Smart Trolley includes indoor positioning system and other additional features.*

**Received on:** 20th-Nov-2015
**Revised on:** 22nd- Jan-2015
**Accepted on:** 20th- Dec-2015
**Published on:** 15th - Jan-2016

**KEY WORDS**

*Internet of things, smart store, smart trolley, mobile application, smart trolley*

***Corresponding author: Email:** akhilsimha1059@gmail.com **Tel:** +91-9916913776

## INTRODUCTION

An Recent advancement in the technology has led everything in this world to go and connect to the Internet. The internet of things (IoT) is the novel paradigm which has rapidly spread in the scenario of the emerging modern wireless communication [1]. Unquestionably, the main strength of the IoT idea is the high impact it will have on several aspects of everyday-life and behavior of potential users [2]. Supermarkets have been self-serving, where the shoppers select their required items and proceed towards the billing counters. In this Smart Store concept, we present how Internet can help the Store Owner and also to the customer to access various features of the store from just an application.

With little to none assistance, locating the shopping items in a big, sometimes mazelike, store can be very time-consuming, physically exhaustive and mentally frustrating [3]. The idea of supermarket combined with the internet facility is called as Smart Store. It uses the data that is being collected in cloud via Internet to perform the necessary analysis of the demand and need in the store. Refilling of the stock in the smart store is done with the help of Smart Shelf, which is concept of Smart Trolley. Other features may also be added.

**Figure-1** shows the connection between the three main members who are associated with the store; the dealer or the factory representative, the store owner and the customer or shopper.

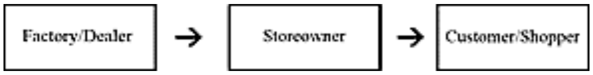


**Fig: 1. Connections**

The paper presents the idea of Smart Store from three people's viewpoint and how Internet will help each one of them. This paper also includes the idea of implementing smart trolley in the stores.

## STORE OWNER

The Store Owner will be provided with an app, where he will be notified about the location/place of the product in the store and its details such as quantity, cost, demand, expiry date, etc. A brief description has been provided in the following sections:

### Stock Refilling Notification

In the current situation, the Store Owner has to manually check the stock and notify the dealer or the factory representative for the need of stock. But in a Smart Store, a Smart Weighing Scale is used to measure the weight of the products in a Smart Shelf. The complete weight of an individual shelf gives the total weight of the products, while dividing them with an individual product weight will give the total quantity available.
Total / Individual = Number
Total: Total weight of the shelf including the product
Individual: Weight of the individual product
Number: Number of Product left in the stock.

Every product has a certain weight, which can be measured. For certain products the weight may be low, so a precision Smart Scale has to be utilized. The Smart Scale is connected to the cloud service via Wi-Fi, enabling it to automatically update the quantity in the store as well as the app. If the quantity in the shelf reaches a minimum threshold the owner gets the notification about the stock requirement.

Smart Shelf consists of Smart Weighing Scale. It is connected to the Internet via Wi-Fi. The shelf generally consists of three rows. Third Row is for the stack (case) of products.

**Fig: 2. Smart Shelf**

## RELATED WORK

As If the customer wants to buy a complete set of the individual product, then he can select the box from this shelf. Second Row is for the individual products, which may be picked by the customer. First Row is for the products, which are about to reach the expiry date. Owner of the store may put a discount on these items. The Smart Scale measures the weight of each shelf.

### Analytics

As the quantity of the shelfis are available, the same date is updated in the cloud-based service via Internet. An algorithm is written to notify the owner about the most demanding product and non-demanding product. If a product is of more demand, then the shelf has to be always filled with product. In order to achieve this, lower threshold to notify about the requirement has to be raised up. If the stock is not in demand, the owner should be able to put a discount on the product to make sure that the product continues to be in the sale.

### Expiry Date Notification

When the dealer delivers the product to the store, the owner can scan the barcode present on it and enter the details regarding the product on the cloud. This will include the expiry date of the product also. If the product present in the shelf is about to reach the expiry date, the owner gets a notification regarding the date. Then the owner can shift the product to the First Row and apply some discount in order to attract the customer to buy the product.

### Product Location

As the smart scale used in the system is connected to the Internet, it can also be utilized to locate the product in the store. Inside the store, facility is also provided in the trolley to locate the product. The consumer can also locate the product in the nearby store by using the app in their smartphone.

Information Terminal can be placed at various places of the store to avail the location of a product if the customer/user is not having smartphone.

## CONSUMER

The consumer will be provided with the app of all the smart stores registered. The consumer can open the app and search for his required product. If the customer specifies the name of the brand, other details then the product will be displayed directly on the smart phone screen. If a general name is mentioned, then all the relevant products will be displayed. User then needs to enter the quantity required which will then filter all the stores nearby with the quantity available. The app will also be able to display whether the store is open or closed depending on the response sent by Smart Door of the shop connected to the cloud via Wi-Fi network. If there are many stores available with the requirement, the consumer will be asked to select the store or the product depending on their convenience.

After this the consumer can continue shopping or go for the payment screen, depending on the needs. Finally, the user need to opt for delivery options such as home delivery or pickup from the store.

In an outdoor environment, the Global Positioning System (GPS) works efficiently in positioning and targeting different types of entities [4]. The app and sore is GPS enabled, which will automatically help both the customer and the store to locate each other.

## DEALERS

Dealer will receive request orders from the associated stores. If there is any change in cost of any of the commodity, a cross check message is sent to the store owner. After confirmation of the order and payment, the app shows the nearest route to deliver the goods to the stores office using the GPS technology. A confirmation message will be sent to both dealer and Store Owner after the delivery is successful.

## SMART TROLLEY

Almost every customer uses a trolley. In order to reduce the time, the customer just has to scan the product in the trolley using the barcode scanner provided. It also consists of a small display screen where various details regarding the store or the product will be displayed.

In-Store mapping system can be utilized using the Display screen present in the trolley. The Indoor Positioning system automatically locate the store items and send the location information to the display screen of the trolley via cloud networking.

**Fig: 3. Smart Trolley**

The working of the smart trolley has been explained in detail below:

- The customers take a trolley and enter the shopping area. The trolley gets registered at the owner's desk.
- When the customer selects a product, they need to scan the barcode using the barcode scanner present in the trolley, which is connected, to the cloud.
- As soon as the product gets added, the trolley compares the weight of the goods present in trolley with the database present in the cloud.
- The display screen enables the customer to see all the products that are added in the trolley.
- After the completion of shopping, the customer can then push the trolley to the exit zone of the store and can swipe their bankcard using the Swiping machine, which can also be added next to the display screen in the trolley.

If any customer enters the billing zone of the store and the weight of the trolley doesn't match with the products list in the screen, a notification to the security will be sent, where the security can take the necessary action.

## IMPLEMENTATION

The web technologies have since enabled the fusion of digital and real worlds by providing a simple yet intuitive platform to interact with cloud services and virtual reality platforms [5]. The basic architecture of the system is presented in the **Figure–4**.

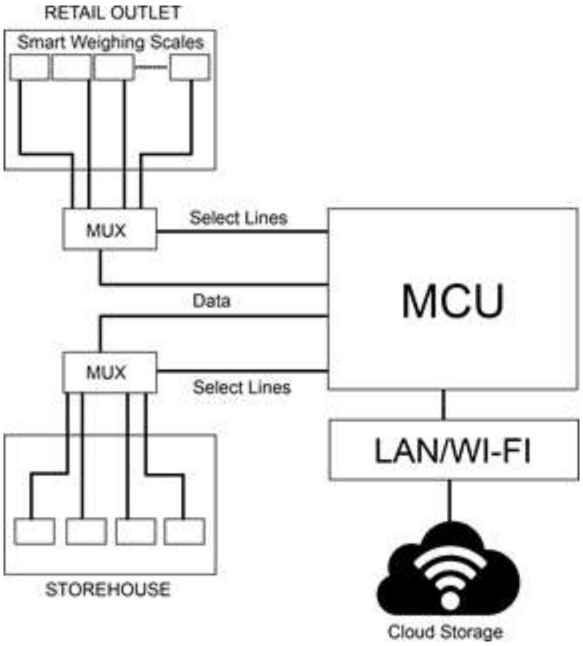


**Fig: 4. Architecture**

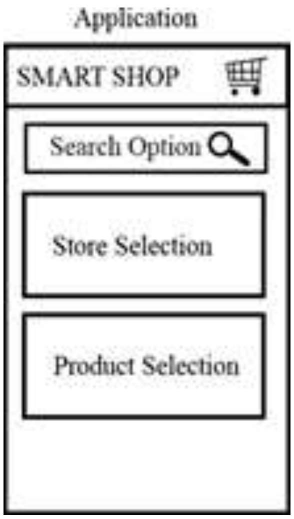


**Fig: 5. Basic template of an application.**

The idea involves the usage of a simple microcontroller (such as TI SimpleLink Wi-Fi CC3200 LaunchPad) which requires less power and is easily available [6]. A rectenna for the harvesting of electromagnetic energy associated to the European RFID band has been utilized [7]. The MCU should be able to access the Internet at every point of the time. Data acquisition is to collect data by sensors or other measurement equipment [8]. The microcontroller should be capable of updating the data in the Cloud storage via a LAN or Wi-Fi network. Smart Weighing scale is connected in the similar manner to the internet.

Application differs accordingly with the kinds of user interfaces. It consists of three modules, one for the customer, one for the Store Owner and other for dealer. The app is equipped with GPS in order to trace the location of the customer.  A separate login id will be provided to various users, like owner, dealer and the customer.

## RESULTS

The 89V51 microcontroller is used in the system. The trolley contains a barcode scanner for billing purpose. The microcontroller is connected to the barcode scanner using MAX232IC. An on-board ultra-low-power MCU manages the sensor data sampling and the wireless communication by means of a new generation UHF I2C-RFID chip whose EPC code is dynamically updated with actual sensor measurements [9]. There is a quantity mismatch detector which is 38 KHz IR trans-receiver. The cost of the product and quantity is displayed using LCD display. Any product chosen by the customer need to be scanned by the bar coder in the trolley. After scanning the barcode, customer do have the option to delete the item/product from the list. A quantity mismatch will occur if the product purchased is directly kept in the trolley rather than scanning the barcode.

The system is tested in the super bazar store and experiments are conducted for number of items/products. The system found to be functioning satisfactorily.

## ADVANTAGES OF THE PROPOSED SYSTEM

- The implementation of this idea will reduce a lot of human efforts and human intervention as well.
- In the proposed system shop owner need not be concerned about the demand and supply balance of the store as the data regarding the product gets updated in the cloud.
- Shop owner need not be concerned about the stock coming and going out as all the updated details are available in the cloud at any moment.
- Customers can access the real time information of the store and the products available in it.

- With help of this app, the dealer or the factory representative gets an efficient way of collecting different data of the product from the registered stores and supply the goods when needed or on demand.
- The customer need not wait in the queue for billing. The bill is automatically generated as the Smart Trolley is enabled with a barcode reader.

## CONCLUSION

Many applications are proposed having online facility with various stores and other details, but none of them provides the facility to search the product in the current and nearby stores. This paper presents a novel approach for smart shopping along with Smart Scale which incorporates the idea of Smart Trolley. The app is very helpful in saving precious time. The idea has the feasibility and advancement that can be done in a normal store to enhance it to a Smart Store.

## CONFLICT OF INTEREST

Authors declare no conflict of interest.

## FINANCIAL DISCLOSURE

No financial support received to carry out this research.

## REFERENCES

[1] Asghar MH, Mohammadzadeh N, Negi A. [2015] Principle application and vision in Internet of Things (IoT), in Computing, Communication & Automation (ICCCA), 2015 International Conference on, vol., no., pp.427–431, 15–16 May.

[2] Luigi Atzori, Antonio Iera, Giacomo Morabito [2010] The Internet of Things: A survey, Computer Networks, Volume 54, Issue 15, 28 October 2010, Pages 2787–2805, ISSN 1389-1286.

[3] Hicks D, Mannix K, Bowles HM, Gao B J. [2015] SmartMart: IoT-based In-store mapping for mobile devices, in Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom), 2013 9th International Conference on , vol., no., pp.616-621, 20-23 Oct.

[4] Al Nuaimi, K Kamel, H. [2011] "A survey of indoor positioning systems and algorithms," in Innovations in Information Technology (IIT), 2011 International Conference on, vol., no., pp.185–190, 25–27 April

[5] Pereira PP, Jens E, Rumen K, Jerker D, Asma R, Mia J. [2013] Enabling cloud connectivity for mobile internet of things applications. In: Proceeding of the IEEE 7th international symposium on service oriented system engineering (SOSE), pp 518–526.

[6] Asghar MH, Mohammad zadeh N, Negi A. [2015] Principle application and vision in Internet of Things (IoT), in Computing, Communication & Automation (ICCCA), 2015 International Conference on, vol., no., pp.427–431, 15–16 May.

[7] Hicks D, Mannix K, Bowles H M, Gao B J. [2015] SmartMart: IoT-based In-store mapping for mobile devices, in 9th International Conference on Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom), vol., no., pp.616-621, 20-23 Oct. 2013.

[8] Al Nuaimi, K Kamel, H. [2011] "A survey of indoor positioning systems and algorithms," in Innovations in Information Technology (IIT), 2011 International Conference on, vol., no., pp.185–190, 25–27 April

[9] Luigi Atzori, Antonio Iera, Giacomo Morabito.[ 2010] The Internet of Things: A survey, Computer Networks, Volume 54, Issue 15, 28 October 2010, Pages 2787–2805, ISSN 1389-1286.

## ABOUT AUTHORS

***Prof. Ramesh Nayak** is currently working as associate professor in Canara Engineering College, Benjanapadavu, Mangaluru, Karnataka, India. He is currently faculty in the department of Information Science and Engineering. He received his M.Tech degree from University of Mysore, Mysore, India. He is pursuing PhD in image processing. His research interest includes Image processing, Data Mining, Computer Networks. He has teaching experience of 14 years and research experience of 3 years. He has published research papers in national, International conferences and Journals.*

THE IIOAB JOURNAL ISSN: 0976-3104

***Shreenivas Pai*** *is currently an engineering student of Electronics and Communication at M.S.Ramaiah Institute of Technology, Bangalore. Karnataka, INDIA.*

***Akshay Nayak*** *is currently an engineering student of Information Science and Engineering at M.S.Ramaiah Institute of Technology, Bangalore, Karnataka, INDIA.*

***Akhil Simha*** *N is currently an engineering student of Electronics and Instrumentation at M.S.Ramaiah Institute of Technology, Bangalore, Karnataka, INDIA.*

THE IIOAB JOURNAL

ARTICLE OPEN ACCESS

# A STUDY OF HUMAN ACTIVITY RECOGNITION USING ADABOOST CLASSIFIERS ON WISDM DATASET

**Kishor H Walse**[1*], **Rajiv V Dharaskar**[2], **Vilas M Thakare**[3]

[1]*Dept. of Computer Science and Engineering, Anuradha Engineering College, Chikhli, INDIA*
[2]*DISHA Technical Campus, Raipur, CG, INDIA*
[3]*Dept. of Computer Science and Engineering, S.G.B. Amravati Univeristy, Amravati, INDIA*

## ABSTRACT

*Human activity recognition is bringing much attention because of its applications in many areas like health care, adaptive interfaces and a smart environment. Today's smartphone is well equipped with advanced processor, more memory, powerful battery and built-in sensors. This provides an opportunity to open up new areas of data mining for activity recognition of Daily Living. In this paper, the benchmark dataset is considered for this work is acquired from the WISDM laboratory, which is available in public domain. We performed experiment using AdaBoost.M1 algorithm with Decision Stump, Hoeffding Tree, Random Tree, J48, Random Forest and REP Tree to classify six activities of daily life by using Weka tool. Then we also see the test output from weka experimenter for these six classifiers. We found the using Adaboost,M1 with Random Forest, J.48 and REP Tree improves overall accuracy. We showed that the difference in accuracy for Random Forest, REP Tree and J48 algorithms compared to Decision Stump, and Hoeffding Tree is statistically significant. We also show that the accuracy of these algorithms compared to Decision Stump, and Hoeffding Tree is high, so we can say that these two algorithms achieved a statistically significantly better result than the Decision Stump, and Hoeffding Tree and Random Tree baseline.*

**Received on:** 15th-Dec-2015
**Revised on:** 24th-Dec-2015
**Accepted on:** 26th– Dec-2015
**Published on:** 15th –Jan-2016

KEY WORDS

*Human Activity Recognition; Classifier; Sensor; Accelerometer; Boosting*

***Corresponding author: Email:** walsekh@acm.org, k.h.walse@ieee.org **Tel:** +91-9689271947 **Fax:** +91-7264-242063

## INTRODUCTION

In the world, the first time it is happening that the proportion of older persons ( 60 years or older) increases in the proportion of young (below 15). For the first time in history, the number of older persons in the world will exceed the number of young by year 2050. [1]. Such ageing population need care. Activity recognition is a significant research area can provide a solution to such problem. This area has many applications in healthcare, elder care, user interfaces, smart environments, and security [2,3]. Image and video based human activity recognition has been studied since a long time but they have limitation of mostly require infrastructure support, for example, the installation of video cameras in the monitoring areas [4]. There are alternative approaches are available such as a body worn sensors or a smartphone which have built-in sensors to recognize the human activity of daily living. But a normal human can't wear so many sensors on the body excluding a patient [5,6]. Today's smartphone is well equipped with powerful sensors and long lasting battery with small in size provide an opportunity for data mining research and applications in human activity recognition using smartphones. These smartphones having accelerometer, gyroscope, GPS, microphones, cameras, light, temperature, compasses and proximity [7]. Some existing works have explored human activity recognition using data from accelerometer sensors [8-10]. Many researches received very good accuracy by using tri-axial accelerometer for activity recognition the daily [11].

## RELATED WORK

In this paper, we reviewed the work done so far in the area of human activity recognition. We found many researchers [12,7,13,15] have worked on it. We discussed various aspects of these studies and their limitations. Some of these aspects included their experimental setup, dataset used, a sensor-selection, position of sensors, a sampling rate, windowing, a feature selection, classifier selection etc. JR Kwapisz, et al [7] tri-axial accelerometer is used with twenty-nine users. There are many research areas in this topic because it is related to human activity. There is wide scope in the direction to increase the usability of the smartphone. Researchers can make various

THE IIOAB JOURNAL ISSN: 0976-3104

research according to the need of a user, as this system is now occupying its position in the human healthcare and military department [14].

# MATERIALS AND METHODS

## Data Collection

In this paper, we have uses a standard HAR dataset which is publicly available from the WISDM group [6]. Android smartphone based application was used to collect data. Each user was asked to take the smartphone in a front leg pocket and performed five different activities in supervised condition which were walking, jogging, walking upstairs, walking downstairs, sitting, and standing. While performing these activities, the sampling rate for accelerometer sensor was kept of 20Hz. WISDM HAR dataset consists the accelerometer's raw time series data and detail descriptions is shown in the **Table– 1**.

**Table: 1. WISDM Dataset Description** [2]

| Description | Nos. of Record | % of Records |
|---|---|---|
| Total Nos. of Samples | 10,98,207 | 100% |
| Nos. of Attributes | 6 | |
| Any missing value | None | |
| **Ativity wise distribution** | **Total nos. of Samples** | **Percentage** |
| Walk | 4,24,400 | 38.6% |
| Jog | 3,42,177 | 31.2% |
| Up-stairs | 1,22,869 | 11.2% |
| Down-stairs | 1,00,427 | 9.1% |
| Sit | 59,939 | 5.5% |
| Stand | 48,395 | 4.4% |
| Transformed Examples | | |
| Total Nos. of samples | 5,424 | |
| Nos. of attributes | 46 | |
| Any missing value | None | |
| **Activity wise distribution** | **Total nos. of samples** | **Percentage** |
| Walk | 2,082 | 38.4% |
| Jog | 1,626 | 30.0% |
| Up-stairs | 633 | 11.7% |
| Down-stairs | 529 | 9.8% |
| Sit | 307 | 5.7% |
| Stand | 247 | 4.6% |

## Feature Generation

Before applying the classifier algorithm, it is necessary to transform the raw sensor's data. The raw accelerometer's signal consists of a value related each of the three axes. To accomplish this J.R. Kwapisz et al [7] has segmented into 10-second data without overlapping. This is because he considered that 10seconds data consist of sufficient recreations that consist of 200 readings. Then they have generated features that were based each segment data of 200 raw accelerometer readings. A total 43 features are generated. All these are variants are based on six extraction methods. Average, Standard Deviation, Average Absolute Difference and Time between Peaks for each axis are extracted. Apart from these Average Resultant Acceleration and Binned Distribution is also extracted.

## Classification

In this paper for classification of human activity of daily living, we have used the classifiers available in the Weka tool. In this paper , we have presented selected classifier algorithms like Decision Stump, Hoeffding Tree, Random Tree, REP Tree, J48 and RAndom Forest, decision tree algorithms along with Adaptive Boosting available in Weka Adaboost.M1 with default setting.

## Performance Measures

During this experimentation following performance measures has been used.
The Overall *accuracy* is used to summarize the overall classification performance for all classes. It is defined as follows:

$$\text{Overall Accuracy} = \frac{\text{TP}}{\text{TP+FP+FN+TN}} \qquad \text{....(1)}$$

The *precision* is defined as follows:

$$\text{Precision}= \frac{\text{TP}}{\text{TP+FP}} \qquad \text{....(2)}$$

The *recall*, also called sensitivity or *true positive rate*, is defined as follows: Sensitivity is used to relate the test's ability to identify a condition correctly.

$$\text{recall} = \frac{TP}{TP+FN} \qquad \text{....(3)}$$

The Specificity is defined as follows:

$$Specificity = \frac{TN}{TN + FP} \qquad \text{....(4)}$$

The *F-measure* combines precision and recall in a single value:

$$\text{F-measure}=\left(2\,\frac{\text{Precision*Recall}}{\text{PRecison+Recall}}\right) \qquad \text{....(5)}$$

Kappa statistic:
Cohen's kappa statistic, κ , is a measure of agreement between categorical variables X and Y. The equation for κ is:

$$\kappa = \frac{p_0 - p_e}{1 - p_e} \qquad \text{....(6)}$$

Mean Absolute Error (MAE) is defined as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|f_i - y_i\right| = \frac{1}{n}\sum_{i=1}^{n}\left|e_i\right| \qquad \text{....(7)}$$

RMS E :
Root Mean Square Error (RMSE) is also called as Root Mean Square Deviation (RMSD) is defined as

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(\bar{y}_i - y_i\right)^2}{n}} \qquad \text{....(8)}$$

MCC:
The Matthews correlation coefficient (MCC ) is

$$MCC = \frac{TP\times TN - FP\times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \qquad \text{....(9)}$$

## Experimental Method

This paper follows following steps to perform experiment with standard dataset.

- Acquisition of standard WISDM HAR Dataset for Human Activity Recognition through a mobile device which is available in public domain.
- Partitioning dataset into training, testing and cross validation by using 10-fold cross-validation.
- A Selection of Meta Adaboost.M1 classifier for classification with selected decision tree classifier with default parameters.
- Examination of each classification model on 10-fold cross validation.
- Comparative analysis on the basis of performance measures such as, classification accuracy, TP rate, FP rate, minimum RMSE, F-measure, precision, recall and ROC.
- We used experiment environment from weka in determining mean and standard deviation performance of a classification algorithm on a WISDM dataset.
- we choose decision tree classifiers, experiment type has been chosen as 10-fold cross-validation in which WISDM dataset is divided into 10 parts (folds) and compare their results with meta classifier Adaptive Boosting. The confidence kept at 0.05

# RESULTS AND DISCUSSION

Finally, we used weka experimenter to evaluate the performance of the classifiers mentioned in an earlier section on standard WISDM dataset. Each classifier is trained and tested using 10-fold cross validation with 10 times' repetition. In this section, the summary of the results are presented.

## Confusion Matrix for Classifiers

The Confusion Matrix for Decision Stump, Hoeffding Tree, Random Tree, REP Tree, J48 and Random Forest are shown in the Tables– 2– 7. As shown a confusion matrix in the Table– 2 and performance criteria in table 8 for Decision Stump, the classifier found confused over the Jogging stairs standing and Laying Down. Hoeffding Tree and Random tree as shown in the Tables– 3, 9, Tables– 4, 10 for respectively which are failed to classify the stairs' activity successfully. In confusion matrix the major misclassification denoted by yellow color. It is found that there is common misclassification of the stairs and sitting with walking has been observed. But still the performance of the REP Tree, J49 and Random Forest is much better compared with others.

**Table: 2. Confusion Matrix for Adaboost.M1 Meta Classifier with Decision Stump**

| classified as | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a = Walking | **1754** | 0 | 431 | 0 | 0 | 0 |
| b = Jogging | 117 | **0** | 0 | 13 | 0 | 0 |
| c = Stairs | 251 | 0 | **0** | 0 | 0 | 0 |
| d = Sitting | 49 | 0 | 0 | **1361** | 0 | 0 |
| e = Standing | 14 | 0 | 0 | 826 | **0** | 0 |
| f = Lying Down | 2 | 0 | 0 | 617 | 0 | **0** |

**Table: 3. Confusion Matrix for Adaboost.M1 Meta Classifier with Hoeffding Tree**

| classified as | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a = Walking | **2011** | 4 | 7 | 81 | 39 | 43 |
| b = Jogging | 1 | **122** | 0 | 0 | 4 | 3 |
| c = Stairs | 15 | 2 | **174** | 33 | 12 | 15 |
| d = Sitting | 25 | 5 | 1 | **1177** | 104 | 98 |
| e = Standing | 23 | 4 | 2 | 46 | **744** | 21 |
| f = Lying Down | 10 | 2 | 0 | 28 | 33 | **546** |

**Table: 4. Confusion Matrix for Adaboost.M1 Meta Classifier with Random Tree**

| classified as | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a = Walking | **2124** | 4 | 22 | 29 | 3 | 3 |
| b = Jogging | 3 | **121** | 1 | 2 | 2 | 1 |
| c = Stairs | 27 | 3 | **218** | 1 | 2 | 0 |
| d = Sitting | 24 | 1 | 1 | **1349** | 19 | 16 |
| e = Standing | 8 | 1 | 2 | 23 | **800** | 6 |
| f = Lying Down | 2 | 1 | 0 | 22 | 5 | **589** |

Table: 5. Confusion Matrix for Adaboost.M1 Meta Classifier with REP Tree

| classified as | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a = Walking | **2153** | 2 | 6 | 19 | 4 | 1 |
| b = Jogging | 4 | **120** | 1 | 2 | 2 | 1 |
| c = Stairs | 5 | 0 | **242** | 3 | 1 | 0 |
| d = Sitting | 23 | 0 | 1 | **1358** | 15 | 13 |
| e = Standing | 9 | 1 | 1 | 7 | **818** | 4 |
| f = Lying Down | 2 | 1 | 0 | 12 | 5 | **599** |

Table: 6. Confusion Matrix for Adaboost.M1 Meta Classifier with J48

| classified as | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a = Walking | **2166** | 1 | 5 | 9 | 2 | 2 |
| b = Jogging | 3 | **123** | 0 | 1 | 2 | 1 |
| c = Stairs | 17 | 0 | **234** | 0 | 0 | 0 |
| d = Sitting | 17 | 1 | 1 | **1371** | 15 | 5 |
| e = Standing | 6 | 2 | 1 | 2 | **827** | 2 |
| f = Lying Down | 2 | 2 | 0 | 13 | 6 | **596** |

Table: 7. Confusion Matrix for Adaboost.M1 Meta Classifier with Random Forest

| classified as | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a = Walking | **2170** | 0 | 2 | 9 | 4 | 0 |
| b = Jogging | 1 | **126** | 0 | 1 | 1 | 1 |
| c = Stairs | 7 | 0 | **244** | 0 | 0 | 0 |
| d = Sitting | 19 | 1 | 4 | **1365** | 15 | 6 |
| e = Standing | 7 | 1 | 1 | 5 | **826** | 0 |
| f = Lying Down | 2 | 2 | 0 | 9 | 4 | **602** |

## Performance Criteria for Classifiers

Table: 8. Performance Criteria for Adaboost.M1 Meta Classifier with Decision Stump

| Activity | TP-Rate | FP-Rate | Precision | Recall | F-Measure | MCC | ROC Area | PRC Area |
|---|---|---|---|---|---|---|---|---|
| **Walking** | 0.803 | 0.133 | 0.802 | 0.803 | 0.802 | 0.669 | 0.826 | 0.737 |
| **Jogging** | 0 | 0 | 0 | 0 | 0 | 0 | 0.731 | 0.048 |
| **Stairs** | 0 | 0 | 0 | 0 | 0 | 0 | 0.809 | 0.112 |
| **Sitting** | 0.965 | 0.469 | 0.419 | 0.965 | 0.584 | 0.444 | 0.741 | 0.408 |
| **Standing** | 0 | 0 | 0 | 0 | 0 | 0 | 0.721 | 0.248 |
| **Lying Down** | 0 | 0 | 0 | 0 | 0 | 0 | 0.721 | 0.187 |
| **Weighted Avg.** | 0.573 | 0.175 | 0.431 | 0.573 | 0.474 | 0.384 | 0.773 | 0.468 |

Table: 9. Performance Criteria for Adaboost.M1 Meta Classifier with Hoeffding Tree

| Activity | TP-Rate | FP-Rate | Precision | Recall | F-Measure | MCC | ROC Area | PRC Area |
|---|---|---|---|---|---|---|---|---|
| **Walking** | 0.92 | 0.023 | 0.965 | 0.92 | 0.942 | 0.905 | 0.989 | 0.984 |
| **Jogging** | 0.938 | 0.003 | 0.878 | 0.938 | 0.907 | 0.905 | 0.995 | 0.937 |
| **Stairs** | 0.693 | 0.002 | 0.946 | 0.693 | 0.8 | 0.802 | 0.968 | 0.813 |
| **Sitting** | 0.835 | 0.047 | 0.862 | 0.835 | 0.848 | 0.796 | 0.966 | 0.933 |
| **Standing** | 0.886 | 0.042 | 0.795 | 0.886 | 0.838 | 0.808 | 0.976 | 0.906 |
| **Lying Down** | 0.882 | 0.037 | 0.752 | 0.882 | 0.812 | 0.789 | 0.976 | 0.892 |
| **Weighted Avg** | 0.878 | 0.032 | 0.885 | 0.878 | 0.879 | 0.844 | 0.979 | 0.939 |

Table: 10. Performance Criteria for Adaboost.M1 Meta Classifier with Random Tree

| Activity | TP-Rate | FP-Rate | Precision | Recall | F-Measure | MCC | ROC Area | PRC Area |
|---|---|---|---|---|---|---|---|---|
| **Walking** | 0.972 | 0.02 | 0.971 | 0.972 | 0.971 | 0.952 | 0.977 | 0.957 |
| **Jogging** | 0.931 | 0.002 | 0.924 | 0.931 | 0.927 | 0.925 | 0.964 | 0.861 |
| **Stairs** | 0.869 | 0.005 | 0.893 | 0.869 | 0.881 | 0.875 | 0.938 | 0.792 |
| **Sitting** | 0.957 | 0.019 | 0.946 | 0.957 | 0.951 | 0.934 | 0.969 | 0.917 |
| **Standing** | 0.952 | 0.007 | 0.963 | 0.952 | 0.958 | 0.95 | 0.974 | 0.929 |
| **Lying Down** | 0.952 | 0.005 | 0.958 | 0.952 | 0.955 | 0.949 | 0.975 | 0.92 |
| **Weighted Avg** | 0.957 | 0.015 | 0.957 | 0.957 | 0.957 | 0.943 | 0.972 | 0.928 |

Table: 11. Performance Criteria for Adaboost.M1 Meta Classifier with REP Tree

| Activity | TP-Rate | FP-Rate | Precision | Recall | F-Measure | MCC | ROC Area | PRC Area |
|---|---|---|---|---|---|---|---|---|
| **Walking** | 0.985 | 0.013 | 0.98 | 0.985 | 0.983 | 0.971 | 0.998 | 0.998 |
| **Jogging** | 0.923 | 0.001 | 0.968 | 0.923 | 0.945 | 0.944 | 0.999 | 0.98 |
| **Stairs** | 0.964 | 0.002 | 0.964 | 0.964 | 0.964 | 0.962 | 0.998 | 0.982 |
| **Sitting** | 0.963 | 0.011 | 0.969 | 0.963 | 0.966 | 0.954 | 0.996 | 0.987 |
| **Standing** | 0.974 | 0.006 | 0.968 | 0.974 | 0.971 | 0.966 | 0.995 | 0.976 |
| **Lying Down** | 0.968 | 0.004 | 0.969 | 0.968 | 0.968 | 0.964 | 0.996 | 0.989 |
| **Weighted Avg.** | 0.973 | 0.01 | 0.973 | 0.973 | 0.973 | 0.964 | 0.997 | 0.99 |

Table: 12. Performance Criteria for Adaboost.M1 Meta Classifier with J48

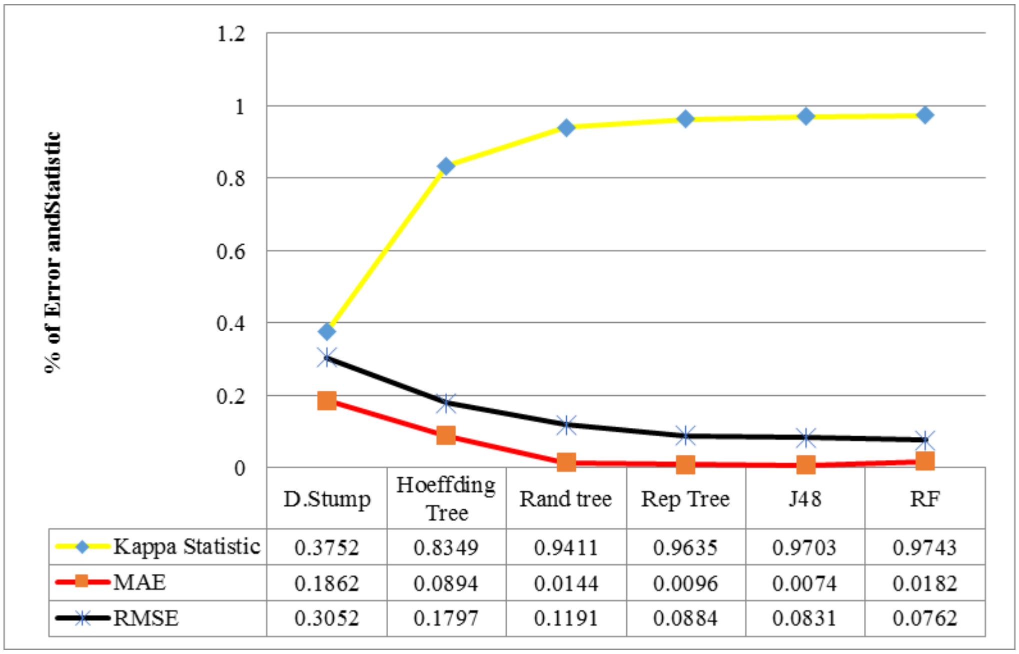
| Activity | TP-Rate | FP-Rate | Precision | Recall | F-Measure | MCC | ROC Area | PRC Area |
|---|---|---|---|---|---|---|---|---|
| **Walking** | 0.991 | 0.014 | 0.98 | 0.991 | 0.985 | 0.976 | 0.999 | 0.999 |
| **Jogging** | 0.946 | 0.001 | 0.953 | 0.946 | 0.95 | 0.949 | 0.999 | 0.985 |
| **Stairs** | 0.932 | 0.001 | 0.971 | 0.932 | 0.951 | 0.949 | 0.999 | 0.982 |
| **Sitting** | 0.972 | 0.006 | 0.982 | 0.972 | 0.977 | 0.969 | 0.998 | 0.996 |
| **Standing** | 0.985 | 0.005 | 0.971 | 0.985 | 0.978 | 0.973 | 0.999 | 0.992 |
| **Lying Down** | 0.963 | 0.002 | 0.983 | 0.963 | 0.973 | 0.97 | 0.998 | 0.992 |
| **Weighted Avg** | 0.978 | 0.008 | 0.978 | 0.978 | 0.978 | 0.971 | 0.999 | 0.995 |

Table: 13. Performance Criteria for Adaboost.M1 Meta Classifier with Random Forest

| Activity | TP-Rate | FP-Rate | Precision | Recall | F-Measure | MCC | ROC Area | PRC Area |
|---|---|---|---|---|---|---|---|---|
| **Walking** | 0.993 | 0.011 | 0.984 | 0.993 | 0.988 | 0.981 | 1 | 0.999 |
| **Jogging** | 0.969 | 0.001 | 0.969 | 0.969 | 0.969 | 0.968 | 1 | 0.996 |
| **Stairs** | 0.972 | 0.001 | 0.972 | 0.972 | 0.972 | 0.971 | 1 | 0.995 |
| **Sitting** | 0.968 | 0.006 | 0.983 | 0.968 | 0.975 | 0.967 | 0.999 | 0.998 |
| **Standing** | 0.983 | 0.005 | 0.972 | 0.983 | 0.978 | 0.973 | 0.999 | 0.997 |
| **Lying Down** | 0.973 | 0.001 | 0.989 | 0.973 | 0.98 | 0.978 | 0.999 | 0.995 |
| **Weighted Avg.** | 0.981 | 0.007 | 0.981 | 0.981 | 0.981 | 0.975 | 0.999 | 0.998 |

Table: 14. Performance Measures for Adaboost.M1 Meta Classifier with all classifiers

| Performance Measures | Decision Stump | Hoeffding Tree | Random Tree | REP Tree | J48 | Random Forest |
|---|---|---|---|---|---|---|
| Correctly Classified Instances | 57.31% | 87.84% | 95.69% | 97.33% | 97.83% | 94.44% |
| Incorrectly Classified Instances | 42.69% | 12.16% | 4.31% | 2.67% | 2.17% | 5.56% |
| Kappa statistic | 0.3752 | 0.8349 | 0.9411 | 0.9635 | 0.9703 | 0.9203 |
| Mean-absolute-error | 0.1862 | 0.0894 | 0.0144 | 0.0096 | 0.0074 | 0.0503 |
| Root mean squared error | 0.3052 | 0.1797 | 0.1191 | 0.0884 | 0.0831 | 0.1275 |
| Relative absolute error | 76.37% | 36.66% | 5.89% | 3.93% | 3.03% | 20.54% |
| Root relative squared error | 87.41% | 51.46% | 34.11% | 25.33% | 23.81% | 36.46% |
| Coverage of cases (0.95 level) | 98.56% | 99.87% | 95.81% | 98.22% | 98.07% | 99.95% |
| Mean rel. region size (0.95 level) | 59.96% | 69.24% | 16.75% | 17.23% | 16.80% | 32.22 % |
| Total Number of Instances | 5435 | 5435 | 5435 | 5435 | 5435 | 5418 |
| Time taken to build model: | 0.16 seconds | 2.48 seconds | 0.06 seconds | 2.13 seconds | 7.73 seconds | 2.27 seconds |



| | D.Stump | Hoeffding Tree | Rand tree | Rep Tree | J48 | RF |
|---|---|---|---|---|---|---|
| Kappa Statistic | 0.3752 | 0.8349 | 0.9411 | 0.9635 | 0.9703 | 0.9743 |
| MAE | 0.1862 | 0.0894 | 0.0144 | 0.0096 | 0.0074 | 0.0182 |
| RMSE | 0.3052 | 0.1797 | 0.1191 | 0.0884 | 0.0831 | 0.0762 |

**Fig: 1. Kappa Statistic, Mean Absolute and Root Mean Squared Errors for the Classifiers**

ISSN: 0976-3104

Table: 15. Ranking Table with test output with all 06 classifiers on WISDM Dataset

| Dataset | Random Tree | Decision Stump | Hoeffding Tree | J48 | Random Forest | REP Tree |
|---|---|---|---|---|---|---|
| WISDM Dataset | (100) 89.76 | 63.69 * | 75.54 * | 93.94 v | 94.60v | 94.61v |
| | (v/ /*) | (0/0/1) | (0/0/1) | (1/0/0) | (1/0/0) | (1/0/0) |

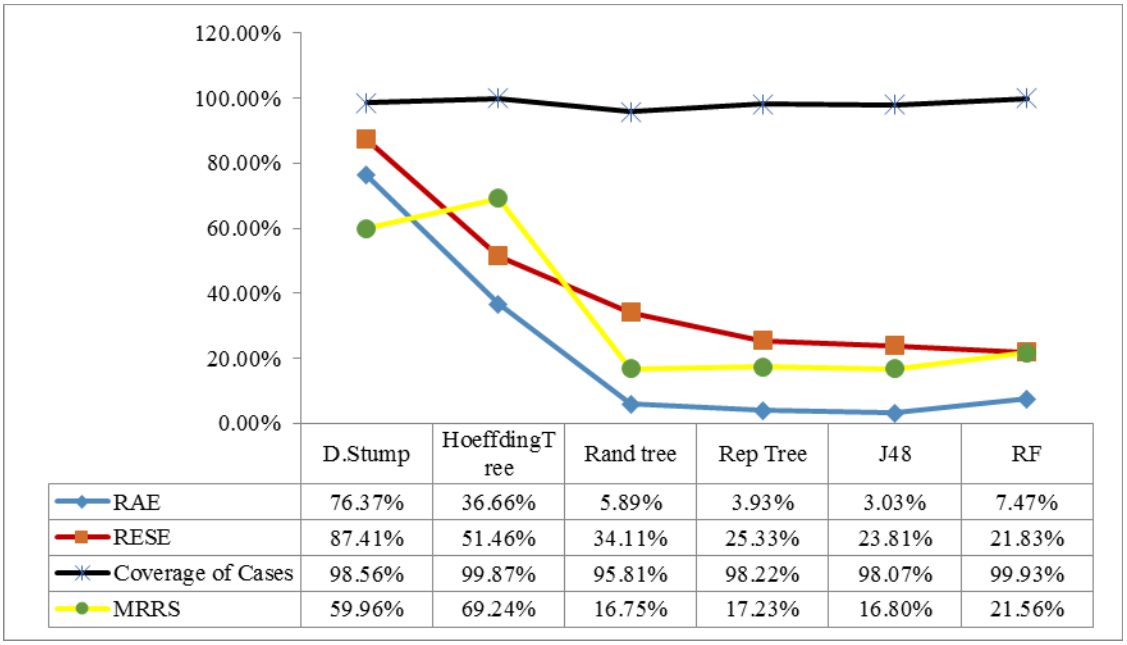


Fig: 2. Kappa Statistic, Mean Absolute and Root Mean Squared Errors for the Classifiers

The **Table– 15**, the ranking of all six classifier algorithm. While performing experiment, each classifier was repeated 10 times on the dataset and the mean accuracy is shown and the standard deviation in rackets of those 10 runs. The table shows Random Forest, REP Tree and J48 algorithms have a little "v" next to their results. That indicate how each classifier is statistically significant win against others on the WISDM dataset. This means an accuracy of a classifier is better than the accuracy of another classifier algorithm with the statistically significant difference.

## CONCLUSION

We can conclude that the Random Forest, REP Tree and J48 algorithms which have a little "v" next to their results means that the difference in the accuracy of these algorithms compared to Decision Stump, and Hoeffding Tree is statistically significant. We can also see that the accuracy of these algorithms compared to Decision Stump, and Hoeffding Tree is high, so we can say that these two algorithms achieved a statistically significantly better result than the Decision Stump, and Hoeffding Tree and Random Tree baseline.

## CONFLICT OF INTEREST

Authors declare no conflict of interest.

## FINANCIAL DISCLOSURE

No financial support received to carry out this research.

## REFERENCES

[1] World Population Aging http://www.un.org/esa/population/publications/worldageing19502050/pdf/62executivesummary_english.pdf

[2] Khan, Adil Mehmood and Lee, Young-Koo and Lee, Sungyoung Y and Kim, Tae-Seong [2010] A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer, Information Technology in Biomedicine, *IEEE Transactions*;14:5–1166.

[3] Westerterp, Klaas R [2009] Assessment of physical activity: a critical appraisal, *European journal of applied physiology*,105:6–823.

[4] Poppe, Ronald. [2010] A survey on vision-based human action recognition, Image and vision computing, 28:6–-976.

[5] Casale, Pierluigi, Pujol Oriol, Radeva Petia. [2011] Human activity recognition from accelerometer data using a wearable device, Pattern Recognition and Image Analysis, 289, Springer.

[6] Krishnan Narayanan C, Colbry Dirk Juillard, Colin and Panchanathan Sethuraman. [2008] Real time human activity recognition using tri-axial accelerometers, Sensors, signals and information processing workshop.

[7] Kwapisz, Jennifer R and Weiss, Gary M and Moore, Samuel A. [2010] Cell phone-based biometric identification, Biometrics: Theory Applications and Systems (BTAS), 2010 Fourth IEEE International Conference.

[8] N Ravi, N Dandekar, P Mysore, and ML Littman. [2005] Activity recognition from accelerometer data. IAAI-05: *American Association for Artificial Intelligence*.

[9] L Bao and SS Intille. [2004] Activity recognition from user-annotated acceleration data. In Pervasive, pages 1–17.

[10] J Lester, T Choudhury, N Kern, G Borriello, and B Hannaford. [2005] A hybrid discriminative/generative approach for modeling human activities. *In Proc. of the International Joint Conference on Artificial Intelligence (IJCAI*), pages 766–772.

[11] Dean M. Karantonis, Michael R. Narayanan, Merryn Mathie, Nigel H. Lovell, Branko G. Celler [2006] Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring, *IEEE Transactions on Information Technology in Biomedicine*, 10(1): 156–167.

[12] GM Weiss and JW Lockhart.[2012] A comparison of alternative client/server architectures for ubiquitous mobile sensor-based applications," in Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12, 2012, p. 721.

[13] JW Lockhart, GM Weiss. [2014]Limitations with Activity Recognition Methodology & Data Sets," in 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, 747–756.

[14] JW Lockhart, T Pulickal, and GM Weiss. [2012]Applications of mobile activity recognition," in Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12, p. 1054.

[15] S Gallagher.[2014] Smartphone Sensor Data Mining for Gait Abnormality Detection," Fordham University, New York, 2.

## ABOUT AUTHORS

*__K. H. Walse__ is working as Assist. Prof.at Anuradha Engineering College, Chikhli Distt. Buldana. Prior to this He was Principal at Shreeyash Polytechnic, Aurangabad during 2010-12. He was with Anuradha Engineering College as an Assistant Professor of Computer Science and Engineering where he leads the HCI Research Group(1996-2010) Prior to joining Anuradha Engineering College in 1996, he was an Lecturer of Computer Science and Engineering at SSGMCE, Shegaon (1994-1996). He had been a technical program Convener for ACM ICAC-2008 International Confernce. He is Fellow Member of IEI and IETE, India, Senior Life Member of Computer Society of India, Professional Member of ACM and IEEE Professional Society. He is working as Editor-in-Chief for the reputed International Journal of Computer Science and Applications (IJCSA), He has published many research papers in International Journal and International Conferences.*

*__Dr. Rajiv V. Dharaskar__ is former Director, Disha Education Society (DES, DIMAT. Disha Technical Campus) Raipur.. He is former Director, MPGI Group of Institutes Integrated Campus, Nanded. He is former Professor and Head, PG Department of Computer Science and Engineering, G H Raisoni College of Engineering, Nagpur. He is Ph.D. in Computer Science & Engineering, M.Tech. (Computers), P.G. Dip. (Computers), M.Phil., and M.Sc. He is having 31 years of teaching and 25 years of R&D experience in the field of Computers & IT. He has published series of 6 books and over 300 research papers on computer engineering. He has guided 19 Ph.D. scholars on various subjects like Digital Forensics / Cyber Security, Software / Usability Engineering, HCI, Mobile Computing, E-Commerce, E-Learning etc. He is on editorial or review board of prestigious International Journals and worked as a Reviewer for dozens of International Conferences.*

*__Dr. V. M. Thakare__ is working as Professor and Head in Computer Science, Faculty of Engineering & Technology, Post Graduate Department of Computer Science, SGB Amravati University, Amravati. He has done his Ph.D in Computer Science and Engineering. He represented many Expert Committees of ACITE, CEDTI, YCMOU. He is having teaching experience more than 21 years on UG, PG level. He has guided many students for Ph.D., M.Phil, students. He guided more than 300 M.E. projects. He has published many research papers in reputed international journals, International conferences and national conferences. He is also working on various bodies of university. He is Life member of Computer Society of India, ISTE, and Institution of Engineers.*