MOP: Predicting Multiple Output in Multi-Sharing System

Dhanalakshmi B K, Srikantaiah K C, Venugopal K R

Abstract*: *Cloud computing is relatively advanced field in which we believe resource utilization hasn't yet been optimized to its complete potential and inaccuracy of prediction leads to several minutes of delays in instant resource allocation due to scarcity of resources in Multi-Sharing System. In this paper, we develop Extraction of Transaction Log Files to Predict Multiple Output (MOP) in Multi-Sharing System based on resource utilization for higher accuracy using prediction techniques Random Forest and majority voting algorithms. The goal is to gratify upcoming resource demands and to avoid over or under provisioning of resources. The accuracy results show that the proposed model provides higher accuracy in predicting resource utilization for upcoming resource demands and prediction cost and time are reduced.

Keywords*: *Cloud Computing , Log File, Majority Voting, Multiple-Output Prediction, Random Forest.

I. INTRODUCTION

In the contemporary world of digitization, we often find ourselves amidst a formidable amount of raw data. Although present-day Data Miners have instilled numerous methods to address the obstacles that come with this ocean of data, we find that to process data in cloud computing and its ability to dynamically allocate resources (inculcating techniques such as prediction) is indispensable in terms of effectiveness and efficiency. The transactions and activities of data are stored in log files on Systems. The impediment lies in extracting and analyzing of these log files, which is a tedious task [1]. So, we mine log files to extract essential information like resource usage, time logs and historical data of importance. Mining vital information from them is performed by pre-processing. Pre-processing is a conversion of unstructured data into structured data. It consists of four major tasks: Data Cleaning and Filtration, Data Cube Construction, Data formatting. The aforementioned structured data that pre-processing engenders is converted into work-flows. [2].

A workload is an aggregation of multiple work-flows [3], workloads are classified based on resource usage and predicting accurate resource utilization (CPU, memory, Network I/O, disk etc.) is a meticulous job. Existing systems portray a simple form of single output prediction. We enhance this concept to form and focus on predicting multiple outputs for various levels of workload. Although Single output prediction is much easier than Multiple Output prediction (MOP), the latter it has its own advantages that are exclusive to MOP. In MOP, the best out of all outputs is selected and the service provider provisions resources to multiple users in multi-sharing system.

Manuscript published on 30 September 2019
* Correspondence Author
Dhanalakshmi B K, Department Computer science and Engineering, S J B Institute of Technology, Bangalore, India.
Srikantaiah K C, Professor in the Department of Computer Science and Engineering at S J B Institute of Technology, Bangalore, India.
Venugopal K R, Ph.D in Computer Science from Indian Institute of Technology, Madras.

A. Motivation

It is seen that Single Output Prediction (SOP) produces a higher rate of failure due to a restricted resource reservation that could rather be avoided. Prediction with a single output has its own disadvantage, if the prediction results are wrong then either the cost of resource utilization is too high, or the resource may remain idle for an undesirably long period of time. The MOP model is designed to overcome this problem by minimizing the error rate of prediction, cost, time and performing efficient resource utilization.

B. Contribution

In the existing system, predicting single output is not accurate enough to dynamically allocate resources for different levels of workloads as it does not account for multiple factors. It potentially causes under-provisioning or over-provisioning of resources. MOP, extracts and pre-process the log files and generates the workflow and aggregation of workflows into workload by using Random Forest algorithm, it predicts multiple outputs and selects the best output out of multiple predictions using majority voting. This drastically improves the efficiency of the prediction model in terms of cost, time consumption and resource usage.

C. Organization

The following sections are structured as described below: Section II covers the Related Work in the field of multiple output prediction. Section III states the Problem Definition. Section IV Description about Proposed Model. Section V represents the Evaluation Criteria. Experimental Setup and performance evaluation are shown in Section VI. Section VII Covers the Conclusion and Future Work.

II. RELATED WORK

This segment comprises of relevant research in Multiple-Output Predictions.

Khan, *et al.*, [4], proposed a prediction model to predict resource utilization for workload by extracting the required data from log files by using regression and queuing network model.

It requires estimating the pre-function of resource usage and benchmarking each transaction of users to obtain matrix of resources utilization. The results predict the resources for various multiple types of workload.

Da silva, *et al.*,[5] designed an application for periodical work-ow which are mostly used for business purpose to estimate the required resources of various periodical application workloads. Integer programming model and Precedence Tree based Heuristic (PTH) is used to minimize the renting cost which overheads the scheduling problems.

Pietri,*et al.*,[6] focuses on summarizing basic methods of log file analysis. It concludes that log file analysis has for long been a neglected field of computer science and breaks down three case studies to portray the different applications of log file analysis.

Pham,*et al.*,[7] proposed a provisioning strategy for instant resource allocation is delayed hardware resources in cloud. So, to overcome this problem, Linear Regression and Neural Network are used to meet the future
Resource demands and predicts the resources.

Zheng,*et al.*,[8] identifies Early detection of anomalies such as lengthy delays or unnecessary costs in cloud environments due to failure in resource management is essential in modern workflow systems in Post-detection, the cause must be identified and certain actions must be performed to mitigate the effects of these anomalies and Hierarchical Temporal Memory (HTM) is used in order to detect performance anomalies. This model works on real-time metrics accumulated by indefinitely monitoring the resource consumption of workflow tasks being executed

Comer,*et al.*,[9] mentioned the load balancing technique for auto scaling feature to reduce the traffic fluctuation and to increase the back-end capacity in cloud computing. The start and end time of virtual machines is determined by load balancer. It is effective for centralizing systems.

Vora,*et al.*,[10] introduced an efficient computing resource provisioning in the cloud, and it characterized the capabilities of workload prediction for virtual machines. It traces for data which is repeatable patterns of workload by cross vm correlations workload from the dependencies which is running in various virtual machines.

Chen,*et al.*,[11] proposed a technique co-clustering to group virtual machines which patterns of workloads correlated frequently and identifies the virtual ma-chine group which are active during the period of allocation. The Hidden Markov Modeling (HMM) correlation method is used for clustering the virtual machine to predict the workload pattern variations.

Islam,*et al.*,[12] introduced a prediction model for measuring resource usage and introduced neural network and linear regression for provisioning strategies to fulfill the resource user demand needs to minimize the on-demand resources in cloud computing .

Ali Yadavar, *et al.*,[13] introduced auto scaling systems for Infrastructure as a Service (IaaS) for predicting the resources accurately by using prediction time series algorithm based on scalar vector machine(SVM). Amazon EC2 infrastructure and TPC-W is used to generate various types of workload.

Sangmyeon Park,*et al.*,[14] proposed a prediction method about power consumption of CPU, Memory, and Hard disk by using utilization rate and by using prediction algorithms they have compared and analyzed actual and predicted power consumption.

K.PushpaLatha,*et al.*,[15] proposed an task scheduling load balancing algorithm to decrease the migration time and response time by using task scheduling with load balancing technique (TSLB algorithm) and compression technique is associated to maximize the resource utilization and to schedule the resources instantly to virtual machine by using load leveling strategy to perform various operations by customer.

L Shakkeera, *et al*.,[16] improvising the scalability of resources by using Load balancing algorithm to optimize the cost for Iaas cloud infrastructure. The strategies of cloud load balancing techniques evaluated the Quality of Service (QoS) performance metrics like cost, average execution times, throughput, CPU usage, disk space, memory usage, network transmission and reception rate, resource utilization rate and scheduling success rate.

Ashalatha R, et *al.*, [17] mentioned the load balancing technique for auto scaling feature to reduce the traffic fluctuation and to increase the back-end capacity in cloud computing. The start and end time of virtual machines is determined by load balancer. It is effective for centralizing systems.

M.Kriushanth, *et al*., [18] surveyed cloud computing and emphasis on auto scaling mechanism for computing the resources based on the user needs. It is a collection of servers and provider for the user which benefits in cost and time where user need not to purchase any hardware devices and install it can be accessible by cloud from any point of time. The major service provider is Rackspace, Salesforce, Amazon, Google, IBM, Dell and HP.

III. PROBLEM DEFINITION

Consider a log file *'D'* of user activities in multi-sharing system and it is pre-processed to remove redundant, irrelevant, missing values. Based on pre- processed the workflows are generated and checks for same level of workflows and combined to generate one new workload. our objective is to Predict multiple outputs for each workload by applying random forest technique to avoid under-provisioning and over-provisioning of resources and to minimize the cost and time increase accuracy.

IV. PROPOSED MODEL

A. System Architecture

The proposed model consists of several steps to predict multiple outputs for efficient resource utilization such as (i) Log file and Preprocessor (ii) Workflow Generator (iii) Workflow Aggregator (iv) Predictor (v) Multiple output predictor (vi) Majority Voting
Selector as shown in Figure 1.

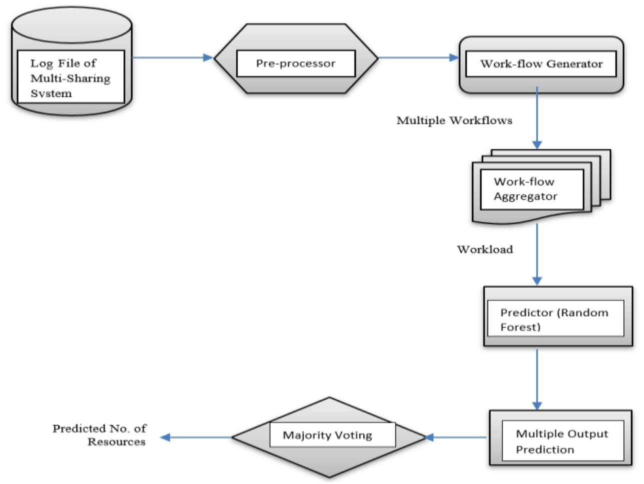

Fig. 1.System Architecure.

Log File and Pre-Processor:

The log file stores the information of user's transactions and activities and log file size ranges from petabytes (PB) to exabytes (EB) or even to zettabytes (ZB) are sent to pre-processor. Pe-processor cleans the log files by removing all the redundant, irrelevant and missing values from log files. Next, it selects the most and least used resources using data filtering and converts the structured file to the required format by using data formatting. Later, the structured file format is passed to workflow generator phase.

Workflow Generator:

The Workflows are an activity of a user and each user workflow is drawn by using decision trees by creating first task as root, sub node and till it reaches leaf node. The generation of workflows for each user is shown in Figures from 2 to 7. For example: Consider shopping history of an users, each user performs a various activity like login, search, bidding an item, send invoice, receive invoice , payment details, Packing an item, received the item. The A B C D E F are cloud user; the user is represented as root in dark circle and the first activity is represented as sub node in thin dotted lines and intermediary transactions are represented in thin circles and the last activity is represented as a leaf node thin dark circle. Now, the user 'A' performs only two activity ie., the login and search from auction list and user B performs only3 activities ie., login and search from auction list and bidding an item, user C perform only 2 activities ie., login and search from auction list, user D perform only 3 activities ie., login and search from auction list and bidding an item. User E and F performs all activities ie., login, search, bidding an item, send invoice, receive invoice , payment details, packing an item and received the item. These workflows are the input for workload aggregator.

Workflow Aggregator:

The workflow aggregator is an aggregation of same levels of workflows to obtain workloads, based on the user activity. If the user activity is higher, then level of nodes are also higher in number or else the level of the nodes are lower in number, then we check for the same level of nodes and if the level of the node value is same then such workflows are grouped together to form one workload this done to reduce prediction time and cost. Form the given example: User (A and C) have the same level of node, (B and D) have the same level of node, (E and F) have the same level of node, user E and F performed all activities ie.,. Here, User (A and C) (B and D) (E and F) are grouped together as shown in Figure 8 to 10. Now, predicting resources for each workflow is time consuming and costly. So, we have combined the workflows to get workload and these workloads are given as input to Predictor.

Multiple output predictor:

The workload is the input for Multi output predictor. It predicts multiple outputs based on random forest technique, where it is a decision tree based . Each tree depends on the random features selected and each workload it starts generating tree and becomes a forest. Next, we split the tree into sub-nodes and calculates the nodes by using features ,for these nodes, apply rules for each randomly created decisions tree to predict the output and stores as the predicted output as target value. For each workload, resource us- age is predicted by generating the greater number of trees and multiple predicted values are obtained.

Majority Voting Selector:

The multiple outputs are considered as input, the voting starts for decision tree and calculate the number of votes for each predicted value. Next, the highly voted predicted value as the final prediction from the random forest algorithm.

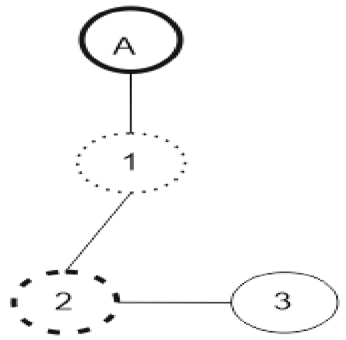

Fig. 2. Workflow User A..

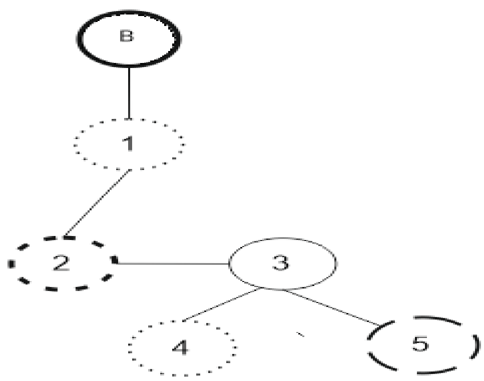

Fig. 3.Workflow User B.

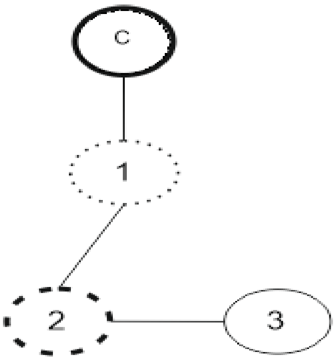

Fig. 4.Workflow User C.

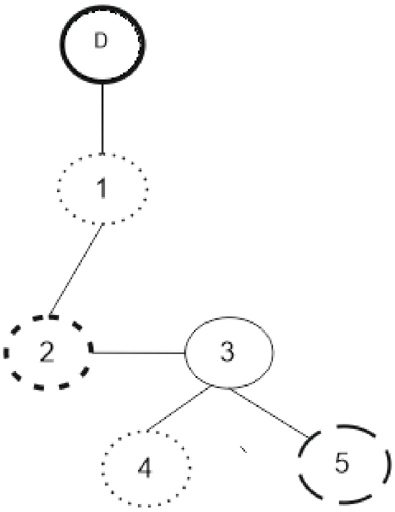

Fig. 5.Workflow User D.

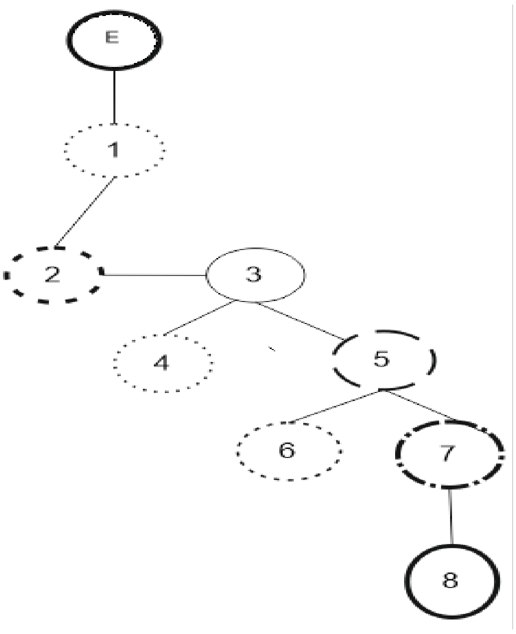

Fig. 6.Workflow User E.

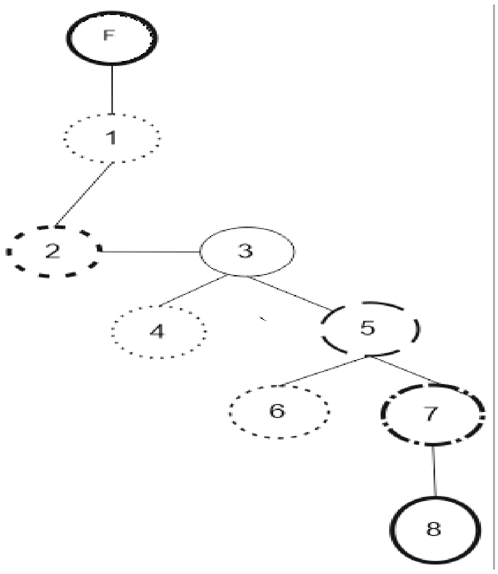

Fig. 7.Workflow User F.

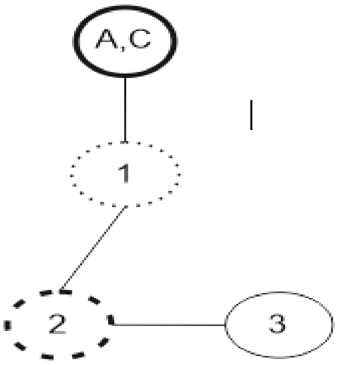

Fig. 8.Workflow User A

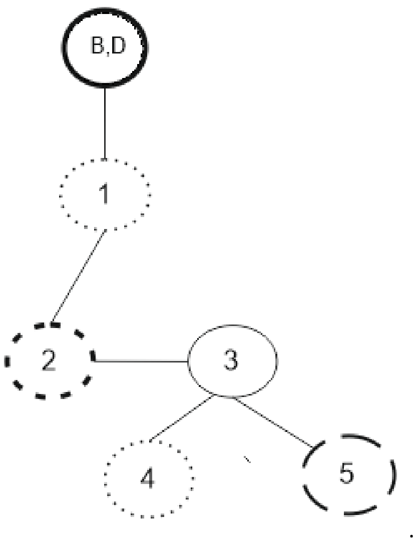

Fig. 9.Workflow User B

Retrieval Number C5482098319/2019©BEIESP
DOI: 10.35940/ijrte.C5482.098319
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

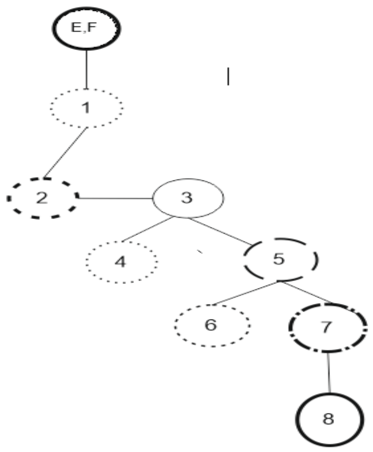

Fig. 10. Workflow User C.

For Example: Consider predicted value for each tree then calculate the votes. Let's us say there are *1000* random trees, predicts *3* unique values *a,b,c* then *'a'* is nothing but when *1000* random tree predicted out of this how many trees predicts *'a'*. Same for next two values *b and c*. If 'a' is getting high votes and suppose *'y'* is voted more. Let take *100* trees out of which *70* trees is voted for y the predicted value, then *'y'* is selected as final value as prediction value, and this is called as majority voting. Then for reserving the resources in multi-sharing system with higher accuracy

B. Mathematical Model

Consider the log files *D* of each users *U* where, $U=\{ U_1, U_2 .... U_j \}$ and pre-process the log files and obtained *D* and extract the information about each user such as *{ CPU usage, Memory, Bandwidth, Cost, Time}* the results of pre-processing are obtained by using data mining techniques. The resources are *R={ CPU, Memory, Disk, Bandwidth }* Based on resource usage of each user we construct workflow U_i, Where,

$$U_i = \{Wfi_1, Wfi_2 \ldots\ldots\ldots Wfi_k\} \quad (1)$$

Where, $1 \leq i \leq k$ and by constructing the work-ow, we obtain the levels of nodes (from root to leaf node). Resource usage of each user is computed based on these levels, and we classify the on-demand, satisfied and contributed users and highest resource usage consumed by users. Next, checks the levels from root to leaf node, if levels are same then we combine such workflows into one workload *(Wd).* Where, each workload contains more than one workflow

$$W_d = \{Wfl_1 + Wfl_2 + \ldots\ldots + Wfl_m\} \quad (2)$$

where, $1 \leq i \leq m$.

The user who exceed their resource usage from reserved resources then such users are said to be on-demand user. The actual resources usage R_{actual}l and reserved resources $R_{reserved}$ are also extracted from log files. The resource usage of each user is computed by actual and reserved resources by using

$$R_{usage} = R_{actual} - R_{reserved}.$$

If the R_{usage} value is positive, then user has requested for on-demand resources and assigns a value 1(yes) else assign 0(Not requested for on-demand resources). The total number of on-demand users is obtained by adding the on-demand request. The on-demand users are classified into class 1 and class 2 based on the amount of resource usage, if on-demand users have utilized all the resources in R then it is classified into class 2, otherwise classified as class 1. Once it is classified as class 2, the prediction is done only once because all on-demand user shave used same type of resources and for class 1 we need to apply prediction for each type of resources. This process is done to reduce time and on-demand users. Next, Predict the Resource usage (R_{Predi}) of i^{th} resources in $R(r_i)$ by actual resources (R_{actual}) of each workload is decision tree based using random forest algorithm. when number of decisions tree are generated with pair of actual and predicted value of all the workloads is defined as.

$$L = \{(R_{actual1}, R_{predi1},) \ldots\ldots (R_{actualn}, R_{predin})\} \quad (3)$$

Where,

R_{actual} is an actual resource and R_{predi} is predicted resources for each workload.

By considering the given workload

$$h = \{ h1(R_{actual}), \ldots\ldots\ldots\ldots hl(R_{actual})g\} \quad (4)$$

Now, if each h_1 is a decision tree, then group all h is a random forest. Consider parameter of the decision tree for classifier $h1(x)$ to be $\phi = \{\phi_{k1}, \phi_{k2 \ldots\ldots\ldots} \phi_{kp}\}$ *Where,* $i \leq k \leq p$ and thus decision tree *k* leads to a classifier

$$h1(x) = h(X \mid \phi_k) \quad (5)$$

For the final selection *(x)* which combines all the classifiers $h_k(x)$ and each tree casts a vote for the most popular class at input x, and the class with the most votes wins. Specifically given data $D = (R_{actuali}, R_{predi})$ (6)

We train an $h_k(x)$ level of classifiers *n*

Each classifier $h_k(x) = h(X \mid \phi_k)$ is in our case a predictor of *n*, $y = +1$ outcome associated with input *X*

Based on the prediction results the administrator reserves the resources from cloud service provider and allocates the resources to multi-user in multi-sharing system and in detail explained in *Algorithm 1 and 2*.

Algorithm 1: The Workflow and Workload Generation

Input: Pre-processed Log File
Output: Workflow and Workload Generation
Procedure: Workflow (Wf)
Workload (Wd) (Combination of Workflow)
Workload $Type_1$ $Wd_{T1}[W_{f1}, W_{f2}, W_{f3}, W_{f4}, .., W_{fn},]$
Workload $Type_2$ $Wd_{T2}[W_{f1}, W_{f2}, W_{f3}, W_{f4}, .., W_{fn},]$
Workload $Type_D$ $Wd_{TD}[W_{f1}, W_{f2}, W_{f3}, W_{f4}, .., W_{fn},]$
Workload becomes Workloads (Wds)
Analytical Data File (AF)
//Gathering and grouping Work Flow and Work Load
While Not AF Log File in Collection
For Each Wf $k=1$ to *n*
Switch (Wf - Type)
Case: Transaction Type *T1*
Add to Wd_{T1}
Break;

Case: Transaction Type *T2*
Add to Wd_{T2}
Break;
Default:
Add to Wd_{TD}
Break;
End Switch
End For Each
Add $Wd_{T1}, Wd_{T2} \dots Wd_{TD}$ *to* Wd_s
End While
//Understand how much resource each type of Work Flow consumes
For Each Workload Type randomly pick a *Wf*
Process (Wf)
Get Memory Used
Get CPU Bandwidth
Get CPU Cycle
End Process
End for Each

Algorithm 2: MOP the proposed algorithm:

Purpose: To predict resources for on-demand users
Input: Log files, *k*, *m*
Output: Multiple output predictions
Step 1: Obtain structured log file *D* from given log file using data mining pre-processing techniques.
Step 2: workflows and workloads ()
Step 3: For each user $u_i \in U$ extracts $R_{actuali}$ *and* $R_{reservedi}$
Step 4: Compute $R_{usagei} = R_{actuali} - R_{reservedi}$
if R_{usagei} is Positive then On-demand request_ *[i]* =1
else
On-demand_ request*[i]* =0
Step5: for each user $u_i \in U$ for which on-demand request[i]=1
Compute the number of resource utilization (*nri*),
if (*nri* = number of resource types in *R*) *ie.*, user u_i
Utilized all resource types in *R* then classify user u_i as Class_2 else Class_1
Step 6: Choose randomly k features from *m* features where *k* < *m* using equation (4)
Step 7: By using best split method calculate the node d by selecting *K* features.
Step 8: Start Splitting the node into sub nodes by using the best split method
Step 9: Execute the steps 6 to 8 repeatedly till it reaches *l* number of nodes.
Step 10: Constructs forest by repeating steps 6 to 10 till all predicted value for all *k* trees
Step 11: Consider the test features and apply the rules of each randomly created decision tree to predict the output and this output are stored as the predicted output target value.
Step 12: Next, counts the votes for each predicted output target value.
Step 13: Finally, highly voted predicted output target as the final prediction value by using random forest algorithm

C. Evaluation Criteria

We assess the prediction models for accuracy by using from various prediction algorithms based on the metrics: Mean Absolute Percentage Error (MAPE) , PRED(25), Root Mean Squared Error (RMSE) and R_2 Prediction Accuracy is defined as [2]. These metrics have been elaborated in detail.

Mean Absolute Percentage Error (MAPE) The MOP prediction model is evaluated based on the metric Mean Absolute Percentage Error by using the formula:

$$\text{MAPE} = \frac{1}{n}\sum_{n}^{i=1} |\,\text{Ractual} - \text{R\^{}predi}\,| \qquad (7)$$

Where R_{actual} is the actual output, $R\wedge_{predi}$ is the predicted output and *n* is No. of iteration made to prediction model from data set. A lesser value of MAPE indicates a good fit of the prediction model.

PRED (25) The calculation of PRED(25) metric is defined number of observations with relative value error should fall within *25* of the actual value and it is defined as

$$\text{PRED}(25) = \frac{\text{No ofobservationswithrelativeerror} \le 25}{\text{No: ofobservations}} \qquad (8)$$

Root Mean Squared Error (RMSE) The Mop model is evaluated by using metric Root Mean Square Error (RMSE) is defined as:

$$\text{RMSE} = \sqrt{\frac{\sum_{n}^{i=1} |\,\text{Ractual} - \text{R\^{}predi}\,|}{n}} \qquad (9)$$

Where R_{actual} is the actual resource output, *R^predi* is the predicted resource output and *n* is No. of iteration made to prediction model from data set. A lesser RMSE value shows more efficient prediction model.

R_2 Prediction Accuracy The goodness-of fit of the prediction model is evaluated by using R_2 Prediction Accuracy metric and it is defined as:

$$R^2 = 1 - \frac{\sum_{n}^{i=1} |\,\text{R\^{}}_{\text{predi}} - \text{R}_{\text{actual}}\,|}{|\text{R\^{}}_{\text{predi}} - \text{R}_{\text{actual}}\,|} \qquad (10)$$

where, R_{actual} is the actual output, $R\wedge_{predi}$ is the predicted output and *n* is No. of iteration made to prediction model from data set. The R_2 value falls within the range [*0*, *1*] and it is used more for linear regression model and the value *1.0* is a best fit prediction model.

V. Experimental Setup

We carried out the experiment of our prediction model in the cloud simulator by implementing all our approaches. we have carried out this experiment in two different phases: Transactional Log Data Collected from multi-sharing system of Cloud users and Training of the Prediction model with the Historical Log Data.

First, log files are extracted, and it contains very elaborate information about each request. Data is selected carefully by pre-processing the log files. The three
Parameters to be considered are namely, (i) Cost of each transaction, (ii) CPU utilization per transaction and (iii) Time taken to execute each transaction. The software application Comindware Tracker is used to generate workflows and converted into workload.

In our experiment, we use a cloud simulator tool of version *3.03*. The configuration of the computer is as follows: CPU (*64-bit Intel Pentium i7 CPU 2.9 GHz) with 16 GB RAM* and *2TB* hard disk. The IDE of choice is Net beans 8.1 editor. The model is coded in Java language and executed on an Intel Core *i7* processor environment with *2TB* memory. Our model, MOP implements a grouping of workflows into workload and predicting resources for each workload using Random Forest and Machine Learning techniques. The size of the work- load does not exceed 1GB of data. The obtained results are shown in the Figure 8 and 9.

A. Results and Discussion

we evaluate the prediction accuracy of Multiple output prediction and single out-put prediction in terms of the metrics Mean Absolute Percentage Error (MAPE),Root Mean Squared Error (RMSE), R2 prediction accuracy and PRED(25).

	CPU(Cycles)			Bandwidth(GB)			Memory(TB)		
User	Single Output Prediction	Reserved	Resource Usage	Single Output Prediction	Reserved	Resource Usage	Single Output Prediction	Reserved	Resource Usage
	SOP			SOP			SOP		
A	20	20	10	10	10	10	1	1	2
B	30	30	25	30	30	25	1	1	50
C	20	20	10	10	10	10	1	1	2
D	10	10	8	10	10	8	50	50	60
E	40	40	38	20	20	10	95	95	2
F	20	20	20	15	15	15	1	1	75
G	40	40	38	20	20	48	95	95	95
H	50	50	58	40	40	30	10	10	20
I	40	40	30	30	30	30	40	40	30
J	6[illegible]	60	30	30	30	30	50	50	45

Fig. 11. . Resource Utilization based on Single Output Prediction.

Effect of Resource Utilization: The resource utilization in Single Output Prediction(SOP) using RPMRS model and in Multiple Output Prediction(MOP)using MOP model is as shown in Figure 11 and 12. From the Figure 13, the resource utilization is 95 percent compared to existing model because of multiple prediction value which reduces the contributor user, on-demand users, time.

Effect of Cost: The existing model and proposed model is compared with 3 parameters: Cost, On-demand request and accuracy. The RPMRS model the cost is low but

	CPU(Cycles)			Bandwidth(GB)			Memory(TB)		
User	Multi Output Prediction	Reserved	Resource Usage	Multi Output Prediction	Reserved	Resource Usage	Multi Output Prediction	Reserved	Resource Usage
	MOP			MOP			MOP		
A	20	20	20	10	10	10	1	1	1
B	30	30	30	30	30	25	1	1	1
C	20	20	20	10	10	10	1	1	1
D	10	10	10	10	10	8	50	50	50
E	40	40	40	20	20	10	95	95	95
F	20	20	20	15	15	15	1	1	1
G	40	40	40	20	20	48	95	95	95
H	50	50	50	40	40	30	10	10	10
I	40	40	40	30	30	30	40	40	40
J	60	60	60	30	30	30	50	50	50

Fig. 12. Resource Utilization based on Multiple Output Prediction.

due to single output prediction so, it leads to on-demand resources. The MOP model the cost is very low compared to RPMRS model due to no on-demand resources because it predicts multiple output as shown in Table 1.

Table- I: Shows the Comparison between Existing Model and Proposed Model

Model	***Cost***	***Accuracy***	***On-demand resources***
RPMRS	Low	95 % Accuracy	Yes
MOP	Very Low	98% Accuracy	No

Effect of On-Demand: From user point of view, the proposed model compares with existing model with and without prediction to reduce the on-demand request in multi-sharing system due to prediction of future resource requirements is not accurate. In MOP, upcoming resource demands are accurately predicted by using random forest algorithm. So, no point of requesting on-demand resources and contributing of resources and 95 percent users are satisfied users as shown in Table 1 and 2.

Table- II: Comparison of On-demand users with Prediction and without Prediction

	With Prediction			***Without Prediction***		
No.of User	On-Demand User	Contributor	Satisfied	On-Demand User	Contributor	Satisfied
100	30	20	50	03	02	95

Table- III: Comparison table of Accuracy between Single Value Prediction and Multiple Value Prediction

Single value Prediction		Multiple Value Prediction	
Gradient Descent	Random Forest	Gradient Descent	Random Forest
2.1%	2.1%	2.0%	1.1%

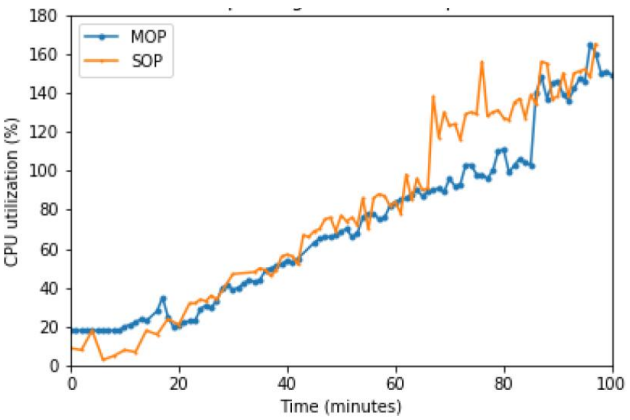

Fig. 13. Comparison between single value prediction and Multiple value prediction..

Effect of Accuracy: The prediction accuracy of resource requirement in Single Output Prediction (SOP) using gradient descent and in Multiple Output Prediction (MOP) using Random forest is shown as shown in Table 3 and with various metrics is shown in Table 4. From the table, prediction error is less in MOP is because of using random forest method.

Table- III: Comparison table of Accuracy between Single Value Prediction and Multiple Value Prediction based on different types of metrics

	Accuracy	
Metric	Single Value Prediction	Multiple Value Prediction
MAPE	2.1%	1.0%
PRED(25)	2.1%	1.1%
RMSE	2.3%	1.1%
R_2	2.1%	1.0%

Comparison of single prediction output and multiple prediction output is shown in Figure 10. The X-axis represents the time(minutes) and Y-axis rep- resents CPU utilization, the accuracy of prediction value with single output is not efficient as shown in Figure 10. The multiple prediction value can prediction exact resource utilization for future demand.

VI. Conclusion and Future Work

In this paper, an effective multiple output prediction models for adaptive resource utilization is proposed as the superior alternative over Single Output prediction. We consider this to be a progressive and a revolutionary approach for proactive resource utilization. After, closely evaluating multiple machine learning model, we concluded that the Random Forest algorithm with the usage of decision trees along with the majority algorithm creates the best possible for predicting multiple outputs for resource utilization in multi-sharing system with a tremendous improvement in accuracy in contrast to Single Output Prediction. This includes good planning and scheduling for interactive e-commerce applications where responsiveness is crucial. The quality of being instantaneous also enhances user Experience. Future work prediction of multiple outputs for heterogeneous environment.

REFERENCES

1. Valdman, J.: Log _le analysis. Dep. Comput. Sci. Eng. (FAV UWB)., Tech. Rep.DCSE/TR-2001{04. 51 (2001).W.-K. Chen, Linear networks and Systems (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.
2. Islam, S., Keung, J., Lee, K., Liu, A.: Empirical prediction models for adaptive resource provisioning in the cloud. Futur. Gener. Comput. Syst. 28, 155-162 (2012)
3. Rodriguez, M.A., Kotagiri, R., Buyya, R.: Detecting performance anomalies in scientific workows using hierarchical temporal memory. Futur. Gener. Comput. Syst.88, 624635 (2018).
4. Khan, A., Xifeng, Y., Shu, T., Anerousis, N.: Workload characterization and prediction in the cloud: A multiple time series approach. Network. Oper. Manag. Symp.(NOMS), 2012 IEEE. 12871294 (16)AD.
5. da Silva, R.F., Juve, G., Rynge, M., Deelman, E., Livny, M.: Online Task Resource Consumption Prediction for Scienti_cWorkows. Parallel Process. Lett. 25, 1541003 (2015).
6. Pietri, I., Juve, G., Deelman, E., Sakellariou, R.: A Performance Model to Estimate Execution Time of Scientific Workflows on the Cloud.
7. Pham, T.P., Durillo, J.J., Fahringer, T.: Predicting Workow Task Execution Time in the Cloud using A Two-Stage Machine Learning Approach. IEEE Trans. Cloud Comput. 7161, 113 (2017).
8. Zheng, Z., Wu, X., Zhang, Y., Member, S., Lyu, M.R., Wang, J.: QoS RankingPrediction for Cloud Services. Ieee Trans. Parallel Distrib. Syst. (2012).
9. Comer, D.K., Clark, C.R., Canelas, D.A.: Writing to learn and learning to writeacross the disciplines: Peer-to-peer writing in introductory-level MOOCs. Int. Rev.Res. Open Distance Learn. 15, 2682 (2014).
10. Vora, M.N.: Predicting Utilization of Server Resources from Log Data. Int. J.Comput. Theory Eng. 6, 118123 (2014).
11. Chen, L., Li, X., Ruiz, R.: Resource renting for periodical cloud workow applications. IEEE Trans. Serv. Comput. 07, 11 (2017).
12. Sadeka Islam,:Empirical prediction models for adaptive resource provisioning in the cloud. FGCS elsevier. 28, 155162 (2012)
13. Ali Yadavar,:Towards an Autonomic Auto-Scaling Prediction System for Cloud Resource Provisioning. 10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems.(2015)
14. Sangmyeon Park,:Prediction Method about Power Consumption by Using Utilization Rate of Resource in Cloud Computing Environment. IEEE. (2016)
15. K.PushpaLatha,: A Cost E_ective Load Balancing Scheme for Better Resource Utilization in Cloud Computing. Journal of Emerging Technologies in Web Intelligence,VOL. 6, 280-290 (2014)
16. L Shakkeera.:Improving Resource Utilization Using QoS Based Load Balancing Algorithm For MultipleWork-ows in IAAS Cloud Computing Environment. ICTACT Journal on Communication Technology, 750-757, (2013).
17. Ashalatha R,:Evaluation of Auto Scaling and Load Balancing Features in Cloud. International Journal of Computer Applications, 0975 8887,(2015).
18. M.Kriushanth.:Auto Scaling in Cloud Computing: An Overview. International Journal of Advanced Research in Computer and Communication Engineering,2870- 2875, (2013).

AUTHORS PROFILE

Dhanalakshmi B K is a Research Scholar, Department Computer science and Engineering, S J B Institute of Technology, Bangalore, India. She obtained B.E and M.E from Visvesvaraya Technological University in 2008 and 2013, My research interest is in the area of Grid computing, Cloud Computing, Data mining, Big data, Machine Learning, Artificial Intelligence.

Srikantaiah K C is currently working as Professor in the Department of Computer Science and Engineering at S J B Institute of Technology, Bangalore, India. He obtained his B.E from Bangalore Institute of Technology, M.E from University Visvesvaraya College of Engineering, Bangalore in 2002 and Ph.D degree in Computer Science and Engineering from Bangalore University, Bangalore, in the year 2014. He is guiding five Ph.D students in VTU. During his 15 years of service, he has 20 research papers to his credit. He has authored a book on Web Mining Algorithms. He has awarded best paper presentation award in the conference ICIP 2011 and his name is listed in Marquis who is who in the World 2014, 2015, 2016. His research interest is in Data Mining, Web Mining, Big Data Analytics, Cloud Analytics and Semantic Web.

Author-3

Venugopal K R is currently Vice-Chancellor of Bangalore, University. He obtained his Bachelor of Engineering from University Visvesvaraya college of Engineering. He received his master's degree in computer science and Automation from Indian Institute of Science Bangalore. He was awarded Ph.D in Economics from Bangalore University and Ph.D in Computer Science from Indian Institute of Technology, Madras. He has a distinguished academic career and has degrees in Electronics, Economics, Law, Business Finance, Public Relations, Communications, Industrial Relations, Computer Science and Journalism. He has authored and edited 57 books on Computer Science and Economics, which include Petrodollar and the World Economy, C Aptitude, Mastering C, Microprocessor Programming, Mastering C++ and Digital Circuits and Systems etc., He has filed 101 patents. During his three decades of service at UVCE he has over 550 research papers to his credit. His research interests include Computer Networks, Wireless Sensor Networks, Parallel and Distributed Systems, Digital Signal Processing and Data Mining. He is a Fellow of IEEE.