Turkish Journal of Computer and Mathematics Education Vol.12 No. 5 (2021), 1372-1377

Research Article

A Study and Analysis of Forecasts in Resource Allocation Using ARIMA in Cloud Environment

L. Rajalakshmi[a], E. Sathiyamoorthy[b]

[a]School of Information Technology and Engineering, VIT University, Vellore, Tamil Nadu, India.
E-mail: l.rajalakshmi2017@vit.ac.in
[b]School of Information Technology and Engineering, VIT University, Vellore, Tamil Nadu, India.
E-mail: esathiyamoorthy@vit.ac.in

Article History: Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

Abstract: Cloud computing refers to the delivering or usage of hosted services over internet rather than a traditional data center. Hosted services can be renting infrastructure/resources on demand or using the cloud as a platform to develop applications or using the cloud to host software that are accessed by clients. The bottom line is to obtain affordable resources from a provider and pay as you go in a flexible manner. In doing so, not all resources need to be obtained upfront. The initial capacity can be rented out and the remaining can be scaled as per the need. To handle such scalability, auto-scaling systems helps tackling the need to maintain the finite set of resources that can serve the current need and on the other hand also reduce the resources when the current need decreases. Very often in a cloud based environment it makes sense to adapt proactive strategies to scale the resources than to react after the surge had occurred. The proactive strategies use a quantified metric as a input to provision resources on demand that could meet the future expectations. This metric is obtained by carefully analysing the historical data of the application and in turn can influence the scaling decisions. Conclusions are drawn about the accuracy of the metric based on different timelines of historical information along with the confidence levels with which the prediction is done.

Keywords: Auto Scaling, Time Series, Workload Prediction.

1. Introduction

Cloud computing has facilitated a smoother transition from a traditional data center to a virtualized pool of resource that are dispersed geographically yet aggregated as one unified resource. The pool of resources can be hired from a cloud provider using a pricing scheme. The resources in use can be up scaled or downscaled based on the requirement. This is where the scaling strategies play a effective role. The behaviour of a scaling strategy can be reactive or proactive in nature. Proactive strategies tend to use a quantified metric to scale resources before the surge occurred where as reactive strategies tend to use the excessive required demand after the surge has occurred. The current work is more focussed on analysing the predictions from the historical data over varied timelines with different confidence level. This analysis helps making better decisions on which prediction can serve as the quantified metric for the scaling system in a cloud computing environment.

2. Literature Survey

The literature specific to the study discusses the various auto scaling systems influenced by decisions coming from various models or techniques. Authors of [Chenhao Qu et al.2016] propose a fault tolerant model for hosting web application using spot instances. The scaling system is reactive in nature which uses fault tolerant semantics. The system optimizes the response time of requests and the cost associated with the instances. [Roy N et al.2011] deduce a time interval to change the resources using the prediction. The scaling system is proactive in nature which considers the cost factors which influence the behaviour of the system to scale up or down. [Anshul Gandhi et al.2014] uses a modelling engine which helps to categorize workload in order to assess the options to scale. The predictor component outputs a medium term prediction within a monitoring window. Authors conclude that the combination of both scale up and scale out serves the optimal scaling policy. [Tania Lordio-Botran et al.2014] reviewed the existing auto scaling systems and opined that it is advisable to focus efforts on proactive auto scaling systems by taking advantage of time series analysis techniques with capaibilities that can output a prediction as the quantified metric. [Anshuman Biswas et al.2015] propose a broker based architecture through which the user can request the resources and billed in terms of seconds for usage instead of cost per hour. The architecture is entrusted to create a private cloud where in the resources are leased via a public cloud. The prediction uses deadline constraint along with machine learning. Efforts are focussed on increasing the broker profit by also considering to reduce the user cost. According to [H. Shumway and David S. Stoffer2011] time series is the systematic approach to study time correlations. The study focusses mostly on mathematical and statistical questions. The observation made at fixed time intervals are discrete where as the observations made between interval [0,1] are termed as

continous. A plot for the time series can help tracing the adjacent time periods to reconstruct some hypothetical time series which may have produced a discrete sample. The plot basically has time as the horizontal axis and random variables(Xt) as the vertical axis. There can be different time series for different scenarios but the only distinguishing factor is the degree of smoothness. Authors of [Rodrigo N. Calheiros et al.2015] adapt ARIMA model for workload prediction and focus more on resource utilization and low QoS impact. The accuracy of the model is about 91 percent. [Yazhou Hu et al.2016] propose a scale up or scale down using a trigger strategy. The strategy adopts pattern matching technique and the performance of the trigger strategy is better than the threshold approach. Time series is used for the analysis of the monitoring data. [Joseph Doyle et al 2016] identify the metric compute unit seconds (CUS). It specifies the time a workload takes to complete. [Wei Fang et al.2012] predict workload using ARIMA model for proactive provisioning of cloud resources. The observation is that the CPU intensive application are handled well by the system. [Samuel A. Ajila et al.2013] evaluates techniques like Support vector machines, Neural networks and Linear regression with SLA metrics like throughput and response time. The workload pattern used is randomo in nature and its simulated in a realistic fashion. The Support Vector machine scores better than the other two techniques. [Anshuman Biswas et al.2014] proposes a scaling technique which behaves proactively and adapts the resources based on the system load. Support vector machine scores better than the linear regression. ARIMA has been used for prediction because previous research have presented that web workloads tend to have strong correlation between them as specified by [G.Urdaneta et al.2009].

3. Existing System

The system uses the Wikipedia traces obtained from the Wikimedia foundation. The request traces as well as the project wise count is open to access. Using this data as the historical information for the Wikipedia application, ARIMA model is fit to it and prediction is obtained.

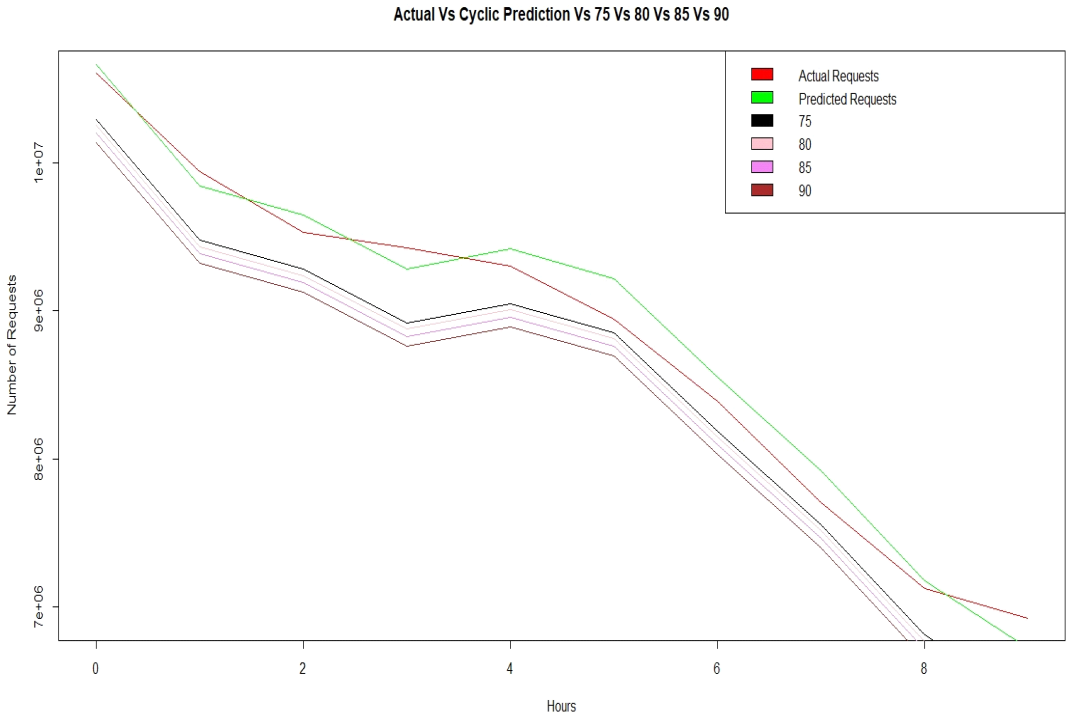

Figure 1. Actual Vs Predicted Vs 75 Vs 80 Vs 85 Vs 90 Forecasts for Short Term

The prediction is obtained as two types, they are, one time forecast and cyclic forecast. One time forecasts are obtained at one shot after training the model whereas the cyclic forecasts take into account the actual value at the end of each hour and consider the same for the next round of prediction. The existing system focus on analyzing the impact of one time forecast in comparison with the cyclic forecasts take along the 80% confidence level. The inference is that the cyclic forecasts serve as a better alternative when dealing with systems that require forecasts at regular intervals. The entire process of procuring the historical information, fitting the model and forecasting the next hour prediction are done entirely by statistical engine.

Table 1. Table Showing Short Term Actual, Predicted and Lower Edge of 75% Confidence Level

Actual	Predicted	Low75% level
10605152	10660567	10293095
9938548	9842539	9475422
9525577	9646089	9279303
9424458	9283901	8917425
9297976	9414700	9048514
8939574	9216971	8851098
8390395	8552662	8186876
7703478	7918568	7553048
7121185	7172993	6807668
6921598	6720916	6355939

Table 2. Table Showing Medium Term Actual, Predicted and Lower Edge of 75% Confidence Level

Actual	Predicted	Low75% level
10308935	10931415	10526502
9916133	9418201	9012862
9547318	9778569	9373030
9502738	9374426	8969002
9128225	9512096	9106846
8753892	8739665	8334379
8076945	8473181	8068096
7380748	7519748	7114609
6636664	6859618	6454648
6447901	6086581	5681730

Table 3. Table Showing Long Term Actual, Predicted and Lower Edge of 75% Confidence Level

Actual	Predicted	Low75% level
9427713	9770008	9408071
8903183	8735665	8373709
8567638	8663677	8301797
8576782	8452242	8090460
8570379	8687713	8326021
8284930	8635730	8274131
7740688	8162843	7801218
7180763	7461457	7099746
6660319	7020915	6659227
6331705	6574302	6212581

4. Proposed System

Table 4. Table Showing MAPE Scores for 75% Level Compared with Actual Values and Predicted Values

	Actual Value MAPE	Predicted Value MAPE
75% level Short Term forecasts	3.79984	4.410345
75% level Medium Term forecasts	4.532267	5.061852
75% level Long Term forecasts	2.266137	4.669871

The proposed system uses the functionality of the afore mentioned statistical engine to obtain the forecasts along with different confidence levels viz 75,80,85 and 90%. The historical data for training the ARIMA model is not available readily for use. Hence as a pre-requisite the raw data from the Wikimedia foundation is obtained and it is analyzed to obtain the number of requests for the English Wikipedia resources. The training data for the model spans across 3 weeks of January 2011 from 01 to 21 used for the short term forecasts, 6 weeks of data between January 01 2011 to February 11 2011 used for medium term forecasts and at last 10 weeks of data between January 01 2011 to Mar 11 2011 used for long term forecasts. The test data is the actual values for the number of requests observed in the next consecutive 9 hours for the data used for short, medium and long term respectively. The values chosen for analysis are the point forecast as well as the lower level of the 75% confidence level. For short term this is mentioned in Table 1. For medium term this is mentioned in Table 2 and for the long term this is mentioned in the Table 3. The request count with the training data period is transformed to a time series. Once the historical data is available for use with the statistical engine then the system uses it for forecast. For the purpose of this study forecasts are outlined as short term, medium term and long term forecasts. Each forecasts is obtained across 4 difference confidence level in addition to the point forecast.

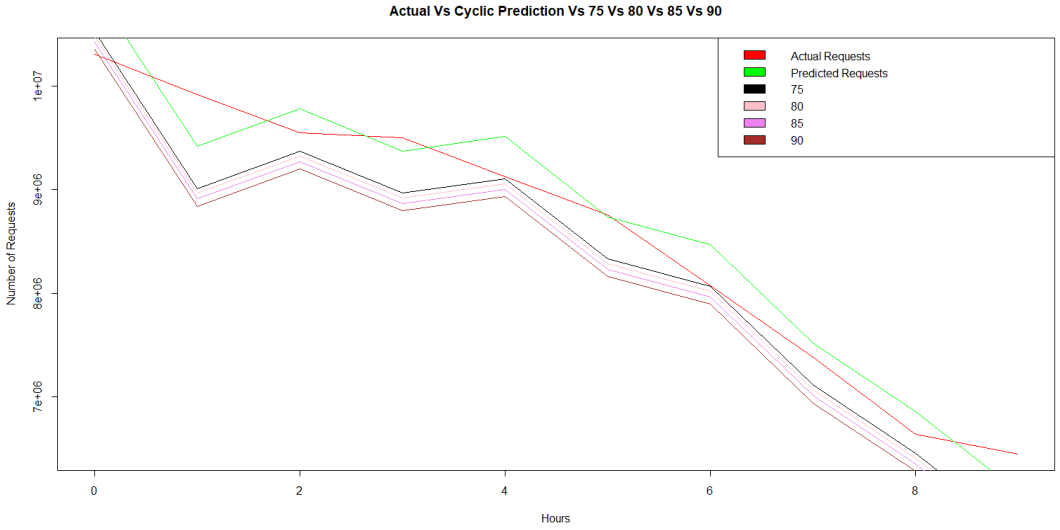

Figure 2. Actual Vs Predicted Vs 75 Vs 80 Vs 85 Vs 90 Forecasts for Medium Term

The confidence level used are 75,80,85 and 90% respectively. Mean Absolute Percentage Error (MAPE) is used to measure the prediction accuracy of forecast. Its wide usage in statistics helps deducing the accuracy of forecast. The accuracy is expressed as a difference between the actual value and the forecast value divided by the actual value again. This value is again summed up for each forecast point and divided by the number of fitted points on which the forecast was based on. The accuracy of the forecasts is shown in Table 4.

5. Results and Discussion

Analysing the accuracy of the forecasts and the lower end of confidence levels with the actual values. For values along the lower end of the 75% confidence level the MAPE scores are lower with actual values than the MAPE scores for the predicted values. For clarity the mape scores for 80%, 85% and 90% are also analyzed and compared to the same. This observation is true across different timelines, be it, short term or medium term or long term. Looking at the lower end of 75% confidence level values it is adapting towards the actual values of the workload requests.

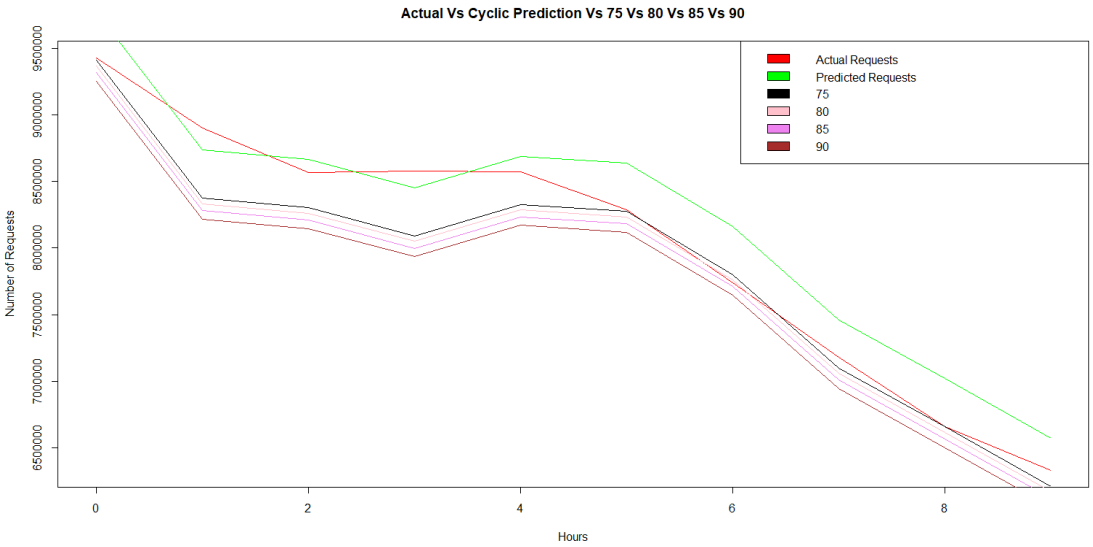

Figure 3. Actual Vs Predicted Vs 75 Vs 80 Vs 85 Vs 90 Forecasts for Long Term

For short term and medium term, the predicted values are better adapting to the actual values followed by the lower 75% level, this is evident from Figure 1 and Figure 2. Whereas for the long term, the lower end of the confidence level 75% is far better than the predicted values which shows a more adapting trend towards the actual values as shown in Figure 3. Thus, when using forecasts with a substantial amount of historical data available to

train the model, it is better to look closely the MAPE scores of lower end of 75% confidence level in comparing with predicted values from the model.

6. Conclusion and Future Work

Auto scaling systems helps tackling the dynamic requirements during resource utilization. These systems when equipped with proactive behaviour tend to analyse historical information for a particular application in order to draw insights out of a prediction model. The usage of prediction can directly affect the scaling decision which signifies the need to fine tune it before usage. The analysis done here by forecasting across varied timelines along with different confidence levels helps establishing a clear cut idea about how forecasts behave when there are different trends in historical data and also how the confidence level plays a significant role in establishing the lower bounds of the forecast. The accuracy comparison here best aligns the appropriate confidence level that comes closer to the actual values which sometimes could serve as a better alternative than the predicted values. This finding can further be extended to tackle the scaling policies by associating it with the spot instances. Spot instances have been a low cost alternative and the dynamic changes in the spot market based on the demand supply have forced auto scaling systems to be smarter. This means the scaling system must consider varied factors like calculating optimal bid price, choosing amongst heterogeneous resources and placing request in the appropriate time. The future work is to tackle such factors associating the proactive scaling strategy with spot instances.

References

1. Shumway, H., & David, S. Stoffer. (2011). Robert H. Shumway and David S. Stoffer Time Series Analysis and Its Applications with R Examples, Springer Texts in Statistics, Third Edition.
2. Roy, N., Dubey, A., & Gokhale, A. (2011). Efficient autoscaling in the cloud using predictive models for workload forecasting. In IEEE 4th International Conference on Cloud Computing, 500-507.
3. Qu, C., Calheiros, R.N., & Buyya, R. (2016). A reliable and cost-efficient auto-scaling system for web applications using heterogeneous spot instances. Journal of Network and Computer Applications, 65, 167-180.
4. Gandhi, A., Dube, P., Karve, A., Kochut, A., & Zhang, L. (2014). Modeling the impact of workload on cloud resource scaling. In IEEE 26th International Symposium on Computer Architecture and High Performance Computing, 310-317.
5. Hu, Y., Deng, B., Peng, F., & Wang, D. (2016). Workload prediction for cloud computing elasticity mechanism. In IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), 244-249.
6. Ajila, S.A., & Bankole, A.A. (2013). Cloud client prediction models using machine learning techniques. In IEEE 37th Annual Computer Software and Applications Conference, 134-142.
7. Calheiros, R.N., Masoumi, E., Ranjan, R., & Buyya, R. (2014). Workload prediction using ARIMA model and its impact on cloud applications' QoS. IEEE transactions on cloud computing, 3(4), 449-458.
8. Doyle, J., Giotsas, V., Anam, M.A., & Andreopoulos, Y. (2016). Cloud Instance Management and Resource Prediction for Computation-as-a-Service Platforms. In IEEE International Conference on Cloud Engineering (IC2E), 89-98.
9. Fang, W., Lu, Z., Wu, J., & Cao, Z. (2012). Rpps: A novel resource prediction and provisioning scheme in cloud data center. In 2012 IEEE Ninth International Conference on Services Computing, 609-616.
10. Lorido-Botran, T., Miguel-Alonso, J., & Lozano, J.A. (2014). A review of auto-scaling techniques for elastic applications in cloud environments. Journal of grid computing, 12(4), 559-592.
11. Biswas, A., Majumdar, S., Nandy, B., & El-Haraki, A. (2014). Automatic resource provisioning: a machine learning based proactive approach. In IEEE 6th International Conference on Cloud Computing Technology and Science, 168-173.
12. Raj Kumar, R., & Iyengar, N.C.S. (2014). Secure and synchronised mobile JXTA cloud ecosystem for sharing patient's healthcare information and medical reports. International Journal of Computers in Healthcare, 2(1), 15-27.
13. Biswas, A., Majumdar, S., Nandy, B., & El-Haraki, A. (2015). Predictive auto-scaling techniques for clouds subjected to requests with service level agreements. In IEEE World Congress on Services, 311-318.
14. Urdaneta, G., Pierre, G., & Van Steen, M. (2009). Wikipedia workload analysis for decentralized hosting. Computer Networks, 53(11), 1830-1845.