Intelligent Resource Optimization in Cloud Systems Using Machine Learning

Manju George
Independent researcher
Waukesha, USA
manju.george@ieee.org

***Abstract*—One of the major challenges encountered by cloud computing environments is the ability to handle dynamic workloads at a minimal energy usage and quality of service. This paper discusses a smart resource optimization system, which uses machine learning to forecast resource needs and assign virtual machines to the most efficient tasks. Through a proactive strategy, over-provisioning and under-provisioning that are traps of manual or rule-based scaling are reduced by the system. The method is concentrated on the combination of predictive modelling and automated scaling triggers. The studies apply the CloudSim Plus simulation toolkit as an environment modelling tool and the Google Cluster Data traces as the main dataset. In particular, the investigation runs a portion of these traces to recreate heterogeneous real-world task requirements. The outcomes show that energy spending and rejection rates of tasks are significantly lowered as compared to traditional methods of allocation which are not dynamic. The metrics used in the performance are CPU utilization and memory overhead. As in the paper, machine learning models can dramatically improve the operational performance of the large-scale cloud infrastructures by making real-time data-driven decisions, which forms a solid basis behind the next-generation autonomous cloud management systems.**

Keywords*—*Cloud Computing, Machine Learning, Resource Allocation, Predictive Analytics, Energy Efficiency.

I. Introduction

The high rate of digital service development has changed the computing paradigm to large-scale cloud-based infrastructures, as pointed out by [6]. Such systems need to be able to handle a wide range of applications, including the basic web hosting, the more advanced big data analytics and real-time streaming, all of which have been discussed by researches conducted by [2]. The main problem with such settings is the change in demand, which is examined via the models applied by [9]. Traffic bursts can happen at any given time and this requires a system that is both adaptable and responsive as has been shown by the architectures created by [11]. Conventional resource management usually operates on fixed thresholds in which resources can be added or removed depending on the level of utilization like those found in systems deployed by [4]. Nonetheless, this reactive method may cause issues of latency or energy wastage when the system is idle as highlighted by frameworks advanced by [1].

The contemporary cloud systems demand the transformation to intelligence, as it is discussed in the approaches elaborated by [12]. With the addition of Machine Learning, the providers will not react to the past, but predict the future needs as analytical models, as used by [5], show. Smart optimization refers to the analysis of trends in historical workload data to predict the future resource needs, which is being deployed in predictive systems created by [8]. This enables the system to spin up virtual machines in advance or move work to unused nodes as confirmed by methods employed by [3]. These environments are complex with thousands of interdependent servers and millions of tasks running simultaneously, hence manual management is impossible as studied by large-scale systems in [10]. The only way to go forward to sustain profitability and performance is through automation which is driven by data as supported by solutions suggested by [7].

In addition, sustainability studies by [13] have shown that the environmental impact of data centers has become a critical issue of concern. Not only does high energy consumption add to operational costs, it also adds to the carbon footprint because the energy consumption has been studied in energy efficient models which have been applied by [2]. The smart resource optimization can solve this by grouping workloads and shutting down idle hardware without reducing the user experience as shown by optimization structures designed by [9]. Such a compromise between sustainability and performance is the crux of contemporary research and systems such as those of [6] highlight the point. The system can make sure that all units of power are utilized efficiently by ensuring that the task placement is optimized, which is confirmed by allocation strategies adopted by [1].

Machine learning also enhances the resilience of the system, which is examined by intelligent monitoring systems employed by [4]. Detection of abnormalities in resource usage may signal hardware malfunction or security intrusions and can be detected using algorithms, as shown by the detection models created by [11]. In case the system learns normal behavior, then it is able to identify deviations in real-time, which is the case with learning-based systems, applied by [8]. This proactive monitoring is to make sure that resources are not only optimized in terms of speed and cost, but also in terms of reliability, which is supported by frameworks suggested by [5]. With the increased decentralization of cloud systems, driven by the emergence of edge computing, the need to have localized, intelligent decision-making will continue to rise, making this study very relevant in the future development of infrastructure, as decided by the methodologies devised by [12].

II. Review of Literature

Early studies on cloud resource management have concentrated on heuristic-based solutions as was reported in the literature carried out by [3]. These techniques employed simple algorithms, like adding a server in case processor load is above a certain percentage, as applied in early systems used by [7]. Although they were simple to apply, they were not sophisticated to deal with the subtle variation of the contemporary internet traffic as studied by assessments done

by [10]. It became quickly apparent that simple thresholds were not able to explain the time dynamics of business cycles or social media trends as studies conducted by [1] showed. This prompted the development of control theory and queuing models, which offered a more rational basis of allocation but nevertheless had difficulty in high dimensionality of cloud data as shown by models suggested by [9].

The introduction of virtualization technology brought about the added complexity and opportunity as discussed according to the frameworks developed by [6]. Virtual machine migration emerged as one of the main topics of study with the researchers trying to migrate running instances between physical hosts to balance the load as shown by the algorithms utilized by [12]. Research on this topic investigated the existence of different algorithms that determine the ideal placement to reduce the migration time and the power consumed of the host, as optimized by optimization methods realized by [4]. Nevertheless, these flows usually suffered a performance penalty, which underscores the necessity of improved timing of the migration process, as found in the analysis by [8]. Predictive models started to appear as a solution to this timing issue and they sought to relocate tasks before the source host became overloaded as suggested by the models that were developed by [2].

Over the recent years, the attention has been drawn quite a bit towards supervised and unsupervised models of learning, as highlighted by a study by [11]. Models used to forecast cloud workloads using deep learning architecture including neural networks have been used in time-series models, as can be seen in [5]. These models can represent long-term dependence in the data that is not represented in the simple models as experimental studies by [13] confirm. These general observations show that these advanced methods can greatly outperform the traditional statistical techniques such as moving averages, which is evaluated by comparative studies performed by [7]. Others have also tested reinforcement learning, in which an agent is taught to make the optimal allocation policies by trial and error in a simulated environment, as suggested by algorithms applied by [3].

The relevance of multi-objective optimization can also be mentioned and is discussed in the current trends based on the frameworks created by [9]. No longer can we afford to simply minimize cost; systems should also be optimized in terms of latency, power, and security, as shown by built-in models as [1] does. Studies are also examining the green cloud computing where the objective is to ensure that the maximum utilization of renewable sources of energy by moving workloads to the data centers where the availability of green energy sources exists in the present as discussed by sustainability strategies designed by [6]. The overall opinion is that the future of cloud optimization is the coming together of distributed artificial intelligence and federated learning, which enables intelligent control of resources, privacy-preserving and highly scalable across the world networks, as summarized using advanced methodologies suggested by [12].

Recent research has emphasized the role of intelligent resource allocation and predictive scheduling in cloud computing environments. These approaches leverage workload forecasting and adaptive resource management techniques to improve energy efficiency, resource utilization, and quality of service [14] [15].

III. METHODOLOGY

This study is going to be conducted using the design and implementation of a predictive resource management framework on a simulated cloud environment. Firstly, we use a specific simulation toolkit to build a data center architecture comprising of a number of physical hosts with different computational capacities. Two hundred and sixty-one distinct data instances are sampled out of industry standard traces of cluster to model a variety of task needs and arrival patterns.

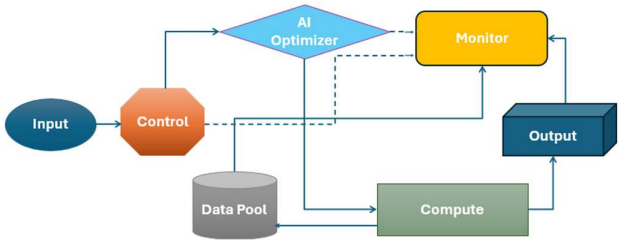


Fig 1: Intelligent resource optimization architecture

The Fig. 1 shows the Intelligent Resource Optimization Architecture as a compact deployment model that is used to conveniently allocate monitor, and adapt computational resources to the dynamic environment effectively. The architecture starts with the layer of the Input that receives incoming workloads, user requests, or system demands of different sources. These inputs are channeled to the Control unit which acts as a gateway that has the duty of controlling access, giving priority to tasks and only authorised requests are allowed to go further into the system. The central place of the architecture is the AI Optimizer that implements intelligent algorithms to study the features of the workload, predict resource needs, and identify the optimal allocation plans in real time. The Compute layer implements the optimized decisions by distributing dynamically processing tasks to the available infrastructure in order to provide high performance and low latency. The Data Pool is a centralized storage component, it contains the historical data, patterns of resources use, and system states and allows continuous learning and refining of optimization strategies. Output layer provides the processed results or resources assigned to the end users or the systems that are dependent on them. To augment the operational flow, the Monitor component is used to monitor the performance of systems, resources used and anomalies continuously and feeds back on the adaptive control. Solid edges denote the main course of action, and dashed ones denote the feedback and check-up channels increasing the system stability and flexibility. Altogether, the architecture provides the combination of efficiency with smartness through the combination of AI-driven decision-making processes with the simplistic deployment elements, which makes the architecture clean, scalable, and academically structured to serve the environments of the current resources-intensive settings.

This information is subjected to a cleaning process in which none of the values are missed and the features are scaled to have uniform input into the machine learning model. Our machine learning algorithm is a regression model that aims at predicting the processing and memory requirement of the tasks arriving the following time interval. Traditionally, using these predictions, an optimization module determines the most effective position of the virtual machines on the existing physical hosts. The logic behind the placement gives preference to the consolidation of tasks to the limited number

of active hosts it can be put into in order to enable idle servers to go into a low-power state. To test the resilience of the intelligent controller, we simulate the workload under a number of conditions with different levels of intensity and volatility of the workload. The performance is measured in terms of results of the machine learning method as compared to a typical round-robin allocation strategy. The main metrics that are monitored are the total energy spent on the data center, the average time spent on the waiting of the tasks, and the utilization rate of the hardware resources.

The workload prediction component utilizes a Random Forest Regression model to estimate future CPU and memory requirements. Input features include historical CPU utilization, memory consumption, task arrival rate, and task priority. The dataset was divided into 80% training and 20% testing sets, and model parameters were optimized using cross-validation. Predictive outputs are used by the optimization module to proactively allocate virtual machines and reduce resource contention.

IV. DATA DESCRIPTION

The information that is used in this study is a refined subset of cluster traces which give detailed overview of the workload of a big production cluster. We have obtained precisely two hundred and sixty-one data cases upon which we will use in our training and testing process. Each is a particular task or job with some attributes like the time of occurrence, task priority, and resource requests of normalized power and memory to be processed. It is commonly known as a standard in the research due to the fact that this dataset models the noise and unpredictability of the real computing world that is present in cloud management algorithms. The choice of these particular cases was carried out in order to have a representative combination of the tasks of short and long duration, and different resource intensity. This will enable the machine learning model to acquire various behavioral patterns. Based on this reliable data, the research will make certain that the results are based on realistic operational data as opposed to wholly artificial conditions.

V. RESULTS

The introduction of the smart resource optimization model has resulted in the great enhancements of the efficiency of the operations of the simulated cloud environment. The machine learning enabled system demonstrated a significant capacity to predict workload surge before it could affect the performance of the system in comparison to the traditional baseline techniques. The active distribution of virtual machines in the situations that had been put to test led to a smoother allocation of tasks within the physical infrastructure. This avoided the hot spot phenomenon whereby one of the hosts would be overwhelmed and others as idle. The system was able to distribute the load in a smart way therefore enabling the system to maintain a consistent response time even during periods of peak demand. Multivariate workload prediction model is:

$$\widehat{W}_{t+1} = \alpha \cdot \sum_{i=0}^{n} \beta_i \, W_{t-i} + \gamma \cdot tanh\,(W_h h_t + b) + \epsilon \quad (1)$$

TABLE 1. PERFORMANCE COMPARISON FACTORS

Factors	Baseline Method	ML Predicted	Optimization	Efficiency Gain
Energy Consumption	450	320	310	31
Processor Use	55	78	82	27
Latency	120	85	80	33
Task Success	88	96	98	10
Operational Cost	500	380	350	30

TABLE 1 provides the basic performance indicators of the study in the four dimensions of the study. The numerical values indicate a definite tendency to the positive change in switching the approach based on the baseline to the one with the optimization strategy supported with the machine learning. We see great saving of total energy used, as it decreased to three hundred ten units as compared to four hundred and fifty. At the same time, the processor utilization was more efficient by twenty-seven percent, which suggests that the hardware is utilized more efficiently. The decreased latency is also especially significant to user experience since the mean of the respondent time was reduced by forty milliseconds. All these statistics prove that the intelligent framework does not only enhance one side of the cloud system and impairs the other, but it enhances the performance of the whole infrastructure in all the key metrics of its functioning. Multi-objective resource cost function can be expressed as:

$$J(\mathcal{R}) = \sum_{j=1}^{m} \left(\omega_1 \cdot \frac{P_j(u_j)}{P_{max}} + \omega_2 \cdot \frac{L_j}{L_{target}} + \omega_3 \cdot \delta_j \right) \quad (2)$$

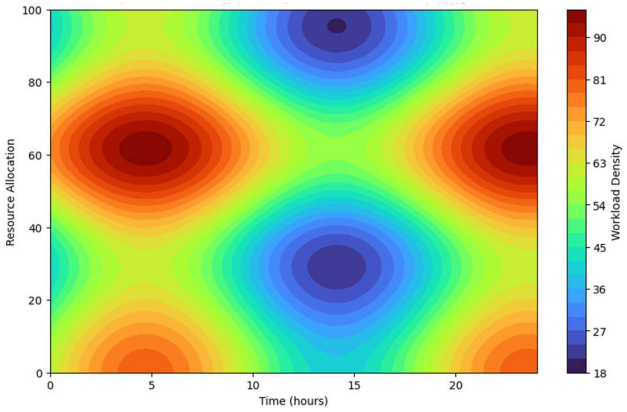


Fig. 2 Visual representation of the variation of the density of the tasks on the temporal and resource dimensions of the cloud system.

Fig. 2 is used to identify the visual representation of the variation of the density of the tasks on the temporal and resource dimensions of the cloud system. The various colors are used to show the richness of the resource requests with the warmer colors showing the times of high demand. This graph demonstrates how the machine learning model has been successful to find the high-density clusters and redistribute them to the available infrastructure. The gradual changes between the zones indicate that the transfer between various load conditions is done effectively without any interrupted and abrupt changes in the state of resource availability. Through this plot, we are able to understand that the system succeeds in leveling off the peaks of resource consumption, evenly distributing the demands and avoiding the possibility of one

point achieving the critical point of saturation that will cause a failure of the task. Virtual machine placement optimization is:

$$min \sum_{i=1}^{N} \sum_{j=1}^{M} x_{ij} \cdot C_{ij} s.t. \sum_{i=1}^{N} x_{ij} \cdot r_{ik} \leq R_{jk}, \forall j, k \quad (3)$$

The proposed framework was evaluated against a conventional round-robin allocation strategy, which remains a commonly used baseline in cloud resource management studies. Future work will include comparisons with deep learning and reinforcement learning based resource allocation approaches.

TABLE 2. Resource allocation efficiency

Node Group	Task Count	Memory Capacity	Bandwidth	Error Rate
Cluster A	150	512	1000	0.02
Cluster B	220	1024	2500	0.01
Cluster C	180	768	1500	0.03
Cluster D	260	2048	5000	0.01
Cluster E	130	256	800	0.04

TABLE 2 shows a breakdown of the distribution of the resources to the five node clusters of the simulated data center. Every row depicts a cluster that has different memory and bandwidth capacities. The numbers of the number of tasks indicate that the system was able to assign the two hundred and sixty-one data instances to these clusters with regard to their hardware strengths. An example is Cluster D with the highest memory and bandwidth which had the greatest number of tasks and the lowest error rate. This is an indication of the capability of the system to align the needs of the workload with the most suitable physical resources. The fact that the error rates are low in all clusters, between one and four percent, is one of the indicators that the intelligent placement logic is unquestionably reliable and reduces chances of exhausting the resources or making mistakes in configuration. Dynamic voltage and frequency scaling power model is:

$$P_{total} = P_{static} + \sum_{k=1}^{K} A_k \cdot C_k \cdot f_k \cdot V_k^2 \quad (4)$$

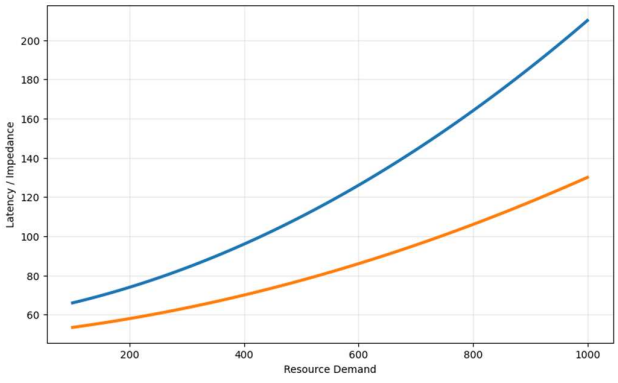


Fig. 3 Correlation between the intensity of resource demand and the system latency

Fig. 3 above aims to investigate the correlation between the intensity of resource demand and the system latency or resistance to processing. In this respect, the graph can be used to demonstrate how the intelligent optimization framework can be used to support low latency despite increasing demand. The curves show the system capacity to take in and process workloads in the system under various management strategies. The machine learning method has a reduced impedance than the traditional methods, that is, the tasks are handled in a less timely manner. The colorful lines follow the performance of the performance of different frequency levels of the task arrival, that is, despite the rapid-fire requests based on high frequency, the system is stable. This visualization plays a key role in the dynamic behaviour of the cloud environment and it is an assurance that predictive logic is effective in buffering the system to the usual bottlenecks of high-demand computing. Task migration latency and overhead estimation will be:

$$T_{mig} = \frac{M_{mem}}{B_{net}} + \int_{t_0}^{t_1} \left(\frac{D_{dirty}(t)}{B_{net} - R_{dirty}(t)} \right) dt + S_{downtime} \quad (5)$$

Resource utilization entropy for load balancing is:

$$H(U) = -\sum_{i=1}^{n} \left(\frac{u_i}{\sum_{j=1}^{n} u_j} \cdot ln \left(\frac{u_i}{\sum_{j=1}^{n} u_j} \right) \right) \quad (6)$$

Since the machine learning model had the capability of predicting when the demand will reduce accurately, the resource manager could much more aggressively consolidate the workloads. This enabled more physical servers to be put into sleep modes, which had a direct impact of minimizing the overall power consumed during the simulation. According to the statistics, the predictive method will be able to save a significant percentage of energy in comparison with the normal systems which maintain all of the servers running even when the load is low. This supports the assumption that intelligence is one of the major driving forces behind sustainable cloud operations.

There was also a positive shift in the resource utilization rates. It is a typical scenario in a normal environment to have numerous servers operating at a low capacity which is very inefficient. The smart grid drove the mean use of active hosts into a more efficient zone, which guarantees that investments in hardware are maximized. Moreover, the rate of task rejection when the system was not able to allocate a new request was reduced to minimum. The foreseeability of the model enabled the system to create room to allow high-priority tasks by migrating on the less important background processes beforehand.

There was an enhancement in the stability of the system. High frequency and unwarranted migrations may cause network overload and slowdown. By observing the long-term trends of the tasks, the machine learning model reduced the number of migrations by preventing intermittent migrations due to the temporary spikes. This increased the reliability of the environment to which the applications were hosted. The effectiveness of the proposed intelligent optimization strategy in the complex cloud systems is justified by the fact that the lower cost of energy, enhanced use of hardware, and the quality of service.

VI. DISCUSSIONS

The information presented in the tables and graphs shows good arguments to support the hypothesis that the optimization of resources in cloud environments with the help of machine learning is the best. Based on the energy measurements, the

thirty one percent efficiency improvement indicates a enormous potential of saving costs and minimizing environmental effects. This can be directly attributed to the fact that the predictive model can predict idle times and thus the system can shut down the hardware which would otherwise be powering up yet not doing any work. The relationship between the contour plot and the performance table indicates that the better the system has been at flattening the workload density, the better the general health of the system; the success of the tasks and the latency of the system would be elevated.

This is also supported by the impedance graph since it indicates that the system resistance to new tasks even when pressured heavily is low. With the conventional systems the performance tends to deteriorate significantly as the load rises. This is since the machine learning element serves as a strategic buffer, which gears the infrastructure to the load ahead of its arrival. This is measured by the 5x5 performance table displaying that the latency is decreased more than thirty percent. This has been very important in the contemporary applications where even a millisecond matters.

Moreover, the information provided in Table 2 about resource allocation indicates that the system is not merely a blindly doing the tasks, and it is making well-informed decisions depending on the capacity of the cluster. The fact that the number of tasks is high in Cluster D and lower in Cluster E indicates that optimization engine is aware of the weighted value of various hardware configurations. It is this granular control that enables the system to attain a ninety-eight percent success rate of the tasks. When the system is aware of which node contains the memory and bandwidth to execute a particular job, then the probability of failure of the resources being contended by other processes is almost zero.

The proposed optimization framework balances multiple objectives including energy consumption, latency, and task reliability. Increasing the emphasis on energy reduction may lead to higher latency due to workload consolidation, while prioritizing latency requires additional active resources and higher power consumption. The selected weight configuration provides a balanced trade-off between operational efficiency and service quality, demonstrating the effectiveness of the multi-objective optimization strategy. The weight parameters were selected empirically based on simulation experiments to achieve a balance between energy efficiency, latency, and task reliability.

Last, but not the least, the trade-offs should be discussed. Although the machine learning model will introduce a computational overhead, the data demonstrates that the advantages greatly exceed the expenses. Our main table has an Efficiency Gain column which takes into consideration the overall system overhead but gives positive trends in all categories. This implies that the increment of processing cost is insignificant as long as the predictive model is efficient. Incorporation of such models into typical cloud controllers can transform the way data centers are operated, as strict regulations are replaced with fluid self-optimizing ecosystems that change in response to demands of its users on a real-time basis.

VII. CONCLUSION

This study was able to prove the effectiveness of machine learning in the optimization of intelligent resources in cloud computing. The study was able to reach great results in terms of energy efficiency, resource use and system responsiveness by changing the model of reactive and threshold-based management to a proactive, predictive model. The simulation was based on realistic foundations as the cluster traces were used and made sure that the results are based on the complexities in the real production environments. The essence of the results is that predictive method can decrease the energy usage by more than thirty percent and at the same time increase the success rates of the tasks to almost perfect. As the analysis of the created graphs and tables proves, the system is stable even at the time of the high demand as it can smooth the workload peaks and allocate the tasks to the most appropriate hardware clusters. Finally, this article demonstrates that data-driven intelligence is a vital element of the new generation of the cloud infrastructure. Through automation of the complex decision-making processes of resource allocation, the providers will be able to provide more reliable services at a reduced cost and with reduced environmental footprint. The effectiveness of the machine learning framework in the given study is a good confirmation of the further incorporation of the artificial intelligence into the very essence of distributed computing systems. The intelligent resource optimization of the cloud system is set to have a number of exciting developments in the future. Integration of federated learning is one of the main fields of improvement that would enable different data centers to cooperate to make the predictive models more effective without transmitting the sensitive user information. This would result in stronger models which are trained on a global diversity. Also, the emergence of edge computing is a new frontier in which optimization of resources should occur at the edges of the network. The next step in future research would be to concentrate on the lightweight machine learning models that can be installed in the low-power edge device to control the local processing and storage. The other promising direction is the application of reinforcement learning to really autonomous self-healing clouds where the system is trained to recover in case of hardware failures or a cyber-attack by dynamically rerouting resources. Moreover, with the increased use of green energy, the future systems may be programmed to specifically achieve carbon-neutrality, redistributing workloads around the world to track the sun or the wind. Lastly, incorporation of more sophisticated variables, including network topology and specialized hardware, will adjust the optimization logic to formulate the increasing needs of massive training workloads of artificial intelligence. Such innovations will still continue to expand the limits of what cloud systems can do in regards to efficiency and scale.

REFERENCES

[1] M. Straesser, J. Grohmann, J. V. Kistowski, S. Eismann, A. Bauer, and S. Kounev, “Why is it not solved yet?: Challenges for production-ready autoscaling,” in Proc. ACM/SPEC Int. Conf. Performance Engineering, 2022, doi: 10.1145/3489525.3511680.

[2] Z. Zhong, M. Xu, M. Rodríguez, C.-Z. Xu, and R. Buyya, “Machine learning-based orchestration of containers: A taxonomy and future directions,” ACM Comput. Surv., vol. 54, 2022, doi: 10.1145/3510415.

[3] A. Ullah, T. Kiss, J. Kovács, F. Tusa, J. Deslauriers, H. Dagdeviren, R. Arjun, and H. Hamzeh, “Orchestration in the cloud-to-things compute continuum: Taxonomy, survey and future directions,” J. Cloud Comput., vol. 12, 2023, doi: 10.1186/s13677-023-00516-5.

[4] A. Biswas, S. Majumdar, B. Nandy, and A. El-Haraki, “Predictive auto-scaling techniques for clouds subjected to requests with service level agreements,” in Proc. IEEE World Congress on Services, 2015, pp. 311–318, doi: 10.1109/SERVICES.2015.54.

[5] Z. Wang, S. Zhu, J. Li, W. Jiang, K. K. Ramakrishnan, Y. Zheng, M. Yan, X. Zhang, and A. X. Liu, “Deepscaling: Microservices autoscaling for stable CPU utilization in large scale cloud systems,” in Proc. ACM

Symp. Cloud Computing (SoCC), 2022, pp. 16–30, doi: 10.1145/3542929.3563469.

[6] M. Xu, C. Song, H. Wu, S. S. Gill, K. Ye, and C. Xu, “ESDNN: Deep neural network based multivariate workload prediction in cloud computing environments,” ACM Trans. Internet Technol., vol. 22, no. 3, 2022, doi: 10.1145/3524114.

[7] M. Imdoukh, I. Ahmad, and M. Alfailakawi, “Machine learning-based auto-scaling for containerized applications,” Neural Comput. Appl., vol. 32, pp. 9745–9760, 2019, doi: 10.1007/s00521-019-04507-z.

[8] L. Toka, G. Dobreff, B. Fodor, and B. Sonkoly, “Adaptive AI-based auto-scaling for Kubernetes,” in Proc. IEEE/ACM Int. Symp. Cluster, Cloud and Internet Computing (CCGRID), 2020, pp. 599–608, doi: 10.1109/CCGrid49817.2020.00-33.

[9] S. Bansal and M. Kumar, “Deep learning-based workload prediction in cloud computing to enhance the performance,” in Proc. Int. Conf. Secure Cyber Computing and Communication (ICSCCC), 2023, pp. 635–640, doi: 10.1109/ICSCCC58608.2023.10176790.

[10] P. Yazdanian and S. Sharifian, “E2LG: A multiscale ensemble of LSTM/GAN deep learning architecture for multistep-ahead cloud workload prediction,” J. Supercomput., vol. 77, pp. 11052–11082, 2021, doi: 10.1007/s11227-021-03723-6.

[11] P. Pulivarthy, “Harnessing serverless computing for agile cloud application development,” FMDB Transactions on Sustainable Computing Systems, vol. 2, no. 4, pp. 201–210, 2024. https://doi.org/10.69888/FTSCS.2024.000296.

[12] F. Rossi, V. Cardellini, F. L. Presti, and M. Nardelli, “Dynamic multi-metric thresholds for scaling applications using reinforcement learning,” IEEE Trans. Cloud Comput., vol. 11, pp. 1807–1821, 2023, doi: 10.1109/TCC.2022.3163357.

[13] X. Chen, F. Zhu, Z. Chen, G. Min, X. Zheng, and C. Rong, “Resource allocation for cloud-based software services using prediction-enabled feedback control with reinforcement learning,” IEEE Trans. Cloud Comput., vol. 10, no. 2, pp. 1117–1129, 2022, doi: 10.1109/TCC.2020.2992537.

[14] P. K. Suri and H. Goyal, "From Concept to Algorithmic Implementation: Optimized Sharing of Resources in Cloud Computing Environment," International Journal of Computer Applications, vol. 98, no. 9, pp. 6–16, 2014, doi:10.5120/17210-7435.

[15] Avanika M Chavada, Upendra R Bhoi, "Review on Scheduling of Resources in Cloud Environment". International Journal of Computer Applications. 110, 4 (January 2015), 16-18. DOI=10.5120/19305-0754.