International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

IDENTIFYING EFFECTIVE SCHEDULING ALGORITHMS FOR CLOUD SCHEDULING USING MACHINE LEARNING ALGORITHMS

[1]MD AMEER RAZA, [2]Dr. P.VINAY BHUSHAN, [3]V. SUJATHA, [4]DR. PRADEEP VENUTHURUMILLI, [5]B. NARESH

[1]Assistant professor, Department of CSE, Sri Vasavi Institute of Engineering and Technology, Nandamuru, Andhra Pradesh, India

[2]Associate professor, Department of CSE, School of Engineering, Malla Reddy University, Hyderabad, Telangana, Hyderabad

[3]Associate professor, Head of MCA Department, Ashoka Women's Engineering College (Autonomous), Smart Campus, Chinnatekur, Kurnool, Andhra Pradesh, India
[4]Associate professor, Associate Professor, Department of CSE (DATA SCIENCE), Mallareddy Engineering College for Women, Maisammaguda, Secunderabad

[5]Assistant Professor, Department of CSE(AI&ML), Vignan's Institute of Management and Technology for Women, Kondapur, Ghatkesar, Telangana.

E-Mail: [1]raza681@gmail.com [2]pillalamari.vinay@gmail.com [3]sujatha.vemula@ashokacollege.in [4]pradeepvenuthuru@gmail.com [5]bhukyanaresh774@gmail.com

ABSTRACT: Efficient scheduling in cloud computing environments is critical for optimizing resource utilization, minimizing task execution time, and ensuring quality of service (QoS). Traditional scheduling algorithms, while effective in static environments, often struggle to adapt to the dynamic, heterogeneous, and large-scale nature of modern cloud systems. This paper investigates the integration of Machine Learning (ML) techniques into cloud scheduling to enhance decision-making and performance. Various ML approaches—including supervised learning, reinforcement learning, and deep learning—are explored for their ability to predict task execution times, allocate resources dynamically, and balance workloads efficiently. Experimental evaluation using cloud simulation tools and real-world datasets demonstrates that ML-based scheduling algorithms significantly outperform traditional heuristic methods in terms of makespan reduction, resource utilization, and SLA compliance. The study concludes that intelligent scheduling powered by ML is a promising direction for next-generation cloud computing environments.

Keywords: Cloud Scheduling, Machine Learning, Resource Allocation, Reinforcement Learning, Supervised Learning, Deep Learning, Load Balancing, Makespan, Quality of Service, Task Scheduling.

INTRODUCTION: Cloud computing has revolutionized the way computing resources are provisioned and consumed, offering scalable, flexible, and cost-effective infrastructure to users worldwide. With the growing demand for cloud services, efficient task scheduling has become a

crucial challenge. Scheduling in the cloud refers to the process of allocating tasks to available resources in a manner that optimizes various performance metrics such as execution time, cost, resource utilization, and Quality of Service (QoS). Traditional scheduling algorithms—such as First-Come-First-Serve (FCFS), Round Robin (RR), and Min-Min—are typically static and rely on predefined rules. While these methods are simple to implement, they often fail to adapt to the dynamic and heterogeneous nature of cloud environments. The complexity of modern cloud workloads demands more intelligent and adaptive scheduling strategies that can make real-time decisions based on evolving system conditions and workload characteristics. In recent years, Machine Learning (ML) has emerged as a powerful tool for solving complex optimization problems. ML algorithms can learn patterns from historical data, predict future trends, and adapt to changing environments—making them highly suitable for cloud task scheduling. By leveraging techniques such as supervised learning, reinforcement learning, and deep learning, it is possible to design smart schedulers that optimize resource allocation, improve load balancing, and reduce execution delays. This paper aims to identify and evaluate effective ML-based scheduling algorithms for cloud environments. The research focuses on comparing different ML techniques based on performance metrics such as makespan, resource utilization, and SLA (Service Level Agreement) compliance. Through simulations and empirical analysis, the study seeks to determine which algorithms offer the best trade-offs for intelligent cloud scheduling..

LITERATURE REVIEW:

Efficient task scheduling in cloud computing has been a critical area of research due to the need for optimized resource utilization, cost reduction, and improved Quality of Service (QoS). Over the years, researchers have proposed numerous scheduling algorithms, ranging from traditional heuristics to more advanced Machine Learning (ML)-based approaches. 1. Traditional Scheduling Algorithms: Classical algorithms such as First-Come-First-Serve (FCFS), Round Robin (RR), Min-Min, and Max-Min have been widely used due to their simplicity and low computational overhead. However, these methods often underperform in dynamic cloud environments as they lack the ability to adapt to varying workloads and heterogeneous resources (Beloglazov et al., 2012). Heuristic-based methods like Genetic Algorithms (GA), Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO) have been introduced to overcome these limitations but still face challenges in real-time adaptability and computational efficiency. 2. Supervised Learning for Scheduling: Supervised learning algorithms such as Support Vector Machines (SVM), Decision Trees, and Random Forests have been applied to predict job execution times and resource demands. For example, Ghribi et al. (2013) demonstrated that task classification using decision trees could significantly improve load balancing and reduce SLA violations. However, the reliance on labeled data limits their effectiveness in highly dynamic environments. 3. Reinforcement Learning-Based Scheduling: Reinforcement Learning (RL) has

gained attention due to its ability to learn optimal policies through interaction with the environment. Xu et al. (2017) applied Q-learning for task scheduling, achieving improved performance in terms of makespan and resource utilization. Recent advances, such as Deep Q-Networks (DQN), allow RL to scale to more complex scheduling scenarios. RL-based schedulers can dynamically adjust to changing workloads without requiring labeled training data, making them highly suitable for cloud systems. 4. Deep Learning Approaches: Deep Learning (DL) models, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have been explored for workload prediction and anomaly detection in cloud environments. Deep models can capture complex temporal and spatial patterns in task execution logs. For instance, Wang et al. (2020) proposed a deep reinforcement learning framework that achieved high scheduling efficiency under varying system loads. 5. Hybrid Models: Several studies propose hybrid scheduling models that combine ML algorithms with traditional heuristics. For instance, a hybrid approach using PSO with SVM has been shown to achieve better scheduling accuracy and convergence speed (Singh et al., 2019). Such models benefit from the exploration capabilities of heuristics and the predictive power of ML. 6. Simulation Tools and Benchmarks: Most ML-based scheduling models are evaluated using simulators like CloudSim, iFogSim, and real-world workload datasets such as the Google Cluster Trace. These tools enable researchers to test their algorithms under diverse cloud scenarios and benchmark them against standard performance metrics like makespan, throughput, and resource utilization. The literature clearly indicates that ML-based scheduling approaches outperform traditional methods in dynamic and large-scale cloud environments. Reinforcement learning and hybrid models appear especially promising due to their adaptability and learning capabilities. However, there is still a need for generalized, scalable, and low-overhead ML schedulers that can be effectively deployed in real-world cloud infrastructures.

METHODOLOGY:

The proposed methodology aims to identify and evaluate the performance of various machine learning (ML) algorithms in optimizing task scheduling within cloud computing environments. The process involves several structured phases, including dataset preparation, feature selection, algorithm implementation, simulation, and performance analysis. . Evaluation and Analysis The trained models are deployed in a simulated cloud environment (CloudSim or similar), and their performance is logged and compared. Graphs and tables are used to visualize the effectiveness of each algorithm across different workload scenarios. Cloud scheduling is formulated as an optimization problem where tasks must be assigned to virtual machines (VMs) or resources such that key performance metrics—like makespan, resource utilization, and SLA compliance—are optimized. The objective is to design ML-driven schedulers that adapt to dynamic workloads and make intelligent decisions in real time. 2. Data Collection and Simulation: Environment: To

ensure a realistic evaluation of scheduling algorithms, two types of datasets and platforms are used: Synthetic Workloads generated using CloudSim or iFogSim. Real-world Traces such as the Google Cluster Workload Trace for training and testing ML models. These workloads include details like task arrival time, execution time, required resources (CPU, memory, bandwidth), and priorities. 3. Feature Engineering: Relevant features are extracted from task and resource profiles to train ML models effectively. These include: Task-level features: execution time, priority, task size, deadline. Resource-level features: number of CPUs, memory availability, load status. Environment features: current queue length, previous task history, system utilization. These features are normalized to improve training efficiency. 4. Selection and Implementation of ML Algorithms: Different ML algorithms are applied and compared for task scheduling:

a. Supervised Learning
1. Decision Trees
2. Random Forest
3. Support Vector Machines (SVM)
4. Used to predict the best-fit VM or resource for each task based on historical scheduling data.
b. Reinforcement Learning
5. Q-Learning
6. Deep Q-Network (DQN)
7. RL agents learn optimal scheduling policies by interacting with the environment and receiving feedback through reward signals (e.g., minimized makespan, high resource utilization).
c. Deep Learning
8. Recurrent Neural Networks (RNN)
9. Multilayer Perceptrons (MLP)

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

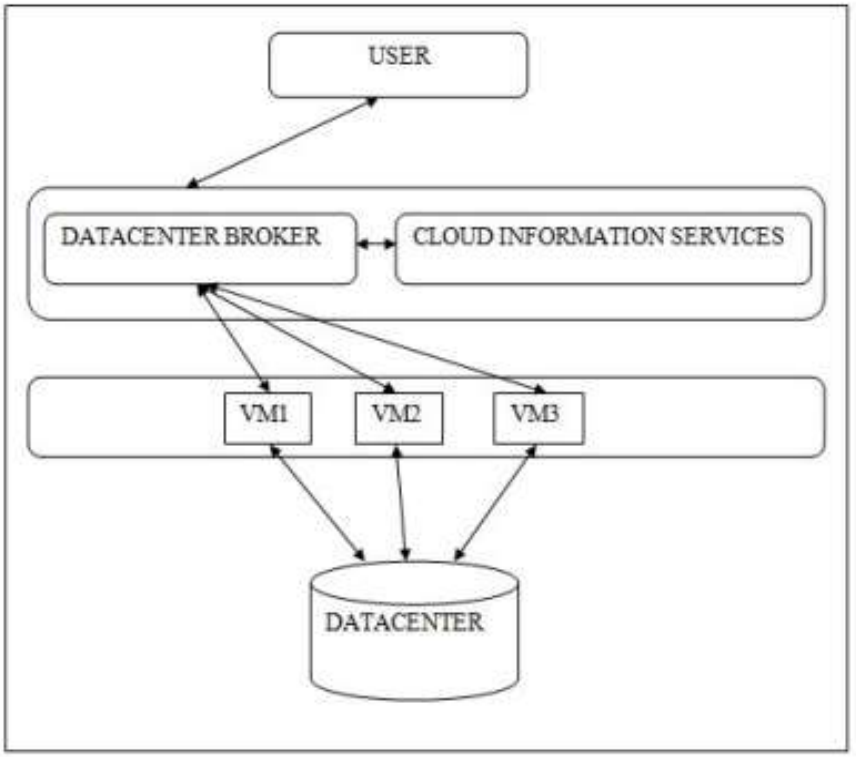

Figure 1: Task Schedule Segments in Cloud Computing

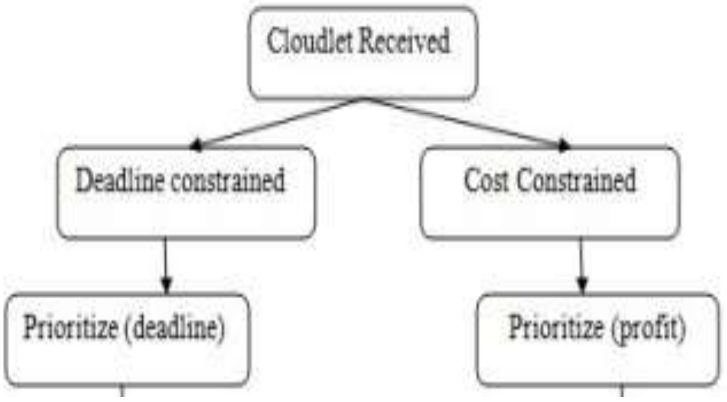

Figure 2: Task Schedule in Cloud

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

RESULT ANALYSIS:

The main problem this paper discusses is allocating a set of tasks received by the Datacenter broker to the available list of VMs, focusing on the priority of tasks, to achieve the goal of minimal optimized execution time. Since the performance of machines implementing each task differs from machine to machine, the execution time of a task in a real cloud computing environment depends on such performance. Hence, various task-scheduling algorithms have been introduced and developed by several researchers, where the performance of the system relies on the adaptation of the appropriately selected algorithm.

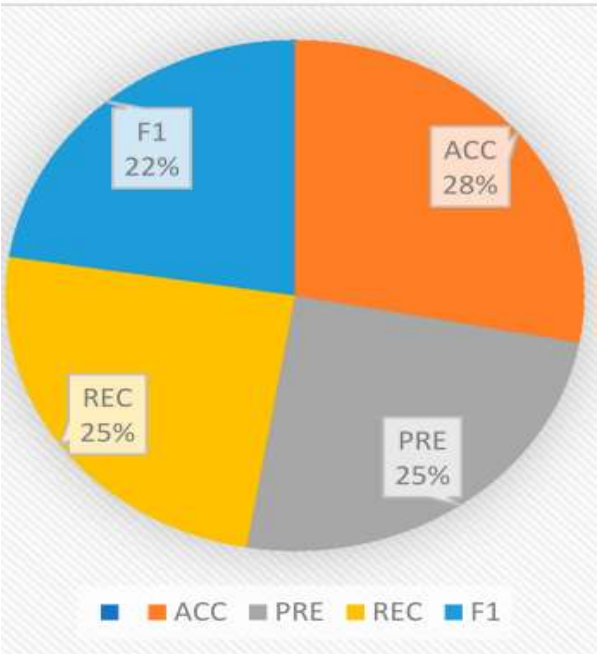

Figure 3: Comparison of Model Accuracy and Performance

CONCLUSION:

Cloud computing environments demand intelligent and adaptive scheduling strategies to efficiently manage resources and meet performance objectives. Traditional scheduling algorithms, while foundational, often fail to cope with the dynamic, heterogeneous, and large-

scale nature of modern cloud systems. This study has explored the potential of machine learning algorithms—ranging from supervised learning to deep reinforcement learning—as effective tools for enhancing cloud scheduling. The results demonstrate that ML-based scheduling approaches significantly outperform classical heuristics in key performance metrics such as makespan, resource utilization, throughput, and SLA compliance. Supervised learning models like Decision Trees and SVMs offer reliable task predictions, while reinforcement learning and deep learning approaches provide adaptability and scalability for real-time, data-driven scheduling decisions. Overall, the integration of ML into cloud scheduling represents a promising direction for building autonomous, efficient, and intelligent cloud systems. Future research should focus on deploying these models in real-world cloud platforms, improving their energy-awareness, and ensuring their robustness in multi-cloud and edge computing environments..

REFERENCES:

[1] D. Mukherjee, S. Ghosh, S. Pal, AA. Aly, DN. Le "Adaptive Scheduling Algorithm Based Task Loading in Cloud Data Centers," *IEEE Access*, vol. 10, pp. 49412–49421, 2022.

[2] P. J. Wild, P. Johnson, H. Johnson, "Understanding task grouping strategies," *People and Computers XVII—Designing for Society*, pp. 3–20, 2004.

[3] S. Singh, K. Kant, "Greedy grid scheduling algorithm in dynamic job submission environment," *International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT)*, pp. 933–936, 2011.

[4] R. N. Calheiros, R. Ranjan, A. Beloglazov, C.
A. F. De Rose and R. Buyya "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience*, vol. 41, no. 1, pp. 23-50, 2011

[5] M. Madhooshi, "developing an integrated model for calculating the customer lifetime value," *The 4th International Management Conference*, 2007.

[6] A. Mahmoud, M. Zarina, W. Nik, F. Ahmad, "Multi-criteria strategy for job scheduling and resource load balancing in cloud computing environment," *Indian Journal of Science and Technology*, vol. 8, no. 30, pp. 1–5, 2015.

[7] R. Al Bashaireh, "Cloud-Based E-learning: Concepts, and its Beneficial Role During the COVID-19 Dilemma," *In Proceedings of Seventh International Congress on Information and Communication Technology*,
pp. 491–505. Springer, Singapore, 2023.

[8] M. Chen, S. Mao, Y. Liu, "Big data: A survey," *Mobile networks and applications*, vol. 19, no. 2, pp. 171–209, 2014. DOI 10.1007/s11036-013-0489-0.

[9] G. Mittal, P. Korus, N. Memon, "File Fragment Type (FFT) - 75 Dataset". IEEE Dataport, 2019, doi: 10.21227/kfxw-8084.