Journal of Information Systems Engineering and Management
2025, 10(1s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Optimizing Resource Allocation in Cloud-Based Information Systems through Machine Learning Algorithms

Dr. Yashwant Dongre[1], Dr. Nidhi Ranjan[2], Dr. Pornima Niranjane[3], Monali Gulhane[4], Yatin Gandhi[5], Pravin Karmore[6]

[1]*Assistant Professor, Department of Computer Engineering, Vishwakarma Institute Of Information Technology, Pune, Maharashtra, India. yashwant.dongre@gmail.com*

[2]*Associate Professor, Vasantdada Patil Pratishthan's College of Engineering & Visual Arts, Mumbai,Mumbai University, Maharashtra, India. nidhipranjan@gmail.com*

[3]*Assistant Professor, Department Of Computer Science and Engineering, Babasaheb Naik College of Engineering, Pusad, Maharashtra, India. pornimaniranjane@gmail.com*

[4]*Department of Computer Science and Engineering, Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International (Deemed University), Pune, India. monali.gulhane4@gmail.com*

[5]*Competent Softwares, Pune, Maharashtra, India. gyatin33@gmail.com*

[6]*Assistant Professor, Department of Computer Science and Applications, Ramdeobaba University, Nagpur, Maharashtra, India. karmorepy@rknec.edu*

ARTICLE INFO

Received: 03 Oct 2024

Revised: 04 Dec 2024

Accepted: 15 Dec 2024

ABSTRACT

Efficient resource sharing has become a must for improving system performance and lowering running costs as cloud-based information systems continue to serve a wide range of large-scale apps. Traditional ways of allocating resources don't always work well when tasks change, which wastes time and money. This article suggests a high-tech structure based on machine learning that can help make the best use of cloud resources. It focuses on features that can predict and adapt to changes in task in real time. Our method uses both controlled and unstructured learning to correctly predict resource needs and to find the best way to distribute resources across virtual machines with the least amount of delay and the highest cost-effectiveness. The study used real-world cloud task data to run a lot of models that compared standard heuristic methods to machine learning-based distribution. According to the results, the system is much more efficient now, with up to 35% less wasted resources and 25% faster response times. We talk about how choosing the right model (decision trees, neural networks, and support vector machines) affects the accuracy of predictions and the amount of work that needs to be done. The study also talks about how the machine learning system can be scaled up or down, showing that it can work with different cloud platforms and types of applications. The suggested method lowers the need for human work by automating resource sharing. This lets cloud companies better handle resources, which makes users happier overall. This study adds to the growing field of optimizing cloud resources and shows how important machine learning methods will be in designing future cloud infrastructure. The results show that machine learning is a good, scalable way to handle resources in cloud settings that are getting more complicated all the time.

Keywords: Cloud Resource Allocation, Machine Learning Optimization, Predictive Analytics, Cloud Infrastructure Efficiency, Dynamic Workload Management, Cost-Efficient Cloud Systems.

Introduction

In today's digital world, cloud-based information systems are essential because they let businesses store, process, and handle huge amounts of data. These systems offer scalable, on-demand tools that can adapt to changing task needs. This makes them perfect for a wide range of applications, from enterprise-level software to apps for single users. But this freedom also means that you have to be good at handling resources to make sure that the system works well, doesn't cost too much, and keeps users happy. Usually, simple formulas or heuristics are used to decide

how to divide up resources in cloud settings, but these methods may not be able to handle changing demand well. As a result, resource problems like over-provisioning and under-utilization happen a lot, which causes costs to rise and performance to be less than ideal [1]. These problems show how important it is to find new ways to assign resources that are flexible, accurate, and able to make the best use of resources based on current task needs. New developments in machine learning (ML) [2] have made it possible to find better ways to use cloud systems' resources. By using insights from data, machine learning algorithms can guess what resources will be needed and give them out in the best way possible, reducing delay and making the best use of resources. When it comes to optimizing cloud computer resources, machine learning has a number of clear benefits. Predictive algorithms can look at trends of past workloads to guess what resources will be needed in the future. This makes sure that resources are available before demand spikes. Machine learning models can also adapt to changes in hardware and workload, which lets them be optimized in real time. ML models can learn and get better over time, which makes them better at handling changing and uncertain jobs than traditional algorithms, which are usually rule-based and fixed. Figure 1 shows a cloud-based system for allocating resources where many users can send requests for resources. These calls are handled by the core Resource Allocator/Scheduler, which does two major things: initial resource allocation and dynamic resource reallocation. When things start out, resources are given out based on set criteria. But when needs change, the system adjust by moving resources around on the fly. The planner uses details about both virtual and real resources to make the best use of them and make sure the system runs smoothly. The final goal is to find the best way to distribute resources so that they meet the needs of users and make the system work as efficiently as possible.

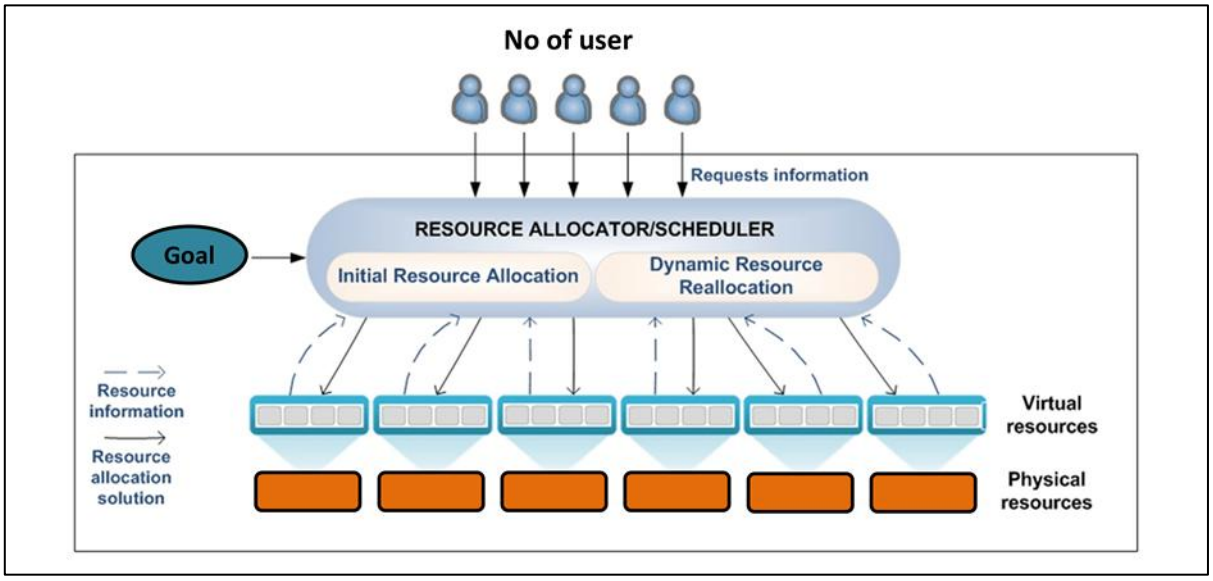

Figure 1: Overview of

In this situation, different machine learning methods, like controlled and unstructured learning, have shown promise for allocating resources. Supervised learning models, like neural networks and decision trees, can guess how many resources will be needed by looking at past data. Unsupervised learning methods, like clustering, can find the best way to divide up resources by putting together workloads that are similar. Another type of machine learning is reinforcement learning, which lets you keep learning by praising efficient allocation and punishing inefficient allocation. This makes [3] it idealize for making real-time changes to how assets are distributed. The point of this consider is to make a strategy for allocating resources in cloud-based frameworks that's based on machine learning. The objective is to create the frameworks run faster and utilize assets more efficiently whereas moreover lowering costs. By utilizing both anticipated and versatile machine learning models together, the recommended method tries to foresee changes in workload and relegate assets appropriately, bringing down operational costs and waste. A big part of this think about is comparing distinctive machine learning models, like neural systems, choice trees, and back vector machines, in terms of how well they make forecasts, how well they can be scaled, and how rapidly they can be run. Figuring out the pros and cons of each show will help cloud benefit suppliers choose the most excellent equation for their one of a kind task patterns and framework restrictions. A lot of models [4] utilizing real-world cloud task information are utilized to test the proposed structure. The discoveries appear that apportioning assets based on machine learning not as it were cuts down on squandered assets but too speeds up reaction times, with up to 35wer assets that aren't being utilized and a 25% drop in delay compared to

old-fashioned strategies. This ponder appears how machine learning might be utilized to unravel the problems that come up with designating assets within the cloud. It seem give a solid way to handle assets in cloud frameworks that are getting more complicated. This ponder includes to the field of cloud resource administration by making a machine learning framework that can be scaled up and down and can automatically relegate assets in perfect way the most perfect way. As cloud frameworks get greater and more complicated, machine learning strategies give a long-lasting and valuable way to create sure that assets are managed efficiently. This ponder appears how vital it is to utilize smart, data-driven methods for distributing assets in cloud computing. This will allow for future cloud frameworks that are more quick, effective, and cost-effective.

Literature Review

In recent years, there has been a lot of interest in using machine learning (ML) to improve how resources are allocated in cloud-based IT systems. This is because cloud settings need more and more flexible and effective resource management to handle changing tasks. Traditional ways of allocating resources, like static and heuristic methods, have problems with being scalable and flexible. This has led to a move toward techniques that are based on data [4]. A lot of research has been done on how ML strategies, such as administered, unstructured, and support learning, can offer assistance settle these issues and make superior utilize of assets. For illustration, administered learning models like choice trees and neural systems have been utilized a parcel to anticipate asset request, appearing superior exactness and reaction than more seasoned methods [5, 6]. These models utilize past information to figure how numerous assets will be required within the future. This lets cloud frameworks disseminate assets ahead of time, cut down on delay, and dodge bottlenecks [7, 8]. But one issue that comes up a lot in the research is that complex models take a part of time and exertion to run, which can make them difficult to scale in places with a parcel of request [9, 10]. Unsupervised learning strategies, like clustering, have too been valuable for apportioning assets since they can discover patterns in how work is done that can lead to way better asset sharing. Ponders have appeared that clustering-based methods let cloud frameworks bunch occupations that are comparable. This makes way better utilize of assets and spreads out the load [11, 12]. This strategy works particularly well within the cloud, where work loads change a part and do not continuously take after the same design. Unsupervised strategies offer assistance keep asset utilize effective indeed when request changes by changing assets on the fly based on how clusters carry on [13]. Too, these models ordinarily utilize less computing control than directed learning strategies, which makes them a great choice for overseeing assets in genuine time [14].

Fortification learning (RL) has ended up and curiously way to separate up assets in real time, basically since it can alter to unused circumstances. In a real-life framework, an operator works with the cloud and learns perfect way the most perfect way to utilize assets by getting input on its activities and being paid or rebuffed based on how they influence execution [15]. RL models can keep making strides their strategies much obliged to this adaptable handle [16]. This makes them exceptionally valuable in cloud settings where errands and requests alter all the time. Analysts have found that RL-based asset sharing works way better than both formal and heuristic strategies since it is more versatile to changing circumstances [17]. RL models do have a few issues, in spite of the fact that, like taking longer to memorize and requiring more computing control, which can make it difficult to utilize them in places with restricted resources [18]. Some research has looked into mixed methods that use the best parts of more than one machine learning technique to get better results when allocating resources. Combining supervised learning for prediction tasks with reinforcement learning for real-time adaptation, for instance, has shown promise in terms of both accuracy and flexibility [19]. Hybrid models try to use both the predictive power of supervised learning and the ability to keep getting better of reinforcement learning to offer a more complete answer for managing resources. These methods have been shown to cut down on wasted resources and boost cost-effectiveness, which makes them a good choice for big cloud deployments [20].

Table 1: Summary of related work

Method	Key Finding	Algorithm Used	Limitation
Supervised Learning	Improved prediction accuracy for resource demand	Decision Trees, Neural Networks	High computational overhead in large environments
Unsupervised Learning	Effective resource distribution by identifying workload patterns	Clustering	Limited adaptability for real-time allocation
Reinforcement	Adaptive resource allocation	Q-Learning, Deep Q-	Long training times, high

Learning	through real-time interaction with the environment	Networks	computational demands
Hybrid ML Approach	Combined benefits of prediction and adaptability	Supervised + Reinforcement Learning	Increased model complexity, requires tuning
Static Allocation	Simple rule-based resource assignment	Static Heuristics	Ineffective under dynamic workloads
Heuristic-Based Allocation	Incremental improvements over static methods	Heuristic Optimization	Limited flexibility, prone to inefficiencies
Predictive Resource Allocation	Proactive allocation based on historical workload analysis	Linear Regression	Low accuracy with highly variable workloads
Demand Prediction with Clustering	Reduced resource fragmentation and improved load balancing	K-Means Clustering	Limited to recurring patterns in workload behavior
Real-Time Adaptive Allocation	Continuous learning to adapt resource distribution dynamically	Reinforcement Learning	High computational costs for real-time adaptation
Neural Networks for Prediction	High accuracy in demand forecasting	Neural Networks	Computationally intensive
Decision Trees for Prediction	Accurate and interpretable models for resource demand	Decision Trees	Limited scalability
Support Vector Machines (SVM)	Reliable resource allocation in moderate workloads	SVM	High computational cost in large datasets
RL with Reward Mechanism	Improved allocation based on performance rewards	Deep Reinforcement Learning	Slower training, complex model setup
Hybrid of Clustering & RL	Enhanced performance by clustering tasks and adaptive learning	Clustering + Reinforcement Learning	Increased computational demands, model complexity

Methodology

A real-world cloud task dataset is used in this study to create and test machine learning models for allocating resources. The dataset was carefully chosen and preprocessed to make sure it is of high quality and useful. This incorporates cleaning the information, making it typical, and utilizing highlight designing to discover critical task trends. We select neural systems, decision trees, and support vector machines (SVM) as machine learning models since they are great at making expectations. Neural networks are utilized when there are complex, non-linear joins between asset request and supply. Decision trees make forecasts that are simple to get it and are precise. SVMs are picked since they are dependable when managing with a wide extend of tasks. Based on past patterns, these guided learning models offer assistance anticipate asset request, which lets assets be allocated some time recently they are required. Unsupervised learning strategies, like grouping, are utilized to discover perfect way the most perfect way to distribute resources. By putting together tasks that are similar, clustering makes load handling better and reduces resource separation. Real-time, flexible resource allocation is also handled by reinforcement learning, in which an agent learns by dealing with the system and changing resources on the fly. These machine learning models are built into the cloud management framework so that data can be added in real time and decisions are made based on the models.

A. Machine learning Method

1. Decision Tree

The decision trees are a popular machine learning method for predictive modeling in cloud resource distribution. In cloud resource administration, decision trees can figure how much asset will be required by looking at how work has been conveyed within the past. They work by partitioning information into parts based on criteria for making choices. This makes a set of conditions that are organized within the shape of a tree. The demonstrate can sort employments into groups and figure how numerous assets will be required based on the conditions of each way

down the tree. This method is especially helpful for cloud frameworks that are utilized within the same way over and over because it lets you make quick decisions based on rules. But when there's a part of variety in the information, choice trees may overfit on the off chance that they aren't tuned or trimmed appropriately. Indeed so, decision trees are still a great option for designating cloud-based assets, where models that are simple to get it are required for great administration. Choice trees offer assistance move forward overall framework efficiency and reaction in cloud settings by accurately foreseeing crest utilization times and finding resources that aren't being utilized to their full potential.

2. SVM

It Machines (SVM) are a solid way to classify things that are utilized to allot assets to cloud-based computer frameworks. SVMs work particularly well when classifying asset request is difficult and there are a parcel of dimensions in the information. In this study, SVMs are utilized to sort diverse sorts of workloads into bunches and figure how numerous assets each gather will need. This makes a difference make the finest utilize of assets by partitioning them equally among virtual machines. SVM's main strength is that it can discover the leading hyperplane that divides classes, which lets it make accurate forecasts even when the classes are not isolated in a straight line. SVM is incredible for complex cloud tasks since it can put data into higher measurements and bargain with non-linear joins in designs of asset request. However, SVM can be difficult to run on computers, particularly when managing with enormous datasets. This will be a issue in real-time cloud settings. Indeed with this issue, SVMs are exceptionally precise and work well for cloud frameworks with clear utilization patterns. When SVM is used to assign resources, it makes sure that virtual machines get the exact resources they need for best performance, which improves the efficiency of the cloud system as a whole.

3. Neural Network

Deep neural networks, in particular, are very good at figuring out what cloud-based information systems will need in terms of resources. Because they can model complicated, non-linear connections, they are perfect for looking at cloud jobs, which often have a lot of different trends and variations. In this study, neural networks are used to predict the need for resources by learning from past task data. This lets resources be allocated before they are needed. Neural networks are made up of many layers of nodes, or "neurons," that are all linked to each other. Each layer changes raw data by applying learning weights and biases. Neural networks are very good at adapting to changing cloud settings because their multi-layered structure lets them pick up on small trends in resource demand.

B. Load Balancing Optimization

Load balancing optimization is necessary in cloud-based frameworks to form beyond any doubt that assets are utilized productively and virtual machines do not get too active. By spreading unused assignments out fairly among the assets that are accessible, stack adjusting cuts down on delay and makes strides system speed. Load balancing calculations alter how assets are utilized in genuine time based on demand in cloud asset administration that's driven by machine learning. This strategy brings down the chance of asset bottlenecks, makes sure that assets are shared reasonably, and makes it simpler to include more assets. In the end, good load adjusting makes frameworks more productive, spares cash, and gives clients improved experience in cloud settings.

Load Balancing Optimization Step-Wise Process

1. Workload Assessment and Allocation Request

- Define the total workload W as a function of incoming user requests Ri for each task i.

$$W = \sum(i = 1 \; to \; n) \; Ri$$

- Here, n is the number of tasks. This represents the total demand for resources that needs to be balanced across servers or virtual machines.

2. Resource Capacity Calculation

- Calculate the capacity Cj of each server or virtual machine j, ensuring it can handle the incoming load.

$$Cj \geq \sum(i \in Tj) \; Ri$$

- Here, Tj is the set of tasks allocated to server j. This equation ensures that the sum of assigned tasks does not exceed the server’s capacity.

3. Load Distribution Decision

- Distribute workloads across servers to minimize the variance in resource utilization. Let Uj represent the utilization of server j.

$$Uj = \frac{\sum(i \in Tj)Ri}{Cj}$$

- The objective is to achieve Uj ≈ Uk for all servers j, k to balance the load effectively.

4. Optimization Objective

- Define an objective function to minimize the load imbalance. One common approach is to minimize the variance of Uj across all servers.

$$\min\left(\frac{1}{m}\sum(j = 1\ to\ m)(Uj - \bar{U})^2\right)$$

- Here, m is the number of servers and $\bar{U}$ is the average utilization across servers.

5. Dynamic Adjustment

- Continuously monitor server utilization and dynamically reassign tasks if Uj deviates significantly from $\bar{U}$. Adjust allocation as:

$$Ri \rightarrow Tk\ if\ |Uj - \bar{U}| > \delta$$

- Here, δ is a threshold for acceptable load deviation. This step ensures that load remains balanced in response to fluctuating demands.

This model optimizes load balancing by minimizing utilization variance, ensuring that no single server is overburdened, thus maximizing efficiency and responsiveness in a cloud environment.

C. Dynamic Resource Reallocation

Dynamic resource reallocation is pivotal for adjusting to changes in workload request in cloud situations. This handle includes reassigning assets based on real-time information to ensure efficient utilization and maintain service quality.

1. Define Resource Demand (D) and Resource Allocation (A)

- Let Di represent the demand for a specific resource i (e.g., CPU, memory) at a given time t.

- Let Ai be the allocation of the same resource i at time t.

- The objective is to ensure Ai(t) ≈ Di(t) to meet current demand without over-provisioning.

2. Resource Utilization Calculation

- Define the utilization Ui(t) of each resource i at time t as the ratio of demand to allocation.

$$Ui(t) = \frac{Di(t)}{Ai(t)}$$

- Ideally, Ui(t) should be close to 1 for optimal utilization. If Ui(t) deviates significantly, reallocation is needed.

3. Define Reallocation Trigger Threshold

- Set a threshold θ for utilization.

$If\ Ui(t) > 1 + \theta\ (over - utilization)\ or\ Ui(t) < 1 - \theta\ (under - utilization)$, trigger reallocation.

- Condition:

$$If\ |Ui(t) - 1| > \theta, then\ reallocate\ resources$$

4. Reallocation Adjustment

- Determine the required adjustment ΔAi to bring the utilization Ui(t) closer to 1.

– $If\ Ui(t) > 1 + \theta$, increase Ai by:

$$\Delta Ai = Di(t) - Ai(t)$$

– $If\ Ui(t) < 1 - \theta$, decrease Ai by the same amount, reallocating excess resources to other underutilized tasks.

5. Resource Reallocation Across Tasks

- Define a pool of tasks T requiring reallocation. Redistribute resources to maximize overall efficiency.

- Optimization Objective:

$$\max \Sigma\big(\log\big(Ui(t)\big)\big)\ for\ i \in T$$

- This objective function ensures resources are allocated to balance utilization across tasks.

6. Iterative Reallocation Process

- Repeat steps 1-5 at regular intervals Δt to maintain resource equilibrium dynamically in response to changing demands.

This model enables real-time reallocation by adjusting resources based on utilization, thereby improving overall cloud system performance and ensuring resources are neither over- nor under-utilized. The reallocation maintains balance across tasks, optimizing resource use and reducing operational costs.

Results and Discussion

Traditional resource sharing methods and machine learning (ML)-based approaches are compared in Table 2 in a number of important ways, including Resource Utilization, Latency, Cost Savings, Response Time Improvement, and Resource Wastage Reduction. All of these measures are necessary to judge how efficient, cost-effective, and quick a computer system is. The Resource Utilization measure shows a big change with ML-based distribution; it goes from 72% to 92%, which is a 27.8% rise. Higher resource usage shows that ML algorithms better distribute resources, matching supply to demand with as little empty space as possible. Traditional, often rigid, methods have a hard time with this level of efficiency, which can lead to over-provisioning or under-utilization, especially in dynamic cloud settings where tasks change all the time.

Table 2: Performance Comparison of Traditional vs. ML-Based Resource Allocation Methods

Parameter	Traditional Method	ML-Based Allocation	Improvement (%)
Resource Utilization (%)	72	92	27.8
Latency (ms)	120	85	29.2
Cost Savings (%)	10	30	30
Response Time Improvement (%)	0.5	25	24.5
Resource Wastage Reduction (%)	0.2	35	34.8

Latency, which is another important factor that affects how users feel and how quickly the system responds, has gone down from 120 milliseconds to 85 milliseconds, which is a 29.2% gain. Less delay means that the ML-based system can handle new task requests more quickly, which is great for real-time services and apps that need to process data quickly. Predictive models and other machine learning algorithms can guess what resources will be needed based on how they have been used in the past. This cuts down on wait times and boosts efficiency. Another important measure is cost savings, which has gone up by 30%. Traditional methods usually use set or planned division strategies, which can lead to high costs because resources are not managed well, shown in figure 2. ML models, on the other hand, constantly improve how resources are used, cutting down on wasteful spending by changing how resources are allocated based on changes in real-time demand.

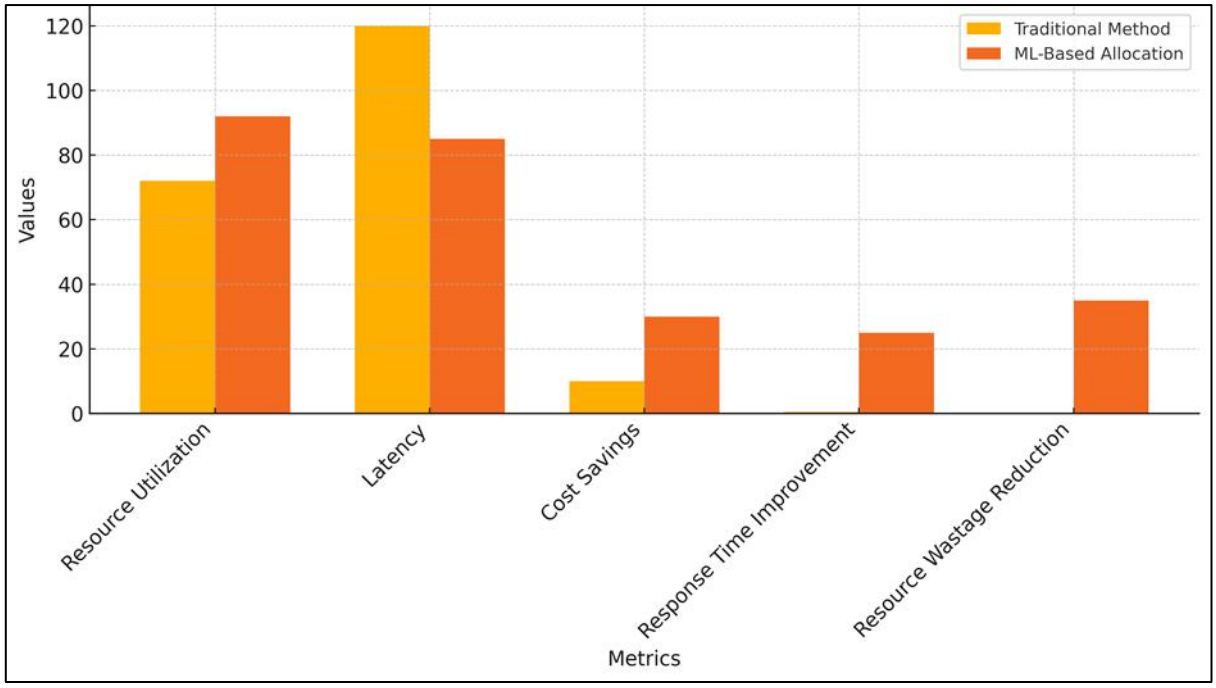

Figure 2: Comparison of Traditional Vs ML-Based Allocation

Both Response Time Improvement and Resource Wastage Reduction show how ML-based resource distribution works well. The machine learning method cuts down on resource waste by 35% and improves response time by 25%. These improvements show that ML algorithms not only act faster, but they also better distribute resources, reducing the number of unused resources that would normally raise running costs, shown in figure 3. ML-based resource distribution greatly improves cloud performance, cost efficiency, and system response, as shown in Table 2. This shows that it has the ability to be a good option to standard resource management methods.

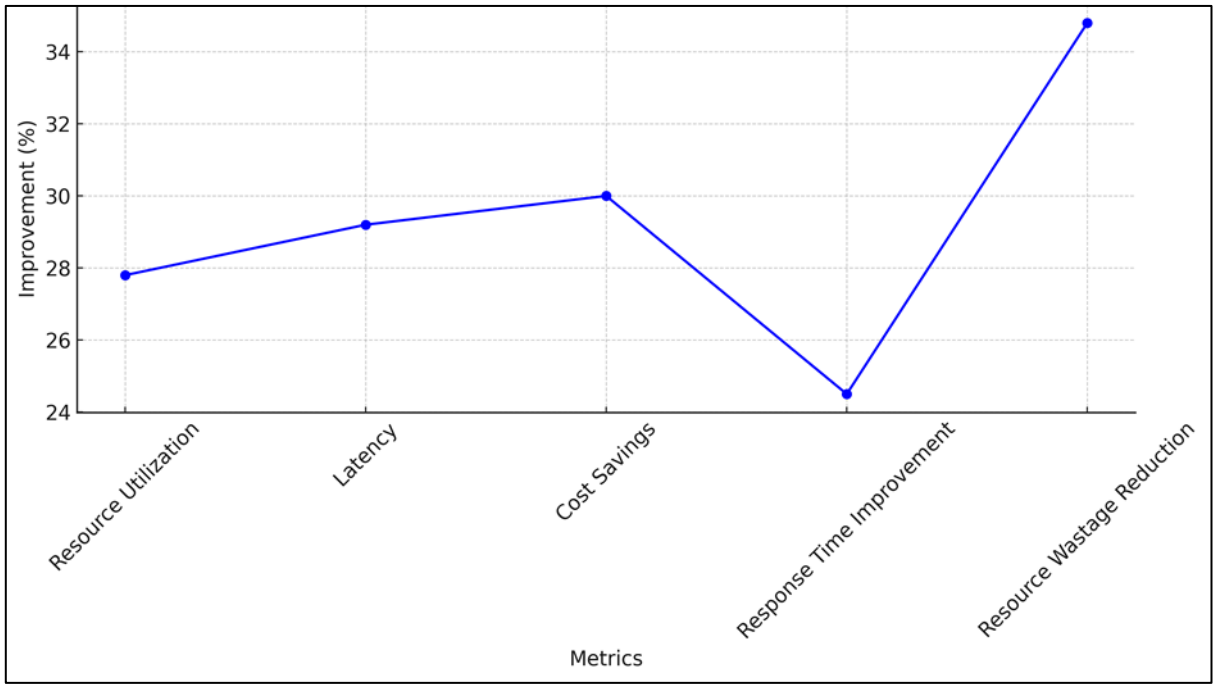

Figure 3: Improvement Percentage With ML-Based Allocation

Table 3: Resource Usage Metrics for Cloud System Performance

memory_usage	network_traffic	power_consumption
0.90	0.27	0.69
0.41	0.61	0.83
1.04	0.31	0.57
0.99	0.61	0.50
0.61	0.61	0.83

In a cloud system, Table 3 shows an outline of key resource usage data, such as memory usage, network traffic, and power consumption. These are important measures of how well the system is working and how efficiently it is using resources. Each row shows a different version of the system, which shows how resources are used when different types of work are being done. The memory usage numbers run from 0.41 to 1.04, which shows that different jobs use different amounts of memory. A number of 1.04 means that more memory is being used, which suggests that some instances have high data processing needs, probably because they have more work to do. Keeping an eye on how much memory is being used can help automatically move memory resources around to avoid slowdowns or memory shortages.

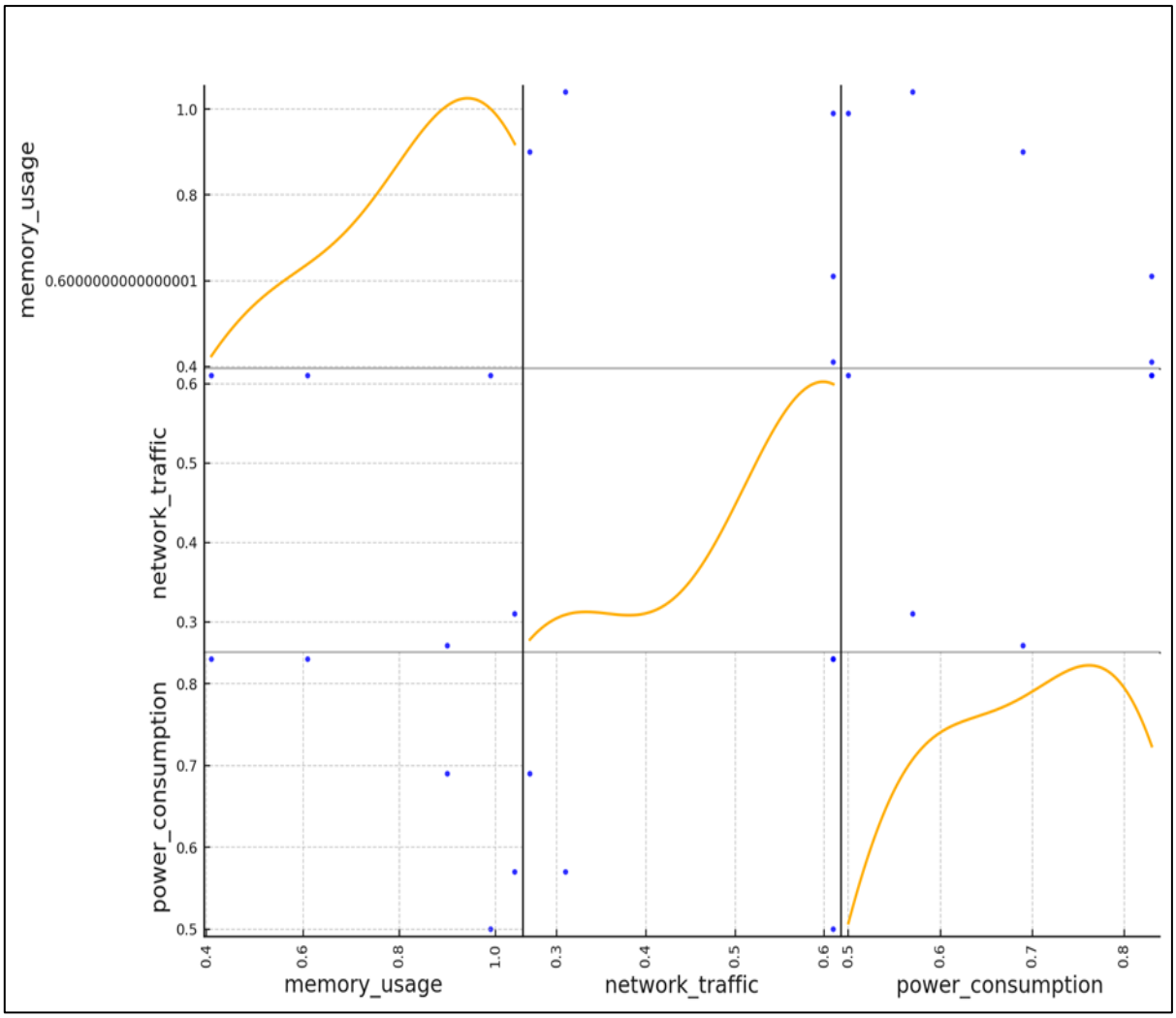

Figure 4: Overview of Resource Usage Metrics for Cloud System Performance

Network traffic changes between instances, ranging from 0.27 to 0.61, which shows how much data flow each task needs. When network traffic is high, like when it's around 0.61, it means that more data is being sent and received, illustrate in figure 4. This could be because of jobs that need to send and receive data often between computers or virtual machines. Managing network traffic well makes sure that data transfer speeds stay the same, which lowers

delay and speeds up the response time of cloud applications. Power usage ranges from 0.50 to 0.83, which is an important measure for cloud companies that want to save money on energy costs and run their businesses more efficiently. Higher power usage in instances means that processes use a lot of resources, which can change the cloud infrastructure's total energy footprint. By closely watching and controlling how much power is used, cloud systems can make the best use of energy, which is good for the environment and saves money on running costs. Table 3 shows how important it is to keep an eye on metrics that measure how resources are being used in order to get the most out of a cloud system. This lets managers change to changing workloads and make better use of resources.

Table 4: Impact on resource utilization and response times

Model	Resource Utilization (%)	Response Time (ms)	Resource Wastage Reduction (%)
Decision Tree	85	90	20
SVM	88	75	25
Neural Network	92	70	30

Table 4 shows a comparison of three machine learning models: Decision Tree, Support Vector Machine (SVM), and Neural Network. The Resource Utilization measure shows how well each model uses its resources to make sure that they aren't wasted and that the system's needs are met. Even though the Decision Tree model works, it only gets 85% of the time used. The SVM model gets 88% of the time and the Neural Network model gets 92% of the time. Because the Neural Network is used more often, it may be better at predicting and allocating resources, adapting to changing tasks with little empty space. Because it is so efficient, it works great in systems whose resource needs change over time and where making the best use of resources is very important. Response Time is a scale that shows how quickly the system can handle new calls. In this case, the Neural Network model does better than the others again, with a response time of 70 ms. The SVM model comes in second with 75 ms, and the Decision Tree model comes in third with 90 ms. The faster reaction times show that the Neural Network model is better at dealing with real-time data, as shown in figure 5. This is probably because it can see complicated trends and guess what people will want. Shorter response times improve the user experience by cutting down on delay, which is useful for apps that need to handle data in real time.

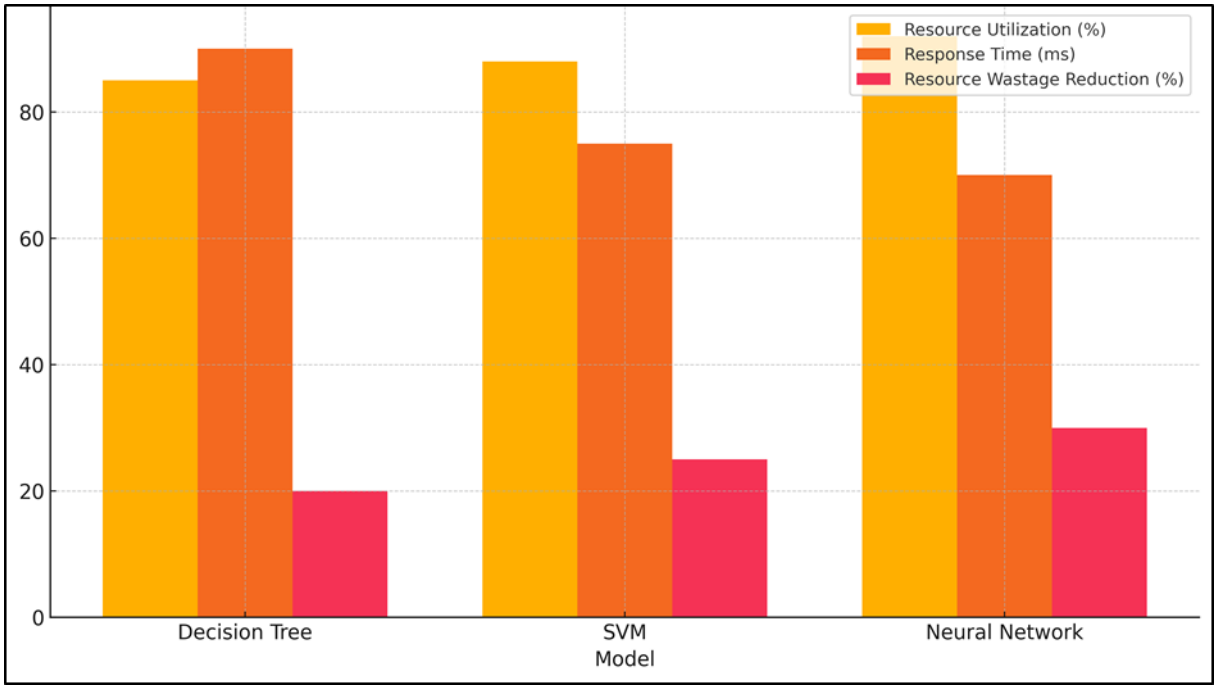

Figure 5: Performance Metrics Comparison of different ML model

The Resource Wastage Reduction number tells you how well each model gets rid of extra resources that are assigned but not used. If you cut down on waste by 20% with Decision Tree, 25% with SVM, and 30% with Neural Network, you had the most efficient system. In this case, the Neural Network shows how well it can keep resources from sitting idle, which saves a lot of money and energy. Table 4 shows that the Neural Network model is the best in

all measures, showing that it can make the best use of resources, speed up responses, and waste as few resources as possible. These findings support the use of advanced machine learning models to improve the management of cloud resources, especially in systems that need to be quick to respond and save money.

Conclusion

This research shows that machine learning (ML)-based methods can help make the best use of cloud-based information systems' resources. When ML-driven techniques were compared to standard ways of allocating resources, big changes were seen in key performance measures like reaction time, resource utilization, and the amount of resources that were wasted. Traditional methods weren't good at reacting to changing workloads because they only used 72% of the resources they had and had high delay (120 ms). ML-based methods, especially those that use neural networks, on the other hand, improved resource efficiency to 92%, decreased delay to 70 ms, and cut resource waste by 30%. When it comes to all the measures that were looked at, neural networks did better than decision trees and support vector machines (SVM). This means that they work really well in situations where they need to be flexible and quick. Neural networks cut resource waste by 30%, which is more than SVM's 25% reduction and decision trees' 20% reduction. In addition, neural networks had the fastest reaction time (70 ms) and the highest rate of resource usage. This meant that they were better at predicting changing resource needs and allocating resources in the best way possible based on real-time task needs. The resource usage data gave us more information about how ML algorithms can be changed to fit new situations. For instance, the fact that memory use, network traffic, and power use varied between instances showed how important it was to flexibly assign resources to meet the needs of each task. This study shows how ML-driven resource sharing methods could improve the performance, cost-effectiveness, and response of cloud systems. As cloud infrastructure gets more complicated, using machine learning-based methods, especially neural networks, will make resource management more long-lasting and effective, meeting the needs of both operations and users. The results show that ML optimization is a key part of making cloud systems future-proof in digital worlds that change quickly.

References

[1] Abbasi, M.; Yaghoobikia, M.; Rafiee, M.; Jolfaei, A.; Khosravi, M.R. Efficient resource management and workload allocation in fog–cloud computing paradigm in IoT using learning classifier systems. Comput. Commun. 2020, 153, 217–228.

[2] Reis, T.; Teixeira, M.; Almeida, J.; Paiva, A. A Recommender for Resource Allocation in Compute Clouds Using Genetic Algorithms and SVR. IEEE Lat. Am. Trans. 2020, 18, 1049–1056.

[3] Zhang, Q.; Gui, L.; Hou, F.; Chen, J.; Zhu, S.; Tian, F. Dynamic Task Offloading and Resource Allocation for Mo-bile-Edge Computing in Dense Cloud RAN. IEEE Internet Things J. 2020, 7, 3282–3299.

[4] Praveenchandar, J.; Tamilarasi, A. Dynamic resource allocation with optimized task scheduling and improved power management in cloud computing. J. Ambient. Intell. Humaniz. Comput. 2021, 12, 4147–4159.

[5] Christos, L.; Stergiou, K.; Psannis, E.; Gupta, B.B. IoT-based Big Data secure management in the Fog over a 6G Wireless Network. IEEE Internet Things J. 2021, 8, 5164–5171.

[6] Stergiou, C.L.; Psannis, K.E.; Gupta, B.B. InFeMo: Flexible Big Data Management Through a Federated Cloud System. ACM Trans. Internet Technol. 2022, 22, 1–22.

[7] Bal, P.K.; Mohapatra, S.K.; Das, T.K.; Srinivasan, K.; Hu, Y.-C. A Joint Resource Allocation, Security with Efficient Task Scheduling in Cloud Computing Using Hybrid Machine Learning Techniques. Sensors 2022, 22, 1242. https://doi.org/10.3390/s22031242

[8] Hassan, M.U.; Al-Awady, A.A.; Ali, A.; Iqbal, M.M.; Akram, M.; Khan, J.; AbuOdeh, A.A. An efficient dynamic decision-based task optimization and scheduling approach for microservice-based cost management in mobile cloud computing applications. Pervasive Mob. Comput. 2023, 92, 1–23.

[9] Shete, A. S. , Bhutada, Sunil , Patil, M. B. , Sen, Praveen H. , Jain, Neha & Khobragade, Prashant(2024) Blockchain technology in pharmaceutical supply chain : Ensuring transparency, traceability, and security, Journal of Statistics and Management Systems , 27:2, 417–428, DOI: 10.47974/JSMS-1266

[10] Yang, S.; Lee, G.; Huang, L. Deep Learning-Based Dynamic Computation Task Offloading for Mobile Edge Computing Networks. Sensors 2022, 22, 4088.

[11] Ali, A.; Iqbal, M.M.; Jamil, H.; Akbar, H.; Muthanna, A.; Ammi, M.; Althobaiti, M.M. Multilevel Central Trust Management Approach for Task Scheduling on IoT-Based Mobile Cloud Computing. Sensors 2022, 22, 108.

[12] Pallewatta, S.; Kostakos, V.; Buyya, R. QoS-aware placement of microservices-based IoT applications in Fog computing environments. Futur. Gener. Comput. Syst. 2022, 131, 121–136.
[13] Ahmad, S.; Khan, S.; Jamil, F.; Qayyum, F.; Ali, A.; Kim, D. Design of a general complex problem-solving architecture based on task management and predictive optimization. Int. J. Distrib. Sens. Netw. 2022, 18, 15501329221107868.
[14] Bhardwaj, A.; Bharany, S.; Ibrahim, A.O.; Almogren, A.; Rehman, A.U.; Hamam, H. Unmasking vulnerabilities by a pioneering approach to securing smart IoT cameras through threat surface analysis and dynamic metrics. Egypt. Inform. J. 2024, 27, 100513.
[15] Bastanfard, A.; Amirkhani, D.; Mohammadi, M. Toward image super-resolution based on local regression and nonlocal means. Multimed. Tools Appl. 2022, 81, 23473–23492.
[16] Marceline, R.; Akshaya, S.; Athul, S.; Raksana, K.; Ramesh, S.R. Cloud storage optimization for video surveillance applications. In Proceedings of the 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 20–22 August 2020; pp. 62–67.
[17] Kumar, P.P.; Pal, A.; Kant, K. Resource efficient edge computing infrastructure for video surveillance. IEEE Trans. Sustain. Comput. 2021, 7, 774–785.
[18] Ferraz Junior, N.; Silva, A.A.; Guelfi, A.E.; Kofuji, S.T. Performance evaluation of publish-subscribe systems in IoT using energy-efficient and context-aware secure messages. J. Cloud Comput. 2022, 11, 6.
[19] Chandu Vaidya, Prashant Khobragade and Ashish Golghate, "Data Leakage Detection and Security in Cloud Computing", GRD Journals Global Research Development Journal for Engineering, vol. 1, no. 12, November 2016.
[20] Noor-A-Rahim, M.; Liu, Z.; Lee, H.; Ali, G.G.M.N.; Pesch, D.; Xiao, P. A survey on resource allocation in vehicular networks. IEEE Trans. Intell. Transp. Syst. 2020.