Accurate Server Resource Allocation Using Network Performance Prediction

Lingala Mithun Kumar Reddy[1], Pusalapati Chandu[2], Pulipakula Akhil Kumar[3]

Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (SIMATS) Chennai, India.

E-mail: [1]mithunkumarreddyl5017.sse@saveetha.com, [2]chandup2119.sse@saveetha.com, [3]akhilkumarp3024.sse@saveetha.com

Abstract

Server resource allocation is an important problem for cloud computing and distributed systems. Modern data center services experience highly variable loads, which are combined with networks with complex dynamics. Without optimal resource allocation, improper resource allocation leads to poor resource efficiency, longer transaction times, decreased quality of service (QoS), and higher costs. This has proved to be insufficient for contemporary, timeliness-sensitive, and large-scale data centers. This research proposed a smart resource allocation technique called smart SRCA (Server Resource Capacity Allocation) with the aid of network performance prediction. This method effectively utilized historic and real time network data to provide appropriate resource allocation. In this study, we utilize server parameters and network performance metrics such as latency, average bandwidth utilization, packet loss, throughput, CPU load and current number of active users. These metrics help in training the proposed models for network performance prediction (NPP - Network Performance Prediction) and workloads prediction. Based on these models, appropriate server capacity can be dynamically allocated for CPU, memory, storage and network bandwidth, which are all factors affecting service quality provided to users. The results of the performance evaluation confirmed that incorporating the predictive analytics techniques into resource allocation not only improved the data center's capability in making smart decisions about the resources, also

Extended version of article received from National Conference on Intelligent Networks and Systems for Innovation using Generative AI in High-End Technologies
DOI: https://doi.org/10.36548/jitdw.2026.1.002
Received: 14.01.2026, received in revised form: 18.02.2026, accepted: 06.03.2026, published: 20.03.2026

assisted in increasing data center performance metrics such as average throughput, resource utilization, transaction speed and reliability. In future, the users can use the deep learning models for prediction with a real time SRCA implementation on edge computing architecture and implement using Software-Defined Networking (SDN) protocols.

Keywords: Network Performance Prediction, Server Resource Allocation, Machine Learning, Predictive Analytics, Network Optimization, Bandwidth Management, Quality of Service (QoS), Low Latency.

1. Introduction

The emergence and rapid adoption of cloud computing, edge computing, and large-scale distributed systems have significantly transformed the current information technology infrastructures. Data centers and cloud computing have become popular computing paradigms for providing services with high availability and low latency, including online services, multimedia services, and online analytics. However, the high demands and variability of users have put immense pressure on server infrastructures to efficiently allocate resources. Inefficient resource allocation causes servers to overload, underperform, or slow down, thus compromising the service-level agreement (SLA).

Network performance is a key aspect in effective management of server resources since parameters such as latency, bandwidth, packet loss, and throughput are key determinants of network performance. Traditional approaches to managing network resources are typically reactive in nature, responding to network congestion or performance degradation when they are encountered. However, traditional approaches are ineffective in a network environment that is constantly in a state of change. In order to effectively respond to such network environments, a predictive and intelligent approach to managing network resources is necessary. It is necessary to have a predictive approach to network management so that network performance can be effectively anticipated. With the development in machine learning technologies, data-driven approaches to network performance prediction are now possible. Machine learning is able to effectively predict network performance based on historical and real-time network data. The current research aims to propose a framework for Smart Server Resource Allocation Using Network Performance Prediction.

For instance, in complex and large-scale network infrastructures, intelligent resource management systems are required to support multiple workloads while reducing latency and operational costs. Network performance prediction is also used to enable servers to prepare for future workloads, thus providing a balanced workload and quality of service (QoS). In this paper, we are concerned with building a predictive-based resource allocation model using network performance indicators. We used simulated network infrastructures to test our proposed resource allocation approach and assess its effectiveness in real-world network scenarios. We used response time, resource utilization, and resource allocation accuracy as key performance metrics for our proposed resource allocation approach. Our results show that predictive-based resource allocation improves server performance significantly compared to traditional resource allocation methods. This research contributes to building smarter server resource management systems for efficient and scalable network applications in the next generation of network infrastructures.

2. Literature Review

With the high-speed development of cloud computing and distributed systems, here are large numbers of studies on the intelligent resource allocation methods. The conventional methods are mainly responsive, and the efficiency and speed cannot satisfy the demand of dynamic workload and the variation of network situation. Recently, a large number of studies using prediction method and the method of machine learning in the field of the server resource scheduling have been emerging. Penney et al. proposed the dynamic resource allocation based on the learning-based policy in [1], which enables smart resource scheduling according to the development situation of the future network, in order to achieve the adaptability to the varying network situation. Saxena and Singh presented an autoscaling model based on neural network using the forecasting methods in cloud data center in [2], in order to enhance the energy efficiency and real-time decision in cloud computing, so as to effectively improve the utilization of server resources, as can be seen in their work [3].

Resource management using Machine Learning is a popular paradigm for the next generation resource management systems. In the paper [4] surveying Machine Learning for resource management in Cloud Environment have elaborately described the challenges such as scalability, diversity and timeliness that the RMs for cloud have to encounter. Further, the research [5] have performed resource demand prediction for cloud environment based on QoS characteristics, while in the article Resource Management in Massive IoT-Enabled 5G Networks: Challenges and Future Directions, the study [6] have discussed QoS provisioning in massive IoT-enabled 5G networks which need careful resource management to cater to the enormous number of devices and stringent latency requirements of the wireless links. Another technology that has received interest as a potential solution for reducing latency and enhancing efficiency is edge computing. The work [7] presented dynamic resource allocation algorithm design, which was targeted at edge computing. This indicates the importance of edge computing. In addition, time series prediction was used for resource prediction. The research [8] used LSTM to predict CPU utilization in a cloud computing system, and the paper [9] presented the CloudInsight framework for workload prediction, which was targeted at enhancing efficiency in resource prediction.

Machine learning has been gaining popularity in recent researches for cloud and mobile network environments. Hassan et al. proposed a smart resource allocation framework for mobile cloud networks using machine learning approaches to improve the orchestration and performance of the systems. The work [11] proposed LSTM based prediction models for cloud resource utilizations using deep learning to understand the pattern in workloads. The research [12] carried out a comprehensive survey that focused on machine learning approaches for the cloud resource allocation. It gives a detailed comparison of different machine learning algorithms for cloud resource allocation such as supervised, unsupervised, ensemble, and instance-based learning.

However, despite the large body of research in predictive resource allocation, there is still a need to carry out comparative studies that compare different machine learning algorithm approaches under the same conditions. Most of the existing research focuses on deep learning models and other algorithm-based approaches without proper comparisons. The need to bridge this gap is what this research seeks to achieve through the comparative analysis of the Random Forest and K-Nearest Neighbours (KNN) algorithm approaches to predicting optimal server resource allocation through network performance metrics. The proposed approach combines predictive analytics and network parameters to improve resource utilization and system performance.

3. Methodology

The proposed system is designed as a data-driven predictive system that incorporates network monitoring, machine learning models, and resource allocation strategies. The proposed methodology involves multiple steps, including data collection, preprocessing, modelling, prediction, and resource allocation.

3.1 Data Collection and Feature Selection

The first step requires the collection of both historical and real-time data on the performance of the network. The data set includes parameters such as latency, bandwidth, packet loss, throughput, CPU, and the number of active users. These parameters are chosen based on the fact that they have a direct influence on the performance of the server. The data collected from the server environment is then used as input for the machine learning models. Feature selection takes place, where the most relevant parameters are chosen, thereby ensuring the accuracy of the prediction. Any redundant data is removed from the set, thus improving the efficiency of the models.

3.2 Data Preprocessing

Before training the models, data preprocessing is carried out to maintain data quality. Data preprocessing includes the following steps:

- Data Cleaning: Removing missing values in the data
- Normalization: Scaling the data to a certain range to improve model performance
- Encoding: Converting data into numerical form, especially for categorical data
- Dataset Splitting: Splitting data into training and testing sets, comprising 70% and 30%, respectively Normalization is critical for models such as KNN, which use distance calculations.

3.3 Model Development

Two machine learning algorithms are implemented and compared in this study:

- Random Forest is an ensemble learning method that builds multiple decision trees using bootstrap sampling. Each tree is trained on a random subset of data and features, which improves generalization and reduces overfitting. The final prediction is obtained through majority voting among all trees. This model is effective in handling complex and non-linear relationships between network parameters and resource requirements.

- KNN is an instance-based learning algorithm that classifies new data points based on similarity with existing data. It calculates the distance (typically Euclidean) between the new input and training samples, and assigns the class based on the majority of the nearest neighbors.

3.4 Training and Prediction

The models are trained based on the processed dataset. The patterns between the features and the output, i.e., between the network metrics and the optimal resource allocation decisions, are learned during the training process. The trained models are tested with new, unseen data to assess the prediction capability of the models. The result obtained from the models will show if the resources should be scaled up, down, or left as they are.

3.5 Resource Allocation Mechanism

On the basis of predictions made by the machine learning models, the system adjusts server resources. The resource allocation strategy includes: CPU Scaling: increasing or decreasing processing capabilities Memory Allocation: allocating memory according to the need of the workload Bandwidth Allocation: allocating network capabilities to avoid congestion Load Balancing: balancing the workload across servers.

4. Simulation Results and Discussions

The proposed system was tested using various simulations, which were carried out using a server allocation dataset. The dataset contains various network performance parameters, such as latency, bandwidth, packet loss, CPU, and number of active users, which were used to train and test machine learning models, such as Random Forest and K-Nearest Neighbours, to determine how effectively they can be used for optimal resource allocation decisions.

Table 1. Performance Comparison of Random Forest and KNN Models

Metric	Random Forest	KNN
Accuracy	0.9067	0.783
F1 Score	0.8841	0.611
Recall	0.9382	0.600
Precision	0.8570	0.622

Initially, both models are trained using the preprocessed dataset and evaluated based on their performance metrics, which include accuracy, precision, recall, and F1-score. As presented in Table 1, the accuracy of the Random Forest model is 90.38%, which is significantly higher than the 85.2% accuracy recorded in the KNN model. Therefore, this implies that the Random Forest model is more effective in learning the complex relationships between the network conditions and the resource requirements. The model is also effective in the optimal allocation decision since the precision is 0.857 and the recall is 0.938. On the other hand, the KNN model is characterized by low performance in comparison to the Random Forest model in terms of recall (0.600) and F1-score (0.611).

High performance of the Random Forest model can be attributed to its learning strategy, which is based on the use of multiple decision trees to make predictions. On the other hand, KNN is based on distance similarity and is thus affected by feature scaling and is computationally expensive. The limitations of KNN make it difficult to perform well in real-time and server environments where high-speed predictions are required.

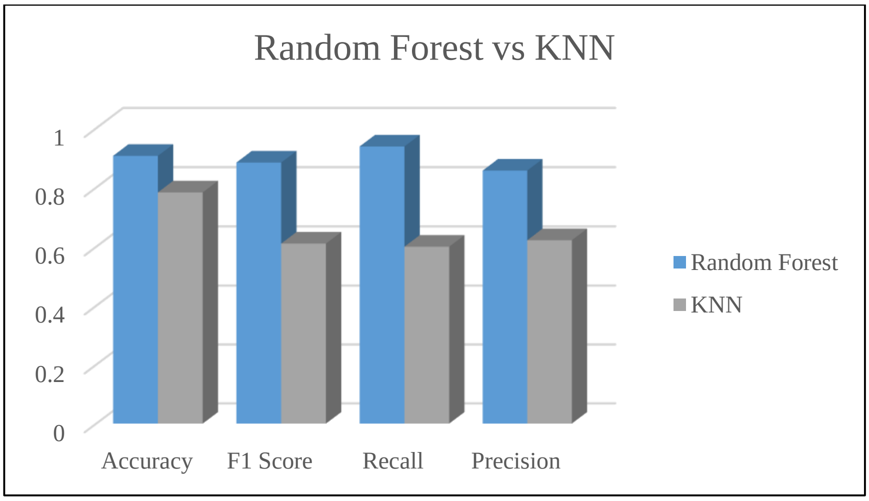

Figure 1. Performance Comparison

Table 2. Statistical Analysis of Model Accuracy

Algorithm	Mean Accuracy (%)	Standard Deviation	Standard Error
Random Forest	90.38	3.51	1.11
KNN	85.21	2.83	0.897

Table 3. Effect Size Analysis of Network Parameters Influencing Server Resource Allocation

Bandwidth	Packet_Loss	Active_User	CPU_Usage
106.1810	3.2101	410	89.6153
192.6071	0.4207	335	98.5714
159.7991	0.8081	322	82.6317
139.7987	4.4927	418	100.3544
73.4027	3.0321	111	47.7416

The effect size analysis, as shown in Table 3, further provides additional insights into how different parameters affect resource allocation decisions within the network. For instance, parameters such as bandwidth, packet loss, active users, and CPU usage have significant variations, thus emphasizing their importance in resource allocation decisions. This is because the Random Forest approach effectively captures these variations due to its capability to handle non-linear relationships, while KNN fails to generalize these variations.

Table 4. Regression-Based Performance Evaluation

Performance	MAE	RMSE	R2
Random Forest	5.21	6.51	0.93
KNN	5.91	7.25	0.91

In addition to classification performance, regression-based evaluation metrics such as Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R² score were also analysed, as shown in Table 4. The Random Forest model achieved a lower MAE (5.21) and RMSE (6.51) compared to KNN (MAE = 5.91, RMSE = 7.25), indicating more accurate predictions with smaller error margins. Moreover, the R² score for Random Forest (0.93) is higher than that of KNN (0.91), suggesting that Random Forest explains a greater proportion of variance in the dataset.

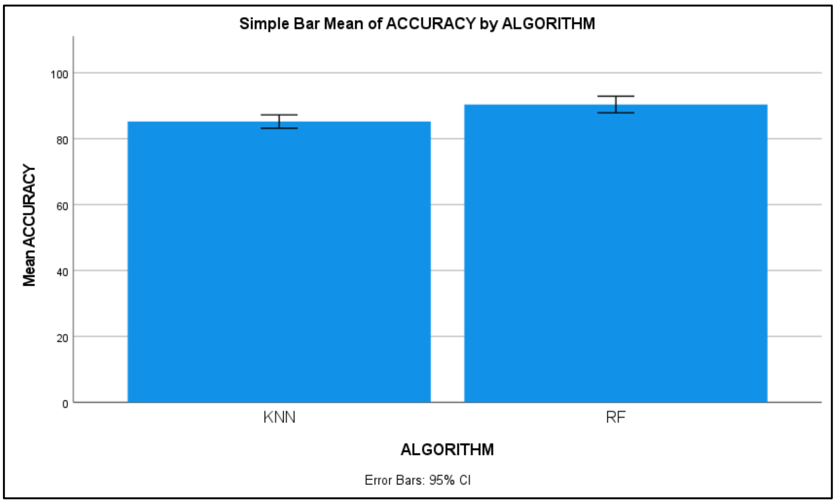

Figure 2. Mean Accuracy Comparison

The graphical charts (Figure 1 and Figure 2) further elaborate the performance comparison between the two models. From the 3D bar graph, it is evident that the Random Forest model excels over the KNN model for all the evaluation parameters. Additionally, the mean accuracy bar chart also indicates a considerable difference between the two models, which further supports the fact that the Random Forest model is more suitable for the prediction of server resources. From the simulation results, it is evident that the incorporation of machine learning prediction improves the allocation of server resources significantly. The Random Forest model, in particular, reflects higher accuracy, reliability, and consistency. Therefore, the Random Forest model can be considered a reliable option for the prediction of server resources.

5. Conclusion

This study proposes server resource allocation based on network performance prediction. The dynamic workload of applications and frequent changes in network conditions lead to resource waste. Using machine learning to predict network conditions before occurring allows a server to make an efficient decision in advance and therefore reduce the reaction time. Some indicators for network performance are latency, bandwidth used, throughput and packet loss. Using these indicators, we could easily implement a resource management system that learns and adjusts itself accordingly. By comparing the accuracy of predicting an optimum resource configuration by using Random Forest and K-Nearest Neighbours (KNN) algorithm, our experiments indicate that machine learning algorithms do work. While Random Forest deals with complex and high-dimensional data for network performances and achieves higher accuracy in its predictions, KNN is a simple algorithm and by taking advantages of it we can easily identify network performances with similar conditions. While KNN has limitations concerning high dimensionality data and the need for scaling features, our results indicate that machine learning can be utilized efficiently to predict server resource configurations and therefore help us in achieving higher resource utilization, lower latency and quality of service. Consequently, an intelligent resource management system that is scalable and cost-effective can be implemented in today’s data center and cloud computing environments based on our proposed idea of using network performance prediction in the server resource allocation algorithm. This research provides strong foundations for future studies using more advanced machine learning techniques based on Deep Learning (DL) and also testing the proposed idea in real-time environments based on Software Defined Networking (SDN) and Edge Computing (EC).

References

[1] Penney, Drew, Bin Li, Jaroslaw J. Sydir, Lizhong Chen, Charlie Tai, Stefan Lee, Eoin Walsh, and Thomas Long. "Prompt: Learning Dynamic Resource Allocation Policies for Network Applications." Future Generation Computer Systems 145 (2023): 164-175.

[2] Saxena, Deepika, and Ashutosh Kumar Singh. "A Proactive Autoscaling and Energy-Efficient VM Allocation Framework Using Online Multi-Resource Neural Network for Cloud Data Center." Neurocomputing 426 (2021): 248-264.

[3] Saxena, Deepika, and Ashutosh Kumar Singh. "Workload Forecasting and Resource Management Models Based on Machine Learning for Cloud Computing Environments." arXiv preprint arXiv:2106.15112 (2021).

[4] Khan, Tahseen, Wenhong Tian, Guangyao Zhou, Shashikant Ilager, Mingming Gong, and Rajkumar Buyya. "Machine Learning (ML)-Centric Resource Management in Cloud Computing: A Review and Future Directions." Journal of Network and Computer Applications 204 (2022): 103405.

[5] Nawrocki, Piotr, and Patryk Osypanka. "Cloud Resource Demand Prediction Using Machine Learning in the Context of Qos Parameters." Journal of Grid Computing 19, no. 2 (2021): 20.

[6] Murthy, AVS Santhosh RK, and K. Sathish. "Enhancing the Connectivity of IoT Devices by Linking with 5G Using Random Forest and KNN." In 2025 Tenth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), IEEE, 2025, 1-5.

[7] Feng, Jie, Qingqi Pei, F. Richard Yu, Xiaoli Chu, Jianbo Du, and Li Zhu. "Dynamic Network Slicing and Resource Allocation in Mobile Edge Computing Systems." IEEE Transactions on Vehicular Technology 69, no. 7 (2020): 7863-7878.

[8] Nääs Starberg, Filip, and Axel Rooth. "Predicting a Business Application's Cloud Server CPU Utilization Using the Machine Learning Model LSTM." (2021).

[9] Kim, In Kee, Wei Wang, Yanjun Qi, and Marty Humphrey. "Forecasting Cloud Application Workloads with Cloudinsight for Predictive Resource Management." IEEE Transactions on Cloud Computing 10, no. 3 (2020): 1848-1863.

[10] Hassan, Mahmood Ul, Amin A. Al-Awady, Abid Ali, Muhammad Munwar Iqbal, Muhammad Akram, and Harun Jamil. "Smart Resource Allocation in Mobile Cloud Next-Generation Network (NGN) Orchestration with Context-Aware Data and Machine Learning for the Cost Optimization of Microservice Applications." Sensors 24, no. 3 (2024): 865.

[11] Ouhame, Soukaina, Youssef Hadi, and Arif Ullah. "An Efficient Forecasting Approach for Resource Utilization in Cloud Data Center Using CNN-LSTM Model." Neural Computing and Applications 33, no. 16 (2021): 10043-10055.

[12] Bodra, Deep, and Sushil Khairnar. "Machine Learning-Based Cloud Resource Allocation Algorithms: A Comprehensive Comparative Review." Frontiers in Computer Science 7 (2025): 1678976.