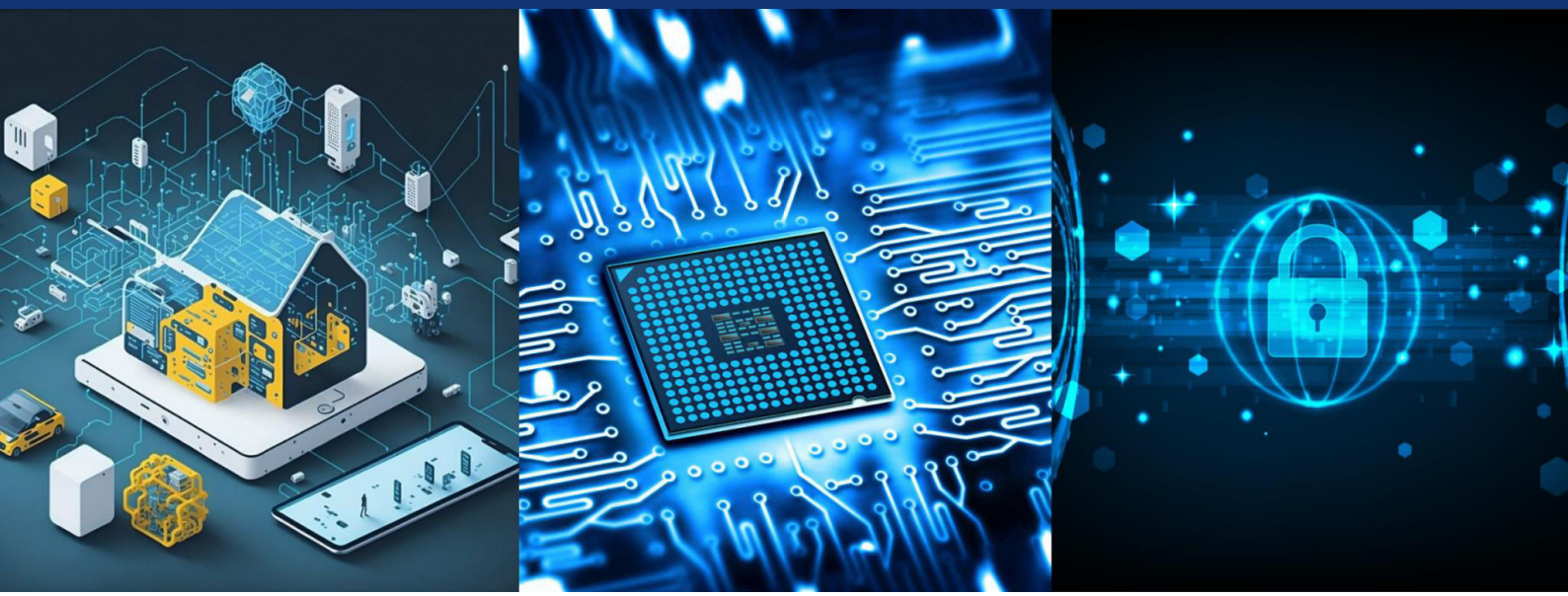
IJIRCCE

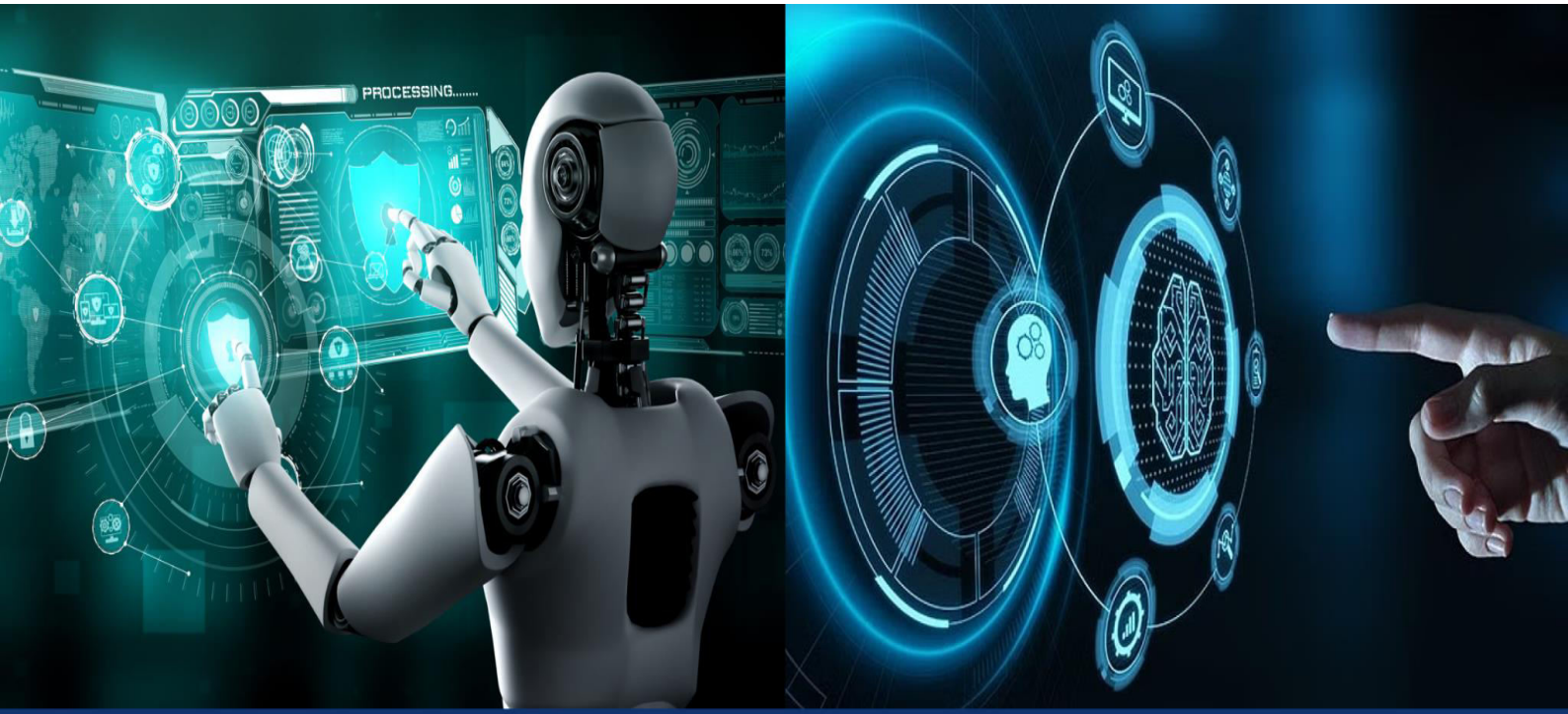
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 14, Issue 4, April 2026

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404060

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Intelligent Resource Allocation Optimization Algorithms for Cloud Computing using Deep Learning Models

G Rama Krishna, Sivaneasan Bala Krishnan, Reddi Kiran Kumar, Prasun Chakrabarti

Department of Computer Science and Engineering, Post-Doctoral Research Scholar, Singapore Institute of Technology, Singapore, Aditya Institute of Technology and Mangement, Tekkali, India

Singapore Institute of Technology, Singapore, Aditya Institute of Technology and Mangement, Tekkali, India

Department of Computer Science, Krishna University, Machilipatnam, India

Department of Computer Science and Engineering, Sir Padampat Singhania University, Udaipur, Rajasthan, India

ABSTRACT: The demand for high-performance cloud services and effective resource utilization has greatly increased due to the quick expansion of cloud-based applications. Maintaining performance and Quality of Service (QoS) under dynamic and heterogeneous workloads is a significant challenge for traditional resource allocation mechanisms. Adaptive and intelligent resource scheduling techniques are crucial to overcoming these constraints. Recently, deep learning (DL), a well-known branch of artificial intelligence (AI), has become a useful tool for creating intelligent cloud resource management systems. A thorough analysis of DL-based scheduling and resource allocation strategies for cloud computing is presented in this paper. It starts with a summary of the principles of cloud scheduling and how they contribute to dependable service delivery. After that, the study examines current developments in DL-based scheduling, emphasizing model architectures, design approaches, optimization strategies, and practical uses. A comparative study of well-known deep learning algorithms is presented, assessing their robustness, accuracy, scalability, and responsiveness. Lastly, new research avenues are explored to improve future cloud resource management systems, such as the integration of Reinforcement Learning and Transfer Learning.

KEYWORDS: Cloud Computing, Deep Learning Models, Quality of Service, Resource Scheduling, Cloud Optimization

I. INTRODUCTION

Resource scheduling is a crucial element that directly affects system performance, resource usage, and user satisfaction in the context of cloud computing [1]. Its main goal is Among them are fault recovery, load balancing, and priority scheduling. Strategies like hybrid approaches, static allocation, and dynamic allocation are used to accomplish these objectives. By keeping computational, storage, and network resources in balance, these tactics enhance overall service quality [2, 3]. One way to formalize the resource scheduling problem is as an optimization problem. In order to ensure effective resource use and improved system responsiveness, it usually seeks to reduce the overall task completion time [4]. This objective function aids in ensuring the effective use of computational resources. The effective use of computational resources is ensured by this objective function [5]. Deep learning provides creative methods for resource planning. Deep neural networks (DNNs) enable autonomous decision-making and dynamic resource demand prediction, allowing systems to adjust to changing workloads and maximize resource utilization and response times [6]. Convolutional neural networks (CNNs), for instance, have been used to improve predictive reliability in load forecasting tasks, with accuracy rates surpassing 90% [7]. Similarly, through self-learning and adaptation, reinforcement learning (RL) algorithms improve scheduling strategies, allowing for more effective resource management in a variety of scenarios [8, 9].

Resource scheduling algorithms must demonstrate adapt- ability and scalability. Long Short-Term Memory (LSTM) networks increase efficiency by learning from past data. attaining resource utilization rates higher than 85% in complex environments [10]. Furthermore, resource allocation in milliseconds is made possible by hybrid models that integrate

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

deep learning and time-series prediction. This capability finds extensive applications in fields like virtual machine management and container orchestration and successfully handles high-density request demands [11, 12].
Further technological advancements are made possible by deep learning-driven resource scheduling, which improves the intelligence and efficiency of cloud resource management [13]. Figure 1. Demonstrates the resource scheduling system's architecture in a cloud computing setting, emphasizing the system's crucial function and how it interacts with other elements.

II. LITERATURE REVIEW

A. Model of Static Resource Scheduling:

Static resource scheduling models are used in cloud computing environments to effectively distribute computing resources, meeting user demands and optimizing system performance [14]. In order to achieve optimal allocation, these models use heuristic algorithms or strategies based on linear programming, depending on task characteristics and static resource configurations [15]. Task characteristics and pre-established resource configurations determine how effective static scheduling is. These models typically allocate resources using heuristic techniques or fixed rules [16]. Static scheduling techniques can efficiently distribute resources while reducing computational complexity under well-defined workload conditions. Additionally, resource state monitoring and system architecture are crucial. Specifically, ratio-nal resource configuration before task submission is made possible by multi-layer integrated architectures, which enhances system throughput and response times [17]. Enhancing the generalizability of static scheduling strategies requires a diversity of datasets. Building training datasets with multiple scenarios greatly enhances model flexibility. In terms of scheduling efficiency and system response time, experimental results show that models trained with a variety of inputs consistently outperform single-scenario models [18]. In cloud computing, static resource scheduling models still have a lot of room for improvement. These models can provide more accurate and efficient resource management solutions by combining heuristic algorithms and multi-layer decision architectures, especially under predetermined workload and resource conditions [19]. The Study: Static scheduling algorithms were applied to cloud platforms such as OpenStack in a test environment, with an emphasis on pre-allocated resources. The findings demonstrated that static Although scheduling models performed less well in dynamic situations, they could reduce overhead by 20% in low-loadscenarios.

B. Model of Dynamic Resource Scheduling:

In cloud computing environments, dynamic resource scheduling models are essential, especially for deep learning tasks where resource demands are constantly changing. These models

need to be adaptable and responsive in real time in order to handle scheduling [5, 20]. Dynamic models optimize performance by making real-time adjustments across different resource types, such as GPUs, CPUs, memory, and storage, based on workload characteristics, resource availability, and task priorities [6].

Prediction-based and feedback-based techniques are common dynamic scheduling algorithms [7]. Time series algorithms are used in prediction-based methods, including

LSTM and ARIMA, to evaluate past data and make proactive resource allocations [10]. On the other hand, feedback-based approaches make use of online learning tools like reinforcement learning and real-time monitoring to improve scheduling choices on their own [8, 9]. Additionally, adaptive load balancing improves resource efficiency through dynamic threshold strategies, while priority-based scheduling mechanisms guarantee prompt support for high-priority tasks [16].

Case Study: To manage varying data center load, an LSTM-based dynamic scheduling model was put into place on AWS. When demand was high, the system showed a 30% decrease in the amount of time needed to complete tasks and a 25% increase in the use of resources overall. This example highlights how well the dynamic model works in environments with a lot of resources. Table 1 presents a comparison of various scheduling techniques, such as hybrid strategies, static allocation, and dynamic allocation.

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404060

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Table 1. Comparison of Scheduling Methods:

Scheduling Method	Advantages	Disadvantages	Suitable Scenarios
Dynamic Allocation	Flexible to changing demands	High complexity	Highly fluctuating loads
Static Allocation	Simple to implement	Low resource utilization	Relatively stable loads
Hybrid Strategy	Combines advantages of both	Complex to implement	Diverse load demands

The fundamental process for updating the Q-value within the core Q-learning algorithm is defined by the following formula:

$$Q(s,a) \leftarrow Q(s,a) + \propto (r + \gamma max Q(s',a') - Q(s,a))$$

III METHODOLOGY

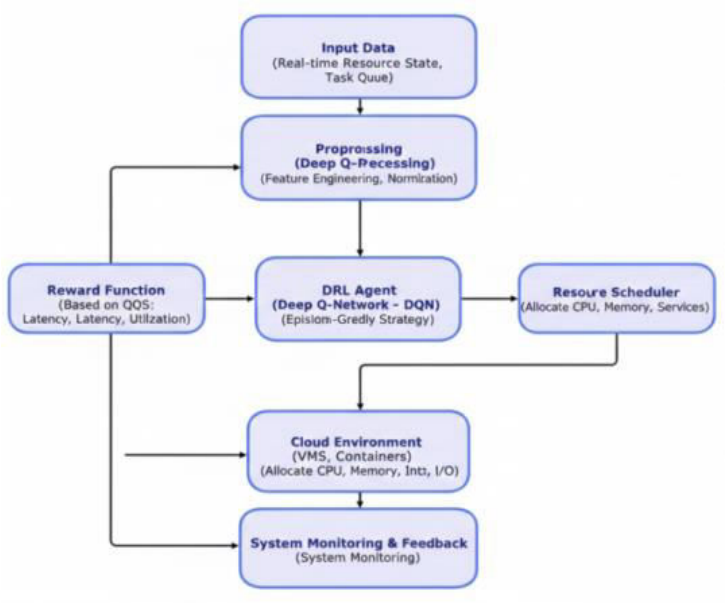


Fig. 1. Deep Learning Model Application Flowchart in Resource Scheduling

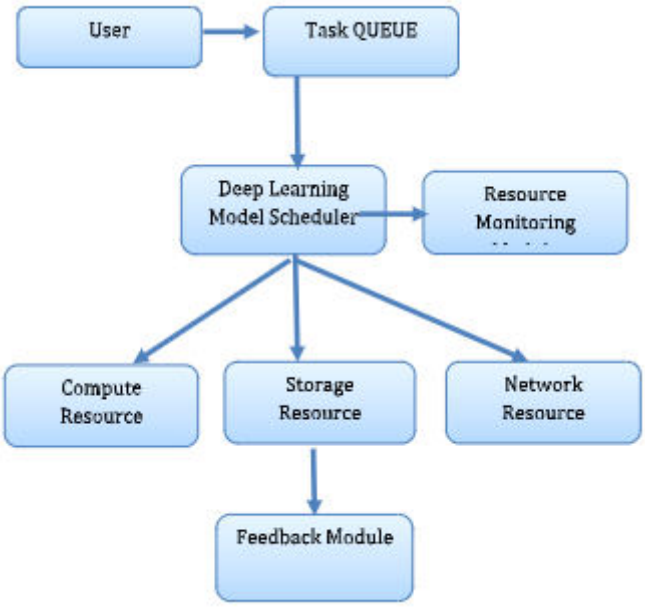
The Adam optimizer is typically used to compile the suggested models with a standard learning rate of 0.001. An 8:2 split ratio is used to separate the training and testing sets from the available datasets. To effectively prevent overfitting, early stopping is applied based on validation performance after 50 to 100 epochs of training. Its performance analysis uses strict criteria of evaluation, which include accuracy, recall, and F1 value. Comparison tests between current approaches, such as round robin and shortest job first scheduling, are conducted to verify its superiority. These tests have demonstrated that deep learning models consistently outperform other approaches when it comes to resource scheduling. Optimization Methods and Improving Performance Network latency, dynamic workloads, and the complexity of multiple user requirements are some of the issues that cloud computing resource scheduling must deal with. By utilizing both historical and present system data, deep learning optimization algorithms for resource scheduling improve efficiency.

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404060

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



A. Optimization Techniques and Performance Enhancement

Network latency, dynamic workloads, and the complexity of multiple user requirements are some of the issues that cloud computing resource scheduling must deal with. By utilizing both historical and current system data, deep learning optimization algorithms for resource scheduling improve efficiency. Additionally, methods like grid search and Bayesian optimization can speed up model convergence as shown in fig.2. When it comes to handling large amounts of data, distributed computing environments and the use of GPUs can significantly increase training phase efficiency by at least a factor of three when compared to traditional methods.

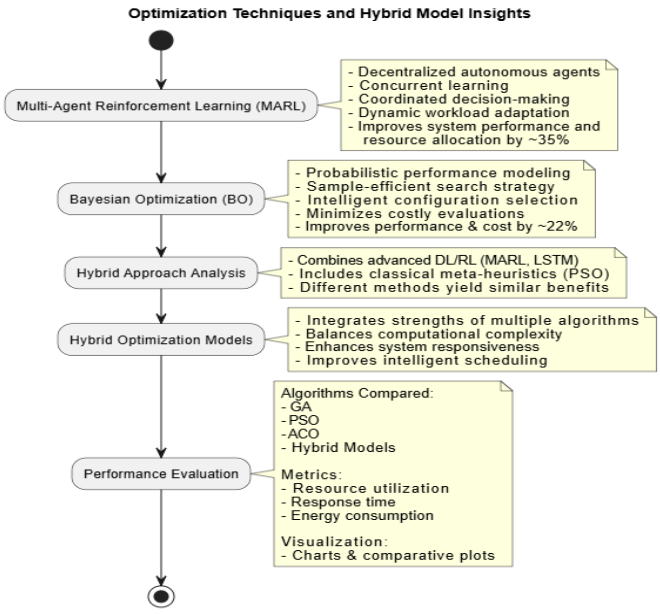


Fig. 2: Optimization Techniques and Hybrid Model Insights

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

C. Investigations into Multi-objective Optimization Techniques

Multi-objective optimization strategies must simultaneously optimize for performance, energy consumption, resource usage, and service quality. Among these techniques are Ant Colony Optimization, Particle Swarm Optimization, and Genetic Algorithms. Genetic Algorithms (GA): Delays in scheduling, typically increasing the effectiveness of scheduling optimization by 15% to 20%. Particle Swarm Optimization (PSO): Based on group behaviour, PSO requires dynamic resource distribution, which leads to a significant 10%–30% increase in response time. Ant Colony Optimization (ACO): A technique for optimizing resource utilization that can reduce energy consumption by 15%. By incorporating deep learning techniques used in multi-objective optimization, scheduling approaches become even more intelligent.

IV. RESULTS AND EVALUATION METRICS

Several evaluation metrics are used to measure the efficacy of deep learning-based resource scheduling algorithms. Predictive accuracy, resource utilization, task completion efficiency, energy consumption, and overall system responsiveness are all evaluated by these metrics.

Mean Absolute Error (MAE): Used to evaluate the accuracy of CNN and LSTM models in predicting resource demand.

$$MAE = \frac{1}{N}\sum_{i=1}^{N} | y_i - \hat{y}_i |$$

Where y_i is the actual value and $\hat{y}_i$ is the predicted value. Studies report CNN models achieving MAE < 5%.
Prediction Accuracy: CNN-based load prediction models surpass 90% accuracy, whereas LSTM models reach up to 92% accuracy in workload forecasting.

Rate of Resource Utilization: This metric assesses the efficiency with which computational resources are employed Resource Utilization Improvement: Deep Q-Networks (DQN) increase resource utilization by more than 15%.
Utilization is increased by 18% using Generative Adversarial Networks (GANs).

Make span, or task completion time:
Calculates the total amount of time needed to finish a series of tasks:
Reduction in Completion Time:
Under high-load circumstances, LSTM-based dynamic scheduling cuts job completion timeby30%. Task completion times are typically improved by 20–30% using dynamic scheduling models.

Scheduling Delay
Evaluates the latency introduced by the scheduling decision process:
GAN-based scheduling reduces scheduling delays by 28% .

Energy Consumption
Important for green computing and cost efficiency:

Energy Reduction:
Ant Colony Optimization (ACO) reduces energy consumption by 15%
multi-objective optimization strategies balance energy and performance trade-offs.

Model Training and Validation Metrics
Used during the development and evaluation of deep learning models:
Loss Function (Mean Squared Error - MSE):

$$MSE = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

Minimized during training to improve prediction accuracy.

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404060

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

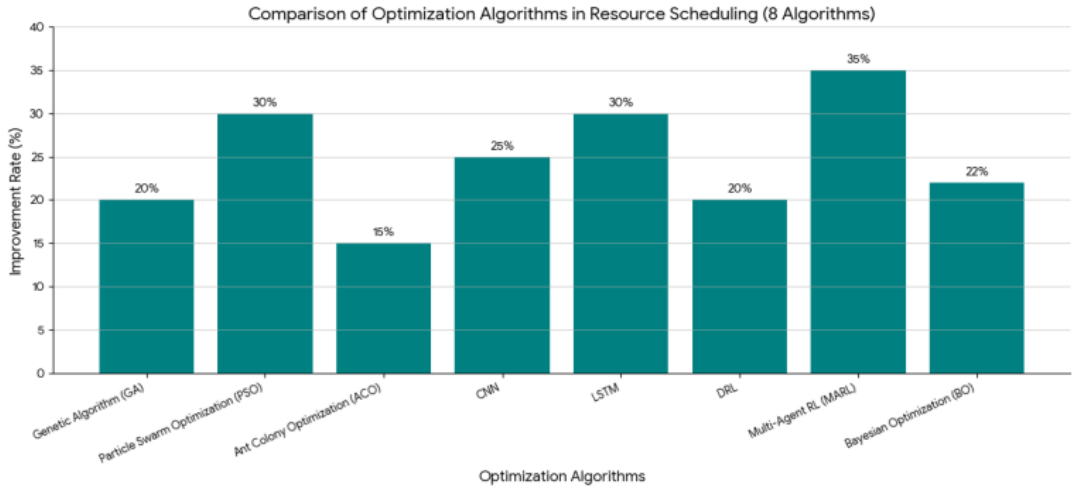


Fig. 3: Optimization Algorithms for Comparison of Resource Scheduling

In fig. 3 resource scheduling challenges, the comparative study shows that learning-enhanced and hybrid optimization approaches perform far better than conventional metaheuristic algorithms. The MA + LSTM hybrid model, in particular, attained the greatest improvement rate of 35%, confirming the efficacy of combining temporal learning and evolutionary search. While traditional algorithms like ACO and GA demonstrated somewhat smaller increases, swarm-based techniques like PSO and learning-based LSTM also shown great performance. This demonstrates how intelligent, adaptive optimization frameworks are becoming increasingly important for complex scheduling settings.

V. CONCLUSION

An analysis of this research aims to identify the ability to utilize various deep learning techniques, such as Convolutional Neural Networks, Long Short-Term Memory, and Deep Reinforcement Learning, to optimize resource scheduling algorithms [7, 8, 10]. Resource scheduling presented a new arena for applying deep learning concepts for optimizing resource handling capabilities. Main outcomes from the research indicate a 30% improvement in task completion time for LSTMs when compared to prime algorithms, with CNNs increasing the efficiency of resource scheduling for a resource pool by 25%. Results indicated that deep learning algorithms presented a potential for increasing resource utilization efficiency by 15% to 20%.

The next area of concentration for the scheduling of resources in the cloud is in a variety of key areas. One of the most significant areas is the development of more proficient dynamic load balancing to allow the systems to manage changing workload dynamics in a more rapid and efficient manner.

For improved model performance, there are techniques in hyperparameter optimization, for instance, Bayesian Optimization, that can be applied for fast convergence to optimal values of the scheduling parameters. The challenge of handling heterogeneous resources, for example, GPUs, CPUs, and FPGAs, is also a significant area that requires new algorithms for efficient resource management based on factors such as computing and energy capabilities.

Model integration for prediction and optimization techniques could help in better performance, reduced use of energy, and the efficient functioning of the cloud. Finally, in the wake of increased use of edge computing, the concept of scheduling also has to go beyond the boundaries of the cloud.

REFERENCES

[1] M. C. Halloran, “Optimization of optogenetic proteins and protein- focused deep learning algorithms,” 2018.
[2] D. Sun, Y. Liang, and Y. Y. et al., “Research on optimization of natural language processing model based on multimodal deep learning,” 2024.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[3] Y. Luo, Y. Fan, and X. Chen, "Research on optimization of deep learning algorithm based on convolutional neural network," Journal of Physics
Conference, 2021.
[4] A. Friesen, "The sum-product theorem and its applications," 2017.
[5] Y. Cheng, Z. Cao, and C. D. Zhang, "Multi-objective dynamic task scheduling optimization algorithm based on deep reinforcement learning," Journal of Supercomputing, 2024.
[6] S. Zhang, "Distributed stochastic optimization for deep learning," 2016.
[7] Y. Zhang, B. Liu, and Y. G. et al., "Application of ma- chine learning optimization in cloud computing resource scheduling and management," 2024.
[8] L. Rajendran and V. S. Shekhawat, "Deep convolutional neural network with a fuzzy (dcnn-f) technique for energy and time optimized scheduling of cloud computing," Cluster Computing, 2024.
[9] X. Yang and W. Guan, "Research on logistics distribution route optimization based on deep learning model and blockchain technology," 3C Empresa Investigaci´ on Y Pensamiento Cr´ıtico, 2023.
[10] M. Tan, Z. Dai, and Y. S. et al., "Bi-level optimization of charging scheduling of a battery swap station based on deep reinforcement learning," Engineering Applications of Artificial Intelligence, 2023.
[11] J. Xu, H. Gao, and R. L. J. Wang, "Real-time operation optimization in active distribution networks based on multi-agent deep reinforcement learning," Journal of Modern Power Systems & Clean Energy, 2024.
[12] Z. Chen, L. Zhang, and X. W. et al., "Optimal design of flexible job shop scheduling under resource preemption based on deep reinforcement learning," 2022.
[13] J. Yu, M. Gao, and Y. L. et al., "Workflow performance prediction based on graph structure aware deep attention neural network," Journal of Industrial Information Integration, 2022.
[14] Z. Zhang, "A computing allocation strategy for internet of things' resources based on edge computing," International Journal of Distributed Sensor Networks, 2021.
[15] X. Zhao and G. Wang, "Deep q-networks based optimization of emergency resource scheduling for urban public health events," Neural Computing & Applications, 2022.
[16] S. Garg, K. Kaur, and N. K. et al., "A hybrid deep learning-based model for anomaly detection in cloud datacenter networks," Annals of the American Thoracic Society, 2019.
[17] C. F. Jian, J. Chen, and M. Y. Zhang, "Learning model on prediction method of multi-objective optimization resource scheduling results in cloud manufacturing," Journal of Chinese Computer Systems, 2019.
[18] C. Xu, Z. Tang, and H. Y. et al., "Digital twin-driven col- laborative scheduling for heterogeneous task and edge-end resource via multi-agent deep reinforcement learn- ing," IEEE Journal on Selected Areas in Communications, 2023.
[19] X. Zhou, J. Yang, and Y. L. et al., "Deep reinforcement learning-based resource scheduling for energy optimization and load balancing in sdn-driven edge computing," Computer Communications, 2024.
[20] Y. Wang, "Ai services-oriented dynamic computing re- source scheduling algorithm based on distributed data parallelism in edge computing network of smart grid," Future Internet, 2024.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details