INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

OPEN ACCESS

Received: 23-08-2024

Accepted: 25-10-2024

Published: 09-12-2024

Citation: Thilagavathy C (2024) Leveraging Machine and Deep Learning Models for Load Balancing Strategies in Cloud Computing. Indian Journal of Science and Technology 17(45): 4722-4731. https://doi.org/10.17485/IJST/v17i45.2728

* **Corresponding author**.

thilaga.c@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Leveraging Machine and Deep Learning Models for Load Balancing Strategies in Cloud Computing

C Thilagavathy[1]*

1 Assistant Professor, Department of Information Technology, CMS College of Science and Commerce (Autonomous), Coimbatore, Tamil Nadu, India

Abstract

Objectives: To evaluate the efficiency of task prediction and resource allocation for load balancing (LB) in the cloud environment using the combined approach like random Forest(RF) for task prediction and Particle Swarm optimization for optimization and Convolutional Neural Networks (PSO-CNN) for resource prediction and allocation. **Methods:** The ensemble approach in the present study uses Random Forest (RF), a machine learning (ML) model for task prediction and Particle Swarm Optimization (PSO+CNN), a bio-inspired algorithm and Deep Learning (DL) model for optimization and resource allocation. The study employs PSO techniques to optimize CNN in order to address the investigation of algorithmic optimization in DL. The results show that the suggested model outperforms the other models like CNN-LSTM(Long Short-term memory), CNN-GRU(Gated Recurrent Unit), and PSO – SVM(Support Vector Machine) to increase the performance and efficacy of the cloud systems. The experiment is implemented using Python and assessed using Google Cluster dataset that is accessible to the public. **Findings:** The use of ML and DL techniques are found to be more efficient in cloud infrastructure than the conventional methods. The study examines the performance of the RF, PSO and CNN and the hybrid RF-PSO-CNN models. The accuracy, precision, and F1. Score metrics were used to assess the performance of the classification models. The recommended model RF-PSO-CNN outperforms them with an accuracy of 90% than the contrasted methods like CNN-LSTM, CNN- GRU and PSO-SVM. As a result, both the classification assessment metrics and resource consumption show that the proposed model performs effectively. **Novelty:** The novel ensemble approach suggests the combined RF-PSO-CNN for LB in cloud Computing. The task predicted by RF is assigned to the resource chosen by PSO and CNN, thereby improving the efficiency of task prediction and resource allocation. Most of the research uses any two ML or DL methods for either predicting the tasks to be scheduled or which resource to allocate. The study uses a combination of the ML (RF) method, bio-inspired algorithm (PSO) and a DL (CNN) model for both task and resource prediction concurrently and it examines the effectiveness of LB in the cloud context.

Keywords: Load Balancing (LB); Task scheduling; Resource allocation; Random Forest (RF); Convolutional Neural Networks (CNN); Particle Swarm Optimization (PSO)

1 Introduction

Task scheduling (TS) is an important challenge for CC, a technology that offers scalable and adaptable resources for computing. This challenge has an immediate impact on system efficiency and customer satisfaction. The TS problem is NP-complete, which makes solving it tough.[1] Effective utilization of resources is also equally important in CC through proper task allocation to the resources. Despite the increasing complexity of Cloud computing, upcoming cloud systems will necessitate enhanced utilization techniques. In the dynamic world of cloud computing, LB sets out as an essential element for optimizing performance, assuring availability, and increasing the scalability of data center networks. Cloud infrastructures are made up of hosts, data centers, and virtual machines (VMs) and allocating tasks is vital for obtaining optimal results. Effective scheduling needs to be done to minimize response times, costs, and wasted time. Optimizing resource utilization and improving overall cloud service efficiency requires effective scheduling of tasks[2].

Virtualization technology allows for the hosting and running of various programs on a global scale, resulting in resource and application scalability. CC resources are virtualized and characterized as VM's, which are an alternative technique that offers resources dynamically under the CC model. CC services have numerous issues, including resource management, fair task distribution, CPU usage, task scheduling, electrical consumption, security, quality of service, energy conservation, and so on. Researchers in CC have focused on the scheduling problem of distributing heterogeneous dispersed resources[3].

Recent advances in LB approaches are those that leverage ML in cloud infrastructure. Many ML based workload prediction algorithms have been created by leveraging their cognitive and learning characteristics[4]. ML enables the effective distribution of workloads across servers by evaluating real-time data. This results in better resource utilization and improved performance indicators such as throughput and response time. The integration of ML enables dynamic LB decisions based on evolving workload trends. This guarantees optimal utilization of resources and response to fluctuating demand. Organizations that use ML in cloud LB can achieve improved performance, scalability, fault tolerance, cost effectiveness, and overall operational excellence in their cloud systems.

A hybrid model for resource allocation employing Monte Carlo Tree Search (MCTS) and LSTM are studied, and the simulation shows that maintaining constant traffic patterns boosted MCTS performance. However, the search methods are difficult to adopt because traffic patterns might change quickly[5].

The LB strategy for splitting workload across virtual servers based on an altered version of the double Q-learning technique is suggested in [6]. The suggested algorithm is executed on a LB controller and utilizes requests from users utilizing software defined network techniques. The biggest disadvantage of the CC architecture is the longer execution time, which results in job deadlines. Previously, numerous methodologies were proposed to reduce the execution time, but these strategies are only applicable to a few workloads.

Neelakantan et al.[7] proposed a unique Whale-based Convolution Neural Framework (WbCNF) to enhance the task allocation mechanism and reduce task completion time. The findings indicate that the number of tasks required for the experiment has decreased due to the calculation time. Badri et al.[8] provides a novel technique for optimal job allocation with improved security in the CC system. A unique CNN opti mizing modified butterfly optimization (CNN-MBO) technique for assigning tasks is proposed, with the goal of optimizing throughput while decreasing makespan. Moreover, a modified RSA(Rivest, Shamir and Adleman) technique is used to encode the data, ensuring safe transmission of data. Zhen et al. [9] introduces a scheduling technique built on hierarchical reinforcement learning (RL). An algorithm like this might balance LB, power usage, and job completion time all at once. It has two agents: one low-level (L-Agent) for resource allocation and one high-level (H-Agent) for task selection. The H-Agent uses one-dimensional CNN(1DCNN) and multi-hop attention graph neural networks (MAGNA) to represent task and resource information. In a dynamic context, the L-Agent maximizes the mapping of tasks to resources by utilizing a double deep Q network (DQN) to determine the optimal strategy and objective function. Extensive computing resources, including processing capability and memory, may be required for DQN to train effectively, particularly in intricate settings.

Abdennebi et al. [10] proposes methods for load distribution and balancing using heuristics and ML. It uses RF, multiple linear regression (MLR), and Adaboost(Ada) techniques to dynamically select a processing unit for every receiving request based on CPU and GPU response time forecasts. An AI-assisted hybrid solution known as the Content-aware ML-based LB Scheduler (CA-MLBS) is proposed by Adil et al. [11]. The allocation system CA-MLBS uses ML and meta-heuristic techniques to classify files based on their kind. To do this, a SVM-based classifier is utilized to categorize user tasks based on the data (video, audio, picture, or text). A meta-heuristic method using PSO is utilized to map users' tasks to the cloud.

The most common algorithms in the reviewed papers include LR, RF classifier (RF) ANN, CNN, and LSTM with Recurrent Neural Network (LSTM -RNN). It is evident from the literature study that due to the dynamic nature of the Cloud environment, the effectiveness of the algorithms depends on various parameters such as the number of user requests, the number of VM's and the availability of the resources. The table lists the recent studies in resource allocation for the cloud using the methods to compare with the suggested ensemble approach.

Table 1. Recent literature studies in Resource prediction methods (Considered methods)

References	Method used	Purpose	Limitations
[1]	GA and GWO	Task scheduling	Best solution will not be obtained for complex problems
[2]	Tabu Search (T), Bayesian Classification (B), and Whale Optimization (W)	Optimization to improve the effectiveness of task scheduling	Tabu search varies depending on the problem to be solved
[6]	Double Q-Learning	LB for the fair allocation of resources	High computational requirements
[7]	Whale based Convolutional Neural Network	Task allocation	Low optimization and precision
[8]	CNN-MBO	Task scheduling	Lack of transparency
[9]	1DCNN and MAGNA	Efficient allocation of resources	Computationally expensive
[12]	CNN with GRU	Resource Allocation To minimize energy and cost	The use of GRU is computationally costly and needs enormous volumes of data for training.
[13]	DCNN with LSTM	Workload Prediction Energy usage and SLA's	Slower , requires more computational resources and labeled data.
[14]	PSO with SVM Resource demand prediction	Estimate future resource demands and allocate accordingly	SVM is not suitable for large datasets and it consumes more memory.

Any LB techniques suggested in the literature either predict the task or predict the resource. For an efficient LB, allocating the task to the available resources is crucial; it involves not only the resource prediction but also the job or task Selection. The tasks or user's requests are also to be chosen in such a way for the predicted resources. For instance, a lengthy task should not be allocated to a certain machine that involves critical processing. Current research on cloud LB heavily utilizes DL methods. However, because of the restricted resource-level provisioning, there is still an issue with acquiring noisy multilayered issues in workload.

This study focuses on allocating the task to an appropriate resource by considering the criteria for both tasks and resources and addressing the constantly changing characteristics of tasks in the cloud paradigm, a scheduling mechanism that automatically makes decisions based on impending tasks on the cloud interface and currently running tasks in the underlying virtual resources is suggested Firstly it uses, RF for task classification due to its ease, simple, faster training and time-consuming nature. Secondly, it uses PSO and CNN for resource prediction because of their automatic feature extraction and to lower the method's overall complexity, as well as the dependency and number of training epochs needed respectively.

The main contribution of the study include

1. Suggesting an ensemble ML, DL and bio-inspired algorithm for LB employing an amalgamation of RF-PSO-CNN techniques.
2. Assessing the efficiency of the combined model RF-PSO-CNN for task and resource allocation.

2 Methodology

This study suggests an LB technique by combining two or more AI strategies. The main focus of the study is to maximize resource usage and improve the prediction of resources using lightweight techniques. The proposed technique uses an amalgamation of existing techniques like Random Forest for task classification and CNN-PSO for optimization and resource allocation. The evaluation is conducted using the Google cluster workload records dataset, which is a publicly available dataset. Each task in the dataset is made up of several concurrent tasks that are carried out on different machines. The dataset includes metrics like CPU, memory, and disk usage. The suggested method is contrasted with hybrid techniques like CNN-LSTM, CNN-GRU and PSO-SVM. The overall architecture of the proposed load balancer using RF, PSO and CNN is presented in Figure 1.

The user's requests are collected by the Queue Manager before the jobs tasks get allocated. The task data is also collected by the Task data collection mechanism which is going to be a dataset for the Random Forest (RF) algorithm for the classification of tasks. Once the task is decided by the RF algorithm, the task is now ready for execution and is handled again by the queue manager for further allocation of Virtual machines. Similarly, the virtual Manager uses the CNN predicted results for identifying the VM.

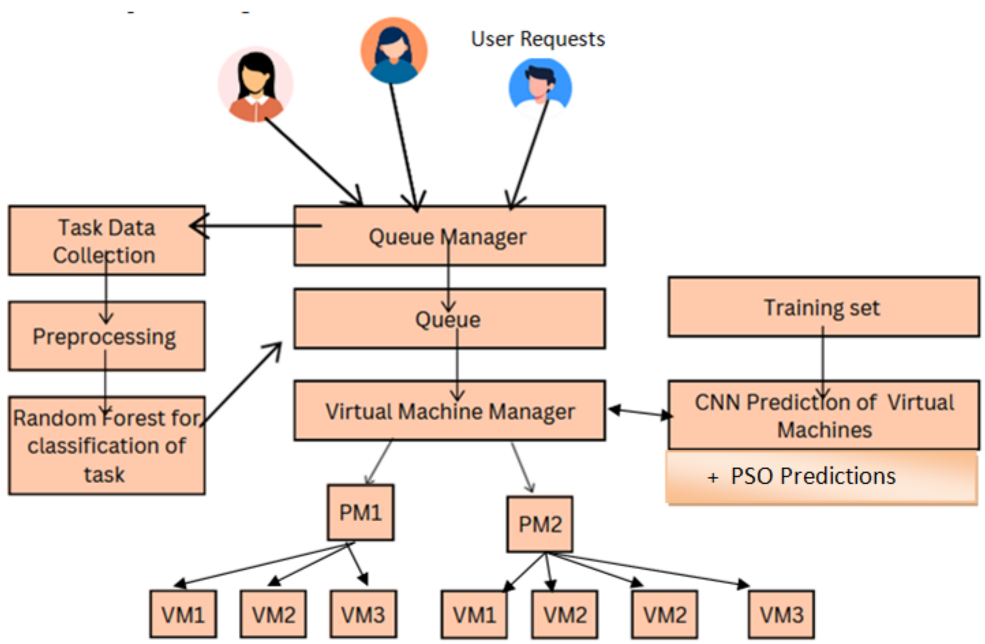

Fig 1. Overall Architecture for the classification of tasks and Resource Prediction

2.1 Random Forest for Task classification

A classifier called RF aggregates multiple decision trees derived from various dataset subsets and takes an average of them to improve the dataset's predictive performance. It is based on the principle of ensemble learning, which combines several classifiers to solve a challenging issue and increase the accuracy of the model. RF can analyze large data sets with many attributes, making it capable of detecting trends and abnormalities[15]. The steps involved in the working of RF are as follows and it is depicted in Figure 2.

Step -1: Select samples at random from a training set or collection of data.

Step -2: Generate a decision tree for every set of training data.

Step -3: Voting will be done by averaging the choices in the decision tree.

Step -4: Lastly, determine the ultimate outcome by selecting the most popular anticipated result.

In CC, the RF feature selection technique is employed to determine the task to be allocated to the VM based on the learning from the collected data and the present data. The basic parameters considered for task allocation are listed in Table 1. The task is classified based on priority, size, energy usage, the type of data the task involved, the number of currently residing on the queue, etc.

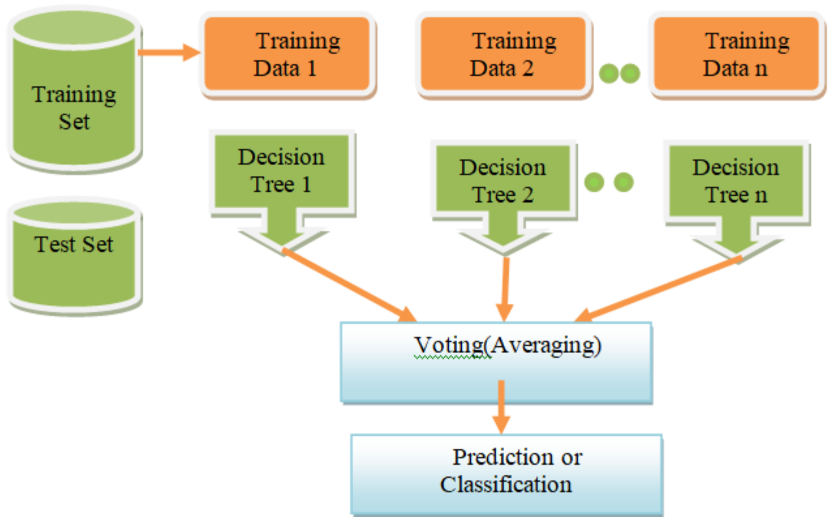

Fig 2. Random Forest Algorithm

2.2 PSO and CNN for Resource Prediction

PSO is a population-based methodology, and it utilizes a number of particles to form the swarm. Each particle represents a possible result. The group of possible answers cooperates and communicates simultaneously. Each particle in the swarm finds the best landing site by exploring the search zone. A swarm, or collection of flying elements, represents the changing solutions, and the search space, thus, is the set of possible solutions. Over the course of the generations (iterations), each particle saves both the optimal solution of the swarm and its own best solution. It then modifies two parameters: position and velocity, or flight speed. Specifically, each particle continuously modifies its flying speed based on its own flight experiences as well as those of its neighbors. Like this, it makes an effort to change where it is by using its current position, speed, and the separation between its current position and the optimal distance for both the swarm and the individual. When the flock of particles, which resemble birds, reaches the global optimal level, the issue of optimization is solved. It continues to travel into an appropriate zone. The two main equations of PSO are as follows

Velocity Equation:

$$v_i^{t+1} = v_i^t + c_1 r_1 (pbest_i^t - p_i^t) + c_2 r_2 (gbest^t - p_i^t) \tag{1}$$

Position Equation:

$$p_i^{t+1} = p_i^t + v_i^{t+1} \tag{2}$$

In Equations (1) and (2) V_i =velocity of a particle, c_1, c_2 denotes acceleration constants, r_1, r_2 are the random numbers between 0 and 1, t is the iteration number, pbest is the best solution of a particle, gbest is the best solution of swarm, and f is the fitness or objective function.

Then the CNN architecture is set to train using PSO and the flowchart for the algorithm is shown in Figure 3. It is a type of DL neural network architecture consisting of three types of layers: convolutional layers, pooling layers, and fully connected layers. The one-dimensional CNN is used to extract unique distinguishing features from the VM's data.

The suggested model uses one-dimensional CNN to forecast CPU use on cloud environments at different time intervals by using connected CNN. It is used to predict or choose a VM based on the CPU power, usage, configuration, data handling facility etc[16,17]. The CNN component retrieves complex distinctive characteristics derived from VM workload information. Table 2 lists some of the basic parameters which help the CNN to predict the VM for the tasks execution.

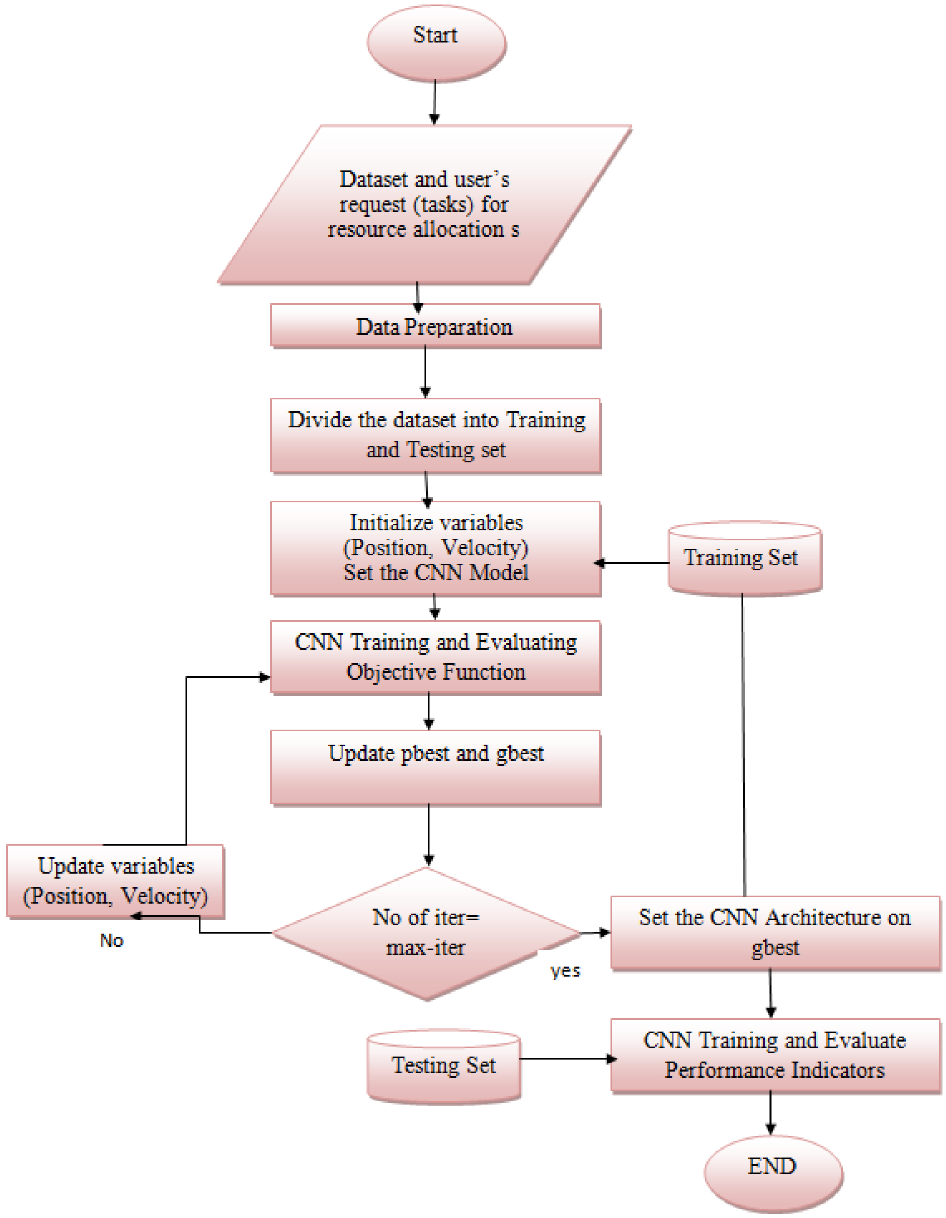

Fig 3. Flowchart PSO-CNN Algorithm

3 Results and Discussions

Firstly, RF algorithm classifies the real time tasks based on the available dataset. The task is classified based on priority, size, energy usage, the type of data the task involved, the number of currently residing on the queue, etc., then it chooses the next task to be scheduled and informs the queue manager for further scheduling. RF calculates the average of all the predictions made by decision trees, canceling out any biases. Therefore, it is unaffected due to the over fitting issue. It can expand with data quantity, making it appropriate for cloud operations containing vast amounts of metrics and logs. It typically requires less computing power. The main disadvantage of RF is that it is time consuming because it produces predictions relatively slowly, since they use an enormous amount of decision trees. All of the trees in the forest have to predict and vote on the identical input. The criteria for task classification are given in Table 2.

Table 2. Criteria for Task Classification

Parameters for task classification
Priority
Length of the job
Energy usage of the job
Nature of data of the job(audio,video, text)
Number of jobs at time (users request)
Approximate response time based the size, data type etc,

Secondly, PSO+CNN is used to extract the features of the VM's and predict the corresponding VM for the task to run. The Table 3 lists the parameters considered for resource allocation using PSO+CNN

Table 3. Criteria for task Allocation to VM

S.No	**Parameters for workload allocation to Virtual machines**
1	Number of virtual machines in the cluster
2	Processing power
3	Data Handling capacity
4	Configuration settings
5	Scheduling algorithm
6	CPU utilization
7	Turnaround time
8	Response time
9	Throughput
10	Waiting time in the queue
11	No of users request
12	Current workload
13	Platform/Software
14	Network utilization
15	Disk space utilization
16	RAM utilization

Variables for both real and virtual machines include CPU power, RAM disc storage, and network bandwidth [14]. For instance, the formula from Equations (1), (2), (3), (4) and (5) will assist in predicting overloaded machines by analyzing the present status of each machine.

$$CPU\ utilization = \frac{\sum Currently\_running\_task}{Total\ Capacity} * 100 \tag{3}$$

$$RAM\ \ utilization = \frac{\sum RAM\ \_current\_utilization}{Total\ RAM} * 100 \tag{4}$$

$$BW\_\ utilization = \frac{\sum BW\ \_current\_utilization}{Total\ RAM} * 100 \tag{5}$$

$$Storage\ utilization = \frac{\sum Storage\_current\_utilization}{Total\_Storage\_Capacity} * 100 \tag{6}$$

$$Resource\ utilization = \sum^{CPU_{utilization}+RAM_{utilization}+BW_{utilization}+Storage_{utilization}} * \tag{7}$$

The experiments were conducted in Python by varying the number of tasks as 20,40,60,80 and 100 with the number of VM's as 20. The resource utilization in terms of CPU, memory, bandwidth and storage were observed for each iteration. The resource utilization is calculated using the Equation (7).

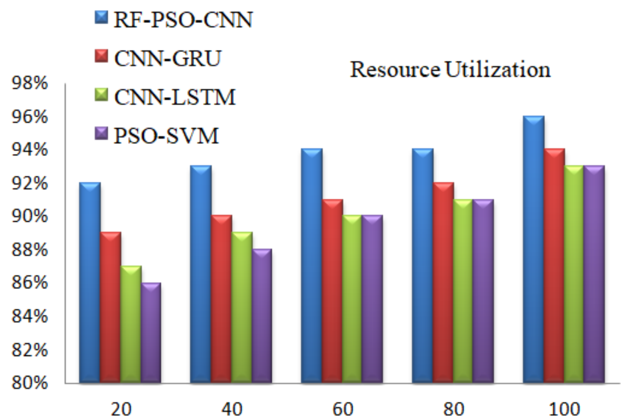

Fig 4. Resource utilization of the proposed methods with compared methods

The results of the ensemble method RF-PSO-CNN are then compared with the hybrid methods like CNN-LSTM, CNN-GRU, PSO+SVM. The graphical representation of the comparative analysis is shown in Figure 4. The percentage of resource utilization is found to be higher for the ensemble method RF-PSO-CNN than the other contrasted methods. Hence it is evident that the results of RF+PSO+CNN show improved performance than the other two methods with regard to resource utilization. This study focuses on only the resource utilization criteria of the methods. Similarly, the other parameters like makespan, response time, and execution time can also be observed and studied further.

3.1 Classification Evaluation

The ensemble and the hybrid classification models are then evaluated for performance using the classification performance metrics like accuracy, precision, and F1-Score.

One of the most commonly employed metrics is accuracy. The following formulas were used to calculate accuracy, precision and F1-Score.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{8}$$

$$Precision = \frac{TP}{TP+FP} \tag{9}$$

$$F1-Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{10}$$

The performance of the tasks and resource predictions for the suggested model and the compared model is observed in terms of accuracy, precision and F1. Score and the results are represented in Figure 5.

The performance of the classification models was evaluated using metrics like accuracy, precision and F1 Score. These measured values are found to be higher in the case of the suggested model RF+PSO+CNN than the other hybrid models considered in this study. Thus, the suggested model performs well in terms of resource utilization as well as the classification assessment metrics. The method is also compared with the other state of art methods listed in the table and the resource usage is estimated.

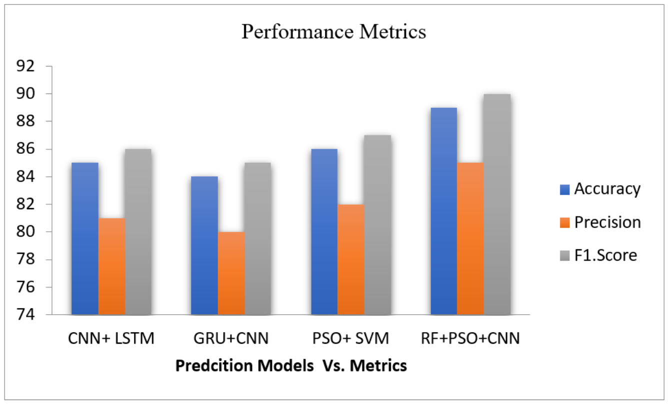

Fig 5. Prediction Models Vs Metrics

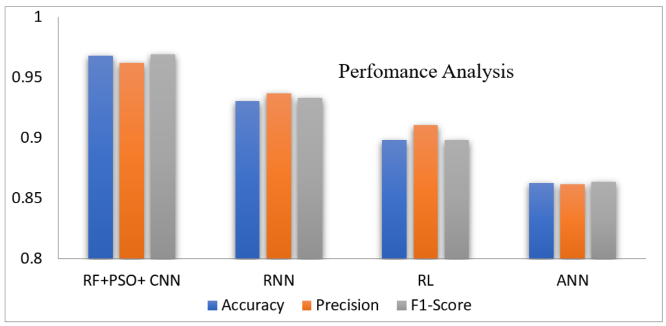

Fig 6. Prediction Model vs Metrics (other existing methods)

Figure 6 shows the graphical representation of the comparative analysis of the suggested RF+PSO+CNN method with Recurrent Neural Networks(RNN), Reinforcement Learning(RL), and Artificial Neural Networks(ANN). The analysis of the performance indicators shows that the suggested methods outperform the other contrasted methods.

Several researchers have consistently suggested enhanced PSO algorithms to address engineering issues across several domains[18,19]. Furthermore, the PSO algorithm used in this study is found to be effective in solving the problem in less time. It uses a simple framework with high convergence speed, and it is applied to various optimization problems[20]. PSO can avoid local optima and adjust to search spaces to discover the optimal solution to difficult issues.

The primary difficulty in using CNN was deciding on its architecture and parameters. Experimental parameters frequently have limited generalizability and only pertain to particular study data. A new method has been devised to optimize CNN that evolves CNN's architecture and parameters using the random yet supervised mode of the PSO algorithm. Effective optimization of the parameters was achieved by training the dataset in the assigned search space[21]. Even though the suggested RF-PSO-CNN is better in many aspects, there are still certain drawbacks and more advancements are required.

4 Conclusion

This study explores the use of RF for task prediction and PSO +CNN for resource prediction for LB in cloud. These results show that the suggested model, RF-PSO-CNN outperforms the other models like CNN-LSTM, GRU-CNN and PSO-SVM with an average accuracy of 94%. Moreover, the optimized CNN of PSO algorithm performs better than the conventional methods. It is also observed that the usage of ML and DL techniques in cloud systems poses several challenges due to the changing workloads and its dynamism. Furthermore, it is difficult to create robust ML and DL models to adapt to altering patterns and changing system behavior. The study is further extended to explore the other ML and DL models for LB in the cloud to improve efficiency in terms of resource usage and prediction performance. It is concluded the PSO-CNN model is capable of accurately predicting the resources for the tasks in the cloud context. It has a beneficial leading role in the study of evolutionary deep learning-based task scheduling and resource allocation.

References

1) Behera I, Sobhanayak S. Task scheduling optimization in heterogeneous cloud computing environments: A hybrid GA-GWO approach. *Journal of Parallel and Distributed Computing*. 2024;183. Available from: https://doi.org/10.1016/j.jpdc.2023.104766.
2) Sandhu R, Faiz M, Kaur H, Srivastava A, Narayan V. Enhancement in performance of cloud computing task scheduling using optimization strategies. *Cluster Computing*. 2024;27:6265–6288. Available from: https://doi.org/10.1007/s10586-023-04254-w.
3) Srikanth GU, Geetha R. Effectiveness Review of the Machine Learning Algorithms for Scheduling in Cloud Environment. *Archives of Computational Methods in Engineering*. 2023;30:3769–3789. Available from: https://doi.org/10.1007/s11831-023-09921-0.
4) Saxena D, Kumar J, Singh AK, Schmid S. Performance Analysis of Machine Learning Centered Workload Prediction Models for Cloud. *IEEE Transactions on Parallel and Distributed Systems*. 2023;34(4):1313–1330. Available from: https://ieeexplore.ieee.org/document/10029931.
5) Jeyaraman J, Bayani SV, Malaiyappan JNA. Optimizing Resource Allocation in Cloud Computing Using Machine Learning. *European Journal of Technology*. 2024;8(3):12–22. Available from: https://doi.org/10.47672/ejt.2007.
6) Tennakoon D, Chowdhury M, Luan TH. Cloud-based load balancing using double Q-learning for improved Quality of Service. *Wireless Networks*. 2023;29:1043–1050. Available from: https://doi.org/10.1007/s11276-018-1888-8.
7) Neelakantan P, Sudhakar N. Proficient job scheduling in cloud computation using an optimized machine learning strategy. *International Journal of Information Technology*. 2023;15(12):2409–2421. Available from: https://www.researchgate.net/publication/370745314_Proficient_job_scheduling_in_cloud_computation_using_an_optimized_machine_learning_strategy.
8) Badri S, Alghazzawi DM, Hasan SH, Alfayez F, Hasan SH, Rahman M, et al. An Efficient and Secure Model Using Adaptive Optimal Deep Learning for Task Scheduling in Cloud Computing. *Electronics*. 2023;12(6):1–14. Available from: https://www.mdpi.com/2079-9292/12/6/1441.
9) Zhang Z, Xu C, Liu K, Xu S, Huang L. A resource optimization scheduling model and algorithm for heterogeneous computing clusters based on GNN and RL. *The Journal of Supercomputing*. 2024;80:24138–24172. Available from: https://doi.org/10.1007/s11227-024-06383-4.
10) Abdennebi A, Elakaş A, Taşyaran F, Öztürk E, Kaya K, Yıldırım S. Machine learning-based load distribution and balancing in heterogeneous database management systems. *Concurrency and Computation: Practice and Experience*. 2022;34(4). Available from: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.6641.
11) Adil M, Nabi S, Aleem M, Diaz VG, Lin JCW. CA-MLBS: content-aware machine learning based load balancing scheduler in the cloud environment. *Expert Systems*. 2023;40(4):1–21. Available from: https://onlinelibrary.wiley.com/doi/epdf/10.1111/exsy.13150.
12) Khodaverdian Z, Sadr H, Edalatpanah SA, Nazari M. An energy aware resource allocation based on combination of CNN and GRU for virtual machine selection. *Multimedia tools and applications*. 2024;83:25769–25796. Available from: https://doi.org/10.1007/s11042-023-16488-2.
13) Sabyasachi AS, Sahoo BM, Ranganath A. Deep CNN and LSTM Approaches for Efficient Workload Prediction in Cloud Environment. *Procedia Computer Science*. 2024;235:2651–2661. Available from: https://doi.org/10.1016/j.procs.2024.04.250.
14) Sanjay N, Sreedharan S. Hybrid Bio-Inspired Optimization-based Cloud Resource Demand Prediction using Improved Support Vector Machine. *International Journal of Advanced Computer Science & Applications*. 2024;15(1). Available from: https://thesai.org/Publications/ViewPaper?Volume=15&Issue=1&Code=IJACSA&SerialNo=77.
15) Singh S, Verma SK. Congestion and Accident Alerts Using Cloud Load Balancing & Random Forest in VANET. *Wireless Pers Commun*. 2023;128:43–65. Available from: https://link.springer.com/article/10.1007/s11277-022-09473-9.
16) Leka HL, Fengli Z, Kenea AT, Tegene AT, Atandoh P, Hundera NW. A Hybrid CNN-LSTM Model for Virtual Machine Workload Forecasting in Cloud Data Center. In: 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP). IEEE. 2021;p. 474–478. Available from: https://ieeexplore.ieee.org/document/9674067.
17) Simaiya S, Lilhore UK, Sharma YK, Rao KBVB, Rao VVRM, Baliyan A, et al. A hybrid cloud load balancing and host utilization prediction method using deep learning and optimization techniques. *Scientific Reports*. 2024;14:1–18. Available from: https://doi.org/10.1038/s41598-024-51466-0.
18) Bi Y, Ni W, Liu Y, Lai L, Zhou X. Task Scheduling with Improved Particle Swarm Optimization in Cloud Data Center. In: B L, L C, ZG W, H L, C L, editors. Neural Information Processing. ICONIP 2023;vol. 14449 of Lecture Notes in Computer Science. Springer. 2024;p. 277–287. Available from: https://link.springer.com/chapter/10.1007/978-981-99-8067-3_21.
19) Qiao J, Wang G, Yang Z, Luo X, Chen J, Li K, et al. A hybrid particle swarm optimization algorithm for solving engineering problem. *Scientific Reports*. 2024;14(1):1–30. Available from: https://www.nature.com/articles/s41598-024-59034-2.
20) Pirozmand P, Jalalinejad H, Hosseinabadi AA, Mirkamali S, Li Y. An improved particle swarm optimization algorithm for task scheduling in cloud computing. *Journal of Ambient Intelligence and Humanized Computing*. 2023;14(4):4313–4327. Available from: https://link.springer.com/article/10.1007/s12652-023-04541-9.
21) You X, Zheng Z, Yang K, Yu L, Liu J, Chen J, et al. A PSO-CNN-Based Deep Learning Model for Predicting Forest Fire Risk on a National Scale. *Forests*. 2024;15(1):1–18. Available from: https://www.mdpi.com/1999-4907/15/1/86.