International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
***Article Received**: 02 March 2023 **Revised**: 28 April 2023 **Accepted**: 12 May 2023*

SGA Model for Prediction in Cloud Environment

Smitha Krishnan[1], Dr B.G Prasanthi[2]
[1]Department of Computer Science
SB College
Changanassery,kerala
nov30smitha@gmail.com
[2]Department of Computer Science
St Josephs College
Benguluru
nitai2009@gmail.com

Abstract— With virtual information, cloud computing has made applications available to users everywhere. Efficient asset workload forecasting could help the cloud achieve maximum resource utilisation. The effective utilization of resources and the reduction of datacentres power both depend heavily on load forecasting. The allocation of resources and task scheduling issues in clouds and virtualized systems are significantly impacted by CPU utilisation forecast. A resource manager uses utilisation projection to distribute workload between physical nodes, improving resource consumption effectiveness. When performing a virtual machine distribution job, a good estimation of CPU utilization enables the migration of one or more virtual servers, preventing the overflow of the real machineries. In a cloud system, scalability and flexibility are crucial characteristics. Predicting workload and demands would aid in optimal resource utilisation in a cloud setting. To improve allocation of resources and the effectiveness of the cloud service, workload assessment and future workload forecasting could be performed. The creation of an appropriate statistical method has begun. In this study, a simulation approach and a genetic algorithm were used to forecast workloads. In comparison to the earlier techniques, it is anticipated to produce results that are superior by having a lower error rate and higher forecasting reliability. The suggested method is examined utilizing statistics from the Bit brains datacentres. The study then analyses, summarises, and suggests future study paths in cloud environments.

Keywords: Cloud environment; Genetic algorithm; CPU Workload Forecasting; Seasonal Genetic algorithm; Resource management.

I. INTRODUCTION

With the virtualisation of physical resources in computer servers, cloud computing is a web-based virtual environment that could offer end users on-demand operations [1]. Owing to multitenant usage, shifting workload constraints, and increasingly complicated systems, resource monitoring is frequently a challenging operation in an information centre. Workloads in today's computer servers are largely non-linear. According to an IBM research, for example, cloud workloads' aggregate Central processing unit and memory consumption ranges from 17.76percent to 80 Percent [2]. With cloud computer servers, there is a significant resources inefficiencies because a cluster's Central processing unit and memory utilisation cannot surpass 60%, per a Google research [3]. Because of the workload's non-linear user activity, the quality of services is reduced, efficiency is unpredictable, and power utilization is excessive. Moreover, it drives up operational expenses and reduces profit for service operators. Resource utilisation must be optimised because data centres are costly to set up and maintain. Increased capacity utilization as well as lower running expenses could be achieved while maintaining the application's Quality of Service using a smart resource forecasting technique.

Based on extensive historical workloads, a projection system generates predictions regarding future need for a certain resource, such as Central processing unit, storage, disc, and networks. These forecasts could help with resource management choices including resource provisioning and Virtual machine aggregation as well as non-linear resource usage and power consumption in computer servers. For example, an effective resource allocation could be handled by a resource provisioning system depending on these future findings. Moreover, choices might be more pro-active than the existing reactive methods. In this context, CPU workload forecasts using ML algorithms are possible [4]. Since they are founded on actual information and have the ability to gain insight into extremely non-linear workload patterns brought on by numerous parameters in data centre systems, ML-based forecasts are the best option. Current resource forecasting studies concentrate on memory and central processing unit utilisation and disregard required provisioned resources like storage and central processing unit [5] [6]. These supplied resources significantly increase the amount of power utilized when a novel virtual machine is instantiated on a server [7]. Moreover, they disregard resource measurements like disc throughput, that directly affects a host's power utilisation [8].

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
***Article Received**: 02 March 2023 **Revised**: 28 April 2023 **Accepted**: 12 May 2023*

When combining virtual machine instances to conserve resources, networking performance is a crucial measure to take into account [9]. Moreover, a variety of methods based on machine learning have been employed to complete this research, but no single approach is effective at handling all non-linear workloads. In predicting both provisioned and utilised non-linear workloads with different metrics, such as provisioned Central processing unit, provisioned memory, Central processing unit workload utilization, memory consumption, disc throughput, and network throughput, it might be beneficial to implement an Standard genetic algorithm method that involves several learning algorithms.

I. Energy assessment, together with workload predictions, is essential to data centre resources planning. Data centre operators want to reduce entire energy usage through efficient utilization of resources because power utilisation is a major problem in computer servers. In contemporary computer servers, hosts are equipped with a range of monitors to keep an eye on power consumption [10]. Recent studies have concentrated on estimating the electrical usage for every virtual machine utilizing different energy frameworks. Calculating the power usage of virtual machine at the software level is challenging, though. For illustrate, the occurrences generated by every virtual machine on each core's last level caches are used to calculate the power usage of storage [11]. Determining the power of every Virtual machine is a challenging process because it needs to gather these information from last-level caches to estimate power usage [12]. Consequently, researchers decided to look at characteristics of comparable Virtual machines in various energy-consumption phases rather than computing power for every Virtual machines. In order to accomplish this, the accessible power utilization characteristics are examined, and Virtual machine with comparable patterns are found using genetic algorithm.

II. PROBLEM ISSUES IN CLOUDS

Recent years have seen tremendous growth in the field of cloud services, which is now becoming a commercial reality. The system hasn't been developed properly yet, though. There remain a few things that require attention.

- Task scheduling
- Resource management

The primary issues with grids and cloud technology both revolve around task management and resources provision. In the field of Information Technology, cloud computing is an innovative technique. The cost-benefit of different computing platforms is influenced by how frequently network operators deliver cloud resources to customers. Nonetheless, a variety of academics have provided a large number of task allocation methods, which are explored in relevant literature.

In this article, a more accurate CPU workload prediction technique using genetic algorithm (GA) is presented to It primarily uses Bit brains statistics, that comprises the provided and utilised resources efficiency of thousands of virtual machines deployed across many Clouds. For two tasks—predicting workloads and estimating the power condition of virtual machines— each offer a predicting proposed framework [13]. The Forecasting Mechanism is currently being implemented in this article. The primary findings of this study, in brief, are as described in the following: Utilizing characteristics made up of resources that have been allocated and used from a remote data centre that hosts the cloud, it investigates genetic algorithms for workload forecasting in nonlinear settings. Assessment methods including provisioned Central processing unit, RAM, memory and central processing unit utilisation, disc throughput, and network throughput are among the characteristics. It offers an innovative method for estimating the energy state at the VM level based on Virtual machine characteristics which might have an influence on power usage. The evolutionary algorithms have the lowest Root mean square error values for all characteristics in the CPU workload forecasting methods. The two main categories of resource management techniques are proactive as well as reactive. The proactive approach is simpler and more successful. There could be a few instances of inconsistent allocation of resources.

- Over-provisioning: Additional resources are given to processes
- Under-provisioning: Insufficient resources have been allocated for the active processes to function.
- Oscillation: Combination of the first two states

III. RELATED WORKS

Because of differences in the demanding workloads, cloud resource management needs complicated regulations and judgements to guarantee the proper utilisation computational power. In cloud systems, choosing the proper resource consumption isn't a simple easy decision to make. In order to appropriately predict the required resources, an effective resource forecasting system could be very useful in cloud resource administration. In this research, offer an ensembles CPU load forecasting approach that employs a Bayesian information parameter to choose the most suitable constituent model for every time frame according to cloud resource utilisation record. Moreover, use a pair of smooth filters to lessen the influence of anomalies on the recorded data sets. Researchers also provide a structure for handling cloud resources that includes a forecasting module to even more precisely forecast resource consumption. The experimental outcomes using the CoMon project's database show which the

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
***Article Received**: 02 March 2023 **Revised**: 28 April 2023 **Accepted**: 12 May 2023*

suggested method outperforms previous process that determines techniques in regards to precision. Yet, the difference between the inaccuracy gain brought on by omitting one of the estimation techniques and excluding other forecasting model is not appreciably distinct [14].

The ability to anticipate Central Processing Unit utilisation of physical computers in cloud information centres is still difficult, and very little has been researched in this area. The purpose of this paper is to introduce the DP-CUPA, a deep belief network and particle swarm optimization-based Central Processing Unit consumption forecasting system. There are 3 main phases in the DP-CUPA. The historical Central Processing Unit utilization statistics are initially pre-processed and standardised. After that, the grey and autoregressive algorithms are developed to serve as the basic estimation techniques and to add more input data to the Deep Belief Network training process. Lastly, the Deep Belief neural network is trained to forecast CPU utilisation and the Swarm optimization is utilised to evaluate the Deep neural characteristics. With the help of numerous experimentations and a real database of Google cluster utilization traces, the DP-effectiveness CUPA's is assessed. Yet, the PSO's limits and the disruption brought on by the supplementary information produced by the fundamental forecasters AR and GM [15].

Because of load changes, Software as a Service cloud services find it challenging to anticipate future resource requirements and, as a result, to deploy the necessary resources. In addition, cloud computing platforms have a huge number of virtual machines, that makes the forecasting issue more challenging owing to correlations in the massive amounts of workloads information stored in these Virtual machines. Because of this, precise resource utilisation prediction is still difficult, and only a small number of research have looked at how to anticipate central processing unit usage for virtual machines in cloud computing environment. In accordance with the concept of autonomous computation and a deep learning technique, this study suggests an intelligent and autonomous workload prediction model for cloud resource supply. In specifically, researchers present an effective deep learning algorithm based on a diffusion convolutional recurrent neural network that could be used to forecast future demands for central processing unit consumption and evaluate how to adapt to workloads changes in the upcoming interval. Because of the prevalence of inconsistent and nonlinear workloads in systems that use cloud computing, current deep learning algorithms that are frequently used couldn't indeed manage reliable real-time forecasting. The suggested deep learning algorithm aims to increase prediction precision and reduce the discrepancy among workload predictions and real demands. Utilizing trials on a real-world dataset of PlanetLab's central processing unit usage traces, the efficacy of the suggested DCRNN-based deep learning model was assessed. According to the data, the suggested method significantly outperformed than other deep learning algorithms already in usage, with a mean absolute percentage error and a root-mean-square error. However, the effectiveness of the suggested approach may differ in the event of workloads with a random process that are highly erratic [16].

One of the main features associated with cloud technology is flexibility, which draws a lot of Software as a Service suppliers to it in hopes of lowering the price of their services. By dynamically allocating and releasing computational assets in accordance with actual computation complexity, costs are minimised. The Service Level Agreement may be broken, nevertheless, if additional virtual capabilities are launched slowly. As a result, scaling up computing assets beforehand draws considerable interest when it comes to cloud provisioning predictions. The majority of current techniques, though, do not take multi-seasonality in cloud operations into account. This study suggests an approach for cloud resource provisioning assumption based on the Holt-Winters exponential smoothing technique. HoltWinters' exponential smoothing technique is extended in the suggested method to estimate cloud workloads with multiple seasonal changes. By using the Artificial Bee Colony algorithm to refine its characteristics, the suggested algorithm's accuracy of the model has increased. The effectiveness of the proposed methodology has been assessed and contrasted with double- and triple-exponential smoothing techniques. The findings demonstrate that the suggested algorithm is better competing techniques. Yet, in some circumstances, gaps among events might vary and must be taken into account when making predictions [17].

The rivalry for end customers among Cloud providers delivering comparable services increases as businesses switch from desktop programmes to cloud-based Software as a Service applications installed on public Clouds. Cloud-based businesses must provide their consumers with high-quality service to thrive in this cutthroat industry. Otherwise, they run the danger of losing customers to rivals. Yet, because workloads change over the course of time, it's occasionally difficult to achieve the quality service with an economical quantity of resources. This problem is fixable using proactive dynamic resource provisioning, which could also predict the future resources application demands and assign those assets prior to reducing them when they're no longer needed. In this article, researchers demonstrate how an ARIMA-based Cloud workload forecasting component for Software solutions was implemented. Using actual traces of web server queries, researchers present the prediction built on the ARIMA framework and assess its reliability of prospective workload forecasting. We also assess how the acquired accuracy affects

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
***Article Received**: 02 March 2023 **Revised**: 28 April 2023 **Accepted**: 12 May 2023*

resource utilisation effectiveness and quality of service. According to simulated findings, the system could accomplish a median precision, that leads to efficient resource use with little negative influence on QoS. Nevertheless, creating new virtual machines takes time [18].

The flexibility of cloud technology enables cloud service providers to effectively manage high processing and storage demands. Maintaining an appropriate level of cloud utilisation and service-level agreement requires proactive allocation of cloud workloads. It's necessary to forecast resource requirements for a few minutes in advance to solve issues like delay associated with newer virtual machine setup, energy reduction, and effective resource provisioning. Due to the extremely dynamic nature of cloud activities, Processor and memory utilisation changes widely. Moreover, there is a significant forecasting errors and inaccurate outcomes with the current forecasting models. Thus, an unique tuned support vector regression (TSVR) technique that precisely chooses 3 SVR characteristics utilizing a hybrid genetic algorithm and particle swarm optimization method is suggested in this work. The approach incorporates a chaotic sequence to increase prediction precision and avoid early convergence at the same time. Researchers conducted a simulation research utilising Google cloud traces to show the prediction precision of the TSVR algorithm. The simulation findings demonstrate that, in accordance of accepted measures, the suggested tuned support vector regression approach outperforms more traditional models in terms of predictive accuracy. Unfortunately, it was unable to accurately estimate some large changes in workload [19].

The autoregressive integrated moving average model finds linear elements in the Processor and storage use characteristics in the cloud records. The artificial neural network employs the residuals from the autoregressive integrated moving average model to identify and amplify nonlinear elements in the traces. Using both linear and nonlinear elements, the resources use patterns are anticipated. The Savitzky-Golay filter derives a variety of projected numbers from the expected and historical values. In a cloud environment, point value forecasting might not be the most effective technique for projecting multi-step resource use. The estimating error could be mitigated by including a variety of possible standards. To deal with the mistake caused by overestimating or underestimating memory and central processing unit consumption, researchers utilize the method described by Engelbrecht HA and van Greunen M. Statistically based evaluation makes use of the predictive performance utilizing BitBrain and Google's 29-day trail. However, due to the accumulating of forecast errors, it may not always be the case [20].

A growing number of businesses are moving their operations to cloud data centres as a commonly utilized Technology solution. To attain a high level of service quality for their customers, cloud service suppliers must give benefits for cloud computing that are both flexible and expense. The demands of Virtual Machines vary over time, which makes it difficult for Cloud computing services to achieve Quality of service with cost-effective resources. To effectively manage cloud resources, it is imperative to offer a reliable approach for task scheduling for virtual machines. In this research, researchers first evaluate the effectiveness of certain typical modern load estimation techniques. In order to give sufficient time for task scheduling based on anticipated workload, we recommend a strategy that involves conducting the forecast a specific amount of time before the projected time point. Researchers offer a clustering-based workload forecasting strategy that initially groups all the jobs into various classes and then trains a forecasting model for every group separately in order to further increase predictive performance. The results of the trace-driven trials using the Google cluster trace show that the workloads forecasting techniques that utilize clustering outperforms existing comparative approaches and increase forecast accuracy to almost 90percentage points in both memory and CPU resources. The greater parameter, though, results in much more expensive testing and training period [21].

To meet current obligations, a prominent technique for inferring complex multidimensional data from cloud settings is workload forecasting utilizing Deep Learning. Both the structure and the information's accuracy affect the model's overall performance. As a result, the data used to train the approach has to be of a high standard. Existing studies in this field, however, may have relied on a single source of information or have neglected to consider the necessity of uniformity for objective and reliable analysis. The effectiveness of Learning algorithms decreases as a consequence. The paper presents a technical indicator of exploiting the time series characteristics of actual workload from the Standardized Workload Format's Simultaneous Workloads Archives utilising deep learning frameworks such as Recurrent Neural Networks, Multilayer Perception, Convolutional Neural Networks and Long Short-Term Memory. The Mean Absolute Error and Root Mean Squared Error error metrics are utilized to assess the robustness of such algorithms. The results show that, when contrasted to the other methods, the Long Short-Term Memory model performs the best. Moreover, to the extent possible, the literature lacks significant information on the use of DL to task scheduling in systems that use cloud computing. Researcher’s offer a thorough foundation on resources planning and load forecasting utilising DL to solve these problems. Finally, we compare several bodies of work's models, error measurements, and data sources. Nevertheless, the approaches

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
***Article Received**: 02 March 2023 **Revised**: 28 April 2023 **Accepted**: 12 May 2023*

examined as a whole appear to have inherent flaws in terms of the restricted hybrid modelling configurations, the ignorance of the simulation model, as well as the constrained selection of datasets mostly due to accessibility [22].

In this requirement, researchers use the MSaDE learning method, an improved form of the differential evolution technique, identifying cloud workloads utilizing artificial neural networks. The artificial neural network forecasting proposed based on MSaDE approach is evaluated with two test scenarios for the works have proposed of National Aeronautics and Space Administration server and Saskatchewan servers at varied look-ahead periods. In order to show the increased accuracy of training the neural network-based prediction prototype to employ the MSaDE approach, training is carried out using both the self-adaptive evolutionary algorithm technique and the backpropagation neural technique. Designers examined the root - mean - square squared error across all forecast ranges and the mean root mean error function. The results show that, when contrasted to certain other methodologies, the MSaDE-based neural network-based forecasting method predicts cloud workloads more precisely. The suggested artificial neural network proposed methodology, which increased the artificial neural network model's effectiveness as well as accuracy rate for the purpose of forecasting future workloads, is trained using the MSaDE method. However, this process requires a lot of time [23].

The CPU Workload prediction methodologies literature review is displayed in Table 1 below. As a conclusion, it indicates that the suggested approach outperforms the already employed methods.

IV. MATERIALS AND METHODS

IV.I Data Collection

Rnd information set was employed to perform the projection in this research. This is a summary of the collected database. The database comprises the statistical information of 1,750 virtual machines from a distributed Bitbrains datacentre. Bitbrains is a service operator with a focus on regulated hosting and enterprise solutions compute. Many large financial institutions and credit card issuers are among the clients. In this work, a Python software was employed to acquire the stimulation outcome. The descriptions of varying factors in cloud settings are shown in Table 2. According to the month that the measurements are collected, the records in the Rnd directory are divided into 3 sub-directories. Every file has a row-based structure, where every row indicates an efficiency parameter measurement. The "; \ t" character divides every column in a row. In this work, a Python software was employed to acquire the stimulation outcome [24].

Table 1: Description of various parameters in cloud environments

Parameters	Description
Timestamp	No. of milliseconds since 1970-01-01
Network transmission and reception rate	In terms of Kilobytes per seconds
Memory provisioned	The amount of virtual machine storage in Kilobytes
CPU capacity provisioned	The quantity of cores times the velocity per core determines the Processors' performance in Megahertz
CPU usage	In terms of Megahertz and Percentage
Memory usage	The amount of currently utilised storage in Kilobytes
Disk read and write rate	In terms of Kilobytes per seconds
CPU cores	Amount of allocated virtualized Central processing unit cores

IV.II Proposed Methodology

A resource manager uses utilisation projection to distribute workload between physical nodes, improving resource consumption effectiveness. When performing a virtual machine distribution job, a good estimation of CPU utilization enables the migration of one or more virtual servers, preventing the overflow of the real machineries. In a cloud system, scalability and flexibility are crucial characteristics. Predicting workload and demands would aid in optimal resource utilisation in a cloud setting. To improve allocation of resources and the effectiveness of the cloud service, workload assessment and future workload forecasting could be performed. The operation of the suggested prediction model is the main topic of this section in this study, a simulation approach and a genetic algorithm were used to forecast workloads. The outcome of the CPU workload estimator and the data centre's present condition. Fig. 1 depicts the suggested system's workflow.

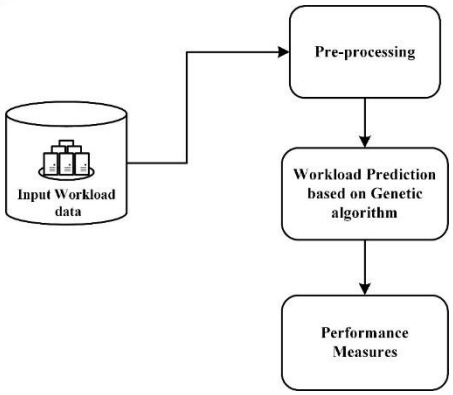

Figure 1: Workflow of the proposed System

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
***Article Received**: 02 March 2023 **Revised**: 28 April 2023 **Accepted**: 12 May 2023*

Pre-processing

The measured time-space for Virtual machine usage for the suggested technique is saved in historical workload statistics and utilized as parameters for the forecasting model utilizing the genetic algorithm. The data comprises Central processing unit and memory utilisation rates. During training the network, the required information is taken into account as Virtual machine utilisation, which combines memory and processing power usage. The statistics are initially aggregated as part of the pre-processing procedure to create a distinct time interval. Utilizing received information from a database and the min-max normalisation technique, the following step. Equation 1 was used to normalise both memory and CPU usage in this case to a range of (0,100). Dataset is divided into two databases, referred to as the training sample and evaluation database, after being pre-processed. Throughout the entire dataset, 80percent of the information is utilised for training, and the leftover 20percent is employed to assess the forecasting accurateness over 2 stages.

$$\grave{V}_j = \frac{v_j - v_{minimum}}{v_{maximum} - v_{minimum}} \quad (1)$$

Genetic Algorithm Approach

The Genetic Algorithm is a reliable optimisation metaheuristic technique that is motivated by natural as well as biological decisions based on Darwin's theory of survival [25]. The Genetic Algorithm is independent of any particular conceptual reality and without any assumptions like linearity, steady, or homogeneity. Chromosomes, a demographic set, fitness criteria, mutations, and selection procedures are all involved. This offers a collection of chromosome-based solutions known as demographics. In accordance with the belief that the newly generated community would be superior to the older demographic, the solutions from one population are employed to create a new group. Also, according to the fitness value, alternatives are selected to generate innovative solutions. The aforementioned process would be continued until the final population's quantity of offspring matches the number of individuals in the starting community. These procedures employ the genetic operations mutations and crossovers. In this investigation, the mutation and crossover probabilities were both set to 0.01 for both the double point crossovers and Gaussian metaheuristic algorithms. Even though the fundamental principles of utilizing GA to solve optimization problems are the same, differing requirements could affect how the procedure is carried out. The optimality of the resultant solution generated by a specific algorithmic implementation is largely dependent on selecting the appropriate genetic characteristics. The population size, maximum epoch, elitism ratio, mutation rate, crossovers rate, and other fundamental characteristics would all be present in every GA research.

The genetic algorithm's overall flowchart is displayed in Figure 1. The following list includes the Genetic Algorithm's required measures:

step 1: initiate: chromosomes should be created via random populations.
step 2: fitness: find out how each chromosome's fitness value functions in its populations.
step 3: new population: the procedures that are given below should be followed to establish a new demographic. *selection: recognize two parental chromosomes from such a group of people based on their fitness.* *crossover: cross the parents in order to produce a new-born spring, with the potential for a crossover. in the absence of crossovers, children are identical replicas of their families.* *mutation: create new progeny with a chance of mutation at every gene.* *accepting: in the new population, locate new offspring.*
step 4: replace: utilize the newly established demographic for the algorithm's subsequent iterations.
step 5: test: the optimum result for the present demographic would halt and then resume whenever the ended prerequisites are met.
step 6: replace: utilize the newly established demographic for the algorithm's subsequent iterations.
step 7: loop: return to step 2.

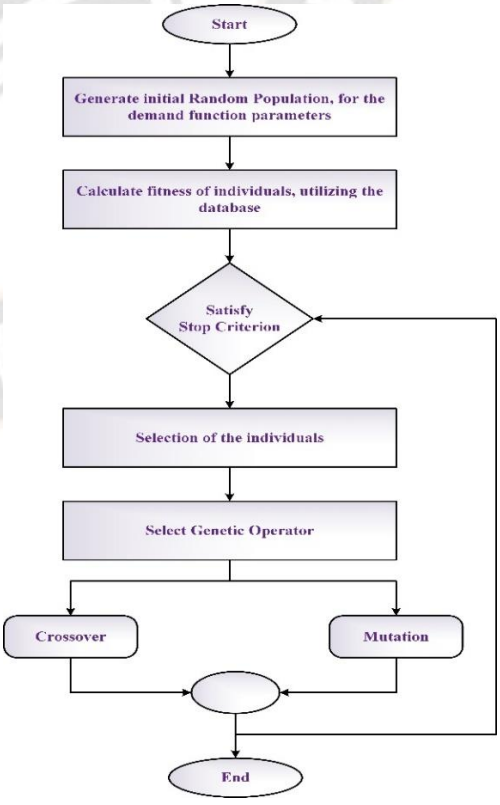

Figure 2: Workflow model of Genetic algorithm

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
Article Received: 02 March 2023 Revised: 28 April 2023 Accepted: 12 May 2023

Seasonal Genetic Algorithm (SGA)

A modification on the standard Genetic Algorithm (GA) known as the Seasonal Genetic Algorithm (SGA) adds a seasonal element to enhance its effectiveness in resolving optimization issues. In Seasonal Genetic Algorithm, every cycle of the population of feasible solutions corresponds to a seasonal as it changes through duration. This method's goal is to give the algorithm a periodicity that corresponds to the seasonal variations seen in many real-world challenges. Similar to a conventional Genetic Algorithm, the population of feasible solutions in Seasonal Genetic Algorithm is randomly initialised and then iteratively evolves over a number of decades. Every candidate solution is assessed utilizing a fitness value at the conclusion for every iteration, and the top solutions are chosen to make up the following generation. The procedure is repeatedly carried out in Seasonal Genetic Algorithm, for each cycle standing in for a season. The fitness feature might change with every season to account for the shifting weather patterns. Seasonal Genetic Algorithm may adapt to shifts in the challenge space over time by including a seasonal component, and it is especially useful in issues where the ideal solution shifts over time or is impacted by various influences that fluctuate periodically.

To determine the fitness of chromosomal as well as the s_o , the research has created an aggregate function. The suggested aggregation functions model is illustrated in equation (2), wherein I represents a chromosome's genes for $u = 1, 2, 3, 4$, and I measures the importance of s_u in calculating s_o for every one of the four criterion evaluations. In the range [0, 1], every δ_u is a numerical value that was created at random. The weight vector $\sum \delta_u s_i$ of the criterion evaluations is the numerator in the right-hand side of. To allow for the possibility that the criterion evaluations had varying degrees of effect on the s_o, the numerator is computed by dividing with $\sum \delta_u$. To avoid situations in which all the values of δ_u were "0", the variable one was inserted. The s_i that seems to be nearer to s_o may benefit more from the normalisation procedure.

$$s_o = \frac{\sum_{u=1}^{4} \delta_u s_u}{\sqrt{\sum_{u=1}^{4} \delta_u + 1}} \qquad (2)$$

The fitness ratio of every chromosomal in the demographic must be calculated in order to train the conceptual approach utilizing the GA, as was previously described. With k representing a specific characteristic and s_v a being the $v - th$ average rating from the data source, fitness value in equation (3) is employed to compute the inaccuracy created by every chromosomal depending on the root mean square error calculation technique.

$$f = \sqrt{1/m \sum_{v=1}^{m} \left(s_c^v - \frac{\sum_{u=1}^{4} \delta_u s_u}{\sqrt{\sum_{u=1}^{4} \delta_u + 1}} \right)^2} \qquad (3)$$

The GA-based method now demands a standard explanation for the fundamental genetic processes since it has the aggregating and fitness functions. The chromosomes' genes have been altered employing a switch process that randomly selects two copies of each gene from each chromosomal and swaps their locations. The research utilized the standard crossovers technique, which couples two parents to create 2 offspring. Despite the fact that the traits are chosen at random from the parents, and some fundamental combinations of the parents' genes that may be utilized to make kids. To create 2 additional chromosomes, the procedure randomly selects half of each parent's entire number of genes.

Seasonal Genetic Algorithm (SGA)
Step 1: Initialization The procedure begins by initialising a random population of candidate solutions, typically expressed as a collection of genes. Typically, the scope of the task and the capabilities of the computing infrastructure define the overall population.
Step 2: Fitness evaluation A fitness value that gauges efficiency or effectiveness is employed to assess every potential solution in the population. The fitness function could be changed according to the seasonal to represent shifting state of the environment. Higher fitness options are deemed superior and are more likely to be chosen for reproduction.
Step 3: Selection The selection procedure decides whether solutions are employed to replicate and construct the following generation. In SGA, alternatives with greater fitness values are given preference in the selection process. To represent the shifting issue situations, the selection pressure could be changed throughout the seasons.
Step 4: Reproduction The chosen options are then blended using genetic operators like crossover and mutation to create offspring that have a combination of their parents' traits. Throughout various seasons, these genetic operators could also be altered to adapt shifting environmental conditions.
Step 5: Seasonal adaptation The seasonal component of SGA modifies the fitness function and genetic algorithms to take into account the shifting issue variables. As an illustration, in a problem of seasonal optimization, the fitness value might alter depending on the season or outside variables like the weather. Similar to this, it is possible to modify the genetic operators to respond to changes in the problematic situation.
Step 6: Termination Until a terminating requirement, such as a maximum number of generations or a workable solution, is reached, the method keeps evolving the population across several seasons. The algorithm's outcome is the optimal solution it has been able to find.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
Article Received: 02 March 2023 Revised: 28 April 2023 Accepted: 12 May 2023

In conclusion, Seasonal Genetic Algorithm (SGA) contains a seasonal component that modifies the genetic operators and fitness value to reflect the shifting issue conditions. A population of potential solutions are evolved by the algorithm over several seasons, and the optimal solution are chosen to form the subsequent generation. The SGA technique could deal with challenges with shifts in the environment or where the ideal answer fluctuates annually by adding a periodicity. In this research, an SGA approach was employed in order to anticipate the CPU workload. Statistics from 1750 virtual machines connected to fast Storage Area Network devices were obtained for this study using Bit brains' fast storage trace. Every file contains the provisioned Processor capacity, provisioned CPU utilisation, provisioned memory capability, actual memory consumption, provisioned network Input and Output throughput, provisioned disc I/O throughput, as well as the amount of cores provisioned. Bit brains trace data from the SGA resource workload was utilized to evaluate the suggested technique. The prediction's reliability is evaluated using MSE, MAE, and Root mean square Error measures.

Model operation

This system would forecast the seasonal demand for our Central processing unit, and the demand function's characteristics are determined utilizing a genetic algorithm. There have been numerous phases to the project. Predicting the CPU's requirement at $n - th$ sec for $(n-1)$ times. Genetic Algorithm is utilized to forecast the demand function's characteristics.

Demand Function

It is presumable that the demand function is a time-dependent variable and has a periodical character for seasonal fluctuations. Figure 3 displays a depiction of a sinusoidal waveform. As a result, the periodical pulse is sinusoidal in character and could be represented in equation 4.

$$\boldsymbol{x = P\sin\big(Q(y+R)\big) + S} \qquad (4)$$

$Where, \quad P = Amplitude$

$Q = Period\ of\ \frac{2\pi}{Q}$

$R = Vertical\ shift$

$S = Phase\ shift\ left$

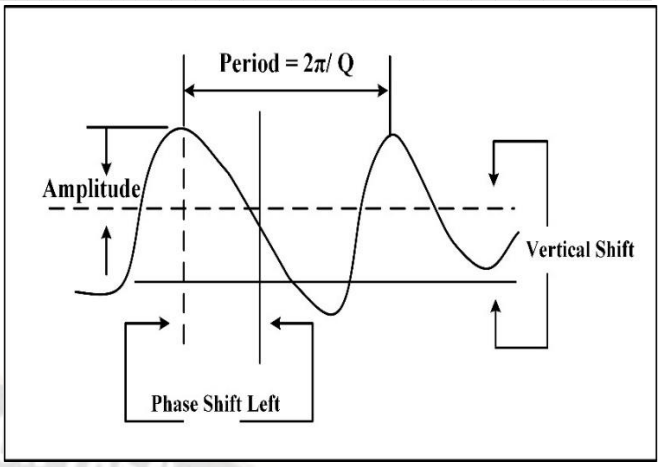

Figure 3: Sine Wave Representation

By altering the Demand functions, it may anticipate the central processing unit consumption at the 50th second in the past (i.e., the 49th second).

$$\boldsymbol{CPU\ Demand\ Dem} = \frac{x.sin(y.t_u - 1 + z) + D + Q.t_u - 1}{U_{i-1}{}^{e}} \qquad (5)$$

$Where\ D = Vertical\ offset$

$Q = Trend\ factor$

$A = Amplitude$

$Y = Frequency$

$Z = Horizontal\ offset$

$U = CPU\ utilization$

$e = Elasticity\ of\ CPU\ utilization$

Fitness Function

An individual's possibility of being a member of the following generations is based on comparative fitness. Using the following expression, the Fitness function can be determined.

$$f = \frac{\Sigma C_t}{t} - Dem \qquad (6)$$

$$i.e., Demand_{actual} - Demand_{pred}$$

Where C is the actual CPU usage at time t and f represents the fitness ratio.

V. RESULTS AND DISCUSSION

V.1 Assessment of models

In this portion, investigations are carried out to confirm our modelling approach for predicting CPU demand. Using the Bit Brains data centre would allow for an evaluation of the CPU load prediction method. In order to arrive at the relative CPU loading rates, which fall between [0,1], the absolute load ratings were divided by the respective capacity.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
***Article Received**: 02 March 2023 **Revised**: 28 April 2023 **Accepted**: 12 May 2023*

The database contains CPU workload time series information points with a measurement interval within one second. To assess the effectiveness of genetic algorithm CPU Workload forecasting, 3 quantitative performance metrics are being used. Below is a representation of these characteristics, which include mean squared error, mean absolute error and root mean square error. The suggested approach for error measurement evaluation is shown in Table 2.

Table 2: Performance Assessment of the proposed system

Measures / Methods	*MSE*	*MAE*	*RMSE*
DL	85.45	78.99	75.22
TSRV	87.33	89.23	65.99
ARIMA – ANN	65.98	76.54	81.45
DCRNN	70.01	61.99	90.34
Proposed SGA	61.77	54.88	50.87

Mean Squared Error

It measures the mean squared difference among expected and observed values. The improved forecast is represented by a lower MSE rating.

$$MSE = \frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2 \quad (7)$$

Mean Absolute Error

It is a gauge of the typical size of forecast inaccuracies.

$$MAE = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j| \quad (8)$$

Root Mean Square Error

The variance of the forecasting inaccuracy is represented by the RMSE.

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2} \quad (9)$$

Figure 4 compares the seasonal genetic algorithm under consideration with earlier techniques employing error measures. Additionally, it indicates that the suggested technique has a lower inaccuracy rate than existing approaches.

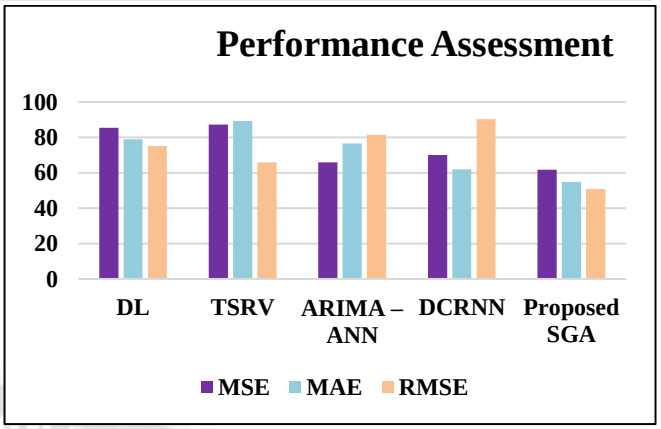

Figure 4: Performance Assessment of the proposed system

4.2 Simulation Outcome

The simulation was performed in Python, and the findings for a selected data points are presented below.

- ***crossing_ over_ probability = 0,1***
- ***mutation_ probability = 0,5***
- ***generation = 1000***

Phase 1

The (x-axis) of the CPU use graph is plotted against time (y-axis). Real statistics is depicted in blue, and projected information is depicted in green. The fitting of the framework and its predictions are shown graphically to be appropriate due to the reduced error among the estimated and actual statistics. The graphic representation of initial phase is shown in the figure 5.

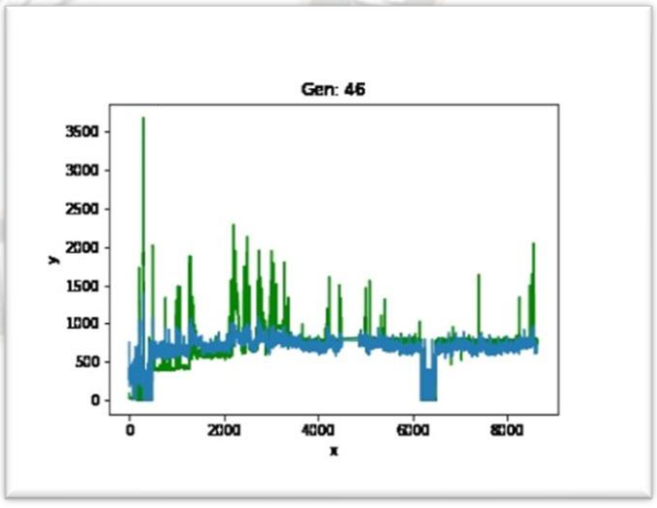

Figure 5: CPU Utilization-time Graph for Initial Phase

Phase 2

A various fitness estimation method could be employed in conjunction with the identical methodologies. The Mean Squared Error technique is utilized to estimate fitness.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.7046
***Article Received**: 02 March 2023 **Revised**: 28 April 2023 **Accepted**: 12 May 2023*

The figure demonstrates the graphical depiction for the same data. The graphic representation of the same is shown in the figure 6. Again, this demonstrates the extent to which the model works.

$$F(MSE) = mean\,(sqrt(Demreal - Dem_{pred})2)) \quad (10)$$

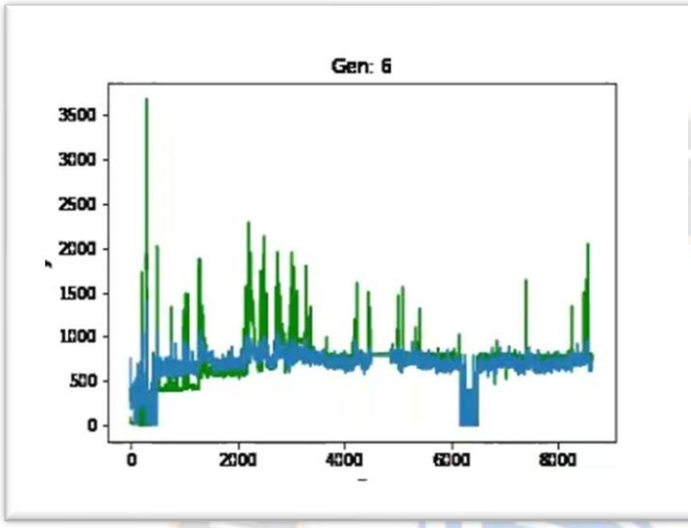

Figure 6: CPU Utilization-time Graph for Phase 2

Phase 3

Using Root mean square error, the Fitness is determined in the third phase simulation portion. The graphic representation of the CPU Utilization-time Graph for Phase 3 is shown in the figure 7.

$$RMSE = mean\,(sqrt\,((Dem_{real} - DEmpred)2)) \quad (11)$$

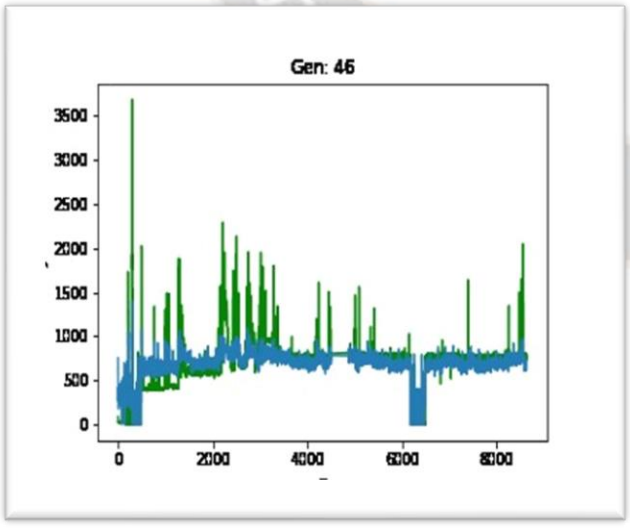

Figure 7: CPU Utilization-time Graph for Phase 3

Phase 4

The tuning variable, also known as the learning algorithm, is used to produce the global minima in the stage four simulation segment titled "Adding Normalization to the Fitness Values." Its value is 0.01, for example.

$$Error = mean\,((Dem_{real} - Dem_{pred})2) + tuning\_param * sum((ind)) \quad (12)$$

VI. CONCLUSION

In cloud technology, workload forecasting constitutes one of the crucial components of proactive management systems and auto-scaling. In order to increase cloud efficiency, reduce power usage rates, achieve the necessary quality of service levels, forecast the power usage of data centres, and increase the scalability of providers of cloud services, precise workload forecasting is of greatest priority. Yet workload forecasting in the environment of cloud computing is a difficult problem, and there are numerous techniques integrating deep learning, data mining, and mathematical analysis to solve this problem. Hence, an unique strategy focused on CPU workload prediction in a cloud computing context was presented in this study. The current method of the research, that employs the seasonal genetic algorithm technique, constitutes a systematic framework. An algorithm has been created to aid with seasonal trend prediction. Both server-based and serverless solutions could be used to verify concept. According to the test findings, this approach performs better than other algorithms. Other alternatives could be tested as future research in place of the sine curve. It is possible to alter the fitness function expression.

Funding:This research did not get any funding

REFERENCES

[1] S.-Y. Hsieh, C.-S. Liu, R. Buyya, and A. Y. Zomaya, “Utilization-prediction-aware virtual machine consolidation approach for energy-efficient cloud data centers,” J. Parallel Distrib. Comput., vol. 139, pp. 99–109, May 2020, doi: 10.1016/j.jpdc.2019.12.014.

[2] R. Birke, L. Y. Chen, and E. Smirni, “Data Centers in the Wild: A Large Performance Study”.

[3] C. Reiss, A. Tumanov, G. R. Ganger, R. H. Katz, and M. A. Kozuch, “Heterogeneity and dynamicity of clouds at scale: Google trace analysis,” in Proceedings of the Third ACM Symposium on Cloud Computing, San Jose California, Oct. 2012, pp. 1–13. doi: 10.1145/2391229.2391236.

[4] D. Jeff, “ML for system, system for ML, keynote talk in Workshop on ML for Systems, NIPS.” 2018.

[5] E. Cortez, A. Bonde, A. Muzio, M. Russinovich, M. Fontoura, and R. Bianchini, “Resource Central: Understanding and Predicting Workloads for Improved Resource Management in Large Cloud Platforms,” in Proceedings of the 26th Symposium on Operating Systems Principles, Shanghai

China, Oct. 2017, pp. 153–167. doi: 10.1145/3132747.3132772.

[6] F. Farahnakian, T. Pahikkala, P. Liljeberg, J. Plosila, N. T. Hieu, and H. Tenhunen, "Energy-Aware VM Consolidation in Cloud Data Centers Using Utilization Prediction Model," IEEE Trans. Cloud Comput., vol. 7, no. 2, pp. 524–536, Apr. 2019, doi: 10.1109/TCC.2016.2617374.

[7] P. Roose, M. Soltane, D. Makhlouf, and K. Okba, "PREDICTIONS & MODELING ENERGY CONSUMPTION FOR IT DATA CENTER INFRASTRUCTURE," presented at the Advances in Intelligent Systems and Computing, 2018, vol. 912, p. 1. doi: 10/document.

[8] W. Lin, W. Wu, H. Wang, J. Z. Wang, and C.-H. Hsu, "Experimental and quantitative analysis of server power model for cloud data centers," Future Gener. Comput. Syst., vol. 86, pp. 940–950, Sep. 2018, doi: 10.1016/j.future.2016.11.034.

[9] R. Shaw, E. Howley, and E. Barrett, "An energy efficient anti-correlated virtual machine placement algorithm using resource usage predictions," Simul. Model. Pract. Theory, vol. 93, pp. 322–342, May 2019, doi: 10.1016/j.simpat.2018.09.019.

[10] M. Aldossary and K. Djemame, "Energy-based Cost Model of Virtual Machines in a Cloud Environment," in 2018 Fifth International Symposium on Innovation in Information and Communication Technology (ISIICT), Oct. 2018, pp. 1–8. doi: 10.1109/ISIICT.2018.8613288.

[11] C. Gu, P. Shi, S. Shi, H. Huang, and X. Jia, "A Tree Regression-Based Approach for VM Power Metering," IEEE Access, vol. 3, pp. 610–621, 2015, doi: 10.1109/ACCESS.2015.2430276.

[12] K.-J. Ye, Z.-H. Wu, X.-H. Jiang, and Q. He, "Power management of virtualized cloud computing platform," Chin. J. Comput., vol. 35, no. 6, pp. 1262–1285, 2012.

[13] S. Shen, V. Van Beek, and A. Iosup, "Statistical Characterization of Business-Critical Workloads Hosted in Cloud Datacenters," in 2015 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, May 2015, pp. 465–474. doi: 10.1109/CCGrid.2015.60.

[14] S. Tofighy, A. A. Rahmanian, and M. Ghobaei-Arani, "An ensemble CPU load prediction algorithm using a Bayesian information criterion and smooth filters in a cloud computing environment," Softw. Pract. Exp., vol. 48, no. 12, pp. 2257–2277, 2018.

[15] Y. Wen, Y. Wang, J. Liu, B. Cao, and Q. Fu, "CPU usage prediction for cloud resource provisioning based on deep belief network and particle swarm optimization," Concurr. Comput. Pract. Exp., vol. 32, no. 14, p. e5730, 2020.

[16] M. S. Al-Asaly, M. A. Bencherif, A. Alsanad, and M. M. Hassan, "A deep learning-based resource usage prediction model for resource provisioning in an autonomic cloud computing environment," Neural Comput. Appl., pp. 1–18, 2021.

[17] A. A., "Using Multiple Seasonal Holt-Winters Exponential Smoothing to Predict Cloud Resource Provisioning," Int. J. Adv. Comput. Sci. Appl., vol. 7, no. 11, 2016, doi: 10.14569/IJACSA.2016.071113.

[18] R. N. Calheiros, E. Masoumi, R. Ranjan, and R. Buyya, "Workload Prediction Using ARIMA Model and Its Impact on Cloud Applications' QoS," IEEE Trans. Cloud Comput., vol. 3, no. 4, pp. 449–458, Oct. 2015, doi: 10.1109/TCC.2014.2350475.

[19] M. Barati and S. Sharifian, "A hybrid heuristic-based tuned support vector regression model for cloud load prediction," J. Supercomput., vol. 71, pp. 4235–4259, 2015.

[20] K. L. Devi and S. Valli, "Time series-based workload prediction using the statistical hybrid model for the cloud environment," Computing, vol. 105, no. 2, pp. 353–374, Feb. 2023, doi: 10.1007/s00607-022-01129-7.

[21] J. Gao, H. Wang, and H. Shen, "Machine learning based workload prediction in cloud computing," in 2020 29th international conference on computer communications and networks (ICCCN), 2020, pp. 1–9.

[22] Z. Ahamed, M. Khemakhem, F. Eassa, F. Alsolami, and A. S. A.-M. Al-Ghamdi, "Technical Study of Deep Learning in Cloud Computing for Accurate Workload Prediction," Electronics, vol. 12, no. 3, p. 650, 2023.

[23] M. Attia, M. Arafa, E. Sallam, and M. Fahmy, "Application of an enhanced self-adapting differential evolution algorithm to workload prediction in cloud computing," Int J Inf Technol Comput Sci, vol. 11, no. 8, pp. 33–40, 2019.

[24] A. Hussain and M. Aleem, "GoCJ: Google cloud jobs dataset for distributed and cloud computing infrastructures," Data, vol. 3, no. 4, p. 38, 2018.

[25] R. Malhotra, N. Singh, and Y. Singh, "Genetic algorithms: Concepts, design for optimization of process controllers," Comput. Inf. Sci., vol. 4, no. 2, p. 39, 2011.