I.J. Image, Graphics and Signal Processing, **2025, 5, 42-61**
Published Online on October 8, 2025 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijigsp.2025.05.04

DS-RAE: Advance Hybrid Double Residual Self-Attention and Recursive Autoencoder Approach for Dynamic Cloud Workload Forecasting

G. M. Kiran*
Shridevi Institute of Engineering and Technology, Karnataka 572106 India
E-mail: gmkiran899@gmail.com
ORCID iD: https://orcid.org//0009-0006-6241-1528
*Corresponding Author

A. Aparna Rajesh
Aryan institute of engineering and technology, Odisha 752050 India
E-mail: abkakade22@gmail.com
ORCID iD: https://orcid.org//0000-0001-6932-7616

D. Basavesha
Shridevi Institute of Engineering and Technology, Karnataka 572106 India
E-mail: balakrishnanapp2020@gmail.com
ORCID iD: https://orcid.org//0009-0000-0388-4524

Received: 08 February, 2024; Revised: 14 May, 2024; Accepted: 12 May, 2025; Published: 08 October, 2025

Abstract: Infrastructure as a service is used for resource management. Resources that will be available on demand are effectively managed using the resource management module. Predicting CPU and memory usage assists with resource management when cloud resources are provided. This study uses a hybrid DS-RAE model to forecast CPU and memory utilization in the future. Predictions are made using the range of values found, which is helpful for resource management. The memory and CPU use patterns in the cloud traces are identified by the Double Channel Residual Self-Attention Temporal Convolution Network (DSTNW) model as having linear components. The Recursive Autoencoder (RAE) model for tracing and enlarging nonlinear components and power consumption was developed using the DSTNW model. Gathers the raw data taken from the system's operational state, such as bandwidth, disk I/O time, disk space, CPU, and memory utilization. Discover patterns and oscillations in the workload trace by preprocessing the data to increase the prediction efficacy of this model. During data pre-processing, missing value edge computing and z-score normalization are used to select the important properties from raw data samples, eliminate irrelevant elements, and normalize them. After that, preprocessing utilizes a dynamization of the sliding window to improve the proposed model's accuracy on non-random workloads. Next, utilize a hybrid DS-RAE to attain accurate workload forecasting. Comparing the suggested methodology with existing models, experimental results show that it offers a better trade-off between training time and accuracy. The suggested method provides higher performance, with an execution time of 32 seconds and an accuracy rate of 97%. According to the simulation results, the DS-RAE workload prediction method performs better than other algorithms.

Index Terms: Resource Management; CPU; Memory Utilization; Z-Score Normalization Dynamization of Sliding Window; Hybrid DS-RAE Model

1. Introduction

Deployment of several virtual computers on a group of physical computers (PMs) is known as cloud computing [1]. It makes widespread use of virtual machine consolidation possible, which is an essential tool for increasing resource efficiency and lowering energy costs. On the other hand, improper virtual machine (VM) consolidation can lead to frequent live migration, which can seriously impair quality of service and increase resource overhead [2]. Precisely projecting future virtual machine workloads is the most challenging problem when it comes to resolving cloud

computing's underutilization and energy usage. The quantity of client requirements gathered together with their delivery time is referred to as the workload. Reasonable virtual machine resource allocation to predicted workloads is made possible by accurate workload prediction [3]. These days, most cloud systems consist of several interconnected service components, each of which performs a distinct function [4]. These non-independent, heterogeneous virtual machines (VMs) are used to distribute these service components. Therefore, provide a workload prediction approach based on several VM workload datasets instead of a few VM workload [5].

Predicting variations in workload patterns additionally allows managing task scheduling in a proactive manner. Load balancing of cloud resources is eventually assisted by proactive scheduling through load forecasting. Therefore, in cloud computing architecture, a reliable forecast of future demand is crucial for optimal resource allocation, energy conservation, load balancing, and task scheduling. Numerous methods were offered for time series forecasting analysis to anticipate virtual machine workloads, such as LSTM [6], recurrent neural networks [7], workload pattern-based [8], Restricted Boltzmann Machine [9], and standard Auto Regressive Integrated Moving Average (ARIMA) [10]. The time series CPU usage of cloud service machines was then predicted using a deep learning model, which was then trained on prior information for a specific task to predict the enhanced effective resource allocation. Predicting virtual machine workload, however, does not usually take into consideration the overall workload patterns of virtual machines. Additionally, the VMs' interconnectedness is not considered. Most studies fail to recognize that a cloud computing system is really just a complex network structure made up of multiple virtual machines connected both spatially and temporally.

Provide an effective hybrid DS-RAE model for VM workload prediction technique in order to address these problems. There are two parts to the proposed model architecture for workload prediction. Use a double channel residual self-attention temporal convolution network in the first part to increase prediction accuracy and more accurately represent the sequence's long-term dependencies. Track the quick variations in power usage in the recursive autoencoder as a result. Thus, the proposed workload prediction methodologies can enhance workload forecasting accuracy. The following is presented as the proposed model's contribution:

- A combined DS-RAE with an edge computing framework is developed to forecast workload in a cloud system.
- Cloud workload data is gathered and includes a range of metrics related to the system's operational state, such as bandwidth, disk I/O time, disk space, CPU and memory utilization.
- Enhance data quality at the collected input datasets, pre-processed are done through utilizing missing value edge computing and z-score normalization.
- Dynamization of the sliding window size is used after pre-processing to increase the prediction accuracy of non-random workloads.
- Forecasts future CPU and memory consumption using a hybrid DS-RAE model to improve the proposed model's predictability.

Following is the arrangement of the remaining sections: In Section 2, some more current methods of workload prediction are shown. Section 3 discusses the proposed methodology and its organization. Section 4 presents the results, whereas Section 5's experiments conclude the work.

2. Related Work

In the design of cloud computing, several techniques have been developed that allow it possible to forecast virtual machine workloads with efficiency. The following section reviews several of the current workload balancing scheme approaches.

Xu et al. [11] recommended a modified GRU-based training strategy for the prediction model to attain exceptional accuracy in forecasting for cloud workloads with significant volatility. This work addresses the problem of high-dimensionality by converting high-dimension data into supervised learning time series using sliding windows for multivariate time series. A modified GRU-based training strategy for high variance cloud workloads was offered, utilizing the converted data. Although this strategy works well, it has to be included into a prototype container-based system in order to increase performance.

Zhou et al. [12] invented the Ensemble Forecasting Approach (EFA) to handle workloads in the cloud that are very dynamic. The influence of large variable and unstable workload on forecast accuracy can be lessened by using VMD to divide the workload into several separate elements. Include an additional forecasting module that uses a multi-head attention mechanism to train the local and global internal representations of the sequences using Local RNN and an R-Transformer to extract the nonlinear correlation from the input IMF. Instability is this system's main issue.

Ahamed et al. [13] evaluated models that use adaptive strategies to better allocate resources or train a models further to anticipate numerous correlations of virtual machines in an effort to increase forecast accuracy. Furthermore, maintain that uniformity is necessary for an unbiased analysis and that the quality of the datasets can influence the DL model's outcome. This system's primary shortcomings are its limited selection of datasets and lack of data on the simulation environment.

Maiyza et al. [14] discovered a route within the cloud workload prediction space that accounts for the future orientation of movement within a modern classification hierarchy. Additionally, it introduces VTGAN models, which use 1D CNN as a discriminator and stacked LSTM or GRU as generators. These models are based on GAN networks. The main advantage of VTGAN models is their ability to deal with large-scale nonlinear interactions in cloud workloads in an efficient manner. This method uses a substantial amount of energy and has poor computational performance.

Dogani et al. [15] suggests a method for predicting host workload in cloud computing that combines DWT, BiGRU, and the attention mechanism. In cloud computing, DWT can estimate future host workload by splitting nonstationary and nonlinear data into predictable subbands, and it can also find long-term dependencies in BiGRU. The Alibaba Cluster and Google Cluster Databases were used to assess the recommended method; however, the cost is high.

Alqahtani [16] discovered that a cloud workload trace, a gated recurrent unit with a sparse auto-encoder, and step-wise learning decay scheduling provide a more advantageous compromise between training time and forecast accuracy. Compared to other RNNs-based models with modest and large learning rates as well as complex deep learning algorithms, the GRU-SWSLD model performs better in terms of optimization, convergence, and handling of high-dimensional data of cloud workload trace that have diverse patterns. This model's primary flaw is its intricate design.

Huang et al. [17] introduced DynEformer, a transformer that uses static context awareness and global pooling for determining workload in dynamic MT-ECP. By using an information merging process, the global poo which is based on workload factors that have been identified improves local workload prediction. The real-world MT-ECP use case demonstrates how DynEformer may enhance decision-making and maximize the advantages of application deployment. With this model, long-term prediction is not feasible.

Patel and Bedi [18] discovered that a cloud workload trace, a gated recurrent unit with a sparse auto-encoder, and step-wise learning decay scheduling provide a more advantageous compromise between training time and forecast accuracy. Compared to other RNNs-based models with modest and large learning rates as well as complex deep learning algorithms, the GRU-SWSLD model performs better in terms of optimization, convergence, and handling of high-dimensional data of cloud workload trace that have diverse patterns. Large datasets are not suited for this purpose.

Leka et al. [19] introduced an ensemble meta-learning workload forecasting technique based on particle swarm optimization (PSO) that employs base models and PSO-optimized network input weights. Using a hybrid ensemble learning strategy, the recommended model consists of three recurrent neural networks (RNNs) with a dense neural network layer on top. The strategy's effectiveness is evaluated by examining the CPU consumption of PlanetLab traces and GWA-T-12. But it is not an energy-efficient method.

ZargarAzad and Ashtiani [20] introduced an adaptive auto-scaling system known as ADA-RP that uses workload description and prediction to offer cloud resource allocation. To create a hybrid learning system for decision-making, ADA-RP integrates K-means and CNN models with the CPU use data from prior task executions. In order to prevent resource overuse and underuse, which could result in needless expenses and QoS violations, the framework offers adaptive auto-scaling. This method is not appropriate to be employed for making decisions in non-repeatable workload scenarios.

Numerous systems are developed for optimal workload balancing in cloud environments, according to the literature. According to the articles listed above, workload balancing presents a number of important difficulties. Instability [12], limited selection of datasets [13,18], has poor computational [11,14], elevated expenses [15], need to improve classification performance [17,19,20] and complex designing [16]. Thus, the proposed method relies on hybrid DS-RAE for effective workload balancing in a cloud computing scenario.

3. Proposed Methodology

Technology known as cloud computing came into being to provide minimal maintenance costs and scalable access to computer resources. Cloud customers in this domain take advantage of long-term and on-demand pricing techniques by using virtualized resources. A precise estimate of the cloud workload is essential for effective resource management, enhanced service quality, and prevention of SLA (service-level agreement) violations. While the latter offers a more economical alternative, it necessitates precise projections of future workload requirements, which is a difficult undertaking. Propose a unique hybrid Forecasting Approach (DST-RAE) for substantially enhanced dynamic cloud workload prediction in order to handle this problem. Figure 1 shows the proposed model process flow.

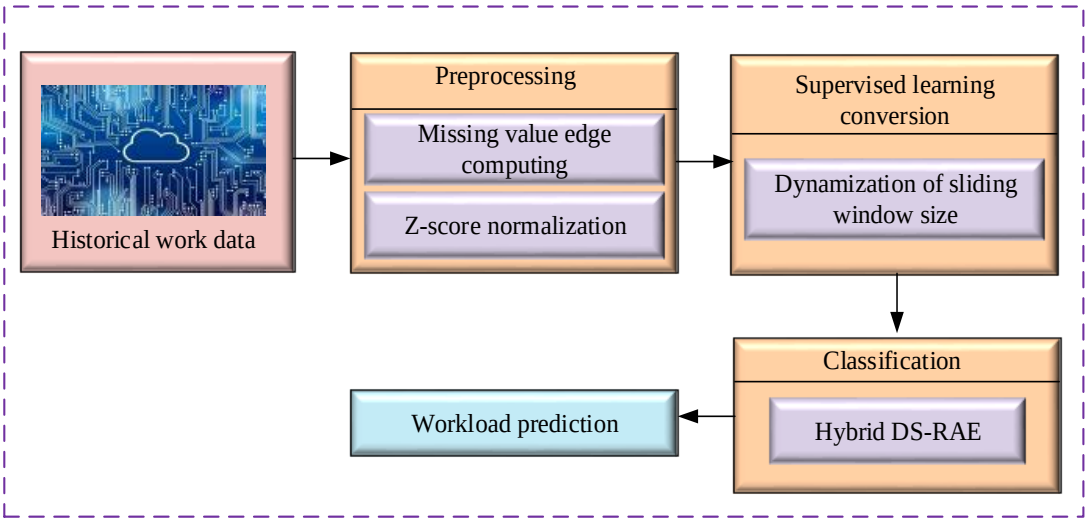


Fig. 1. Architecture of proposed model.

The operational state of the system is measured by a number of metrics obtained from cloud workload data, including bandwidth, CPU and memory utilization, disk I/O time, and disk space. Cloud-based services are primarily restricted by CPU load, which is one of the most significant and finite resources in a computer system. As a result, the industry believes that increasing CPU consumption is essential to enhancing cloud centers resource allocation. When pre-processing first occurs, z-score normalization and missing values with edge computing approach to be normalized original time series data's and delete the missing data to make inference from the database. Afterwards, dynamization of the sliding window size is used to increase the window's size, although this will not significantly affect the prediction's accuracy and ability to forecast CPU usage in the future. This study predicts future memory and CPU use using a hybrid DS-RAE model. Using the DS-RAE model, it is anticipated that the workload in the cloud would require less CPU and memory in the future, allowing for better control over cloud accuracy. When these elements work together, predicting a highly dynamic workload in a cloud scenario becomes more accurate with ease. The below section clearly explains the proposed methodology.

3.1. Data Acquisition

A time series containing the cloud centre's historical CPU utilization is available, $\vec{x} = (x_1, x_2, \ldots x_t)$, which is a series of numbers that have been recorded and are arranged chronologically with regular intervals of time, to extract information from historical data. At time t, the CPU utilizes the record value is x_t.

3.2. Data pre-processing

The raw data that was gathered from the realistic cloud traces is managed in this stage through a workload preparation and data cleaning component. To enhance performance and workload prediction, the workload data can be pre-processed to provide a suitable data structure for the time series prediction. The suggested model employs two distinct pre-processing techniques, namely missing value filled with edge computing and z score normalization to enhance workload prediction. To increase the accuracy of forecast data, start by removing the columns of cloud workload raw data that contain empty data using a disregard these data. Next, use edge computing with missing value filled to improve forecast data accuracy. Following dataset normalization, the z-score Normalization is a dimensionless processing technique that converts a physical system's absolute value into a certain relative value relationship. The dataset needs to be categorized by time using the same timestamp, and the average value of each parameter should be determined. Below is a detailed discussion of the pre-processed technique.

3.2.1. Edge Computing Based Missing Value Filling

The distribution of sensor data can be obtained by using Gated Recurrent Units for Filling (GRUF). This section introduces a specific data-filling algorithm. One generative model capable of developing new samples that match the original dataset's distribution is the Generative Adversarial Network (GAN). A GAN consists of a discriminator (D) and a generator (G). GAN adopts the binary game notion. The two sides/players in this game are the discriminator and the generator. Everyone aspires to maximize their individual gains. While the discriminator searches the input sample for authenticity, the generator aims to transform a low-dimensional randomized vector into a realistic high-dimensional sample.

A randomly developed low-dimensional vector is fed into the generator, which outputs a dynamically formed sample. The generator-generated sample or a true sample from the raw dataset is fed into the discriminator, and its output indicates the probability that the incoming sample is authentic. The discriminator's main goal is to ascertain whether the sample generated by the generator is valid, while the generator's main goal is to create new samples that have the ability to deceive the discriminator. The following is the GAN formula:

$$\min_{G}\max_{D} V(D,G) = E_{x\sim P_{data}}(x)\left[log\left(D(x)\right)\right] + E_{z\sim P_{Z(z)}}(x)\left[log\left(1-D\left(G(z)\right)\right)\right] \quad (1)$$

Wherein z represents the random input vector of the generator, $P_Z(z)$ signifies the distribution of new samples, $P_{data}(x)$ denote the distribution of the original dataset, the generator's output is denoted by $G(z)$, and the discriminator's output probability is represented by $D(x)$.Afterward, the filled data are given to Z score normalization.

3.2.2. Z-score normalization

The initial data series needs to be updated as the actual, highly-dynamic cloud demand leads the CPU consumption figure to fluctuate significantly over time. Furthermore, neural network methods may converge more quickly as a result. The majority of the information's values in cloud workloads fall into a narrow range, so applying Min-Max normalization may cause a large number of smaller numbers to be concentrated in a narrow range, which is detrimental to the forecasting component's training and convergence. For that reason, Z-score normalization is used in this section to normalize the original sequences. The transformed data, with a mean of 0 and a standard deviation of 1, follows the conventional normal distribution. The formula for Z-score normalization is given below:

$$x' = \frac{x-mean(X)}{\sigma} \quad (2)$$

Where, $mean(X)$ is the mean value of X, X is the set of x, and σ is the standard deviation. Afterwards, the sliding window collects the pre-processed data in order to improve prediction performance.

3.3. Dynamization of the sliding window size

The minimal sliding window size when there is a non-random workload because it shows a repeatable pattern. The prediction's accuracy in this case won't be much impacted by the window's increased size. In the scenario with a random workload model, on the other hand, expanding the window size improves prediction accuracy. The data fluctuates significantly in unpredictable surroundings, which is the explanation. As a result, if there is insufficient historical analysis, the prediction measure will not be able to extract the relationships between the data points adequately. As a result, expanding the window's size will allow the inclusion of more historical data points, improving prediction accuracy.

To control the computation time, it was decided in this part to dynamically modify the sliding window's size in accordance with the recent history under examination. Determine a minimum size for the sliding window if the workload is not random, as the latter shows a recognizable pattern. The prediction's accuracy won't be significantly impacted in this case by the window's increased size. On the other hand, in a setting where the workload model is unpredictable, expanding the window size improves prediction accuracy. The explanation is that there are a lot of data fluctuations in random situations. Consequently, the prediction measure will not be able to effectively extract the correlations between the data points if there is not enough historical data to evaluate. As a result, expanding the window's size will enable the inclusion of more historical data points, improving prediction accuracy.

Therefore, as a maximum sliding window size will only be considered when necessary, figuring out the size of the dynamic history will determine the calculation time. Aim for enough historical data in both situations random or not to improve the accuracy of the prediction. During execution cost, a window of minimum size m will be defined for non-random workloads and a window of larger size m' for random workloads. Consider that m and m'values will be obtained through challenges. The workload pattern at a given instant t is included in a sliding window of length SW_{size}, which spans the period $\left[t_{i-SW_{size}+1,\ldots,}t_i\right]$.. $SW = \left(x_{i-SW_{size}+1,\ldots,}x_i\right)$, where SW_{size} indicates the number of samples in the sliding window, is the workload vector to be discovered within the sliding window. From initial to last, the vector's components are organized. And finally, x_i a current sample, shows how many requests were made by end users.

3.4. Hybrid DS-RAE model

Integrating machine learning algorithms and feeding the results of one approach into another, hybrid deep learning models for forecasting generate an effective model for accurate and exact predictions. These combinations are chosen to reduce variation, improve prediction accuracy, and minimize feature noise and biasness while considering the specific requirements and challenges of the intended application. Figure 2 shows a schematic illustration of the hybrid forecasting model. Through pre-processing of the complex and raw data samples, this approach develops a hybrid forecasting model for performance evaluation and implementation for the intended applications.

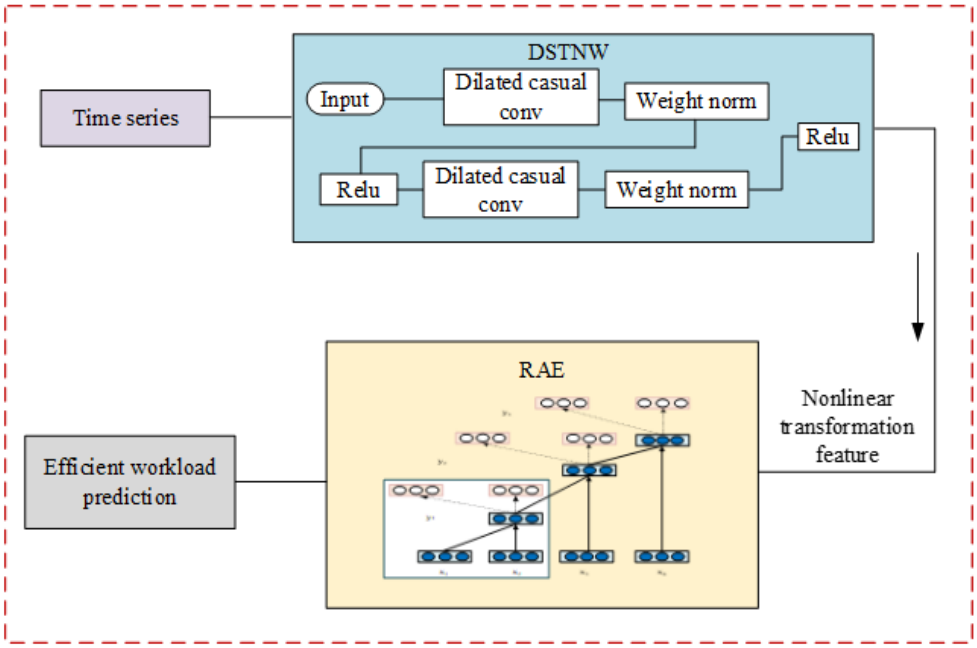


Fig. 2. Architecture of hybrid DS-RAE structure.

The following section discusses the major advances made to workload prediction models based on the hybrid DS-RAE technique.

3.4.1. DSTNW

The Double Attention Temporal Convolution Network (DSTNW) is a useful tool for enhancing prediction precision and effectively capturing sequence long-term dependencies. The DSTNW network's details are provided in this section, as illustrated in Figure 2. The DTN and SM modules that makeup DSTN are covered in the following:

3.4.1.1. DTN unit

The TCN unit is replaced by the DTN unit, which is a single-channel dilated causal convolution unit. The DTN model translates a single-channel dilated causal convolution in TCN into a double-channel dilated causal convolution. The total of the two pathways' outputs makes up the outputs of the two channels. The input that travels through both double sides of two layers of the same dilated causal convolution (DCC) and outputs is one of the routes. Once the first layer's weights are initialized, the input enters the DCC. The Relu activation function is used to apply a nonlinear transformation to the output. To lessen the model's over-fitting, dropout regularization is used for the nonlinear output. An alternative route involves the input passing through a one-dimensional convolutional layer and directly reaching the output. The residual block (RC), which comes from the residual neural network, comprises the two routes. It is beneficial for building a deep neural network. According to causal convolution, DCC increases the value of the expansion coefficient d to widen the network's receptive field to accommodate longer historical data. It is a three-layer causal convolutional network with a receptive field of three, an expansion coefficient of one, and a convolution kernel k of two.

The predicted load sequence $\hat{y}_t$ is calculated from the input sequence $[x_{t-2}, x_{t-1}, x_t]$ and has nothing to do with the input sequence $[x_{t+1}, x_{t+2}, \dots]$. Information leaking would not occur from the usage of causal convolution in TCN. The network receptive field is expanded by raising the expansion coefficient in DCC since the receptive field for the causal convolution in TCN is limited. Under the same number of layers, the receptive field of DCC is increased to 4. After implementing the suggested DCC, longer history sequence data can be obtained using the TCN. The action of dilated convolution is displayed as:

$$F(t) = \sum_{v=0}^{u-1} f(v) X_{t-dv} \tag{3}$$

Where X_{t-dv} is the sequence data, $f(v)$ and $F(t)$ represents the dilated convolution operation is the filter function, v is the value of the vth element in the input sequence data, d stands for the expansion coefficient, and u for the length of the input sequence data. This strategy can achieve significant differences and improve expressive capacity because of its effective expansion of the receptive field. It is possible to better capture some long-term relationships in the sequences by dividing the dilated causal convolution into two parallel sides.

3.4.1.2. SM unit

The self-attention mechanism is a fundamental component of neural networks and represents a significant advancement over the classical attention process. It can assist the model in concentrating more on informative data that significantly influences the outcome by attempting to capture the internal correlations of the data. The temporal dimension's properties are captured in the time series by means of the self-attention mechanism. By putting varying weights on each temporal element in the time series, the sequence receives distinct contributions to the temporal dimension.

The essential part of the self-attention process is a self-attention layer. It is made up of three components: values, keys, and queries. The input data is multiplied by matching weight matrices to obtain these three vectors $X_w, X_y \; and \; X_z$ respectively:

$$W = inputs \times X_w \tag{4}$$

$$Y = inputs \times X_y \tag{5}$$

$$Z = inputs \times X_z \tag{6}$$

The result of multiplying W and Y by a ratio factor $\sqrt{d_y}$ is divided by a softmax function and then multiplied by Z to get the output of the self-attention mechanism:

$$output\ 1 = softmax\left(\frac{WY^T}{\sqrt{d_y}}\right)V \tag{7}$$

A residual link is utilized at the conclusion of the temporal self-attention process to prevent loss or distortion of information throughout the hierarchical transmission of information within the network, hence addressing the issue of gradient disappearance and explosion in deep neural networks:

$$output\ 2 = output\ 1 + input \tag{8}$$

A residual temporal self-attention mechanism module is presented to obtain more significant temporal information in extended sequences and better capture the relationship between features and load sequences. The purpose of this module is to record the contributions made by the various sequence elements. The accuracy and depth of the model may be more easily improved by the network by connecting the residual mechanism and self-attention mechanism. In addition, by deepening the network, cross-layer connections in residual networks can boost efficiency without causing vanishing or bursting gradients.

3.4.1.3. Recursive autoencoder

Recursive auto encoders are used for short-term fine-grained forecasting, which may track the rapid variations in power usage within a data centre. An effective unsupervised or semi-supervised technique for embedding a phrase or sentence in continuous space is the recursive auto-encoder. A four-word phrase can be learned as a vector representation in RAE architecture by iteratively joining two child vectors in bottom-up fashion. The four words, w_1, w_2, w_3 and w_4, are initially projected into real valued vectors, x_1, x_2, x_3 and x_4, according to the convention. Every node in RAE reuses a typical auto-encoder. The parents vector y_1 is computed by the auto-encoder for two children, $c_1 = x_1$ and $c_2 = x_{2,}$.

$$y_1 = f\left(W^{(1)}[c_1 : c_2] + b^{(1)}\right) \tag{9}$$

To assess effectively the parent's vector resembles its children, the standard auto-encoder recreates the children in a reconstructed layer.

$$[c_1' : c_2'] = f'\left(W^{(2)}\, y_1 + b^{(2)}\right) \tag{10}$$

Typically, an auto-encoder aims to minimize the reconstruction errors among the inputs and the reconstructions during training:

$$E_{rec}([c_1; c_2]) = {}^1/_2 \, \|[c_1; c_2] - [c_1'; c_2']\|^2 \tag{11}$$

In this way the workload of the cloud is computed to reduce the power consumption.

Table 1. Overview for the proposed model in the component

No.	Layer Name	Purpose	Operations/Details	Output Shape
1	Input Layer	Accepts raw input data	Input sequence (e.g., time series, joint data, features)	[B, C, T] or [B, C, T, N]
2	Temporal Convolution Block 1	Capture short-term temporal features	1D Dilated Conv → BatchNorm → ReLU → Dropout (optional)	[B, C1, T, N]
3	Temporal Convolution Block 2	Capture long-range dependencies	Deeper 1D Conv with larger dilation → Residual Connection	[B, C2, T, N]
4	Temporal Attention Module	Emphasize important time steps	Self-Attention across time → Weighting → Output fusion	[B, C2, T, N]
5	Channel/Spatial Attention Module	Highlight key channels/features or spatial nodes	Squeeze-and-Excitation (SE) or Channel-wise/Node-wise Attention	[B, C2, T, N]
6	Feature Fusion Layer	Combine attention and temporal features	Element-wise Add or Concatenate + 1×1 Conv + ReLU	[B, C3, T, N]
7	Global Pooling Layer	Compress temporal/spatial features	Global Average Pooling over T (and N if present)	[B, C3]
8	Fully Connected (FC) Layer	Final output prediction	Dense Layer(s) → Softmax (classification) or Linear (regression)	[B, num_classes] or [B, 1]

Algorithm 1: Pseudo code for proposed workload prediction approach

Input: Raw Data
Output: Work load prediction

Start
Initialize the VM's CPU, memory, and bandwidth parameters.
$F = \{f_1, f_2, \ldots, f_n\}$ *// raw features collected*
{
 # Pre-processing the input data F
A=missing value with edge computing (F) // filling the missing value data
B=Z score normalization(A) // Normalize the data
}
{
*# S*upervised learning time series //convert pre-processing into time series
H= Dynamization of sliding window
}
{
 # Classification
O = Hybrid DS-RAE (H) // detect the output using eqn. (3)-(11)
}
}
Output: Predict workload types
End

4. Result and Discussion

Techniques for workload prediction have been put forth in an effort to enhance capacity planning, guarantee effective resource management, and, ultimately, uphold service level agreements (SLAs) with end users. In this situation, suggest a fresh method for estimating the volume of requests that will reach a SaaS service so that the virtualized resources required to fulfill user requests can be ready. The technique will be used to accomplish two goals at once: get accurate forecast results and response time. Dynamizing the sliding window size is the approach that was selected to control the calculation time for analyzing the recent history, as the computational complexity increases with the size of the input in the prediction model. After that, a prediction will be made based on resources were used on various data points in the past. Additionally, there are more applications for the suggested strategy in prediction. Tests were conducted using actual workload traces to estimate the efficiency of the suggested approach in comparison to cutting-edge techniques. The suggested approach provides a balance between prediction accuracy and execution time. Python software, an NVIDIA GeForce RTX 3070 GPU, an Intel Core i7 CPU, and a 64GB RAM combination are used

to analyze the performance of the suggested model.

The workload prediction unit gathers data on storage space, bandwidth, CPU and memory utilization, etc. primarily preprocessing is done using an edge computing approach to normalize the original time series collected data and remove missing values in order to obtain inferences from the database. This is done by utilizing a z-score normalization. The sliding window then gets wider by dynamizing its size; however, this has no appreciable impact on the prediction's accuracy or capacity to estimate CPU utilization in the future. This study uses a hybrid DS-RAE model to forecast CPU and memory utilization in the future. With the DS-RAE model, it predicts that the cloud workload's future CPU and memory utilization would increase, allowing for better cloud accuracy direction.

Table 2. Simulation Parameter for the Suggested Algorithm

RAE	Learning Rate	0.001
	Epoch	100
	Loss Function	MSE
	Optimization Method	Adam
DRSA	Dropout	0.2
	Activation	Gelu
	Norm_type	Layer norm
	Num_Heads	12

4.1. Dataset description

The GitHub site was used to gather the dataset [21]. A collection of Planet Lab virtual machine CPU usage logs gathered on ten days chosen at random in March and April 2011. Determine the CPU utilization % and categorize it into ten groups. The CPU use percentages of >90%, 80-90%, 70-80%,..., 0-10% reflect Classes 0, 1, 2,..., 9.

Dataset 2 contains the cloud computing performances metrics dataset by Abdur Raziq Khan on the kaggle provided the simulated data aimed to analysis and improving the energy efficiency of cloud computing system. It includes CPU,memory usage, network traffic etc. The data seems to reflect the operational behaviour of virtual machine in the cloud environment.

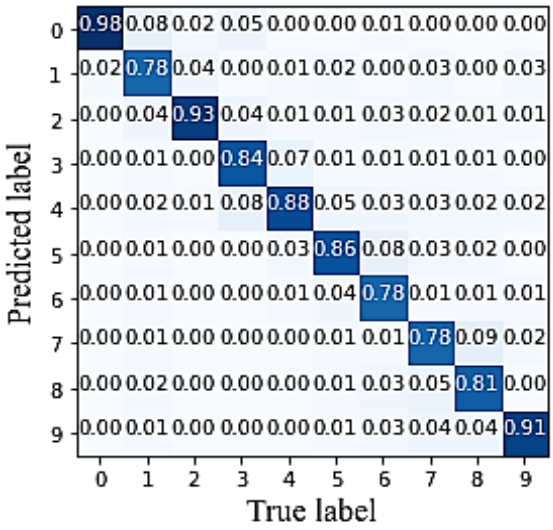


Fig. 3. Confusion matrix of proposed method.

The suggested model confusion matrix model is shown in Figure 3. The process of defining a classification algorithm's performance involves creating a confusion matrix. A confusion matrix is generated during the predictive analysis process and contains both positive and negative rates. Figure 3 makes it clear that 0.98,0.78,0.93,0.84,0.88,0.86,0.78,0.78,0.81 and 0.91 are correctly predicted in 0, 1,2,3,4,5,6,7,8and 9 class respectively. The accuracy, sensitivity, specificity, and null error rates are also measured using the positive and negative rates. The performance measures are compared with those of other methods, such as GA-DBN, RNN-LSTM models, and ARIMA-ANN.

As observed in Figure 4, the suggested model DS-RAE is contrasted with the techniques currently in use. The proposed DS-RAE model has 0.0034MSE, GA-DBN has 0.25MSE, RNN-LSTM has 0.045MSE, ARIMA-ANN has 0.105MSE. Consequently, demonstrates that the suggested method had a lower error value as compared to conventional methods. The model gather linear and non-linear trends making the use of resources data in the MSE. Compared to existing system inaccurate the output having the issues. But this model accurate learning in the workload behaviour.

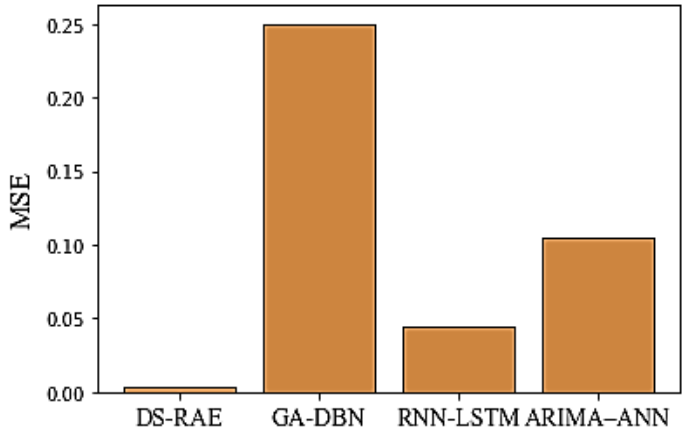


Fig. 4. MSE comparison between the suggested and current models.

4.2. Comparison Analysis

The proposed effective DS-RAE for appropriate workload balancing in VM without any delay and failure is compared with some existing techniques like the GA-DBN approach, RNN-LSTM and ARIMA-ANN. Performance metrics are used to compare suggested and current techniques.

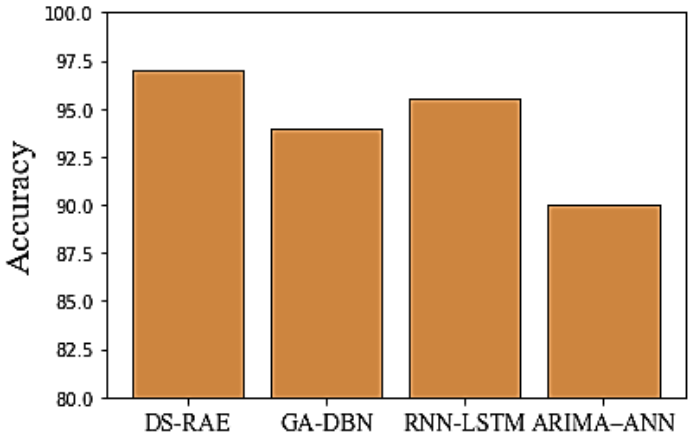


Fig. 5. Accuracy comparison between the suggested and current models.

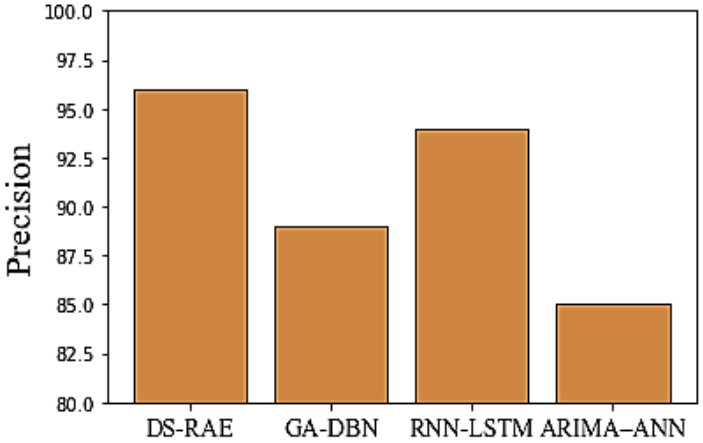


Fig. 6. Comparison of the suggested and current models precision.

An accurate system is one that forecasts a value with a minimal margin of error. Figure 5 illustrates the accuracy rate of the suggested approach, which is 97%, in comparison to GA-DBN at 94%, RNN-LSTM at 95%, and ARIMA-ANN at 90%. Greater accuracy shows better performances to predict the work load. While the other model don't capture the resources usage patterns then it shows in low accuracy, handling more complex.

Figure 6 compares the precision of the desired and current techniques. The quantity of anticipated favourable events is what is meant by accurate measurement. In comparison to other current approaches, such as GA-DBN, RNN-LSTM and ARIMA-ANN with corresponding precision values of 88% and 93%, 85% the suggested method's precision value was discovered to be 96.2%. The proposed system adaptability to rapidly change the cloud environment to enhances the precision, balance between accuracy and execution time so it offered the high precision without significant cost.

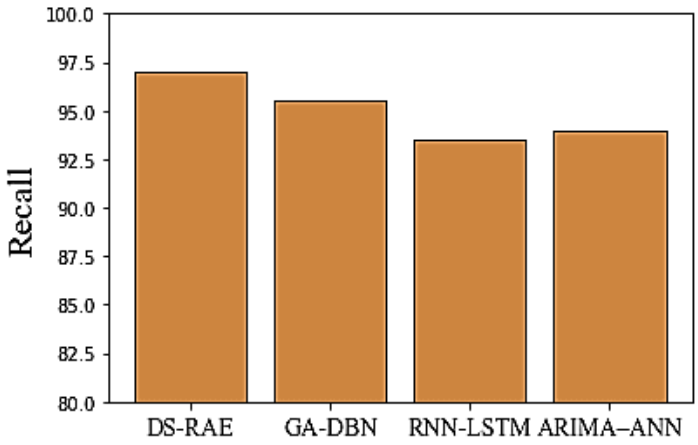


Fig. 7. Recall comparison between the suggested and current models.

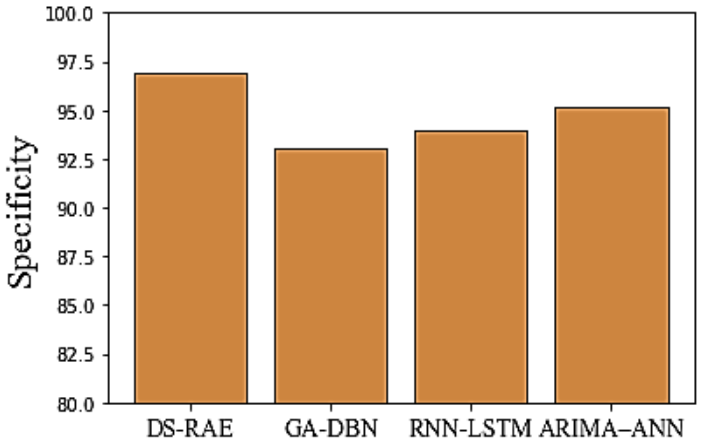


Fig. 8. Specificity comparison between the suggested and current models.

Recall is designed by separating the total number of components that correctly fall into the positive category by the total number of true positives, as shown in Figure 7. In comparison to other existing approaches, such as GA-DBN, RNN-LSTM and ARIMA-ANN with corresponding recall values of 95%, 93% and 94%, the proposed method's recall value was discovered to be 97%. The algorithm minimize the false negative and maximizes recall leading to superior performances compared to the other approaches. It can ability to adapt to dynamic and changing the workloads to effectively detect and predict the resources usage.

Figure 8 compares the suggested and current approaches levels of specificity. The degree to which a model can predict the real negatives of every imaginable sort is known as specificity. In comparison to various current approaches, such as GA-DBN, RNN-LSTM and ARIMA-ANN with corresponding specificity values of 93%, 94% and 90.3%, the proposed method's specificity value was determined to be 96%. The present model correctly identify the true negative instances cases with no significant resources uuage occur and carefully designed to avoid focusing too heavily on positive events which helps it generalize more effectively. As a result it can better identify periods of low activity or non-critical states. So, the model accurately forecast cloud workloads with greater specificity.

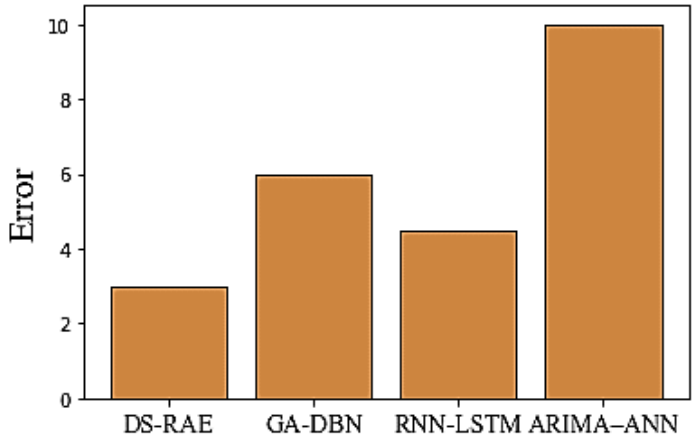


Fig. 9. Error comparison between the suggested and current models.

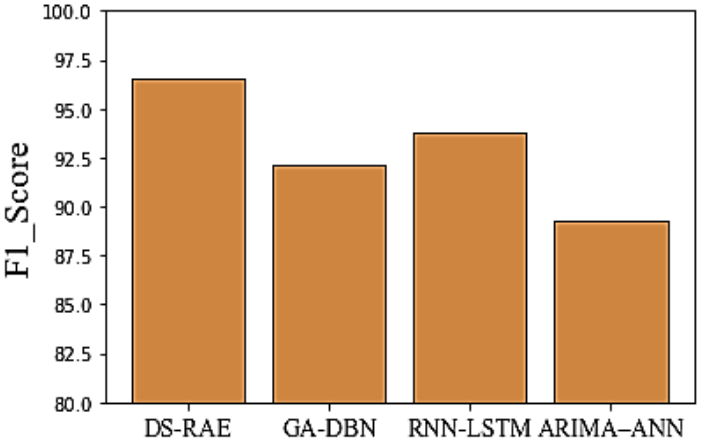


Fig. 10. F1-Score comparison between the suggested and current models.

The error difference between the suggested and present approaches is illustrated in Figure 9. The proposed method's error rate is 3%, GA-DBN is 6%, RNN-LSTM is 5%, and ARIMA-ANN is 10%. It demonstrates that the suggested model has lower error rates than the conventional models. Proposed model generalise across dynamic and complex workload patterns to further minimize both false positive and false negatives leading to a lower overall error rate compared to the existing system.

After that, the F1_Scores of the suggested and used methods are examined. Figure 10 shows a comparison of the current and expected F1 scores. The suggested method has an F1 Score of 96%, GA-DBN of 92%, RNN-LSTM of 93%, and ARIMA-ANN of 89%. F1-Score which is harmonic mean of precision and recall reflects the models ability to maintain a balance between correctly identify the positive instances and minimize the false positives.

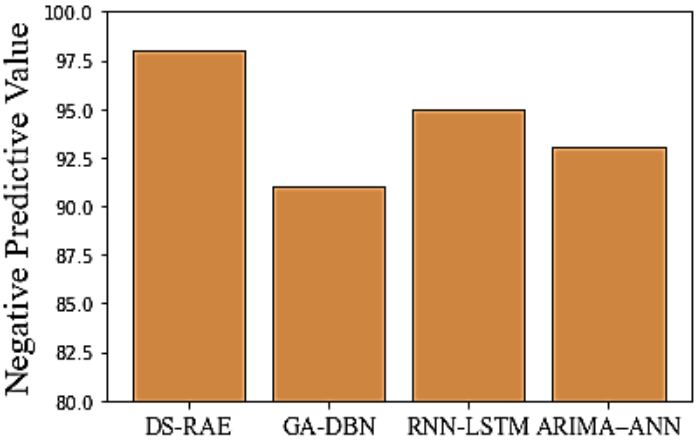


Fig. 11. NPV comparison between the suggested and current models.

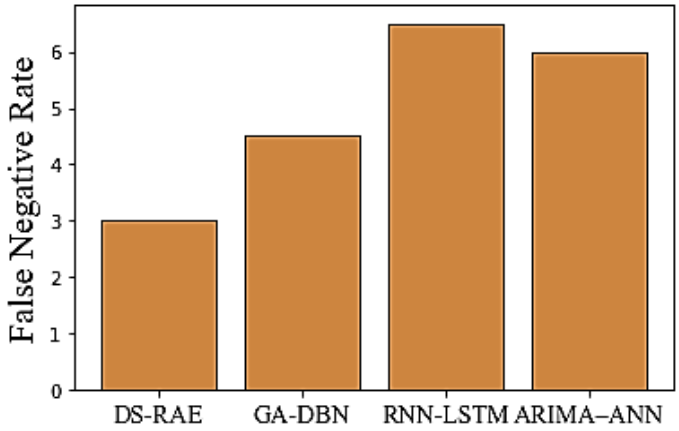


Fig. 12. FNR comparison between the suggested and current models.

The proposed method's NPV is 98%, GA-DBN is 91%, RNN-LSTM is 95% and ARIMA-ANN is 93% as shown in Figure 11. After that, the NPV of the proposed and existing methods is examined. The existing system often struggles with the strike an effective balances between model complexity and training efficiency. As a result the DS-RAE not only achieved the higher NPV but also provided the more accurate and balanced model.

The false negative rate for both planned and current approaches is displayed in Figure 12. False negative rates for the suggested approach are 3%, GA-DBN is 5%, RNN-LSTM is 7%, and ARIMA-ANN is 6%. It may struggles in capturing the long term dependencies, nonlinear work load patterns in the current model. So the result in the high false negative rate , reducing the effectiveness in the dynamic environment. The present model demonstrated a significantly lower false negative rate underscoring its reliability and precision in proactive in the workload and efficient resources management.

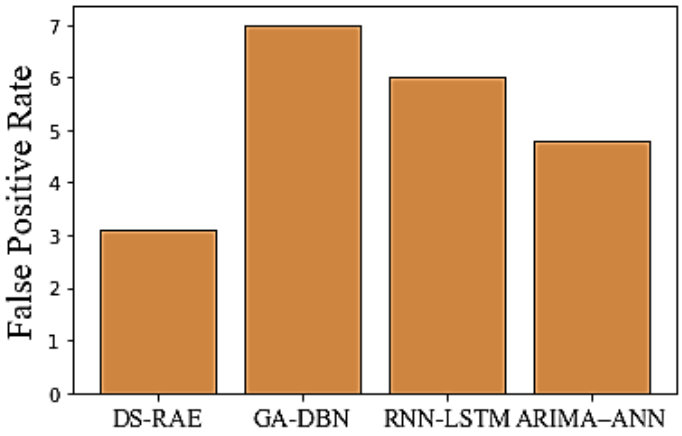


Fig. 13. FPR comparison between the suggested and current models.

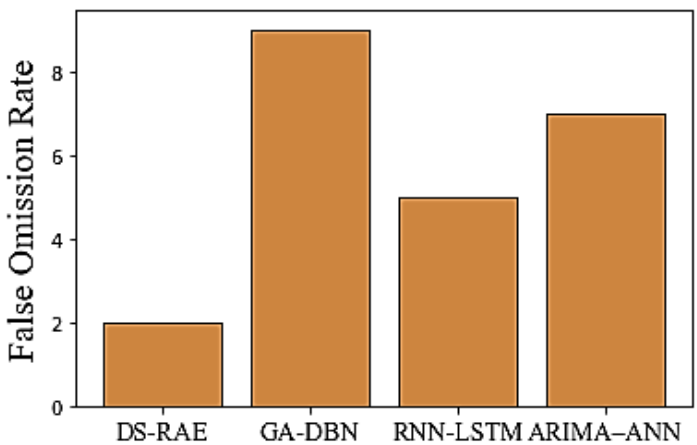


Fig. 14. Comparison of FOR in proposed and existing model.

The false positive rate for both planned and traditional techniques is displayed in Figure 13. The proposed method's false positive rate is 4%, GA-DBN is 7%, RNN-LSTM is 6% and ARIMA-ANN is 5%. DS-RAE effectiveness in accurately differentiating between normal and high demand helping to avoid unnecessary resources allocation. Traditional model do not offer the same combination of dual modelling which often result in high positive rates. In contrast DS-RAE model achieved the low false positive rate reflecting the strong performances .

The false omission rate difference gives the amount of false negative transactions as a percentage of all transactions with a negative outcome in figure 14. The proposed method's false omission rate is 2%, GA-DBN is 9%, RNN-LSTM is 5% and ARIMA-ANN is 7%. It describes the pervasiveness of the false omission rate among all negative transactions. The proportion of false negatives the model incorrectly predicts a low demand scenario despite actual high demand among all predicted outcomes.A lower FOR indicates the strong ability to avoid underestimating resources requirement, thereby reducing the risk of service degradation. In this proposed model represents the low false omission rates so it capture the workload dynamics.But the current model achieved the high values.

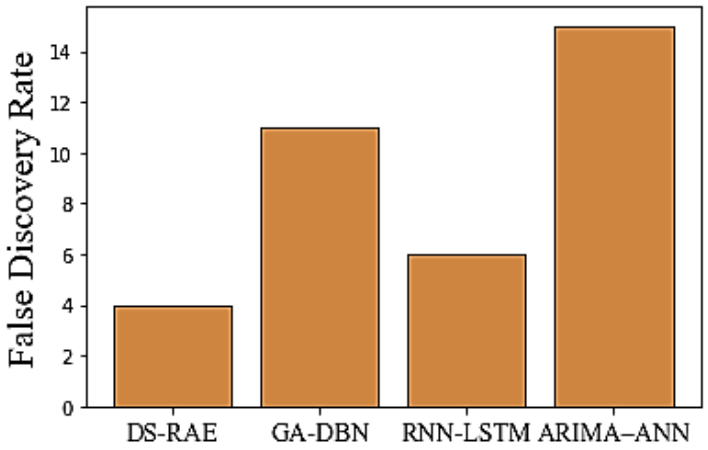


Fig. 15. Comparison of FDR for proposed and existing model.

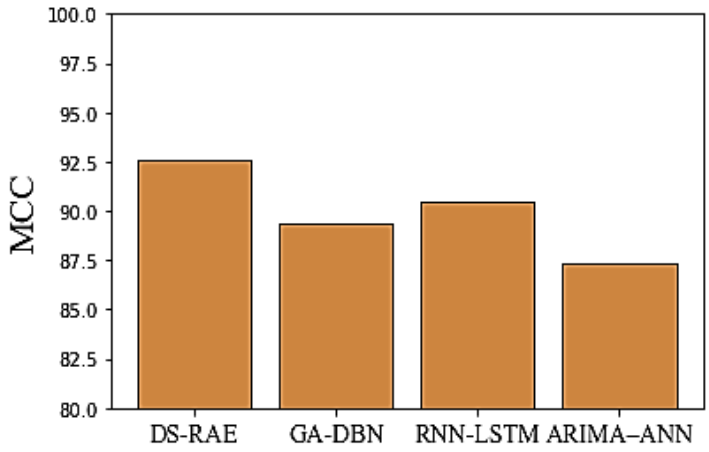


Fig. 16. Comparison of Matthews's correlation coefficient.

Figure 15 demonstrates false discovery rate (FDR) which controls to find as many significant features as feasible with a minimal number of false positives. The proposed method's FDR value is 4%, GA-DBN is 11%, RNN-LSTM is 6% and ARIMA-ANN is 5%. FDR among the predictive model highlight the proposed model minimize the false positive during cloud overload while the other model having the issues is greater tendency to misclassify normal workload patterns, notable number of false positives ,possibly due to overfitting short term trends. So the proposed model enhanced the prediction.

Figure 16 compares the MCC of the suggested and existing approaches. MCC in the suggested method has a value of 92.6%, GA-DBN of 89.32%, RNN-LSTM of 90.51%and ARIMA-ANN of 87.32%.The high score for the proposed model signifies the strong ability to identify the both and low demand situation while maintaining the low error rate, reliability and predictive balances. The other model capturing the low dependencies ,may lead to convergences issues.

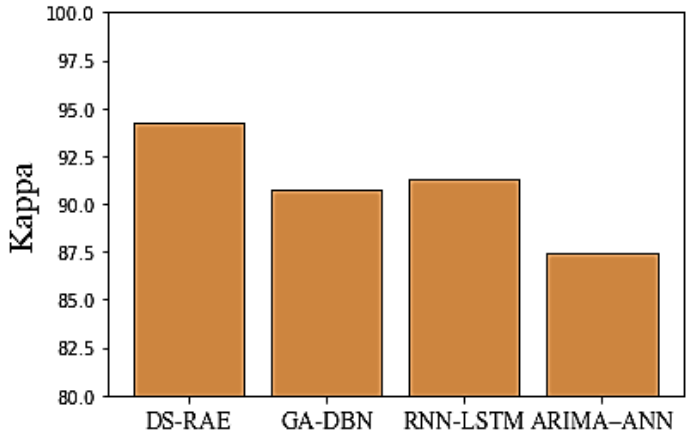


Fig. 17. Comparison of kappa for proposed and existing model.

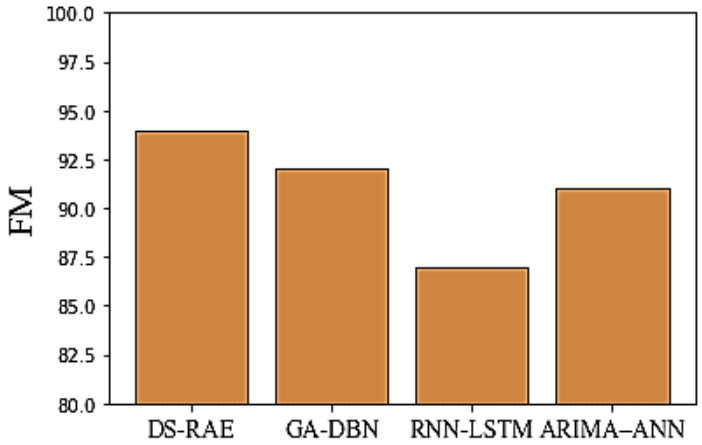


Fig. 18. Comparison of FM for proposed and existing model.

The kappa comparison between the suggested and current approaches is shown in Figure 17. A statistical indicator of the consistency of different variables across rates is called kappa. The proposed method's kappa value is 94.2%, GA-DBN is 90.7%, RNN-LSTM is 91.3% and ARIMA-ANN is 87.4%. It is ability to produce the highly consistent and dependable result across diverse and dynamic work load patterns but also strong additionally the existing model consistency in due its limited capacity for model complex,

The Fowlkes-Mallows (FM) Index is a statistical metric used to evaluate the similarity between two clusterings or partitions of data. The proposed method's FM value is 94%, GA-DBN is 92%, RNN-LSTM is 87% and ARIMA-ANN is 91% as shown in Figure 18.The FM are particularly significant in evaluating the balance between precision and recall in the classification and a high FM values .This suggest that the DS-RAE is not only capable for identify meaningful grouping in the dynamic and nonlinear data but also minimize the incorrect classification .The comparative model FM scores is low it may struggle with fail to capture the nonlinear variation in complex workload.

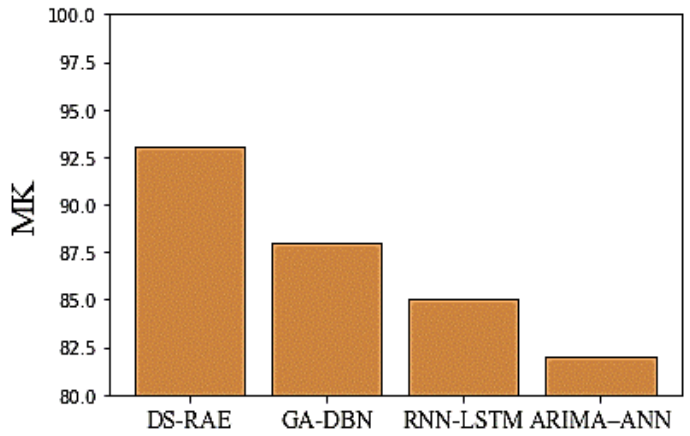


Fig. 19. Comparison of MK for proposed and existing model.

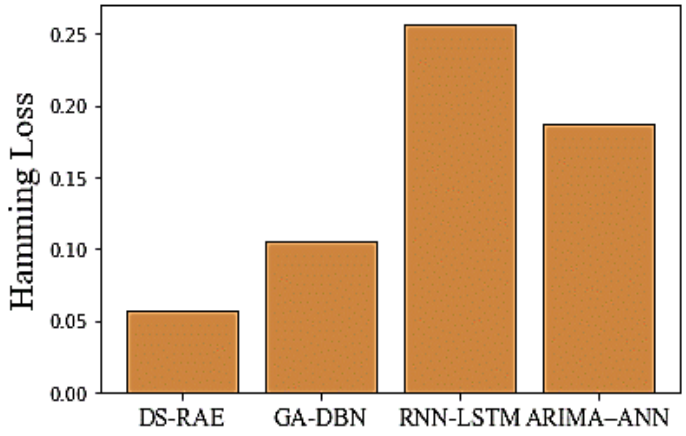


Fig. 20. Comparison of hamming loss for proposed and existing model.

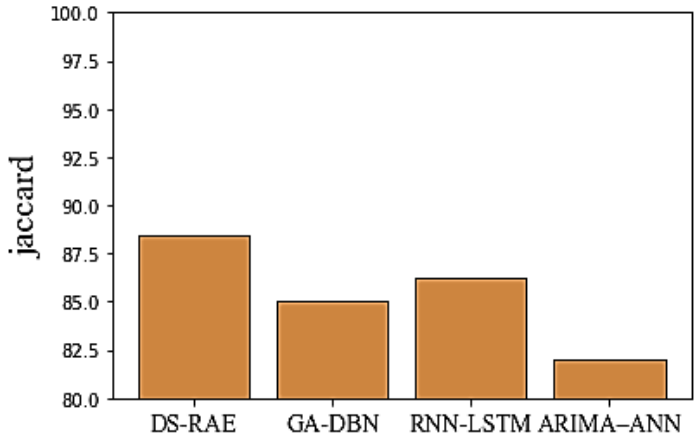


Fig. 21. Jaccard comparison between the provided and current ones.

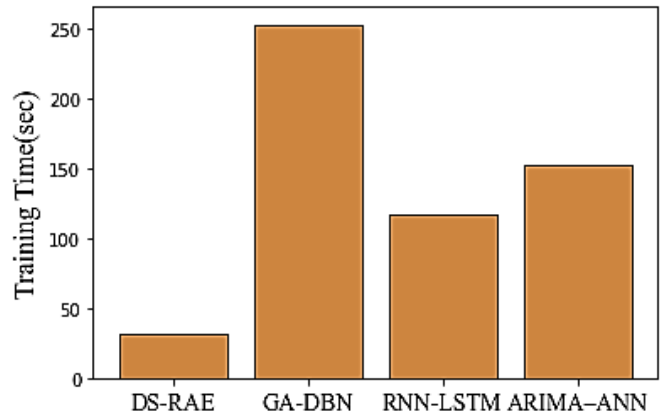
Markedness(MK) is used in the context of confusion matrices, to measure the quality of a binary classification system's performance. The proposed method's MK value is 93%, GA-DBN is 88%, RNN-LSTM is 85% and ARIMA-ANN is 82% as shown in Figure 19. A high values indicates that the model not only identify true positive and true negative prediction effectively but also minimize the false alarms thereby reducing both over and under provisioning in cloud resources management while the other model is less reliable prediction.

The percentage of incorrect labels to all labels is known as the hamming loss. Figure 20 compares the hamming losses of the suggested and existing approaches. The proposed method's hamming loss value is 0.057%, GA-DBN is 0.105%, RNN-LSTM is 0.257% and ARIMA-ANN is 0.187%. The low rate is indicates the DS-RAE is highly effective at identifies correctly in resources demand classes with the minimal misclassification and reflects the high precsion . In constrast the current method is lack in nonlinear multi variate data.

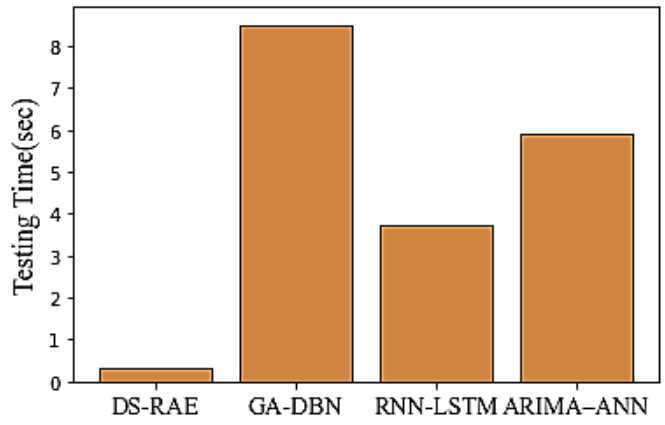
Figure 21 shows a Jaccard comparison of the suggested and existing approaches. The proposed method's jaccard index value is 88%, GA-DBN is 85%, RNN-LSTM is 86% and ARIMA-ANN is 82%. The DS-RAE provided the more accurate and consistent prediction of overlapping cloud workload patterns. It may struggle with overfitting limited sequences memory , weak adaptability to complex workload behaviour in the existing system.

4.3. Computational Timing Analysis in Workload Prediction

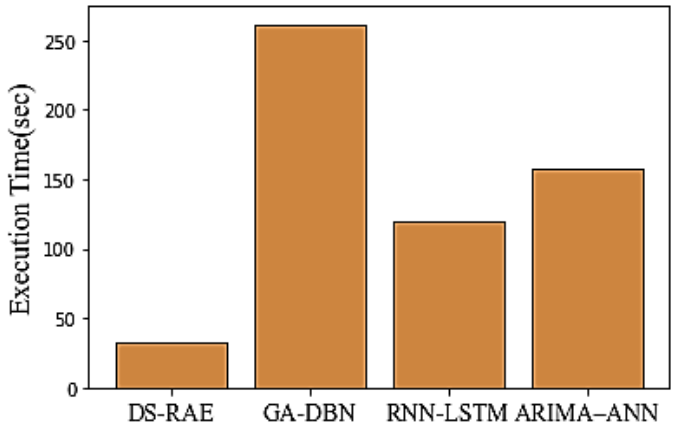
The amount of time required to finish a calculation is known as the computation time. Testing and training timing are included in the analysis of computational timing. Training time is the duration of time an algorithm requires to use training data in order to train a model. Testing is done to determine if a trained model functions well or not. Throughout the testing phase, a classifier provides predictions using a trained model. A thorough explanation of training and test times can be found in the section below.



(a)



(b)



(c)

Fig. 22. Computational timing of (a) Training time, (b) testing time, (c) execution time.

The proposed and existing approaches for training and testing time comparison is shown in Figure 22(a) and 22(b). The proposed method's training time is 31.67sec, GA-DBN training time is 252.91sec, RNN-LSTM training time is 116.41sec and ARIMA-ANN training time is 152.14sec. The proposed method's testing time is 33sec, GA-DBN testing time is 8.51sec, RNN-LSTM testing time is 3.73sec and ARIMA-ANN testing time is 5.913sec. It may minimize the parameter explosion and leading to quicker convergences, and it is more reliable accurate workload predict the load in the testing time

The execution time of proposed and existing models are compared to validate the process in Figure 22(c). The proposed DS-RAE model provides a solution at 32sec, but the traditional GA-DBN model provides a solution at 261sec, RNN-LSTM model consumes 120sec to offer the solution, and ARIMA-ANN model takes 158sec to offer a solution. The comparison provides the proposed model provides a fast processing time as compared to other models as well as improves the prediction than the traditional approaches. While the model is accurate and lightweight for cloud workload.

4.4. Evaluation metrics comparison

Proposed model metrics such as accuracy compared with some other existing methods is clearly shown in Table 1. The traditional models are considered as self-adaptive differential evolution algorithm (SADE), Auto Adaptive Differential Evolution (AADE) algorithm and Back-propagation Neural Network (BP).

Table 3. Ablation study for the suggested algorithm

Model	Accuracy	Precision	Recall	F1-Score	Specificity
Missing values edge computing +Hybrid DS-RAF	86%	85%	75%	76%	83%
Z-Score Normalisation+Hybrid DS-RAE	83%	79%	81%	75%	85%
Missing values edge computing + Z-Score Normalisation+Hybrid DS-RAE	95%	90%	89%	85%	84%
Missing values edge computing+ Z-Score Normalisation+Dynamization of slinding windoiw size+Hybrid DS-RAE	97%	96.2%	97%	96%	96%

Table 3 represents the Ablation for the suggested model. The column denotes the accuracy, precision, recall, and F1-score, respectively. The values for Missing values edge computing+ Z-Score Normalisation+Dynamization of sliding window size+Hybrid DS-RAE the accuracy is 97%, precision is 96.2%, recall is 97%, f1score is 96%, specificity is 96%.

Table 4 Comparison for the DS-RAE and Current Algorithm in the Dataset 2

Models	Accuracy	Jaccard	Hamming Loss	Kappa
DS-RAE	97.63%	84%	0.055	92.1
GA-DBN	90%	89%	0.040	88%
RNN-LSTM	95%	93%	0.054	90%
ARIMA-ANN	89%	91%	0.051	89%

Table 4 represents the comparison for proposed and existing model in the dataset 2. The performances metrices such as Accuracy,jaccard,hamming loss,kappa. The DS-RAE the values attained in the accuracy is 97.63%,jaccard is 84%,hamming loss is 0.055 and the kappa is 92.1%.

Table 5. Analysis of proposed model for the different cloud environmnet

Cloud	VM Type	WorkLoad Type	Latency(ms)	Throughput(req/s)	Cost($/hr)	Boost Time(s)
AWS	EC2 c5.xlarge	Compute	95	512	0.20	34
Azure	D4 v3	IO-intensive	114	472	0.22	48
GCP	N2-standard 4	Memory Intensive	92	522	0.18	30
IBM Cloud	Bx2-4x16	General purpose	132	403	0.33	53
Oracle Cloud	VM StandardE4.Flex	DB	124	432	0.16	44

Table 5 demonstrate the different cloud environment. The parameter is VM type workload type, latency, throughput, cost, boost time. The AWS the VM type is EC2 c5.xlarge, workload types types is compute,latency is 95,throughput is 512,cost is 0.20,boost time is 34.

5. Conclusion

Proposed approach concentrates on developing a hybrid DS-RAE model that can accurately forecast workload in specific circumstances. Precise forecasting of forthcoming demand is essential for optimizing performance quality and reducing resource consumption in cloud systems. Based on the hybrid DS-RAE method, this model presents a multi-step ahead memory and CPU load prediction model. Experiments have demonstrated that this idea works effectively in a dynamic cloud computing setting. The service provider will estimate demand and use the forecasted outcomes to carry out the required preparations. If a workload in a dynamic cloud environment entirely shifts to a different pattern, there could be serious forecast mistakes. To attain high forecast accuracy in that case, a novel predictive model needs to be developed. The workload prediction unit collects the raw data after it has been gathered from the system's operational state, CPU and memory utilization etc. Identify patterns and oscillations in the workload trace by pre-processing the data to increase the forecasting accuracy of this model. The missing value with edge computing and z score normalization, data pre-processing is used to determine the important features from raw data samples, remove unrelevent element and normalize data. The first step is to gather the raw data from the cloud workload and use missing values to exclude any irrelevant elements. Used the z-score scaling technique to achieve data normalization and

transform the supplied resource consumption statistics into a standard scale. The multivariate data should then be transformed into a supervised learning time series using a dynamic sliding window so that deep learning can be managed. Next, use efficient hybrid DS-RAE to accurately anticipate workload. Workload forecasting performance is improved by the proposed technique when compared to commonly used workload prediction models, as demonstrated by results from experiments on real-world datasets. The suggested DS-RAE model is effective at predicting virtual machine workloads. The results show that standard deep learning is not as effective as GS-RAE models in classifying and predicting cloud workloads. 92% MCC, 96% specificity, 94% kappa, and 97% accuracy are all obtained by the suggested method. The comparative analysis demonstrates that the suggested model provides an efficient result than the present approaches. Future work concentrates on the optimal dynamization of sliding window size to improve accuracy.

Acknowledgement

Author contributions: The corresponding author claims the major contribution of the paper including formulation, analysis and editing. The co-authors provide guidance to verify the analysis result and manuscript editing.

Compliance with ethical standards: This article is a completely original work of its authors; it has not been published before and will not be sent to other publications until the journal's editorial board decides not to accept it for publication.

References

[1] A. Belgacem, S. Mahmoudi, and M. A. Ferrag, "A machine learning model for improving virtual machine migration in cloud computing," *The Journal of Supercomputing*, pp. 1-23, 2023.

[2] K. M. Hassan, F. E. Z. A. El-Gamal, and M. Elmogy, "Intelligent Mechanism for Virtual Machine Migration in Cloud Computing," In *World Conference on Internet of Things: Applications & Future* (pp. 67-83), 2023. Singapore: Springer Nature Singapore.

[3] S. Karthick, "Semi Supervised Hierarchy Forest Clustering and KNN Based Metric Learning Technique for Machine Learning System," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 9, pp. 2679-2690, 2017.

[4] J. A. Barron-Lugo, J. L., Gonzalez-Compean, I. Lopez-Arevalo, J. Carretero, and J. L. Martinez-Rodriguez, "Xel: A cloud-agnostic data platform for the design-driven building of high-availability data science services," *Future Generation Computer Systems*, vol. 145, pp. 87-103, 2023

[5] D. Huang, L. Costero, A. Pahlevan, M. Zapater, and D. Atienza, "CloudProphet: A Machine Learning-Based Performance Prediction for Public Clouds," *arXiv preprint arXiv:2309.16333* (2023)

[6] K. L. Devi, and S. Valli, "Time series-based workload prediction using the statistical hybrid model for the cloud environment," *Computing*, vol. 105(2), pp. 353-374, 2023

[7] B. M. Hardas, S. Kaushik, A. Goyal, Y. Dongare, M. G. Aush, and S. Chiwhane, "Deep Learning with Multi-Headed Attention for Forecasting Residential Energy Consumption," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12(2s), pp. 109-119, 2024

[8] S. Khurana, G. Sharma, and B. Sharma, "A fine tune hyper parameter Gradient Boosting model for CPU utilization prediction in cloud," 2023.

[9] D. Saxena, J. Kumar, A. K. Singh, and S. Schmid, "Performance analysis of machine learning centered workload prediction models for cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 34(4), pp. 1313-1330, 2023

[10] B. Gort, M. Serrano, and A. Antonopoulos, "Forecasting Trends in Cloud/Edge Computing: Unleashing the Power of Attention Mechanisms," *Authorea Preprints*, 2023

[11] M. Xu, C. Song, H. Wu, S. S. Gill, K. Ye, and C. Xu, "esDNN: deep neural network based multivariate workload prediction in cloud computing environments," *ACM Transactions on Internet Technology (TOIT)*, vol. 22(3), pp. 1-24, 2022

[12] S. Zhou, J. Li, K. Zhang, M. Wen, and Q. Guan, "An accurate ensemble forecasting approach for highly dynamic cloud workload with vmd and r-transformer," *IEEE Access*, vol. 8, pp. 115992-116003,2020

[13] 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(3), 650, 2023

[14] A. I. Maiyza, N. O. Korany, K. Banawan, H. A. Hassan, and W. M. Sheta, "VTGAN: hybrid generative adversarial networks for cloud workload prediction," *Journal of Cloud Computing*, vol. 12(1), pp. 97, 2023

[15] J. Dogani, F. Khunjush, and M. Seydali, "Host load prediction in cloud computing with Discrete Wavelet Transformation (DWT) and Bidirectional Gated Recurrent Unit (BiGRU) network," *Computer Communications*, vol. 198, pp. 157-174, 2023

[16] D. Alqahtani, "Leveraging sparse auto-encoding and dynamic learning rate for efficient cloud workloads prediction," *IEEE Access*, 2023

[17] S. Huang, Z. Wang, H. Zhang, X. Wang, C. Zhang, and W. Wang, "One for All: Unified Workload Prediction for Dynamic Multi-tenant Edge Cloud Platforms," *arXiv preprint arXiv:2306.01507*, 2023.

[18] Y. S. Patel, and J. Bedi, "MAG-D: A multivariate attention network based approach for cloud workload forecasting," *Future Generation Computer Systems*, vol. 142, pp. 376-392, 2023

[19] H. L. Leka, Z. Fengli, A. T. Kenea, N. W. Hundera, T. G. Tohye, and A. T. Tegene, "PSO-Based Ensemble Meta-Learning Approach for Cloud Virtual Machine Resource Usage Prediction," *Symmetry*, vol. 15(3), pp. 613, 2023

[20] M. ZargarAzad, and M. Ashtiani, "An auto-scaling approach for microservices in cloud computing environments, 2023.

[21] Dataset 1: http://github.com/beloglazov/planetlab-workload-traces%20

[22] 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)* (pp. 1-9), 2020. IEEE.

[23] D. Saxena, and A. K. Singh, "Auto-adaptive learning-based workload forecasting in dynamic cloud environment. *International Journal of Computers and Applications*, vol. 44(6), pp. 541-551, 2022

Authors' Profiles

Dr Kiran G. M. received the Ph.D. degree in Computer Science and Engineering from Visveswaraya Technological University, Belagavi, India, and presently works as an Associate Professor for the Department of Computer Science and Engineering at Shridevi Institute of Engineering and Technology, Tumakuru. He has more than 15 years of teaching experience. He has completed his Bachelor of Engineering in Information Science and Engineering (2009) from Channabasaveshwara Institute of Technology, Karnataka, and M.Tech in Computer Science and Engineering (2011) from Sri Siddhartha Institute of Technology, Karnataka. He has published several papers and book chapters in international conferences and journals, received 5 best paper awards, holds 4 patents, and his research interests include Cloud Computing, Grid Computing, and Ontologies.

Dr. Aparna Rajesh Atmakuri received the Ph.D. degree in computer science and engineering from Visvesvaraya Technological University, Belagavi, India. She is currently working as an Associate Professor with the Department of CSE-AI/ML, Keshav Memorial Engineering College, Hyderabad-India. She has published several papers and book chapters in international conferences and journals, authored four technical books, and hold three patents. Her research interests include cybersecurity, cloud computing, the IoT, and Deep Learning.

Dr. Basavesha D. is presently working with the Department of CSE at Shridevi Institute of Engineering and Technology, Tumkur, Karnataka, India, in the capacity of Professor and Head of the Department. Dr. Basavesha D has received his PhD degree in Computer Science and Engineering from Visvesvaraya Technological University in the year 2022. He completed his postgraduate study (M.Tech) in NMAMIT, Nitte, and graduate study (BE) in Computer Science & Engineering from Sri Siddhartha Institute of Technology, Tumkur. His research interests are Big Data, Data Mining, Data Science, Machine Learning, Algorithms, DBMS, Network Security, and Cryptography. He has more than 15 years of teaching and research experience. He has published 15 international journals and 5 international conference papers in Scopus-indexed and WoS Q2 journals. He has two patents in his name (India and Australia). Dr. Basavesha D has participated in various faculty development programs and conducted value-added programs. He delivered more than 15 tech talks in various colleges and industries. He has guided more than 50 UG and 15 PG students for their academic projects. He also worked as coordinator for the NACC and NBA accreditation process.

How to cite this paper: G. M. Kiran, A. Aparna Rajesh, D. Basavesha, "DS-RAE: Advance Hybrid Double Residual Self-Attention and Recursive Autoencoder Approach for Dynamic Cloud Workload Forecasting ", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.17, No.5, pp. 42-61, 2025. DOI:10.5815/ijigsp.2025.05.04