Journal of Theoretical and Applied Information Technology
31st March 2025. Vol.103. No.6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

DEEP LEARNING-DRIVEN FORECASTING MODELS FOR IOT DATA IN CLOUD COMPUTING ENVIRONMENTS: LEVERAGING TEMPORAL CONVOLUTIONAL NETWORKS

[1]OTHMAN EMRAN ABOULQASSIM, [2]FARHAT EMBARAK, [3]JAYASHREE S, [2]ABDULHAMID ELTHEEB

[1]Computer Science Department, Higher Institute of Science & Technology Ajdabiya, Ajdabiya, Libya
[2]Information Technology Department, Computing and Information Technology Faculty, University of Ajdabiya, Ajdabiya, Libya
[3]Assistant professor, SNS College of Engineering, Coimbatore 641 107[3], jayaa1974@gmail.com
[4]Computer Science Department, College of Information Technology, University of Ajdabiya, Ajdabiya, Libya.

Email: [1]othmanaltaponi@histaj.edu.ly [2]farhat.embarak@uoa.edu.ly, [3]jayaa1974@gmail.com [4]abdulhamid.o.eltheeb@gmail.com

ABSTRACT

Data-driven techniques for machines tool wear detection and forecasting have gained prominence in the past several years. The study investigates how well Temporal Convolutional Networks perform in cloud computing contexts for IoT data prediction. Because TCNs are good at capturing temporal patterns and long-term relationships, they are useful for time-series forecasting problems. Utilizing convolutional layers, TCNs differ from conventional Recurrent Neural Networks in that they analyse data in addition, to enhancing adaptability and decreasing training time. Dilated convolutions are included in TCNs to further improve their capacity to identify trends over long periods without adding to the computational complexity, which makes them appropriate for connections that last and recurrent trends in IoT data. The study shows that TCNs perform better than existing models like RNN, LSTM, GRU-LSTM, and CNN-LSTM in terms of metrics like R^2 alongside and Mean Absolute Percentage Error. The study was conducted on a Python platform running Windows 11. TCNs attained an MAE of 98.7%, RMSE of 97.6%, MAPE of 98.0%, and R^2 of 97.7%, according to the results. Although the error metrics are greater, the significant R^2 value suggests a strong model fit. The study draws attention to many problems with TCNs, such as the requirement for large labelled datasets, understanding, data quality, and computationally demanding requirements. The study also highlights how scalability and flexibility offered by cloud platforms enable effective management of massive IoT data streams and real-time analysis. The results indicate that TCNs may greatly increase resource use and forecasting accuracy in IoT-cloud environmental systems, but more development and study are required to fully realize their capabilities.

Keywords: *IOT Data Forecasting, Cloud Computing In Iot, Temporal Convolutional Network (Tcns), Deep Learning, Dilated Convolutions, Machine Tool Wear Detection*

1. INTRODUCTION

The IoT network is a collection of smart gadgets, including smartphones, tablets, computers, appliances for the home, and sensors, linked together over the World Wide Web. This connection provides the collaboration with other techniques such as machine learning to optimise and analyse the big data and complex operations in connected and rapidly changing environments [1]. In IoT networks, devices must communicate effectively to ensure uninterrupted data transmission while minimizing congestion and interference. Genetic algorithms can optimize communication routes, reduce latency, and enhance data transmission speed [1].

Many applications, including smart homes, energy networks, and transportation, are made possible by this kind of system, which is growing into an indispensable component of daily life. A dynamic worldwide network of interlinked items and devices, IoT may gather and share data amongst itself as well as interact with one another [2]. For servers, storage, databases, and analytics, both hardware and software are utilized since IoT provides a higher level of personal comfort despite its risks. As seen by

Amazon's 2006 EC2 debut, cloud computing has developed over the previous 50 years by offering services such as servers, storage, databases, and data analysis, among others [3]. However, the situation remains delicate. Cloud computing crudity beyond the planet for data. Since it reaches the upper troposphere, it may be analyzed on the same device as the data is generated. The benefits of the cloud are customized for the IoT, but smart gadgets are still vulnerable to security risks, especially in light of the advent of edge computing. Smart devices gather and send you private information that is not protected from a confidentiality standpoint, especially for elderly people who face difficulty in dealing with it and their community [4][5][6]. This makes the devices very appealing to bad software, which may lead to major events like data breaches and lost personal safety. Historically, network administration and security have required a high degree of precision in traffic scaling, and statistical approaches like ARMA and ARIMA have been used to achieve this. IoT benefits from cloud computing's scalability and flexibility, but the security of smart devices is still an issue, particularly in light of the advent of edge computing. Smart gadgets gather and send vast amounts of personal data, which attracts malware and presents risks including compromised personal safety and information exposure. Statistical models like ARMA and ARIMA have historically been used to estimate network traffic accurately, which is essential for resource allocation and security. To solve security and performance concerns, this study suggests using Temporal Convolutional Networks (TCNs) for improved IoT data forecasting in cloud contexts [7]. This proposed study is needed to enhance IoT data forecasting in cloud environments, improving resource allocation, security, and performance using advanced Temporal Convolutional Networks (TCNs). According to the report, there are other ways to address the safety and efficiency concerns associated with IoT devices in addition to fully disclosing how they operate and creating automatic servicing procedures. To solve security and performance challenges, the study recommends adopting TCNs for enhanced IoT data prediction in cloud environments. Improved Internet of Things data forecasting in cloud environments is necessary for more effective, secure, and equitable resource allocation. Internet of Things devices equipped with sophisticated TCNs will safeguard the Internet of Things in the future.

Earlier research has looked into several deep-learning strategies to improve IoT data forecasting accuracy in cloud computing settings. Wu et al [8]. proposed deep learning approach to minimize the latency of wireless communication between edge clouds and end users. Deep learning methods hold great potential for resolving such complex real-world problems, as they are overwhelmingly advantageous in a variety of IOT applications. While broadcasting networks are still in their early stages of investigation, businesses and academics are already quite concerned about this confluence. This research mainly assesses the latest advances in academia and the noteworthy technical use of DL in wireless communications advancement. By doing this, it seeks to address new practical and theoretical issues as well as basic ideas that will serve as guidelines for research for the next wireless network layouts and uses. In deep learning-driven wireless communications, this study highlights the core ideas and methods of end-to-end interaction, signal acknowledgment, channel estimate, and reduction of detecting, decoding, encoding, security, and secrecy. More instances of the primary challenges, potential benefits, and developing developments regarding the combination of DL systems in mobile communication scenarios are given. Takur et al. [9] recommended to gives a thorough rundown of how deep learning is affecting the Internet of Things (IoT), covering how sensor data is analyzed to find patterns and anticipate outcomes, and how this will affect various sectors highlighting manufacturing, healthcare, DL structures, architectures, IoT connection terminology, IoT apps, and the functions and problems of DL in IoT are all covered in this survey report. The paper also includes quantitative results that show how IoT and DL may affect circumstances like energy use and precision farming. All things considered, the survey study is an invaluable tool for academics who want to learn more about the possibilities of IoT and DL in their subject.

Several deep learning techniques have been studied in the past to improve IoT data predictions in cloud computing environments. Using DL to lower wireless connection latency among end users and outer clouds. While this approach shows promise, it has issues with scalability and processing in real-time. Examined how DL affects the Internet of Things by examining sensor data to find trends and forecast results in industries like healthcare and industry such as [10][11]. Although thorough, this method frequently suffers due to elevated computing expenses and the requirement for huge datasets to train algorithms efficiently. Furthermore, even though they are strong, conventional DL models like CNNs and LSTMs may have trouble with intricate temporal connections and need a lot of fine-tuning. While several of these constraints can be addressed by more recent techniques like TCNs, which also enable improved handling of consecutive data and fewer training durations, additional research is still needed to fully optimize their use in a variety of IoT contexts.

The key contribution of the proposed study id following below,

- Employing TCNs to overcome the drawbacks of conventional mathematical approaches like ARMA and ARIMA to increase the forecasting accuracy of IoT data in cloud computing settings.
- Demonstrates how more precise forecasts of network activity and utilization of resources made possible by sophisticated DL frameworks may optimize the use of resources in the Internet of Things wireless networks.
- Utilizes DL techniques to identify and neutralize possible attacks, improving both the safety of individual information and device integrity. This approach addresses safety issues in cloud and IoT contexts.
- Demonstrates the advantages of the scalability and flexibility of online computing and how TCNs can effectively handle the increasing amount and intricate nature of IoT data.
- Provide a thorough analysis of the most recent developments and uses of DL in the Internet of Things, involving topics such as identifying anomalies, signal identification, and end-to-end communication. This evaluation will help plan upcoming research and practical applications.

The remainder of this study is structured as follows. The review of literature of this study is briefly explained in Part 2. Part 3 briefly discuss the problem statement of this study. Part 4 describe the methodology of the proposed study. Section 5 discusses the result and conclusion. This paper ends with concluding remarks in Part 6.

2. LITERATURE REVIEW

Saxena et al. [12] proposed a study that gives a comprehensive evaluation of machine learning models for predicting cloud workloads. This research is relevant to Deep Think IoT. The strength of DL in the Internet of Things because it focuses on advanced deep learning practices to elevate IoT applications. The study includes building machine learning models such as SVM, Random Forests, and Neural Networks to predict cloud resource usage accurately through optimization. It underscores the usage of those models in efficiently handling the cloud environment and resource management improvement. To collect data, the researchers turned to the cloud data centers workload traces of data gathering, then followed them with the preprocessing and feature extraction process. The data was partitioned in training and testing subsets, usually the models were trained on 80% and tested on 20% or 70% and 30%, to assess and assure the performance. There are several issues with using quantum neural networks and explainable AI to solve the forecasting challenge. These solutions frequently include intricate optimization techniques, computational demands are rising overall, and open, equitable, and precise AI-powered resource control can be challenging.

The methodology suggested by Al-Ghuwairi et al. [13] uses a framework for forecasting built on the Facebook Prophet approach in conjunction with a feature selection (FS) method to evaluate its efficacy. We present the FS approach, an integrated feature selection method that combines stationary, causality, and anomaly detection tests with time series analytic methodologies. The difficulty of making false linkages among time series abnormalities and assaults is particularly addressed by this method. Our findings show that the number of predictors used in the forecasting model was significantly reduced from 70 to 10 while metrics for performance including Made, DTW, MAE, MSE, and RMSE were all improved. Moreover, this methodology has led to reductions in cross-validation, prediction, and training durations of around 97%, 15%, and 85%, respectively. While the use of memory stays the same, utilization time has decreased dramatically, leading to a large reduction in resource usage. By using less training data and resources, the suggested approach decreased the total amount of input forecasting factors and increased forecast accuracy. The investigations were carried out using a sizable, current dataset. The tests have shown promising results, including increased forecast accuracy, less complexity, fewer input predictors, and shorter prediction times. Although the suggested approach produces respectable results, further training data and hyperparameter adjustment are required to maximize its efficacy. Subsequent research necessitates more complex trials, feature selection, and evaluation against other forecasting models.

To solve the problems and go above the limitations of the earlier work, Xu et al. [14] introduced a deep neural networking approach dubbed esDNN, which is an algorithm for supervised learning based on efficient prediction of cloud workload. Initially, a sliding windows representation is employed to generate a multivariate data set which can be utilized to train the DL. To obtain accurate prediction, an upgraded GRU built with the greatest amount of recent data is then employed. In addition, the author presents the outcomes of authentic trails operating on Google and Alibaba data centers to illustrate the functionality of esDNA. The outcomes demonstrate that the esDNN is capable of accurately and efficiently predicting the cloud workload. When esDNN is contrasted with the most sophisticated baselines, it is included in the research

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

carried out by the cloud prediction system that predicts esDNN the best. by up to 15% compared to the other one using only GRU. Specifically, by reducing the number of active hosts, which lowers host expenditures, this study uses esDNN to auto-Sc to the computer. Using the strategy for supplying optimization in a container-based prototype platform like Kubernetes has a downside. To accommodate some edge computing apps that require a low response time to consumer mobility and position changes, earlier research suggested and put into practice automated unloading techniques that were highly complex. Achieving effective, flexible system management becomes further complicated and challenging when automated esDNN is implemented with the MAPE paradigm.

Humayun [15] Research was suggested by Humayun [15] to identify the burgeoning tendencies in CC, BGD, and IoT. The integration of these methods and how they affect different applications in real-time, as well as the advantages and difficulties of each method, present market trends, and potential areas for future study with a particular emphasis on the healthcare industry. Additionally, the study offers a theoretical framework that combines CC, BGD, and IoT to create an IoT-focused cloud architecture. Since the healthcare sector is one of the most significant real-world applications, this study also covered the new real-time applications of IoT, CC, and BGD. Additionally, some statistics from reputable sources were included to demonstrate how these three theories are gaining market share in the sector. However, by using these three paradigms in real-world applications, certain statistics on the benefits and drawbacks of the merging of IoT, CC, and BGD may be discovered. Ultimately, this study has explored how the combination of each of these models contributes to maximizing the advantages of modern innovations, and we have offered a structure that illustrates the interrelationships among these three conceptions.

To recreate printed documents using text-based steganography, Stergiou et al. [16] presented a safe architecture that makes use of big data analysis in cloud settings interconnected via IoT networks. This has been accomplished by repeatedly encoding text into an item after it has been transformed into characters. The structure ensures the confidentiality and integrity of data by using cutting-edge encryption methods to safeguard data during storage and transport. To evaluate and deal with the data, it incorporates several ML techniques, strengthening privacy protocols in cloud-based Internet of Things networks. The study's findings demonstrate that under certain circumstances, knowledge retrieval in its entirety is accomplished. Even if an item can be torn, data can still be partially recovered from its ripped sections, aiding in the content's reconstruction. The research is not without limits, though. The cover object is an RTF file. For implantation, only texts are utilized. Paper of the A4 size is used for printing. Exclusively text data is utilized with excellent scanners and OCR software, such as the HP Scanjet G4050A4 scanner with OCR Application C1957A. This may also be expanded to include additional formats including data in tables, graphics, and logos. The research is not without limits, though. The cover object is an RTF file. For implantation, only texts are utilized. Paper of the A4 size is used for printing. Exclusively text data is utilized with excellent scanners and OCR software, such as the HP Scanjet G4050A4 scanner with OCR Application C1957A. This may also be expanded to include additional formats including data in tables, graphics, and logos.

Haji et al. [17] suggested research that examined the uses of IoT and cloud computing innovations to find complementing elements in a distinct setting from the primary forces behind the Future Internet. The future of the Internet is being drawn by the Internet of Things and cloud computing. New uses for these modern innovations are always emerging, providing fascinating new avenues for research as well as industry. We identified the primary research issues of relevance for each of the many fresh applications that were made possible by the adoption of the IoT and Cloud Computing paradigms. The Internet of Things (IoT) enables connections and interactions between a vast number of objects to share data, expertise, and data that improves the standards of daily life. As a substitute, cloud computing offers suitable, flexible, and on-demand network access, which enables the contribution of processing power that aids in the integration of dynamic data from different sources of data. However, there are too many obstacles and challenges in this research to apply cloud computing and IoT in FI. The goal of the present study is to provide an overview and clarification of the core ideas behind cloud computing and the Internet of Things.

This literature review looks at a wide range of machine learning and deep learning approaches that may be used for IoT and cloud workload prediction jobs. It also highlights the issues, optimal solutions, and improvements. They address difficulties with computational demands, management of resources, and accuracy enhancements across diverse real-world applications, and they describe the most beneficial ways to track the optimal use of resources HRM through cutting-edge machine learning, creative neural network algorithms, and effective architectures for the case of the characteristic vital tasks involved.

3. PROBLEM STATEMENT

The IoT is made up of networked, web-connected devices that use sensors to collect data from their surroundings, analyses it, and transmit it via a network. Cloud computing is a type of computing service wherein Internet-based resources, such as servers and storage, are made available to clients based on their specific needs. Although resource administration and cloud workload forecasting technologies have advanced significantly, there are still many significant problems that negatively impact these systems' functionality. Saxena et al. [12] emphasized the computing requirements and computational complexity regarding quantum neural networks and explainable AI for predicting problems, pointing out that such approaches frequently call for complicated optimization strategies. Al-Ghuwairi et al. [[13]] noted that although their Facebook Prophet-based forecasting technique and feature selection model greatly shortened the training and forecasting timeframes, their system's efficacy is limited by the requirement for substantial hyperparameter tweaking and more training data. Moreover, the esDNN approach for cloud workload prediction was presented by Xu et al. [14], who also showed enhanced prediction accuracy and resource optimization. They did, however, also note that sophisticated offloading techniques and adaptability in handling system operations are required, especially when integrating such as Kubernetes. The manifestation of these flaws highlights how crucial it is to adopt a more flexible and successful tactical planning approach for IoT data forecast in cloud systems. To address these drawbacks, the proposed method makes use of TCNs to produce and analyse the projection structure through ML, thereby minimizing computational prerequisites, requiring less application, and delivering precise and effective cloud-managed approaches. Additionally, this will increase the system's administrative adaptability. Other disadvantages are also considered, including higher processing needs, intricate unloading strategies, and resource consumption restrictions. This approach will increase the method's IoT data forecasting effectiveness and adaptability in cloud computing platforms.

4. PROPOSED TCN FOR FORECASTING IOT DATA IN CLOUD COMPUTING ENVIRONMENTS

The proposed IoT seeks to identify Internet of Things data via cloud computing. To make precise and effective predictions, these networks appropriately identify time-related relationships in the time-series data. For many IoT applications, the use of TCNs in the cloud facilitates scalability, real-time analysis, and decision-making. TCNs are capable of recording long-range correlations via causal convolutions and dilation, they are perfect for anticipating IoT data. Because TCNs provide parallelism during training and do not experience gradients that diminish as RNNs do, they are an excellent choice for effectively managing enormous amounts of time-series data.

Fig. 1. Internet of Things

Fig 1. depicts several facets and uses for IoT. This illustrates the vast and varied IoT ecosystem. The "IoT" is represented by the middle green hexagon, which shows symbols of many gadgets and technologies, including wearables, cloud computing, networking, sensors, and data analytics.

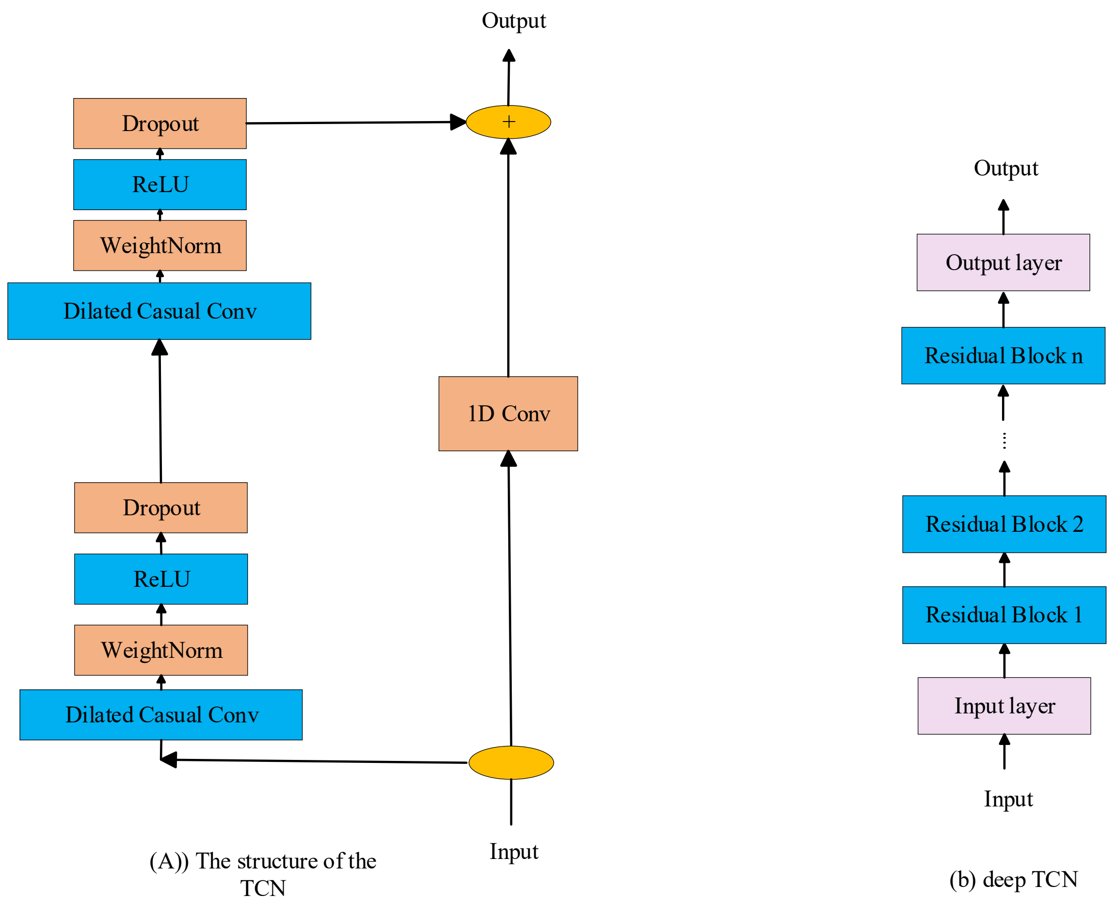

Fig. 2. Architectures of proposed model TCN

The Fig 2 provides an outline of a simple TCN is, emphasizing its main elements and the organization's framework. The proposed frame work utilize IoT data as their input and preprocess it, the framework consists of an IoT data collecting and preliminary processing block, a Dilated Causal Convolution block, and a residual block. These effectively provide the projected output by employing the TCN to capture connections in time.

4.1. Data Collection

In this study, "Real-world IoT data for environmental analysis IOT sensor DHT for temperature, humidity and heat index" was collected from Kaggle. The dataset is an accumulation of data that a DHT sensor stored for almost a year. Investigating relationships between temperature and humidity and time series analytics can benefit from this data. Shortly, it will improve and clean the dataset with more thorough data [18].

Table I: Type Styles

NO.	**_time**	**Heat_index**	**Humidity**	**Temperature**
0	2022-043T21:38:00Z	26.389790	32.400000	26.800000
1	2022-413T21:39:00Z	26.39[13]28	32.433333	26.800000
2	2022-043T21:40:00Z	26.377855	32.100000	26.800000
3	2022-04-T21:41:00Z	26.401618	31.916667	26.850000
4	2022-04-T21:42:00Z	26.414282	31.716667	26.883333

Table I. displays time-series data recorded from an IoT sensor. Each row represents a timestamp, with columns or Heat_index, Humidity and Temperature For instance, at "2022-04-13T21:38:00Z", the heat index was 26.39, humidity was 32.40%, and temperature was 26.80°C. The data is collected at one-minute intervals, capturing variations in environmental conditions over time [19].

4.2. Data Pre-processing

The process of cleaning, converting, and integrating raw data to get it ready for additional analysis or data processing steps is known as data preparation. To enable more efficient data mining and analysis, it seeks to transform redundant, noisy, or incomplete data into a usable format. For this proposed study, the data pre-processing involved data cleaning, splitting, and normalization.

4.2.1. Data Cleaning: *A series of crucial yet required steps known as "data cleaning" get the data ready so that accurate modelling may take place. You can discover neighbouring data points that provide estimations of gaps and replace the missing values using forward-filled, backward-filled, or interpolated imputation processes. Damaged columns or rows may be eliminated if imputation is impractical due to a large amount of missing data. To maintain the dataset's data credibility, misfits are identified using numerical methods such as the Z-score IQR, and these variations are then either rectified or eliminated. Two primary activities are involved in data cleaning: (1) locating duplicates and eliminating them, and (2) discovering gaps in the data via in-depth inspection and some kind of imputing to fill in the gaps using techniques like forward fill or interpolation. This guarantees the dataset's accuracy and error freeness for the successful forecasting of TCN systems [20].*

4.2.2. *Data Splitting:* To get reliable findings, data splitting for forecasting time- series involves dividing the data into validation, training, and test sets whilst preserving the chronological structure of the data. Periodic partitioning of the dataset guarantees the use of past events to forecast future events, hence preserving reasonability *[21]*. Typically, the data is divided into three categories for training purposes: 70% for training, 15% for validation, and 15% for testing.

4.2.3. *Data Normalization:* In general, the process of producing clean data is known as data normalization. On closer inspection, nevertheless, it becomes clear that data normalization has two functions or meanings:

a) Data normalization is the act of arranging data so that it looks consistent in every entry and document.

b) Enhancing entry-type consistency facilitates lead development, cleansing, and better segmentation of data processes.

$$X' = \frac{X - X_{minimum}}{X_{maximum} - X_{minimum}} \quad (1)$$

Where X is the pixel value, $X_{minimum}$ is the image minimum pixel value, and $X_{maximum}$ is the maximum pixel value in the image. This scales the pixel values to a range of [0, 1] or [0, 255] depending on the requirement of the model [18].

4.2.4. *Handling missing data:* One of the most important steps in data preparation is handling missing data, which entails methods for filling in dataset gaps. Incomplete datasets are the outcome of missing data, which happens when an element is devoid of data points. Typical techniques include:

a) Deletion : Remove any rows or columns that have missing values.

b) Imputation: Use the mean, median, mode, or forecasts to fill in the blanks.

The missing data handling technique was briefly explained in eqn (2)

$$x_{new} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (2)$$

where x_{new} is the imputed value and x_i are the observed values [23].

4.3. Feature Extraction

Fig 3. represents the feature extraction process within a Temporal Convolutional Network (TCN) model designed for forecasting IoT data. The raw IoT data is first passed through several 1D convolutional layers and a residual block to protect data. Included in the TCB are Dropout, ReLU, and WeightNorm. The data travels to the output layer via a completely linked structure.

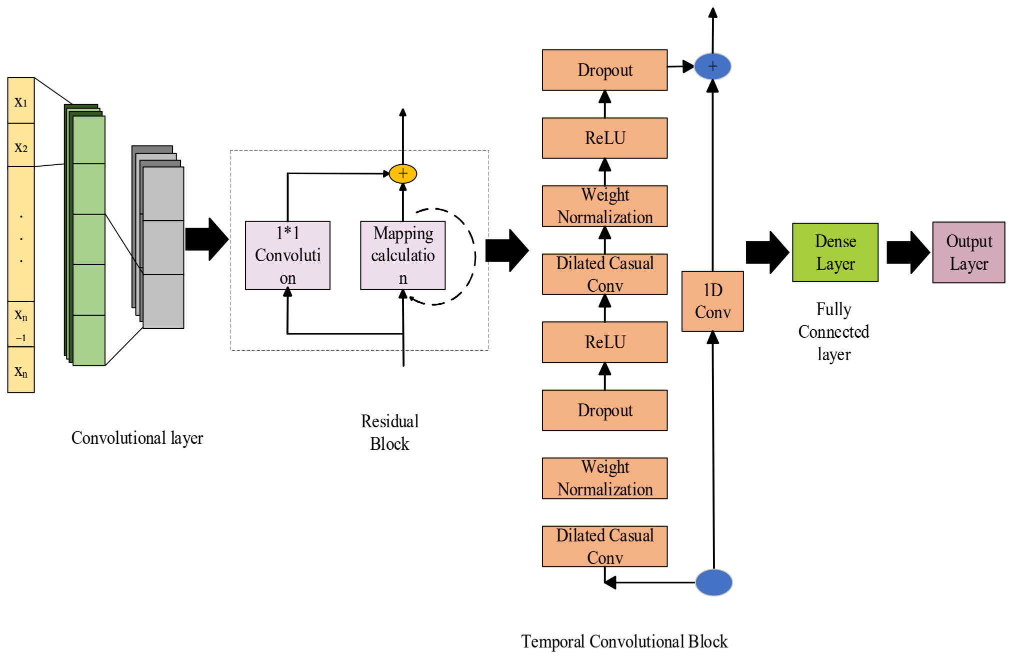

Fig. 3. Architectures of proposed model TCN

4.3.1. Convolutional Layer: Applying 1D convolutions to the input data, convolutional layers are a kind of spatial layer. When certain features, including patterns and trends, are identified, it becomes easier to predict a model correctly since the time-series data's potential impacting regions of interest are identified. The layers' function is to provide the model with a sense of time so that it may make predictions based on the sequence's context *[24]*.

4.3.2. *Residual Block:* To swiftly combine the next layers and skip certain system layers, the residual block performs residual activities. In the latter, gradients can move through the network instead of being choked, and the key information is also maintained thanks to the architecture. Remaining blocks give the structure a shortcut across layers, making it easier for it to understand complex sequences; Their justification did not center on achieving the standard deviation of gradient dispersion during training. La Giak without weight conditioning and with a degree of normalization *[25]*.

4.3.3. TCB*:* TCB consists of three parts: The first one discusses TCB's feature extraction capabilities. In such a situation, the WeightNorm approach provides the opportunity to get the same weights, whereas the normal method would result in an unreliable and inefficient procedure that changes constantly. Rectified Linear Units, or ReLUs, introduce non-linearity to the data to help it absorb and handle complex data structures and relationships. Dropout is a common technique that helps to regulate the neural network by randomly removing a particular number of units during the training phase. Excessively dependent on a single neuron, dropout is unable to sufficiently generalize to new information [26].

4.3.4. Fully Connected Layer*:* The layer in which the temporal and convolutional blocks were used to improve the retrieved features before they were attached in the FCL. The system discovers how the features relate to one another and how to create a forecast based on the input by combining these features in a sophisticated way *[28]*.

4.3.5. Output Layer*:* The output layer, the network layer also known as the output layer, is the portion of the network that will examine the data that has been handled and develop value predictions. It makes predictions about future data points based on the information the network has learned.

Algorithm 1: Pseudo Code for Deep Learning-Driven Forecasting Model for IoT Data in Cloud Computing Environments

Input: IoT data sequence A

Journal of Theoretical and Applied Information Technology
31st March 2025. Vol.103. No.6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Output Forecasted IoT data sequence B

Load the dataset

Normalize the IoT data sequence A

Split the data into training and testing sets // ***Data Preprocessing***

Initialize TCN model with specified hyperparameters

Layers of the TCN

Dilated 1-D Convolutional Layers

Skip Connections

Batch Normalization

Parametric ReLU (PReLU) activation

Deepwise Convolutional Layers // ***Model Initialization***

for epoch in range(num_epochs)

for each batch in training data

Forward pass through the TCN layers

Compute loss

Backward pass to compute gradients // M***odel Training***

Use the trained model to predict on the testing set

Calculate performance metrics // ***Model Evaluation***

Use the trained TCN model to forecast future IoT data points // ***Forecasting***

Output the forecasted IoT data sequence B // ***Output***

5. RESULLT AND DISCUSSION

The outcomes of the TCN-based models for forecasting will be presented and analyzed in the study's results section. This part will contain an analysis of the TCN approach's performance on many IoT datasets, a thorough comparison with other cutting-edge techniques, and a computational efficiency assessment.

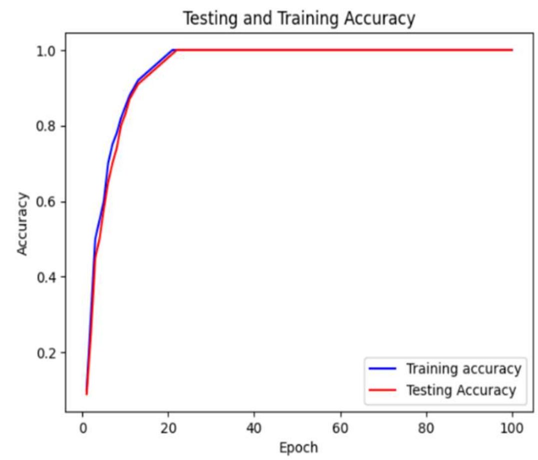

Fig. 4. Testing and Training Accutacy

Figure 4 shows how testing and training accuracy increased dramatically across epochs. Both accuracies started at zero and grew gradually; by the 100th period, training accuracy had reached 0.99 and the accuracy of testing had reached 0.95. This illustrates how well the model learns and how well it generalizes, guaranteeing reliable results in practical settings. By the 100th epoch, the training accuracy of the suggested model had reached 0.99, while the testing accuracy had reached 0.95, indicating strong learning and generalization. A consistent reduction in testing and training losses signifies efficient learning, low excessive fitting, and robust efficiency, guaranteeing accurate and dependable forecasting in real-world scenarios.

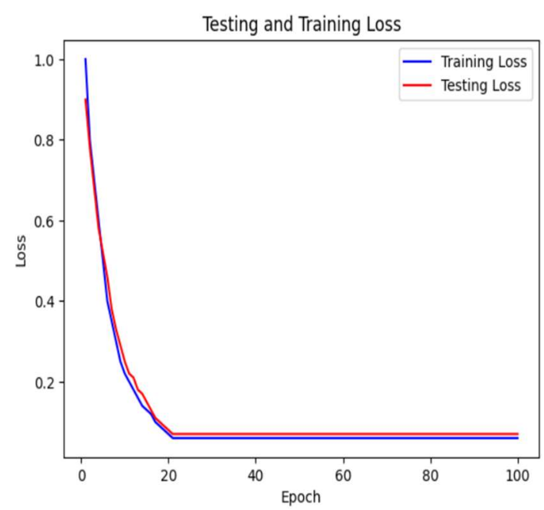

Fig. 5. Testing and Training Loss

The testing and training loss for a deep learning model throughout epochs is depicted in the figure 5. Since both losses are excessively high at first, the model has not yet been trained. This complicates the process of learning about weights for us. Successful

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

learning is shown by rapid declines in the early epochs. Losses are the same after 20 epochs of loss. Conversely, the loss from training is decreasing over time, with a few small oscillations, but it is also growing in tandem with the testing loss, which is falling, meaning the model is grasping the general notion of the right direction it has to travel.

(a) (b)

Fig. 6. Training and Validity (a) Accuracy, (b) Loss

In fig 6, the graphs show the model's learning process. The training loss decreases while the accuracy increases, indicating improvement. Validation loss and accuracy fluctuate more, suggesting variability in performance on unseen data. If validation loss rises or accuracy plateaus significantly, it may indicate overfitting, where the model learns training data well but generalizes poorly to new data.

5.1. Performance Evaluation

Usually, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and Coefficient Determination are used as performance measures in this study (R^2).

5.1.1. MAE: The median magnitude of errors in forecasting, or the discrepancy between predicted and real outcomes, is determined using the mean absolute mistake formula *[29]*. The MAE of the proposed study is explained in eqn (3),

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \tilde{y}_i| \quad (3)$$

5.1.2. *RMSE:* By computing the inverse of the root mean squared errors, RMSE, in contrast to MAE, draws attention to more significant mistakes *[30]*. The RMSE of the proposed study is explained in eqn (4),

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2} \quad (4)$$

5.1.3. *MAPE:* The mean absolute percentage error, or MAPE, is an equation that shows how accurate a forecasting tool is by calculating the real mistake as a proportion of the actual data The MAPE of the proposed study is explained in eqn (5),

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} \left|\frac{y_i - \tilde{y}_i}{y_i}\right| \times 10 \quad (5)$$

5.1.4. *R-Squared:* It displays the proportion of the dependent variable's volatility that can be forecast using the independent variables *[31]*. This formula is used to calculate it is computed with the following formula of eqn (6),

$$R^2 = 1 - \frac{\sum_{i-1}^{n}(y_i - \tilde{y}_i)^2}{\sum_{i-1}^{n}(y_i - \bar{y})^2} \quad (6)$$

where y_i is the actual value, $\tilde{y}$ is the predicted value, and $\tilde{y}_i$ is the mean of the actual values.

Table II: Performance Evaluation

Metrics	Percentage ((%)
MAE	98.7
RMSE	97.6
MAPE	98.0
R2	97.7

Table II illustrate the performance indicators of the model according to the given percentage. The average size of the mistakes is revealed by the 98.7% MAE (Mean Absolute Error). It results in a mismatch between the expected and actual numbers, while the 97.6% RMSE suggests a significant degree of prediction error unreliability. The average percentage error of the MAPE forecasts is 98%, as indicated by 98.0%. The R2 value of 97.7%, demonstrates a strong correlation between the expected and real values, indicating that the framework describes 97.7% of the variance in the data throughout the study of non-specialists.

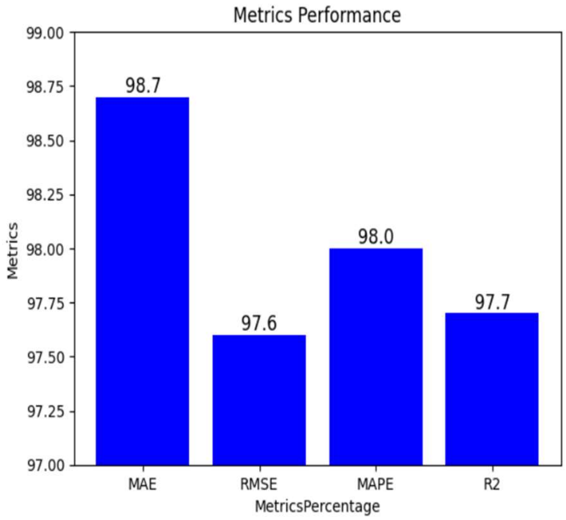

Fig. 7. Graphical represent of Performance Evaluation

Fig 7. explains the measurements are displayed as vertical bars on the bar chart, each of which represents a percentage. Because of the significant mistakes and variations in the forecasts, MAE, RMSE, and MAPE indicators show that they are high. Conversely, the R2 bar is nearly non-existent, indicating a good fit between the model and the primary explanation power. The difference between this set of data and the beautiful graph shows that, despite the model having a high coefficient of determination and being quite precise, it is affected by very real prediction errors caused by the MAE, RMSE, and MAPE, indicating the need for further work to improve the accuracy of the forecasts.

Table III: Performance Evaluation

Model	**MAE (%)**	**RMSE (%)**	**MAPE (%)**	R^2 **(%)**
RNN	75	80	93.5	88
LSTM	97.9	97.0	97	95
GRU-LSTM	90.9	83.3	96	93
CNN	66.7	41.2	95	90
CNN-LSTM	80	74.8	97.5	97
TCN (proposed)	98.7	97.6	98.0	97.7

Performance metrics for the following models are compared in the table II: CNN [32], CNN-LSTM [33], GRU-LSTM [34], RNN [34], LSTM [35], and TCN (proposed). The TCN model has the best R2 and the highest values of MAE, RMSE, and MAPE showing that it adequately clarifies the variation in the data even with greater error rates.

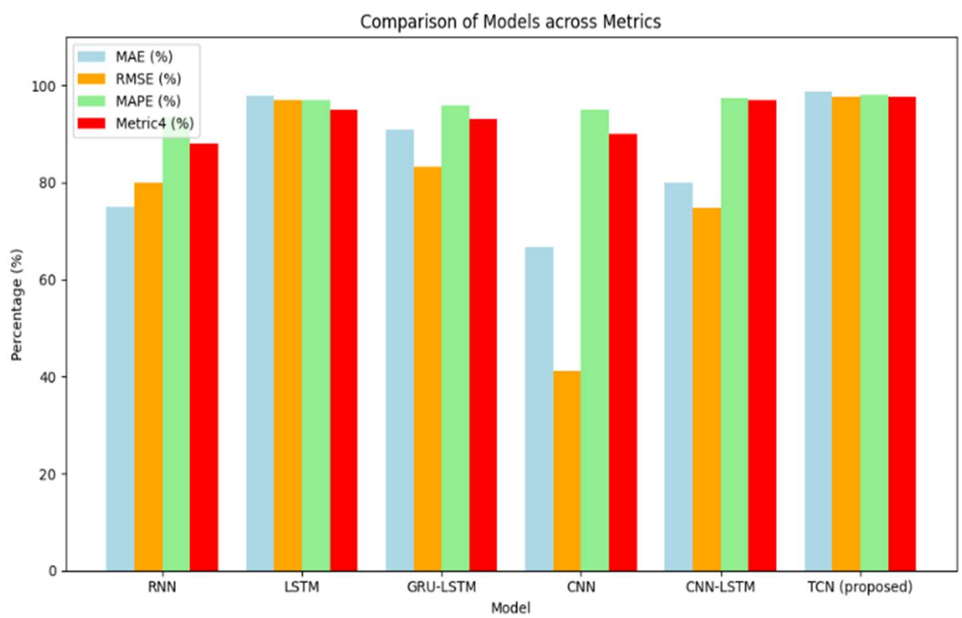

Fig. 8. TCN Comparison graph with Existing Models

In fig 8, the graph compares seven models across four metrics: MAE, RMSE, MAPE, and Metric4. RNN shows moderate performance, while LSTM and GRULSTM improve notably, particularly in MAE and RMSE. CNN excels in MAE and RMSE, and CNN+LSTM combines their strengths, performing well across all metrics. The proposed TCN model outperforms all others, with the lowest values in MAE, RMSE, and MAPE, reflecting higher accuracy and efficiency. This visual comparison highlights TCN's superiority and aids in identifying the most effective model overall.

5.2. Discussion

According to the study, because time-series prediction operations properly capture temporal patterns and long-term correlations, all aspects of IoT development will be completed. Be aware that neural networks are used in over 60% of situations on average by IoT ML algorithm and Artificial hardware. Because TCNs can store long-term associations and modify sequence data well, they are very beneficial for IoT data prediction. Unlike conventional RNNs, TCNs use convolutional layers to deal with input in tandem; as a result, they can adapt more flexibly and accelerate learning to a certain degree. One special aspect of expanded TCN is that it allows learning across a large number of periods by using dilations, which don't add to the computational difficulty [36]. This is especially important for data from the Internet of Things since the data source frequently deals with repeating events and long-term connections. Additionally, they have a fixed-size responsive field, meaning that every characteristic in the input sequence makes an identical contribution to the output forecasts. This means that they improve predicting accuracy without changing. Because the data that is received by the gadget is constantly changing and requires prompt, immediate evaluation, TCNs are a dependable instrument for data presentation in the Internet of Things. Data quality is still another important consideration. The majority of the data in the IoT data collection is distorted, missing, or sampled sporadically. This might make preprocessing the method of forecasting more challenging and lower the precision of the prediction. While confronting the loss of data and extreme/excessive data when the data is dispersed and variable is difficult, it is crucial [37][38]. The granularity of the framework is one issue that might result in erroneous predictions of the model. Even while TCNs may eventually understand complicated trends, the decision hierarchy of a model like that may appear a little erratic and confusing. The "black-box" DL models might be the cause of problems in situations where comprehension and data are critical. Moreover, one of TCNs' limitations on capacity is that training them requires a significant number of labelled datasets. IoT data labelling is a common practice, but it requires a significant investment of resources and time [39]. Data labelling is a labor-intensive and resource-intensive activity. Without annotated data, it can be challenging to develop and refine reliable forecasting models. Making forecasts on IoT data using TCNs and cloud computing technologies is an enigma since there are benefits and drawbacks. Another benefit of the cloud is its scalability and flexibility, which makes it feasible to store and handle enormous amounts of data from the IoT without having to shell out a lot of money for the necessary equipment upfront [40]. TCNs have high computational demands, but the cloud can assist them by providing powerful GPUs and TPUs that accelerate model inference and training. Moreover, cloud solutions provide real-time data distribution and continuous integration, both of which are essential for Internet of Things devices given the need for accurate and timely forecasting. Since TCN models may be rapidly modified and updated on the cloud, their predictions need to be sufficiently flexible to take into account the most recent circumstances and data.

6. CONCLUSION

The research projects demonstrated how TCNs improve IoT data forecasting precision in cloud computing scenarios. Furthermore, TCNs use DL to solve specialized statistical techniques with restrictions, such as ARMA and ARIMA, which enable the capture of precise time series. A Python platform was used to conduct the TCN investigation, and the findings produced [13] Windows. The experiment can also demonstrate how TCNs can improve the monitoring system by spotting threats and behaviors and by increasing resource utilization. The results showed that the mean absolute errors (M-MAE) of the TCNs were 98.7%, the root mean squares error (RMS-ME) was 97.6%, the M-APE was 98.0%, and the R^2 was 97.7%. However, even if it may be inferred that the error measurements are realistically greater, a high R^2 value is still a reliable sign of a well-fitting model. The construction of a new AI predictor and equipment for multiple initiatives via the use of state-of-the-art AI in the study and the incorporation of huge datasets with labels in the study were discussed complexities. The paper goes on to emphasize the significance of cloud platforms, which aid in the efficient administration of enormous amounts of data in addition to offering flexibility and scalability. The results and discussion effectively fulfill the study's key contributions by demonstrating the advantages of TCNs in handling IoT data forecasting and resource optimization. The TCN model shows superior performance in training and testing accuracy (0.99 and 0.95, respectively), which illustrates strong learning and generalization. The consistent reduction in losses and minimal overfitting further affirm its reliability. Performance evaluation metrics (MAE, RMSE, MAPE, and R^2) highlight that the TCN model surpasses traditional models like RNN, LSTM, and CNN, achieving high accuracy in network activity prediction and resource utilization. Moreover, its strong results in MAE, RMSE, and MAPE emphasize TCN's effectiveness in mitigating errors and making precise forecasts, which is crucial for improving IoT network security and scalability. This validation of TCN's performance establishes it as a robust tool for future applications in anomaly detection and resource management in IoT environments. The future work of this study includes refining the TCN architecture to enhance forecasting accuracy, integrating real-time IoT data processing, and ensuring scalability for larger datasets and diverse IoT environments. It also involves adding advanced cybersecurity measures, comparing TCN with emerging models like transformers, and exploring energy-efficient solutions for large-scale forecasting. Additionally, multimodal data fusion will be investigated to improve model robustness across various IoT tasks. These advancements will push the boundaries of IoT data forecasting and resource optimization, making TCN-driven solutions more accurate, adaptable, and sustainable.

REFERENCES

[1]. Aboulqassim, o. A comparative study of genetic algorithms and it's applications in secondary schools, chapter 1 and 3, ttp://search.mandumah.com/record/866774, sert univesity, (2010). Md no: 866774, pages 1-150.

[2]. A. Derhab, A. Aldweesh, A. Z. Emam, and F. A. Khan, "Intrusion detection system for internet of things based on temporal convolution neural network and efficient feature engineering," Wirel. Commun. Mob. Comput., vol. 2020, no. 1, p. 6689134, 2020.

[3]. M. Abouelyazid, "Forecasting Resource Usage in Cloud Environments Using Temporal Convolutional Networks," Appl. Res. Artif. Intell. Cloud Comput., vol. 5, no. 1, pp. 179–194, 2022.

[4]. Adbelrjal, F., Ismail, N. A., & Othman, S. (2020). The impact of online social community platforms on the elderly daily activities: a systematic literature review. Int J Future Gener Commun Netw, 13(1), 877-902.

[5]. Embarak, F., Azman Ismail, N., Shahin, O. R. R., & Alabdali, R. N. N. (2022). Design of autonomous online social community architecture for older adults. Computers and Electrical Engineering, 100, 107900. https://doi.org/10.1016/j.compeleceng.2022.107900.

[6]. Embarak, F., Ismail, N. A., Ghazali, M., Mahmud, M., Zuraifah, N., Othman, S., Hasan, L., Said, C. S., Yong, Y., & Alsayed, A. O. (2023). Online Social Interaction Community Model for Unmet Needs of Older People.

Journal Of Theoretical And Applied Information Technology, 101, 22. Www.Jatit.Org

[7]. J. Bi, X. Zhang, H. Yuan, J. Zhang, and M. Zhou, "A hybrid prediction method for realistic network traffic with temporal convolutional network and LSTM," IEEE Trans. Autom. Sci. Eng., vol. 19, no. 3, pp. 1869–1879, 2021.

[8]. H. Wu, X. Li, and Y. Deng, "Deep learning-driven wireless communication for edge-cloud computing: opportunities and challenges," J. Cloud Comput., vol. 9, no. 1, p. 21, 2020.

[9]. D. Thakur, J. K. Saini, and S. Srinivasan, "DeepThink IoT: the strength of deep learning in internet of things," Artif. Intell. Rev., vol. 56, no. 12, pp. 14663–14730, 2023.

[10]. S. B. JUGUNTA *ET AL.*, "EXPLORING THE INSIGHTS OF BAT algorithm-DRIVEN XGB-RNN (BARXG) FOR OPTIMAL FETAL health CLASSIFICATION IN pregnancy MONITORING," *INT. J. ADV.* comput. *SCI. APPL.*, VOL. 14, NO. [13], PP. 731–741, 2023, DOI: 10.14569/IJACSA.2023.014[13]74.

[11]. A. O. ALSAYED, N. A. ISMAIL, L. HASAN, AND F. EMBARAK, "A COMPREHENSIVE REVIEW OF MODERN METHODS TO IMPROVE DIABETES SELF-CARE MANAGEMENT SYSTEMS," INT. J. ADV. COMPUT. SCI. APPL., VOL. 14, NO. 9, PP. 182–203, 2023, DOI: 10.14569/IJACSA.2023.0140920.

[12]. D. Saxena, J. Kumar, A. K. Singh, and S. Schmid, "Performance analysis of machine learning centered workload prediction models for cloud," IEEE Trans. Parallel Distrib. Syst., vol. 34, no. 4, pp. 1313–1330, 2023.

[13]. A.-R. Al-Ghuwairi, Y. Sharrab, D. Al-Fraihat, M. AlElaimat, A. Alsarhan, and A. Algarni, "Intrusion detection in cloud computing based on time series anomalies utilizing machine learning," J. Cloud Comput., vol. 12, no. 1, p. 127, 2023.

[14]. 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 Trans. Internet Technol. TOIT, vol. 22, no. 3, pp. 1–24, 2022.

[15]. M. Humayun, "Role of emerging IoT big data and cloud computing for real time application," Int. J. Adv. Comput. Sci. Appl., vol. [13], no. 4, 2020.

[16]. C. L. Stergiou, A. P. Plageras, K. E. Psannis, and B. B. Gupta, "Secure machine learning scenario from big data in cloud computing via internet of things network," Handb. Comput. Netw. Cyber Secur. Princ. Paradig., pp. 525–554, 2020.

[17]. L. M. Haji, O. M. Ahmad, S. Zeebaree, H. I. Dino, R. R. Zebari, and H. M. Shukur, "Impact of cloud computing and internet of things on the future internet," Technol. Rep. Kansai Univ., vol. 62, no. 5, pp. 2179–2190, 2020.

[18]. L. Enrique054, "Real-world IoT data for environmental analysis IOT sensor DHT for temperature, humidity and heat index." 2023.

[19]. L. Enrique054, "Real-world IoT data for environmental analysis IOT sensor DHT for temperature, humidity and heat index." 2020.

[20]. J. Liu, Y. Cao, Y. Li, Y. Guo, and W. Deng, "A big data cleaning method based on improved CLOF and Random Forest for distribution network," CSEE J. Power Energy Syst., 2020.

[21]. V. R. Joseph, "Optimal ratio for data splitting," Stat. Anal. Data Min. ASA Data Sci. J., vol. 15, no. 4, pp. 531–538, 2022.

[22]. L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu, and L. Shao, "Normalization techniques in training dnns: Methodology, analysis and application," IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 8, pp. 10173–10196, 2023.

[23]. Q. T. Phan, Y. K. Wu, and Q. D. Phan, "A hybrid wind power forecasting model with XGBoost, data preprocessing considering different NWPs," Appl. Sci., vol. [13], no. 3, p. [13]00, 2021.

[24]. J. Cao et al., "Do-conv: Depthwise over-parameterized convolutional layer," IEEE Trans. Image Process., vol. 31, pp. 3726–3736, 2022.

[25]. Y. Cao, Y. Ding, M. Jia, and R. Tian, "A novel temporal convolutional network with residual self-attention mechanism for remaining useful life prediction of rolling bearings," Reliab. Eng. Syst. Saf., vol. 215, p. 107813, 2021.

[26]. M. Tian, H. Dong, X. Cao, and K. Yu, "Temporal convolution network with a dual attention mechanism for φ-OTDR event classification," Appl. Opt., vol. 61, no. 20, pp. 5951–5956, 2022.

[27]. P. Ma, Y. Wang, J. Shen, S. Petridis, and M. Pantic, "Lip-reading with densely connected temporal convolutional networks," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 2857–2866.

[28]. J. Zhu, L. Su, and Y. Li, "Wind power forecasting based on new hybrid model with TCN residual modification," Energy AI, vol. 10, p. 100199, 2022.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[29]. D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," Peerj Comput. Sci., vol. 7, p. e623, 2021.
[30]. R. Zhang, F. Sun, Z. Song, X. Wang, Y. Du, and S. Dong, "Short-Term Traffic Flow Forecasting Model Based on GA-TCN," J. Adv. Transp., vol. 2021, no. 1, p. 1338607, 2021.
[31]. M. Ge, N. F. Syed, X. Fu, Z. Baig, and A. Robles-Kelly, "Towards a deep learning-driv intrusion detection approach for Internet of Things," Comput. Netw., vol. 186, p. 107784, 2021.
[32]. A. Nazir et al., "A deep learning-based novel hybrid CNN-LSTM architecture for efficient detection of threats in the IoT ecosystem," Ain Shams Eng. J., vol. 15, no. 7, p. 102777, 2024.
[33]. D. Javeed, T. Gao, M. T. Khan, and I. Ahmad, "A hybrid deep learning-driven SDN enabled mechanism for secure communication in Internet of Things (IoT)," Sensors, vol. 21, no. 14, p. 4884, 2021.
[34]. X. Zhang, M. Zhao, and R. Dong, "Time-series prediction of environmental noise for urban IoT based on long short-term memory recurrent neural network," Appl. Sci., vol. 10, no. 3, p. [13]44, 2020.
[35]. J. Zhang, Y. Zeng, and B. Starly, "Recurrent neural networks with long term temporal dependencies in machine tool wear diagnosis and prognosis," SN Appl. Sci., vol. 3, no. 4, p. 442, 2021.
[36]. Z. Liu, Y. Wang, L. Li, X. Fang, and J. Wang, "Realtime prediction of hard rock TBM advance rate using temporal convolutional network (TCN) with tunnel construction big data," Front. Struct. Civ. Eng., vol. 16, no. 4, pp. 401–413, 2022.
[37]. H. M. Elfadil, H. F. H. Hamed, O. Abdulhamid, and F. Embarak, "Enhancing Robotic Performance and Adaptability through Advanced Predictive Models with Deep Reinforcement Learning," Int. J. Innov. Res. inComputer Commun. Eng., vol. 13, no. 1, p. 966, 2025, Accessed: Mar. 18, 2025.
[38]. O. E. Aboualqassim, M. Osman, and F. S. A. Elghaffi, "Utilizing a Hybrid CNN-Autoencoder Model for Efficient Feature Extraction in Medical Image Analysis and Alzheimer's Disease Diagnosis," Int. J. Innov. Res. Comput. Commun. Eng., vol. 13, no. 1, p. 951, Jan. 2025, Accessed: Mar. 18, 2025.
[39]. Asmaa, Aljilany, A.-H. Marwa, A. Alaa, E. Farhat, and E. Abdulhamid, "Cultivating Independent Learners: Designing an Innovative Interactive Model to Enhance Self-Directed Learning and Autonomy in Elementary Education," International Journal of Innovative Research in Science Engineering and Technology. Accessed: Jan. 28, 2025
[40]. J. Fan, K. Zhang, Y. Huang, Y. Zhu, and B. Chen, "Parallel spatio-temporal attention-based TCN for multivariate time series prediction," Neural Comput. Appl., vol. 35, no. 18, pp. 13109–13[13]8, 2023.