International Research Journal of Multidisciplinary Scope (IRJMS), 2024; 5(3):706-720

Original Article | ISSN (O): 2582-631X DOI: 10.47857/irjms.2024.v05i03.0829

IoT Generated Multi-Modality Data Analysis Using a Deep Learning Framework for Managing Sustainability in Smart Environments

Doraswamy B[1]*, Lokesh Krishna K[2], Hariprasad Tarigonda[3]

[1]Department of Electrical & Computer Engineering, College of Engineering & Technology, Bule Hora University, Ethiopia, [2]Department of Electronics & Communication Engineering, Sri Venkateswara College of Engineering, Tirupati, Andhra Pradesh, India, [3]Department of Mechanical Engineering, Mohan Babu University, Tirupati, Andhra Pradesh, India. *Corresponding Author's Email: birasalapatid@gmail.com

Abstract

Incorporating the Internet of Things (IoT) systems into Smart environments can reinforce stability and sustainability. IoT-deployed smart environment monitoring devices supports energy monitoring, water consumption, and other resource utilization details. Several earlier research works have focused on energy, water consumption, or other sustainable parameter monitoring that cannot retain and manage sustainability. This paper aims to maintain and manage the sustainability of smart environments by creating a Deep Learning Framework (DLF) to analyze the multi-modality data generated by IoT devices. The DLF integrates and incorporates an IoT dashboard, IoT network, Deep and Reinforcement Learning algorithms, Optimization algorithm, and Smart Regulator. The proposed DLF monitors multiple smart environments using IoT devices that generate multi-modality data. The IoT devices are clustered as networks that monitor different smart environments, generate data, and send it to the cloud data centers. Before that, the Particle Swarm Optimization algorithm is used to preprocess and optimize the data. Based on the data modality, deep learning algorithms are activated automatically to process and predict the conditions of the smart environment concerning abnormalities and utility consumptions. Smart metering, building, fleet, and air quality monitoring and predictions are some of the additional solutions given by the DLF. It incorporates cloud and IoT dashboards to improve the overall monitoring performance and reduce cost efficiency. The performance of the DLF is verified by simulating and comparing its output with the other state-of-the-art methods, and it is found that the proposed DLF outperforms the others.

Keywords: Deep Learning Algorithms, IoT Data Processing, IoT-Monitoring, Reinforcement Learning, Smart Environment, Sustainable Development.

Introduction

Most smart environment applications recently require automation to control and manage their operations. It helps to reduce energy consumption, time complexity, and other environmental impacts. The word IoT (Internet of Things) means a collection of devices that perform their task through the Internet. These devices communicate among themselves and other devices, servers, administrators, and other cloud-based resources through the Internet. All the devices in the IoT are connected as a network. IoT-used smart buildings improve residents' quality of life, reducing costs with increased efficiency. This paper concentrates on managing sustainability using IoT networks for multiple smart environments.

Some of the advantages of using IoT are improved efficiency, data-driven decision-making, cost savings, and enhanced customer experience. Because of these advantages, IoT is used in healthcare, manufacturing, retail, agriculture, and transportation applications; among them, one of the critical applications is the smart environment. An IoT-incorporated smart environment comprises sensors, displays, and other small-scale computational devices that monitor and collect data, which helps to understand their positive and negative impacts of the backdrop for better optimization. The final solution of the smart environment is obtained by integrating multiple solutions for multiple criteria—for example, IoT-based water management, waste management, energy management, and others. IoT deployment provides a platform for future enhancements and optimizations that further improve the operating

(Received 24th March 2024; Accepted 12th July 2024; Published 30th July 2024)

efficiency and sustainability of the environment. These smart solutions include light and water-flow on/off management based on the usage pattern, watering lawns and gardens, managing vehicle parking, and maintaining room temperature. It also extends to applications like smart ports, beaches, buildings, and industries. These applications make use of IoT for their optimization. Various applications for smart environments include agriculture, healthcare, Industries, and houses. As IoT is the sole strong point of the smart environment, the associated risks and challenges in the IoT technology implementation need to be assessed. IoT has numerous risks, including lack of security and privacy, interoperability, data overload, cost and complexity, and regulatory and legal issues. However, they still have the potential if the following measures are considered: planning IoT strategy, choosing secure IoT products, monitoring the devices, managing data effectively, and building a secure ecosystem. With future developments, by adopting different technologies like edge computing, Artificial Intelligence and Machine Learning, Blockchain, and sustainable technologies, the performance of IoT networks needs to be further improved.

The proposed model uses multiple deep learning models for processing the multi-modality data obtained from different smart environments to optimize data and train the models. The PSO algorithm gets the data optimization, which helps reduce the computational and time complexity. It makes the training process of the deep learning models easier. Data optimization pre-processes the data by eliminating errors, duplication, and missing values to make them uniform. The data obtained from different environments are used to train the proposed deep learning models, and the trained output is used for testing the data using a separate deep learning model. The model's performance is compared with other existing models. It is a novel approach towards a heterogeneous generalized optimization model for different kinds of smart environments.

Several earlier research works related to data collection from the smart environment is studied and analysed including their merits and demerits. The authors have presented a study to define how IoT-based devices are incorporated with various real-time applications in a smart environment. The uses of sensor networks and RFID technology in smart environments are discussed in detail. The result of the study indicates that the IoT-based model provided better results regarding monitoring the environment (1). However, it required an additional technique to manage the security system. The design of a 3-dimensional user interface model for processing the data of an IoT-based smart environment system is described in previous study (2). This model will link the user and the physical object by presenting a virtual representation, and the user will get a clear identification of devices within the environment. It will be more beneficial for the user to connect with the nearby device in less time. Though this model performs better, it requires an updated 3D model to detect the changes in the internal space of the model. An automated smart agriculture monitoring system using IoT is proposed in past research (3). The major focus of this research model is to improve the production of the crops by accurately monitoring the crop-yielding area and weather. The attributes of the smart agriculture environment are predicted through the proposed model and shared with the farmers through MMS. This research shows that this model is more useful for farmers to understand how to cultivate the right crops at the right time. Some of the articles from the year 2011 to 2019 were reviewed, where they focused on improving the efficiency of IoT and big data-based models. The reviews concluded that the IoT and big data-based model is more beneficial to incorporate with the applications in the smart environment (4).

The authors have researched to define the usage of IoT-based applications in smart grids and environments (5). It indicates that IoT applications are more suitable for improving the energy internet's stability and efficiency. The role of IoT and machine learning-based models in smart environments is discussed in previous research (6). The issues and challenges of implementing an ML-based system are concerned with the efficiency of the ML and IoT models, and the performance is compared with various existing studies. Finally, the author concluded that data security and privacy are the leading challenges in smart environments. The authors presented a study demonstrating the state-of-the-art machine learning (ML) based applications in smart cities (7). The outcomes from this work emphasize that five ML-based models, such as SVM, ANN, DT, Bayesian hybrids, and neuro-fuzzy, are widely applied techniques in smart cities to address the issues. This study's overall analysis explained that most of the research works have discussed only the deep learning models from 2018 and 2019 compared to ML-based models. The complications encountered in the implementation of IoT technologies in the problem-solving and continuous monitoring processes are elaborately discussed in previous study (8). The authors proposed an automatic detailed diagnosis approach to address these issues. The proposed model was non-intrusive, easy to use, low-cost, and flexible. The ADD framework uses low-power devices and sensors to optimize energy consumption, enable continuous monitoring, and improve problem-solving capabilities (9). Some of the critical issues in the IoT networks in smart cities are discussed, and an analytical framework that uses big data analytics to improve environmental sustainability is proposed in this work. A novel IoT framework for supply chain management in agricultural industries is presented in past study (10). Multiple performance factors, like Shannon entropy, were estimated for weighing the SCPM during the data analysis. The fuzzy TOPSIS model was used for ranking the key performance factors. It is concluded that flexibility and responsiveness are the key indicators of the SCPM framework for obtaining sustainability.

The authors discussed the importance of IoT usage for smart monitoring in various applications in recent years (11). During the COVID-19 pandemic, there was an improvement in the ecosystem. The release of CO_2 emissions in the atmosphere forms a large amount of energy consumption and affects the environment. The author has aimed to classify the sustainability of cloud/edge/IoT/fog ecosystems with the help of AI techniques and found that they are efficient in optimizing the cloud, edge, IoT, and fog computing models for improving sustainability. The implementation of different digital technologies using IoT for manufacturing servitization and industries is described in past research (12). Also, the author has designed to digitalize the business model network and innovation impact in manufacturing servitization sustainability. The authors discuss the solution for the waste management (13). Dumping massive amounts of waste products provides toxic gases that make the environment and human unhealth. IoT monitoring is used to collect garbage information, classify waste, and control the odor of gases. The IoT data is analyzed using deep learning algorithms to provide an artificial ecosystem to increase sustainability. This method, known as AEOIDL-SWM, contains a camera and sensor to collect information and classify. The AEOIDL-SWM enhances the outcome analysis, which is better than the deep learning models. The federated learning technology integrated with IoT for healthcare applications is discussed in previous study (14). The FL-IoT framework monitors and analyzes healthcare data using deep learning algorithms. This framework helps to address the local training data to detect skin diseases and personalize the data without sharing, and the DFL model does this process. The improved model of IoT devices, known as the area under the curve (AUC), has a higher accuracy of 99%.

From the literature survey, it is noticed that the adoption of IoT in surveillance monitoring applications improves overall efficiency regarding sustainability and energy. Several research works and applications proposed in earlier days used in smart environments provided scalable architectures for optimizing and improving the sustainability of the environment. Creating an efficient, sustainable environment is challenging using the available technology and renewable resources. Some earlier research works have proposed optimization algorithms for resource utilization and communication, but their efficiency is poor. Few works used machine learning and deep learning algorithms to optimize the IoT-based sustainable environment for effective data transmission. These algorithms learn the energy consumption patterns of the devices and optimize the power consumption of the devices alone. Some of the earlier work have proposed regression algorithms for processing time-series data. Among various models and methods used, ensemble algorithms provided better performance regarding energy saving. They were also used to optimize the networks to improve energy-based sustainability. Sustainability is enhanced in various ways, such as providing continuous energy, water, waste management, and Internet connection. The sustainability providers need a complete solution for multiple environments in a unique form. The proposed work is aimed to implement multiple deep-learning models to analyze and predict the abnormalities that occur in the environment. The sustainability efficiency is increased by optimizing the network structure by optimum node placement, clustering, and routing among the nodes in the network.

The rest of the paper is organized as follows: Section-2 discusses about the proposed reinforcement-based deep learning model, while Section-3 provides the simulated results and its discussion. Lastly Section-4 presents the conclusions of the proposed work.

Materials and Methods

The problem statement of the proposed work is written as follows: Let us consider $'N'$ number of Smart Environments (SE) interconnected into a Smart Environment Network (SEN),

$$SEN = \{SE_1, SE_2, \dots SE_i, \dots, SE_N\} \; \forall \, i = 1 \; to \; N \quad [1]$$

Deployed and monitored with M IoT-Devices (IoTD)

$$IoTD = \{IoTD_1, IoTD_2, \dots IoTD_j, \dots, IoTD_M\} \; \forall j = 1 \; to \; M \quad [2]$$

The IoTD of each SE belongs to the IoT-Network (IoTN) and is represented as:

$$IoTN = \{IoTN_1, IoTN_2, \dots IoTN_k, \dots, IoTN_n\} \; \forall \; k = 1 \; to \; n \quad [3]$$

where, the device $IoTD_j$ deployed in the network $IoTN_k$ is denoted as $IoTD_{jk}$. The data d^i_{jk} represents the surveillance data generated from j^{th} IoT device of k^{th} network deployed in i^{th} environment. Figure 1 shows the proposed Smart Environment Network (SEN) model.

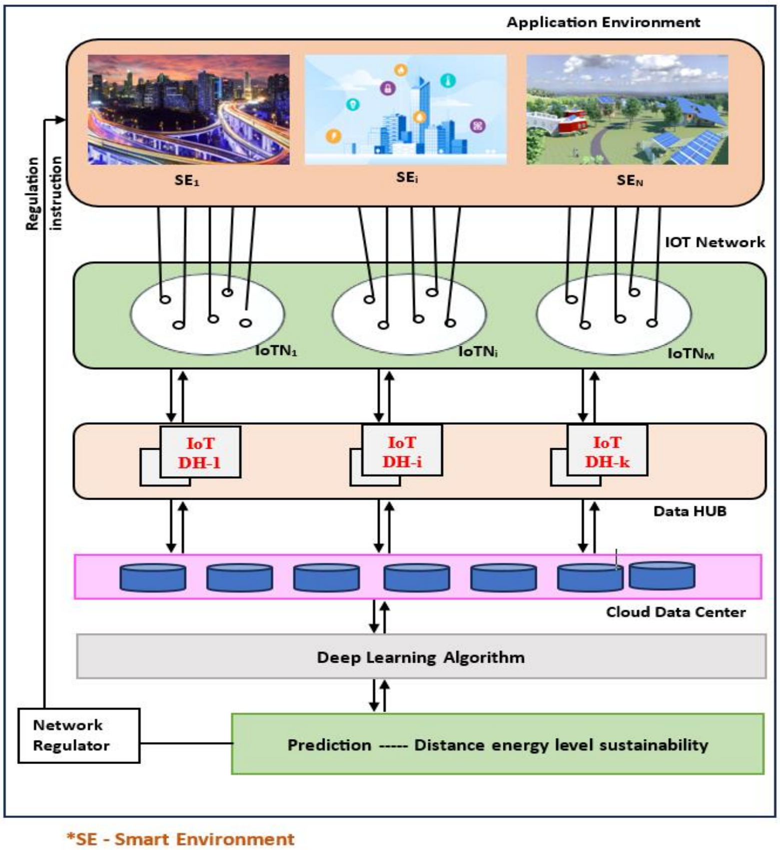

Figure1: Proposed IoT Deployed Smart Environment Network Model

The environmental monitoring and data generation process is carried out continuously. The data is generated from the entire IoTN for a fixed interval T defined as

$$d^i_{jk} = \int_{t=0}^{T} d_t \; \text{-----------}[4]$$

The data generated from three different SEs are represented as:

$$D_1 = \sum_{i=1, j=1, k=1, t=0}^{j=M, k=n, t=T} d^{ki}_{jt} \; \text{--------}[5]$$

$$D_2 = \sum_{i=2, j=1, k=1, t=0}^{j=M, k=n, t=T} d^{ki}_{jt} \; \text{--------------}[6]$$

$$D_3 = \sum_{i=3, j=1, k=1, t=0}^{j=M, k=n, t=T} d^{ki}_{jt} \; \text{--------------}[7]$$

The individual data is fed separately to different LSTMs, such as MVLSTM, UVLSTM, and MVCNN-LSTM, for predicting the abnormalities. Based on the anomaly, the network regulator is activated and performs the task through which the sustainability is maintained. Additionally, the data indicates the

IoT device, network, time, and smart environment, which helps the management to rectify the abnormalities manually to provide fast and efficient sustainability. The individual SE and IoTD are connected to a network based on the contour integral theorem.

Contour Integral Theorem

This paper uses the contour integral theorem for interconnecting all the SEs into a single SE, and it is considered a network. For example, Smart Healthcare (SE1), Smart City (SE2), and Smart Agriculture (SE3) are integrated into the Smart Environment Network (SEN), as shown in Figure 2. The region R is covered by a curve C and multiple sub-regions called multi-connected regions, as shown in Figure 2. Let us assume that the contour C is a curve that covers all the SEs inside the SEN. An integral function for a single region along the x-axis from a starting point a and ending point b is written as

$$\int_a^b \mathrm{f}(x)dx = -\int_a^b \mathrm{f}(x)dx \text{ // for clockwise and anti-clockwise}$$

For multiple regions of curve C is

$$\int_C f(x)dx = \int_{C_1} f(x1)dx + \int_{C_2} f(x2)dx + \int_{C_3} f(x3)dx + \cdots. \text{ // data from multiple SEs}$$

$$f(z) = u(x,y) + iv(x,y) \text{ --------------[8]}$$

$$\int_C f(z)dz = \int_C (u+iv)(dx+idy) = \int_C (udx - vdy) + i\int_C (udy + vdx) \text{ ---------- [9]}$$

where, the data D is $\int_C f(z)dz$. Since the data generated from the SE network is multi-modal, multiple deep learning algorithms are incorporated to analyze and predict the conditions.

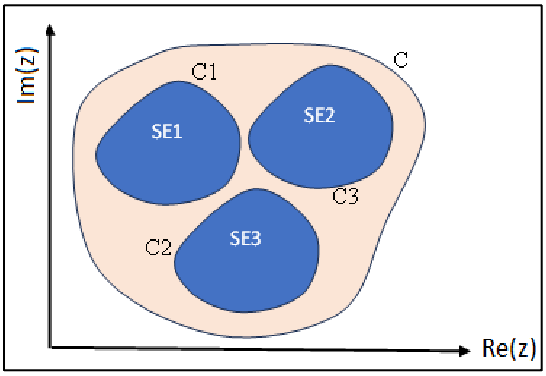

Figure 2: Multi-Region Integration for Smart Environment Network

Proof: the sum of data obtained from the sub-regions covered by curves C_1, C_2, and C_3 is the same as the SEN ($C = C_1 + C_2 + C_3$).

A novel DLF to improve sustainability efficiency concerning various environmental factors is proposed in this paper. Multiple smart environments, such as smart agriculture, healthcare, and buildings, are considered here to generate multi-modality data and verify the DLF's efficiency. Each smart environment is deployed with a separate IoT network with many IoT devices interconnected in the network through the Internet. Smart environments include smart cities, buildings, agriculture, healthcare, and manufacturing. They are linked with smart devices and computers for daily tasks and settings. Smart homes are created and then moved to create smart buildings, cities, and environments. Smart Environment Network (SEN) integrated with an IoT Network (IoTNet) where the IoTNet monitors the entire SEN round the clock, generates the data and is passed to the data hub. Before sending to the data hub, data D is considered as a time series data and mapped with time, t_i.

However, in recent times, to improve energy efficiency and optimal resource sharing and utilization, multiple smart environments are interlinked as an intelligent network, where the network size is increase at any time. The data analysis process in the IoT-deployed smart environment network is illustrated in Figure 3. The PSO algorithm optimizes the data generated from all IoT devices deployed in the IoT network. During the optimization, the PSO clears the data error and missing values, normalizes in terms of data type and size, and provides the necessary security to the overall model.

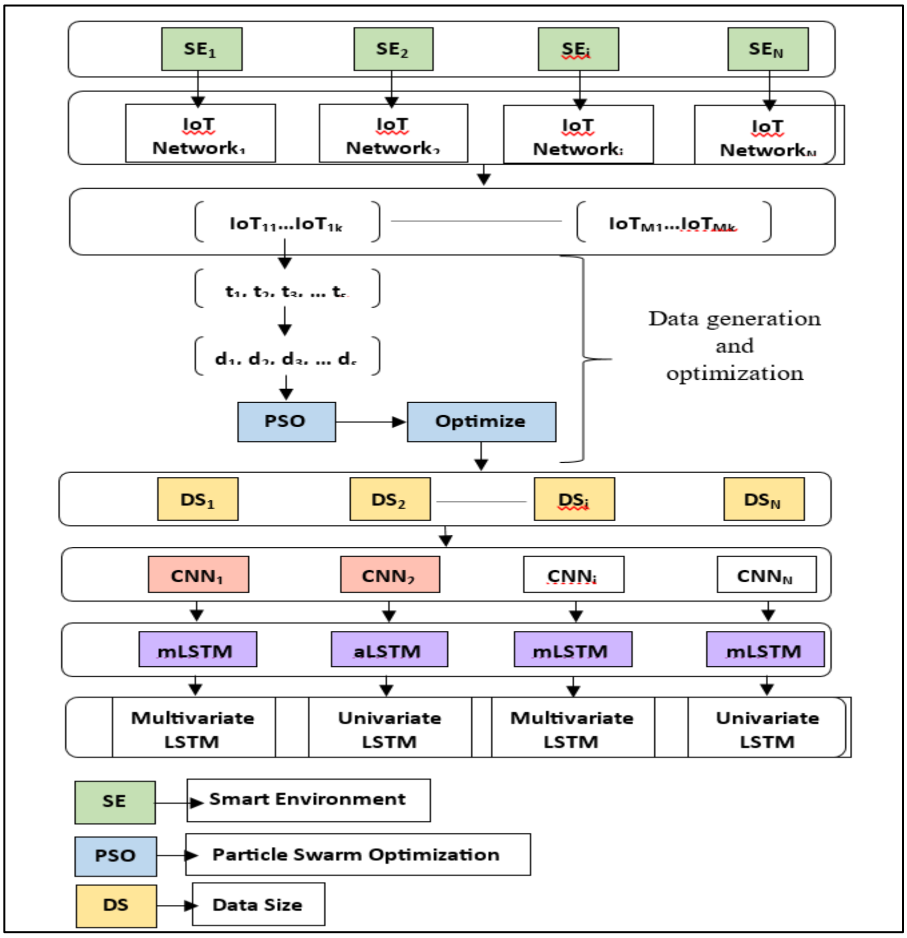

Figure 3: Data Preparation

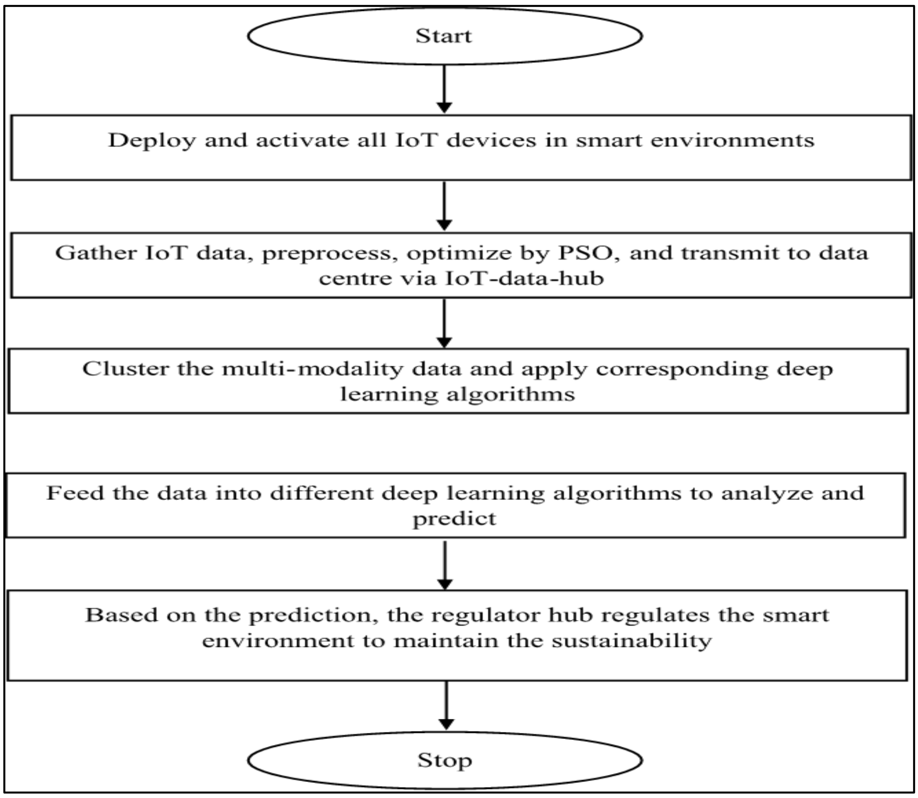

Figure 4: Data Flow Diagram

The network nodes are selected optimally and do the best settings in the environment for optimized data routing. The flow-chart for processing the data is presented in Figure 4. All the nodes and the environments with all the parameters are chosen manually before creating the proposed network model. A smart environment can represent any building, from home to commercial ones, with different devices for different optimization. As the data obtained from multiple devices are heterogeneous, the data must be pre-processed and normalized.

Feature Extraction

The features of the data are extracted to learn their dynamic behavior. The feature extraction reduces data redundancy and considers only the main variables corresponding to the features. It reduces space, increases processing speed, and also avoids overfitting. Some widely considered methods, like the Pearson correlation coefficient, are used for the feature extraction. Feature values are the specific values that can index the overall performance of an object or the entire dataset.

Pearson Correlation Coefficient

It compares two variables regarding covariance and standard deviation and obtains its quotient, which helps identify their relation. It ranges between$[-1,1]$, and it is denoted as cor, which can be written as follows,

$$Cor = \frac{COV(X,Y)}{\sigma_X \sigma_Y} = \frac{E\left((X-\mu_X)(Y-\mu_Y)\right)}{\sigma_X \sigma_Y} \text{--------------[10]}$$

The cor represents the correlation between the variables that scales linearly. If the value of $cor =$ 0, then it shows the non-linearity of the variables. However, there are different types of correlation, like curve, linear, non-linear, and no correlation. The potentiality and generalization of the proposed model differs based on the variable's correlation. The X and Y represent the random variables, and their mean and variance are represented as μ and %sigma. A correlation of nearly 0.95 is obtained for a 12 set of features. The dataset consists of 9 essential features.

Data Pre-processing

The data pre-processing reduces redundancy, variations, outliers, and missing values. As the data is obtained from different devices, different data is obtained of different measures and needs to be processed with equitable contribution. In the preprocessing, normalization is adopted to improve a unique set of values. The algorithm must process the raw data to extract and classify features based on the applications. People consider all types of home appliances, monitoring devices, and automation devices for a smart environment. The data obtained from SEN is optimized based on their functioning and the necessity to automate and improve sustainability. Based on the statistical properties, the raw data is normalized and retains the data's essential characteristics. It is used to fit in more information.

$$A' = \left(\frac{A-minvalueofA}{maxvalueofA-minvalueofA}\right) * (B - C) + C \text{--------------[11]}$$

A is the real data, the B is the bias, and the C is the correction based on the map. Parametric normalization is applied to unstructured data through mean and standard deviation. It is usually done through the **z**-score variable, as follows,

$$f'_i = \frac{f_i - \bar{y}}{std(Y)} \text{--------------[12]}$$

The normalized values obtained are represented as f'_i, where f_i is the row rate Y for the ith column. The standard deviation the Y is calculated as follows,

$$Std(Y) = \sqrt{\frac{1}{(m-1)} \sum_{i=1}^{m} (f_i - \bar{Y})^2} \text{--------------[13]}$$

$$\bar{Y} = \frac{1}{m} \sum_{i=1}^{m} f_i \text{--------------[14]}$$

It normalizes each row, and it can lead to situations where all the values in the rows are the same that correspond to 0. The normalization converts the values obtained between 0 and 1, and the decimal values range between $\mathbf{-1}$ and $\mathbf{1}$. The same strategy can be written as follows,

$$f_i = \frac{f}{10^q} \text{--------------[15]}$$

f_i is the scaled values, the value range is f, and the smallest integer q , which is $Max(|f_i|) < 1$.

Particle Swarm Optimization (PSO)

PSO is an intelligent algorithm mainly used to find the optimal solution from the group of values. Speed and position are the independent variables, and density is considered to be the function value. Based on the search history, the direction and speed of the search are adjusted by the PSO model. It analyzes the difference between the population and optimal search history. Finally, all the swarm data are gathered in the same position, and based on this data location, the model's final optimal solution is identified. The main advantage of implementing PSO algorithms is that they can process the input data more efficiently with high computational speed than traditional models. It is more suitable to work with a large set of input data. During the optimization process, a gradient function is not required to calculate the gradient information of the input value. The structure of the PSO algorithm can update the particle's position and velocity repeatedly.

In most applications, the PSO algorithm is used as an optimization model to optimize the objective function. In d-dimensional space, the n and ith particles contain population P and position and velocity vector x and V, respectively. Then, using the fitness function, the fitness of the variable is evaluated. It is achieved by assessing the model's precision, recall, accuracy, F-measure, and size of the features with the total number of features in the input. The fitness values of the input data are evaluated using the equation [16].

$$F(X) = \alpha(1 - P) + (1 - \alpha)\left(1 - \frac{N_f}{N_t}\right) \text{--------[16]}$$

The optimal solution is identified after performing the different iterations in the N-dimensional space. The continuous particle search identifies the local optimal solution for the position and velocity. This optimal solution is considered the local optimal solution of the swarm model. At each time of iteration, the position and velocity of the function are updated to produce the final global optimal solution. Using equations [17] and [18], the position and velocity of the model are updated.

$$v_{id}(t+1) = wv_{id}(t) + c_1r_1\big(p_{id}(t) - x_{id}(t)\big) + c_2r_2\left(p_{gd}(t) - x_{id}(t)\right) \text{----------[17]}$$

$$x_{id}(t+1) = x(t) + v_{id}(t+1), \forall i = 1,2,\ldots.,N; \forall d = 1,2,\ldots,D \text{ ---------[18]}$$

The above equations $n, d, \omega, \wedge t$ represent the total number of particles, particle dimension, the local and global optimal solution capability, and number of iterations, respectively. If the ability of the global optimization is more robust, then the power of the local optimization becomes weaker and vice versa. It occurs when the generated output value is more prominent. In equations [17] and [18], $C_1C_2 \wedge r_1r_2$ represent the acceleration factor and random number. The value of the acceleration factor is $C_1 = C_2 = 2$, and the random number value is either 0 or 1; using these values, the particle's position and velocity are knowingly updated and optimized for the features of the input data. Only the optimized dataset is prepared to process from the core data generated, and the other data are eliminated due to preprocessing and normalization processes. The prepared dataset is fed into the corresponding deep learning algorithm, such as LSTM-1, LSTM-2, or LSTM-3, based on the data types, such as time series, image, video, and other alphanumeric data.

Long-Short-Term Memory Model

It is a type of recurrent neural network where the output is fed as input over every iteration to improve the model's accuracy. However, the RNN only compared the outcome with the short-term dependencies, which resulted in much less precision. LSTM is an algorithm with long-term dependencies to tackle this issue. It is widely used in the classification of time series data. It is widely used to process sequential data, for example, time, text, and speech. It identifies the long-term dependencies by holding the past data in memory and compares them. It is used in the voice to text conversion, NLP, and translation. The proposed LSTM model comprises three different gates: input, output, and forget. The Z and D represent the hidden and cell state, and the(X_t, D_{t-1}, Z_{t-1}) and (Z_t, D_t).

The time t for the input, output, and forget gate are represented, O_t, I_t, F_t. The data is sorted using, F_t, the sorted data represented the features that follow the gaze direction,

$$F_t = \sigma(J_{IF}X_t + L_{IF} + J_{ZF}Z_{t-1} + L_{ZF}) \text{ ---------[19]}$$

The equation [19], (J_{IF}, L_{IF}) and (J_{ZF}, L_{ZF}), represents the weights and constraints in the bias of the hidden and input layers mapped to the forget gate. The activation function is represented as σ. The (J_{ZG}, L_{ZG}) and (J_{IG}, L_{IG}) are mapped to the cell gate, and the (J_{ZI}, L_{ZI}) and (J_{II}, L_{II}) are mapped to the input layers.

$$G_t = tanh(J_{IG}X_t + L_{IG} + J_{ZG}Z_{t-1} + L_{ZG}) \text{-----[20]}$$

$$I_t = \sigma(J_{II}X_t + L_{II} + J_{ZI}Z_{t-1} + L_{ZI}) \text{ ---------[21]}$$

$$D_t = F_tD_{t-1} + I_tG_t \text{ ---------[22]}$$

$$O_t = \sigma(J_{IO}X_t + L_{IO} + J_{ZO}Z_{t-1} + L_{ZO}) \text{ ---------[23]}$$

$$Z_t = O_t tanh(D_t) \text{ ---------[24]}$$

The weights and bias corresponding to the output hidden and output gates are shown in O_t and Z_t. The weights and bias mapped to the hidden layer and input layer, (J_{ZO}, L_{ZO}) and (J_{IO}, L_{IO}).

Results and Discussion

Multiple LSTM Model for Different Environments

As the hybrid model consisting of multiple LSTM modules needs to process different types of data collected from different monitoring systems, the preprocessing step is carried out to reduce and organize the data without redundancy. The data consists of information collected from multiple devices, which vary according to the type of application considered for the process.

LSTM Model for Healthcare Data (LSTM-1)

The healthcare data consists of input from different medical devices that monitor the patient's health over longer durations. It is an encoder-decoder model in which the encoder reads the data and encodes them for the decoder to read and make a one-step prediction. The difference between the normal LSTM and LSTM-1 is that it accepts univariate input. While using the LSTM decoder, the data predicted for the prior days are also considered. The LSTM-1 comprises 5 layers, with two LSTM layers. Each LSTM layer comprises 200 units. Table-1 lists the parameters related to the layer architecture of LSTM-1.

Healthcare monitoring data for 14 days are considered as input (15). It is a simple LSTM auto encoder model where the LSTM layer is followed by a repeat vector 10. It defines the time step for each output sequence. It is used to present the series of vectors to the LSTM decoder. The second LSTM hidden layer comprises 200 units, where the decoder gives the entire sequence. The 200 units output a sequence representing the whole data of the 14 days, predicting the case for that specific day. Time-distributed dense layers with 100 units follow the LSTM layer, and the same layer is repeated and obtained as the output layer. The number of neurons in the output layer is defined based on the number of output predictions. The dense layers help to determine each step separately. It can also be replaced by flattening the output obtained from the LSTM decoder.

Table 1: Layer Architecture of LSTM-1

Layer (type)	Output Shape	Param #
lstm_1 (LSTM)	(None, 200)	168800
repeat_vector (RepeatVector)	(None, 10, 200)	0
lstm_2 (LSTM)	(None, 10, 200)	320800
time_distributed (TimeDistributed)	(None, 10, 100)	20100
time_distributed_1 (TimeDistributed)	(None, 10, 1)	101
Total params: 509801 (1.94 MB)		
Trainable params: 509801 (1.94 MB)		
Non-trainable params: 0 (0.00 Byte)		

LSTM Model for Smart City (LSTM-2)

This is an update from the prior LSTM, with LSTM-2 accepting multivariate input obtained from the smart cities (16). The smart building monitoring data is a one-dimensional time series data given as multivariate input. The multivariate input helps the encoder and decoder provide the output sequences as a function with multiple different features. In the smart building, several monitoring purposes need to be considered, like assessing the pattern of the people. Table-2 lists the parameters related to the layer architecture of LSTM-2. In this model, the multivariate input is converted to a flattened model, where the data is converted into one-dimensional, usually the available form. A CNN algorithm is combined with the LSTM for processing this data called the Conv-LSTM, which helps process the spatiotemporal data.

Table 2: Convolution-LSTM-2: Layer Architecture

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 8, 64)	1984
conv1d_1 (Conv1D)	(None, 6, 64)	12352
max_pooling1d (MaxPooling1D)	(None, 3, 64)	0
flatten_1 (Flatten)	(None, 192)	0
repeat_vector_3 (RepeatVector)	(None, 10, 192)	0
lstm_5 (LSTM)	(None, 10, 200)	314400
time_distributed_6 (TimeDistributed)	(None, 10, 100)	20100
time_distributed_7 (TimeDistributed)	(None, 10, 1)	101

Total params: 348937 (1.33 MB)
Trainable params: 348937 (1.33 MB)
Non-trainable params: 0 (0.00 Byte)

LSTM Model for Agriculture LSTM-3

The multiple data observed from the smart agricultural field are analyzed using the LSTM-3 model. The input data are collected from various agricultural fields through IoT devices like cameras, drones, etc, and fed into the proposed LSTM model. Table-3 lists the parameters related to the layer architecture of LSTM-3. The LSTM-3 has 5 layers, to process the input crop field data. Each layer in the model has unique functions to generate the final prediction result. At first, the input data are fed into the convolutional layer to map the features and transfer them into the flattened layer. The flattened layer converts the dimensional of the raw input data to improve the data quality. Likewise, the repeat vector layer creates or adds a new dimension to the input data. The time distribution layer distributes the batch size of the data into sequential lengths. It distributes equally to the output value for each input dataset sample (16).

Table 3: LSTM-3 Layer Architecture

Layer (type)	Output Shape	Param
conv_lstm2d (ConvLSTM2D)	(None, 1, 8, 64)	57088
flatten (Flatten)	(None, 512)	0
repeat_vector_7 (RepeatVector)	(None, 10, 512)	0
lstm_14 (LSTM)	(None, 10, 200)	570400
time_distributed_14 (TimeDistributed)	(None, 10, 100)	20100
time_distributed_15 (TimeDistributed)	(None, 10, 1)	101
Total params: 647689 (2.47 MB)		
Trainable params: 647689 (2.47 MB)		
Non-trainable params: 0 (0.00 Byte)		

The SEN is created by interconnecting smart buildings, agriculture, and industries. The IoTNet deployed over the SEN monitors and generates continuous data, processed by deep learning algorithms, to automate the process of SEN. The proposed deep learning model analyzes the data from IoTNet and tunes the regulator, which regulates the SEN based on the outcome (17). Finally, the performance of the DLF is evaluated by calculating and comparing various performance metrics. Multiple sensors are assumed to be deployed in the IoTNet, where their data is processed separately to evaluate the system's performance.

The dataset used in the experiment is taken from multiple smart environment application used in the earlier research works. The first data set is taken from 255 sensors equipped in 51 rooms in a 4 floor building (18). The physical, static, and dynamic behaviours of the data patterns are obtained for experimenting the IoT deployed with sensors. The dataset was experimented by supervised and unsupervised learning models and results were verified. The WiDS Datathon-2022 dataset is used in the experiment (19). The dataset is obtained by experimenting the IoTNet 100 times in a smart building and agriculture. Both the dataset mixed together and 80% of the data is taken for training the DLF and remaining data is used for testing and validation process. After preprocessing, 93% of the entire data is considered for the experiment compared to the earlier models. Figure-5(a), Figure-5(b), Figure-5 (c), Figure-5 (d), Figure-5 (e), and Figure-5 (f) represent the efficiency of the IoT-based model in monitoring soil moisture, light sensor, air quality, temperature, humidity, and atmospheric pressure in the smart agriculture field. Based on this evaluation, changes in every crop field are easily monitored and provide more benefits to the farmer to increase crop production. Different sensors inserted in the IoTNets are enabled to monitor the environments and generate the data. Each sensor's data is continuously recorded and verified for five days. The data obtained from the light sensors are much less during the day and increased at night shown in Figure-5(a). Soil moisture sensor data recorded the data only during the day, not at night shown in

Figure-5(b). At the same time, the environment's air quality is the same for the whole time. The air quality is assessed by various pollutant levels obtained from different gas sensors deployed with the IoTNet in Figure-5(c). The temperature data obtained from the SEN is similar for four continuous days and slightly increased on the last day shown in Figure-5(d). D-proportional to the temperature, the humidity value recorded from the SEN increases in the first four days and decreases on the fifth day shown in Figure-5(e).

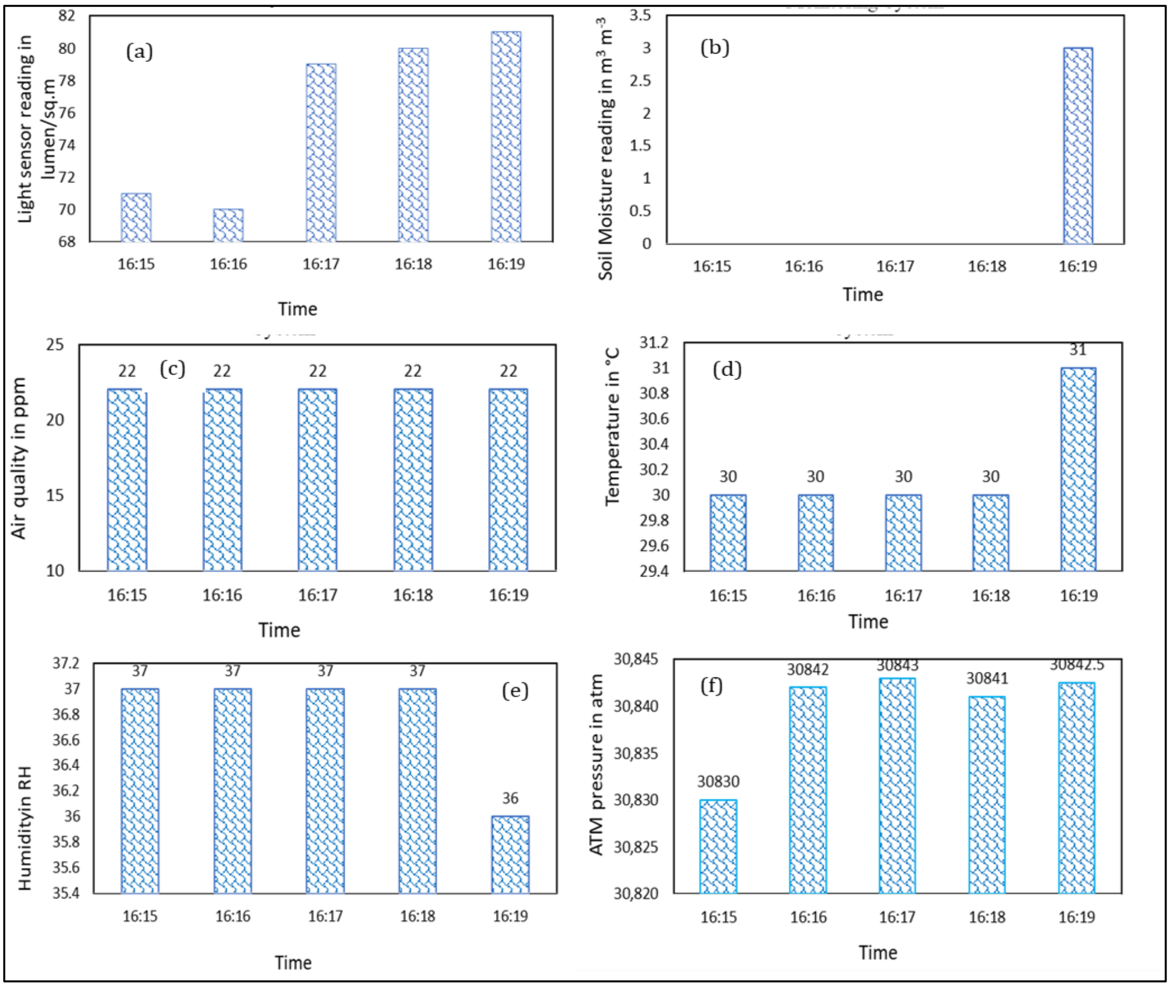

Figure 5: IoT-Based Smart Agriculture Monitoring System, (a) Light Sensor, (b) Soil Moisture, (c) Air Quality, (d) Temperature, (e) Humidity, (f) Atmospheric Pressure

Similarly, Figure 6 (a), Figure 6 (b), Figure 6 (c), Figure 6 (d), and Figure 6 (e) represent the density of the smart building under various measurements of temperature, light intensity, Co_2 levels, air quality and humidity respectively. The graphical representation emphasizes that the IoT-based model more effectively analyses the density ratio of the above measures. Through this analysis, energy usage in smart buildings is more efficiently handled.

Table-4 represents the performance of the IoT-based monitoring system in a smart environment to predict environmental changes. For this, changes in four parameters, such as natural gas, carbon monoxide, air quality, and temperature, are evaluated with respective time. Through this analysis report, people can easily predict them from environmental changes.

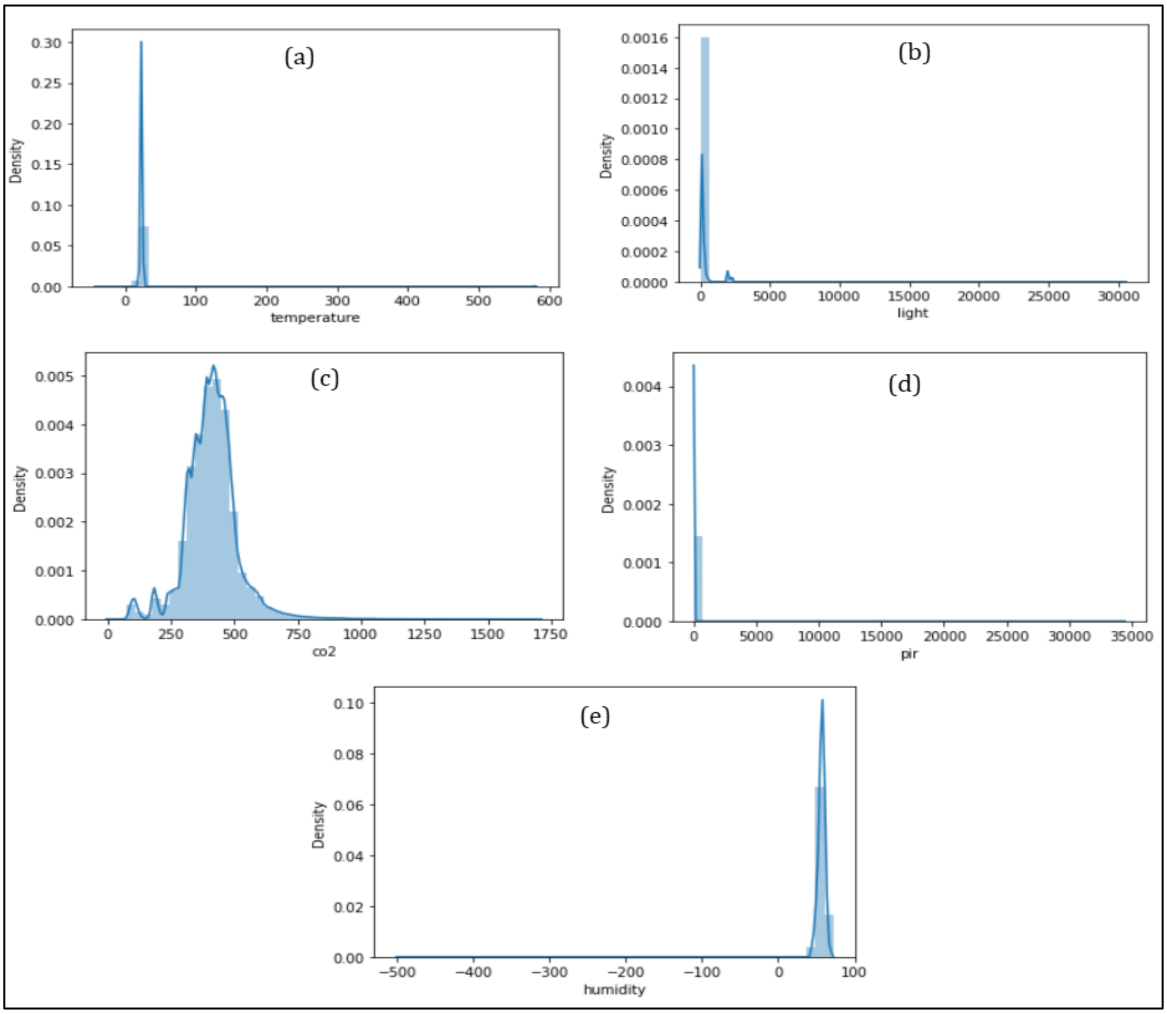

Figure 6: IoT-Based Smart Building Monitoring Systems, (a) Temperature, (b) Light, (c) CO_2, (d) Air, (e) Humidity

Table 4: Various Parameters of Smart Environment

Date	Natural Gas (ppm)	Carbon Monoxide (ppm)	Air Quality	Temperature (°C)
3AM	105	101	180	29.30
4AM	111	105	189	29.40
5AM	113	106	193	29.30

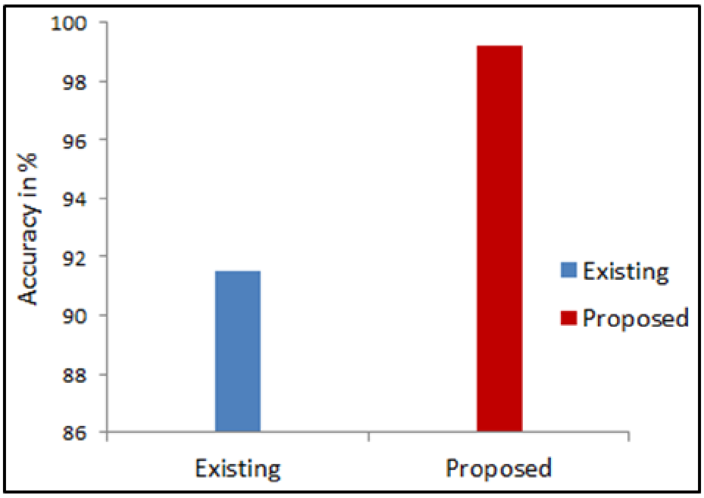

Figure 7: Accuracy Comparison

Now, the efficiency of the proposed deep learning-based framework in analyzing the multi-modality data generated by the IoT-based monitoring system is evaluated. The performance of the DL-based model is predicted by evaluating the accuracy result. Then, the efficiency of the DLF is proved by calculating the accuracy and compared with the existing approach. The accuracy comparison result of the proposed and existing model is depicted in Figure 7. The graphical representation of Figure 7 indicates that compared to the existing model, the proposed DL-based approach more efficiently analyses the data received from the IoT-based monitoring devices and improves the sustainability of the smart environment. The proposed model has analyzed the data with 99.78% accuracy.

Figure-8 depicts the error rate of the proposed and existing model on analyzing the various parameters received from the IoT-based model. The comparison shows that the DLF has processed the data received from the IoT models with a 0.10 error rate. After detecting the accuracy of the proposed model's result in analyzing the data generated by the IoT models, the model's performance is evaluated. As mentioned above, the model's performance is evaluated by calculating and comparing various performance metrics such as precision, recall, and F1-score (Table 5).

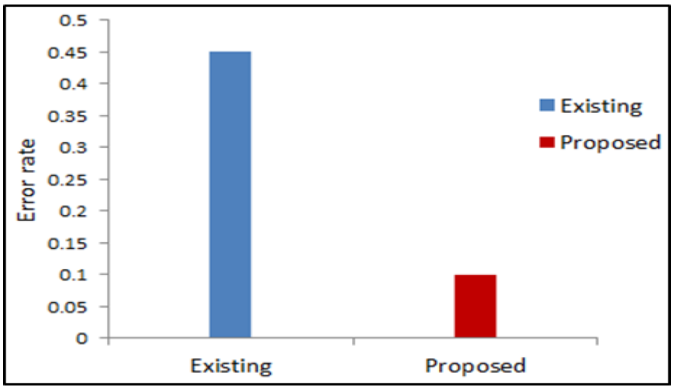

Figure 8: Error Rate Comparison

Table 5: Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score
Extra Tree Classifier	98.53%	96.69%	96.90%	96.79%
Random Forest Classifier	98.23%	96.29%	95.95%	96.12%
XG_Boost	97.92%	95.59%	95.32%	95.45%
Decision Tree	97.37%	94.68%	93.77%	94.21%
Proposed DLF	99.78%	98.97%	98.99%	99.32%

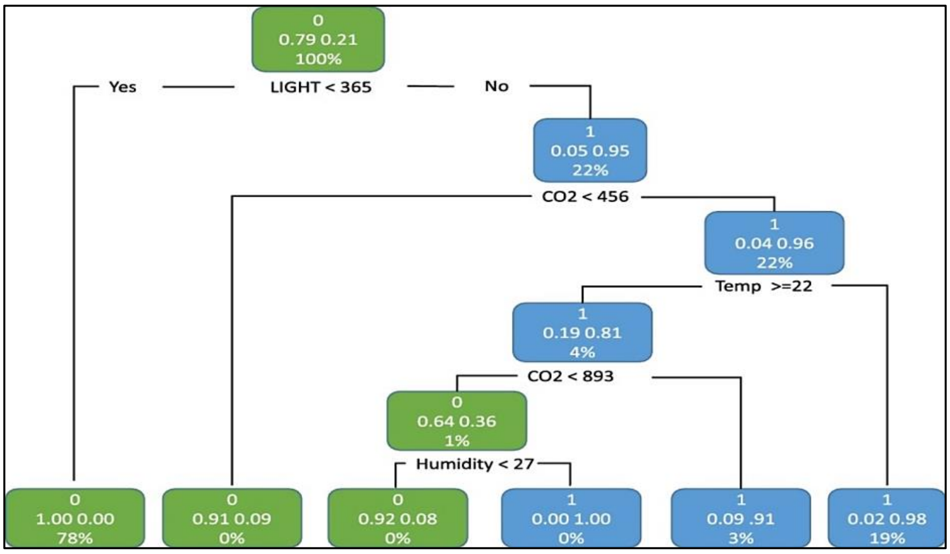

Figure 9: Sustainability Analysis

The performance factors, such as accuracy, precision, recall, and F1-Score obtained from the proposed DLF are compared with the other similar learning models and given in Table-4. From the comparison, it is noticed that the proposed DLF outperforms the others. The proposed DLF obtained 99.78% accuracy, which is higher than others, whereas DLF provides good regulation for sustainability maintenance in SEN, not for a single SE. The regulator model is enabled and regulates the SEN by assessing the internal parameters. For example, in a smart building, the light, temperature, humidity, and CO_2 are assessed as normal or abnormal, as shown in Figure-9. Based on the assessment, the parameters' range is identified, and the regulator can regulate the SEN by automatic or manual instructions.

Conclusion

The proposed deep learning framework for a smart environment network optimization provides a better-optimized control to enhance sustainability in general. The DLF model is designed in such a manner to manage all types of smart environments like smart buildings, healthcare, agriculture, industries, and the smart grid. The data obtained from the nodes are initially subjected to preprocessing and feature extraction to reduce the dimensionality of the data. The proposed model adopts a reinforcement-based deep learning model that provides better time series data processing and better insights. The LSTM algorithm helps process the time series data obtained from the devices and classifies them. After the classification process, the PSO algorithm is used to optimize the nodes. LSTM consists of a long-term memory for better optimization of the network. The LSTM processes the time series data obtained from the IoT devices, and based on the prediction made by the LSTM algorithm, the PSO algorithm optimizes the network. The results show better network optimization, reducing energy consumption and creating a sustainable environment. The model's effectiveness is compared with the existing ones, and the proposed model provides better efficiency in optimizing the IoT networks and is also suitable for various applications. Different parameters are optimized and enhance the monitoring efficiency: air quality, light, humidity, temperature, soil moisture, and atmospheric pressure. Based on the preprocessing optimization, the prediction accuracy increases, and the computational complexity is reduced. The prediction accuracy obtained from the DLF is 99.78% with an error rate of < 1%, which is highly efficient and better than other methods explained in the literature. Further the proposed work is extended to more remote applications that require high computational efficiency.

Abbreviations

Nil.

Acknowledgement

Nil.

Author Contributions

B. Doraswamy: Conceptualization, Methodology, Software, Data curation, Writing – original draft, K. Lokesh Krishna: Visualization, Investigation, Supervision, Software, Hariprasad Tarigonda: Validation, Writing – review & editing.

Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethics Approval

Not applicable.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

1. Vimal Jerald A, Rabara SA, Bai TDP. Internet of Things (IoT) based smart environment integrating various business applications. International Journal of Computer Applications. 2015; 128(8): 32-37.
2. Nazari Shirehjini AA, Semsar A. Human interaction with IoT-based smart environments. Multimedia Tools and Applications, 2017; 76: 13343-13365.
3. Prathibha SR, Hongal A, Jyothi MP. IoT-based monitoring system in smart agriculture. International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT) India. IEEE. 2017; 81-84.
4. Hajjaji Y, Boulila W, Farah IR, Romdhani I, Hussain A. Big data and IoT-based applications in smart environments: A systematic review. Computer Science Review. 2021; 39: 100318.
5. Kabalci Y, Kabalci E, Padmanaban S, Holm-Nielsen JB, Blaabjerg F. Internet of Things applications as energy internet in smart grids and smart environments. Electronics. 2019; 8(9): 972.
6. Ullah A, Anwar SM, Li J, Nadeem L, Mahmood T, Rehman A, Saba T. Smart cities: the role of Internet of Things and machine learning in realizing a data-

centric smart environment. Complex & Intelligent Systems. 2024; 10: 1607-1637.
7. Nosratabadi S, Mosavi A, Keivani R, Ardabili S, Aram F. State-of-the-art survey of deep learning and machine learning models for smart cities and urban sustainability. Inter-Academia. Lecture Notes in Networks and Systems, Springer, Cham. 2020; 101.
8. Martinho R, Lopes J, Jorge D. De Oliveira LC, Henriques C, Peças P. IoT-Based Automatic Diagnosis for Continuous Improvement. Sustainability. 2022; 14(15): 9687.
9. Bibri SE. The IoT for smart sustainable cities of the future: An analytical framework for sensor-based big data applications for environmental sustainability. Sustainable Cities and Society. 2018; 38: 230-253.
10. Yadav S, Garg D, Luthra S. Development of IoT-based data-driven agriculture supply chain performance measurement framework. Journal of Enterprise Information Management. 2021; 34(1): 292-327.
11. Bermejo B, Juiz C. Improving cloud/edge sustainability through artificial intelligence: A systematic review. Journal of Parallel and Distributed Computing. 2023; 176: 41-54.
12. Paiola M, Schiavone F, Grandinetti R, Chen J. Digital servitization and sustainability through networking: Some evidence from IoT-based business models. Journal of Business Research. 2021; 132: 507-516.
13. Al Duhayyim M, Mohamed HG, Aljebreen M, Nour MK, Mohamed A, Abdelmageed AA, Mohammed GP. Artificial Ecosystem-Based Optimization with an Improved Deep Learning Model for IoT-Assisted Sustainable Waste Management. Sustainability. 2022; 14(18): 11704.
14. Elayan H, Aloqaily M, Guizani M. Sustainability of healthcare data analysis IoT-based systems using deep federated learning. IEEE Internet of Things Journal. 2021; 9(10): 7338-7346.
15. Healthcare-2022 Dataset, 2024. Available:https://www.kaggle.com/datasets/prasad22/healthcare-dataset/data.
16. Weather Prediction-2022 dataset, 2024. Availablehttps://www.kaggle.com/datasets/ananthr1/weather-prediction.
17. Luis M. Candanedo, Véronique Feldheim. Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models. Energy and Buildings. 2016; 112: 28-39.
18. Smart Building System – EDA Input data KETI, 2022. Available:https://www.kaggle.com/code/halimedogan/smart-building-system-eda/input.
19. WiDS Datathon-2022 dataset, 2024. Available:https://www.kaggle.com/competitions/widsdatathon2022/data.