DEEP LEARNING BASED AGRICULTURE TRAFFIC PREDICTION USING GATED RECURSIVE DEEP NEURAL NETWORK IOT ENVIRONMENT

[1*] **M. Sofiya**
Research Scholar, Department of Petrochemical Technology
University College of Engineering (BIT Campus), Anna University
Tiruchirappalli, Tamil Nadu, India
Mail id: sofiyajan98@gmail.com

[2]**Dr. M. Arulmozhi**
Professor, Department of Petrochemical Technology,
University College of Engineering (BIT Campus), Anna University
Tiruchirappalli, Tamilnadu, India
Mail id:arulmozhisenthilkumar@gmail.com

Abstract
IoT traffic data can be used to apply deep learning technology to agriculture. The application of deep learning algorithms in agriculture has improved, resource management, and decision-making with positive outcomes. Precision agriculture can assist regulate crop yields by applying nutrients only when necessary to enhance crop quality and lessen adverse environmental effects. This is made possible by IoT capabilities. Identifying redundant data traffic remains a major research challenge in the field of IoT-based agricultural automation, despite the fact that numerous solutions have been offered. Additionally, farmers do not receive the necessary information about water levels, soil conditions, etc. However, using standard methods of processing makes it challenging to handle the complexity and dynamic of low data transfer. To overcome these issues, the suggested Gated Recursive Deep Neural Network (GRDNN) makes use to increase the prediction accuracy of network traffic in an IoT environment and transfer learning to address the issue of insufficient IoT Agriculture data. Then the behavioral support factors of features are filtered through Entity Spectral spider optimization algorithm to find the relational feature weight. Then fitness function evaluates the scalar variations to get the support index based on the behavioral variation. The features get indexed by class by reference to the average mean weight and selected through Cluster-Scalar Entity Feature Selection (CSEFS) model. The selected features are progressed into Soft-max Neural Network (SMNN) which is optimized with GRDNN classifier. This optimization evaluates the entity scalar values to train the features in the Gated Recursive Deep Neural Network (GRDNN) to classify the result. The proposed system predicts the threading levels to categorize the Risk level based on features threshold margins to improve the security as well in IoT. This proposed system achieves high performance compared to the other system in precision rate with least false rate to attain the accuracy.
Keywords: Internet of Things, features, agriculture, spider optimization, Gated Recursive Deep Neural Network, Risk level.

I. Introduction

Nowadays, essentially everyone's life is dominated by the Internet and the Internet of Things (IoT). IoT is a paradigm that links people, things, or networks so they can accurately process and respond to any kind of communication, whether it be virtual or physical. Every industry may benefit from the Internet of Things, including home controllers, healthcare, and agriculture. Our efficient provision of essential services to consumers is achieved by utilising several technologies and protocols for internet connectivity, sensors, data collection, and analysis. Businesses may automate processes and enhance service delivery by utilizing cloud-based data transfer and internet technologies thanks to the Internet of Things.

The Internet of Things prevents the adoption of uniform software architecture for sectors that make use of it. In light of the rising global food demand, smart agriculture is more crucial than ever. In this regard, smart technology has emerged as a crucial tool for modern agricultural practices. The agriculture industry has a wide range of applications, protocols, and prototypes [1]. Ultimately, information is sent over the internet to a backend server. These devices helpsto treat dataset more effectively while also keeping safe and secure. Improvements in patient involvement and satisfaction have coincided with the ease and efficiency of doctor-patient contacts. Additionally, staying in the hospital less time and avoiding readmissions are two benefits of remote patient monitoring. Reduced healthcare costs and improved treatment outcomes are two major benefits of IoT [2].

It can be accomplished by precisely predicting the traffic on the channels well in advance of the actual traffic. Important challenges in the Internet of Things (IoT) context include the need for high transmission rates, minimizing energy consumption by maximizing battery life and channel utilization, minimizing latency in wired or wireless communications, and making the best use of computational and data-intensive resources. All of these challenges necessitate early traffic prediction. These problems include accurate traffic forecasting, which can undoubtedly help with the previously listed problems.

II. Related work

Smart agricultural sensors' management of quality of service (QoS). Selecting the best IoT node is getting harder when taking into account QoS factors like energy usage, latency, and network coverage region [1]. The agriculture sector will see a rise in productivity and operational efficiency with the application of data analytics and the IoT[2]. Wi-Fi sensor networks (WSNs) have given way to the IoT and Data Analytics (DA) as the main forces behind smart agriculture. Existing agricultural systems are being disrupted by evolutionary changes, which also present a number of opportunities and problems [3]. IoT device are carried out to detect the information analysis are complex tasks, requiring various testing methods, including software and Hardware [4].

---

[1] S. P. Singh et al., "A New QoS Optimization in IoT-Smart Agriculture Using Rapid-Adaption-Based Nature-Inspired Approach," in IEEE Internet of Things Journal[2] O. Elijah, T. A. Rahman,
[2] Orikumhi, C. Y. Leow and M. N. Hinda, IEEE Internet of Things Journal, vol. 5, no. 5, pp.
[3] A. AlZubi and K. Galyna, in IEEE Access, vol. 11, pp. 78686-78692, 2023, Doi: 10.1109/ACCESS.2023.3298215.
[4]O. Friha, M. A. Ferrag, L. Shu, L. Maglaras and X. Wang, "Internet of Things for the Future of Smart Agriculture: A Comprehensive Survey of Emerging Technologies," in IEEE/CAA Journal of Automatica Sinica, vol. 8.

Vulnerabilities to wireless spoofing attacks are a severe problem to Next Generation Internet of Things (IoT) networks [5]. IoT devices connected by Wireless sensors of Agriculture-Physical Systems (CPS) have some complex levels: various systems, communication links, delivery ratio, delay, bandwidth, low-level security and high traffic occurrence [6]. IoT develops standard pattern security issues that received significant attention showing the various methods provided in IoT, including anomaly detection system and threat model [7].
Provide an overview of the literature on cutting-edge agricultural IoT technologies, including wireless technologies, cloud/fog computing, software-defined networking (SDN), open-source IoT platforms, drones, and network function virtualization (NFV) [8]. The Internet of Things (IoT) creates "big data," or vast volumes of streaming data, and presents new ways to track food and agriculture processes. Big data from social media, in addition to sensors, is becoming more and more significant for the food business [9]. In comparison to current IoT-based planting and agriculture systems, the suggested solution somewhat lowers network latency. For sensing and operation, a cross-layer based channel access routing [10].
The RF energy harvesting technology is used as an alternative technology for the proposed IoT platform to power the IoT nodes of the platform. Validated a recent module for RF energy harvesting [11]. The key elements of IoT-based smart agriculture will be covered. A comprehensive review of network architectures, layers, topologies and protocols used in Internet of Things-based agriculture [12].
At present, distributed Internet of Things sensors can be powered by robust computer infrastructure seen in agricultural systems that use typical cloud architectures. However, processing, analyzing, and storing sensor data on the cloud requires transporting heterogeneous data via several network layers, which adds a large energy consumption cost to the information and communication infrastructure [13].A comprehensive, reasonably priced SA solution. Small and medium-sized farmers are reluctant to adopt commercial solutions since they are costly [14].

[5] S. Siboni et al., "Security Tested for Internet-of-Things Devices," in IEEE Transactions on Reliability, vol. 68.
[6] M. R. Nosouhi, et al., "Towards Spoofing Resistant Next Generation IoT Networks," in IEEE Transactions on Information Forensics and Security, vol. 17.
[7] Burgs, A. Chattopadhyay and K. -Y. Lam, "Wireless Communication and Security Issues for Cyber–Physical Systems and the Internet-of-Things.
[8] N. Neshenko, et al., "Demystifying IoT Security: An Exhaustive Survey on IoT Vulnerabilities and a First Empirical Look on Internet-Scale IoT Exploitations, 2019.
[9] N. N. Misra, et al., in IEEE Internet of Things Journal, vol. 9, no. 9, pp. 6305-6324.
[10] N. Ahmed, et al., "Internet of Things (IoT) for Smart Precision Agriculture and Farming in Rural Areas," in IEEE Internet of Things Journal, vol. 5.
[11] D. Boursianis et al., "Smart Irrigation System for Precision Agriculture the AREThOU5A IoT Platform," in IEEE Sensors Journal, vol. 21.
[12] M. S. Farooq, S. Riaz, et al., in IEEE Access, vol. 7, pp. 156237-156271, 2019.
[13] E. -T. Bouali, et al., "Renewable Energy Integration into Cloud & IoT-Based Smart Agriculture, 2022.
[14] Pagano, D. Croce, I. Tinnirello and G. Vitale, "A Survey on LoRa for Smart Agriculture: Current Trends and Future Perspectives," in IEEE Internet of Things Journal, vol. 10, no. 4, pp. 3664-3679, 15 Feb.15, 2023, Doi: 10.1109/JIOT.2022.3230505.

LoRa-based solutions can be used in various scenarios by considering scalability, operability, network architecture, and energy efficiency[15]. With the use of effective resource management and control technologies, network design, and the Internet of Things, a sustainable greenhouse environment may be created. There will be significant conversations around IoT-based greenhouse applications, sensors and devices, and communication methods[16].

A method to risk-sensitive reinforcement learning for multi-actor-based Aerial base stations (ABSs) task scheduling in intelligent agriculture. Work offloading with the tight requirement that he finish the IoT work before the deadline is the definition of this challenge [17]. Advancing hardware, software, and Internet of Things technologies in agriculture. Globally, entrepreneurs and public-private sector projects are being created to provide intelligent and sustainable precision agriculture solutions [18].

Increasing the value added to agricultural products is the aim. By extending the operating range, maintaining traditional agricultural planting expertise and management, and utilizing astute analysis and decision-making, smart greenhouse planting technology helps regulate and manage the greenhouse environment while lessening the cost and workload associated with agricultural management [19].Increasing food production in specific agricultural sectors—such as irrigation, soil moisture monitoring, fertilizer optimization and management, early insect and crop disease control, and energy conservation requires the Internet of Things (IoT) and wireless sensors [20]. Wireless sensor networks (WSNs) are a key enabling technology that PAs use to overcome these obstacles in order to use data processing, communication, and gathering to boost agricultural yields. Furthermore, a multitude of other interdisciplinary technologies assist PA in identifying its most innovative application cases [21]. The ground stays productive and barren when nutrients, water, chemical fertilizers, and pesticides are not used properly [22].

[15] T. Pamuklu, et al., "IoT-Aerial Base Station Task Offloading With Risk-Sensitive Reinforcement Learning for Smart Agriculture, 2023.

[16] V. P. Kour and S. Arora, "Recent Developments of the Internet of Things in Agriculture: A Survey," in IEEE Access, 2020.

[17] L. -B. Chen, et al., “in IEEE Sensors Journal, vol. 22, no. 24, pp. 24567-24577, 15 Dec.15, 2022.

[18] M. N. Mowla, et al., "Internet of Things and Wireless Sensor Networks for Smart Agriculture Applications, 2023.

[19] R. K. Singh, et al., Architecture for IoT and Emerging Technologies Based on a Precision Agriculture Survey," in IEEE Access.

[20] S. I. Hassan, et al.,in IEEE Access, vol. 9, pp. 32517-32548, 2021, doi: 10.1109/ACCESS.2021.3057865.

[21]P. Sharma, et al., "Predicting Agriculture Yields Based on Machine Learning Using Regression and Deep Learning.

[22]S. Shore Wala, et al., "Weed Density and Distribution Estimation for Precision Agriculture Using Semi-Supervised Learning," in IEEE Access.

In order to estimate agricultural yield, deep learning technologies such as convolution neural networks and long short-term memory networks are employed. Other machine learning technologies include XGBoost regression, decision trees, and random forests [23]. With a restricted amount of color photos acquired from an autonomous robot, a semi-supervised deep learning-based technique accurately determines the density and distribution of weeds in a field. For site-specific weed control systems, which employ autonomous robots to treat infested regions selectively, this weed density and distribution data can be utilized [24]. In addition to raising crop yield per acre, the primary objective is to preserve the ecosystem and enhance crop quality. Because they deplete nutrients, water, and light and lower agricultural output, weeds are a serious threat to crops. Not only is it costly to evenly spray a field to eradicate weeds, but it is also bad for the environment [25].

III. Materials and Method

The proposed approach is enhanced by the use of deep learning techniques, particularly in the form of the Gated Recursive Deep Neural Network (GRDNN). Attack detection is more effective and efficient when it comes to IoT agriculture security. To verify the advantages of the suggested MSEFS technique aggregating the feature selection, a free IoT Agriculture dataset is utilized. Lastly, the Soft-max Neural Network (SMNN) is used to estimate the classification results. The experimental findings show that compared to current classification techniques, the suggested layered deep learning method can improve detection accuracy rates more effectively.

[23] S. Shore Wala, A. Ashfaque, R. Sidharth and U. Verma, "Weed Density and Distribution Estimation for Precision Agriculture Using Semi-Supervised Learning," in IEEE Access, vol. 9.
[24] V. Gruhn, "Revolutionizing Agriculture: Machine and Deep Learning Solutions for Enhanced Crop Quality and Weed Control," in IEEE Access.
[25] Elsisi, Mahmoud and Minh-Quang Tran. “Development of an IoT Architecture Based on a Deep Neural Network against Cyber Attacks for Automated Guided Vehicles.” Sensors.

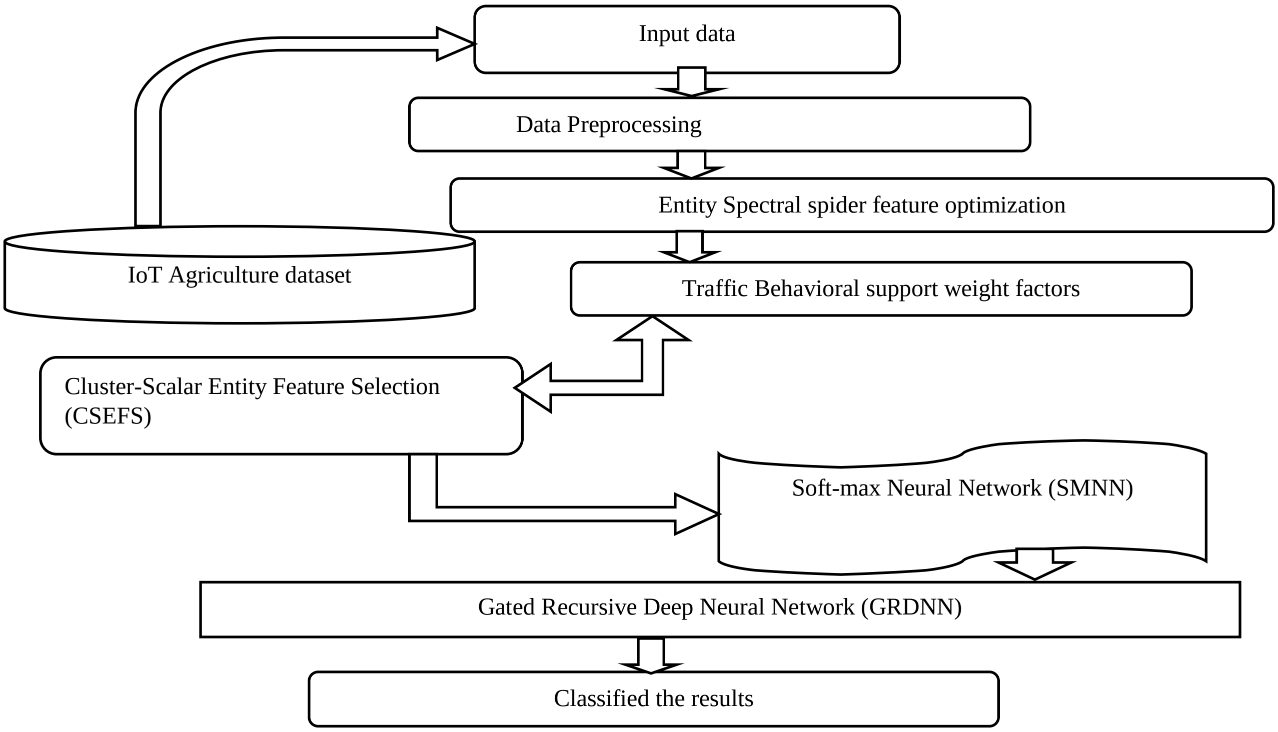


Fig. 2Proposed Architecture diagram

Figure 2 describes a proposed diagram for a Gated Recursive Deep Neural Network (GRDNN) using multi-scalar entity feature selection. Because the effect weights are trained on the satisfaction of activating the logical neurons of the input, feature values are marginalized. The redundant capacities of unseen feature learning approaches are explored by CSEFS. The best categorization is achieved by utilizing the redundant features. Following processing, the intended outcome and the actual output are compared. A computer that is compatible receives the error. During training, the data is analyzed several times, enabling the network to modify weights in order to forecast classification outcomes.

A.Data preprocessing

Preprocessing the data is the first and most important stage in building a deep learning model. To achieve the right outcome, analyses and develop the dataset first. In order to minimize unscaled features, the dataset fields were first preprocessed to confirm the existence of scaled weights. To create the original scaled features from the dataset, data must be cleaned, normalized, and transformed in order to create a trustworthy dataset.

- ❖ **Data cleaning:** Correcting and eliminating incomplete data is known as data cleaning. Process of cleaning the data to identify any missing values and eliminate such rows. To eliminate rows in the dataset that contain null values, use the pandas dropna () function.

$$dropna(axis = 0, how = \text{any}, thresh = None, subset = None, inplace = False) \quad (1)$$

ISSN: 0012-0413

- **Data Normalization:** Finding a standard measure in a dataset is known as normalization. A few characteristics of datasets are the results integrating the results, which requires time, and variance range data, which makes model training more difficult. Normalizationsfacilitate a faster unification of the model, which enhances its performance. The Min-Max feature scaling standardization technique is applied in the model. Multi-scale features are converted into a secure scale range [0, 1].

$$a_{norm} = \frac{(a-a_{min})}{(a_{max}-a_{min})} \quad (2)$$

$Let, normormilize\ (\propto) = a_{max} - a_{min} * new\_minimum + new\_limit$ And $b = \propto a - \propto \min imum(a) + new\_min,\ Let, norm(\beta) = a_{min}(a) - new\_minimum$, be

$$b = \propto a - \beta \quad (3)$$

$Mean\ (b) = mean(x) * \propto -\beta$, and $std\ (b) = std(a) * \propto\ then\ var\ (b) = var(a) * \propto^2$

Where x, y is a random value,$a_{min}\ and\ a_{max}$ is a.$minimum$ () and .$maximum$ () Function of getting pandas. It generates the normalization scaled features from the dataset and evaluated the minimum and maximum values.

B. Entity spectral spider Optimization.

In this phase, the Max-Vibri function is constructed to choose the marginal features, and Entity spectral spider optimization is performed to discover the relational weights. The Euclidian distance formed fitness evaluation, which is the maximum support together with the same relational feature from the spider layer, is used to evaluate feature difference. The population unit, which is regarded as the maximal feature limits at each margin, is fixed to the margins at the iterative layer by the random function.

In order to derive the layer's maximum weight closest to the spider centroid weight, the vibri function defined boundaries. The threshold margins determine how the relative intersection boundaries create the feature cluster. The feature threshold margins specify how accurately the relevant characteristic will be taken into account. To resolve a box constraint in a nonlinear global optimization problem of the following form:

$$\min f(a)\ a = (a^1, a^2, a^3), \ldots a^n$$

The search space is represented by the general web, whereas each solution represents a spider location. The population is split into two search agents in the approach: Male (Ms) and Female (Fs). In an effort to replicate an actual spider colony, the number $a^n$is of females, and Nfe is selected at random from 65 to 95 percent of the total population S; the remaining Nm is regarded as the male population ($Nma = S - Nfe$). In these circumstances, a group of women make up the Fs.

Input: Features Attributes $S_{rw} \rightarrow R(k)$

Output: Dataset values selected feature

Step1: Start

Step 2: Enter the feature values (number of iterations, feature count, maximum feature weight, and minimum feature weights)

Step 3: Preprocessed values of (V) spider $Sp_r, w = 0,1,2, \ldots N$

Step 4: Compute the index values as int ranges (I) no scaled values (F)

Attain centroid threshold margins fix feature weights to layers

$$f \rightarrow (Ts)^w = \sum_{i=0}^{w} \left(\frac{w}{I}\right) Initilazed\ margin^w$$

Compare Fn→W (Euclidian distance variance from vibri) to Ts

Ts - Process of Threshold-features weights

Step 7: Estimate the All-Relational values generated (v_g) features

Remove the scaled terms $Ts \leftarrow K_g$

Totalize all the process's auxiliary scaled features. Real (S) scaled, Non-real (Ns) scaled

In the Scaled and no-scale value of weight features, S_r

Evaluated the Features weights $Wf = \frac{F(S_r)-not-Scale_i}{Scale_i\_not-Scale_i}$(5)

Check if $(v_g as \rightarrow)$

By determining the weight parameters to choose the co-feature weights, it enhances the social spider optimisation. We define spider estimates and give each spider a weight. Each spider that represents the solution has a weight function that is computed as follows:

$Wf_s = \frac{K(Sf_r)-mini_w}{imaxi_w-mini_r}$ ------- (6)

$limited\_feature_r = \underset{Wfs\{1,2,..n\}}{\text{maximum}}(v(fS_r))$(7)

$No_{scale}feature_r = \underset{Wf\{1,2,..n\}}{\text{minimum}}(v(S_r))$ (8)

A return of every feature value as features selected from the dataset v_f

End

End

Step 7: return features v_f

Step 8: End

The feature weights execute feature selection to remove unnecessary features from the original dataset, filter the dataset to assess the relational features of the data, and compute the scaled and non-scaled features.

C. Clusters-Scalar Entity Feature Selection (CSEFS)

By analyzing more relational attributes from the cluster group, the mean rate offers the variation from cluster weight to entity differences. In order to scale the marginal values based on the spider optimization method and assess the feature that is related to the cluster, Multi-Scalar Feature Selection extracts the relationships between the features in this step. The layer threshold weights to choose the closest feature weights, the SOA generate decision layers for feature weight selection. In order to lessen unrelated features, it minimizes uncorrelated features and makes enhancements. A subset of the features is used for classification or prediction using CSEFS after the values are evaluated using the softmax activation function, which also gives each feature a weight.

Input:$S = \{(a_x, b_x)x = 1,2,3 \ldots, N\}$; feature values$(f_v)$

Output: weights matrices $w \in \mathbb{R}^{Dxl}$

Input $W = 1$

For $x \leftarrow 1$ to number of label features f do

Repeat

Calculate the tp, fn, fp weights of features are used to find the distance

$$c_w(n, w_x) = \sqrt{\sum_{x=1}^{n} w_x^2 \, (f_v - f_{xy})^2}$$

$$c_w = \frac{1}{N}\sum_{x \in T} f_y - (X)$$

Identify the value count feature F_c to feed

Maximum weights (maximum value count→support values)

Create a link between features traffic data→Packets id

Step 3: calculate each feature value F_v

For $F = 1 \; to \; N$

Calculate MSF (F_w, C)

End for

Step 4: Sequencing the Feature count (Max, Min) values (F_w, C) and $(F_{seq} = f)$

While $(F_{seq} \neq null)$ to create decision nodes

$(F_w = getrelationalelement(F_{seq})$

$if \left(Msf(Fe_w, C) - \frac{1}{|Fe_{rel}|}\right) \sum_{i \in F_{rel}} Msf(Fe_w) < threshold\ val (9)$

Remove $minF_w \; fromF_{seq} \; and \; continue$,

Else

Remove $minF_w \; from \; F_{seq} \; and \; add \; to \; F_{re}e$

Step 5: Determines the connections there are between the elements.

$(F_w = max \; relational \; feature(F_{seq}W)$

Step 6: Compute the maximum count characteristic value

$$mval \; = \; \sum fweight(F_c) < -\sum val(F_c)$$

$if \left(Msf(Fe_w, C) - \frac{1}{|Fe_{rel}|}\right) \sum_{i \in F_{rel}} Msf(Fe_w) < threshold\ val)$

Add $\; maximum \; F_w \; from \; F_{ree},$

Step 7: compute the Entity scaled feature →ESF.

For all feature←EF

$ESF == \sum weight \; (fe \; (Fe_c))) \text{€} \sum \; fweight \; (F_{ev})! = pds$

Compute relative feature→EFw

Efw = $(F_c + F_v)$

Add to feature set ESF← EFw

Return class feature set

End

Step: Stop

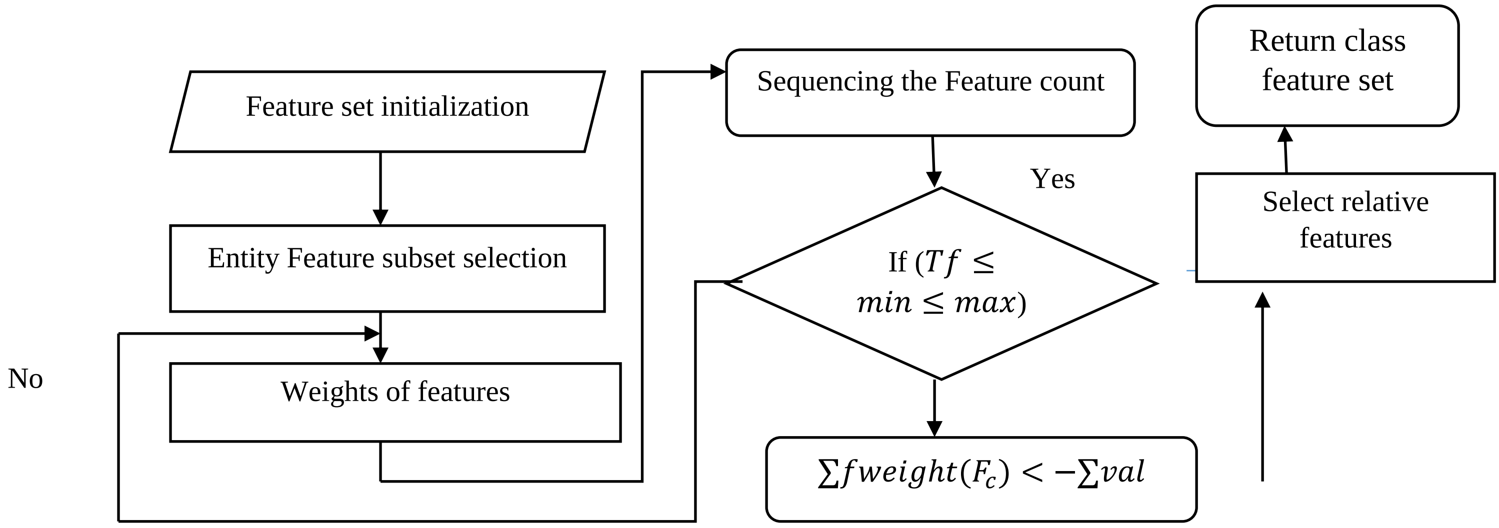


Fig. 2 Flowchart for cluster feature selection

Figure 2 shows the Features get removed that are redundant after being chosen based on clustering. After the data is first cleaned, prioritized features are chosen from the original features using the cluster scalar entity feature selection technique. Based on the variation values, the sensitive features are then chosen from the major features. In order to achieve optimal features, redundant features between sensitive features are eliminated after the sensitive features are entered into a hierarchical clustering algorithm.

D. Soft-max Neural Network (SMNN)

The softmax activation function of neurons is modified by the threshold level. The activation function picks the weights of each layer by training neurons and converting the generated clusters into average depth values. Based on feature weight estimates, RNN training rules produce neurons that are closest to the threshold. Until the intended outcome is obtained, the weights of the neural network connections can now be iteratively changed such that the output matches the actual assignment.

$$f(s) = \begin{cases} R = 1 \; if \; \sum_{i-1}^{n} w_i s_i \geq b \\ R = 0 \; otherwise \end{cases}$$

$$w_{(t+1)} = w_t - \mathbb{N} \triangle w_t \; , b_{(t+1)} = b_t - \mathbb{N} \triangle b_{t.} (13)$$

Where f(x) is a trained neuron's softmax activation using intra-class logistic transformation; the feature weights, b-relational features, R-variables, N-total features, and t-training values were used. To get the nearest mean value, this training function to the neuron's end.A fully connected neuron constructed of the network (N),$N_{i(t)} = \sum_{x=1}^{x} w_{xy} R_{y(t)} + s_{x(t)}, s = 1 \dots x$ A set point optimizes the effective training weights of the present features and the neurons with frequent weights based on the intra-class logical activation function. The weightage w(x) determines whether to activate the input S(x) and R(y).

Input: Feature weights fw, $N_{i(t)} \dashrightarrow If$

Output: Optimized class feature

Step 1: Determine the feature weights' maximum weight and rate.

Step 2: Read dataset values and fwdata values

For all layer (CFS)

Step 3: Compute the hidden layer neuron weight to w- as feature class (fc)

$$\int_{i=1}^{size(IF)} \sum If(fs).class = w$$

Every value that is closest to another value in the impact ratio's relative link f

Step 4: Based on the risk of the Max feature divided into categories, similarity features are categorized.

$$\text{Feature weights} fw = \frac{\int_{a=1}^{size(f)} \sum f(a) == fs(a)}{size(f)}$$

$$\text{Calculate collective rate } fw = \frac{\sum_{i=1}^{size(fw)} cfs}{size(fw)}$$

Step 5: Stop

Neural weights indicating the class are assigned to the input features once they have been trained on the classifier structure. By using hidden neurons to learn related traits, the input components aid in the segmentation of sections. To increase classification accuracy by reference, an activation function modifies every neuron. Two classification issues can be resolved by recurrent neural algorithms, which are artificial neurons with activation functions (including hidden layers). Neural use processing occurs at the individual neuron level. Through manipulation of the weights assigned to the perceptron which defines the weighted input for each neuron—the activation function reduces the number of layers in the two levels.

E. Gated Recursive Deep Neural Network (GRDNN)

The GRDNN predictor based on collaborative transfer learning is proposed in this paper to anticipate IoT-based traffic. The three stages of the GRDNN predictor's operation are transmission, model training, and data processing. To train the model, the GRDNN model has been suggested.

There are numerous hidden layers, an output layer, and an input layer for GRDNN. X = [x, x1... xd] defines the input layer, while Y = [y1, y1...yn] represents the output layer. In the lth hidden layer, there are m neurons, and [h l = h h l 1, h l 2... h l m i], where each artificial neuron is connected to another that has a weight and a threshold.

$$f_x^1 = f_x^1(w_1^x . f_x^{1-l} + f^1)$$

The activation function of neuron h l−1 i is represented by f l i, the bias parameter of neuron i in the lth layer is described by b l i, and the vector of weights for the connection between the neurons of layer (l − 1) and the lth layer is represented by w l i.

$$S(a)_x = \frac{\exp(a_x)}{\sum_y \exp(a_y)}$$

$$\frac{\partial S(z_x)}{\partial z_y} = \begin{cases} S(z_x) * \left(1 - S(z_x)\right) & if\ x = y \\ -s\left(z_y\right) * S\left(z_y\right) & if\ x \neq y \end{cases}$$

Typical activation functions are the sigmoid function, the hyperbolic tangent function, or the rectified linear unit. In this architecture design, the rectified linear unit (ReLU) was used as an activation function in figure 3.

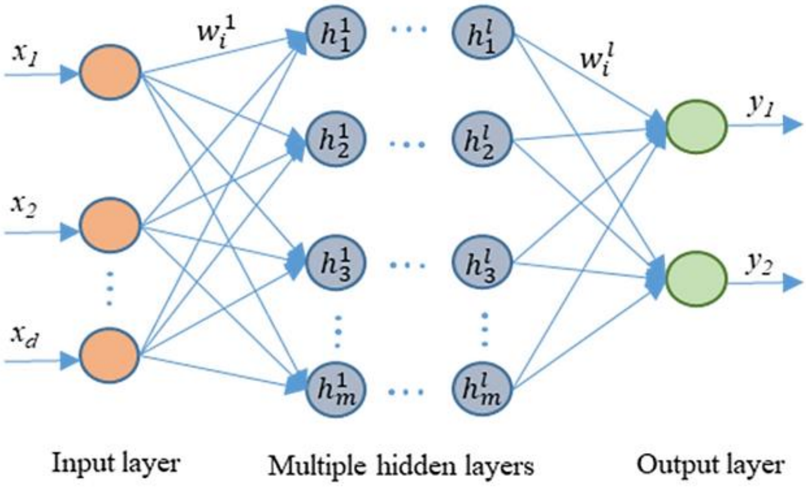


Fig.3Neural Network Architecture

A multi-part loss function is optimized that has adjustable weights to tailor the resultant model to meet detection objectives

Begin

Input values $V = \{v_1, v_2, v_{3,\dots} v_n\}$, maximum iteration (m_i)

Evaluate dataset values using coefficient

$$\lambda_{coeffient} \sum_{x=0}^{f^2} \sum_{y=0}^{B} \alpha_{xy}^{Agri\_dataset} \left[(u_x - u_x)^2 + (u_x - u_x)^2\right] + \lambda_{coeffient} \sum_{x=0}^{f^2} \sum_{y=0}^{B} \alpha_{xy}^{Agri\_dataset} \left[\left(\sqrt{u_x - u_x)^2} + \sqrt{(u_x - u_x)^2}\right]\right.$$

Set $\delta = 0$

While $(\delta \leq \tau) do$

Coefficient values using points based on $x_n \rightarrow x_{n+1}$

Perform all selected features

Calculate the distance using for information of features

$$f_x^1 = f_x^1(w_1^x . f_x^{1-l} + f^1)$$

$$s(f_x^1, \lambda_{coeffient}) = [\sum_{x=0}^{n} (f_x^1 - \lambda_{coeffient})^2]^2$$

End while

$$C_t = f_t\ o\ C_{t-1} + x_t\ o\ \hat{\iota c}_t$$

$$hi_t = o_t\ o\ \sigma_h(C_t)$$

Where, a_t- Input neuron values, F_t- forgetting corrections parameters, x_t- input parameters,of_t-output function parameters,hi_t-hidden parameters,$C_t - layers\ parameters,$ W, V, b- Weight metrics and bias,σ_s-softmax function,σ_{hi}-Hyperbolic tangent function.

IV. Result and discussion

Agricultural datasets are used to evaluate the proposed architecture. Numerical experiments were carried out in the optimization exercise to confirm the performance of the suggested model. Utilizing these benchmark datasets features that support network security, and features that carry out random data processing in simulated situations, the results forecast IoT traffic. An Intel® CoreTM i7 9750HF CPR @ 2.60 GHz, 6 core, 12 logical processors, and 64 GB RAM Windows PC was used for testing. Python is used to create a deep learning model in the Anaconda environment. Uses an analogue tool called the Intent Framework to make the computer configuration time problem more realistic.

Table 1: Simulation parameters used for the proposed method

Simulation Parameters	Simulation Values
Dataset Used	IoT Agriculture dataset
Environment	Anaconda
Language	Python
Total Records	2000
Training	1500
Testing	500

Table 1 shows the simulation settings used for the suggested technique analysis. Processed IoT traffic forecast to assess the effectiveness of the proposed solution. The results of the suggested framework are contrasted with those of the current methods. In terms of accuracy, our proposed framework performs better than previous methods.

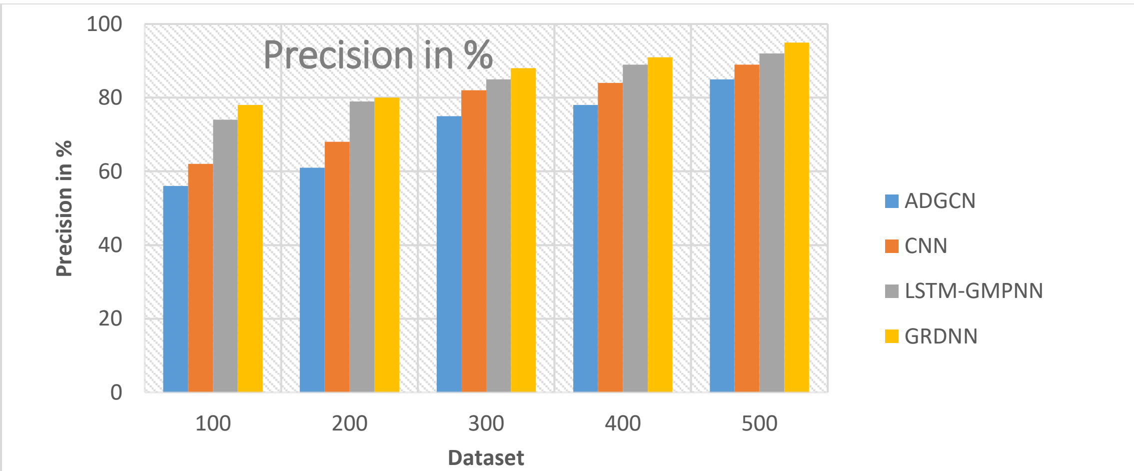


Fig.4. Precision Performance

Figure 4 depicts IoT platforms equipped with deep learning algorithms can analyze historical data from IoT agriculture to predict using various algorithms. These predictive insights empower farmers to proactively plan and adapt their farming strategies for improved.

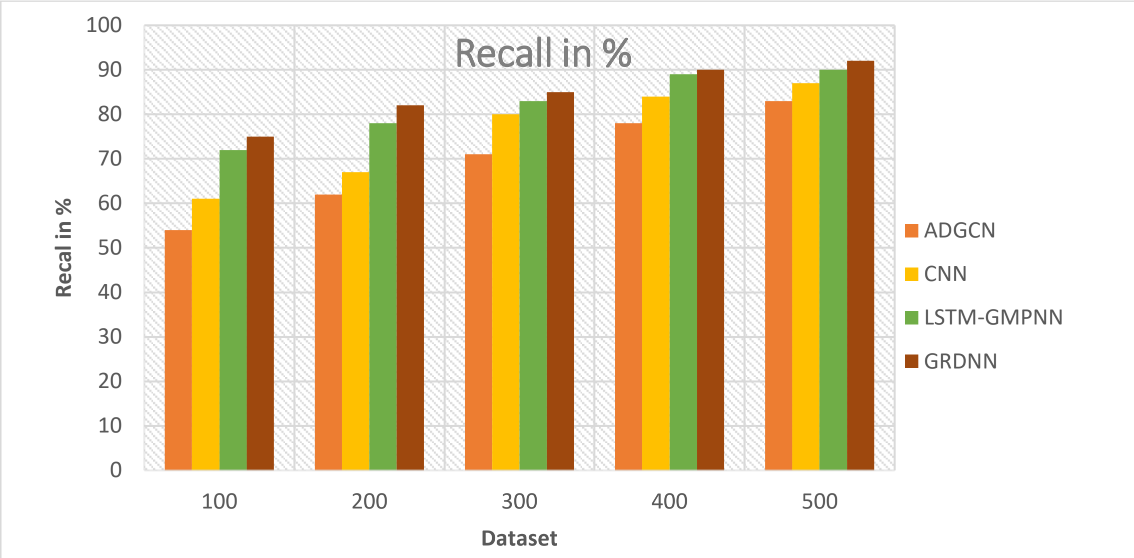


Fig.5. Analysis of Recall

Figure 5 shows recall is the measure of our model correctly identifying True Positives. Thus, for agriculture the dataset, recall is correctly identified as having traffic in agriculture. ProposedGRDNN method to alternative paths with lesser execution latencies, such as ADGCN, CNN and LSTM-GMPNN.

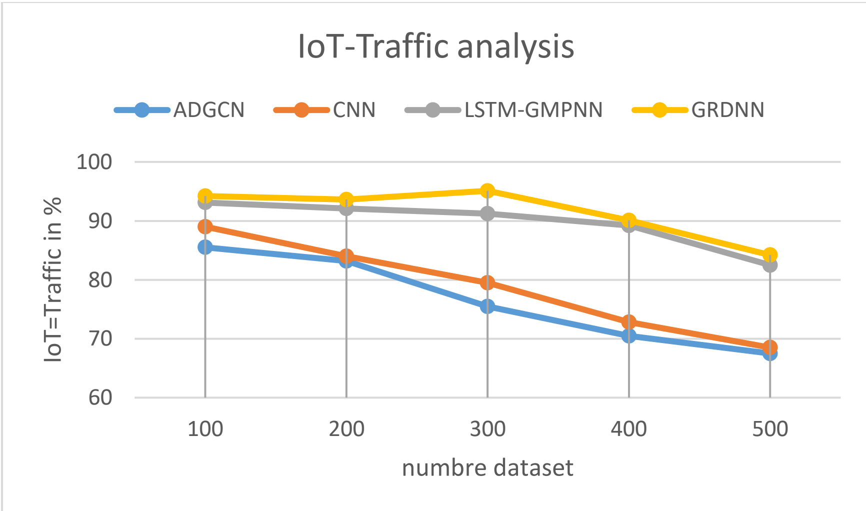


Fig.6. IoT-Traffic analysis

Figure 6 depicts the link stability of the proposed GRDNN vs the existing TADGCN, CNN and LSTM-GMPNN.Schemes between the base station and the aggregation node. When we look at the number of nodes in this method, the GRDNN is 84.2% with 400 nodes. It is more stable than standard TADGCN, CNN and LSTM-GMPNN.

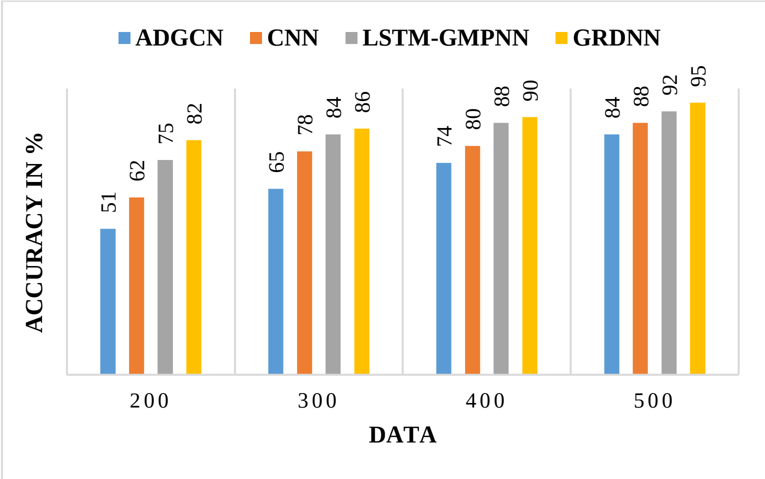
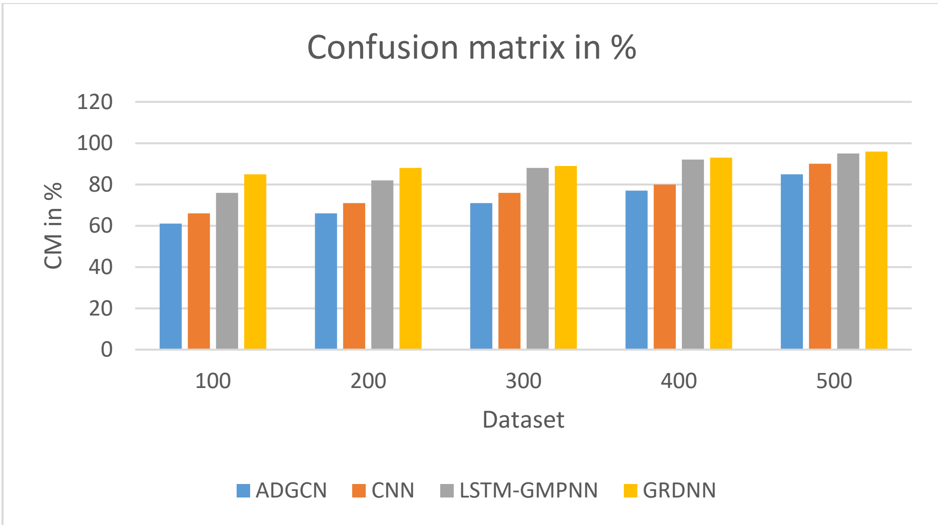


Fig. 7. Performance of Accuracy

The Accuracy of the proposed technique compared to two existing approaches. Figure 7 shows that the suggested GRDNN technique has shorter time delays than other present solutions regarding Accuracy values.



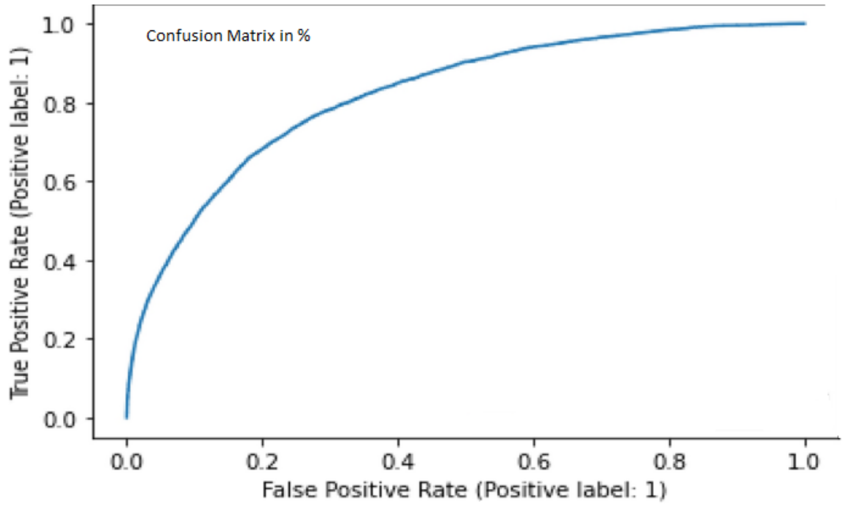


Fig.8. Confusion Matrix

Figure 8depicts proposedGRDNN method is 92%. Similarly, the influence of existing processes shortens the network lifetime of TADGCN, CNN and LSTM-GMPNN.Fitting score, accuracy, and number of dropped features in the last generation of the proposed method. From each service, it can be observed that the average accuracy and number of dropped features are dependent on the weight values.

V. Conclusion

Predicting traffic in advance is essential for managing Internet of Things traffic and optimizing bandwidth and resource usage. Internet of Things devices and traffic prediction is essential to increase channel capacity and lower network latency. In this paper, we examine the topic of IoT flow forecasting and provide a GRDNN neural network-based approach. Established traffic forecasting techniques are also reviewed and considered for comparison. The proposed GRDNN accurately predicts the traffic based on the statistical performance evaluation metrics, as shown in the results. The advantage of GRDNN is that it can solve the problem of low time and performance and information loss of hidden layer. The proposed GRDNN remembers long-term communications and other traffic characteristics in IoT environments.GRDNN method for traffic forecasting will be upon, which may improve the performance efficiency of existing predictors by combining the characteristics of different methodologies. Additionally, this system detects redundant information, which minimizes inefficiencies in network traffic. Furthermore, the suggested framework uses security techniques to thwart privacy breaches on agri data. The performance results demonstrated that the suggested structure, with its maximum energy efficiency and delivery ratio, produced positive results. Its performance has also demonstrated better processing utilization in the presence of traffic devices.