ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2025.0532

IOT-ENABLED FEDERATED LEARNING MODEL FOR CROP YIELD PREDICTION USING MACHINE LEARNING AND OPTIMIZATION ALGORITHMS IN AGRICULTURE SECTOR

J. Jagadeesan and R. Nagarajan
Department of Computer and Information Science, Annamalai University, India

Abstract

The agricultural sector plays an essential part in the financial progress of any country. The manufacture of agriculture is significantly compressed by rice, a crop developed all across the globe. Prompt prediction of diseases is essential for limiting their propagation and decreasing crop damage. However, physically analyzing crop illnesses in areas with massive agricultural regions and limited professionals is hugely complex. Using deep learning (DL) and machine learning (ML) models for analyzing illnesses of farm yields seems effective and appropriate for extensive applications. Federated learning (FL) has become a developing technology for data analysis for vast IoT applications. The manuscript proposes an IoT-Enabled Federated Learning for Crop Yield Prediction Using Machine Learning and Optimization Algorithms (IoTFL-CYPMLOA) technique in agriculture. The IoTFL-CYPMLOA model mainly focuses on enhancing the crop yield prediction model that improves agricultural production. Initially, the data pre-processing stage involves various steps such as categorical to numerical, handling null values, and normalization to clean, transform, and organize raw data into a suitable format. Furthermore, the XGBoost method is utilized for prediction. To improve the XGBoost model's prediction performance, the parameter tuning process is performed by implementing the coot optimization algorithm (COA)method. The analysis of the IoTFL-CYPMLOA technique is examined under the crop yield prediction dataset. The experimental validation of the IoTFL-CYPMLOA technique portrayed results. Compared to recent approaches.

Keywords:
IoT, FL, Coot Optimization Algorithm

1. INTRODUCTION

Agriculture is a significant social concern since it gives considerable food [1]. The incorporated effects of natural weather changeability, soil loss, increasing population, and climate-changing demand models guarantee production and crop growth in a reliable and timely way [2]. These necessities specify that land assessment, crop protection, and crop-yielding forecast are of higher significance to global food production. Consequently, the nation's policymakers require a precise crop-yielding forecast to attain suitable import and export evaluations to improve national food security [3]. Nevertheless, with multiple complicated factors, the crop yielding prediction is difficult. Generally, crop yield depends upon various factors, comprising soil quality, landscapes, harvest planning, genotype, climatic conditions, pest infestations, accessibility and quality of water, and more [4]. Crop yield approaches and processes are non-linear and time specific.

In the ever-changing landscape of technological development, the Internet of Things (IoT) and Artificial Intelligence (AI) project dual transformative forces, which have permeated several industries, carrying forth ground-breaking improvements and developing the method to interact and perceive worldwide [5]. With the myriad fields advancing from their convergence, agriculture has grown into a notable field where the fusion of IoT and AI is producing an impactful performance [6]. During this agricultural context, a single field that requires the most excellent attention in IoT security for crop yielding forecast, and where the symbiotic relations among IoT and AI might be utilized to bring about unprecedented developments in predictive modelling, data analysis, and agricultural efficacy [7]. Such methodologies overwhelm agricultural frameworks that are either non- or linear by guaranteeing a prominent forecast capability [8]. With many other models, the FL method is effective for yielding crop production [9]. FL enables us to keep data secure at the client end and make a method at the server end with higher precision [10]. The server comprises the collective details of each prediction of local models and is utilized for detailed examination.

This manuscript proposes an IoT-Enabled Federated Learning for Crop Yield Prediction Using Machine Learning and Optimization Algorithms (IoTFL-CYPMLOA) technique in agriculture. The IoTFL-CYPMLOA model mainly focuses on enhancing the crop yield prediction model that improves agricultural production. Initially, the data pre-processing stage involves various steps such as categorical to numerical, handling null values, and normalization to clean, transform, and organize raw data into a suitable format. Furthermore, the XGBoost method is utilized for prediction. To improve the XGBoost model's prediction performance, the parameter tuning process is performed by implementing the coot optimization algorithm (COA) method. The analysis of the IoTFL-CYPMLOA technique is examined under the crop yield prediction dataset. The significant contribution of the IoTFL-CYPMLOA technique is listed below.

- The IoTFL-CYPMLOA model cleans and structures raw data by converting categorical variables to numerical values, efficiently handling missing values, and normalizing features. This improves data quality and enhances model performance and reliability in predictions.
- The IoTFL-CYPMLOA method employs the XGBoost approach for precise and scalable crop yield prediction, utilizing its gradient boosting and regularization merits. This enables robust learning from complex, high dimensional data. It significantly improves prediction accuracy and model stability.
- The IoTFL-CYPMLOA approach integrates the COA technique to fine-tune XGBoost hyperparameters, ensuring optimal learning configurations. This adaptive tuning improves convergence and model generalization. It effectually improves performance while minimizing the risk of overfitting.

- Integrating the COA for tuning XGBoost presents a novel strategy combining biologically inspired metaheuristics with advanced ensemble learning. This integration improves the model's capability of navigating complex parameter spaces, resulting in superior prediction accuracy and improved adaptability. The novelty is in applying COA specifically for crop yield prediction using XGBoost.

2. LITERATURE SURVEY

Abu-Khadrah et al. [11] projected a model aimed at sensor operations, which assist agricultural production developments. The control operations are implemented based on the modified control acute sensor and sensor control validation that raises productivity. The operation and sensor controls are established utilizing FL. Oikonomidis et al. [12] focused on giving an outline of the advanced application of DL in crop-yielding prediction. A complete analysis and integration of the primary analysis are accomplished concerning the major motivations, target crops, models, features, and data resources. CNN is the standard model with better performance regarding RMSE. Senapaty et al. [13] focused on progressing a strong method. This method is incorporated with IoT-generated data and FL-based FS models to enhance the database precision.

Diverse FS models are implemented in the database. ML models are utilized to enhance and analyze the performance. The FL model is applied to train the local models utilizing the specific partitioned database. Zafar [14] presented a comprehensive analysis of models, challenges, and future directions in using ML and IoT for predictive analytics in smart farming. This model inspects the existing cutting-edge models in IoT-enabled data acquisition and the application of ML models like SVM, RF, and ANN in agricultural predictive analytics. In [15], aimed at the crop yield prediction challenge. It employs feature attribution approaches to calculate input feature contributions, recognize significant growth levels, and explain lesser precision predictions. Mohan et al. [16] utilized XAI and AI models to forecast crop yield and evaluate the effects of climate change on agriculture.

Existing studies illustrated progress in integrating IoT, FL, ML, and DL for crop yield prediction, yet challenges remain in model interpretability, scalability, and real-time deployment. Many approaches concentrate on specific crops or localized datasets, limiting broader applicability. The research gap is in developing a unified, explainable, and scalable model that effectively integrates diverse data sources while maintaining high accuracy across various agricultural conditions.

3. MATERIALS AND METHODS

This manuscript proposes the IoTFL-CYPMLOA technique in the agriculture sector. The IoTFL-CYPMLOA model mainly focuses on enhancing the crop yield prediction model, which aids agricultural production improvements. It contains three stages: data pre-processing, prediction mode, and parameter tuning. The Fig.1 demonstrates the workflow of the IoTFL-CYPMLOA technique.

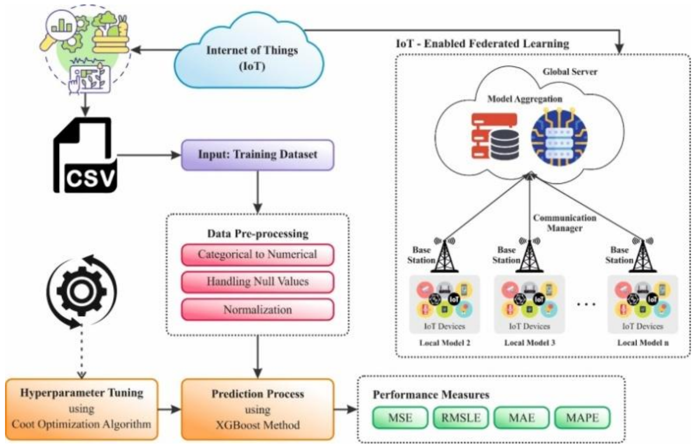

Fig.1. Workflow of IoTFL-CYPMLOA technique

3.1 DATA PRE-PROCESSING

At first, the data pre-processing stage involves various steps such as categorical to numerical, handling null values, and normalization to clean, transform, and organize raw data into a suitable format. To guarantee that the data was ideal for the modelling of ML, a sequence of pre-processing stages was utilized [17]:

- *Processing Missing Values*: ML methods are the methods that might turn out to be biased and/or make low, precise projections when there is missing data. This method might understand a missing value as a particular feature or tendency, making it misleading and vague. The initial stage in the analysis of data of the provided dataset concerned checking for non-numeric and missing data utilizing the isnull() Pandas device function in the programming language of Python. It can be mentioned as NaN, meaning no data exists in these data frame columns. After describing the missing values, the model consented to eliminate the archives with these instances. This model can implement this for the provided data set containing an adequate sum of rows, and some missing rows might not be influenced. The dropna() function from the Pandas DataFrame was utilized to remove rows containing NaNs. This step helps reduce redundant data input before the actual modelling process. However, due to its simplicity, eliminating partial rows was identified as a more suitable approach for this specific dataset.
- *Encoding Categorical Variables*: The inputs utilized in an ML development and training model are arithmetical. This implies that the models applied to the study need to transform qualitative aspects, such as state, crop, and season, into measurable ones. Later, without encoding, the methods would not process these attributes, leading to inefficiency or errors. As the autonomous variable comprised of a definite nature, the utilized encoder category was the label encoder category that only required transforming the classes to numbers. This model sets the integer to all the attribute types, permitting comparison among classes. The process involves intricate mapping of all state names to the integer, guaranteeing that Assam obtained the integer value 1. To implement the encoder, this study applied Scikit-Learn's Label Encoder. This specific model is mainly examined

because of its speed and simplicity in processing characteristic data that needs transformation into an arrangement suitable for ML methods. The other measured encoder category was one-hot, which transforms a feature category into K columns. Thus, all columns are binary variables demonstrating the characteristics in the provided type. Nevertheless, label encoding was chosen as a more efficient alternative due to the drawback of one-hot encoding, which can inflate dimensionality when applied to categorical variables with many possible categories.

- *Normalization*: StandardScaler (a standard scaling function from scikit-learn) is utilized to determine the numeric characteristics for scaling. Normalization moves the average of the data value to 0 and middles the data so that each feature has a similar scale.

The equation applied for standardization is:

$$z = \frac{x-\mu}{\sigma} \tag{1}$$

3.2 PREDICTION USING XGBOOST

IoTFL-CYPMLOA technique employs XGBoost as the prediction method [18]. This is chosen for its high accuracy, efficiency, and robustness with structured data. It utilizes gradient boosting with built-in regularization to prevent overfitting and effectively handles missing values. Compared to models like SVM and RF, XGBoost presents better performance, faster training, and robust generalization. Its ability to highlight crucial features also assists interpretability.

XGBoost is an effective model that incorporates numerous poor learners to enhance the prediction result and method estimate by incorporating numerous trees (usually decision trees). Its basic notion is to utilize a gradient descent optimizer to form novel trees in all iterations and slowly decrease the mistakes of the present method.

As presented in Eq.(2), it is the additive method made from numerous basis methods, namely, by overlaying the estimates of multiple trees to make the last forecast output y_i. At the same time, f_k characterizes the estimation function of the K_{th} tree, x_i is the input instance, and K signifies tree counts.

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in \phi \tag{2}$$

The XGBoost objective function contains dual portions, as demonstrated in Eq.(3): the regularization term $\Omega(f_k)$ and the error term . The term of error calculates the errors amongst the true and the forecast values, and the term of regularization controls the model difficulty. The equation is:

$$\text{Obj} = \sum_{i=1}^{m} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{3}$$

During this equation, y_j refers to the sample being predicted; γ stands for the regularization parameter of the leaf node counts that can be applied to stop the constant nodes splitting; T denotes the tree's depth that represents the leaf node counts within the tree; and λ signifies minimal sample weighting of the leaf node to avoid the leaf node weighting from being too big. w_j symbolizes all leaf node's values.

In the training procedure, the succeeding aims should be reduced in round iteration:

$$\text{Obj}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) \tag{4}$$

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \eta \sum_{k=1}^{K} f_k^{(t)}(x_i) \tag{5}$$

where $\hat{y}_i^{(t-1)}$ and $\hat{y}_i^{(t)}$ denote forecast values in the t-1 and t iteration correspondingly, η represents the learning rate. To avoid overfitting, the model decreases the impact of all trees by adding a penalty to the leaf node counts, preventing the development of leaf nodes, and presenting the rate of learning.

3.3 COA-BASED PARAMETER TUNING

To improve the prediction performance of the XGBoost model, the parameter tuning process is performed using COA [19]. This model is chosen due to its robust global search ability, inspired by the cooperative behaviour of coot birds. It assists in efficiently exploring the solution space. It balances exploration and exploitation, producing optimal hyperparameter settings with fewer iterations. Compared to conventional methods like grid or random search, COA presents faster convergence and better accuracy. This enhances the overall performance and reliability of the prediction model.

Stimulated by the coordinated activities of Coots, a type of marine bird, the COA utilizes a meta-heuristic optimizer model. In this water, they show an extensive range of locomotive patterns designed to bring them to specific food resources or locations. These processes turn into a portion of the architecture of the Coot method. During the performance, the model initiates by random primary majorities, utilizing Eq.(6) as explained in the strategies:

$$\text{CootPos}(i) = \text{rand}(1, N) \times (\text{UB} - \text{LB}) + \text{LB} \tag{6}$$

CootPos(i) refers to the position value of a particular coot; N stands for parameter counts complex or the task's difficulty. LB and UB both act as a representation of the upper and lower limits of the exploration region that is being hunted.

$$\begin{aligned} \text{UB} &= [UB_1, UB_2, \ldots, UB_N], \\ \text{LB} &= [LB_1, LB_2, \ldots, LB_N] \end{aligned} \tag{7}$$

Next, the primary majority arrangement, the coot's locations are successfully updated by utilizing four dissimilar motion designs.

3.3.1 Random Movement:

For this particular motion, the starting location Q task is arbitrarily chosen by applying the process specified in Eq.(8):

$$Q = \text{rand}(1, N) \times (\text{UB} - \text{LB}) + \text{LB} \tag{8}$$

To move away from locally optimal solutions, the position is transformed succeeding Eq.(9):

$$\text{CootPos}(i) = \text{CootPos}(i) + A \times R_2 \times (Q - \text{CootPos}(i)) \tag{9}$$

The value of R_2 is an integer generated at random among (0,1), and A is established by utilizing the process provided in Eq.(10):

$$A = 1 - L \times \left(\frac{1}{\text{Iter}}\right) \tag{10}$$

L embodies the present iteration counts, whereas *Iter* displays the maximal number permitted.

3.3.2 Chain Movement:

The equation in Eq.(11) might be applied to compute the mean location of two coot birds to implement the sequential motion.

$$\text{CootPos}(i) = \frac{\text{CootPos}(i-1) + \text{CootPos}(i)}{2} \tag{11}$$

3.3.3 Fine-tuning Location based on Leader:

In all clusters, the position of a coot bird changes depending on the leader's location, causing the supporter to move nearer to the leader. The equation provided in Eq.(12) can decide the head's designation.

$$K = 1 + (i \bmod \text{NL}) \tag{12}$$

where, i represents the number assigned to the supporter coot bird, K for the index of the leader, and NL for the entire leader counts within this cluster. The location of the coot is upgraded during this particular movement utilizing the equation established in Eq.(13):

$$\begin{aligned}\text{CootPos}(i) = {} & \text{LeaderPos}(K) + 2R_1\cos(2\pi R) \\ & \times(\text{LeaderPos}(K) - \text{CootPos}(i))\end{aligned} \tag{13}$$

CootPos(i) signifies the coot bird's present location, while *LeaderPos*(K) embodies the position of the selected leader. R_1, whereas other random values, specified as R, are chosen from the range [-1, 1], a randomly generated value is designated from the range [0,1].

3.3.4 Leader Movement

According to the concepts offered in Eq.(14), the leader's positions shift from localized best locations to global optimality.

$$\text{LeaderPos}(i) = \begin{cases} B \cdot B_3 \cdot \cos(2\pi R) \cdot & \\ (\text{gBest} - \text{LeaderPos}(i)) + \text{gBest}, & \text{if } B_4 < 0.5 \\ B \cdot B_3 \cdot \cos(2\pi R) \cdot & \\ (\text{gBest} - \text{LeaderPos}(i)) - \text{gBest}, & \text{if } B_4 \geq 0.5 \end{cases} \tag{14}$$

In such a case, randomly chosen values from the range [0,1] are characterized by B_3 and B_4, while g_{best} signifies the optimal position, which might be attained. The equation provided in Eq.(15) is applied to compute the variable value B:

$$B = 2 - L \times \left(\frac{1}{\text{Iter}}\right) \tag{15}$$

COA is applied to determine the parameter contained in the XGBoost model. The MSE is measured as an objective function, and its mathematical formulation is below.

$$\text{MSE} = \frac{1}{T}\sum_{j=1}^{L}\sum_{i=1}^{M}\left(y_j^i - d_j^i\right)^2 \tag{16}$$

While L and M signify the resulting value of data and layer respectively, d_j^i and y_j^i means the appropriate and attained extents for j^{th} unit from the resulting layer in t^{th} time correspondingly. Algorithm 1 specifies the COA model.

Algorithm 1: COA technique

1: **Initialization**: Initialize COOT_POPUL (candidate solutions) randomly within the search space.
2: Set algorithm parameters: coots count, max iterations, etc.
3: **Leader Selection**: Choose a subset of coots as leaders based on fitness (best objective values).
4: **Movement Update**:
 a. **Follower Coots**: Update positions based on leader guidance and random movement.
 b. **Leader Coots:** Adjust positions utilizing a sinusoidal function and memory of best positions.
5: **Fitness Evaluation**: Compute each coot's fitness (objective function value).
6: **Update Best Solution**: Keep track of the global best solution found so far.
7: Repeat steps 2–5 until the maximum number of iterations is reached or convergence criteria are met.
8: **Output**: Return the best solution (optimized parameters).

4. EXPERIMENTAL VALIDATION

The proposed IoTFL-CYPMLOA technique is examined under the crop yield prediction dataset [20]. The method is simulated using Python 3.6.5 on a PC with an i5-8600k, 250GB SSD, GeForce 1050Ti 4GB, 16GB RAM, and 1TB HDD. Parameters include a learning rate of 0.01, ReLU activation, 50 epochs, 0.5 dropouts, and a batch size of 5.

The Fig.2 demonstrates the predicted outcome analysis for actual vs predicted of the IoTFL-CYPMLOA methodology on epoch 50. The figure specifies that the IoTFL-CYPMLOA methodology correctly predicted the result. It is also observed that the predictive outcomes of the IoTFL-CYPMLOA approach are closer to the actual values.

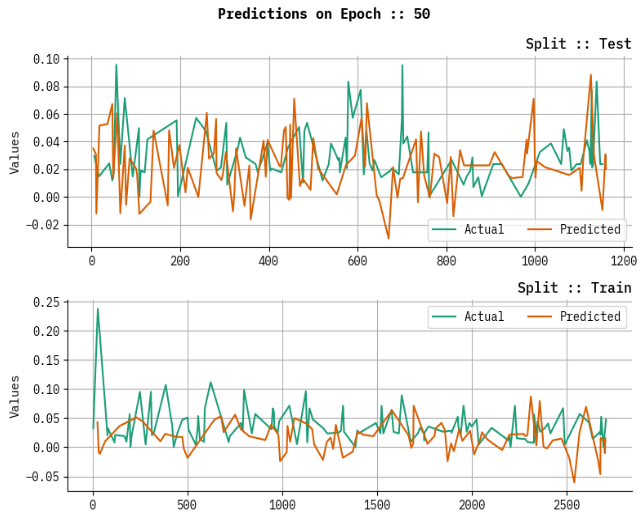

Fig.2. Result analysis for Actual vs Predicted Epoch – 50

The Fig.3 established the predicted outcome analysis for the actual vs. expected results of the IoTFL-CYPMLOA approach on epoch 100. The figure shows that the IoTFL-CYPMLOA method accurately predicted the result. It also implies that the expected solution by the IoTFL-CYPMLOA method is near the actual values.

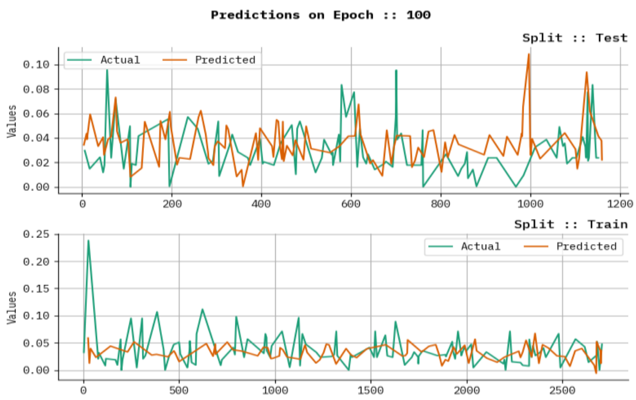

Fig.3. Result analysis for Actual vs. Predicted Epoch – 100

In Fig.4, the TRA loss (TRALOS) and TES loss (TESLOS) analysis of the IoTFL-CYPMLOA technique below epoch 0-100 is exhibited. The values of loss are computed across the range of 0-100 epochs. It is denoted that the TRALOS and TESLOS analyses demonstrate a decreasing tendency, informing the capacity of the IoTFL-CYPMLOA methodology to balance a trade-off between data fitting and generalization. The continuous reduction in loss values promises maximum performance of the IoTFL-CYPMLOA approach and tunes the prediction outcomes over time.

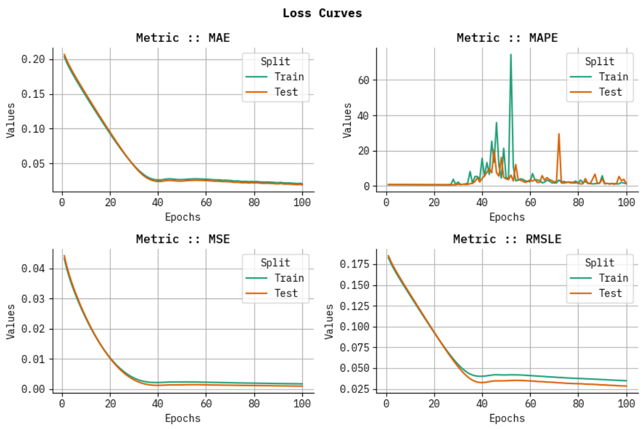

Fig.4. Loss Curve analysis for all metrics with Epoch 0-100

The Table.1 and Fig.5 signify the Iotfl-CYPMLOA technique's training set (TR) and testing set (TS) results below different metrics. With TR, the IoTFL-CYPMLOA technique accomplishes an MSE of 0.0010, RMSLE of 0.0293, MAE of 0.0199, and MAPE of 0.1953. Simultaneously, with TS, the IoTFL-CYPMLOA model achieves an MSE of 0.0026, RMSLE of 0.0407, MAE of 0.0226, and MAPE of 0.2726.

Table.1. TR and TS outcome of IoTFL-CYPMLOA method under various metrics

Metrics	TR	TS
MSE	0.0010	0.0026
RMSLE	0.0293	0.0407
MAE	0.0199	0.0226
MAPE	0.1953	0.2726

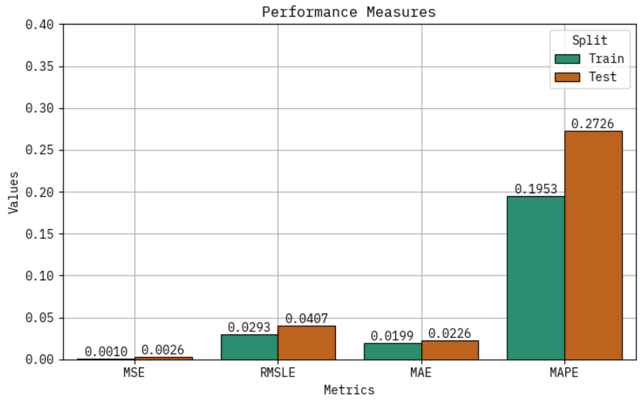

Fig.5. TR and TS outcome of IoTFL-CYPMLOA method under various metrics

The comparison experiment of MSE and MAE of IoTFL-CYPMLOA technique with existing methodologies is ended in Table.2 [21-22]. The Fig.6 delivers the MSE result of the IoTFL-CYPMLOA approachwith existing methods. The table values inferred that the IoTFL-CYPMLOA approach has attained greater performance with an MSE of 0.0010, whereas the existing methods ANN, MLR, 2DCNN+FC+FC, GLM, SVM, KNN, and BRNN have achieved better MSE of 0.0501, 0.0435, 0.0374, 0.0310, 0.0231, 0.0154, and 0.0077, correspondingly.

Table.2. Comparative Results of the IoTFL-CYPMLOA method with existing models

Method	MSE	MAE
ANN	0.0501	0.0672
Multiple linear regression	0.0435	0.0605
2DCNN+FC+FC	0.0374	0.0545
GLM Model	0.0310	0.0490
SVM	0.0231	0.0415
KNN	0.0154	0.0338
BRNN	0.0077	0.0268
IoTFL-CYPMLOA	0.0010	0.0199

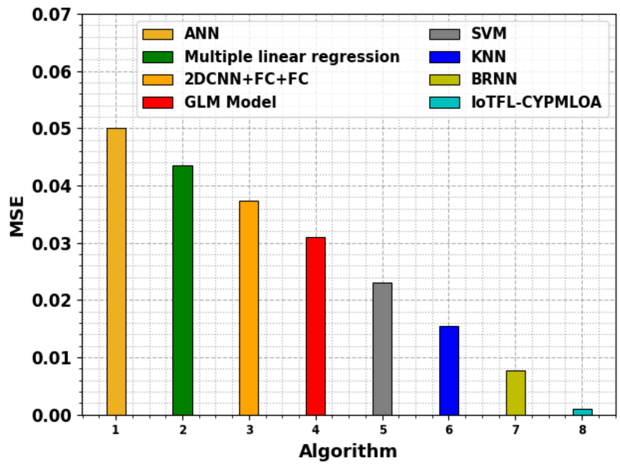

Fig.6. MSE outcome of IoTFL-CYPMLOA method with existing models

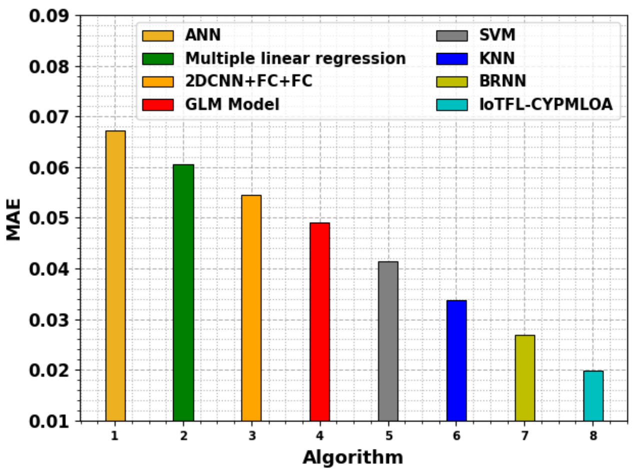

Fig.7. MAE outcome of IoTFL-CYPMLOA method with existing models

The Fig.7 provides the MAE result of the IoTFL-CYPMLOA approach with other approaches. The table values inferred that the IoTFL-CYPMLOA method has attained higher performance with MAE of 0.0199, where the existing techniques ANN, MLR, 2DCNN+FC+FC, GLM, SVM, KNN, and BRNN have gained maximal MSE of 0.0672, 0.0605, 0.0545, 0.0490, 0.0415, 0.0338, and 0.0268, respectively.

5. CONCLUSION

In this manuscript, the IoTFL-CYPMLOA technique in the agriculture sector is proposed. The IoTFL-CYPMLOA technique mainly focuses on enhancing the crop yield prediction model that aids agricultural production improvements. Initially, the data pre-processing stage involves various steps such as categorical to numerical, handling null values, and normalization to clean, transform, and organize raw data into a suitable format. Also, the XGBoost method is employed for prediction. The parameter tuning process is performed using COA to improve the prediction performance of the XGBoost model. The experimental evaluation of the IoTFL-CYPMLOA technique takes place using a benchmark dataset. The simulation outcomes implied the enhanced performance of the IoTFL-CYPMLOA technique compared to recent approaches. The limitations of the IoTFL-CYPMLOA technique comprise limited validation across diverse climatic and geographical regions, which may affect the generalizability of results. It also relies on static input features without accounting for real-time variability in agricultural conditions. Further work may explore dynamic data integration and broader dataset coverage to improve adaptability and accuracy.

REFERENCES

[1] S. Mehta, V. Kukreja and A. Gupta, “Revolutionizing Maize Disease Management with Federated Learning CNNs: A Decentralized and Privacy-Sensitive Approach”, *Proceedings of International Conference for Emerging Technology*, pp. 1-6, 2023.

[2] P. Kumar, G.P. Gupta and R. Tripathi, “PEFL: Deep Privacy-Encoding-based Federated Learning Framework for Smart Agriculture”, *IEEE Micro*, Vol. 42, No. 1, pp.33-40, 2021.

[3] P.S. Nishant, P.S. Venkat, B.L. Avinash and B. Jabber, “Crop Yield Prediction based on Indian Agriculture using Machine Learning”, *Proceedings of International Conference for Emerging Technology*, pp. 1-4, 2020.

[4] R. Medar, V.S. Rajpurohit and S. Shweta, “Crop Yield Prediction using Machine Learning Techniques”, *Proceedings of International Conference for Convergence in Technology*, pp. 1-5, 2019.

[5] F. Abbas, H. Afzaal, A.A. Farooque and S. Tang, “Crop Yield Prediction through Proximal Sensing and Machine Learning Algorithms”, *Agronomy*, Vol. 10, No. 7, pp. 1-16, 2020.

[6] M. Rashid, B.S. Bari, Y. Yusup, M.A. Kamaruddin and N. Khan, “A Comprehensive Review of Crop Yield Prediction using Machine Learning Approaches with Special Emphasis on Palm Oil Yield Prediction”, *IEEE Access*, Vol. 9, pp. 63406-63439, 2021.

[7] P.M. Gopal and R. Bhargavi, “A Novel Approach for Efficient Crop Yield Prediction”, *Computers and Electronics in Agriculture*, Vol. 165, pp. 1-7, 2019.

[8] S. Khaki, L. Wang and S.V. Archontoulis, “A CNN-RNN Framework for Crop Yield Prediction”, *Frontiers in Plant Science*, Vol. 10, pp. 1-6, 2020.

[9] A.X. Wang, C. Tran, N. Desai, D. Lobell and S. Ermon, “Deep Transfer Learning for Crop Yield Prediction with Remote Sensing Data”, *Proceedings of the International Conference on Computing and Sustainable Societies*, pp. 1-5, 2018.

[10] F.V. Naranjo, F.C. Galarza and E.J. Arias, “Internet of Things based Predictive Crop Yield Analysis: A Distributed Approach”, *Full Length Article*, Vol. 13, No. 2, pp. 106-113, 2023.

[11] A. Abu-Khadrah, A.M. Ali and M. Jarrah, “An Amendable Multi-Function Control Method using Federated Learning for Smart Sensors in Agricultural Production Improvements”, *ACM Transactions on Sensor Networks*, pp. 1-23, 2023.

[12] A. Oikonomidis, C. Catal and A. Kassahun, “Deep Learning for Crop Yield Prediction: A Systematic Literature Review”, *New Zealand Journal of Crop and Horticultural Science*, Vol. 51, No. 1, pp. 1-26, 2023.

[13] M.K. Senapaty, A. Ray and N. Padhy, “Enhancing Soil Fertility Prediction through Federated Learning on IoT-Generated Datasets with a Feature Selection Perspective”, *Engineering Proceedings*, Vol. 82, No. 1, pp. 1-12, 2024.

[14] A. Zafar, “Integration of IoT and Machine Learning for Predictive Analytics in Smart Farming: Techniques, Challenges and Future Directions”, *Journal of Advanced Computing Systems*, Vol. 4, No. 7, pp. 1-10, 2024.

[15] H. Najjar, M. Miranda, M. Nuske, R. Roscher and A. Dengel, “Explainability of Sub-Field Level Crop Yield Prediction using Remote Sensing”, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, pp. 1-21, 2025.

[16] R.J. Mohan, P.S. Rayanoothala and R.P. Sree, “Next-gen agriculture: Integrating AI and XAI for Precision Crop Yield

Predictions", *Frontiers in Plant Science*, Vol. 15, pp. 1-16, 2025.

[17] J. Basit, H. Arshad and A. Bibi, "Optimizing Crop Yield Forecasts using Quantum Machine Learning Techniques with High-Dimensional Soil and Weather Data", *Journal of Computers and Intelligent Systems*, Vol. 3, No. 1, pp. 1-15, 2025.

[18] H. Xu, S. Yan, X. Qin, W. An and J. Wang, "Interpretable Intelligent Fault Diagnosis for Heat Exchangers based on SHAP and XGBoost", *Processes*, Vol. 13, No. 1, pp. 1-16, 2025.

[19] L.I. Guo and H. Sheng, "Hybrid Models for Forecasting Allocative Localization Error in Wireless Sensor Networks", *International Journal of Cognitive Computing in Engineering*, Vol. 6, pp. 333-350, 2025.

[20] "Crop Yield Prediction Dataset", Available at https://www.kaggle.com/datasets/ysnreddy/crop-yield-prediction-dataset, Accessed in 2023.

[21] T. Hu, Z. Liu, R. Hu, L. Zeng, K. Deng, H. Dong, M. Li and Y.J. Deng, "Yield Prediction Method for Regenerated Rice based on Hyperspectral Image and Attention Mechanisms", *Smart Agricultural Technology*, Vol. 12, pp. 1-12, 2025.

[22] C. Kumar, J. Dhillon, Y. Huang and K. Reddy, "Explainable Machine Learning Models for Corn Yield Prediction using UAV Multispectral Data", *Computers and Electronics in Agriculture*, Vol. 231, pp. 1-15, 2025.