Journal of Theoretical and Applied Information Technology
30th November 2024. Vol.102. No. 22
© Little Lion Scientific

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

ADVANCING CROP YIELD PREDICTION THROUGH MACHINE AND DEEP LEARNING FOR NEXT-GEN FARMING

UDAYA KUMAR ADDANKI[1] , TEJASWI MADDINENI[2], VIJAY DHAWALE[3], M.L.M.PRASAD[4], DESIDI NARSIMHA REDDY[5], JEEVAN JALA[6]

[1]Assistant Professor, Department of Computer Science and Engineering, GVP College of Engineering for Women(A), Visakhapatnam, AP, India.
[2] Data Engineer, Independent Researcher, Southern Illinois University, Carbondale, USA.
[3]Assistant Professor, Department of MCA, K k Wagh Institute of Engineering Education and Research, Nashik, Maharashtra, India.
[4] Associate Professor of CSE(AI&ML), Joginpally BR Engineering College, Hyderabad, India.
[5] Data Consultant (Data Governance, Data Analytics: enterprise performance management, AI&ML), Soniks Consulting LLC, Texas, United States.
[6]Assistant Professor, Department of Computer Science And Applications, Koneru Lakshmiah Educational Foundation, Vijayawada, AP, India.
E-mail: *[1]udayaka.18@gmail.com, [2]tejas.maddineni@gmail.com, [3]vrdhawale@kkwagh.edu.in, [4]mlm.prasad@yahoo.com, [5]dn.narsimha@gmail.com, [6]jeevanjala@gmail.com.

ABSTRACT

Agriculture has contributed to India's GDP, accounting for 15-18% of the economy. However, Indian agriculture faces persistent challenges threatening its long-term stability, including soil degradation, pest management issues, and fluctuating crop prices. These challenges create significant uncertainty in crop yields. To address this, we propose data-driven solutions using machine learning and deep learning models to improve the accuracy of crop yield predictions. Machine learning models, such as decision trees, random forests, gradient boosting, and ensemble techniques like XGBoost, along with deep learning models like convolutional neural networks (CNNs) and long short-term memory (LSTM) networks, provide reliable predictions and precise forecasting, enabling farmers to achieve more stable and optimized yields. Beyond economic benefits, accurate crop prediction also enhances food security and strengthens rural economies. By advancing precision in agricultural forecasting, these methods can help tackle longstanding agricultural issues, contributing to economic growth, increased profits, and a stable food supply for India. Embracing data-driven approaches is essential to addressing the evolving challenges of the nation's agricultural sector.

Keywords: *Deep learning, Machine learning, Agriculture, LSTM, CNN, Random Forest, Decision tree, Gradient Boosting, XGBoost*

1. INTRODUCTION

Agriculture supports millions of people, is the foundation of the Indian economy, and contributes to the strength and stability of the nation. Despite advancements in the agricultural sector, much still needs to be done. Innovative predictive models such as machine learning (ML) and deep learning (DL) are gaining importance in addressing these challenges [1]. Over 50% of India's workforce was employed in agriculture as of 2018, contributing around 17-18% to the country's GDP [2]. India's total land area has remained relatively constant at around 328,726 thousand hectares from 1971 to 2020. With advanced technologies, farmers can better navigate the complexities of crop growth, which are influenced by variables like soil conditions, weather patterns, and pest infestations.

In the past, agricultural forecasting relied on simple statistical models or expert judgment. These methods, while helpful, often fail to account for the intricate and dynamic factors that influence crop production. New developments in deep learning and machine learning models have made it possible to create increasingly complex forecasting systems that make use of massive information, such as historical yields, soil health, and climate data. With their precise forecasts, these technologies may greatly increase agricultural output by assisting farmers with crop choices, planting plans, and harvesting techniques [3]. Furthermore, forecasting models are being utilized as early warning systems to mitigate risks such as pests and extreme weather events, thus reducing crop losses and improving efficiency.

Real-time monitoring of crop health, soil moisture, and climatic variables is now possible thanks to machine learning algorithms and remote sensing technology like drones and satellite sensors [4]. This integration makes precision farming possible, optimizing irrigation and fertilizer usage, increasing crop yields, and minimizing resource waste. While adopting these technologies faces some hurdles, such as policy challenges and capacity-building needs, these advancements have considerable potential to transform Indian agriculture.

In order to improve agricultural forecasting and increase sustainability and resilience, this study suggests using sophisticated machine learning and deep learning models. Enabling farmers to realize their crops' full potential is the goal. In order to do this, hybrid CNN-LSTM models are investigated for agricultural yield prediction, signaling a move away from conventional incremental approaches and toward deeper, data-driven insights.

2. LITERATURE SURVEY

N. Gandhi et al. [5] proposed a machine learning (ML) model using Support Vector Machines (SVM) for predicting rice crop yields in India. This model aims to provide timely predictions that can aid farmers and policymakers in resource allocation and agricultural planning. However, the model faces challenges in model tuning, and its practical implementation is more complex than anticipated. Additionally, the dataset used for model training could be expanded to include more diverse regions to enhance its generalizability.

P. S. Maya Gopal et al. [20] developed a model that compares several ML algorithms (SVR, K-Nearest Neighbors, Random Forest, and ANN) to identify the best feature subset for crop yield prediction. While the model provides valuable insights into algorithm performance, it has been limited by testing on a single dataset. Furthermore, the lack of specific details about the features considered and the failure to explore a broader set of datasets reduces the robustness of the findings.

M. Khan et al. [6] developed a machine-learning approach to forecast irrigation runoff volume. The key disadvantage is that actual implementation is challenging, owing to the possibility of overfitting the model when applied to real-world data. A more complete method might alleviate this problem, incorporating hyperparameter adjustment and cross-validation on many data sources.

M. Khan and S. Noor [7] evaluated several regression-based ML algorithms for predicting runoff time. However, the model has limitations regarding sensitivity to input variables and its applicability to varied datasets. Expanding the range of datasets used and enhancing feature selection could improve the model's robustness and accuracy.

M. Kavita and P. Mathur [8] explored multiple ML approaches for crop yield prediction, including linear, lasso, ridge regression, and decision trees. They found that while some methods underperformed compared to decision trees, the overall predictions lacked consistency in accuracy. A more granular evaluation of the feature's importance and performance could enhance the model's effectiveness.

A. Suruliandi et al. [9] proposed a model using soil and environmental characteristics to predict crop yield, employing feature selection techniques and a Sequential Minimal Optimization Classifier (SMOC). The model's performance was lacking, particularly with multilayer perceptron (MLP) configurations, as the results exhibited lower accuracy and quality. Future research could explore other classifiers or hybrid models to overcome these limitations.

S. Kunchakuri et al. [10] introduced a KNN-based model using crop type and production area features for crop yield prediction. The model's drawback is its reliance on a limited set of features, which may fail to capture critical factors affecting crop yield. Additional variables, such as climatic data or soil health indicators, could significantly enhance the model's predictive power.

D. J. Reddy and M. R. Kumar [11] proposed a model based on regression techniques like Linear and Support Vector Regression using data such as cultivable land, canal length, and the number of tanks. The primary limitation of this model is the small dataset size and reliance on a single algorithm

(Random Forest), which limits its applicability across different geographical regions.

T. van Klompenburg et al. [12] created machine learning algorithms to estimate agricultural yields using meteorological variables, soil parameters, and historical crop data.. Despite the potential, the model faced challenges with data availability, environmental changes sensitivity, and overfitting risks. More comprehensive datasets with broader regional coverage could improve model accuracy.

S. Bhansali et al. [13] integrated K-Fold validation, decision trees, and Naïve Bayes algorithms with soil and weather data for crop yield and disease prediction [14]. The main critique of this model is that the specific methodologies employed were not clearly explained, which raises concerns about replicability and reliability. A more transparent description of the methodologies and datasets would strengthen the model's credibility [15].

Z. Guo, Q. Liu, and J. Wang [16] introduced a model that employs linear equality constraints alongside a single-layer recurrent neural network for predicting agricultural yields. The fundamental disadvantage of this model is that it does not specify the dataset utilized, which makes it difficult to assess its applicability to other agricultural scenarios.

K. He et al. [17] introduced residual learning for training deep neural networks in image recognition tasks, with the ResNet architecture aimed at addressing vanishing gradient problems. Despite its innovation, the primary disadvantages encompass heightened architectural complexity, elevated memory consumption, and the potential for overfitting, especially in crop yield prediction scenarios where training data may be scarce.

S. Khaki and L. Wang [18] proposed a deep learning model to predict crop yield production using the Syngenta Crop Challenge 2018 dataset. However, the model requires high prediction accuracy and faces challenges with its "black-box" nature, which makes it difficult for stakeholders to interpret results. Overfitting is also a concern, highlighting the need for model refinement.

Maya Gopal and R. Bhargavi [19] proposed a model utilizing Filter and Wrapper methodologies to enhance crop output prediction by the selection of the most pertinent characteristics. [20] The algorithm employs an artificial neural network to predict soybean and corn yields in adverse situations. However, the model's limitation lies in ignoring complex feature interactions and relying on a narrow feature selection set.

F. Raimundo et al. [21] proposed an LSTM-based model for soybean yield prediction using satellite data from Southern Brazil. While LSTM outperforms other algorithms for most forecasts, it fails to deliver superior results in some cases (e.g., DOY 16). Expanding the variety of data sources and including more environmental factors may improve the model's performance.

H. Mureșan and M. Oltean [22] developed deep-learning models for fruit classification using images. The main challenges included handling the variability in fruit appearance and the risk of overfitting, especially in less controlled environments. Further model refinement and better handling of image diversity could improve accuracy.

N. Bali and A. Singla [23] presented a model employing ANN, SVM, and KNN to forecast agricultural yield based on survey data. However, the model's lack of original research and limited analysis of the specific algorithms hindered its ability to provide actionable insights. Future work should focus on deeper analysis and more extensive validation.

P. Mohan and K. Patil [24] proposed using a parallel layer regression model paired with a deep belief network to estimate agricultural production in Karnataka. This model's accuracy is heavily dependent on data quality and availability, emphasizing the need for improved data management and more consistent input characteristics.

E. Khosla et al. [25] presented a model for agricultural production prediction utilizing aggregated rainfall data with ANN and SVR approaches. While the model showed high accuracy (97.5%) during the Kharif season, its performance during other seasons is not well-documented. Expanding the dataset to include more seasonal variations could improve model robustness.

S. Agarwal and S. Tarar [26] introduced a hybrid approach combining ML and deep learning algorithms, incorporating soil parameters, climate data, crop yields, and costs. The model's limitation is its reliance on a limited dataset, raising concerns about its applicability and generalizability.

C. H. Vanipriya, Maruyi, S. Malladi, and G. Gupta [27] proposed a project using ML and IoT algorithms for hydroponic farming systems. However, the complexity and high maintenance costs associated with the model pose significant implementation challenges. Simplifying the system could make it more viable for real-world application.

A. Tomar et al. [28] reviewed plant disease detection using ML models with leaf images as input data. The review highlighted the limited scope, focusing solely on leaf-based methods, which may not apply to other plant species or environments. A broader approach considering other factors like environmental conditions could improve its applicability.

R. Gupta and A.K. Sharma [29] used big data analytics and the MapReduce architecture to forecast agricultural yields in India based on meteorological data. The model's drawback is its focus on a limited number of crops and regions, limiting its applicability across different farming contexts in India.

Annual Report 2020–21 (NIC IN) [30] provided an overview of the organization's activities and achievements, but the report faced issues with incomplete information, selective reporting, and limited stakeholder engagement. A more comprehensive and transparent report could improve stakeholder trust and decision-making.

Crop Production Statistics Information System (Govt India) [31] serves as a centralized platform for crop production data but faces accessibility barriers, challenges in data integration, and privacy concerns. Improving data accessibility and integrating various data sources could enhance the platform's effectiveness.

Agriculture & Farmer Welfare Ministry [32], India introduced initiatives to improve agriculture practices and farmer welfare but faced issues with incomplete project reports, which hindered a complete understanding of the results. Future initiatives could benefit from more detailed and transparent reporting.

The literature review on crop yield prediction demonstrates the efficiency of machine learning models' efficiency in utilizing different data, including weather conditions, soil qualities, and satellite images, such as decision trees, support vector machines, and neural networks. However, gaps remain in dataset diversity, model generalization, and the exploration of feature interactions. Many studies suffer from overfitting, lack hyperparameter tuning, and fail to provide model interpretability, hindering practical adoption. Additionally, real-time data integration, seasonality considerations, and hybrid model exploration are underexplored, limiting model robustness. Comparative research, higher data quality, and scalable solutions are also required to improve the practical usefulness of these models in real-world agricultural contexts. Addressing these gaps would improve crop prediction systems' reliability, transparency, and scalability.

3. METHODOLOGY

3.1 Data Sources

This research uses two datasets from Kaggle: one for agricultural yield prediction and another for crop recommendation. The datasets span from 1997 to 2020 and contain columns such as crop type, crop year, season, state, area, production, rainfall, fertilizer consumption, pesticide application, and yield. The main objective is to estimate agricultural productivity via machine learning techniques, including Decision Trees, Gradient Boosting, Random Forest, XGBoost, and a meta-model. Deep learning methodologies are investigated, including convolutional neural networks (CNN), long short-term memory networks (LSTM), and a hybrid model that combines numerous techniques.

It was divided into training and testing subsets to prepare the data for model training, with 80% for training and 20% for testing. Model efficacy is measured using evaluation criteria such as R-squared, Mean Absolute Error (MAE), and Root Mean Square Error (RMSE). These criteria assist evaluate each algorithm's dependability and accuracy in predicting crop yield.

3.2 Methods

3.2.1 Machine learning methods

3.2.1.1 Decision trees

Decision trees work by segmenting the feature space into discrete regions based on the values of various attributes. At each node, the algorithm chooses a characteristic and threshold that separates the data most effectively into two groups to enhance homogeneity within the target variable across both groups. This procedure, known as recursive binary splitting, continues as the dataset is separated into two subsets at each node based on the feature and threshold selected. The procedure terminates when no more gains of inhomogeneity are achievable or other stopping requirements, such as reaching a maximum depth, are fulfilled. Once a stopping requirement is met, the model generates a leaf node and assigns it a forecast value. In regression tasks such as crop yield prediction, the prediction for a leaf node is generally the average of all target values in the training samples associated with that node. As a result, a decision tree's prediction for a particular input is calculated by passing it from the root to a leaf node, where the mean of the target values of all training samples in that leaf is used as the final forecast.

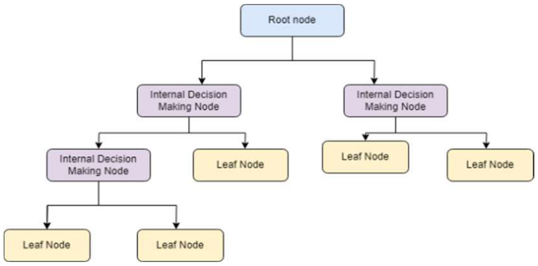

Root node
Internal Decision Making Node
Internal Decision Making Node
Internal Decision Making Node
Leaf Node
Leaf Node
Leaf Node
Leaf Node
Leaf Node

Fig 1: Decision Tree

A decision tree T's prediction $\hat{y}^T$ For xi can be found by passing through the root towards some leaf node and then taking mean values of targets overall training samples falling into such leaves:

$$\widehat{y_i}^T = \frac{1}{N_j}\sum_{x_j \in \, leaf\ node} y_j \qquad (1)$$

Where N_j represents the number of training samples in the leaf node while y_j denotes the jth observation's response variable.

3.2.1.2 Random forest

Random Forest is a decision tree-based model that divides the feature space into regions to forecast the target variable. The Random Forest approach generates many decision trees by sampling the training data with replacement, also known as bootstrapping. Each decision tree is trained on a distinct subset of the original data, limiting overfitting and giving a variety of models [33]. This method enhances the model's stability and accuracy. Random Forest only takes into account a random subset of characteristics at each split in a decision tree. This strategy improves forest diversity since each tree learns from a unique mix of elements [34]. The number of attributes reviewed at each split is typically the square root or logarithm of the total number of features, promoting variation among the trees.

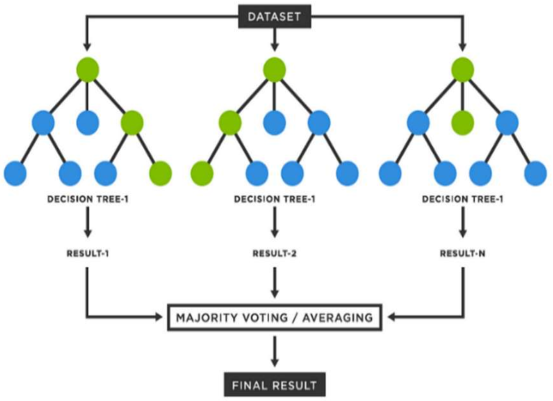


Fig 2: Random Forest

3.2.1.3 Gradient boosting

Gradient Boosting is used in regression applications like crop production prediction to reduce a loss function, which is often the mean squared error (MSE). This loss function quantifies the difference between the actual and projected values, allowing the model to optimize throughout training. The loss function's negative gradient (or derivative) is determined at each procedure stage based on the model's predictions for each data point. This gradient indicates how the forecasts should be adjusted to minimize the loss. It shows how much the model's predictions must alter to enhance performance.

A weak learner, such as a decision tree, is trained to predict the loss function's negative gradient in order to fit the model [35]. This new model focuses on fitting the residuals, the disparities between actual and predicted values from earlier models. By focusing on these residuals, the model hopes to remedy the errors created by previous forecasts.

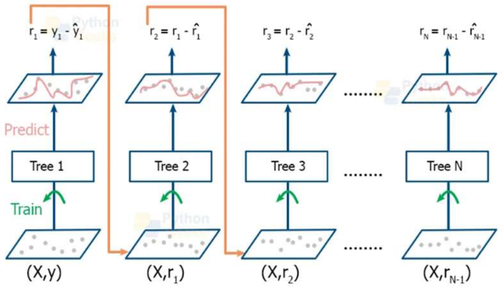


Fig 3: Gradient Boosting

The final forecast is created by averaging the estimates from each model in phases. Each succeeding model is trained to minimize the loss function while incorporating the predictions of the previous models. This cumulative technique enables Gradient Boosting to improve the model's predictions iteratively.

The gradient of the loss function with respect to predicted values $\hat{y}_i$ is given by:

$$Gradient = -\frac{\partial_{Loss}}{\partial_{\hat{y}_i}} \quad (2)$$

For a data point xi, let $\hat{y}_i$ be prediction made by the Gradient Boosting model, which is an aggregate overall individual model:

$$\hat{y}_i = \sum_{K=1}^{K} f_k(x_i) \quad (3)$$

3.2.1.4 XGBOOST

XGBoost is intended to reduce the mean square error loss function in regression problems. The mathematically described objective function is as follows:

$$obj = \sum_{i=1}^{n} \frac{1}{2}(y_i - \hat{y})^2 \quad (4)$$

This function quantifies the error between the actual values y_i and the predicted values $\hat{y}$, driving the optimization process.

In each iteration, XGBoost computes the slope of the loss function in respect to the predicted values. The gradient, which specifies the direction of correction, is calculated as follows:

$$Gradient = -\frac{\partial_{obj}}{\partial_{\hat{y}_i}} = y_i - \hat{y}_i \quad (5)$$

This gradient is essential for updating the model since it explains how the predictions should be changed to decrease loss. XGBoost creates trees replicating the loss function's negative gradient to develop the model. Each new tree is exceptionally trained to forecast residuals, the disparities between actual values and the prior ensemble of tree predictions. This iterative strategy allows the model to refine its predictions gradually.

The prediction $\hat{y}_i$ for a data point x_i according to the XGBoost model is derived by summing the forecasts of all individual trees:

$$\hat{y}_i = \sum_{K=1}^{K} f_k(x_i) \quad (6)$$

Where K denotes the total number of trees in the ensemble, this formula demonstrates how the contributions of several trees combine to produce the final forecast, resulting in a more accurate and robust output.

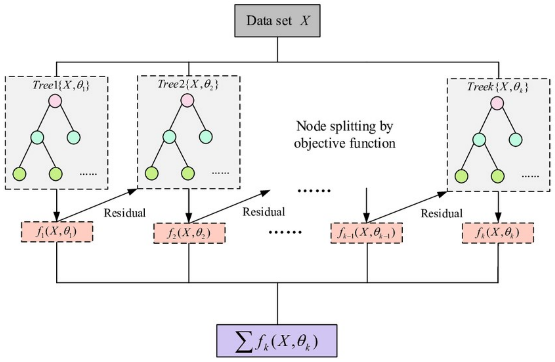


Fig 4: XGBoost

3.2.2 Meta model

In the Meta Model technique, the initial step is to train the base models individually. The specified basic models—Gradient Boosting, XGBoost, and Decision Trees—are all trained with identical training data. Each model is tuned independently to capture distinct parts of the data's underlying patterns.

Once trained, the base models forecast the target variable from test data. The predictions from each base model are then layered horizontally to form a new feature matrix. This matrix effectively stores the predictions from each base model as additional characteristics for the subsequent process stages.

The next step is to train a meta-model, such a Random Forest Regressor [36]. The meta-model receives base model predictions as input characteristics and the actual target values. This extra layer of learning enables the meta-model to integrate the capabilities of the basis models and produce more accurate predictions.

Finally, the meta-model employs the base models' stacked predictions to generate its predictions for the test data. This ensemble technique improves overall performance by exploiting the pooled insights of all base models, as shown in Figure 5.

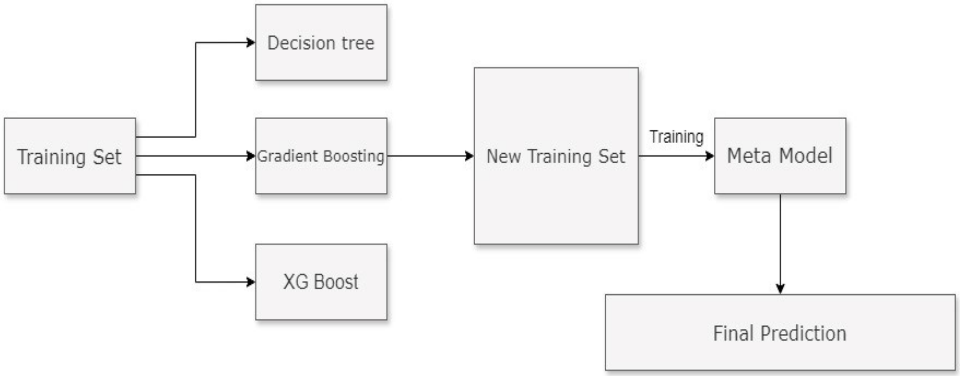


Fig 5: Meta model

3.2.3 Deep Learning Methods

3.2.3.1 Convolutional neural network

Convolutional layers of a Convolutional Neural Network (CNN) identify significant characteristics from incoming data using filters (or kernels). The convolution operation can be mathematically expressed as:

$$y_{i,j} = \sum_{m,n} X_{i+m,j+n} \times W_{m,n} \qquad (7)$$

An Activation Function such as ReLU introduces nonlinearity, defined as:

$$ReLU(X) = \max(0, X) \qquad (8)$$

Pooling layers are then used to downsample the feature maps, lowering their spatial dimensions while retaining critical information. For maximum pooling:

$$\text{Max}\, Pooling = \max(X_{i,j}) \qquad (9)$$

The flattened layer converts the 2D feature mappings into a 1D vector before passing them through fully linked (dense) layers that learn the final mapping onto the target variable. This procedure may be stated numerically as:

$$Y = \text{ReLu}\,(XW + b) \qquad (10)$$

A dropout layer randomly sets some input units to zero during training to avoid overfitting. Lastly, backpropagation and optimization techniques like Adam are used to train the model, and the mean squared error (MSE) loss function evaluates the discrepancy between expected and actual values in regression tasks.

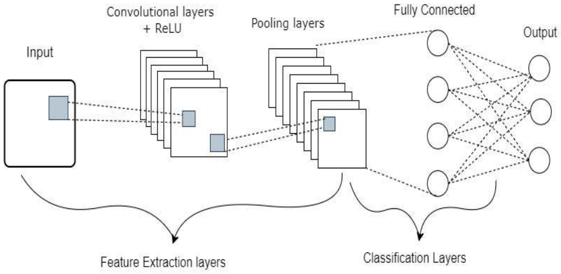


Fig 6: Convolutional Neural Networks

3.2.3.2 LONG SHORT-TERM MEMORY

A recurrent neural network (RNN), the LSTM (Long Short Term Memory), learns long-term relationships in data sequences and addresses the vanishing gradient problem. An LSTM cell comprises three essential gates:

- **Forget Gate (f_t):** Determine what information to discard from the cell state.
- **Input Gate (i_t):** Controls which values in the cell state are to be updated [37].
- **Output Gate (o_t):** Determines which section of the cell will be outputted.

In order to control flow through them, LSTMs use activation functions like sigmoid or hyperbolic tangent function(tanh):

- Sigmoid Function (σ): Squash values between 0 and 1, controlling the gate activations.
$$\sigma(x) = \frac{1}{1+e^{-x}} \qquad (11)$$
- Hyperbolic Tangent Function (tanh): Squashes values between -1 and 1, regulating the input and output transformations.
$$\tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \qquad (12)$$

The LSTM cell state (C_t) which is the current memory cell content depends on the previous state (C_{t-1}) and current input (x_t), Multiple LSTM cells can be connected in layers such that each layer receives its inputs from the preceding one or dense for prediction.

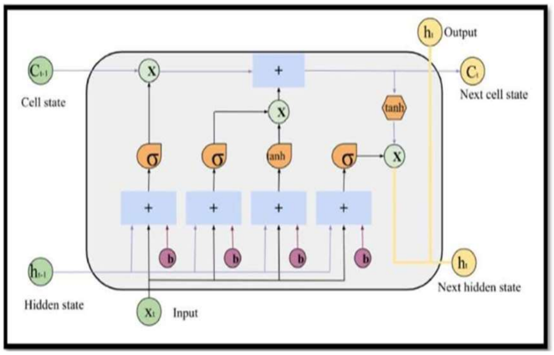


Fig 7: Long Short-Term Memory

3.2.3.3 Hybrid model (CNN, LSTM)

The hybrid model combines CNN and LSTM to exploit spatial feature extraction (CNN) and sequential dependency modeling (LSTM). The CNN component retrieves spatial patterns from the data, whilst the LSTM layers identify temporal correlations. This hybrid approach is ideal for time-series crop prediction tasks.

The CNN layers use filters and pooling to reduce dimensions, and the LSTM layers learn the temporal patterns. The final output is computed using a dense layer:

$$Y = \sigma(X \times W + b) \quad (13)$$

The one-dimensional convolutional layer uses convolutions on the input time series data to pick up spatial patterns. The pooling layer samples the output of the convolutional layer, extracting significant characteristics while minimizing computing complexity. Mathematically, it aggregates information within a pooling window using a pooling operation (e.g., max pooling) [38]:

$$MaxPooling(X) = \max(X) \quad (14)$$

The MaxPooling 1D layer conducts maximum pooling with a predetermined pool size.

LSTM layers identify temporal connections in sequential data [39]. LSTM operations use gate mechanisms (forget gate, input gate, output gate) and activation functions (sigmoid, tanh) to regulate information flow over time. Input sequences are processed by LSTM layers that detect temporal dependencies. Depending on the features extracted from previous layers, these layers are responsible for either classification or regression. To be more precise, if we use an activation function denoted as ∅ then the output of a dense layer can be expressed mathematically like this:

$$Y = \emptyset(XW + b) \quad (15)$$

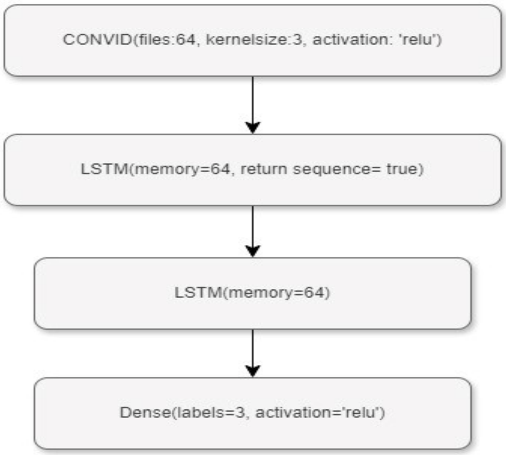

CONVID(files:64, kernelsize:3, activation: 'relu')
LSTM(memory=64, return sequence= true)
LSTM(memory=64)
Dense(labels=3, activation='relu')

Fig 8: Hybrid Model (CNN, LSTM)

Finally, predictions are made by processing outputs from LSTM through some dense layer(s) with activation functions. The model gets compiled using an optimizer and a loss function, both employed during training. This is done by fitting it against train data for certain epochs and particular batch sizes.

4. RESULT AND ANALYSIS

4.1 Performance Evaluation

Various criteria were used to evaluate the effectiveness of each model. These metrics give insights into the models' performance, helping to evaluate how well they match the data and generate predictions:

- **R-squared (R²)**: This measure assesses the degree to which the regression line closely resembles the data points. A better match is indicated by higher values ranging from 0 to 1. The R-squared calculation formula is:

  $$R^2 = 1 - \left(\frac{\text{Sum of Squared Residuals}}{\text{Total Sum of Squares}}\right) \quad (16)$$

- **Mean Absolute Error (MAE)**: The average absolute difference between expected and actual values is measured by this statistic. A lower MAE indicates better model performance. The formula used to determine MAE is:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i^2) \qquad (17)$$

- **Root Mean Square Error (RMSE)**: This metric measures the standard deviation of the residuals in a forecast. Lower RMSE values suggest smaller prediction errors, while higher values suggest larger errors. The formula for calculating RMSE is:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i^2)} \qquad (18)$$

In Figure 9, the chart visually represents the fluctuations in rice harvests over twenty years, highlighting a steady increase in the later years. Most data points emphasize this upward trend above the average line in the final seasons. The line chart shows the annual rice yield between 2000 and 2020. On the x-axis, there is the year of cultivation, while on the y-axis, we have yield in an unknown unit of measurement. The dots are joined by a line, which shows how they change over time. There are some dips and peaks in output during this period, but there is also a sudden rise at the end, meaning that much more rice has been produced lately.

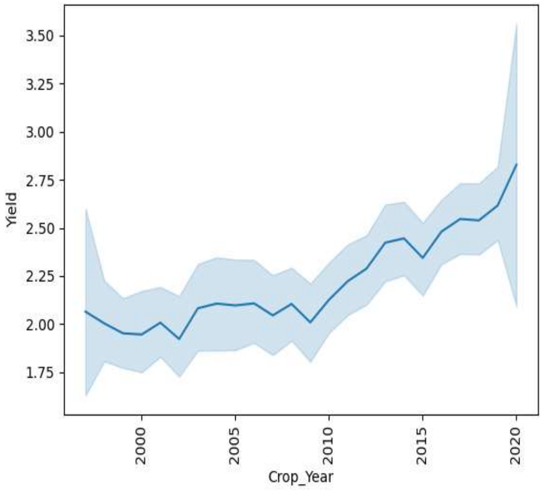


Fig 9: Annual Rice Yield Over Time

Fig 10 below shows the actual versus predicted yield of the meta-model. The graph shows a model against actual yield values at a glance. Actual yield is given on the x-axis, while predicted yield is on the y-axis. Each blue dot represents one point of data: individual yields and their corresponding predictions. The red dashed line captures the trend between actual and predicted values, commonly called the linear regression line. On the other hand, green dots represent outliers where the model's predictions differ greatly from the true values they should take up; these look like an irregular scatter plot for which nearly all points would be near or at the zero height above or below, as shown by a red line.

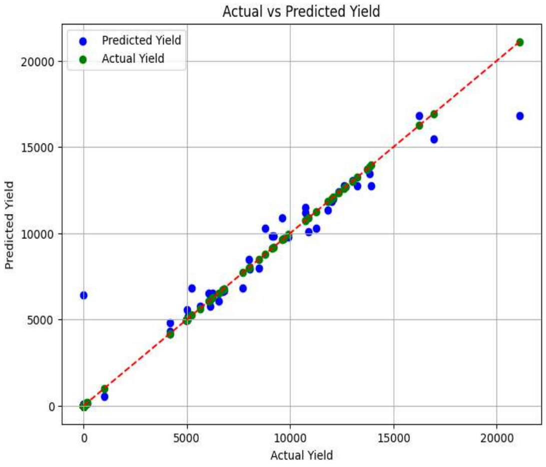
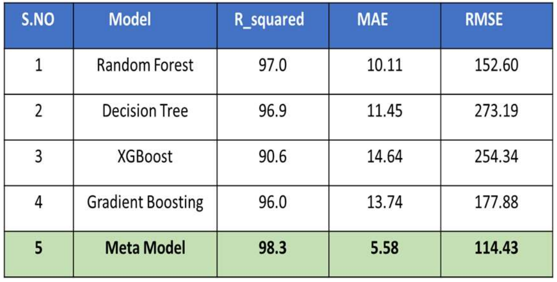


Fig 10: Model performance of meta-model

These metrics are essential for model comparison since they provide numerical values that illustrate the model's performance on specific test datasets. However, it is crucial to recognize that relying on a single measure is insufficient. Multiple metrics—such as R-squared, MAE, and RMSE—must be considered to assess model performance comprehensively.

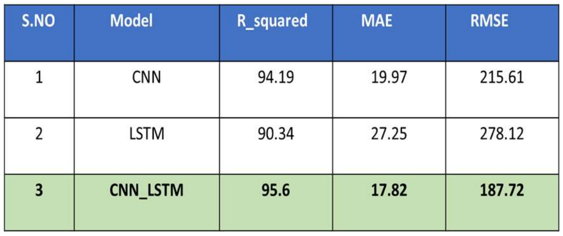
Table 1: Comparison Table of machine learning models



S.NO	Model	R_squared	MAE	RMSE
1	Random Forest	97.0	10.11	152.60
2	Decision Tree	96.9	11.45	273.19
3	XGBoost	90.6	14.64	254.34
4	Gradient Boosting	96.0	13.74	177.88
5	**Meta Model**	**98.3**	**5.58**	**114.43**

Table 2: Comparison Table of deep learning models

Table 1 presents the forecast outcomes. The meta-model exhibited the highest R-squared value of 98.3, meaning it can explain much of the variance in the target variable. In addition, it had the least MAE of 5.58 and RMSE of 114.43 among all other models listed. This means that its predictions have fewer errors compared to base models.



S.NO	Model	R_squared	MAE	RMSE
1	CNN	94.19	19.97	215.61
2	LSTM	90.34	27.25	278.12
3	**CNN_LSTM**	**95.6**	**17.82**	**187.72**

Accuracy and other related metrics are shown in Table 2 for all implemented models, where hybrid CNN-LSTM achieved the highest accuracy with 95.6%.

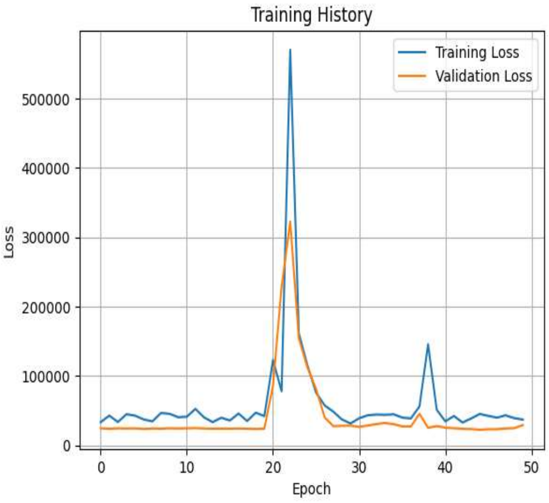


Fig 11: Loss graph of CNN_LSTM

The graph provides the training and validation losses across epochs for ML models. Initially, both losses begin on high notes but steadily diminish after that. Around epoch 15, there's an abrupt increase, suggesting potential overfitting by our model. After that, point loss decreases again, with validation loss settling below training loss; this is what we want to see happening here. This chart shows the model's training progress, which helps identify overfitting or underfitting issues.

4.2 Discussion of Results

The results of the study significantly improve agricultural yield forecasts. Predicting rice yields with machine learning and deep learning models [40] offers a viable way to increase agricultural forecast accuracy. The high R-squared value of the meta-model, alongside its low MAE and RMSE, validates its superiority over base models. This further emphasizes the importance of combining multiple models in a hybrid framework, which can enhance prediction accuracy.

The study's validity can be considered highly robust due to the careful application of performance metrics across different models. The study comprehensively evaluates forecasting techniques by employing multiple models, including deep learning architectures like CNN-LSTM.

Additionally, this study significantly contributes to the overall body of knowledge by demonstrating the potential of hybrid machine-learning approaches to enhance agricultural production forecasts. The models' improved prediction accuracy and performance measures demonstrate how this new information significantly outperforms conventional techniques.

5. CONCLUSION

This paper explored how contemporary machine and deep learning systems hold transformative potential for modern farming, particularly crop prediction. By utilizing advanced algorithms such as decision trees, meta-models, and XGBoost, farmers can gain insights into the various factors impacting crop growth and productivity. This data-driven approach supports informed decision-making, aiming to maximize agricultural yields and improve resource efficiency.

The research on forecasting within next-generation farming systems underscores the revolutionary capability of these methods, especially as they foster the development of accurate models that can capture the complexities of agricultural processes. Ensemble techniques, including Random Forests, Gradient Boosting Machines (GBM), Extreme Gradient Boosting (XGBoost), and meta-models, contribute to model stability and reduce overfitting—critical for reliable crop predictions.

Additionally, by analyzing large datasets encompassing historical crop records, weather patterns, and soil fertility, machine learning models such as convolutional neural networks (CNNs), long short-term memory (LSTM) networks, and hybrid CNN-LSTM models can offer significant advancements in crop yield prediction. These approaches are positioned to drive more sustainable and efficient farming practices, ultimately aiding farmers in managing resources more effectively while adapting to changing environmental conditions.

REFERENCES:

[1] P. Kumar. Pattnaik, Raghvendra. Kumar, and Souvik. Pal, "Internet of Things and analytics for agriculture. Volume 3," 2022.

[2] P. Sharma, P. Dadheech, N. Aneja, and S. Aneja, "Predicting Agriculture Yields Based on Machine Learning Using Regression and Deep Learning," IEEE Access, vol. 11, pp. 111255–111264, 2023, doi: 10.1109/ACCESS.2023.3321861.

[3] C. P. Osorio, F. Leucci, and D. Porrini, "Analyzing the Relationship between Agricultural AI Adoption and Government-Subsidized Insurance," Agriculture 2024, Vol. 14, Page 1804, vol. 14, no. 10, p. 1804, Oct. 2024, doi: 10.3390/AGRICULTURE14101804.

[4] V. M. Vinshon et al., "AI-Equipped IoT Applications in High-Tech Agriculture Using Machine Learning," https://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-6684-9231-4.ch003, pp. 38–64, Jan. 1AD, doi: 10.4018/978-1-6684-9231-4.CH003.

[5] N. Gandhi, L. J. Armstrong, O. Petkar, and A. K. Tripathy, "Rice crop yield prediction in India using support vector machines," in 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), 2016, pp. 1–5. doi: 10.1109/JCSSE.2016.7748856.

[6] M. Khan and S. Noor, "Irrigation Runoff Volume Prediction using Machine Learning Algorithms," vol. 8, p. 23, Feb. 2019.

[7] M. Khan and S. Noor, "Performance Analysis of Regression-Machine Learning Algorithms for Predication of Runoff Time," vol. 8, p. 187, Feb. 2019, doi: 10.24105/2168-9881.8.187.

[8] M. Kavita and P. Mathur, "Crop Yield Estimation in India Using Machine Learning," in 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA), 2020, pp. 220–224. doi: 10.1109/ICCCA49541.2020.9250915.

[9] Suruliandi, M. Ganesan, and s. P. Raja, "Crop prediction based on soil and environmental characteristics using feature selection techniques," Math Comput Model Dyn Syst, vol. 27, pp. 117–140, Jan. 2021, doi: 10.1080/13873954.2021.1882505.

[10] S. Kunchakuri, S. Pallerla, S. Kande, and N. R. Sirisala, "An efficient crop yield prediction system using machine learning algorithm," in 4th Smart Cities Symposium (SCS 2021), 2021, pp. 120–125. doi: 10.1049/icp.2022.0325.

[11] D. J. Reddy and M. R. Kumar, "Crop yield prediction using machine learning algorithm," Proceedings - 5th International Conference on Intelligent Computing and Control Systems, ICICCS 2021, pp. 1466–1470, May 2021, doi: 10.1109/ICICCS51141.2021.9432236.

[12] T. van Klompenburg, A. Kassahun, and C. Catal, "Crop yield prediction using machine learning: A systematic literature review," Comput Electron Agric, vol. 177, p. 105709, Oct. 2020, doi: 10.1016/J.COMPAG.2020.105709.

[13] S. Bhansali, P. Shah, J. Shah, P. Vyas, and P. Thakre, "Healthy Harvest: Crop Prediction And Disease Detection System," 2022 IEEE 7th International conference for Convergence in Technology, I2CT 2022, 2022, doi: 10.1109/I2CT54291.2022.9825446.

[14] M. Rajakumaran, G. Arulselvan, S. Subashree, and R. Sindhuja, "Crop yield prediction using multi-attribute weighted tree-based support vector machine," Measurement: Sensors, vol. 31, p. 101002, Feb. 2024, doi: 10.1016/J.MEASEN.2023.101002.

[15] R. Tripathi et al., "Prediction of rice yield using sensors mounted on unmanned aerial vehicle," Agricultural Research, 2024, doi: 10.1007/S40003-024-00809-4.

[16] Z. Guo, Q. Liu, and J. Wang, "A one-layer recurrent neural network for pseudoconvex optimization subject to linear equality constraints," IEEE Trans Neural Netw, vol. 22, no. 12 PART 1, pp. 1892–1900, Dec. 2011, doi: 10.1109/TNN.2011.2169682.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[18] S. Khaki, L. Wang, and S. V. Archontoulis, "A CNN-RNN Framework for Crop Yield Prediction," Front Plant Sci, vol. 10, p. 492736, Jan. 2020, doi: 10.3389/FPLS.2019.01750/BIBTEX.

[19] P. S. Maya Gopal and R. Bhargavi, "Optimum Feature Subset for Optimizing Crop Yield Prediction Using Filter and Wrapper Approaches," Appl Eng Agric, vol. 35, no. 1, pp. 9–14, 2019, doi: https://doi.org/10.13031/aea.12938.

[20] R. S. Renju, P. S. Deepthi, and M. T. Chitra, "A Review of Crop Yield Prediction Strategies based on Machine Learning and Deep Learning," Proceedings of International Conference on Computing, Communication,

Security and Intelligent Systems, IC3SIS 2022, 2022, doi: 10.1109/IC3SIS54991.2022.9885325.

[21] F. Raimundo, A. Gloria, and P. Sebastião, Prediction of Weather Forecast for Smart Agriculture supported by Machine Learning. 2021. doi: 10.1109/AIIoT52608.2021.9454184.

[22] H. Mureșan and M. Oltean, "Fruit recognition from images using deep learning," Acta Universitatis Sapientiae, Informatica, vol. 10, pp. 26–42, Feb. 2018, doi: 10.2478/ausi-2018-0002.

[23] N. Bali and A. Singla, "Emerging Trends in Machine Learning to Predict Crop Yield and Study Its Influential Factors: A Survey," Archives of Computational Methods in Engineering, vol. 29, Feb. 2021, doi: 10.1007/s11831-021-09569-8.

[24] P. Mohan and K. Patil, Crop production rate estimation using parallel layer regression with deep belief network. 2017. doi: 10.1109/ICEECCOT.2017.8284659.

[25] E. Khosla, D. Ramesh, and R. Sharma, "Crop yield prediction using aggregated rainfall based modular artificial neural networks and support vector regression," Environ Dev Sustain, vol. 22, Feb. 2020, doi:10.1007/s10668-019-00445-x.

[26] S. Agarwal and S. Tarar, "A hybrid approach for crop yield prediction using machine learning and deep learning algorithms," J Phys Conf Ser, vol. 1714, no. 1, Jan. 2021, doi: 10.1088/1742-6596/1714/1/012012.

[27] C. H. Vanipriya, Maruyi, S. Malladi, and G. Gupta, "Artificial intelligence enabled plant emotion xpresser in the development hydroponics system," Mater Today Proc, vol. 45, pp. 5034–5040, Jan. 2021, doi: 10.1016/J.MATPR.2021.01.512.

[28] Tomar, G. Gupta, W. Salehi, C. H. Vanipriya, N. Kumar, and B. Sharma, "A Review on Leaf-Based Plant Disease Detection Systems Using Machine Learning," 2022, pp. 297–303. doi: 10.1007/978-981-16-8248-3_24.

[29] R. Gupta et al., "WB-CPI: Weather Based Crop Prediction in India Using Big Data Analytics," IEEE Access, vol. 9, pp. 137869–137885, 2021, doi: 10.1109/ACCESS.2021.3117247.

[30] WCD, "Annual report 2020-21," 2020.

[31] GOVT india, "Crop Production Statistics Information System (Govt India)," Ministry of statistics and Programme implementation.

[32] IBEF, "Agriculture in India: Industry Overview, Market Size, Role in Development (IBEF)."

[33] D. N. A. C. B. Rodrigues, "Hospitality AI-Driven Customer Journey Analytics: Predicting touchpoints in hotel Customer Journeys," Feb. 2024, Accessed: Nov. 16, 2024.

[34] M. Turhan, "Innovative IGBT-based charging systems for improved submarine battery management," Engineering Science and Technology, an International Journal, vol. 58, p. 101825, Oct. 2024, doi: 10.1016/J.JESTCH.2024.101825.

[35] Courage. Kamusoko, Explainable machine learning for geospatial data analysis : a data centric approach. CRC Press, 2025. Accessed: Nov. 16, 2024.

[36] H. K. . Soni, Sanjiv. Sharma, and G. R. . Sinha, Text and social media analytics for fake news and hate speech detection. CRC Press, 2025. Accessed: Nov. 16, 2024.

[37] Y. Hao et al., "Decoding imaginary handwriting trajectories of multi-stroke characters for universal brain-to-text translation," medRxiv, p. 2024.07.02.24309802, Jul. 2024, doi: 10.1101/2024.07.02.24309802.

[38] Bhanu. Chander, Koppala. Guravaiah, Anoop. B., and G. . Kumaravelan, "Handbook of AI-based models in healthcare and medicine : approaches, theories, and applications," p. 456, 2024.

[39] S. Saradhi, Y. Tian, M. R. Rezaei, and M. Lankarany, "Using a Deep Learning Approach for Model-based Control of Deep Brain Stimulation," bioRxiv, p. 2024.10.29.620970, Oct. 2024, doi: 10.1101/2024.10.29.620970.

[40] Narendra. Khatri, A. Kumar. Vyas, Celestine. Iwendi, and Prasenjit. Chatterjee, "Precision agriculture for sustainability : use of smart sensors, actuators, and decision support systems," p. 471, 2024.