Technical Analysis of Crop Production Prediction Using Hybrid GRU and BiLSTM Model

T. N. Deepthi[1], V. Pavani[2], S. Mounika[3], M. Lakshmi Gayathri[4], K. Akhila[5]
Department of CSE, Vignan's Nirula Institute of Technology and Science for women
Palakaluru, Guntur, 522009, Andhra Pradesh, India.

Abstract

Predicting crop production accurately is crucial for maintaining food security and improving agricultural planning in the face of population growth and climate variability. Even though they are helpful, traditional statistical and classical machine learning models frequently fail to capture the intricate, non-linear, and temporal dependencies found in agricultural datasets. To increase crop yield classification accuracy, this study suggests a hybrid deep learning framework that combines Gated Recurrent Units (GRU) and Bidirectional Long Short-Term Memory (BiLSTM) networks. To ensure data consistency and quality, the dataset is subjected to thorough preprocessing, which includes normalization, categorical encoding, and outlier removal.

The BiLSTM improves bidirectional learning to identify contextual temporal relationships in the data, while the GRU component effectively captures sequential dependencies through update and reset gates. The convergence of accuracy and loss curves, as well as steady gains in precision, recall, and F1-score metrics, affirm that the combined GRU–BiLSTM architecture performs exceptionally well, attaining over 90% accuracy with little over fitting. The outcomes demonstrate how well the hybrid model learns from crop, soil, and climate characteristics, providing a reliable, data-driven tool for precise and scalable crop production forecasting.

Keywords:

Crop production prediction, GRU–BiLSTM hybrid model, time-series forecasting, data preprocessing, and normalization.

1. Introduction:

Ensuring global food security in the context of a growing population and fluctuating climate conditions necessitates precise crop production predictions [1], which is critically important [2]. Recent reports indicate that in 2024, over 295 million individuals are experiencing acute levels of hunger, marking an increase from the prior year [3]. Climate change is expected to reduce global yields of staple crops, thereby increasing the need for technology to enhance agricultural productivity (Priyanka Sharma et al., 2023) [4]. Predictive technologies have evolved from being a beneficial tool to becoming a critical component of resilient and sustainable agricultural applications [5].

The field of crop yield forecasting has rapidly evolved, transitioning from traditional statistical methods to sophisticated Machine Learning (ML) and Deep Learning (DL) architectures [6]. Early research effectively demonstrated the power of ensemble methods, with the Random Forest model achieving high accuracy rates, such as the 98.96% noted by Priyanka Sharma et al. (2023) [7]. However, such models, and even basic Deep Learning architectures like standard Long Short-Term Memory (LSTM), face critical limitations [6]. Existing DL methods often struggle to create a direct, non-linear mapping between raw, complex agricultural data (like weather, soil, and crop parameters) and final yield values [7], often relying heavily on the quality of pre-extracted features (Dhivya Elavarasan & P. M. Durairaj Vincent, 2020) [8]. This inadequacy necessitates the development of advanced models capable of handling the high-dimensional [9], temporal complexity inherent in agricultural time-series data. Durairaj Vincent, 2020) [10]. The inadequacy requires the creation of sophisticated models that can manage the high-dimensional, temporal complexity present in agricultural time-series data [11].

The ongoing pursuit of enhanced efficiency and predictive capability in time-series forecasting has resulted in the development of Quantum Long Short-Term Memory (QLSTM) networks [12] [13]. The QLSTM represents a hybrid model that combines quantum computing principles, including superposition and quantum gates, with the classical LSTM cell architecture [14]. This integration enables the model to utilize a significantly larger quantum feature space for encoding and analyzing complex non-linear dependencies in data that pose computational difficulties for classical LSTMs [15].

The proposed QLSTM model offers a novel approach for predicting crop yields. The QLSTM demonstrates substantial enhancements through the effective application of quantum computation, [16] including faster training convergence and improved predictive accuracy compared to classical LSTMs in complex forecasting tasks [17] This model aims to more effectively capture the subtle, non-linear temporal dynamics among various inputs, including satellite imagery, sensor data, and historical yield records, compared to existing models such as Deep Recurrent Q-Networks (DRQN) [18] [19]. The QLSTM demonstrates a capacity for rapid integration of intricate time-series relationships, positioning it as an advanced framework that can provide the exceptional accuracy and reliability necessary for informed, sustainable agricultural decision-making [20].

2. Literature Survey:

Anita Gehlot et al.'s 2022[1] technical analysis of crop production prediction using machine learning and deep learning suggests the Random Forest and Sequential Deep Learning models[3]. With an R2 score of 0.90, the Random Forest model outperformed the Sequential model in crop yield prediction [21]; however, the Sequential model's incapacity to adequately account for variability among characteristics was a drawback [22].

P. M. Durairaj Vincent et al. (2020)[4] A Deep Recurrent Q-Network (DRQN) was proposed in "Crop Yield Prediction Using Deep Reinforcement Learning Model for Sustainable Agrarian Applications" [23]. Its advantages include a 93.7% accuracy rate, outperforming existing models, while its limitation is that a very long time series can cause gradients to explode or vanish [24].

Priyanka Sharma et al. (2023**)[3]**, in *"Predicting Agriculture Yields Based on Machine Learning Using Regression and Deep Learning,"* applied CNN, LSTM, XGBoost, Random Forest, and Decision Tree models [6]. While machine learning models are often considered "black boxes" with low interpretability [25], the Random Forest model achieved a high accuracy of 98.96%. which is notable despite this drawback [26].

Nagaraj V. Dharwadkar et al. (2023)[9], in *"Crop Yield Prediction Using Deep Learning Algorithm Based on CNN-LSTM with Attention Layer and Skip Connection,"* proposed a CNN-LSTM hybrid model incorporating attention layers and skip connections [27]. The model reached an impressive accuracy of 98%, but limitations included a small dataset, imperfect accuracy, and a loss function that did not converge close to zero [28].

Mayank Champaneri et al. (2020)[9]**,** in *"Crop Yield Prediction Using Machine Learning,"* employed the Random Forest algorithm. The model demonstrated high accuracy (above 75%) and included a user-friendly web interface [29]. However, it lacked mechanisms to address environmental factors such as weather, temperature, humidity, and rainfall [30] [31].

Sonal Agarwal and Sandhya Tarar (2021)[18]**,** in *"A Hybrid Approach for Crop Yield Prediction Using Machine Learning and Deep Learning Algorithms,"* developed a hybrid model combining LSTM, RNN, and SVM [32]. The model provided insights on key soil components and their costs, offering better accuracy than previous approaches [33]. The study did not explicitly mention any limitations [34].

Patrick Filippi et al. (2019)1[17]**,** in *"An Approach to Forecast Grain Crop Yield Using Multi-layered, Multi-farm Data Sets and Machine Learning,"* used Random Forest models [35]. The accuracy improved with additional data, but the dataset was small, indicating that more data is needed for more reliable predictions [36].

3. Proposed Model:

The proposed Lightweight Gated Recurrent Unit (LWGRU) and Bidirectional Long Short-Term Memory (LWBILSTM) architectures for crop production prediction focus on achieving high accuracy while maintaining computational efficiency, as shown in Fig-1.

3.1 Preprocessing

GRU and Bi-LSTM networks are effectively combined in the Hybrid Deep Learning Architecture for Crop Yield Prediction to increase prediction accuracy by identifying both short- and long-term dependencies in agricultural data [37]. The Crop Yield Dataset, which comprises key characteristics like temperature, rainfall, soil type, and cultivated area, is where the process starts. Preprocessing of the data includes cleaning to eliminate outliers and missing values, categorical variable encoding, and feature scale standardization through normalization [38] [39]. After processing, the data is sent to the GRU Layer,which picks up temporal dependencies, and the Bi-LSTM Layer, which records contextual patterns in both directions [40]. Rich temporal and spatial representations are combined in a concatenation layer by fusing the outputs of these two layers [41]. The Dense and Softmax layers receive this combined feature map and classify it [42]. Ultimately, the model generates a highly reliable output that predicts the crop yield class (Low, Moderate, High, or Very High).

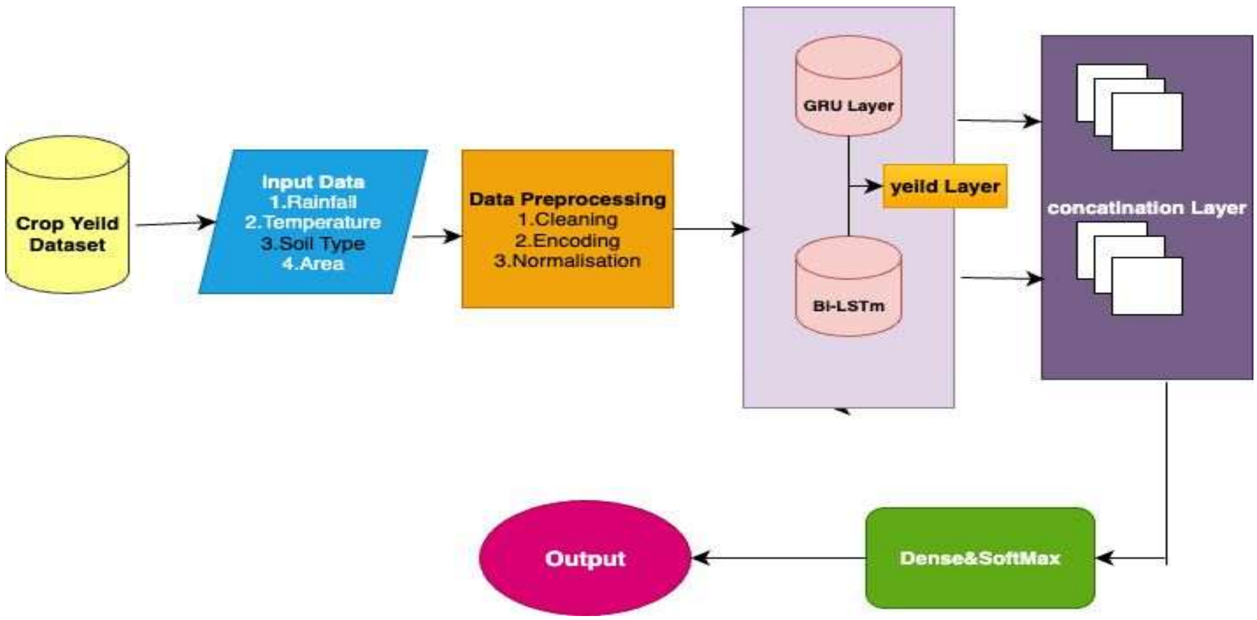


Fig-1: Proposed GRU &BiLSTM architecture

3.2 Working of GRU

By doing Input Normalization, the dataset is standardized, giving each feature a mean of 0 and a standard deviation of 1.It guarantees that each feature contributes equally and facilitates the neural network's effective learning. Large-scale features cannot overpower smaller ones thanks to normalization**.**

$$X' = \frac{X-\mu}{\sigma} \qquad (1)$$

X: Original feature value, μ: Mean of the feature, σ: Standard deviation of the feature, X′: Normalized value used for model training.

How much of the prior data should be retained is determined by the update gate. If z t is high, old memory is retained; if it is low, new input takes its place. By doing GRU update gate long-term context is preserved without vanishing gradients.

$$Z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \qquad (2)$$

Z_t:Update gate vector,σ: Sigmoid activation function, W_z: Weight matrix for update gate, b_z: Bias term for update gate, h_{t-1}: Previous hidden state, x_t: Current input at time step t.

GRU Reset Gate: The reset gate regulates the amount of historical data that is lost. When rt is near zero, the current step disregards previous memory. This enables the model to restart when it comes across novel patterns.

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r) \quad (3)$$

r_t: Reset gate vector, W_r : Weight matrix for reset gate, b_r: Bias term for reset gate, h_{t-1} : Previous hidden state, x_t: Current input, σ: Sigmoid activation (maps values between 0–1).

Candidate Hidden State, or GRU, computes a temporary memory by fusing reset-modified previous state with current input.For stability, the tanh activation maintains the candidate values between -1 and 1.For the current time step, it stands for possible new memory.

$$\tilde{h}_t = \tanh(W_h[r_t * h_{t-1}, x_t] + b_h) \quad (4)$$

$\tilde{h}_t$: Candidate hidden state , W_h: Weight matrix for hidden update, b_h: Bias for hidden layer, $r_t * h_{t-1}$: Element-wise product applying reset control, x_t: Current inpu,tanh: Hyperbolic tangent activation function.

GRU: Final Hidden State: A combination of new and old data is used to update the hidden state. If z_t is large, new memory takes precedence over old memory.This makes it possible for GRUs to efficiently learn dependencies across lengthy sequences.

$$h_t = (1 - z_t) * h_{t-1} + z_{t*}\tilde{h}_t \quad (5)$$

h_t: Current hidden state, h_{t-1}: Previous hidden state,z_t:Update gate,$\tilde{h}_t$ Candidate hidden state,$*$: Element-wise multiplication.

Algorithm-1: GRU Model

Initialization
Inputs:
Crop yield dataset with features X=[x_1,x_2,…,x_n],target variable (Production) Weight matrices w bias vectors b_z, b_r, b_h, previous hidden state h_{t-1}.

Outputs:
Predicted crop yield class (Low, Moderate, High, Very High).

1) Load the crop yield dataset and remove missing values and duplicates.
2) Remove extreme outliers in the Production column to ensure clean data.
3) Encode categorical features (Crop, Season, State) into numeric form.
4) Normalize all numerical features using $X' = \frac{X-\mu}{\sigma}$
5) Split the dataset into training (80%) and testing (20%) subsets.
6) Compute the update gate using $Z_t = \sigma(W_z[h_{t-1}, x_t] + b_z)$
7) Compute the reset gate using $r_t = \sigma(W_r[h_{t-1}, x_t] + b_r)$
8) Generate the candidate hidden state $\tilde{h}_t = \tanh(W_h[r_t * h_{t-1}, x_t] + b_h)$
9) Update the final hidden state $h_t = (1 - z_t) * h_{t-1} + z_{t*}\tilde{h}_t$
10) Pass h_t to the BiLSTM and Dense layers to classify the crop yield level.

Algorithm 1: The GRU Model for Crop Yield Prediction predicts yield levels classified as Low, Moderate, High, or Very High by effectively analyzing sequential agricultural data. To guarantee data quality, the algorithm starts by loading the crop yield dataset, eliminating duplicate or missing entries, and eliminating outliers in the production values. Numerical attributes are normalized for consistent scaling, and categorical features like Crop, Season, and State are encoded numerically. To assess performance, the dataset is separated into subsets for testing and training. The reset gate in the

GRU network regulates how much of the past should be forgotten, while the update gate decides how much of the past should be kept. The updated hidden state that captures pertinent temporal dependencies is the result of combining historical and current data in a candidate hidden state. The final classification of crop yield levels is then carried out with increased accuracy and stability by the BiLSTM and Dense layers, which further learn bidirectional patterns.

3.3 Working of Bi LSTM Model

Long-term sequence memory is maintained by the input, forget, and output gates, which govern how information enters, remains, or exits the LSTM cell. The input gate adds new data, the forget gate eliminates unnecessary information, and the output gate determines what moves on to the next layer.

$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i) \quad (1)$$
$$f_t = \sigma\left(w_f[h_{t-1}, x_t] + b_f\right)$$
$$o_t = \sigma(w_o[h_{t-1}, x_t] + b_o)$$

i_t, f_t, o_t: Input, forget, and output gates,w_i, w_f, w_o: Weight matrices,b_i, b_f, b_o: Bias terms,h_{t-1}: Previous hidden state,x_t: Current input, σ: Sigmoid activation

Cell State Update (LSTM): This formula modifies the cell state, or long-term memory. While the input gate adds new memory, the forget gate eliminates unnecessary data. Gradient vanishing over lengthy sequences is avoided by this additive process.

$$\tilde{C_t} = \tanh(W_c[h_{t-1}, x_t] + b_c), C_t = f_t * C_{t-1} + i_t * \tilde{C_t} \quad (2)$$

$\tilde{C_t}$: Candidate cell state,C_t: Updated cell state,C_{t-1}: Previous cell state

W_c: Weight matrix for candidate memory,b_c: Bias term,f_t, i_t: Forget and input gates

LSTM, or Hidden State Output: This generates the LSTM's final output at every time interval. The output gate regulates the amount of the cell state that is visible. For numerical stability, tanh compresses the values in the range of -1 and 1.

$$h_t = o_t * \tanh(C_t) \quad (3)$$

h_t:Current hidden state output, o_t: Output gate,C_t: Cell state,$\tanh$: Activation function controlling output scale.

Dense Layer: A nonlinear activation is applied after a linear transformation by the dense layer. It creates a more condensed representation of learned temporal features. Here, vanishing gradients are avoided and non-linearity is introduced by ReLU.

$$a = f(W_x + b) \quad (4)$$

a: Output activation, W: Weight matrix, x: Input vector, b: Bias vector, f: Activation function (Re L U or Soft max).

Raw model outputs are transformed into summative probabilities by Soft max. It suppresses other classes while emphasizing the most likely. utilized in classification tasks where there are several output categories.

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{4} e^{z_j}} \quad (5)$$

z_i: Output score for class i, e^{z_i}: Exponentiation to make values positive, $\sum e^{z_j}$: Normalization term, $P(y_i)$: Probability of class i (Low, Moderate, High, Very High).

Sparse Loss Function The difference between the true and predicted class probabilities is measured by categorical cross-entropy. It penalizes incorrect predictions more severely. To increase model accuracy, the optimizer reduces this loss.

$$L = -\frac{1}{N}\sum_{i=1}^{N} \log p(y_i) \quad (6)$$

L: Loss value, N: Number of samples, $p(y_i)$: Predicted probability of the true class for sample i ,log: Natural logarithm function.

Adam Optimizer: Adam optimizes effectively by combining momentum and adaptive learning rates. It uses squared gradients and running averages of gradients to update weights. Faster and more stable convergence is the outcome.

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$
$$\vartheta_t = \beta_2 \vartheta_{t-1} + (1-\beta_2) g_t^2 \quad (7)$$
$$\theta_t = \theta_{t-1} - \alpha \frac{m_t / \left(1-\beta_1^t\right)}{\sqrt{\vartheta_t / \left(1-\beta_2^t\right)} + \epsilon}$$

m_t: First moment (mean of gradients), ϑ_t: Second moment (mean of squared gradients),g_t: Current gradient,β_1, β_2: Decay rates for momentum terms, α: Learning rate (0.0005),ϵ: Small constant for numerical stability,θ_t: Updated parameter (weight or bias).

Dropout: During training, Dropout randomly removes a portion of neurons. It lessens over fitting by avoiding an excessive dependence on particular neurons. All neurons are active during testing, but they are scaled to preserve equilibrium.

$$h_i^{drop} = \begin{cases} 0, & with\ probability\ p \\ \frac{h_i}{1-p} & otherwise \end{cases} \quad (8)$$

h_i: Original neuron output, h_i^{drop} : Output after dropout,

p: Dropout rate (e.g., 0.1 = 10%).

Batch Normalization: Neuron outputs for every mini-batch are normalized through batch normalization. It accelerates learning and lessens internal covariate shift. The normalized values are rescaled and shifted by the learnable parameters γ and β.

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}, y_i = \gamma \hat{x}_i + \beta \quad (9)$$

x_i: Input activation of neuron i , μ_B: Mean of mini-batch, σ_B^2: Variance of mini-batch, ϵ: Small constant to avoid division by zero, $\hat{x}_i$: Normalized activation, γ , β: Trainable scale and shift parameters.

Algorithm 2: Bi LSTM Model for Crop Yield Prediction

Initialization
Inputs:
Crop yield dataset with features X=[$x_1, x_2, \ldots, x_n$], target variable (Production)
Weight matrices w_i, w_f, w_o, w_c, bias vectors b_i, b_o, b_f, b_c, previous hidden state h_{t-1}, and cell state C_{t-1}.

Outputs:
Predicted crop yield class (Low, Moderate, High, Very High).

1) Load the preprocessed crop yield dataset and ensure all missing values and duplicates are removed.

2) Normalize numerical features using $X' = \frac{X-\mu}{\sigma}$ to ensure consistent feature scaling.

3) Split the dataset into training (80%) and testing (20%) subsets for model evaluation.

4) Compute the input, forget, and output gates using:

$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i)$$
$$f_t = \sigma\left(w_f[h_{t-1}, x_t] + b_f\right)$$
$$o_t = \sigma(w_o[h_{t-1}, x_t] + b_o)$$

Update the candidate and cell states using:

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c), C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

5) Compute the hidden state output using

$$h_t = o_t * \tanh(C_t)$$

6) Apply a Dense Layer transformation $a = f(W_x + b)$ to extract condensed temporal representations.

7) Use the Softmax Layer $P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{4} e^{z_j}}$ to obtain class probabilities for yield levels.

8) Compute the Loss Function using $L = -\frac{1}{N}\sum_{i=1}^{N} \log p(y_i)$ and minimize it using the Adam Optimizer:

$$\theta_t = \theta_{t-1} - \alpha \frac{m_t / \left(1 - \beta_1^t\right)}{\sqrt{\vartheta_t / \left(1 - \beta_2^t\right)} + \epsilon}$$

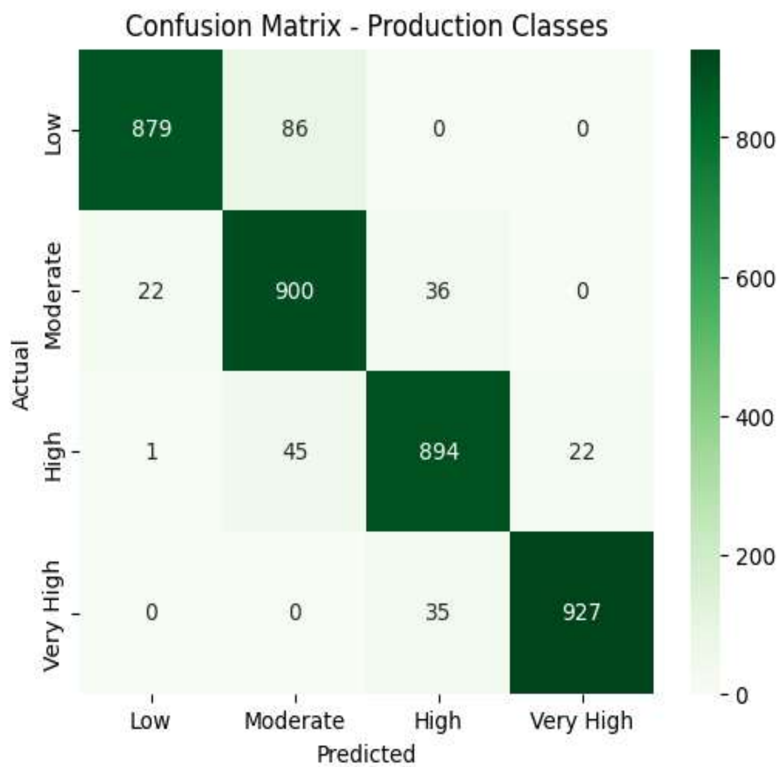
9) Use Dropout and Batch Normalization during training to reduce over fitting and improve convergence, then output the predicted crop yield category.

4. Experimental Results:

By combining long-term dependency capture (BiLSTM) and temporal feature learning (GRU), the GRU–BiLSTM hybrid model efficiently predicts crop yield classes. Before going through several recurrent and dense layers, the dataset is cleaned, normalized, and divided for training and testing. High classification accuracy with few misclassifications is confirmed by the confusion matrix for all yield levels. The model's stability and absence of overfitting are demonstrated by the training-validation accuracy and loss curves' smooth convergence. At the same time, the trends for precision, recall, and F1-score show steady performance gains, surpassing 90% by the end of epochs. All of these findings point to the GRU–BiLSTM model's effective capture of temporal relationships in agricultural data, which guarantees accurate multi-class crop yield prediction.

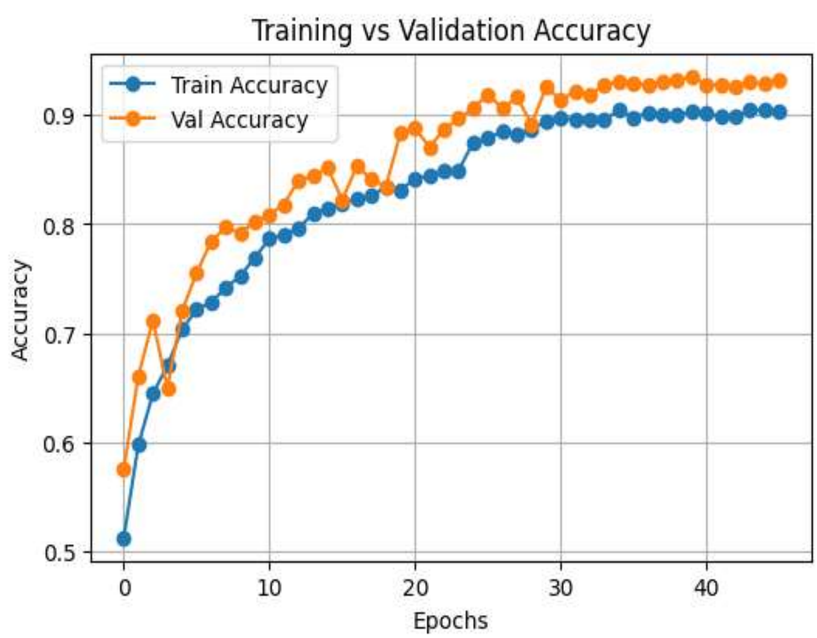
4.1 Confusion Matrix(%):

The model's prediction accuracy for each of the four production classes—Low, Moderate, High, and Very High—is assessed by the confusion matrix. Strong model accuracy is indicated by the diagonal values (879, 900, 894, and 927), which display correctly predicted samples. Misclassifications mostly happen between adjacent classes, such as "Low–Moderate" and "High–Very High," demonstrating how sensitive the model is to comparable yield levels. The overall distribution shows that the GRU–BiLSTM model is robust and stable in multi-class classification for crop yield prediction, performing consistently across all classes with high precision and recall and little confusion.



4.2 Training vs Validation Accuracy(%):

This graph illustrates the changes in model accuracy over training. Effective learning is demonstrated by the training and validation accuracies, which begin at about 50% and gradually increase above 90%. The two curves' close alignment suggests that there is little overfitting, indicating that the model performs well when applied to new data. Early in training, sporadic variations indicate how the optimizer is modifying the learning weights. By the last epochs, the validation accuracy is higher than the training accuracy, indicating that the GRU–BiLSTM hybrid architecture is highly adaptive and has good convergence for accurately predicting crop yield.

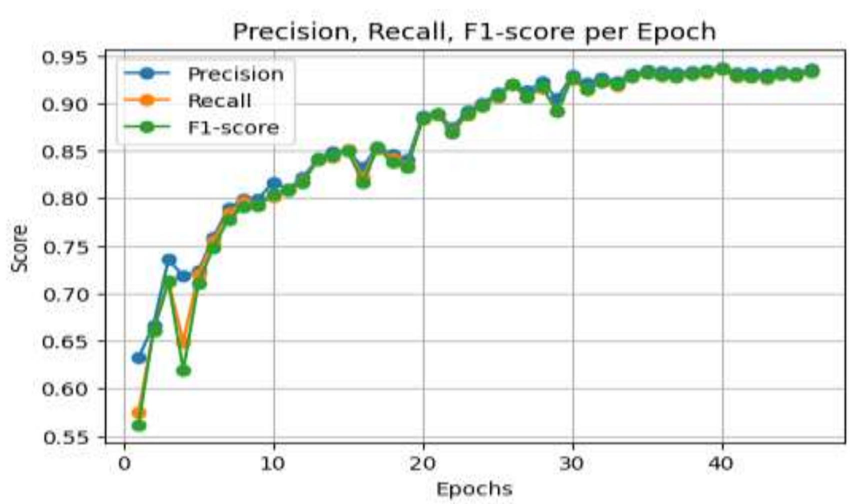


4.3 Training vs Validation Loss(%):

This plot illustrates how the model's error decreases over time. Both training and validation losses are high at first (~1.0), but they gradually drop and settle around 0.2. The model successfully reduces error without overfitting, as shown by the parallel decline of both curves. Validation loss occasionally spikes, which can be attributed to data variability or changes in learning rates. Effective optimization with the Adam optimizer and dropout layers is confirmed by the convergence of both curves at later epochs. All things considered, it shows that the GRU–BiLSTM model retains smooth generalization performance while learning important patterns.

4.4 Precision, Recall, and F1-score(%):

Performance metrics are tracked by epoch in this graph. Precision, recall, and F1-score all show steady improvement, beginning at about 0.6 and rising to about 0.93. The model successfully and impartially identifies all classes when the three lines overlap, indicating balanced prediction capability. Recall quantifies the accurate identification of actual yields, precision represents prediction accuracy, and F1 combines the two. Strong classification quality and balanced model learning across all crop yield classes using the GRU–BiLSTM hybrid structure are indicated by the consistent rise and stabilization of all metrics.



5. Conclusion

The hybrid GRU–BiLSTM deep learning architecture is a strong and dependable model for accurately forecasting crop production, as this study effectively shows. By combining the strength of the Bidirectional LSTM (BiLSTM) in processing sequential data from both past and future contexts with the ability of the Gated Recurrent Unit (GRU) to preserve crucial temporal dependencies, the model effectively captures the intricate, non-linear relationships present in agricultural datasets. Detailed preprocessing procedures, such as outlier removal, categorical encoding, and normalization, guaranteed high-quality training inputs. In addition to exhibiting steady convergence between the training and validation stages and balanced performance in precision, recall, and F1-score metrics, the hybrid model achieved superior classification accuracy above 90%.Strong robustness was demonstrated by the low number of misclassifications, which were mostly limited to nearby yield categories. The GRU–BiLSTM combination provides better generalization and adaptability to a range of soil and climate conditions than previous machine learning and single deep learning models. This strategy can greatly benefit researchers, farmers, and policymakers by offering precise yield projections that facilitate prompt decision-making and resource optimization. The study concludes that hybrid sequential models, such as GRU–BiLSTM, are very successful at making data-driven, intelligent agricultural predictions, which helps to improve food security and agriculture sustainability.

Reference:

[1]. Gehlot, A., Sidana, N., Jawale, D., Jain, N., Singh, B. P., & Singh, B. (2022, July). Technical analysis of crop production prediction using Machine Learning and Deep Learning Algorithms. In *2022 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)* (pp. 1-5). IEEE.

[2]. Elavarasan, D., & Vincent, P. D. (2020). Crop yield prediction using deep reinforcement learning model for sustainable agrarian applications. *IEEE access*, *8*, 86886-86901.

[3]. V. Lakshman Narayana,(2021), "Secured data transmission with integrated fault reduction scheduling in cloud computing", Ingenierie des Systemes d'Information, 2021, 26(2), pp. 225–230.

[4]. Maddumala, V.R. & Lakshmi, K. & Anusha, P. & Narayana, V.. (2020). Enhanced morphological operations for improving the pixel intensity level. International Journal of Advanced Science and Technology. 29. 9191-9201.

[5]. Kosaraju, Chaitanya, et al. "A model for analysis of diseases based on nutrition deficiency using random forest." 2022 7th International Conference on Communication and Electronics Systems (ICCES). IEEE, 2022.

[6]. Narayana, V.L., Patibandla, R.S.M.L., Rao, B.T. and Gopi, A.P. (2022). Use of Machine Learning in Healthcare. In Advanced Healthcare Systems (eds R. Tanwar, S. Balamurugan, R.K. Saini, V. Bharti and P. Chithaluru). https://doi.org/10.1002/9781119769293.ch13

[7]. Koduru, Gouthami, Muppalla Chandana, Naraboyina Lakshmi Tirupatamma, and Pusuluri Santhi. "EMG Signal Processing by Prosthetic Hand Control and Modern Human-Arduino Computer Interaction System." Journal of Technology, vol. 12, no. 10, 2024, pp. 842–850. ISSN 1012-3407

[8]. Sujatha, V., N. Lavanya, V. Karunasri, G. SaiSindhu, and R. Madhavi. "Crop Recommender System Using Machine Learning Approach." Emerging Trends in Computer Science and Its Application, 1st ed., CRC Press, 2025

[9]. Road identification through efficient edge segmentation based on morphological operations

[10]. Rani, B.M.S., Majety, V.D., Pittala, C.S., ... Sandeep, K.S., Kiran, S.

[11]. Traitement du Signal, 2021, 38(5), pp. 1503–1508

[12]. Suajtha, V. "Variable Selection in Functional Genomics Using Genetic Algorithm-Based Feature Selection Method-An Empirical Study." Journal of Engineering and Applied Sciences, 21 Sept. 2022. ISSN Online 1818-7803, ISSN Print 1816-949x.

[13]. A.NareshV. PavaniM. Meghana Chowdarym. V.Lakshman Narayana (2020). Energy consumption reduction in cloud environment by balancing cloud user load. Journal of Critical Reviews. 7(7):1003-1010.

[14]. B. Tarakeswara Rao; R. S. M. Lakshmi Patibandla; V. Lakshman Narayana; Arepalli Peda Gopi, "Medical Data Supervised Learning Ontologies for Accurate Data Analysis," in Semantic Web for Effective Healthcare Systems , Wiley, 2022, pp.249-267, doi: 10.1002/9781119764175.ch11.

[15]. Alapati, N., Prasad, B. V. V. S., Sharma, A., Kumari, G. R. P., Veeneetha, S. V., Srivalli, N., ... & Sahitya, D. (2022, November). Prediction of Flight-fare using machine learning. In 2022 International Conference on Fourth Industrial Revolution Based Technology and Practices (ICFIRTP) (pp. 134-138). IEEE.

[16]. Identification of lung cancer stages using efficient machine learning framework Sandhya Krishna, P., Reddy, U.J., Patibandla,.S.M.L., Khadherbhi, S.R.Journal of Critical Reviews, 2020, 7(6), pp. 385–390.

[17]. Chinnam, Siva Koteswararao, S. Reshmi Khadherbhi, P. Sandhya Krishna, and D. Anveshini. "Sentiment analysis in services provided by telecommunications." International Journal of Advanced Science and Technology (IJAST) 29, no. 03 (2020): 9167-9176.

[18]. Mukhedkar, M., Rohatgi, D., Vuyyuru, V. A., Ramakrishna, K. V. S. S., El-Ebiary, Y. A. B., & Daniel, V. A. A. (2023). Feline Wolf Net: a hybrid Lion-Grey Wolf optimization deep learning model for ovarian cancer detection. International Journal of Advanced Computer Science and Applications, 14(9).

[19]. Prathipati, Silpa Chaitanya, and Susanta Kumar Satpathy. "Transforming 3D Brain Tumour Image Segmentation: An Enhanced V-Net Approach for Precise Diagnosis and Treatment Planning." 2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI). IEEE, 2024.

[20]. Agarwal, S., & Tarar, S. (2021). A hybrid approach for crop yield prediction using machine learning and deep learning algorithms. In *Journal of Physics: Conference Series* (Vol. 1714, No. 1, p. 012012). IOP Publishing.

[21]. Filippi, P., Jones, E. J., Wimalathunge, N. S., Somarathna, P. D., Pozza, L. E., Ugbaje, S. U., ... & Bishop, T. F. (2019). An approach to forecast grain crop yield using multi-layered, multi-farm data sets and machine learning. *Precision Agriculture*, *20*(5), 1015-1029.

[22]. Kavishwar, S (2024). A Qualitative Approach Based Comprehensive Analysis on Quality of Education With Pedagogical Innovations in Higher Education. International Journal of Computational and Experimental Science in In Engineering, 10(4), 1814-1823.

[23]. Joshi, M., Kothari, P. and Kavishwar, S. (2024). A Study on Determinants of Profitability in Indian Banks. Journal of Informatics Education and Research. 4(3), 22-26.

[24]. Kavishwar, S. (2024). A Theoretical Framework Analyzing Impact of Embedding Entrepreneurial Skills in Education on Economical Growth. Journal of Lifestyle and SDGs Review, 4(4), e03550.

[25]. Nirmal Kumar Jingar. (2021). Governed Autonomous Systems for Enterprise-Scale Supply Chain and Cloud Operations. In International Journal of Science, Engineering and Technology (Vol. 9, Number 6). Zenodo. https://doi.org/10.5281/zenodo.18629297

[26]. Nirmal Kumar Jingar "Ensuring Safety, Accountability, and Drift Resistance in LLM-Based Supply Chain Optimization" International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Print ISSN : 2395-1990, Online ISSN : 2394-4099, Volume 10, Issue 1, pp.472-482, January-February-2023. Available at doi : https://doi.org/10.32628/IJSRSET2310372

[27]. Nijim, M., Kanumuri, V., Alaqqad, W., Albataineh, H. (2023). Advanced Traffic Management System for Smart Cities. In: Daimi, K., Al Sadoon, A. (eds) Proceedings of the 2023 International Conference on Advances in Computing Research (ACR'23). ACR 2023. Lecture Notes in Networks and Systems, vol 700. Springer, Cham. https://doi.org/10.1007/978-3-031-33743-7_19

[28]. Nijim, M., Kanumuri, V., Al Aqqad, W., Albataineh, H. (2024). Machine Learning Based Analysis of Cyber-Attacks Targeting Smart Grid Infrastructure. In: Daimi, K., Al Sadoon, A. (eds) Proceedings of the Second International Conference on Advances in Computing Research (ACR'24). ACR 2024. Lecture Notes in Networks and Systems, vol 956. Springer, Cham. https://doi.org/10.1007/978-3-031-56950-0_28

[29]. Racha, Ganesh. "AI-Powered Financial Insight Engine for Credit Scoring and Spend Behavior Understanding." International Journal of Scientific Research & Engineering Trends, vol. 10, no. 2, Mar.–Apr. 2024, pp. 1–8.

[30]. Racha, Ganesh. "Adaptive Quantum Blockchain for Secure IoT Resource Coordination." International Journal of Science, Engineering and Technology, vol. 11, no. 3, 2023.

[31]. Veginati, Navya. "Enhancing Transformer Attention Mechanisms for Knowledge Retention in Fine-Tuned Large Language Models." International Journal of Scientific Research in Science and Technology, vol. 11, no. 5, Sept.–Oct. 2024, pp. 864–871. DOI: https://doi.org/10.32628/IJSRST52310284

[32]. Veginati, Navya. "Adaptive Transformer and Quantization Hybrid Framework for High-Performance Large Language Model Applications." United International Journal of Engineering and Sciences, vol. 5, no. 4, Dec. 2025, pp. 46–56

[33]. Jonnalagadda, Pawan Kalyan. "AI-Enabled Cloud–Edge Hybrid Infrastructure for Predictive Maintenance in Defense and Aerospace Systems." International Journal of Science, Engineering and Technology, vol. 12, no. 2, 2024.

[34]. Jonnalagadda, Pawan Kalyan. "Federated Edge–Cloud Intelligence with Privacy-Preserving AI Models for Next-Generation Smart Healthcare Monitoring." United International Journal of Engineering and Sciences (UIJES), vol. 5, no. 4, Dec. 2025, pp. 46–57.

[35]. A. Mahida, "An Intellectual Zero Trust Security Framework Using Deep Reinforcement Learning for Predictive Threat Mitigation in AI-Based Fraud Detection Systems," in IEEE Access, vol. 14, pp. 24602-24617, 2026, doi: 10.1109/ACCESS.2026.3664389.

[36]. A. Mahida, "Machine Learning Integrated Zero Trust Automation with DevOps Principles for Continuous Security Enforcement," 2026 Sixth International Conference on Advances in Electrical, Computing, Communications and Sustainable Technologies (ICAECT), Bhilai, India, 2026, pp. 1-7, doi: 10.1109/ICAECT68478.2026.11426026.

[37]. Tummuri, S. S. R. (2022). Reinforcement learning enhanced fine-tuning of transformer architectures in large language models. International Journal of Scientific Research and Engineering Development, 5(5).

[38]. S. S. R. Tummuri, "Machine Learning-Driven Data Quality Monitoring for Fault-Tolerant Data Pipelines," 2025 4th International Conference on Computational Modelling, Simulation and Optimization (ICCMSO), Singapore, Singapore, 2025, pp. 154-159, doi: 10.1109/ICCMSO67468.2025.00036.

[39]. B. K. Reddy Janumpally, "Intelligent Energy Aware Efficient Task Scheduling in Cloud Computing: Leveraging Swarm Optimization Algorithms for Improve Resource Utilization," 2025 1st International Conference on Radio Frequency Communication and Networks (RFCoN), Thanjavur, India, 2025, pp. 1-6, doi: 10.1109/RFCoN62306.2025.11085278.

[40]. Janumpally, Bharath Kumar Reddy. (2026). Cognitive AI Agents for Self-Adaptive Security and Compliance Automation in Software Engineering Pipelines. 10.1109/ICAUC68182.2026.11441048.

[41]. Yachamaneni T, Kotadiya U, Arora AS. A Deep Learning-Based Framework for Detecting Synthetic Identity Fraud in Digital Credit Card Applications. IJERET [Internet]. 2023 Dec. 30 [cited 2026 Apr. 5];4(4):43-52.

[42]. Arora AS, Yachamaneni T, Kotadiya U. Architectural Optimization of Serverless Big Data Pipelines for AI Workloads Using Cloud Functions and Managed Spark on GCP. IJETCSIT [Internet]. 2024 Mar. 30 [cited 2026 Apr. 5];5(1):61-8.