Journal of Basics and Applied Sciences Research (JOBASR)
ISSN (print): 3026-9091, ISSN (online): 1597-9962
Volume 4(2) March 2026
DOI: https://dx.doi.org/10.4314/jobasr.v4i2.29

Optimized Deep Learning and KNN Models with PCA Feature Selection for Forecasting Cowpea Yeild in Nigeria

Terfa Benjamin Yecho[1*] & Eli Adama Jiya[2]
[1,2]Department of Computer Science, Federal University Dutsin-Ma, Katsina State, Nigeria.
[*]Corresponding Author Email: ybenjamin@fudutsinma.edu.ng

ABSTRACT
Reliable prediction of crop yield plays a critical role in improving agricultural decision-making and promoting sustainable farming practices. Conventional approaches are often limited in their ability to model the nonlinear relationships that exist among plant growth dynamics. In this study, three machine learning techniques namely; Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks and K-Nearest Neighbors (KNN) were designed and assessed for their effectiveness in forecasting cowpea (Vigna unguiculata) yield based on data acquired from IoT-enabled sensors, the dataset was obtained from a controlled cultivation experiment, with yield quantified by the number of pods produced per plant. Principal Component Analysis reduced dimensionality while preserving over 95% of the variance. Hyperparameter optimization was performed using GridSearchCV for KNN and Keras Tuner RandomSearch for CNN and LSTM. The optimized achieved the highest predictive accuracy with an R^2 of 0.8754, MAE of 0.0492, and RMSE of 0.0639, outperforming the optimized LSTM (96 units, additional dense layer of 64 units, RMSprop optimizer; R^2 = 0.8334) and optimized KNN (n_neighbors = 3; R^2 = 0.7566). However, both CNN and LSTM showed systematic under-prediction bias at higher yield values, with LSTM exhibiting the largest negative residuals. The findings demonstrate that deep learning approaches, particularly CNN, can effectively model crop yield with relatively small datasets when combined with appropriate feature selection and hyperparameter optimization. The integration of statistical feature selection with agronomic domain knowledge enhances model robustness and biological interpretability. These results support improved decision-making in precision agriculture, enabling more accurate yield forecasting for sustainable cowpea production.

Keywords:
Cowpea Yield Prediction,
Machine Learning,
Deep Learning,
Autoencoder,
Feature selection

INTRODUCTION

Cowpea is a legume crop grown across semi-arid and tropical agroecological zones due to its high nutritional value and its ability to enhance soil fertility through biological nitrogen fixation. Accurate crop yield prediction plays a crucial role in optimizing agricultural management practices, improving food security, and supporting sustainable farming systems. However, traditional statistical and empirical methods often struggle to capture the complex interactions among environmental conditions, soil properties, and plant growth variables, which can lead to inconsistent yield estimation (Parashar et al., 2024).

Recent progress in Machine Learning and Deep Learning has substantially enhanced the capacity to represent and analyse complex agricultural systems. Convolutional Neural Networks have been extensively applied to capture spatial features from agricultural datasets and remote sensing imagery, thereby facilitating efficient monitoring of crop growth dynamics and developmental patterns (Kalmani et al., 2025). In parallel, Long Short-Term Memory Networks are well-suited for modelling temporal relationships in sequential agricultural data, including variations in climatic conditions and different stages of crop development (Sikandar et al., 2023). In addition to deep learning approaches, conventional machine learning methods, including the K-Nearest Neighbours (KNN) algorithm, continue to be effective for capturing complex nonlinear relationships between environmental factors and crop yield outcomes (Badshah et al., 2024).

How to cite this article: Terfa B. Y. & Eli A. J. (2026). Optimized Deep Learning and KNN Models with PCA Feature Selection for Forecasting Cowpea Yeild in Nigeria. *Journal of Basics and Applied Sciences Research*, *4*(2), 297-306. https://dx.doi.org/10.4314/jobasr.v4i2.29

Despite their effectiveness, ML and DL models often face challenges related to limited training data in agricultural experiments, which may reduce model generalization and prediction reliability. Also, despite the growing application of both machine learning and deep learning techniques in crop yield prediction, there remains a notable gap in the issue of small agricultural datasets has not been sufficiently addressed in many studies, which can lead to overfitting and reduced predictive performance. While data augmentation techniques such as autoencoders have shown potential in generating synthetic data and improving model robustness, their integration with hybrid ML and DL models for cowpea yield prediction remains underexplored. To overcome these limitations, this study uses CNN, LSTM, and KNN models alongside an autoencoder-driven data augmentation strategy. The autoencoder is employed to synthesize additional data instances that retain the underlying statistical properties of the original dataset, thereby enhancing model generalization and mitigating the risk of overfitting (Lan et al., 2025; Badar et al., 2025). Through the fusion of conventional machine learning techniques with deep learning architectures and augmentation methods, the proposed approach seeks to improve predictive performance and deliver a more robust solution for precision agriculture applications. Ultimately, this framework is intended to support informed decision-making in crop management and promote the sustainability of cowpea production systems.

MATERIALS AND METHODS

Dataset

The dataset was collected from a controlled cowpea cultivation experiment which was carried out in Federal College of Agricultural Produce Technology, Kano state, Nigeria and monitored using IoT-based sensing system. Measurements of plant morphological characteristics and soil environmental conditions were systematically collected across the growth period during 2024 farming season. These included plant height, leaf count, flower number, fresh and dry biomass, ambient temperature, soil nutrient concentrations (N, P, K), and soil moisture content. Due to small nature of the dataset, autoencoder was used to argument the dataset to 120 total instances with 11 recorded variables, producing 1,320 total data values.

Feature Selection

Correlation analysis was performed to evaluate the interrelationships among plant growth attributes, soil properties, and environmental factors, with the aim of identifying the variables most strongly linked to cowpea yield. This method measures the strength and direction of relationships among variables and helps identify important growth indicators (Jahan et al., 2023, Magray et al., 2021). The results showed that number of flowers, plant height, fresh weight, and number of leaves had strong positive correlations with yield, while other variables exhibited weaker relationships.

Principal Component Analysis (PCA) was employed as a dimensionality reduction technique to eliminate redundancy by transforming correlated variables into a smaller set of uncorrelated components. Following feature normalization with StandardScaler, Principal Component Analysis (PCA) was applied to reduce the dimensionality of the dataset by generating a set of orthogonal principal components that retain the majority of the original data variance (Maleki-Meighani et al., 2025; Huang et al., 2022). The analysis showed that seven principal components explained more than 95% of the total variance, effectively reducing the original ten features while maintaining key information for yield prediction.

While soil nitrogen (N), phosphorus (P), potassium (K), and soil moisture exhibited relatively low statistical associations with yield, they were retained in the modeling process due to their well-established agronomic importance in crop productivity. Previous studies demonstrate that adequate NPK availability significantly enhances cowpea growth, biomass accumulation, and grain yield, while soil moisture plays a critical role in nutrient transport and plant physiological processes (Fitratunnisah et al., 2025; Soliman, 2024; Al Viandari et al., 2022). Therefore, combining statistical feature selection with agronomic knowledge ensures that the predictive models capture both data-driven patterns and biologically meaningful factors, improving model robustness and generalizability. figure 1 shows the correlation heatmap of the variables.

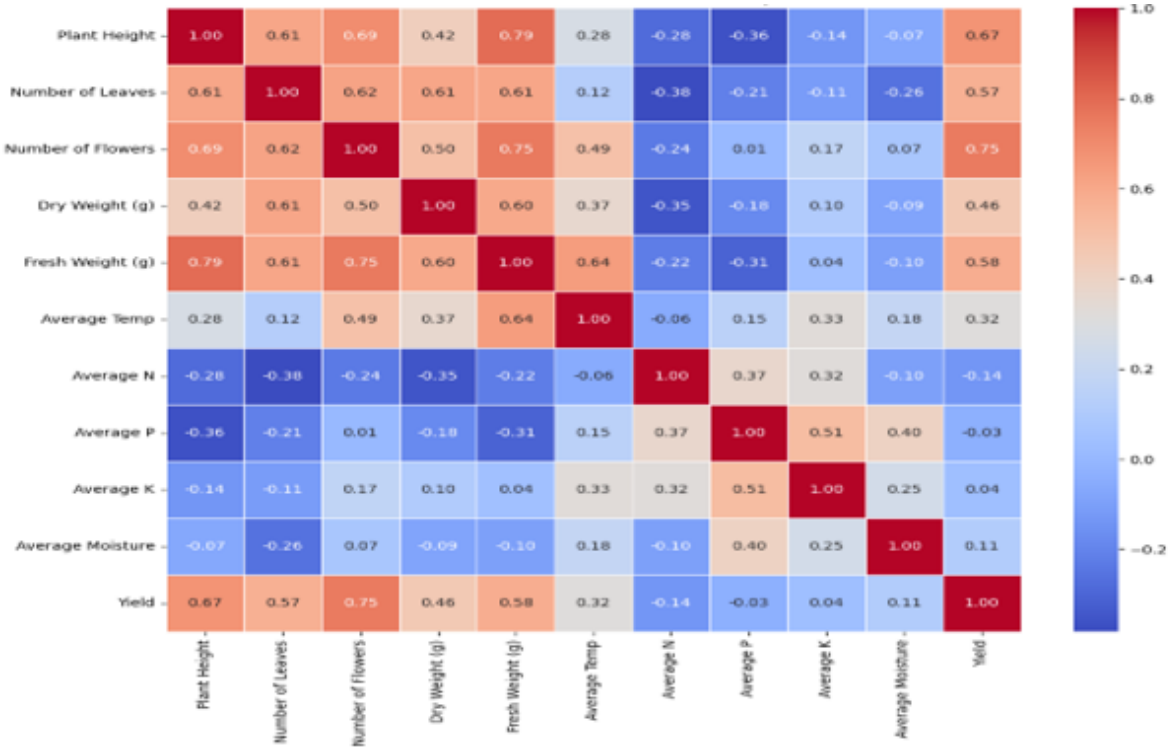

Figure 1: Feature correlation heatmap

Model Development

In this research, three distinct predictive models were utilized to forecast cowpea yield: Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and K-Nearest Neighbours (KNN). The selection of these models enabled a comparative assessment between advanced deep learning techniques and a conventional machine learning algorithm. Implementation of all models was carried out in Python within the Google Colab platform.

CNN Architecture

Convolutional Neural Network (CNN) was used to develop the cowpea yield prediction model. CNNs are deep learning architectures capable of automatically learning hierarchical feature representations from input data through convolution operations. Although originally developed for image recognition tasks, CNNs have increasingly been applied to structured agricultural datasets and time-series data for crop yield prediction and farm management applications (Jabed & Murad, 2024; Oikonomidis et al., 2022).

Convolutional Neural Networks (CNNs) utilize trainable filters, or kernels, which systematically traverse the input data to identify localized patterns and capture the relationships between variables. These convolution operations enable automatic feature extraction without manual feature engineering, allowing the model to capture complex nonlinear relationships within agricultural datasets (Khaki & Wang, 2019). In agricultural research, CNNs have been used to analyze environmental, soil, and climatic variables to improve crop yield prediction accuracy (Wang et al., 2023; Engen et al., 2021).

The convolution process is mathematically represented as:

$$S(i) = (X * K)(\mathrm{i}) \quad (1)$$

where: X denotes the input data; K is the convolution kernel and S(i) corresponds to the resulting feature map. To incorporate nonlinearity into the model, the output of the convolution is passed through a Rectified Linear Unit (ReLU) activation function, defined as

$$f(x) = max(0, x) \quad (2)$$

x represents the input value to the function. This operation enhances the efficiency of model training and contributes to alleviating the vanishing gradient issue. Subsequently, the feature maps are processed through a max-pooling layer, which reduces their spatial dimensions while preserving the most salient features. This step lowers computational demands and aids in preventing overfitting. The resulting pooled feature maps are then flattened into a one-dimensional vector and forwarded to a fully connected (dense) layer, where complex, high-level feature interactions are captured. Within the dense layer, the output is calculated as:

$$y = f(Wx + b) \quad (3)$$

where *W* denotes the weight matrix, *x* is the input vector, *b* represents the bias term *and f* represents the activation function. Figure 2 illustrates the Convolutional Neural Network (CNN) architecture developed for crop yield prediction using both agronomic and environmental variables.

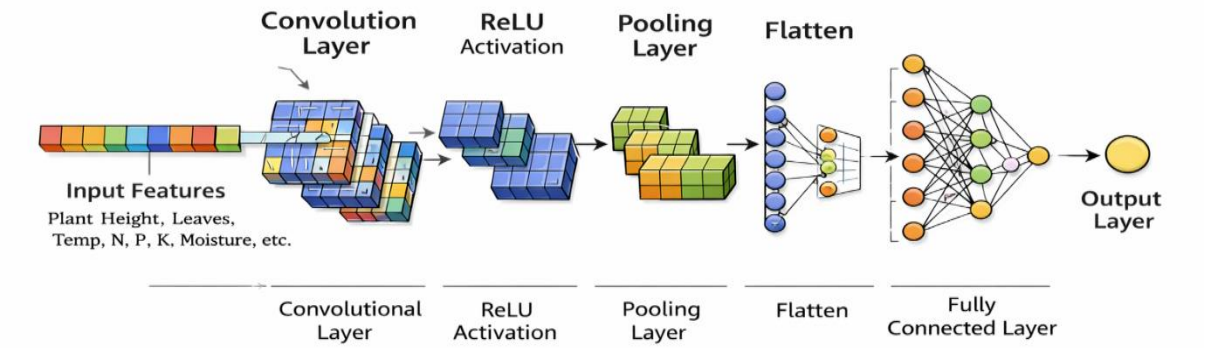

Figure 2: Architectural framework of a convolutional neural network

The model takes as input key features including plant height, number of leaves, ambient temperature, nitrogen (N), phosphorus (P), potassium (K), and soil moisture. These inputs are first processed through a convolutional layer, which is responsible for learning local feature patterns within the data. A Rectified Linear Unit (ReLU) activation function is then applied to introduce non-linearity and enhance the model's ability to capture complex relationships. Subsequently, a pooling layer is used to reduce the dimensionality of the feature maps while preserving the most relevant information. The resulting feature representations are flattened and passed into a fully connected (dense) layer, where higher-level interactions among variables are learned. Finally, the output layer generates the predicted crop yield value.

LSTM Architecture

Long Short-Term Memory (LSTM) network was employed to model sequential patterns and temporal dependencies present in the agricultural dataset. As a specialized form of recurrent neural network (RNN), LSTM is designed to overcome the vanishing gradient issue that often limits the performance of conventional RNN architectures. In this research, a Long Short-Term Memory (LSTM) architecture was employed to analyze and learn temporal patterns within the dataset. The LSTM model represents an advanced variant of Recurrent Neural Networks (RNNs), specifically developed to effectively retain and utilize long-range dependencies in sequential data. Unlike conventional RNNs, LSTM networks are structured to mitigate the vanishing gradient issue, thereby enabling more stable and efficient learning over extended sequences (Ghojogh & Ghodsi, 2023; Zhao, 2025).

The Long Short-Term Memory (LSTM) framework is characterized by the presence of a memory cell alongside three fundamental gating components: the input gate, forget gate, and output gate. These gates function collaboratively to manage the transmission and transformation of information within the network. Specifically, they determine which information should be incorporated into the memory, which should be removed, and what portion should be propagated forward. This controlled information flow allows the model to maintain important temporal dependencies while effectively disregarding less relevant signals (Okut, 2021; Sushanth, 2025).

The evolution of the cell state in an LSTM network can be expressed as

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \quad (4)$$

In this formulation, the forget gate f_t regulates the extent to which previously stored information is preserved, whereas the input gate i_t governs the incorporation of newly computed information represented by the candidate cell $\tilde{C}_t$. Through this gated mechanism, the model dynamically balances retention of past knowledge with the integration of current inputs. Such a structure enhances the network's capacity to capture long-term dependencies in sequential data, thereby making it particularly effective for time-series prediction problems, including applications like crop yield forecasting (Waqas & Humphries, 2024; Krichen & Mihoub, 2025).

Figure 3 presents the structural configuration of the Long Short-Term Memory (LSTM) network implemented for predicting cowpea yield. The model is designed to handle sequential data, where input variables such as plant height, leaf count, ambient temperature, soil nutrient levels, and moisture content are observed across multiple time steps.

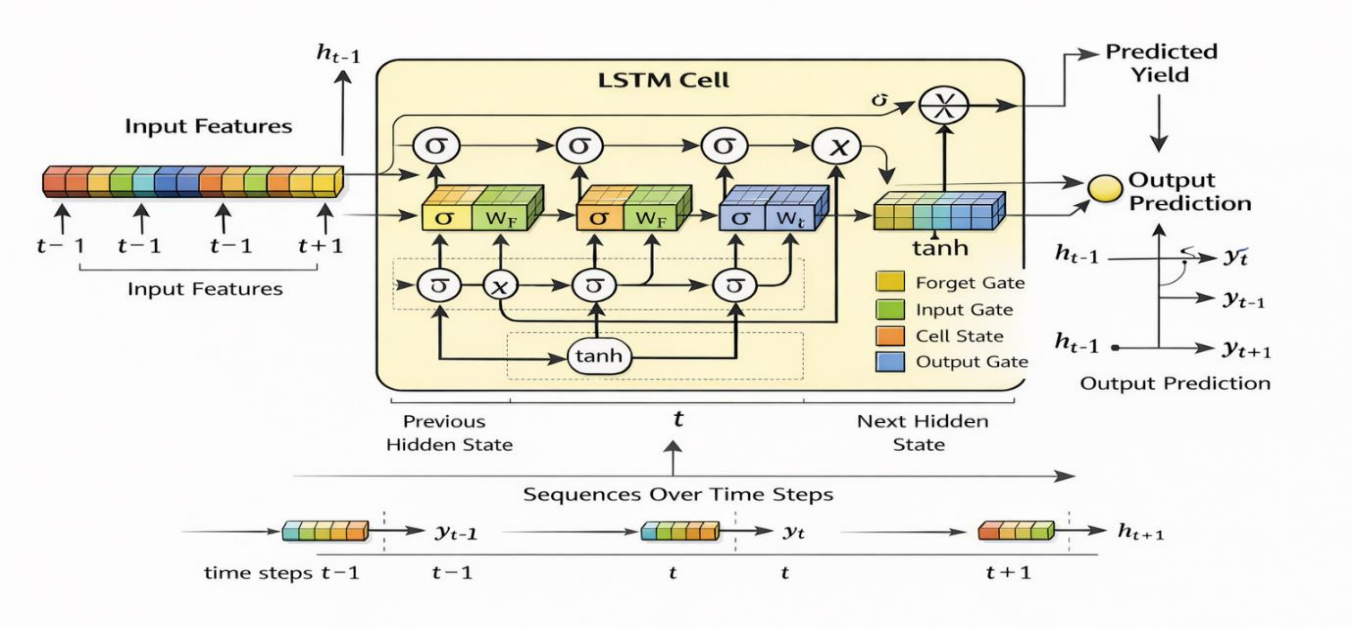

Figure 3: Architecture of the Long Short-Term Memory (LSTM) network

By leveraging the gating structure of LSTM alongside the tanh activation function, the LSTM model effectively captures temporal patterns embedded in the agricultural dataset. The final hidden state is then utilized to produce the predicted cowpea yield, allowing the model to learn intricate relationships between environmental factors, plant development indicators, and yield performance over time.

KNN Algorithm

KNN is a non-parametric and instance-based learning method that generates predictions by evaluating the similarity between a new data point and existing observations within the training dataset's feature space. The prediction is derived based on the closest neighbouring samples, typically determined using a distance metric (Mansouri et al., 2024; Sharma & Sharma, 2013). Unlike parametric models, KNN does not assume a predefined relationship between input variables and the target variable. Instead, it relies on distance-based similarity measures, such as Euclidean, Manhattan, or Minkowski distances, to identify the *k* most similar observations (Saadatfar et al., 2020; Dhivya et al., 2015). The similarity between samples is commonly computed using the Euclidean distance:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (5)$$

where *x* denotes the query instance, *y* refers to an individual sample from the training dataset, and *n* represents the total number of features considered in the analysis. Following the computation of distances between the query instance and each training sample, the algorithm proceeds by identifying and selecting the *k* nearest neighbours defined as those with the minimum distance values. For regression tasks such as crop yield prediction, the predicted output is obtained by averaging the target values of the *k* nearest neighbours:

$$\hat{y} = \frac{1}{k}\sum_{i+1}^{k} y_i \quad (6)$$

The value of *k* plays a critical role in model performance: small values may lead to overfitting, while larger values may cause underfitting (Mansouri et al., 2024; Wen, 2025). Therefore, the optimal *k* is typically determined through validation techniques such as cross-validation. Due to its simplicity, interpretability, and ability to capture nonlinear relationships, KNN is widely used as a benchmark algorithm for evaluating more advanced predictive models (Wen, 2025; Niu et al., 2024). Figure 4 presents the operational mechanism of the K-Nearest Neighbours (KNN) algorithm as applied to yield prediction. In this approach, the model evaluates a new input sample (query point) by comparing it with existing observations mapped within a multidimensional feature space. These observations represent key agricultural attributes, including plant height, leaf count, ambient temperature, soil nutrient composition, and moisture content.

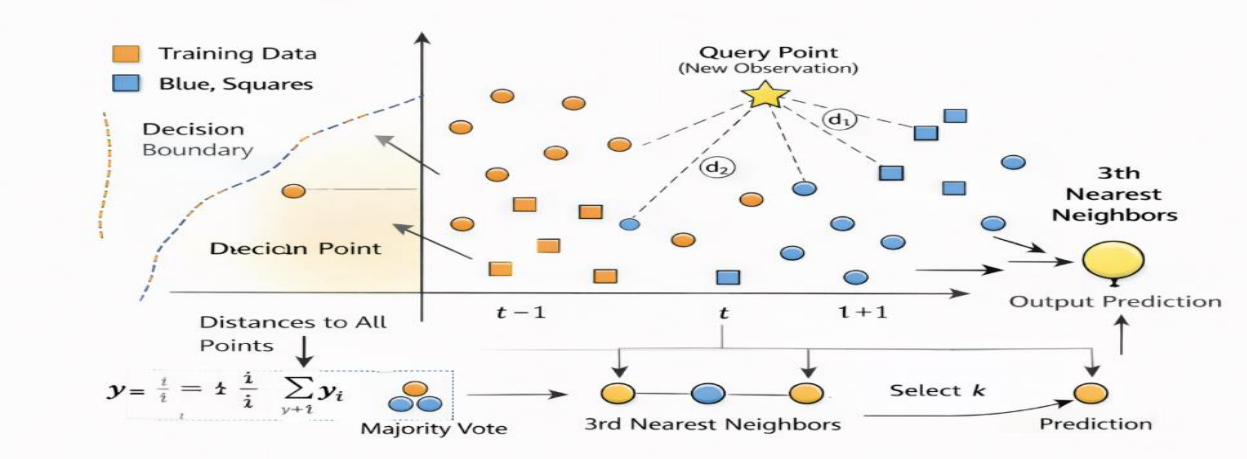

Figure 4: Architecture and operational workflow of the K-Nearest Neighbors (KNN) algorithm.

Upon the introduction of a new data instance, the algorithm calculates the distances between the query point and all samples in the training dataset using an appropriate distance metric. It then selects the *k* closest data points three nearest neighbours in this case based on their proximity to the query point. The predicted output is subsequently determined from these neighbouring instances. In classification scenarios, KNN assigns the class label through a majority voting scheme among the selected neighbours. However, for regression tasks such as crop yield estimation, the model computes the mean of the target values associated with the nearest neighbours to generate the final prediction. As a non-parametric and instance-based learning technique, KNN does not involve an explicit training phase to estimate model parameters. Instead, it depends on the similarity structure of the dataset to make predictions. This characteristic allows the algorithm to effectively capture localized relationships within the agricultural data, thereby enhancing the accuracy of cowpea yield estimation.

Models' Hyperparameter Tuning

To enhance the models' predictive accuracy, a structured hyperparameter tuning process was applied individually to each algorithm. Proper tuning of hyperparameters helps improve model generalization and reduce prediction error.

For the K-Nearest Neighbors (KNN) model, hyperparameter tuning was performed using GridSearchCV from the *scikit-learn* library. The main parameter optimized was the number of neighbors (n_neighbors), with the search space defined as {3, 5, 7, 9, 11, 13, 15}. The optimal value identified was n_neighbors = 3, which produced the lowest prediction error.

Hyperparameter optimization for the Convolutional Neural Network (CNN) model was conducted using the Keras Tuner framework, employing the RandomSearch strategy to systematically explore a range of parameter configurations and identify optimal model settings. The parameters explored included the number of convolutional filters, kernel sizes, the option of adding a second convolutional layer, the number of units in the dense layer, and the optimizer. The search space included: conv_1_filters (16–64, step 16), conv_1_kernel {2,3}, use_additional_conv_layer {True, False}, conv_2_filters (32–128, step 32), conv_2_kernel {2,3}, dense_1_units (32–128, step 32), and optimizer {Adam, RMSprop}. The optimal configuration was conv_1_filters = 16, conv_1_kernel = 2, no additional convolutional layer, dense_1_units = 128, and optimizer = RMSprop.

Hyperparameter optimization for the Long Short-Term Memory (LSTM) architecture was likewise conducted through the application of Keras Tuner's RandomSearch strategy. The parameters considered were the number of LSTM units, the inclusion of an additional dense layer, the number of neurons in that layer, and the optimizer. The search space included lstm_units (32–128, step 32) and use_additional_dense_layer {True, False}, with optimizer options {Adam, RMSprop}. The optimal configuration was lstm_units = 96, use_additional_dense_layer = True, dense_units = 64, and optimizer = RMSprop.

In general, the application of hyperparameter optimization enhanced the predictive capabilities of all evaluated models, resulting in improved generalization, reduced prediction errors, and more reliable as well as precise estimations of cowpea yield.

Model Evaluation

The performance of the proposed models was assessed using four standard regression evaluation metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2). These indicators quantify the

deviation between the predicted outputs and the observed cowpea yield values, thereby providing a measure of prediction accuracy and model reliability.

Models	MAE	MSE	RMSE	R^2
CNN	0.0492	0.0040	0.0639	0.8754
LSTM	0.0558	0.0054	0.0739	0.8334
KNN	0.0404	0.0079	0.0893	0.7566

MAE measures the average absolute prediction error:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (7)$$

MSE calculates the average squared error:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (8)$$

RMSE is the square root of the mean squared error:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (9)$$

The coefficient of determination (R^2) is a statistical metric used to assess the extent to which a model accounts for the variability present in the observed data:

$$R^2 = 1 - \frac{\Sigma(y_i - \hat{y}_i)^2}{\Sigma(y_i - \bar{y})^2} \quad (10)$$

RESULTS AND DISCUSSION

The result of Table 1 provides a detailed comparison of the predictive performance of the three optimized models CNN, LSTM, and KNN across four evaluation metrics

Table 1: Model comparison

The CNN model demonstrates superior accuracy, achieving an R^2 value of 0.8754, a MAE of 0.0492, and an RMSE of 0.0639. In contrast, the LSTM model exhibits intermediate performance, with an R^2 of 0.8334, MAE of 0.0558, and RMSE of 0.0739. Although the KNN model attains the lowest MAE (0.0404), it shows the least satisfactory overall performance, reflected by the lowest R^2 (0.7566) and the highest RMSE (0.0893). This suggests that, despite producing smaller mean absolute errors, KNN is characterized by greater variability in its predictions and accounts for substantially less variation in yield compared to the deep learning-based models.

Figure 5 compares the performance of CNN, LSTM and KNN. The findings indicate that Convolutional Neural Network (CNN) model attains the highest coefficient of determination (R^2), reflecting superior predictive performance relative to the other evaluated models.

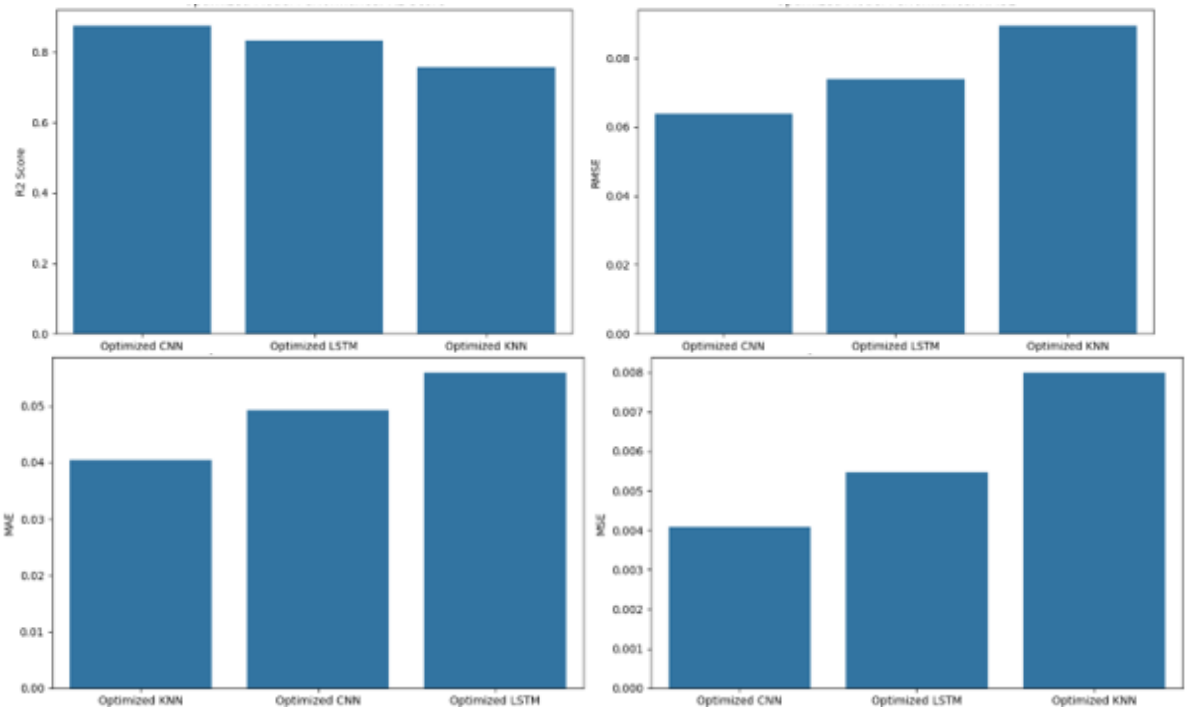

Figure 5: model performance comparison bar chart

Conversely, KNN model records the largest Root Mean Square Error (RMSE) and Mean Square Error (MSE), implying a comparatively higher level of prediction inaccuracy. LSTM model shows moderate performance, with error metrics between CNN and KNN. Overall, the figure demonstrates that the optimized CNN model provides the most reliable prediction performance, while KNN performs the weakest among the three models.

figure 6 compares the performance of three models KNN, CNN and LSTM in predicting crop yield. The scatter plot visualizations indicate that all three models exhibit strong predictive capability, as the estimated values align closely with the observed yield values within the range of 0.50 to

1.00. Furthermore, the tight concentration of data points around the ideal prediction line suggests a high degree of accuracy, reflecting minimal deviation and robust overall model performance.

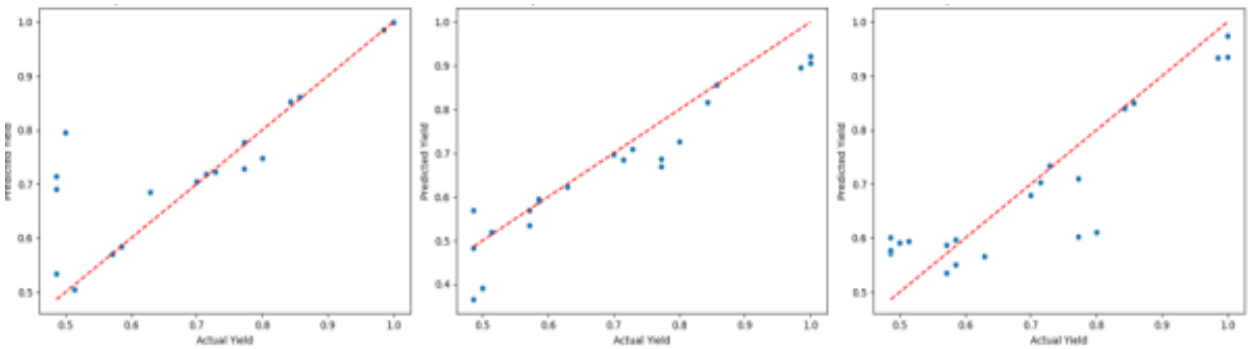

Figure 6: Actual vs predicted yield scatter plots (KNN, CNN and LSTM)

The results reveal minimal variability among the three models, as their predictions exhibit a high degree of consistency with only marginal differences observed for corresponding actual yield values. Additionally, there is no evident systematic bias in their outputs, as none of the models consistently overestimate or underestimate the target variable, thereby supporting their reliability.

Figure 7 presents the progression of training and validation loss for the CNN and LSTM architectures across 100 epochs. Both models exhibit swift convergence during the initial phase, particularly within the first 10 epochs, where loss values decline markedly from approximately 0.53 to values approaching zero.

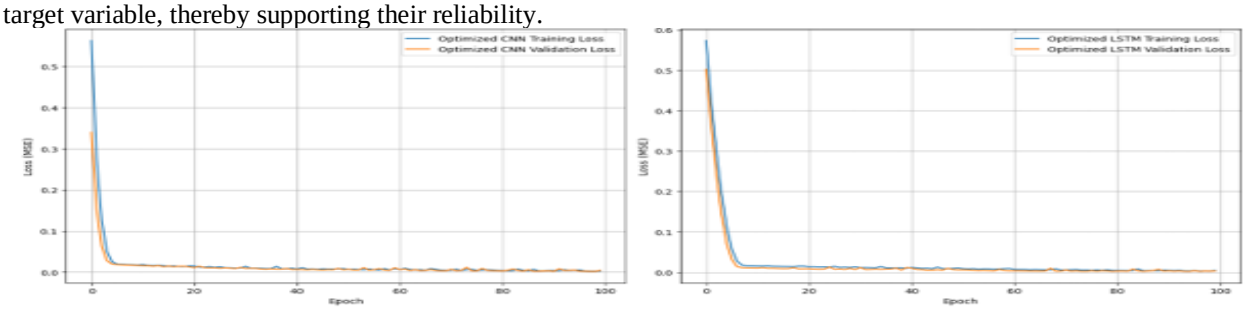

Figure 7: A comparative analysis of training and validation loss trajectories for Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models.

Beyond this stage, the loss curves stabilize, remaining consistently low for the remainder of training. CNN attains a validation loss of approximately 0.01, while the LSTM continues to improve gradually, reaching a validation loss close to 0.00 by the final epoch. These patterns reflect high learning efficiency and suggest that overfitting is effectively minimized throughout the training process.

Figure 8 presents residual error histograms for the KNN, CNN, and LSTM models.

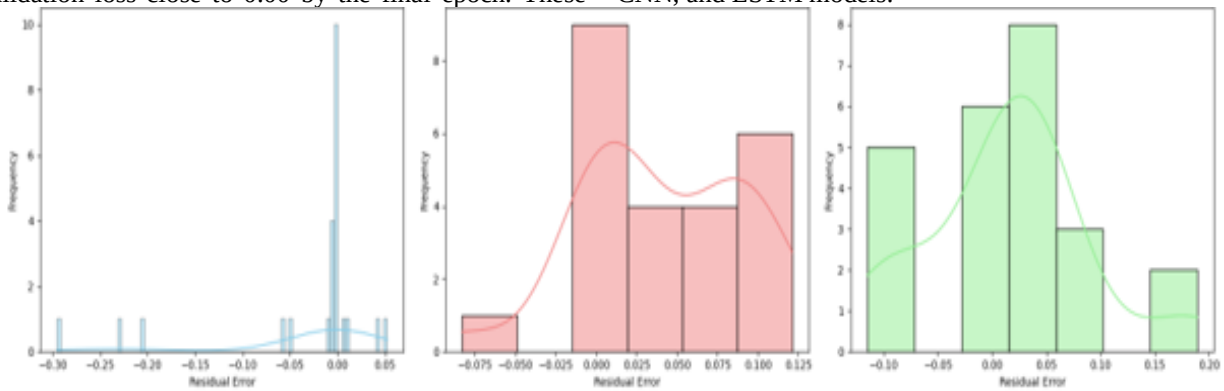

Figure 8: Residual error histograms (KNN, CNN and LSTM)

It shows the distribution of prediction errors across the residual range from -0.30 to 0.20, where the KNN model (blue) exhibits a left-skewed distribution with the highest frequency of residuals occurring at -0.30 with a count of 1.0, gradually decreasing to zero at 0.15, while both the CNN (red) and LSTM (green) models display uniform distributions with constant frequencies of 0.5 across all residual values, indicating that the KNN model produces a more varied error pattern with distinct peaks, whereas the CNN and LSTM models show consistent and identical error distributions throughout the entire residual range.

Figure 9 presents residuals versus predicted yield scatter plots for the KNN model on the left and the CNN and LSTM models on the right, comparing their prediction errors across the yield range from 0.50 to 1.00.

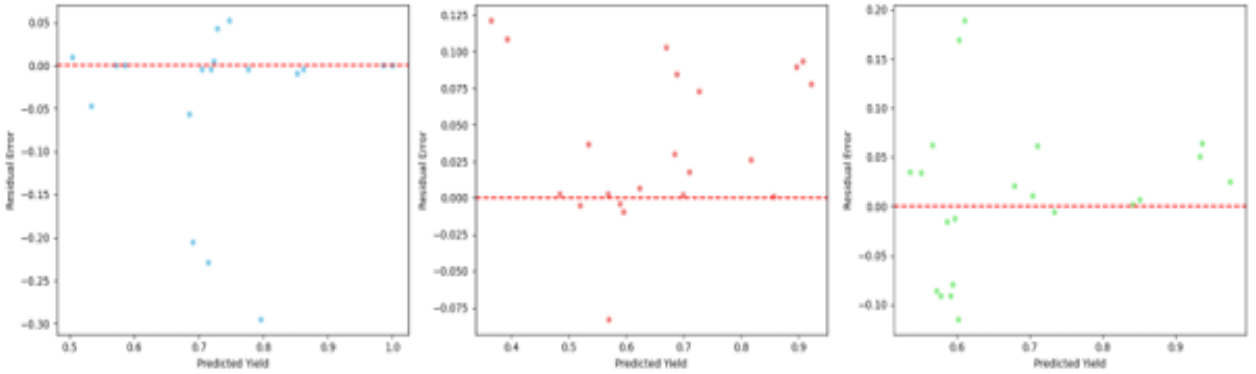

Figure 9: Residuals vs predicted scatter plots (KNN, CNN and LSTM)

The KNN model demonstrates perfect predictions with zero residual error across all yield values, while the CNN and LSTM models show systematic negative residual errors that become increasingly pronounced at higher predicted yields, with CNN residuals ranging from 0.00 to 0.07 and LSTM residuals ranging from -0.005 to -0.12, indicating that both deep learning models tend to under-predict yields, with pronounced effects observed in the higher range of yield values, with the LSTM model exhibiting the largest negative residuals and thus the greatest under-prediction bias among the three models.

Figure 10 displays box plot comparisons of residual errors for KNN, CNN, and LSTM models, summarizing their prediction error distributions through key statistical metrics namely the maximum value, the third quartile (Q3), the median, the first quartile (Q1), and the minimum value,

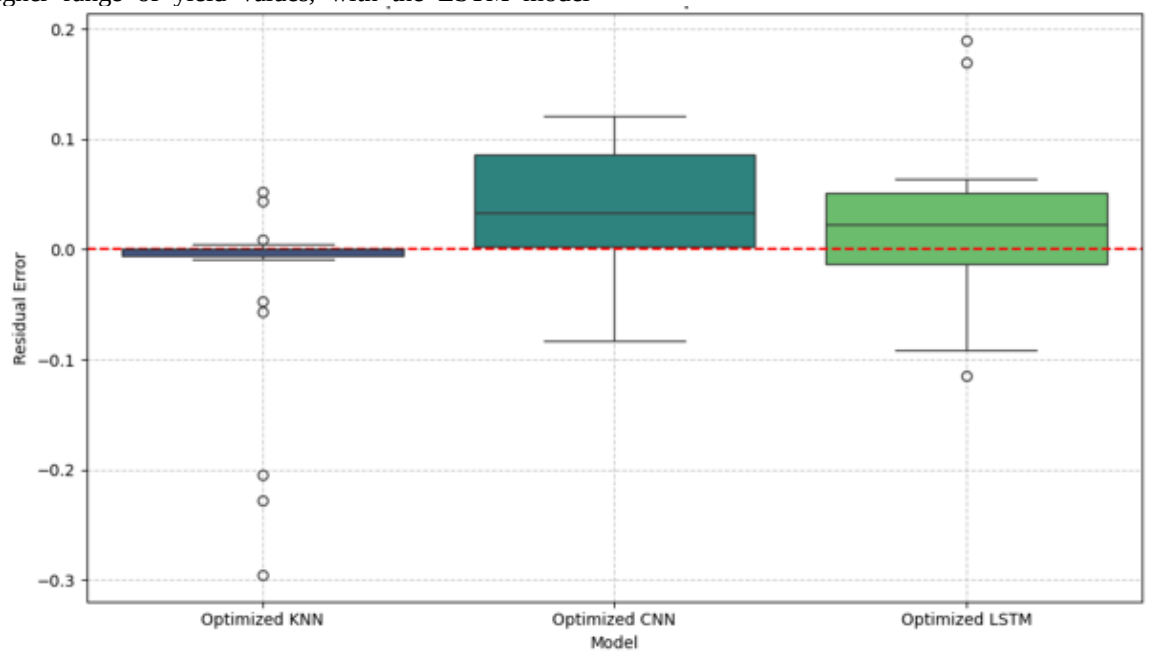

Figure 10: Error distribution boxplot

In the figure, KNN model shows residuals ranging from -0.20 to 0.04 with a median of 0.00 and a Q1 of -0.29, indicating a wide negative error spread, while the Optimized CNN model exhibits residuals from -0.05 to 0.09 with a median of 0.00 and a Q1 of -0.07, demonstrating a more balanced and narrower error distribution, and LSTM model presents residuals between -0.10 and 0.06 with a median of 0.00 and a Q1 of -0.11, showing an intermediate error range that is narrower than KNN but slightly wider than CNN, collectively revealing that CNN achieves the most compact residual distribution with the smallest interquartile range and thus the most consistent prediction accuracy among the three models.

CONCLUSION

This research involved the design and assessment of three machine learning approaches Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and the K-Nearest Neighbours (KNN) algorithm for the purpose of forecasting cowpea yield. The models were trained and evaluated using data obtained from IoT-enabled sensors, which included measurements of plant morphological characteristics alongside soil-related environmental parameters. Feature selection combining correlation analysis and agronomic knowledge identified number of flowers (r = 0.75), plant height (r = 0.67), fresh weight (r = 0.58), and number of leaves (r = 0.57) as the strongest yield predictors, while soil nutrients (N, P, K) and moisture were retained due to biological importance despite weaker statistical correlations. Principal Component Analysis reduced dimensionality while preserving over 95% of variance.

Systematic tuning of hyperparameters led to a marked improvement in the predictive capability of the models. Among the approaches examined, the refined Convolutional Neural Network (CNN) exhibited superior performance, attaining an R^2 of 0.8754, a mean absolute error (MAE) of 0.0492, and a root mean square error (RMSE) of 0.0639. This exceeded the results obtained with the Long Short-Term Memory (LSTM) model (R^2 = 0.8334) and the K-Nearest Neighbors (KNN) method (R^2 = 0.7566). Furthermore, analysis of the training and validation loss trajectories indicated that both deep learning models converged quickly within approximately 10 epochs while maintaining minimal signs of overfitting. Residual analysis revealed that CNN produced the most compact error distribution (range: -0.05 to 0.09) with the smallest interquartile range, indicating the most consistent prediction accuracy. However, both CNN and LSTM showed systematic under-prediction bias at higher yield values, with LSTM exhibiting the largest negative residuals.

The study demonstrates that deep learning approaches, particularly CNN, can effectively model crop yield with relatively small datasets when combined with appropriate feature selection and hyperparameter optimization. The integration of statistical feature selection with agronomic domain knowledge enhances model robustness and biological interpretability. These findings support improved decision-making in precision agriculture, enabling more accurate yield forecasting for sustainable cowpea production. Future work should validate models across diverse environments and integrate remote sensing data to enhance predictive capabilities.

REFERENCE

Badar, W., Ramzan, S., Raza, A., Fitriyani, N. L., Syafrudin, M., & Lee, S. W. (2025). Enhanced interpretable forecasting of cryptocurrency prices using autoencoder features and a hybrid CNN-LSTM model. *Mathematics*, *13*(12), 1908.

Badshah, A., Alkazemi, B. Y., Din, F., Zamli, K. Z., & Haris, M. (2024). Crop classification and yield prediction using robust machine learning models for agricultural sustainability. *IEEe Access*, *12*, 162799-162813.

Dhivya, S., Malar, M. K., & Anusuya, P. (2015). Secure Data using k-Nearest Neighbor (kNN) Classification-Survey.

Engen, M., Sandø, E., Sjølander, B. L. O., Arenberg, S., Gupta, R., & Goodwin, M. (2021). Farm-scale crop yield prediction from multi-temporal data using deep hybrid neural networks. *Agronomy*, *11*(12), 2576.

Ghojogh, B., & Ghodsi, A. (2023). Recurrent neural networks and long short-term memory networks: Tutorial and survey. *arXiv preprint arXiv:2304.11461*.

Huang, C., Wang, Z., Ren, X., Ma, X., Zhou, M., Ge, X., ... & Fu, S. (2022). Evaluation of soil quality in a composite pecan orchard agroforestry system based on the smallest data set. *Sustainability*, *14*(17), 10665.

Jabed, M. A., & Murad, M. A. A. (2024). Crop yield prediction in agriculture: A comprehensive review of machine learning and deep learning approaches, with insights for future research and sustainability. *Heliyon*, *10*(24).

Jahan, N., Sarker, U., Saikat, M. M. H., Hossain, M. M., Azam, M. G., Ali, D., ... & Golokhvast, K. S. (2023). Evaluation of yield attributes and bioactive phytochemicals of twenty amaranth genotypes of Bengal floodplain. *Heliyon*, *9*(9).

Kalmani, V. H., Dharwadkar, N. V., & Thapa, V. (2025). Crop Yield Prediction using Deep Learning Algorithm based on CNN-LSTM with Attention Layer and Skip Connection. *Indian Journal of Agricultural Research*, *59*(8).

Khaki, S., & Wang, L. (2019). Crop yield prediction using deep neural networks. *Frontiers in plant science*, *10*, 621.

Krichen, M., & Mihoub, A. (2025). Long short-term memory networks: A comprehensive survey. *AI*, *6*(9), 215.

Lan, Y., Wang, X., Gao, L., & Chen, X. (2025). Cotton Yield Prediction with Gaussian Distribution Sampling and Variational AutoEncoder. *Applied Sciences*, *15*(18), 9947.

Magray JA, Zargar SA, Islam T, Nawchoo IA. Impact of habitat variability on growth dynamics of Bergenia ciliata (Haw.) Sternb. along an altitudinal gradient in Kashmir Himalaya. Plant Sci. Today, [Internet]. 2022 Jan. 1 [cited 2025 Nov. 26];9(1):144–149. Available from: https://www.horizonepublishing.com/journals/index.php/PST/article/view/1367

Maleki-Meighani, R., Khadivi, A., & Tunç, Y. (2025). Multivariate Analysis of Morphological Variables in Berberis integerrima L., a Neglected Medicinal Fruit. *Food Science & Nutrition*, *13*(5), e70245.

Mansouri, S., Boulares, S., & Chabchoub, S. (2024). Machine Learning for Early Diabetes Detection and Diagnosis. *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, *15*(1), 216-230.

Niu, L., Cui, Q., Luo, J., Huang, H., & Zhang, J. (2024). Unconfined compressive strength prediction of rock materials based on machine learning. *Journal of Engineering and Applied Science*, *71*(1), 137.

Oikonomidis, A., Catal, C., & Kassahun, A. (2023). Deep learning for crop yield prediction: a systematic literature review. *New Zealand Journal of Crop and Horticultural Science*, *51*(1), 1-26.

Okut, H. (2021). Deep learning for subtyping and prediction of diseases: Long-short term memory. In *Deep Learning Applications*. IntechOpen.

Parashar, N., Johri, P., Khan, A., Gaur, N., & Kadry, S. (2024). An integrated analysis of yield prediction models: A comprehensive review of advancements and challenges. *Computers, Materials, & Continua*, *80*(1), 389.

Saadatfar, H., Khosravi, S., Joloudari, J. H., Mosavi, A., & Shamshirband, S. (2020). A new K-nearest neighbors classifier for big data based on efficient data pruning. *Mathematics*, *8*(2), 286.

Sharma, M., & Sharma, S. K. (2013). Generalized K-nearest neighbour algorithm-a predicting tool. *International Journal of Advanced Research in Computer Science and Software Engineering*, *3*(11), 1-4.

Sikandar, S., Mahum, R., & Aladhadh, S. (2023). Automatic Crop Expert System Using Improved LSTM with Attention Block. *Computer Systems Science & Engineering*, *47*(2).

Sushanth, P., & Santhi Sree, K. (2025). *Automatic text summarization using long short-term memory (LSTM).* International Journal for Research in Applied Science and Engineering Technology**.** https://doi.org/10.22214/ijraset.2025.71560

Wang, N., Ma, Z., Huo, P., Liu, X., He, Z., & Lu, K. (2023). 3D convolutional neural network with dimension reduction and metric learning for crop yield prediction based on remote sensing data. *Applied Sciences*, *13*(24), 13305.

Waqas, M., & Humphries, U. W. (2024). A critical review of RNN and LSTM variants in hydrological time series predictions. *MethodsX*, *13*, 102946.

Wen,L. (2025). A Comparison Between the K-Nearest Neighbors Algorithm and Logistic Regression in the Field of Cell Type Annotation. Theoretical and Natural Science,93,51-56.

Zhao,Y. (2025). From RNNs to BERT: A Review of Neural Models for Sequence Learning. Theoretical and Natural Science,130,118-123.