

Development of an Improved Hybrid Deep Learning Model for Cassava Yield Prediction

RAJI Mumini Oyetunji¹, AYOADE Olusola Bamidele^{2*}, AKINDELE Aminat Adejoke³, YUSUF-MASHOPA Kemi Jemilat⁴, ABDULRAUFF Muinat Folake⁵, RAJI Ibrahim Adebayo⁶, MUSAH Fatima Bolanle⁷, ADEGBIJI Abiodun Timothy⁸, AMUDA Oluwafemi Michael⁹

^{1,2,3,4,5,6,7,8}Department of Data Science, Informatics and Computer Science, Emmanuel Alayande University of Education, Oyo, Nigeria

⁹Directorate of Information Communication Technology, Emmanuel Alayande University of Education, Oyo, Nigeria

ARTICLE INFO	ABSTRACT
Published Online: 11 July 2026	Artificial Intelligence (AI) has significantly transformed the agricultural sector by enhancing crop yield forecasting, decision-making, and the analysis of soil and environmental conditions. Consequently, many studies have adopted deep learning (DL) techniques such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks for crop yield prediction. CNNs are effective in extracting spatial features from input data, while LSTMs capture temporal and phenological patterns, both of which are critical for accurate yield estimation. This study proposes a novel hybrid framework for cassava yield prediction that integrates CNN, LSTM, and Harris Hawk Optimisation (HHO). Soil and meteorological datasets were obtained from SoilGrids250m version 2.0 and the Nigerian Meteorological Agency (NIMET), Abuja, respectively, while cassava yield data were sourced from the Food and Agriculture Organisation Statistics. The datasets were preprocessed through noise removal, normalisation, and validation using a 5-fold cross-validation technique. The proposed HHO-CNN-LSTM model was evaluated against a conventional CNN-LSTM model using R-square (R^2), mean absolute error (MAE), mean squared error (MSE), root mean square error (RMSE), and mean absolute percentage error (MAPE). Results showed that the CNN-LSTM model performed poorly, with $R^2 = -0.56726$, $MAE = 1.431207$, $MSE = 2.050261$, $RMSE = 1.431075$, and $MAPE = 13.54528$. In contrast, the HHO-CNN-LSTM model achieved superior performance, recording $R^2 = 0.370277$, $MAE = 0.904463$, $MSE = 0.82379$, $RMSE = 0.907629$, and $MAPE = 8.745989$. These findings demonstrate that integrating HHO significantly improves prediction accuracy and model reliability, making the proposed approach a valuable decision-support tool for sustainable cassava production and agricultural resilience.
Corresponding Author: AYOADE Olusola Bamidele	
KEYWORDS: Agricultural decision-support systems, Cassava yield prediction, Convolutional Neural Network (CNN), Harris Hawk Optimisation (HHO), Long Short-Term Memory (LSTM)	

INTRODUCTION

Crop yield prediction is extremely difficult because it is dependent on a variety of factors such as crop genotype, environmental conditions, management practices, and their interactions. For example, genotype and environmental factors frequently interact with each other, making yield prediction more difficult. Environmental factors such as weather components frequently have complex nonlinear

effects that are difficult to estimate precisely (Khaki *et al.*, 2020). Accurate yield predictions, on the other hand, help farmers make informed economic and management decisions, as well as contribute to famine prevention and global food security (Chang *et al.*, 2023). Ansarifar *et al.* (2021) emphasise the importance of crop yield prediction in addressing global food security concerns, citing genotype, environment, and management, as well as their complex

interactions, as significant barriers. Concurrently, Al-Adhaileh and Aldhyani (2022) identify early crop yield prediction as a critical component in mitigating famine risks and forecasting food availability for the growing global population.

Traditional crop yield prediction methods struggle to capture the complex interactions among genotype, environment, and management practices, resulting in less accurate forecasts. Advancements in AI and machine learning, particularly in hybrid models, present an opportunity to significantly improve prediction accuracy. Hybrid models, such as CNN-LSTM, can better analyse spatial and temporal data, resulting in more accurate predictions. This accuracy is critical for ensuring food security, making informed farming decisions, and promoting economic stability by optimising resource use and mitigating famine risks (Wang *et al.*, 2024). As a result, some researchers have created deep learning models that predict crop yields using a hybrid of CNN and LSTM.

For example, Taremwa *et al.* (2025) developed a convolutional neural network and long short-term memory (CNN-LSTM) model to predict maize yields using biannual remotely sensed data and maize yield labels from the Zonal Agricultural Research and Development Institute (ZARDI) zones in Uganda. The authors employed synthetic data augmentation with the Synthetic Minority Oversampling Technique for Regression (SMOTE) and feature dimensionality reduction based on feature importance analysis, as accessing high-quality yield records across ZARDI zones remains challenging due to high costs and time constraints. The CNN-LSTM model outperformed baseline models due to its ability to select features and perform extensive hyperparameter tuning. It achieved an MSE of 0.107 tonnes², MAE of 0.267 tonnes, RMSE of 0.327 tonnes, and R² score of 0.78.

Zhou *et al.* (2023) created a field-scale maize yield prediction model called the convolutional neural network-attention-long short-term memory network (CNN-attention-LSTM) model, which uses multimodal remote sensing data collected by multispectral and light detection and ranging (LIDAR) sensors mounted on unmanned aerial vehicles (UAVs). The model incorporates meteorological data throughout the crop reproductive stages and employs the normalised difference vegetation index (NDVI), normalised difference red edge (NDRE), soil-adjusted vegetation index (SAVI), and enhanced vegetation index (EVI) for the initial part of the vegetative stage (initial part of the V period), the later part of the vegetative stage (later part of the V period), the reproductive stage (R period), and the maturity stage. The yield estimation produced exceptional results, with an R² value of 0.78 and an RMSE of 8.27%.

Moreover, hybrid CNN-LSTM models are effective at capturing both spatial and temporal relationships in crop yield data, using convolutional layers for spatial features and LSTM units for sequential patterns (Khaki *et al.*, 2020).

However, due to their deep architecture, such models frequently necessitate significant computational resources and lengthy training times (Javaregowda *et al.*, 2025). They are also highly sensitive to hyperparameter choices like learning rate and network depth, which have a significant impact on training dynamics and performance (Alibabaei *et al.*, 2021). In agricultural settings with sparse or noisy datasets, these architectures are prone to overfitting, resulting in poor generalisation performance (Ali *et al.*, 2024). Furthermore, gradient-based optimisation in deep hybrid models may experience slow convergence and difficulty escaping local minima, especially in complex, nonlinear data spaces (Choubey *et al.*, 2025).

As a result, this study uses Harris Hawk Optimisation (HHO) to improve the CNN-LSTM model by addressing these issues. Harris Hawks Optimisation (HHO) is a powerful metaheuristic algorithm that uses a global search mechanism to automatically optimise hyperparameters in deep learning models. HHO reduces the need for manual tuning by adaptively balancing exploration and exploitation while improving model generalisation, convergence stability, and resistance to overfitting (Heidari *et al.*, 2019; Probert, 2012). Its ability to dynamically escape local optima makes it ideal for training complex hybrid models on noisy and nonlinear agricultural datasets like those found in Nigerian crop yield studies (Eiben and Smith, 2015).

In cassava yield prediction, combining HHO with Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks improves structural and parametric optimisation. CNNs efficiently extract spatial features from agro-environmental data, whereas LSTMs model temporal yield patterns influenced by climatic variability (LeCun *et al.*, 2015; Hochreiter and Schmidhuber, 1997). The CNN-LSTM-HHO framework is reliable for real-world cassava yield forecasting and agricultural decision support in Nigeria due to its improved accuracy and robustness through HHO optimisation (Kamilaris and Prenafeta-Boldú, 2018; Liakos *et al.*, 2018).

MATERIALS AND METHODS

This section contains detailed information on dataset acquisition, dataset preprocessing, feature extraction, feature selection, and model formulation for cassava yield prediction.

Dataset Acquisition

This study collected three types of data: soil, weather, and cassava yield. The soil dataset includes data from seven depths: 0, 5, 15, 30, 60, 100, and 200 cm. However, the relevant soil parameters for the six southwestern Nigerian states under consideration, namely sand, silt, clay, gravel content, cation exchange capacity, pH, organic carbon, and bulk density, were obtained from SoilGids250m version 2.0 at soilgrid.org. Srivastava *et al.* (2023) provide a summary of the results of an investigation of the soil physicochemical properties of an experimental

site regarding the dates of cassava planting and harvesting at Ikenne, Ogun State, between September 9, 2014 and September 7, 2015, as shown in Table 1. Furthermore, the weather dataset used in this study only included data from the six cassava-producing states of southwest Nigeria: Ondo, Ekiti, Ogun, Ibadan, and Osun. The climatic dataset consists of data samples with three features: rainfall, minimum temperature, and maximum temperature. Table 2 depicts the weather predictor elements applied over a few years from the dataset(s). Similarly, the cassava yield dataset covers the years 1963 to 2017 and includes two features: the "year" and the "yield outcome." Edamisan *et al.* (2020) used secondary data from the Food and Agriculture Organisation Statistics (FAOSTAT) to compile their research. Figure 1 depicts the data for the 112 samples presented in Table 3.

Dataset Preprocessing

To ensure the dataset's robustness and reliability for model training and testing, several preprocessing steps were taken. First, after smoothing the noisy data, missing values

were imputed in this case with the mean value of the same variable for other nations in the same year. The mean imputation technique was chosen over median imputation because it produces more accurate results. Furthermore, redundant or useless attributes were identified and removed from the dataset to avoid data bias issues and reduce the risk of invalid data errors during training. Following the handling of missing values and attribute reduction, the dataset was normalised using min-max normalisation, as described in equation 1.

$$\theta_n = \frac{\epsilon_n - \text{Min}(\epsilon)}{\text{Max}(\epsilon) - \text{Min}(\epsilon)} \times 100 \quad (1)$$

Where

Θ_n = In the dataset, the nth normalised value

ϵ_n = the nth value in the dataset

Min (ϵ) = Minimum value in the dataset

Max (ϵ) = Maximum value in the dataset

Table 1. Soil Physiochemical Properties at the Ikenne Experimental Site (Source: Srivastava *et al.*, 2023)

Characteristics	Ikenne	
Soil depth (cm)	0-15	15-30
p ^H (H ₂ O)	5.7	5.8
OC(g.kg ⁻¹)	12.3	9.3
N(g.kg ⁻¹)	0.98	0.74
P(mg.kg ⁻¹)	3.63	1.57
Exchangeable cations (cmol.kg⁻¹)		
Ca	3	2.74
Mg	0.81	0.88
K	0.08	0.16
Na	0.07	0.07
Particle size distribution (g.kg⁻¹)		
Gravel content (%)	2-5%	10-15%
Sand	730	710
Silt	60	60
Clay	210	230
Textural class	Sandy clay loam	Sandy clay loam

Table 2. Sample of Climate Datasets Utilised for the Study (Source: NIMET, Abuja)

YEAR	April Avg. Max. Tem (°C)	Aug Avg. Max. Tem (°C)	Oct Avg. Max. Tem (°C)	April Avg. Min. Tem (°C)	Aug Avg. Min. Tem (°C)	Oct Avg. Min. Tem (°C)	April Avg. Rainfall (mm)	Aug Avg. Rainfall (mm)	Oct Avg. Rainfall (mm)
1988	31.18	26.36	29.28	25.30	23.00	24.00	173.5	152.3	201.8
1989	31.00	26.64	28.54	25.10	23.10	23.70	122.0	220.6	164.2
1990	30.93	26.54	28.7	25.20	23.00	23.80	184.8	168.3	167.2
1991	30.34	26.23	28.02	24.20	22.80	23.10	175.2	223.3	190.6
1992	31.41	25.89	28.88	25.00	22.70	23.60	130.9	165.7	159.5
1993	31.54	27.16	29.33	25.20	23.20	23.90	62.50	205.7	139.9
1994	31.02	26.64	28.70	25.20	23.10	23.70	96.90	147.3	192.1
1995	30.95	27.19	28.72	25.30	23.60	23.90	135.8	269.9	187.3

“Development of an Improved Hybrid Deep Learning Model for Cassava Yield Prediction”

1996	30.81	26.63	28.61	24.70	23.10	23.50	147.8	222.7	176.3
1997	30.05	26.90	29.26	24.20	22.80	23.40	211.5	154.9	210.4
1998	33.00	26.61	28.82	26.40	22.90	24.00	91.20	87.2	178.9
1999	30.67	26.88	28.14	24.40	23.20	23.20	100.8	237.6	258.8
2000	30.90	26.50	28.70	25.00	23.00	24.00	96.10	244.8	146.7
2001	30.76	25.86	29.41	25.10	23.00	24.30	169.9	98.10	73.10
2002	30.67	26.57	28.54	25.00	23.30	23.70	104.4	229.2	212.6
2003	30.74	26.87	29.65	25.10	23.50	24.40	164.1	125.4	170.7

Note: AVG. Max. Tem. (Average Maximum Temperature)

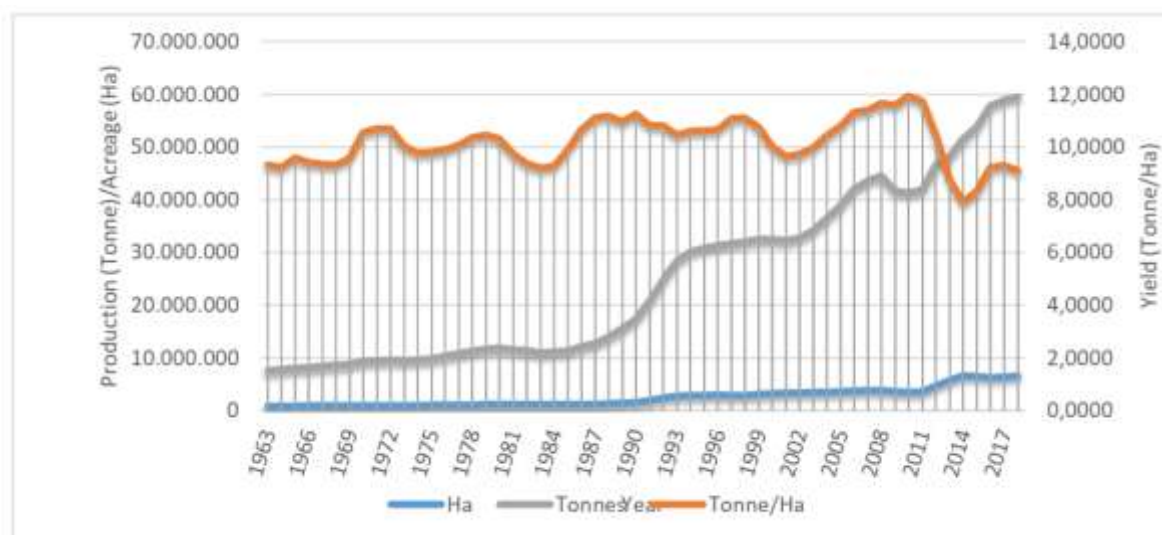


Figure 1.Nigerian Cassava Production, Yield, and Area (1963 – 2018) Edamisan and Kemisola, 2020 computation adapted from FAOSTAT, 2019

Table 3.Year and Yield Outcome of Cassava in Nigeria from 1963-2017 (Source: Edamisan and Kemisola, 2020)

Year	Yield (Tonnes/Acre)
1963	9.40
1964	9.20
1965	9.60
1966	9.50
1967	9.40
1968	9.40
1969	9.70
1970	10.40
1971	10.50
1972	10.50
1973	10.00
1974	9.90
1975	9.90
1976	10.00
1977	10.10
1978	10.20
1979	10.10
1980	9.90
1981	9.80
1982	9.70
1983	9.80

1984	10.90
1985	10.80
1986	10.60
1987	11.00
1988	11.00
1989	10.80
1990	11.20
1991	10.80
1992	10.90
1993	10.60
1994	10.70
1995	10.70
1996	10.80
1997	11.20
1998	11.20
1999	11.00
2000	10.00
2001	9.80
2002	9.80
2003	9.90
2004	10.00
2005	10.40
2006	11.00
2007	11.60
2008	11.70
2009	11.80
2010	12.00
2011	11.90
2012	11.40
2013	8.80
2014	7.90
2015	8.20
2016	9.40
2017	9.40

Architecture of the SSA-HHO-CNN-LSTM Model

Figure 2 depicts the general architecture for forecasting cassava production with a CNN-LSTM model based on HHO. This model starts with preprocessed input data. The CNN convolution and pooling layers extract spatial features from the input variable, which are then fed into the LSTM layer. The LSTM layer uses transmitted

spatial data to simulate irregular time information. At the moment, CNN-LSTM hyperparameters are tuned using HHO techniques. The CNN-LSTM approach may produce the expected cassava yield in the completely connected layer; it is made up of several encoded CNN kernels, LSTM hidden units, and fully linked layer units; the HHO selects the best hyperparameter structure and moves the encoded

particles.

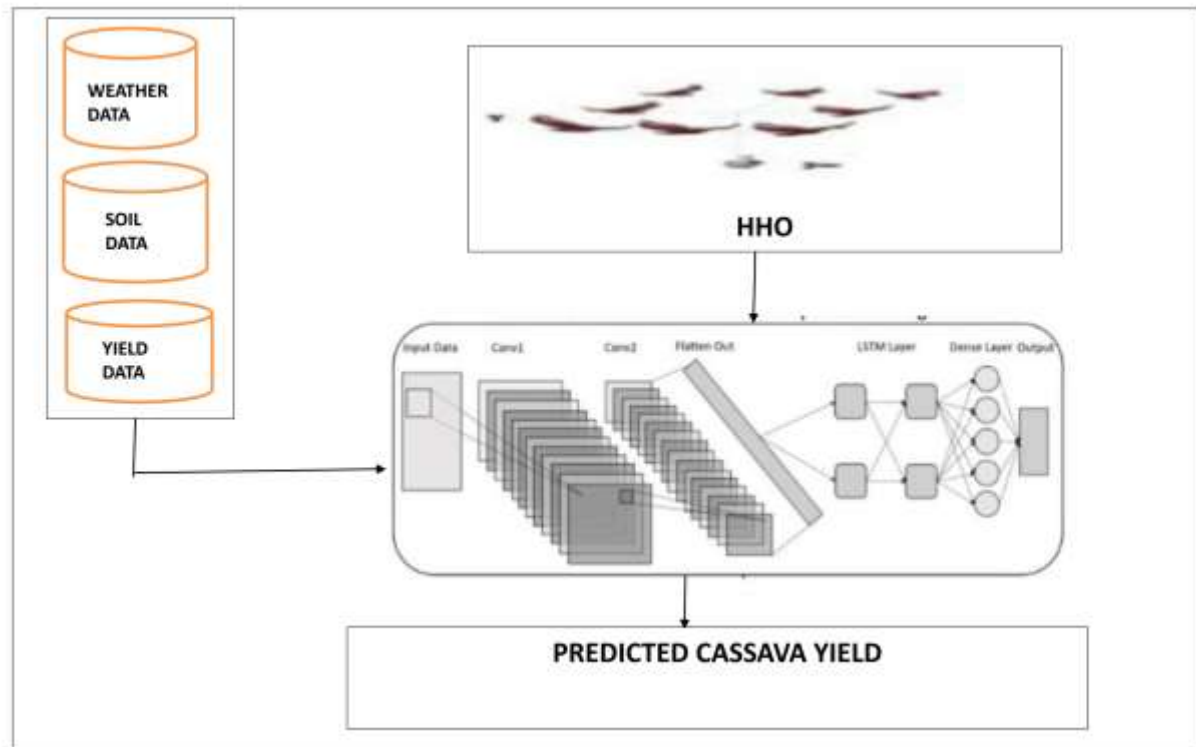


Figure 2. Architecture of the HHO-CNN-LSTM Model

Feature Extraction

The input dataset was analysed using a CNN to extract local features. The dimensions of the time-series meteorological data fed into the CNN are the length of the data (L) and the number of features/channels (N). Convolution filters were used to extract features from time series data. The number of feature maps generated equals the number of filters used in the feature map construction process during the convolution operation of Challa *et al.*'s (2021) model. The soil, weather, and cassava yield data were among the input datasets that were divided into many frames using the sliding window technique, as applied by Wang *et al.* (2021). CNN collects characteristics for each frame separately, disregarding any temporal context outside of the frame borders. Its fundamental components are the input gate, forget gate, output gate, and cell state. Several different time-scale correlations were captured by LSTM.

Feature Selection

Feature selection was carried out by backpropagating positive gradients to identify input parameters that maximise the activation of the targeted neurons, as in models developed by Khaki and Wang (2019) and Khaki et al. (2020). The researchers were able to determine the relative significance of each element in the features extracted. Initially, the CNN-LSTM model was trained on the entire training sample dataset to determine the average activation of each neuron in the LSTM cell's output at time step t. The gradient of activated neurons was set to one, while that of other neurons was set to zero. The

activated neurons' gradients were then backpropagated to the input space, and the magnitude of the gradient was used to identify the significant input variables.

The CNN-LSTM Neural Network Model

The CNN-LSTM model combines a CNN layer for linearly connecting cassava yield prediction factors such as weather, soil datasets, and yield performance and an LSTM layer for temporal feature modelling. We applied a layer of convolution and pooling to previously preprocessed soil, weather, and cassava yield performance datasets. This layer automatically extracts key elements from multivariate data that affect cassava production. The global active cassava yield forecast datasets are represented by the notation $X_t^0 = \{X_1, X_2, \dots, X_t\}$, where t is the year. The output feature X_{ij}^l from the previous convolutional layer is used to compute F_{ij}^l , the feature's outcome. W denotes the convolution kernel's weight, n the index value of the convolution filter, α the activation function, and b_j^l the bias for the jth feature map. Equation 3 represents the feature's outcome. Equation 2 represents the output of the feature vector, F_{ij}^l , from the lth convolutional layer.

$$F_{ij}^l = \alpha(b_j^l + \sum_{n=1}^n W_{n,j}^l X_{i+m-l,j}^0) \quad (2)$$

The extracted features from the convolutional layer will be sent to the pooling layer for sampling. Equation 3.3 depicts how the max-pooling layer works, where P is the

pooling size and S is the stride, which determines how far the pooling zone should be moved:

$$Y_{i,j}^l = \text{Max } f_{i \times S + p,j}^l \quad (3)$$

The CNN-LSTM method shows the relationship between yield performance, weather, and soil datasets for forecasting cassava production. The noise-cancelled features are then passed to the LSTM layer. Equations 4, 5, and 6 show the input, forget, and output gates that make up the LSTM layer.

$$e_i = \alpha (W_{ye}y_i + W_{ke}k_{i-1} + W_{ae}o_{i-1} + b_e) \quad (4)$$

$$f_i = \alpha (W_{yf}y_i + W_{kf}k_{i-1} + W_{af}o_{i-1} + b_f) \quad (5)$$

$$g_i = \alpha (W_{yg}y_i + W_{kg}k_{i-1} + W_{ag}o_{i-1} + b_g) \quad (6)$$

The LSTM layer can be used to develop a simple model for forecasting long-term cassava yield. Because each LSTM gate unit controls a continuous value ranging from 0 to 1, the hidden value of an LSTM cell is determined in each i step using equations 7 and 8.

$$a_i = f_i \circ a_{i-1} + e_i \circ \alpha (W_{ka}k_i + W_{ka}k_{i-1} + b_a) \quad (7)$$

$$k_i = g_i \circ \alpha (a_i) \quad (8)$$

Finally, the final CNN-LSTM layer generates the cassava yield prediction $Q^l = \{Q_1, Q_2, \dots, Q_l\}$, where I represents the number of units in the LSTM. Equation 9 is used to depict a fully linked layer, where b_i^{l-1} represents a bias, W is the weight of the ith node in layer I-1, and the jth node in layer I.

$$d_i^l = \sum_j W_{ji}^{l-1} (\alpha (a_i^{l-1}) + b_i^{l-1}) \quad (9)$$

The Harris Hawk Optimisation (HHO) Model

The Harris Hawk Optimisation (HHO) Model Algorithm Pseudocode is modelled after the actions of the hawk, one of nature's most crafty birds, which strikes a snake from multiple directions. The HHO algorithm represents this behaviour in two main phases: exploration and exploitation. The algorithm 1 depicts the Harris Hawk Optimisation (HHO) Model Algorithm Pseudocode.

Exploration Phase

Each hawk represents one possible solution. Equations 10 and 12 mathematically represent the location where a hawk may or may not find prey.

$$P(i+1) = \begin{cases} P_{rand}(i) - r_1 | P_{rand}(i) - 2r_2 P(i) | & s \geq 0.5 \\ (P_{snake}(i) - P_n(i)) - D & s < 0.5 \end{cases} \quad (10)$$

$$D = r_3 (L + r_4 (U - L)) \quad (11)$$

where $P_n(i)$ represents the average position of the current hawk population; r_1, r_2, r_3 , and r_4 are random numbers within (0,1) that are updated in each iteration; and L and U denote the search space's lower and upper bounds, respectively. For each iteration, $P_{snake}(i)$ represents the snake's current position, $P(i)$ represents the hawks' current position vector, and $P(i+1)$ represents their position vector. $P_n(i)$ represents the average position of each hawk in the current group. $P_n(i)$ is calculated using equation 12:

$$P_n(i) = \left(\frac{1}{N} \right) \sum_{t=1}^M P_t(i) \quad (12)$$

where $P_t(i)$ is the location of the i^{th} hawk in the current iteration, and M is the total number of hawks in the population.

Transition from Exploration to Exploitation

Equation 13 represents the prey or snake's energy:

$$H = 2H_0 \left(1 - \frac{i}{I} \right) \quad (13)$$

where H is the energy of flight for the prey, H_0 is its initial energy, and I is the maximum number of repeats.

Exploitation Phase

The exploitation phase of the HHO algorithm consists of the following steps:

(a) Soft besiege: The prey or snake has enough energy to attempt a series of misleading slithers to escape, but ultimately fails ($r \geq 0.5$ and $|H| \geq 0.5$). Before launching their surprise attack, the hawks attempt to wear down the snake by gently surrounding it. Equations 14–16 can be used to model this behaviour.

$$P(i+1) = \Delta P(i) - H |K P_{snake}(i) - P(i)| \quad (14)$$

$$\Delta P(i) = P_{snake}(i) - P(i) \quad (15)$$

$$K = 2(I - r_5) \quad (16)$$

where K denotes the snake's strength of slither during its attempt to escape, $\Delta P(i)$ is the difference between the snake's and the hawk's locations in the i^{th} iteration, and r_5 is a random number between 0 and 1.

(b) Hard besiege: Equation 17 can be used to represent this phase, which is characterised by the prey's extreme tiredness and low fleeing energy ($r \geq 0.5, |H| < 0.5$).

$$P(i+1) = P_{snake}(i) - H |\Delta P(i)| \quad (17)$$

(c) Soft besiege with progressive, rapid dives. Equation 18 shows that when $r < 0.5$ but $|H| \geq 0.5$, the snake

experiences a mild besiege before the surprise pounce, but still has enough energy to flee.

$$Y = P_{snake}(i) - H|KP_{snake}(i) - P(i)| \quad (18)$$

Based on Levy's flight-based patterns, the hawks dive according to the following rule, which is shown in Equations 19–21:

$$U = X + Y \times LF(J) \quad (19)$$

Equations 20 and 21 calculate the Levy flight function, $LF(J)$, where Y is a random vector of size $1 \times J$ and J is the problem dimension.

$$LF(x) = 0.01 \times \frac{uX\alpha}{|v^{1/\delta}|} \quad (20)$$

$$\alpha = \left\{ \left(\frac{r(1-\delta)X \sin(\Pi\delta/2)}{r(1+\delta/2)X\delta X 2(\delta-1/2)} \right) \right\}^{1/\delta} \quad (21)$$

where δ is a constant of 1.5, and u and v are random variables within (0,1). So, Equation 22 is the ultimate rule for updating the hawks' position.

$$P(i+1) = \begin{cases} Xif LF(X) < LF(P(i)) \\ Uif LF(U) < LF(P(i)) \end{cases} \quad (22)$$

where X and U are as calculated earlier in Equations 18 and 19.

(d) Hard besiege with progressive rapid dives: When $r < 0.5$ and $|H| < 0.5$, a harsh besiege occurs before the surprise pounce, as the snake lacks the energy to flee. Equation 23 can be used to express the following:

$$P(i+1) = \begin{cases} Xif LF(X) < LF(P(i)) \\ Uif LF(U) < LF(P(i)) \end{cases} \quad (23)$$

where X and U are calculated using Equations 24 – 25:

$$X = P_{snake}(i) - H|KP_{snake}(i) - P_n(i)| \quad (24)$$

$$U = X + Y \times LF(J) \quad (25)$$

Algorithm 1: Harris Hawk Optimisation (HHO) Model Algorithm Pseudocode

1. Set the initial iteration $t = 1$
 2. Initialise the random search space $X_i = (x_i^1, x_i^2, \dots, x_i^D)$ for the i th hawk of the D -dimensional problem within the search boundary $[UB, LB]$
 3. While $t \leq T$
 4. Evaluate the fitness value, $f(X_i)$, of all N hawks
 5. Label the best prey location as X_{prey}
 6. For $i = 1, 2, \dots, N$ (each hawk)
 7. Update the escaping energy E by Equation 13
 8. If $(|E| \geq 1)$
 9. Randomly choose one hawk location as X_{rand} from the search space
 10. Calculate the mean position vector X_{ave} using Equation 12
 11. Elseif $(|E| < 1)$
 12. Generate a random escaping chance of prey r in the range $[0, 1]$
 13. If $(r \geq 0.5 \text{ and } |E| \geq 0.5)$
 14. Update the new location using Equation 14
 15. Elseif $(r \geq 0.5 \text{ and } |E| < 0.5)$
 16. Update the new location using Equation 17
 17. Else if $(r < 0.5 \text{ and } |E| \geq 0.5)$
 18. Update the new location using Equation 22
 19. Else if $(r < 0.5 \text{ and } |E| < 0.5)$
 20. Update the new location using Equation .23
 21. Endif
 22. Endif
 23. EndFor
 24. Amend the search space X_i for $i = 1, 2, \dots, N$ based on the search boundary UB and LB
 25. $t = t+1$
 26. EndWhile
 27. Return the best location X_{prey}
-

The HHO-CNN-LSTM Model

Finding the optimal hyperparameter and architectural combinations is critical for deep neural network performance and efficiency in real-world scenarios. These configurations and search procedures are particularly time-consuming for CNN-LSTM, which combines CNN and LSTM to produce a larger number of hyperparameters. The LSTM and CNN hyperparameters have strong interactive effects that increase complexity when compared to monotonous deep learning models used for time series prediction tasks. The HHO model optimises several learning and network parameters to automatically find the CNN-LSTM model's best hyperparameters and topology. These parameters include the learning rate, dropout rate, number and size of filters in two convolutional layers, pooling layer size, and the number of hidden nodes in the LSTM recurrent layer and final dense layer (Xie et al., 2020). Table 4 shows the search ranges for each optimised parameter.

This study investigates the hyperparameters that are critical for CNN-LSTM representational capacity. The hidden node count of the LSTM layer, as well as variables such as dropout and learning rates, all have an impact on training properties and learning efficiency. Thus, the effects and confounding effects of various hyperparameters can be thoroughly investigated by closely monitoring the evolution of the proposed HHO model. Once its optimal configuration has been determined, the CNN-LSTM model is used to

solve time series prediction and classification problems. This is the process of using CNN-LSTM to find the optimal hyperparameters. Initially, the population of the suggested HHO is randomised. Each hawk represents one possible CNN-LSTM model configuration. The proposed CNN-LSTM model is trained on the provided training dataset. Every hawk in the model represents a specific structure and set of parameters. The validation set generates fitness scores, which are the error rate for classification problems and the root mean square error (RMSE) for regression problems. All of the hawks are told to use the solutions with the calculated fitness scores to find global optimality using the proposed HHO method.

Furthermore, after determining the optimal configuration, the best hawk creates the final CNN-LSTM model. Algorithm 2 depicts the HHO hyperparameter tuning of the CNN-LSTM model's algorithm pseudocode as it evolves. Likewise, the general operation of the proposed CNN-LSTM model based on HHO can be summarised by looking at the schematic design in Figure 3. As illustrated in the diagram, the recommended approach begins with defining the input datasets. Following that, preliminary steps such as data purification are carried out. Following this step, the training and testing phases are conducted using a 5-fold cross-validation process. The training and testing datasets were standardised before being passed to the CNN layer for feature extraction.

Table 4. The Hyperparameters' Search Range

Optimized Component	Hyperparameter	Range
Conv	No. of filters in 1 st layer	$[2^0, 2^{10}]$
	Filter size in 1 st layer	[1, 5]
	No. of filters in the 2 nd layer	$[2^0, 2^{10}]$
	Filter size in the 2 nd layer	[1, 5]
Pooling	Pooling size	[2, 5]
LSTM	No. of hidden nodes	[10, 500]
Dense	No. of nodes	[10, 200]
Learning configuration	Learning rate	$[10^{-5}, 10^{-1}]$
	Dropout rate	[0, 0.6]

In addition, the fully linked layer transfers the gathered features to the LSTM layer. The HHO was used to tune the CNN-LSTM model's hyperparameters and get the best results from the HHO-CNN-LSTM model's ability to predict cassava yield. The HHO-CNN-LSTM model was evaluated using four performance criteria: Mean Absolute

Error (MAE), Root Mean Squared Error (RMSE), Coefficient of Determination (R-Squared), and Mean Absolute Percentage Error (MAPE). Finally, the yield forecasted value was normalised to obtain the true cassava yield prediction value.

Algorithm 2: The HHO Optimised CNN-LSTM Model Hyperparameters Tuning Algorithm Pseudocode

```

1   Start
2   Initialise HHO population with each individual representing a specific network configuration of CNN-LSTM
3   Prepare training, validation and test sets
4   For (each hawk i in the population) do
5   {

```

```

6   Decode  $i$  into the corresponding CNN-LSTM network
7   Train the network using the training set
8   Evaluate the network on the validation set and calculate the fitness score
9   } End For
10  Identify the search space for the  $i$ th hawk of the  $D$ -dimensional problem within the search
    boundary with the best fitness scores
11  While ( $t < Max\_iter$ )
12  {
13  Evolve the HHO population according to the search mechanism of the proposed HHO
    method, i.e., lines 4-27 in Algorithm 3.1
14  } End While
15  Output the most optimal solution  $X\alpha$ 
16  Decode  $X\alpha$  into the corresponding CNN-LSTM network
17  Train the identified optimised network on the combined training and validation sets
18  Evaluate the optimised CNN-LSTM network on the test set and output the test result
19  End

```

RESULTS AND DISCUSSION OF FINDINGS

A total of 5,400 datasets were used in a 70:30 split, with 70% used for training and 30% for testing each model. The CNN-LSTM and HHO-CNN-LSTM models were used as classifiers. The datasets produced the best results when all techniques were applied to testing and training operations with a threshold value of 0.24.

Outcomes of Convolutional Neural Network-Long Short-Time Memory

The CNN-LSTM model was trained using three datasets: soil, weather, and past yield. After the CNN analysed the data, the LSTM gained knowledge. Following the output, there was a completely connected layer. The CNN-LSTM model consists of four LSTMs, each with fifty nodes. To prevent overfitting, 0.2 dropout regularisation and 100 epochs were used in a batch of 14. The maximum

number of iterations is 900, with nine iterations per epoch. Furthermore, the ReLu activation function was applied to the model's learning rate of 0.002048. Tables 5 and 6 show the resulting output, which was obtained with a timestep of 14 because the model was trained on a 14-year dataset.

Architecture of the SSA-HHO-CNN-LSTM Model

However, data from Table 5 show that the interval rate between actual and expected yield values ranges from 1.41 to 1.44. Table 6 shows that the CNN-LSTM model produced mean absolute error, mean square error, root mean square error, mean absolute percentage error, and r-square performance assessment metrics of 1.431207, 2.050261, 1.431873, 13.84528, and -0.56726, respectively. Figure 4 depicts the actual and expected cassava yield from 2009 to 2022 using the CNN-LSTM model procedures for testing and training.

Table 5. Results of Actual and Predicted Cassava Yield Using CNN-LSTM from 2009 to 2022

Year	Actual Yield (Tonnage ha ⁻¹)	Predicted Yield (Tonnage ha ⁻¹)
2009	10.04	8.599530806
2010	11.12	9.678381931
2011	11.8	10.37348952
2012	11.72	10.28151126
2013	8.32	6.882534033
2014	9.16	7.725526294
2015	10.96	9.534380174
2016	10.92	9.493170806
2017	10.66	9.223623062
2018	10.94	9.516195554
2019	10.88	9.455557186
2020	9.84	8.420207277
2021	10.04	8.60381114
2022	11.12	9.694006483

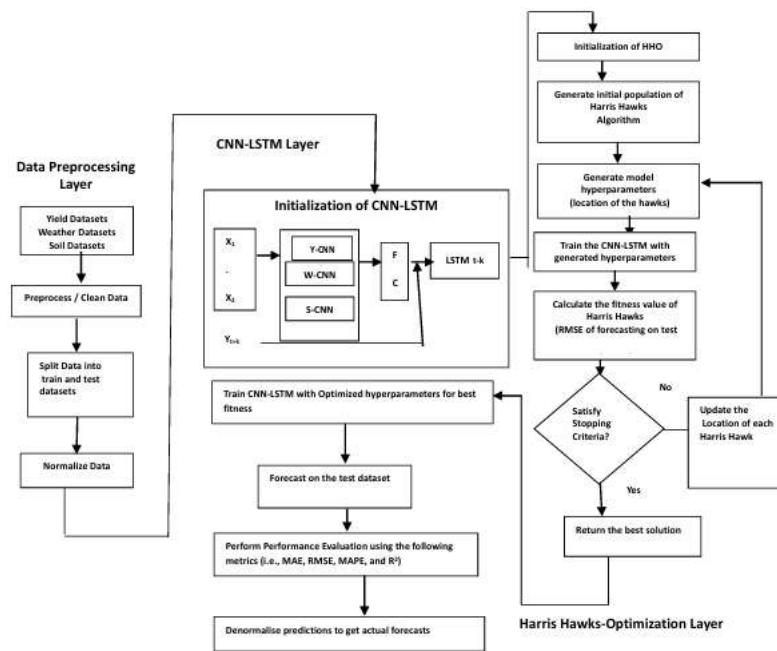


Figure 3. Schematic Diagram of the HHO-CNN-LSTM Cassava Yield Prediction System

Table 6. Results of Evaluation Metrics for CNN-LSTM

Model	MAE	MSE	RMSE	MAPE	R ²
CNN-LSTM	1.431207	2.050261	1.431873	13.84528	-0.56726

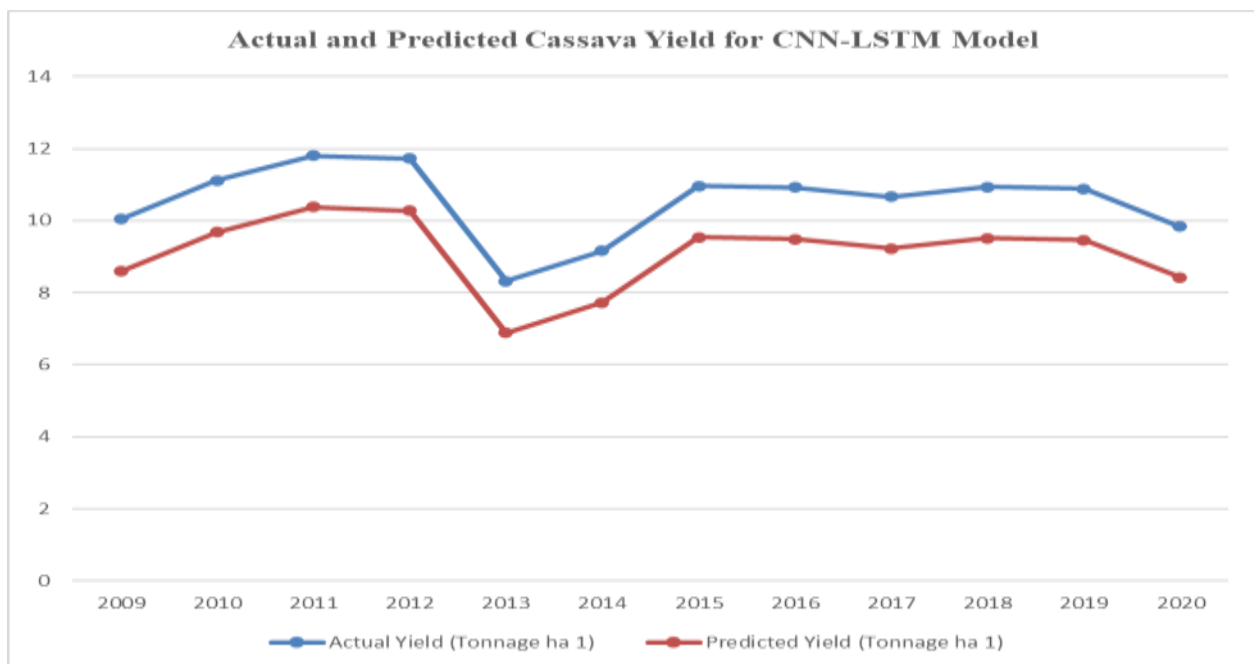


Figure 4. Graphical Representation of Actual and Predicted Cassava Yield of CNN-LSTM from 2009 to 2022

Outcomes of Harris Hawk Optimisation for Convolutional Neural Network and Long-Short-Time Memory

The soil, weather, and prior yield datasets were used sequentially to train the HHO-CNN-LSTM model. The CNN's processed data is passed to the LSTM for further

processing. A fully connected layer came next in the output. The CNN-LSTM model has 50 nodes in each of its four LSTMs. Overfitting was prevented with the use of 0.2 dropout regularisation. The CNN-LSTM hyperparameters were optimised using Harris Hawk Optimisation (HHO) over 100 epochs in 14 batches. A maximum of 900

iterations, or nine iterations per epoch, is allowed. Furthermore, the model learning process employed the ReLu activation function at a rate of 0.002048. The output is shown in Tables 7 and 8, respectively, because the model was trained on a 14-year dataset.

However, the data in Table 7 show that the actual and projected yield values are separated by an interval of

0.87 to 0.92. The HHO-CNN-LSTM model yielded mean absolute error, mean square error, root mean square error, mean absolute percentage error, and R-squared performance assessment metrics of 0.904463, 0.82379, 0.907629, 8.745989, and 0.370377, respectively, as shown in Table 8. Figure 5 depicts the actual and anticipated cassava yields from 2009 to 2022, using the HHO-CNN-LSTM model.

Table 7. Results of Actual and Predicted Cassava Yield HHO-CNN-LSTM from 2009 to 2022

Year	Actual Yield (Tonnage ha ⁻¹)	Predicted Yield (Tonnage ha ⁻¹)
2009	10.04	9.13076
2010	11.12	10.2152
2011	11.8	10.9025
2012	11.72	10.8179
2013	8.32	7.42024
2014	9.16	8.2912
2015	10.96	10.0475
2016	10.92	10.0213
2017	10.66	9.76581
2018	10.94	10.0294
2019	10.88	9.98071
2020	9.84	8.93435
2021	10.04	9.1355
2022	11.12	10.1987

Table 8. Results of Evaluation Metrics HHO-CNN-LSTM

Model	MAE	MSE	RMSE	MAPE	R ²
HHO-CNN-LSTM	0.904463	0.82379	0.907629	8.745989	0.370277

DISCUSSION OF RESULTS

The comparative results presented in Tables 6 and 8 indicate that the HHO-CNN-LSTM model consistently outperformed the baseline CNN-LSTM model in all evaluation metrics, such as MAE, MSE, RMSE, MAPE, and R-Square. Specifically, the HHO-CNN-LSTM model achieved improvements of 0.5, 1.2, 0.5, 5.1, and 0.9, respectively, demonstrating superior predictive accuracy and robustness in cassava yield estimation.

One major reason for this improvement is the effective hyperparameter optimisation provided by the Harris Hawks Optimisation (HHO) algorithm. CNN-LSTM models are highly sensitive to hyperparameters such as learning rate, number of convolutional filters, kernel size, and number of LSTM units. Improper tuning often leads to

suboptimal convergence and reduced generalisation performance. By employing HHO, the search for optimal hyperparameter combinations is guided by adaptive exploration and exploitation strategies, enabling the model to escape poor local solutions and achieve better global optima (Heidari *et al.*, 2019; Yang, 2010). This optimisation directly contributes to lower MAE, MSE, and RMSE values.

Secondly, the reduction in MAPE suggests that the HHO-CNN-LSTM model is more effective in handling variability and uncertainty inherent in agricultural datasets. Cassava yield data are often noisy due to climatic fluctuations, soil heterogeneity, and management practices. Metaheuristic-based optimisation methods such as HHO are well-suited for such complex,

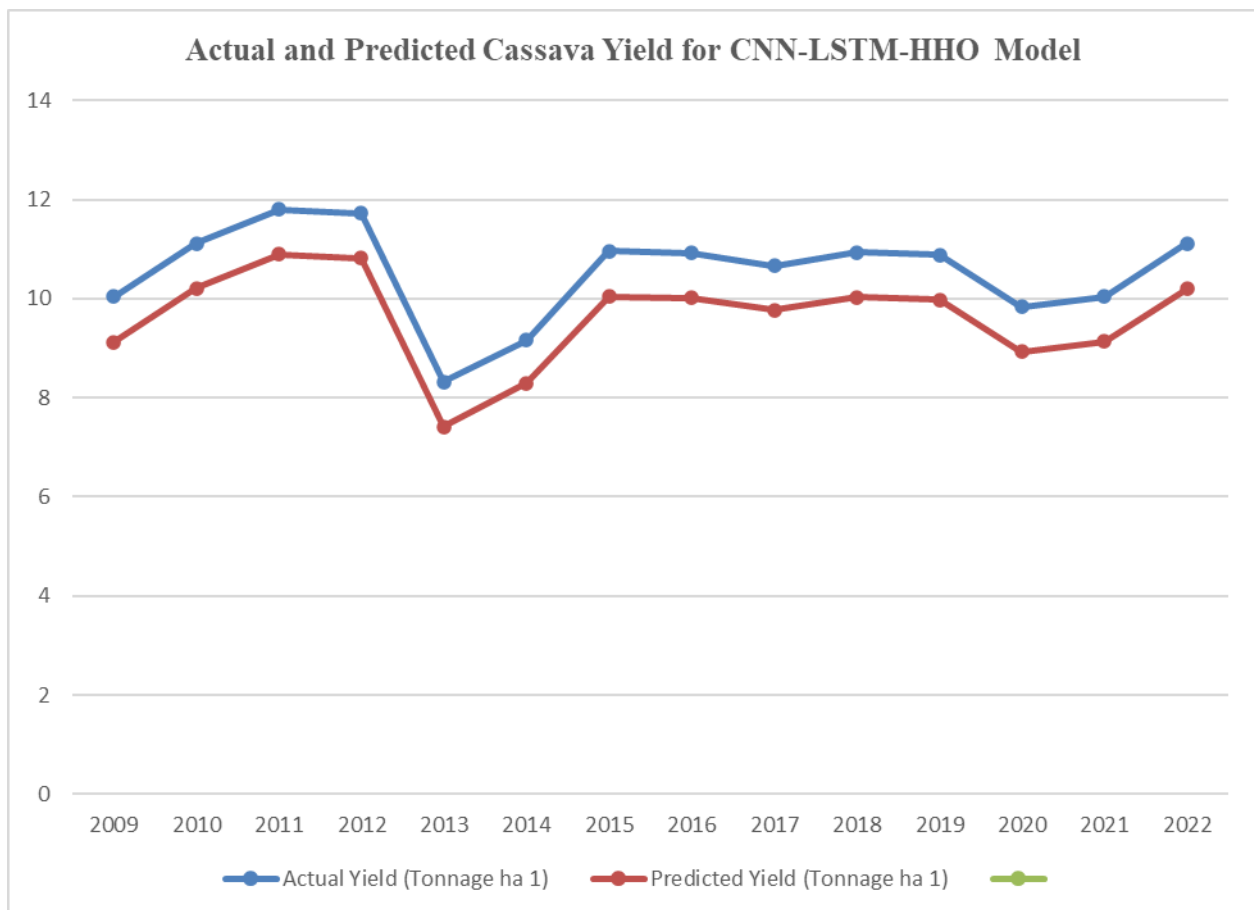


Figure 5. Graphical Representation of Actual and Predicted Cassava Yield of HHO-CNN- LSTM from 2009 to 2022

nonlinear problems because they do not rely on gradient information and can adaptively balance exploration and exploitation during training (Eiben and Smith, 2015). As a result, the optimised model achieves more reliable percentage-based error estimates.

Furthermore, the increase in R-Square value indicates improved explanatory power of the HHO-CNN-LSTM model. This implies that the optimised model captures a greater proportion of the variance in cassava yield data. The synergy between CNN and LSTM enables effective extraction of spatial features and temporal dependencies, while HHO fine-tunes the architecture to maximise learning efficiency (LeCun *et al.*, 2015; Hochreiter and Schmidhuber, 1997). This enhanced feature representation and temporal modelling explain the improved goodness-of-fit observed. Overall, the results show that combining HHO and CNN-LSTM improves predictive performance significantly by improving hyperparameter selection, reducing overfitting, and increasing model stability. This demonstrates the suitability of hybrid metaheuristic-deep learning models for precision agriculture applications, especially yield prediction tasks with complex and nonlinear data patterns.

CONCLUSION AND FUTURE WORK

This study presented a novel hybrid cassava yield prediction framework that combines Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and Harris Hawk Optimisation (HHO). By combining CNN's spatial feature extraction capability, LSTM's temporal dependency learning strength, and HHO's adaptive hyperparameter optimisation mechanism, the proposed HHO-CNN-LSTM model effectively addresses the complexity and nonlinearity inherent in agricultural yield prediction. To ensure the experimental results' reliability and robustness, primary soil from SoilGrids250m version 2.0, meteorological datasets from the Nigerian Meteorological Agency (NIMET), and cassava yield data from the Food and Agriculture Organisation Statistics were preprocessed through noise removal, normalisation, and rigorous validation using a 5-fold cross-validation approach. Using MAE, MSE, RMSE, MAPE, and R-square performance evaluation metrics, the optimised HHO-CNN-LSTM model consistently outperformed the conventional CNN-LSTM model. HHO-based hyperparameter tuning improves predictive accuracy, reduces error margins, and improves model generalisation. These findings show the effectiveness of metaheuristic-driven deep learning optimisation for crop yield prediction, particularly in the Nigerian agricultural context. Overall, the study contributes a robust and scalable decision-support

tool for precision agriculture, offering valuable insights for farmers, policymakers, and agricultural stakeholders in terms of planning, risk mitigation, and food security improvement.

Despite the promising findings of this study, several directions for future research are recommended. Future studies may incorporate remote sensing data, such as satellite-derived vegetation indices (e.g., NDVI, EVI), to improve spatial representation and yield prediction accuracy at finer resolutions. Also, future research may investigate the use of other advanced metaheuristic optimisation algorithms, such as Whale Optimisation Algorithm (WOA), Grey Wolf Optimiser (GWO), or hybrid optimisation strategies, to further evaluate and improve hyperparameter optimisation efficiency and convergence speed. Furthermore, extending the proposed HHO-CNN-LSTM framework to other crops and agro-ecological zones would help validate its generalisability and adaptability beyond cassava production in Nigeria. Finally, future research could focus on real-time yield forecasting systems that integrate the model into web-based or mobile platforms, allowing farmers and agricultural agencies to make timely decisions.

REFERENCES

1. Al-Adhaileh, M. H., and Aldhyani, T.H.H. (2022). Artificial intelligence framework for modeling and predicting crop yield to enhance food security in Saudi Arabia. *PeerJ Computer Science*, 8: 1-24. <https://pmc.ncbi.nlm.nih.gov/articles/PMC9575863/pdf/peerj-cs-08-1104.pdf>
2. Ali, T., Rehman, S.U., Ali, S., Mahmood, K., Obregon, S.A., Iglesias, R.C., Khurshaid, T. and Ashraf, I. (2024). Smart agriculture: utilizing machine learning and deep learning for drought stress identification in crops. *Scientific Reports*, 14(30062): 1-16.
3. Alibabaei, K., Gaspar, P.D. and Lima, T.M. (2021). Crop yield estimation using deep learning based on climate big data and irrigation scheduling. *Energies*, 14(11): 1-21. <https://www.mdpi.com/1996-1073/14/11/3004/pdf>
4. Ansarifar, J., Wang, L. and Archontoulis, S. V. (2021). An interaction regression model for crop yield prediction. *Scientific Reports*, 11(1): 1-14. <https://www.nature.com/articles/s41598-021-97221-7.pdf>
5. Chang, Y., Latham, J., Licht, M. and Wang, L. (2023). A data-driven crop model for maize yield prediction. *Communication Biology*, 6: 1-9. https://pmc.ncbi.nlm.nih.gov/articles/PMC10121691/pdf/42003_2023_Article_4833.pdf
6. Challa, S. K., Kumar, A. and Semwal, V. B. (2021). A multibranch CNN-BiLSTM model for human activity recognition using wearable sensor data. *The Visual Computer*, 38(12): 1-15.
7. Choubey, M., Gupta, Y.K., Dubey, A. and Prasad, R. (2025). Deep learning for precision agriculture: Predicting crop yields in a changing climate. In book: *AgricTech Revolution*, pp. 253-275. https://www.researchgate.net/publication/396927034_Deep_Learning_for_Precision_Agriculture_Predicting_Crop_Yields_in_a_Changing_Climate
8. Edamisan, S. I. and Kemisola, A. (2020). Cassava production in Nigeria: trends, instability and decomposition analysis (1970-2018). *Heliyon*, 6 (10): 1-10.
9. Eiben, A. E. and Smith, J. E. (2015). Introduction to evolutionary computing (2nd ed.). Springer. Pp. 1-294.
10. FAO/STAT, F. (2019). Food and agriculture data.
11. Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M. and Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems*, 97: 849-872.
12. Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8): 1735-1780. <https://www.bioinf.jku.at/publications/older/2604.pdf>
13. Javaregowda, M., Indiramma, R. and Nagaraju, N. (2025). M-Bi-GRU-CNN: a hybrid deep learning model with optimized feature selection for enhanced crop yield prediction. *Multimedia Tools and Applications*, 84(32): 39787-39811.
14. Kamilaris, A. and Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147(1): 70-90.
15. Khaki, S., Wang, L. and Archontoulis, S.V. (2020). A CNN-RNN framework for crop yield prediction. *Frontiers in Plant Science*, 10(1750): 1-14. <https://public-pages-files-2025.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2019.01750/pdf>
16. Khaki, S., and Wang, L. (2019). Crop yield prediction using deep neural networks. *Frontiers in Plant Science*. 10(621): 1-10. <https://doi.org/10.3389/fpls.2019.00621>
17. LeCun, Y., Bengio, Y. and Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.
18. Probert, M. (2012). Engineering optimisation: An introduction with metaheuristic applications, by Xin-She Yang, *Contemporary Physics*, 53(3): 271-272.
19. Liakos, K. G., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine learning in

agriculture: A review. *Sensors*, 18(8): 1-29.
<https://www.mdpi.com/1424-8220/18/8/2674/pdf>

20. Srivastava, A. K., Ewert, F., Akinwumiju, A. S., Zeng, W., Ceglar, A., Ezui, K. S., Adelodun, A., Adebayo, A., Sobamowo, J., Singh, M., Rahimi, J. and Gaiser, T. (2023). Cassava yield gap-A model-based assessment in Nigeria. *Frontiers in Sustainable Food Systems*, 6: 1-20. <https://public-pages-files-2025.frontiersin.org/journals/sustainable-food-systems/articles/10.3389/fsufs.2022.1058775/pdf>
21. Taremwa, D., Ahishakiye, E., Obbo, A., Kisozi, P.K. and Kaggwa, F. (2025). Prediction of maize yield in Uganda using CNNLSTM architecture on a multimodal climate and remote sensing dataset. *Research Square*, 1-32.
22. Wang, H.H., Wang, Y.C., Wee, B.L., Yan, J.K., Yahya, F.B. (2024). Optimizing crop yield prediction: A hybrid approach integrating CNN and LSTM networks. *Journal of Theoretical and Applied Information Technology*, 102(22): 8075-8083.
23. Wang, J., Jiang, W., Li, Z. and Lu, Y. (2021). A new multi-scale sliding window LSTM framework (MSSW-LSTM): A case study for GNSS time-series prediction. *Remote Sensing*, 13(16): 1-16. <http://dx.doi.org/10.3390/rs13163328>
24. Xie, H., Zhang, L. and Lim, C. P. (2020). Evolving CNN-LSTM models for time series prediction using enhanced grey wolf optimizer. <https://core.ac.uk/download/pdf/334954582.pdf>
25. Yang, X. S. (2010). Nature-inspired metaheuristic algorithms. Luniver Press.
26. Zhou, W., Song, C., Liu, C., Fu, Q., An, T., Wang, Y., Sun, X., Wen, N., Tang, H. and Wang, Q. (2023). A prediction model of maize field yield based on the fusion of multitemporal and multimodal UAV data: A case study in Northeast China. *Remote Sensing*, 15(3483): 1-22.