Genetic Algorithm Optimized CNN-LSTM architecture for forecasting volatility of Indian edible oils prices

Achal Lama[1*], KN Singh[1] and Bishal Gurung[2]

[1]ICAR-Indian Agricultural Statistics Research institute, New Delhi-110012, India

[2]North-Eastern Hill University, Shillong- 793022, India

***Presenting author: achallama.iasri@icar.org.in**

Abstract

Volatility modelling is an important area of time series analysis. Agricultural commodity prices tend to be more volatile due to seasonality, inelastic demand, production uncertainty and also because many agricultural commodities are perishable. An increase in price volatility implies higher uncertainty about future prices, a fact that can affect producer's welfare. Therefore, understanding the nature of agricultural commodity price volatility and the ability to accurately forecast the price volatility are important concerns among both policy makers and farming community. Price volatility provides a measure of the possible variation or movement in the price variable. Wide price movements over a short period of time are considered as high volatility. The importance of volatility has led to the development and applications of many significant time series models. In this regard, GARCH (Generalized Autoregressive Conditional Heteroscedasticity) model after its introduction has been widely used owning to its applicability in variety of domains. In recent years, the machine learning technique is introduced for developing learning models and is used to forecast time series with deep learning algorithms. Deep learning algorithms such as Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN) have performed satisfactorily in many disciplines of science, especially in the financial sectors. Deep learning mechanism is used to visualize the pattern and structure of the series, mainly complexity and non-linearity by extracting hidden layers from the target network in time series forecasting. Thus, this study focuses on using these well-established deep learning models to efficiently model and forecast volatile price series of edible oils in India namely rapeseed and mustard, soyabean and sunflower. The datasets were collected from April 1982 to March 2025 from https://eaindustry.nic.in/. As it is a well-known fact that efficiency of these models largely depends upon optimal hyperparameter settings along with appropriate input features. In our case the input features refer to the ideal number of lags. Hence, we approach this problem in two phase manner, first by selecting ideal number of lags using random forest algorithm and then optimizing the hyperparameters of the CNN-LSTM model using Genetic Algorithm (GA). We then implement LSTM, CNN, CNN-LSTM and optimized CNN-LSTM to our real datasets. We document superior results for the proposed optimized CNN-LSTM model. The gain in efficiency ranged between 5-10 (%) in case of RMSE and 7-13 (%) for MAPE, when compared to conventional CNN-LSTM model. This study highlights the importance of hyperparameter optimization and ideal lag selection for improving the efficiency of deep learning models. This study not only provides an effective methodology for forecasting volatile agricultural prices but also presents a framework that can be generalized to other similar time series datasets. Moreover, the findings contribute toward achieving Sustainable Development Goal (SDG) 2.c, which aims to limit extreme food price volatility, thereby aiding policymakers in developing informed and resilient agricultural policies.

Keywords: Volatility, Edible oils, Random forest, LSTM, CNN, CNN-LSTM, SDG 2.c

1. Introduction

Time series forecasting plays a pivotal role in understanding dynamic systems by extracting meaningful patterns from historical observations to predict future behaviour. It has become an indispensable analytical tool across disciplines such as finance, economics, climatology, and agriculture. Unlike cross-sectional data, time series data exhibit temporal dependencies, structural breaks, seasonality, and stochastic volatility, making their modelling both challenging and scientifically significant. In particular, volatility modelling represents a crucial branch of time series analysis, as it captures the magnitude of fluctuations and uncertainty inherent in sequential data. Accurate modelling of volatility is essential not only for prediction but also for risk management, strategic planning, and policy formulation.

Agricultural commodity prices are inherently volatile due to multiple structural and market-driven factors. Seasonality in production cycles, climatic variability, inelastic demand for staple commodities, global trade linkages, and speculative activities contribute to frequent and sometimes abrupt price fluctuations. Additionally, the perishable nature of many agricultural commodities intensifies market instability. High price volatility implies increased uncertainty regarding future returns, which directly affects farmers' income stability, investment decisions, and market participation. For policymakers, unpredictable price movements complicate procurement planning, food security strategies, and inflation control measures. Therefore, understanding and forecasting agricultural price volatility is essential for safeguarding producer welfare and ensuring stable food systems.

Among agricultural commodities, oilseed crops hold strategic economic and nutritional importance, particularly in developing economies such as India. Edible oils derived from rapeseed and mustard, soybean, and sunflower constitute a major component of household consumption and industrial use. India's heavy dependence on both domestic production and imports further exposes oilseed prices to global market shocks, exchange rate fluctuations, and geopolitical disruptions. Consequently, volatility in oilseed prices has far-reaching implications for farmers, consumers, processors, and government agencies. Accurate forecasting of oilseed price volatility is therefore critical for designing procurement policies, buffer stock management strategies, trade interventions, and price stabilization mechanisms. Given their economic significance, developing reliable predictive frameworks for oilseed price dynamics remains an urgent research priority.

Traditional approaches to modelling price volatility have largely relied on statistical time series techniques. The introduction of the Autoregressive Conditional Heteroscedasticity (ARCH) model and its generalized form, the Generalized Autoregressive Conditional Heteroscedasticity (GARCH) model, marked a significant advancement in capturing time-varying variance in economic and financial data. GARCH-type models have been widely applied to agricultural commodity markets, including oilseeds, to model clustering effects and persistence in volatility. Although these models provide a strong theoretical framework for volatility estimation, their linear assumptions and parametric structure limit their ability to fully capture complex nonlinear patterns, long-term dependencies, and structural shifts frequently observed in agricultural price series.

In recent years, machine learning and deep learning approaches have emerged as powerful alternatives for time series forecasting. Unlike traditional statistical models, machine learning algorithms are data-driven and capable of modelling nonlinear relationships without strict distributional assumptions. Deep learning architectures, in particular, have demonstrated remarkable performance in handling high-dimensional and sequential data. Recurrent Neural Networks (RNNs) are specifically designed to process sequential information, while Long Short-Term Memory (LSTM) networks effectively capture long-term dependencies by addressing the vanishing gradient problem. Convolutional Neural Networks (CNNs), although originally developed for image processing, have shown strong capability in extracting local patterns and features from time series data. In agricultural and financial forecasting contexts, these models have been increasingly adopted for price prediction, yield forecasting, and market trend analysis, often outperforming conventional statistical approaches.

Hybrid deep learning frameworks that combine multiple architectures have gained attention for their ability to integrate complementary strengths. For instance, CNN layers can extract relevant features and local temporal structures, while LSTM layers capture long-term dependencies in sequential data. Such hybrid CNN-LSTM models have shown improved predictive performance in complex time series applications. However, the efficiency of deep learning models largely depends on appropriate hyperparameter tuning and optimal input feature selection. In agricultural price forecasting studies, limited attention has been given to systematic lag selection and evolutionary optimization techniques for enhancing model robustness and generalizability. Furthermore, despite growing applications of deep learning in crop price prediction, comprehensive studies focusing specifically on long-term oilseed price volatility modelling using optimized hybrid architectures remain relatively scarce.

Several research gaps therefore persist in the existing literature. First, many studies either rely solely on traditional volatility models or apply deep learning methods without rigorous feature engineering and hyperparameter optimization. Second, the integration of ensemble feature selection methods, such as Random Forest-based lag identification, with evolutionary optimization techniques like Genetic Algorithm (GA) for deep learning tuning remains underexplored in agricultural commodity forecasting. Third, long historical datasets spanning multiple decades, which capture structural changes and evolving market dynamics, are rarely leveraged to their full analytical potential. Addressing these gaps is essential for improving forecast accuracy and enhancing the practical applicability of volatility models.

In this context, the present study aims to develop and evaluate an optimized hybrid deep learning framework for forecasting volatile price series of major edible oilseeds in India, namely rapeseed and mustard, soybean, and sunflower. The study adopts a two-phase methodology involving optimal lag selection using Random Forest algorithms and hyperparameter optimization of a CNN-LSTM model using a GA. The performance of LSTM, CNN, CNN-LSTM, and optimized CNN-LSTM models is systematically compared.

2. Literature Review

Earlier, forecasting approaches specifically tailored to agricultural commodity prices and their volatility were developed, particularly for crops such as rice, wheat, maize, soybean, palm oil, and cotton, with methods ranging from traditional econometric models to advanced machine learning and deep learning techniques. Early modelling efforts for staple crops like rice and wheat employed Autoregressive Integrated Moving Average (ARIMA) and seasonal ARIMA (SARIMA) models to capture inherent seasonality and autocorrelation in price series, often demonstrating reasonable short-term prediction performance but limited capacity to model nonlinear dynamics or volatility clustering[1]. To more explicitly model volatility in commodity returns for crops such as soybean and palm oil, researchers have applied ARCH and GARCH family models, including EGARCH and TGARCH variants, which estimate time-varying variance and volatility persistence, capturing clustering effects in oilseed and grain markets[2,3]. Despite their theoretical appeal, these statistical volatility models often assume linear relationships and specific error distributions that may not fully characterize the complex, nonlinear price behaviour resulting from seasonality, supply shocks, and policy interventions. In response to these limitations, researchers have increasingly explored machine learning approaches applied to agricultural price forecasting; SVR and Random Forest models have been employed for rice, maize, and cotton price prediction, showing improved performance over linear models by capturing nonlinear associations among price history and exogenous market variables. Nonetheless, traditional ML methods are generally better suited for short-term forecasts and do not inherently model temporal dependencies[4]. Recognizing this, deep learning models such as RNN and their gated variants, specifically LSTM networks, have been widely adopted for price forecasting of volatile agricultural commodities like soybean and palm oil5,6. LSTM models have shown superior ability to learn long-term dependencies and dynamic patterns in sequential price data, yielding lower forecasting errors compared to ARIMA and SVR in several crop price studies. CNNs have also been integrated into forecasting frameworks to extract local temporal features from price series, and hybrid CNN-LSTM architectures have been developed for crops including wheat and rice, demonstrating enhanced prediction accuracy by combining local feature extraction with sequence modelling[7,8]. Some studies have combined traditional volatility models with deep learning, developing hybrid frameworks such as LSTM-GARCH for improved volatility forecasting in agricultural markets, where the GARCH component models conditional variance and the LSTM captures sequential nonlinear dependencies[9,10,11]. Although these advanced models perform better than purely statistical approaches, challenges persist in selecting optimal input lags, handling noise and structural breaks in historical price series, and integrating exogenous drivers such as weather, macroeconomic variables, and policy effects.

3. Methodology

In this study, two major deep learning architectures as well as their hybrid have been used for volatile oilseed price prediction. Lag selection were done using random forest method. For the hybrid CNN+LSTM model, GA is used during parameter optimization. A brief description of all model architectures is presented below.

3.1 One dimensional convolutional neural network (1D-CNN)

A One-Dimensional Convolutional Neural Network (1D-CNN) is a deep learning architecture designed for sequential data modelling, widely applied in time series forecasting. Given an input sequence {y_{t-n}, … , y_t}, the objective is to learn a nonlinear mapping to predict y_{t+h}. After preprocessing (normalization, missing value imputation, and train–test split), the series is transformed into sliding windows (lagged vectors) suitable for supervised learning. In a convolutional layer l, the forward propagation for the k-th feature map is defined as:

$$x_k{}^{(l)} = b_k{}^{(l)} + \sum_{i=1}^{N_{l-1}} conv1D(w_{ik}{}^{(l-1)}, s_i{}^{(l-1)}) \tag{1}$$

where $w_{ik}{}^{(l-1)}$ denotes the convolution kernel, $s_i{}^{(l-1)}$ is the previous layer output, $b_k{}^{(l)}$ is the bias term, and N_{l-1} is the number of input channels. The convolution operation performs discrete linear filtering to extract local temporal features (trend segments, short-term dependencies, structural shifts). The intermediate output is passed through a nonlinear activation function:

$$y_k{}^{(l)} = f\left(x_k{}^{(l)}\right) \tag{2}$$

followed by a down-sampling (pooling) operation:

$$s_k{}^{(l)} = y_k{}^{(l)} \# ss \tag{3}$$

where ss is the pooling stride. Pooling reduces dimensionality and enhances translation invariance.

After multiple convolution–pooling blocks, feature maps are flattened and fed into fully connected (MLP) layers for high-level representation learning and regression output. The loss function for an input p, using Mean Squared Error (MSE), is:

$$E_p = \sum_{i=1}^{N_L} ({y_i}^{(L)} - {t_i}^p)^2 \tag{4}$$

where ${y_i}^{(L)}$ and ${t_i}^p$ represent predicted and target outputs, respectively. Gradient-based backpropagation updates parameters via optimization algorithms (e.g., SGD, Adam). The primary advantage of 1D-CNN lies in its parameter sharing, sparse connectivity, and parallelizable convolution operations, enabling efficient extraction of hierarchical temporal features with lower computational complexity compared to fully connected architectures.

3.2 Long short term memory (LSTM)

LSTM is a gated recurrent neural network architecture designed to overcome the vanishing gradient problem in standard RNNs. Conventional RNNs suffer from exponential gradient decay during backpropagation through time (BPTT), limiting their ability to capture long-range temporal dependencies. LSTM resolves this limitation using a memory cell c_t and multiplicative gating mechanisms that regulate information retention, update, and output propagation across time steps.

The forget gate controls retention of prior cell state information:

$$f_t = \sigma(W_{fh} h_{t-1} + W_{fx} x_t + b_f) \tag{5}$$

The input gate regulates incorporation of new information:

$$i_t = \sigma(W_{ih} h_{t-1} + W_{ix} x_t + b_i) \tag{6}$$

The candidate cell state is computed as:

$$\tilde{c}_t = \tanh(W_{ch} h_{t-1} + W_{cx} x_t + b_c) \tag{7}$$

The updated cell state is:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{8}$$

The output gate determines hidden state exposure:

$$o_t = \sigma(W_{oh} h_{t-1} + W_{ox} x_t + b_o) \tag{9}$$

The hidden state is given by:

$$h_t = o_t \odot \tanh(c_t) \tag{10}$$

Here, x_t denotes input, h_t hidden state, W_* weight matrices, b_* bias vectors, and $\odot$ element-wise multiplication. The gated additive update structure enables stable gradient propagation and efficient modelling of long-term dependencies in nonlinear time series.

3.3 Hybrid of 1D-CNN and LSTM

The hybrid 1D-CNN+LSTM architecture integrates convolutional feature extraction with recurrent sequence modeling for enhanced time series forecasting. The 1D-CNN component performs local receptive field-based convolution to extract short-term temporal patterns, structural breaks, and high-frequency features through parameter sharing and sparse connectivity. The resulting feature maps are down-sampled via pooling to reduce dimensionality and suppress noise. These extracted representations are then fed into the LSTM layer, which models long-range temporal dependencies using gated memory cells and controlled gradient flow. Formally, the CNN transforms the input sequence $X \in \mathbb{R}^{T \times d}$into feature representations F, which are sequentially processed by LSTM to compute hidden states h_t via gated updates (f_t,i_t,o_t,c_t). The final dense layer performs regression to generate forecasts. This hybrid structure leverages CNN's hierarchical feature learning and LSTM's long-term dependency modeling, improving predictive robustness in nonlinear and nonstationary time series.

3.4 GA Optimised 1D-CNN+LSTM

In the GA-optimized 1D-CNN+LSTM framework, model hyperparameters are tuned using a Genetic Algorithm (GA) to enhance convergence efficiency and predictive accuracy. GA performs population-based stochastic optimization by encoding hyperparameters, such as number of convolutional filters, kernel size, LSTM units, learning rate, batch size, dropout rate, and optimizer parameters, into chromosomes. An initial population is randomly generated, and fitness is evaluated using a predefined objective function (e.g., validation RMSE or MAPE). Selection operators (e.g., tournament or roulette-wheel selection) identify high-performing individuals, while crossover and mutation operators ensure exploration and exploitation of the search space. The evolutionary process iteratively updates parameter combinations across generations until convergence criteria are satisfied. This global optimization strategy mitigates manual tuning bias, avoids local minima, and improves generalization performance of the hybrid 1D-CNN+LSTM model in nonlinear and high-volatility time series forecasting.

3.5 Evaluation Metrics

Model performances were evaluated using four standard regression metrics: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination (R^2). The equations of these metrics are given below:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2} \tag{11}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y_i}| \tag{12}$$

(13)

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{14}$$

where y_i denotes observed values, $\hat{y}_i$ predicted values, $\bar{y}$ sample mean, and n the number of observations. Lower RMSE, MAE, and MAPE and higher R^2 indicate superior predictive performance.

4. Results and Discussion

4.1 Data Description

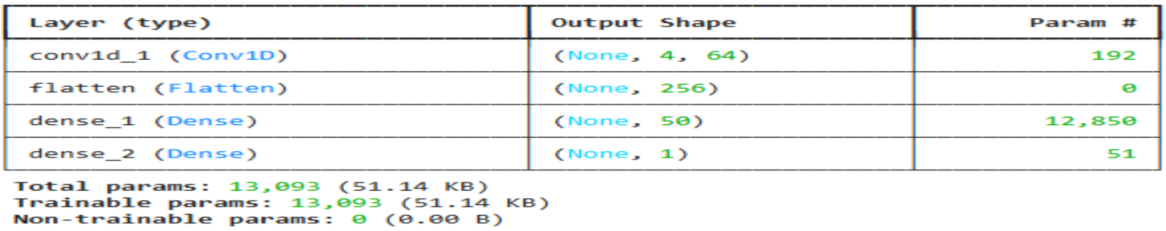
The present dataset comprises monthly edible oil price data for rapeseed-mustard, sunflower, and soybean oils collected from April, 1982 to March, 2025. This dataset provides a comprehensive understanding of price dynamics and market behaviour of major edible oils over the study period. The datasets are collected from https://eaindustry.nic.in/.

4.2 Experimental Setup

A total of 516 monthly data points for edible oil prices of rapeseed-mustard, sunflower, and soybean were utilized in this study, covering the period from April 1982 to March 2025. The dataset was divided into three subsets: training (360 observations), validation (108 observations), and testing (48 observations). Lag or window size selection was performed using the Random Forest method to identify the most informative temporal dependencies in the series. To efficiently handle the time series data, a window-based approach was adopted in which the input data were transformed into overlapping sequences of selected lag length. This approach enabled the models to capture both short-term price fluctuations and long-term temporal patterns inherent in the edible oil price series.

Four forecasting models: 1D Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), hybrid CNN+LSTM, and Genetic Algorithm (GA) optimized CNN+LSTM, were trained using the training dataset and evaluated using the validation dataset. Hyperparameter tuning and parameter optimization were performed using the Grid Search method. The models were trained for 100 epochs with a batch size of 16 to ensure stable learning and convergence during the training process.

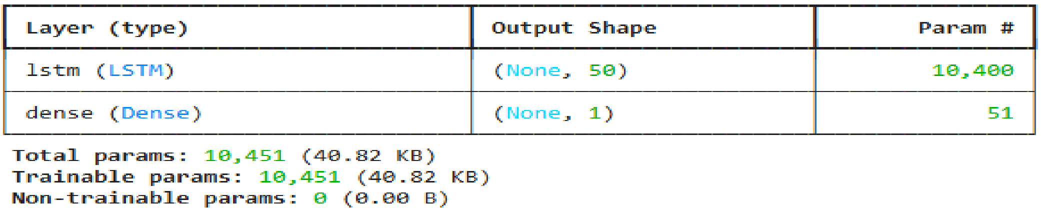
The 1D-CNN model (Fig. 1) was constructed using a Sequential architecture comprising a one-dimensional convolutional layer with 64 filters and kernel size 2 with ReLU activation to extract local temporal features from the input sequence. The extracted features were flattened and passed through a dense layer of 50 neurons with ReLU activation, followed by a single output neuron. The model was compiled using the Adam optimizer and mean squared error loss function.



```
Layer (type)              Output Shape              Param #
conv1d_1 (Conv1D)         (None, 4, 64)                 192
flatten (Flatten)         (None, 256)                     0
dense_1 (Dense)           (None, 50)                 12,850
dense_2 (Dense)           (None, 1)                      51
Total params: 13,093 (51.14 KB)
Trainable params: 13,093 (51.14 KB)
Non-trainable params: 0 (0.00 B)
```

Fig. 1. Architecture of 1D-CNN model

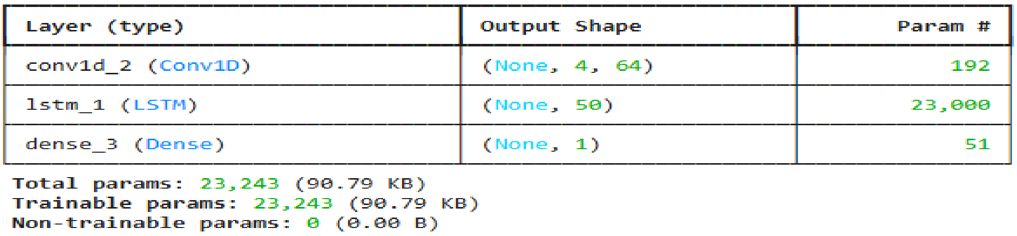
The LSTM model architecture (Fig. 2) consisted of a Sequential framework with a single LSTM layer containing 50 hidden units and ReLU activation, designed to process input sequences of shape (lag, 1). The LSTM layer captured temporal dependencies in the time series data. A Dense output layer with one neuron was added for regression prediction. The model was compiled using the Adam optimizer and Mean Squared Error (MSE) loss function for training.



```
Layer (type)                  Output Shape                 Param #
lstm (LSTM)                   (None, 50)                    10,400
dense (Dense)                 (None, 1)                         51
Total params: 10,451 (40.82 KB)
Trainable params: 10,451 (40.82 KB)
Non-trainable params: 0 (0.00 B)
```

Fig. 2. Architecture of LSTM model

CNN+LSTM hybrid architecture (Fig. 3) integrates convolutional and recurrent layers for sequential forecasting. A 1D convolutional layer with 64 filters and kernel size 2 extracts local temporal features, which are passed to an LSTM layer with 50 units to capture long-term dependencies. Finally, a dense layer produces a single output, optimized using Adam with MSE loss.



```
Layer (type)                  Output Shape                 Param #
conv1d_2 (Conv1D)             (None, 4, 64)                    192
lstm_1 (LSTM)                 (None, 50)                    23,000
dense_3 (Dense)               (None, 1)                         51
Total params: 23,243 (90.79 KB)
Trainable params: 23,243 (90.79 KB)
Non-trainable params: 0 (0.00 B)
```

Fig. 3. Architecture of CNN+LSTM hybrid model

The GA-optimized CNN+LSTM model was designed as a hybrid deep learning architecture integrating convolutional and recurrent layers for time series forecasting. The architecture begins with a 1D convolutional layer with ReLU activation and same padding to extract local temporal features, followed by a max-pooling layer (when applicable) to reduce dimensionality and retain dominant patterns. The extracted features are then passed to an LSTM layer with ReLU activation to capture long-term sequential dependencies. An optional dropout layer is included for regularization. A dense output layer produces the final prediction. Hyperparameters, including filters, kernel size, LSTM units, dropout rate, and learning rate, were optimized using a genetic algorithm. Optimum hyperparameters for all the three datasets are presented in Table 1.

Table 1. Optimum hyperparameters for GA optimised CNN+LSTM models for various edible oil price datasets

Hyperparameters	**Sunflower**	**Soyabean**	**Rapeseed and Mustard**
Filters	64	64	64
Kernel size	5	3	3
LSTM_units	32	32	50
Dropout	0.0	0.0	0.0
Learning rate	0.001	0.0005	0.001

4.3 Results for Sunflower price dataset

The comparative performance of CNN, LSTM, CNN+LSTM, and GA-optimized CNN+LSTM models for sunflower edible oil price forecasting is presented across training, validation, and testing datasets (Table 2). During training, all models exhibited high goodness-of-fit with R² values exceeding 0.99, indicating strong learning capability. The GA-optimized CNN+LSTM achieved the lowest RMSE (3.084) and MAE (2.316), along with the highest R² (0.992), demonstrating superior fitting accuracy. In the validation phase, the GA-optimized model maintained the best performance with RMSE of 5.896 and R² of 0.972, outperforming the standalone CNN (R² = 0.915), LSTM (R² = 0.943), and standard CNN+LSTM (R² = 0.960). On the unseen test dataset, the GA-optimized CNN+LSTM again produced the lowest RMSE (7.739), MAE (5.983), and highest R² (0.895), indicating better generalization capability. The train–validation loss curve (Fig. 4) further illustrates stable convergence after initial epochs, with both losses declining rapidly and remaining closely aligned, suggesting minimal overfitting and effective optimization.

Table 2. Performances of deep learning model architectures for the Sunflower price dataset based on training, validation and test datasets

SUNFLOWER	TRAIN				VALIDATION				TESTING			
	RMSE	MAE	R2	MAPE	RMSE	MAE	R2	MAPE	RMSE	MAE	R2	MAPE
CNN-LSTM GA	**3.084**	**2.316**	**0.992**	**4.774**	**5.896**	**4.580**	**0.972**	**4.774**	**7.739**	**5.983**	**0.895**	**3.911**
CNN-LSTM	3.414	2.612	0.990	5.243	6.825	4.582	0.960	5.243	8.505	6.893	0.874	4.507
LSTM	3.277	2.484	0.991	5.092	5.945	5.258	0.943	5.092	7.923	6.042	0.890	3.957
CNN	3.231	2.434	0.991	4.759	6.258	2.434	0.915	5.879	9.315	7.122	0.848	4.669

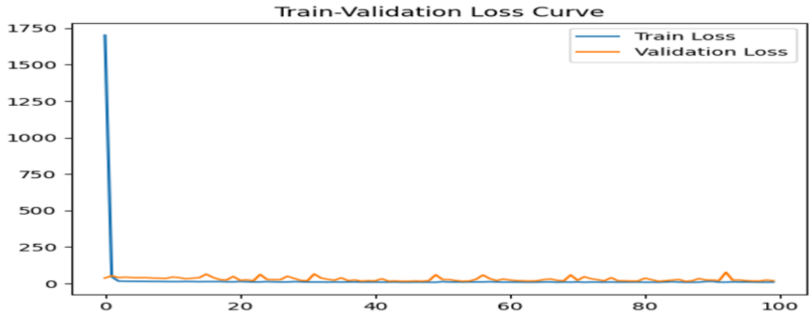


Fig. 4. Training-Validation loss curve for GA optimised CNN+LSTM model based on Sunflower dataset

The results clearly demonstrate that hybridization combined with evolutionary optimization enhances predictive performance for sunflower oil price forecasting. While individual CNN and LSTM models effectively captured spatial and temporal dependencies respectively, their standalone performance was comparatively lower on validation and testing datasets, indicating limited generalization. The CNN+LSTM hybrid improved performance by integrating local feature extraction with sequential memory learning, yet its parameters remained suboptimal without optimization. The incorporation of Genetic Algorithm tuning enabled systematic exploration of hyperparameters such as filter size, kernel size, LSTM units, dropout rate, and learning rate, leading to improved convergence and reduced prediction error. The consistently lower RMSE and higher R² of the GA-optimized model on unseen data suggest improved robustness and reduced variance. Moreover, the narrow gap between training and validation losses observed in the convergence plot confirms stable learning behavior without significant overfitting. The slight decline in R² on the test set (0.895) compared to training reflects inherent market volatility in sunflower oil prices during the later period, yet the model maintained strong explanatory power. Overall, the findings indicate that GA optimization significantly strengthens hybrid deep learning architectures for agricultural commodity price forecasting, particularly under non-linear and dynamic market conditions.

4.4 Results for Soyabean price dataset

For soybean edible oil price forecasting, all models demonstrated strong fitting ability on the training dataset, with R² values around 0.99, indicating effective learning of historical patterns. However, differences became evident during validation and testing. The GA optimized CNN+LSTM model achieved the most balanced generalization performance, recording a test RMSE of 18.047, MAE of 11.814, and the highest test R² of 0.837 (Table 3). The standard CNN+LSTM model showed slightly better validation R² (0.925) but marginally lower test R² (0.826). The standalone LSTM exhibited significant performance degradation on the test set (RMSE = 32.757, R² = 0.463), indicating instability. CNN performed moderately with a test R² of 0.738. The loss curve illustrates sharp initial convergence followed by stable and closely aligned training–validation losses (Fig. 5).

Table 3. Performances of deep learning model architectures for the Soyabean price dataset based on training, validation and test datasets

SOYABEAN	TRAIN				VALIDATION				TESTING			
	RMSE	MAE	R2	MAPE	RMSE	MAE	R2	MAPE	RMSE	MAE	R2	MAPE
CNN-LSTM GA	4.632	2.729	0.990	4.048	15.832	6.745	0.912	6.587	18.047	11.814	0.837	5.034
CNN-LSTM	4.919	2.822	0.989	3.986	12.035	8.658	0.925	5.268	18.627	11.103	0.826	4.645
LSTM	4.791	2.949	0.990	4.427	22.587	9.587	0.945	5.879	32.757	19.616	0.463	7.837
CNN	4.749	3.200	0.990	5.701	17.536	8.587	0.921	6.687	22.873	16.097	0.738	6.630

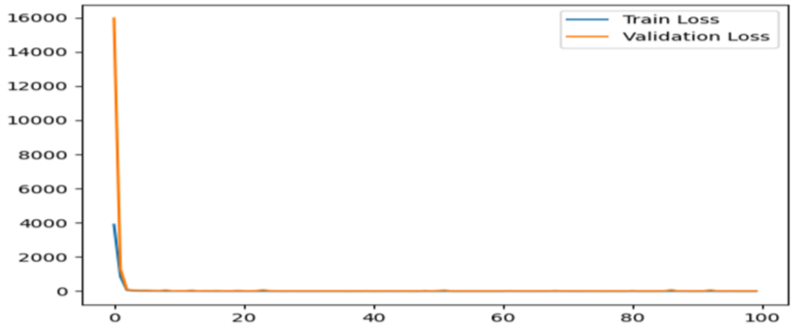


Fig. 5. Training-Validation loss curve for GA optimised CNN+LSTM model based on Soyabean dataset

The findings indicate that while all models captured underlying soybean price dynamics during training, generalization capability varied considerably. The sharp decline in LSTM test performance suggests sensitivity to market volatility and possible overfitting despite strong training accuracy. CNN alone captured short-term local patterns but lacked sufficient long-term memory to handle complex temporal fluctuations. The hybrid CNN+LSTM architecture improved predictive stability by integrating feature extraction with sequential learning. Further enhancement through Genetic Algorithm optimization enabled effective hyperparameter selection, reducing variance and improving robustness. The relatively stable loss convergence pattern supports the effectiveness of GA tuning in achieving improved out-of-sample forecasting under dynamic soybean market conditions.

4.5 Results for Rapeseed and Mustard price dataset

For rapeseed–mustard price forecasting, all models showed strong training performance with $R^2 \approx 0.99$, indicating effective learning of historical trends. On the validation set, CNN achieved the highest R^2 (0.956), followed by GA optimised CNN+LSTM (0.936), LSTM (0.910), and CNN+LSTM (0.897) (Table 4). However, on the unseen test dataset, GA optimised CNN+LSTM outperformed other models with the lowest RMSE (8.245), MAE (6.572), and highest R^2 (0.881). CNN+LSTM recorded a slightly lower test R^2 (0.865), while LSTM and CNN showed comparatively reduced generalization. The loss curve demonstrates rapid convergence in early epochs and stable alignment between training and validation losses thereafter (Fig. 6).

Table 4. Performances of deep learning model architectures for the Rapeseed and Mustard price dataset based on training, validation and test datasets

RPM	TRAIN				VALIDATION				TESTING			
	RMSE	MAE	R2	MAPE	RMSE	MAE	R2	MAPE	RMSE	MAE	R2	MAPE
CNN-LSTM GA	**3.249**	**2.480**	**0.991**	**5.041**	**6.856**	**4.647**	**0.936**	**4.269**	**8.245**	**6.572**	**0.881**	**4.272**
CNN-LSTM	3.447	2.663	0.990	5.352	5.687	4.569	0.897	5.145	8.793	7.132	0.865	4.680
LSTM	3.439	2.617	0.990	5.269	6.014	5.680	0.910	5.258	10.046	7.610	0.824	4.923
CNN	3.420	2.612	0.990	5.134	5.112	5.876	0.956	5.115	9.643	7.698	0.837	5.029

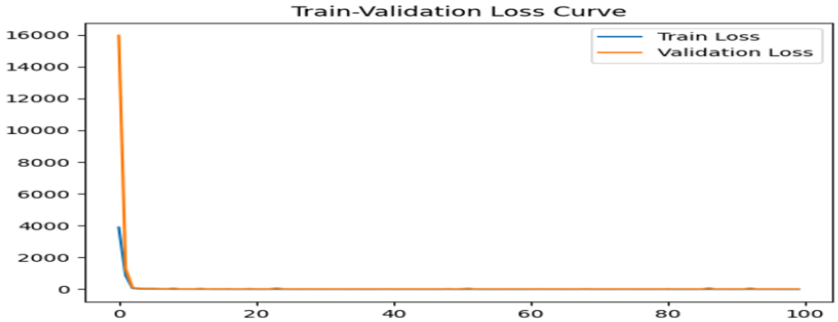


Fig. 6. Training-Validation loss curve for GA optimised CNN+LSTM model based on Rapeseed and Mustard dataset

The results indicate that although CNN performed strongly during validation, its generalization on the test set was inferior to GA optimised CNN+LSTM, suggesting sensitivity to unseen price volatility. The standalone LSTM captured temporal dependencies but exhibited comparatively higher prediction error, reflecting limitations in handling abrupt market fluctuations. The hybrid CNN+LSTM architecture improved stability by combining convolutional feature extraction with sequential memory learning. Further enhancement through Genetic Algorithm optimization refined hyperparameters, resulting in better out-of-sample robustness and lower forecasting error. The close overlap of training and validation loss curves supports stable learning without severe overfitting. Overall, GA optimised CNN+LSTM demonstrated superior predictive reliability for rapeseed–mustard price dynamics under varying market conditions.

5. Conclusion

The comparative evaluation across sunflower, soybean, and rapeseed–mustard price datasets clearly establishes the superiority of the GA optimized CNN+LSTM model over standalone CNN, LSTM, and conventional CNN+LSTM architectures. Although all models demonstrated strong learning ability during training ($R^2 \approx 0.99$), the GA optimized hybrid consistently achieved lower RMSE and MAE and higher R^2 values on validation and testing datasets, reflecting improved generalization and robustness under dynamic market conditions. The integration of convolutional feature extraction, sequential memory learning, and evolutionary hyperparameter optimization significantly enhanced predictive stability while minimizing overfitting.

Beyond methodological advancement, these results carry important policy implications. Reliable volatility forecasts can strengthen procurement planning, trade policy decisions, and price stabilization mechanisms. Improved prediction accuracy enhances market transparency, reduces income uncertainty for farmers, and supports progress toward Sustainable Development Goal 2.c aimed at limiting extreme food price volatility. Overall, the proposed framework offers a scalable and efficient approach for modeling complex agricultural commodity time series and can be extended to other agri-market forecasting applications.

References:

1. Box GEP, Jenkins GM, Reinsel GC. Time-series analysis: forecasting and control. 3rd ed. Delhi: Pearson Education; 2007.

2. Lama A, Jha GK, Paul RK, Gurung B. Modelling and forecasting of price volatility: an application of GARCH and EGARCH models. Agric Econ Res Rev. 2015;28(1):73–82.

3. Dinku T, Worku G. Asymmetric GARCH models on price volatility of agricultural commodities. SN Bus Econ. 2022; 2:181. doi:10.1007/s43546-022-00355-7.

4. Guindani LG, Oliveira GA, Ribeiro MHDM, Gonzalez GV, Lima JD. Exploring current trends in agricultural commodities forecasting methods through text mining: developments in statistical and artificial intelligence methods. Heliyon. 2024;10(23):e40568.

5. He XJ, Chen Z, Lin S. A stacked ensemble of LSTM, GRU and XGBoost with residual learning for corn futures price forecasting. Appl Intell. 2026;56:103. doi:10.1007/s10489-026-07087-3.

6. Jaiswal R, Jha GK, Kumar RR, et al. Deep long short-term memory based model for agricultural price forecasting. Neural Comput Appl. 2022;34:4661–4676. doi:10.1007/s00521-021-06621-3.

7. Ashfaq M, Khan I, Shah D, et al. Predicting wheat yield using deep learning and multi-source environmental data. Sci Rep. 2025;15:26446. doi:10.1038/s41598-025-11780-7.

8. Song C, Liu T, Ning W, Xu T, Song S, Li Z, et al. Wheat yield prediction based on parallel CNN-LSTM-attention with transfer learning model. Agriculture. 2025;15(23):2519. doi:10.3390/agriculture15232519.

9. Li H, Huang X, Luo F, Zhou D, Cao A, Guo L. Revolutionizing agricultural stock volatility forecasting: a comparative study of machine learning and HAR-RV models. J Appl Econ. 2025;28(1). doi:10.1080/15140326.2025.2454081.

10. Ben Romdhane W, Boubaker H. A hybrid HAR-LSTM-GARCH model for forecasting volatility in energy markets. J Risk Financ Manag. 2026;19(1):77. doi:10.3390/jrfm19010077.

11. Manogna RL, Dharmaji V, Sarang S. A novel hybrid neural network-based volatility forecasting of agricultural commodity prices: empirical evidence from India. J Big Data. 2025;12:85. doi:10.1186/s40537-025-01131-8.