CropTrendx: Predicting Agricultural Commodity Price Volatility with Multivariate Learning

Soumya Pachal[1], Shirsendu Roy[1], Sarnaavho Pal[1], Oindrila Paul[1], Debasmita Dutta[1], Krittika Dutta[1], Deepsubhra Guha Roy[2][0000−0001−7194−6950], and Piyali Datta[3*][0000−0001−9966−6619]

[1] Department of CSE(AIML), Institute of Engineering & Management, University of Engineering and Management, Kolkata, India
[2] IEM Centre of Excellence for Cloud Computing and IoT, Department of CSE(AIML), Institute of Engineering & Management, University of Engineering and Management, Kolkata, India.
[3] IEM-IIT Mandi Centre for Joint Research on Human Computer Interaction, Department of CSE(AIML), Institute of Engineering & Management, University of Engineering and Management, Kolkata, India
datta.piyali.in@gmail.com*

Abstract. This research introduces an integrated approach to price volatility forecasting of agricultural commodities like pulses and vegetables. Advanced machine learning techniques such as LSTM and Random Forest are used along with other time series models. Therefore, this would be well-positioned to capture both the linear and complex non-linear patterns of price trends by feeding it real-time market supply, inflation rates, fuel prices, and demand-supply dynamics. Cooperation of the states, especially of West Bengal, is critical in this model. Strategically, after having faced shortages and supply losses of neighboring states, it can release stock to mitigate the price hikes. Coordination of regional states fosters regional cooperation and stabilizes prices while ensuring efficient management of supply for both local markets and exports. The predictive tool generates valuable, actionable information to various stakeholders - the owner of cold storage, farmers, suppliers, and traders, and also the Department of Consumer Affairs. It offers the sound mechanism by which price fluctuations can be predicted; the system thus reduces financial and market risks, improves market efficiency, and promotes transparency. Its adaptability makes it a transformative solution for regulating prices and fair practice in agricultural markets and therefore enhances the stability of the market altogether.

Keywords: Price volatility · Agricultural commodities · Time series models · Ensemble models · Real-time data · Demand supply chain · Price fluctuations.

1 Introduction

Price volatility prediction in agricultural commodities such as pulses and vegetables using advanced techniques of machine learning and time series models

is a key research issue. Due to the significant interest generated by the high accuracy levels of price predictions among all stakeholders including farmers, cold storage owners, suppliers, and policymakers, this initiative aims to stabilize markets, avoid extreme price fluctuations, and mitigate the financial risks associated with price volatility [1], [2]. By doing so, it contributes to market stability, prevent price hikes, and reduce the financial consequences of unpredictable price changes. To achieve these goals, the system leverages core technologies such as long short-term memory (LSTM), which are specifically designed to model complex, non-linear patterns and long-term dependencies within sequential data. In addition, ensemble models like random forest offer robust prediction capabilities by utilizing multiple decision trees to capture both linear and non-linear relationships within the data. The integration of real-time data, such as market supply, inflation rates, and fuel prices, enhances the system's predictive accuracy. Cooperative arrangements among eastern states, especially with West Bengal, further ensure efficient management of regional stock, preventing price surges during times of low supply. This project delivers valuable insights for stakeholders, promoting market efficiency, fair trade, and overall stability in the agriculture market. The key framework of the paper is illustrated in Figure 1.

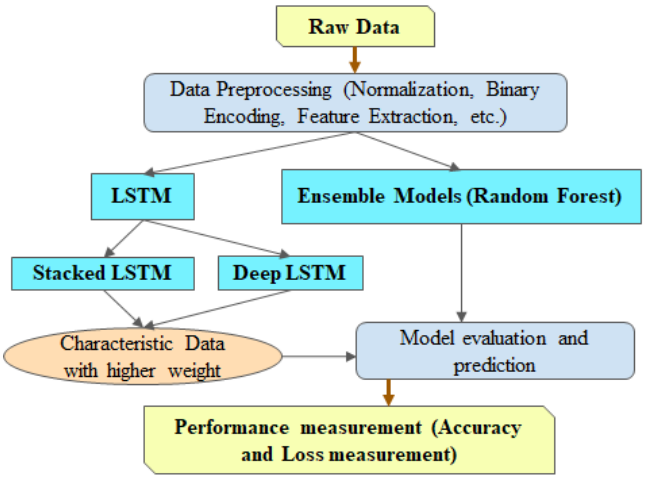

Fig. 1. Overall Flow of the Proposed Work.

[3] used combined forcasting model fo agriculture price prediction. [4] analyzed the effect of Random Forest, which is among the most efficient ML techniques that surpass conventional linear regression while predicting crop prices. The technique deals with complex agricultural data such as climate and cost of transportation, providing relatively frequent lower prediction errors in terms of MAE and RMSE. [5] proposes an LSTM Network for Short-Term Vegetable Price Forecasting that outperforms the traditional methods and other Machine Learning techniques like SVM and CNN. To predict the Chinese stocks return

rate Chen et al.[6] used LSTM model. Chen et al. [7] adopted LSTM model based on the attention mechanism to predict and discuss the trend of the stock market in Hong Kong.

2 Model description

2.1 Random Forest

The Random Forest [8] algorithm is an effective tree learning technique under the disciplines of Machine Learning. At the time of training, it forms several decision trees. A random subset of the dataset creates a tree, and those are measured as a random subset of features in each partition. This randomness introduces variability among individual trees and dramatically reduces the probability of overfitting and improves overall prediction performance [9]. The results of all trees are aggregated through voting (in classification problems) or averaging (in case of regression problems). This makes the decision stable and more accurate as a result of the many trees that provide the basis of such decisions. Finally, for regression-type problems, the average prediction for all trees in the forest is taken by the following.

$$\hat{y} = \frac{1}{K} \sum_{k=1}^{K} h_k(x) \tag{1}$$

An additional benefit of Random Forest is that it provides feature importance, which estimates the importance of each feature in making the prediction. That would probably be done as an average decrease in impurity, such as Gini impurity for classification or MSE for regression, by using a feature to split the data across all of the trees.

$$I(f_i) = \frac{1}{K} \sum_{k=1}^{K} \Delta I_{f_i}^{k} \tag{2}$$

Algorithm 1 *Predicting Price using Random Forest Model*
(Number of data points to be selected = k, Number of decision trees = N)

1: Select k number of data points in random manner from the associated training dataset.
2: Build a decision tree deploying the k data points.
3: Repeat Steps: 1 - 2 for N number of times.
4: Derive the predictions from each decision tree for the new data points. Assign the new data points to the selected category that is elected by majority.

2.2 XGBoost

This is a supervised machine learning model based on Preordering, and its concept involves sequencing all the feature data by the numerical value, looking for the best segmentation point, achieving the effect of reducing forecast error, and improving its accuracy after splitting the data into the left and right nodes

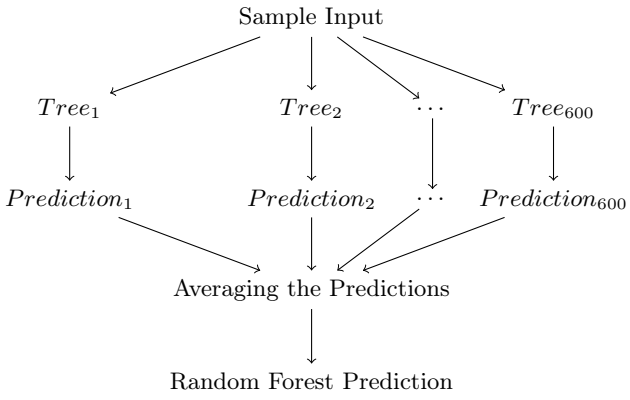

Fig. 2. Schematic Diagram of the Working Principle of a Random Forest Model.

[10]. The XGBoost model in this aspect is a representative of the Boosting algorithm in the technique of ensemble learning, which captures more rightly and accurately the nonlinear characteristics between various predictive variables [11]. The following shows an objective function merging the loss and regularization terms.

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(\hat{y}_i, y_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{3}$$

Where:

- $\mathcal{L}(\phi)$: the objective function.
- Loss function $l(\hat{y}_i, y_i)$ depicts the gap between the predicted value $\hat{y}_i$ and the true value y_i.
- K denotes the count of trees in the model.

$\Omega(f_k)$ is the regularization term for each tree f_k, usually defined as:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{4}$$

Where:

- γ is the regularization term controlling the complexity of the model (penalizes larger trees).
- T is the leave count in the tree.
- λ denotes L2 regularization parameter.
- w_j are the weights of the leaves.

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(\hat{y}_i, y_i) + \sum_{k=1}^{K} \left[\gamma T_k + \frac{1}{2}\lambda \sum_{j=1}^{T_k} w_j^2 \right] \tag{5}$$

2.3 Long Short-Term Memory (LSTM)

In Recurrent Neural Networks, the output from the last step is fed as an input for the next step. RNN consists of infinite loops to store information. Using the concept of memory in an RNN, temporal dimension problems are recommended to be solved by this. However, to develop a forecasting model that uses data sets from previous years, an RNN is generally not a good choice. The latest commodity prediction is quite crucial for precise forecasting, and looking at the latest commodity-related information is necessary for performing accurate forecasting. Nevertheless, some scenarios require the previous data to support the designed model while capturing patterns and movements. The new data may not exhibit all of them. Due to the existing gap between the past and the recent information, conventional RNNs do not learn to connect the information, which introduces *Long-Term Dependency* problem. LSTM is a key variant of RNN that consists of a memory cell in LSTM blocks to preserve information over a long period of time. In addition, a set of gates is used for memory manipulation in this architecture. These gates enable the model to capture and retain long-term dependencies, improving its ability to handle sequential data over extended periods.

LSTM solves the long-term dependency as well as the vanishing gradient problems. No weight adjustments are required in LSTM, and therefore, the complexity of the weight update is minimal. LSTM architecture comprises a memory cell and three regulators referred to as gates (input gate, output gate, and forget gate)[12]. They are deployed for information propagation inside or outside the LSTM unit. Some variations of LSTM units do not include all of these gates. The input gate controls the new value that flows into the cell. The forget gate determines whether to retain the value in the cell, while the output gate decides which value in the cell to propagate for computing the activation function at the output of the LSTM unit[13]. Some recurrent connections also persist. The logic gates' functionalities depend on the weights associated with the aforesaid connections and they are learned during the training phase. Eqs. (6–11) show the equations for Input node ($\gamma^{(t)}$), Input gate ($i^{(t)}$), Output gate ($o^{(t)}$), Cell state ((t)), Hidden gate ($h^{(t)}$), and Output layer ($y^{(t)}$) of LSTM, respectively.

$$\gamma^{(t)} = tanh(\omega^{\gamma x} x^{(t)} + \omega^{\gamma h} h^{(t-1)}) + \beta_\gamma \tag{6}$$

$$i^{(t)} = \alpha(\omega^{ix} x^{(t)} + \omega^{ih} h^{(t-1)}) + \beta_i \tag{7}$$

$$o^{(t)} = \alpha(\omega^{ox} x^{(t)} + \omega^{oh} h^{(t-1)}) + \beta_o \tag{8}$$

$$(t) = \gamma^{(t)} \odot i^{(t)} + (t-1) \odot o^{(t)} \tag{9}$$

$$h^{(t)} = tanh(s^{(t)}) \odot o^{(t)} \tag{10}$$

$$y^{(t)} = (\omega_{hy} h^{(t)} + b_y) \tag{11}$$

3 Experimental Results

3.1 Data Source

For this project, the data source has been drawn from the website **data.gov.in**. It covers the date range of **January 2022 up to the present.** The dataset is a treasure trove of agricultural and market-related information about several Indian states including, but not limited to, **West Bengal**, **Jharkhand**, **Bihar**, **Assam**, **Meghalaya**, and **Odisha**, amongst others. The inclusion of futures prices in the forecasting model benefits cash price predictability [14], and it is also subjected to different temporal and spatial factors of the current prices.This is the goldmine dataset on which the predictive model is constructed, thus allowing the analysis of price volatility in agricultural commodities such as pulses and vegetables. This would include critical variables such as market supply, demand, price trends, and regional stock availability. The participation of several states would ensure the realization of a better understanding of the existing market condition, which would thereby make the model more accurate in providing regional-specific forecasts. The real-time characteristic of the data adds further impetus to the accuracy of predictions, thereby making decision-making at the various stakeholder levels, including farmers, cold storage owners, and policymakers, much more effective.

3.2 Data Processing for Random Forest and XGBoost

For data preprocessing, the dataset of crop prices from various states is first merged to create a consolidated dataset. Both **Random Forest** and **XGBoost** are powerful machine learning models that handle numerical and categorical features effectively. However, appropriate preprocessing is essential to achieve optimal performance. Below are detailed steps for preprocessing data for these models:

- **Encoding for Categorical Variables:** It cannot execute machine learning models on categorical data natively. This native format for categorical data is strings or text, so such categories should be converted into numeric representations. *LabelEncoder* has been applied over State, District, Market categorical columns to convert all of them to numeric labels so that these labels can be further embedded, for Variety we use *Binary encoding* that converts every categorical variable into a series of binary digits. Every category is first translated to an integer, which is then represented as a binary number.
- **Scaling Continuous Features:** Continuous features such as min price, max price, and modal price are on quite different scales. This may cause problems during training by making the model too choosy of those features with the largest values. We apply *MinMaxScaler* on features to scale in the range [0,1] such that the ranges are uniformly distributed across features.
- **Lag Creation:** For modal price we create a lag of 1 day, 10 days. When you generate lag features, the initial n rows-that contain a NaN value because there is nothing that defines those rows previously-will have NaN values,

since there is no previous data for these rows. Deal with these NaNs by, dropping the NaN data. The created lag features (modal price lag 1, modal price lag 10, etc.) can now be included as additional input features when training your Random Forest or XGBoost models.

3.3 Data Processing for LSTM

Data preprocessing is a critical step that transforms raw data into a form suitable for the LSTM model [14]. Categorical Features are encoded and continuous features are scaled as mentioned earlier. Some model-specific data preprocessing steps are described here.

– **Feature Engineering with Date Variables:** Time-based features often follow seasonal trends, making them valuable inputs for the model. Direct use of month may not accurately reflect cyclic relationships (e.g., December and January are close, but numerically far apart). We extract the month from the date and apply trigonometric transformations (sine and cosine) to maintain the cyclic nature of the features.

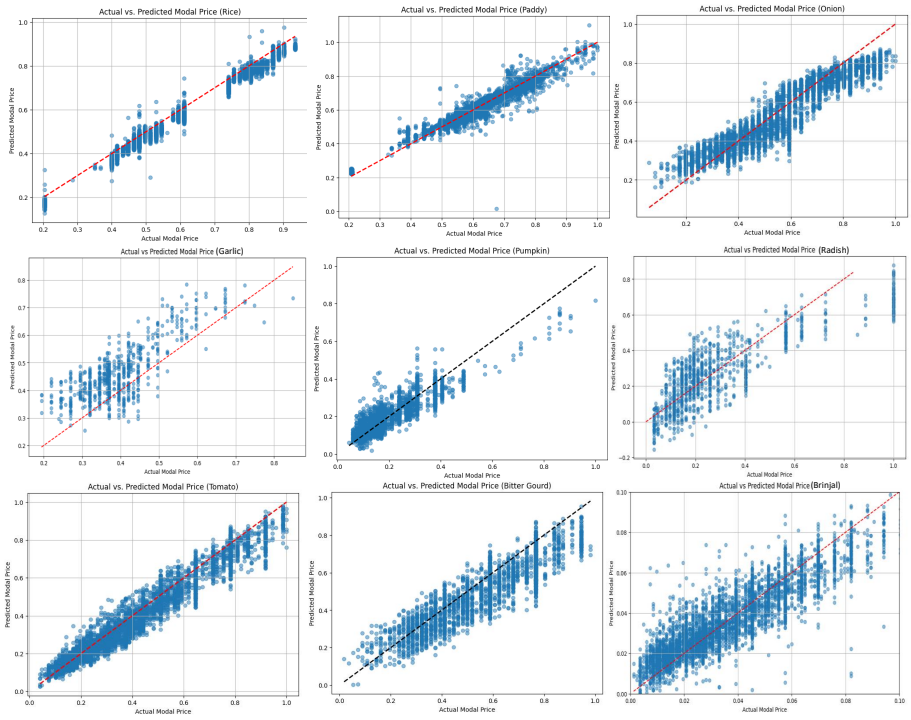

Fig. 3. Scatter plots of the predicted price vs. actual prices associated with different agricultural commodities.

- **Creating Sequential Data:** LSTMs are designed to process sequential data. Hence, arranging the dataset into sequences is essential to capture temporal dependencies and patterns. Data is grouped by unique combinations of categorical identifiers (e.g., state, district, market, variety,), each group is sorted by date, and sequences are created that preserve the order of observations. This step structures the data in a format suitable for LSTM processing.
- **Creating Categorical Sequences for Embedding:** Embedding layers require input sequences for each categorical feature. Hence, each categorical column is converted into sequences that match the shape of X. We extract each group's categorical values, excluding the last time step, and convert them into sequences for embeddings. This ensures categorical data is in the correct format for embedding.

3.4 Data Set

The dataset used in this study is obtained from the publicly available agricultural market price records hosted on the Government of India's open data platform (data.gov.in). The source contains daily market-level price information reported across multiple states. For each crop entry, the dataset includes essential attributes such as commodity, state, district, market, variety, date, minimum price, maximum price, and modal price. These features collectively offer comprehensive spatial and temporal coverage, enabling detailed analysis of price behaviours across diverse regions and commodities.

The raw dataset consists of a large number of daily observations collected from various states and markets over multiple years. As expected from real-world agricultural data, the records contain inconsistencies, missing values, and occasional anomalies. To ensure high-quality input for model training, a systematic preprocessing workflow was adopted. Records with missing or invalid values in key fields—particularly date, commodity, market, or modal price—were removed. The date field was standardized into a uniform format, while categorical variables (commodity, state, district, variety, and market) were cleaned for spelling variations and encoded numerically for model compatibility.

Price attributes (minimum, maximum and modal prices) were examined for outliers and formatting irregularities. Since modal price reflects the most commonly reported price for a given commodity on a given day, it was selected as the target variable for forecasting. Numerical features were normalized to maintain consistency across models. Finally, the dataset was grouped and arranged into chronological order to form structured time-series sequences suitable for Random Forest, XGBoost, and LSTM-based architectures.

This aggregated and preprocessed dataset provides a robust foundation for modelling price dynamics, capturing both spatial differences across markets and temporal variations within commodity price trends. Its breadth and quality enable meaningful insights and reliable predictive performance across the diverse agricultural commodities studied.

Table 1. Different performance metrics to assess our model considering various agricultural commodities.

Testset	Random Forest			XG Boost			LSTM		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
Rice	4.831	5.276	0.221	0.030	0.034	0.053	0.029	0.034	0.054
Paddy	4.831	5.270	0.421	0.032	0.030	0.221	0.022	0.032	0.039
Onion	33.910	35.001	1.112	32.180	106.410	1.360	0.051	0.064	0.110
Pumpkin	37.281	84.690	2.280	41.023	81.064	2.020	0.055	0.070	0.130
Radish	24.209	54.001	1.850	20.021	51.010	1.580	0.077	0.096	0.334
Tomato	53.093	138.450	1.840	66.052	124.460	2.046	0.044	0.059	0.119
Bit_gourd	28.067	47.052	2.320	26.027	46.410	2.029	0.066	0.086	0.150
Banana	49.094	408.061	1.030	45.063	385.660	1.410	0.077	0.096	0.208
Potato	30.083	455.010	0.990	17.043	47.070	1.05	0.012	0.021	0.022
Average	29.489	136.979	1.340	27.497	93.572	1.308	0.0481	0.062	0.129

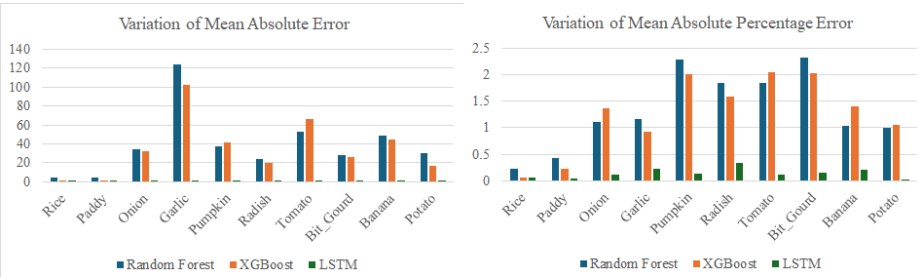

Fig. 4. Comparison of MAE and MAPE performance of Random Forest, XGBoost, and LSTM models across different agricultural commodities.

3.5 Accuracy Measures for Forecasting

Models' performance have been assessed with the three commonly used metrics, MAE: Mean Absolute Error, MAPE: Mean Absolute Percentage Error, and RMSE: Root Mean Squared Error.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|P_i^o - P_i^f| \tag{12}$$

Where, P_i^f and P_i^o be the forcasted and the observed price of a data point i, respectively.

Being an extension of MAE, MAPE satisfies the criteria of clarity of presentation and ease of interpretation. We interpret the criteria for evaluating the performance of the model using the MAPE and RMSE as follows.

- $10 >$ MAPE: Highly accurate
- $20 \geq$ MAPE ≥ 10: Good
- $50 \geq$ MAPE ≥ 20: Reasonable
- MAPE > 50: Inaccurate

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}|P_i^o - P_i^f| \times 100\% \tag{13}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i^o - P_i^f)^2} \tag{14}$$

Due to the squaring process in RMSE, it measures the average magnitude of errors, while a higher weight is given to larger errors.

Table 1 depicts different performance metrics to assess our model considering various agricultural commodities. Figure 3 shows the scatter plots showing the variation of predicted and original prices considering different commodities. Figure 4 depicts various accuracy metrics for different models.

Overall, for all test sets, the MAE of the Random Forest model was 29.489, RMSE = 136.979, and MAPE = 1.340. XGBoost on the other hand was much smaller in size and worked slightly better with average MAE of 27.497, RMSE of 93.572, and MAPE of 1.308. Nonetheless, the performance of the LSTM model is highly evident over Random Forest and XGBoost because it outperformed both of these models with lower error rates. Indeed, it got a mean absolute error of 0.0481, RMSE of 0.062, and MAPE of 0.129. This therefore reflects a clear winner in using LSTM to capture the rich complex temporal patterns of the data.

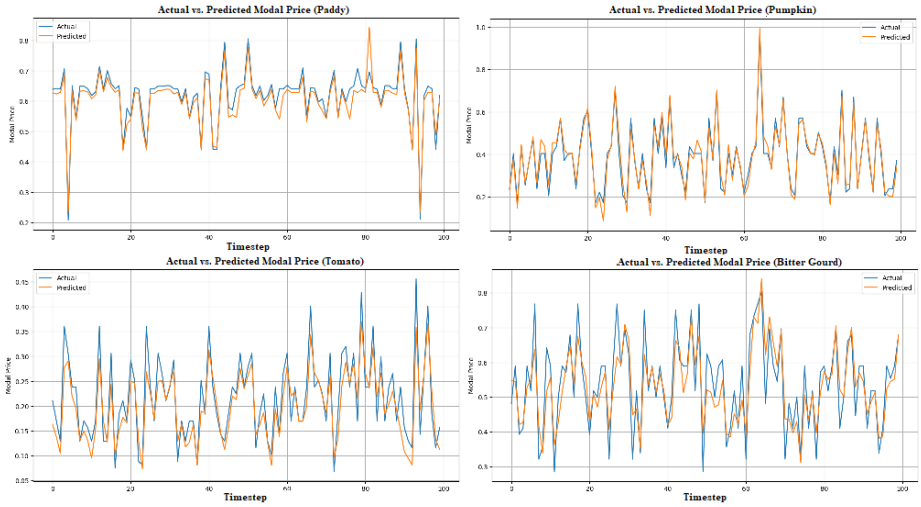

Fig. 5. Actual vs. Predicted Modal Price for different agricultural commodities.

Forget gate of LSTM enables it to forget irrelevant data. The sigmoid layer decides which portion of the old information must be removed. Now, in the next step, the decision regarding storing information from the current input in the

cell is taken by the input gate. Finally, the output gate decides the output of the network. In this paper, the model is initialized by loading the dataset, and scaling has been done using a min–max scaler. This avoids the variation of data in range, magnitudes, and units. The data has been processed in such a way that it suits the model. Next, hyperparameters' optimization is performed for optimizers, epochs, etc., which play a key role in deciding the model accuracy. Failing to appropriately tune the hyperparameters may result in overfitting or underfitting. After parameter tuning, a training-testing division is performed on the dataset. Figure 5 shows the variability of the actual and predicted modal prices for different commodities.

4 Conclusions

In this study, we employed Random Forest, XGBoost, and LSTM models to forecast the prices of a wide range of agricultural commodities. The comparative analysis clearly demonstrates that these models are capable of effectively learning both the linear and nonlinear behaviors present in real market data. Their performance, evaluated through MAE, RMSE, and MAPE, shows strong predictive capability and highlights the reliability of machine learning techniques in understanding commodity price fluctuations.

By integrating multiple algorithms and diverse market features, the proposed framework captures important temporal and structural patterns that influence price dynamics across regions. This makes the forecasting system not only accurate but also practical for real-world use by farmers, traders, storage operators, and policymakers. The insights generated through this approach can support informed decision-making, reduce financial uncertainty, and contribute to more stable and transparent agricultural markets.

Overall, the study reinforces that modern machine learning models, when applied thoughtfully to rich agricultural datasets, can offer meaningful and actionable predictions that benefit stakeholders across the entire agricultural ecosystem.

References

1. M. K. Mohanty, P. K. G. Thakurta, and S. Kar, "Agricultural commodity price prediction model: a machine learning framework," Neural Computing and Applications, Springer, 2023.
2. R. K. Paul, T. Das, and M. Yeasin, "Ensemble of time series and machine learning models for forecasting volatility in agricultural prices," National Academy Science Letters, Springer, 2023.
3. S. Banerjee and A. C. Mondal, "An ingenious method for estimating future crop prices that emphasises machine learning and deep learning models," International Journal of Information Technology, Springer, 2023.
4. I. Mahmud, P. R. Das, M. H. Rahman, A. R. Hasan, K. I. Shahin, and D. M. Farid, "Predicting Crop Prices Using Machine Learning Algorithms for Sustainable Agriculture," Proceedings of the 2024 IEEE Region 10 Symposium (TENSYMP), New Delhi, India, pp. 1–6, IEEE, 2024.

5. Zhang Q, Yang W, Zhao A, Wang X, Wang Z, Zhang L: Short-term forecasting of vegetable prices based on LSTM model—Evidence from Beijing's vegetable data. PLoS ONE 19(7): e0304881, 2024.
6. Chen, K.; Zhou, Y.; Dai, F. A LSTM-based method for stock returns prediction: A case study of China stock market. In Proceedings of the 2015 IEEE International Conference on Big Data, Santa Clara, CA, USA, pp. 2823–2824 , 2015.
7. Chen, S.; Ge, L. Exploring the attention mechanism in LSTM-based Hong Kong stock price movement prediction. Quant. Financ. 19,pp: 1507–1515,2019.
8. S Ray, T Biswas, W Emam, S Yadav, P Lal, P Mishra, "A random forest–convolutional neural network deep learning model for predicting the wholesale price index of potato in India," Potato Research, Springer, 2025.
9. G. H. H. Nayak, M. W. Alam, B. S. Naik, B. S. Varshini, et al., "Meta-Transformer: Leveraging Metaheuristic Algorithms for Agricultural Commodity Price Forecasting," Journal of Big Data, vol. 12, no. 1, article 138, SpringerOpen, 2025.
10. Phan, Q.T.; Wu, Y.K.; Phan, Q.D. A hybrid wind power forecasting model with XGBoost, data preprocessing considering different NWPs. Appl. Sci.11, 1100. 2021.
11. Zhou, Y.; Li, T.; Shi, J.; Qian, Z. A CEEMDAN and XGBOOST-based approach to forecast crude oil prices.pp:1-15,2019.
12. S. Ray, A. Lama, P. Mishra, T. Biswas, S. S. Das, and B. Gurung, "An ARIMA-LSTM model for predicting volatile agricultural price series with random forest technique," Applied Soft Computing, vol. 149, Part A, article 110939, Elsevier, 2023.
13. W. Bao, W. Su, X. Zhao, and J. Zhuang, "A Study on Short-Term Vegetable Price Prediction Based on the CNN-LSTM-Attention Model," Discover Food, vol. 5, article 176, Springer, 2025.
14. C. Sun, M. Pei, B. Cao, S. Chang, and H. Si, "A study on agricultural commodity price prediction model based on secondary decomposition and long short-term memory network," Agriculture, MDPI, 2023.
15. S. A. Alzakari, A. A. Alhussan, A. T. Qenawy, A. M. Elshewey, and M. Eed, "An Enhanced Long Short-Term Memory Recurrent Neural Network Deep Learning Model for Potato Price Prediction," Potato Research, Springer, vol. 68, pp. 621–639, 2025 (published online June 2024).