https://doi.org/10.62836/emi.v4i1.271.

Global Science Publishing

EMI

Economics & Management Information

https://ojs.sgsci.org/journals/emi

Article

Agricultural Product Price Prediction Based on the Quadratic Decomposition of CEEMDAN-VMD

Yuchun Jiang [1,*], Qixiang Miao [2] and Man Tang [3]

[1] *Guizhou University of Finance and Economics, Guiyang 550025, China*

[2] *Qingdao City University, Qingdao 266106, China*

[3] *Guizhou University of Finance and Economics, Guiyang 550025, China*

Abstract: Focusing on agricultural futures price forecasting, a prediction method based on quadratic decomposition is proposed in this paper in response to the non-stationarity, unstructured nature, and nonlinearity of agricultural price-time series data. Then drawing on the successes of deep learning in other financial domains, the quadratic decomposition of CEEMDAN-VMD that effectively addresses the non-stationarity of agricultural price-time series is introduced. And by constructing the CEEMDAN-SE-VMD-LSTM model, the paper performs an in-depth decomposition and refined processing of daily agricultural price data, successfully capturing the subtle characteristics of price fluctuations to achieve higher precision in forecasting. Moreover, the results indicates that the CEEMDAN-VMD model outperforms the comparative models in terms of forecasting accuracy for the three types of agricultural commodities.

Keywords: agricultural futures prices; financial time series prediction; quadratic decomposition; LSTM

1. Introduction

In recent years, the global climate system has shown increasing instability, with frequent extreme weather events becoming significant factors affecting agricultural production and market prices of agricultural products. Particularly during the cyclical transitions between La Niña and El Niño, plenty of regions have experienced unprecedented rapid reversals of drought and flood disasters, posing a severe challenge to global agricultural production. Besides, geopolitical conflicts, especially the Russia-Ukraine conflict, have had profound impacts on the global agricultural market. As a key grain-producing region, Ukraine's disrupted production has led to a tight international grain supply. Concurrently, the conflict has driven up the prices of agricultural inputs, increased production costs, and disrupted supply chain stability, collectively pushing up the prices of agricultural products. Therefore, agricultural product price forecasting is crucial for agricultural production and market stability. Accurate forecasting can help farmers plan their planting structures and sales strategies rationally, avoiding resource wastage due to blind production or market supply and demand imbalances. Moreover, accurate forecasting can provide a scientific basis for the government to formulate agricultural policies and regulate the market, which is beneficial for ensuring farmers' profits and consumers' rights. Additionally, agricultural product price forecasting can also promote collaborative development along the agricultural industry chain and enhance the overall efficiency of agriculture. Therefore, performing researches on agricultural product

Received: 9 December 2024; Accepted: 24 December 2024.

* Corresponding: Yuchun Jiang (1269266425@qq.com)

price forecasting and improving the accuracy and timeliness of forecasts is of great significance for promoting sustainable agricultural development and ensuring national food security.

In the traditional field of time series forecasting, such as ARIMA and VAR, the construction of these classic models often relies on the assumption of linear transformation of data. However, due to the inherent nonlinearity, non-stationarity, and multi-periodicity of agricultural product price-time series data, these traditional models struggle to capture their complex changes, leading to limited prediction accuracy and difficulty in comprehensively addressing the challenges of non-stationary and nonlinear forecasting.

With the emergency of machine learning and deep learning technologies, especially when they are combined with signal decomposition technology, new perspectives and effective pathways have been provided to address the aforementioned issues. Huang's proposed Empirical Mode Decomposition (EMD) method has opened up new horizons for signal and time series analysis [1]. However, subsequent research has found that EMD suffers from the problem of modal aliasing, affecting the decomposition effect. To overcome this deficiency, Huang and Wu developed the Ensemble Empirical Mode Decomposition (EEMD) algorithm, significantly improving the phenomenon of modal aliasing. Furthermore, the Complementary Envelope Empirical Mode Decomposition (CEEMD) proposed by Yeh et al., optimized the reconstruction error and decomposition completeness of EEMD [2]. Meanwhile, the Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) proposed by Torres further reduces the residual noise in the intrinsic mode functions and reconstruction error, enhancing the precision and stability of the decomposition [3]. In addition, the Variational Mode Decomposition (VMD) method proposed by Dragomiretskiy, as an adaptive and non-recursive nonlinear decomposition technique, can efficiently separate white noise in time series, achieving effective noise reduction of the signal [4].

Numerous scholars have combined the aforementioned decomposition techniques with deep learning models, achieving prominent improvements in forecasting accuracy across various domains. For instance, Ling et al. conducted comparative studies on Chinese pork price data using multiple decomposition methods, validating the superiority of the decomposition framework in enhancing prediction accuracy [5]. And Yang et al. delved into the differences in the interconnectivity between China's agricultural futures market and other financial markets at various time scales based on the EEMD method [6]. Concurrently, several researchers have further enhanced predictive performance by integrating decomposition techniques such as CEEMD and VMD with deep learning models like LSTM and Support Vector Regression (SVR) [7,8].

The proposal and application of quadratic decomposition have brought new breakthroughs in time series forecasting. This approach, by combining different decomposition algorithms, significantly reduces the non-stationarity and complexity of time series to a greater extent. Niu and Ji proposed a decomposition method based on Seasonal-Trend Decomposition using Loess (STL) and Variational Mode Decomposition (VMD), combined with the Support Vector Regression of Grey Wolf Optimizer (GWO-SVR), to achieve accurate forecasting of electricity demand [9]. This method captures the seasonal and trend components of the time series through STL, and further decomposes the remaining part through VMD, thereby revealing the intrinsic characteristics of the time series more finely. Liu and Mi proposed that they constructed a hybrid model combining VMD, Singular Spectrum Analysis (SSA), Long Short-Term Memory (LSTM) neural network, and Extreme Learning Machine (ELM) for wind speed prediction [10], which first uses VMD and SSA for a two-stage decomposition of the wind speed time series, then uses LSTM and ELM to predict the decomposed components separately, and finally combines the predicted results. Experimental results show that the accuracy of this hybrid model, which combines two decomposition algorithms, is significantly higher than that of hybrid models combining only one decomposition algorithm, further validating the effectiveness of the two-stage decomposition process in improving prediction accuracy. The further promotion of quadratic decomposition and deep learning algorithms, Cheng et al. combined VMD and EEMD, significantly improving the predictive ability through secondary processing of the participating components. Yan and Mu applied the quadratic decomposition technology of CEEMDAN and VMD to accurately predict the ultra-high frequency financial time series of iron ore futures [11]. Zhang, Zhu, and Fan et al. applied quadratic decomposition technology to the prediction of carbon emission trading prices, air quality, outpatient flow, and other fields, all of which verified the advantages of the integration of quadratic decomposition and deep learning models in forecasting [12–14].

In summary, the combination of quadratic decomposition technique with deep learning models has demonstrated exceptional predictive capabilities across various domains, providing new perspectives and methodologies for the prediction of agricultural product prices. This approach is expected to play an increasingly vital role in the field of agricultural product price forecasting. Based on the aforementioned literatures, this paper proposes a quadratic decomposition method using CEEMDAN, SE, and VMD, which decomposes the price-time series of three agricultural products—soybeans, corn, and cotton—into a series of Intrinsic Mode Functions (IMFs) to sufficiently reduce the non-stationarity of agricultural product time series while avoiding the loss of valuable information. Subsequently, a prediction model is constructed using the Long Short-Term Memory (LSTM) neural network, and its effectiveness is verified through fitting and validation to assess the predictive performance.

The innovation of this paper lies not only in the initial application of quadratic decomposition technique to the prediction of agricultural futures prices, but also in the integration of quadratic decomposition with deep learning, which brings a completely new perspective and superior predictive capabilities to the field of agricultural price forecasting. Through the decomposition of time series data two or more times, the introduction of quadratic decomposition technique in agricultural futures price prediction allows for a deeper exploration of hidden patterns and trends within the data. Compared to traditional single-level decomposition, this multi-level and multi-resolution decomposition method can more accurately capture subtle price changes and long-term trends, thereby enhancing the accuracy and stability of forecasts. This paper also empirically validates the effectiveness of quadratic decomposition technique in the prediction of agricultural futures prices. By comparing and analyzing with other forecasting methods (such as single decomposition models, traditional time series models, etc.), the paper demonstrates the advantages of quadratic decomposition technique in terms of forecasting accuracy, stability, and robustness, thereby strengthening the persuasiveness of the conclusions.

2. Materials and Methods

2.1. Data Sources

This paper utilizes data from 1 July 2014 to 1 July 2024, for the futures prices of soybeans, corn, and cotton in China, among which the soybean and corn prices are based on the daily closing data from the Dalian Commodity Exchange, while the cotton prices are derived from the daily data of the Zhengzhou Commodity Exchange. And the data was downloaded and processed using the Tushare data platform through Python for the main continuous contract data. The price trends for soybeans, corn, and cotton are depicted in Figure 1.

This paper employs Python to perform CEEMDAN decomposition on the raw data, clusters and integrates the Intrinsic Mode Functions (IMFs) based on entropy values, categorizing them into high-frequency, mid-frequency, and low-frequency sequences. Then high-frequency data undergoes a secondary VMD decomposition, with the matrix input into the LSTM forecasting model, while the mid-frequency and low-frequency sequences are vectorially input into the forecasting model to obtain the final prediction results. The results indicate that the quadratic decomposition model outperforms other comparative models in terms of agricultural product price forecasting performance.

2.2. Framework of the Proposed Model

The framework for agricultural futures price forecasting using the CEEMDAN-SE-VMD-LSTM model encompasses four main components: data preprocessing and decomposition, sample entropy calculation and clustering integration, high-frequency as well as mid-low frequency sequence forecasting, and model evaluation. The model framework is depicted in Figure 2.

The specific steps for forecasting are as follows:

Step 1: Data Preprocessing and Decomposition

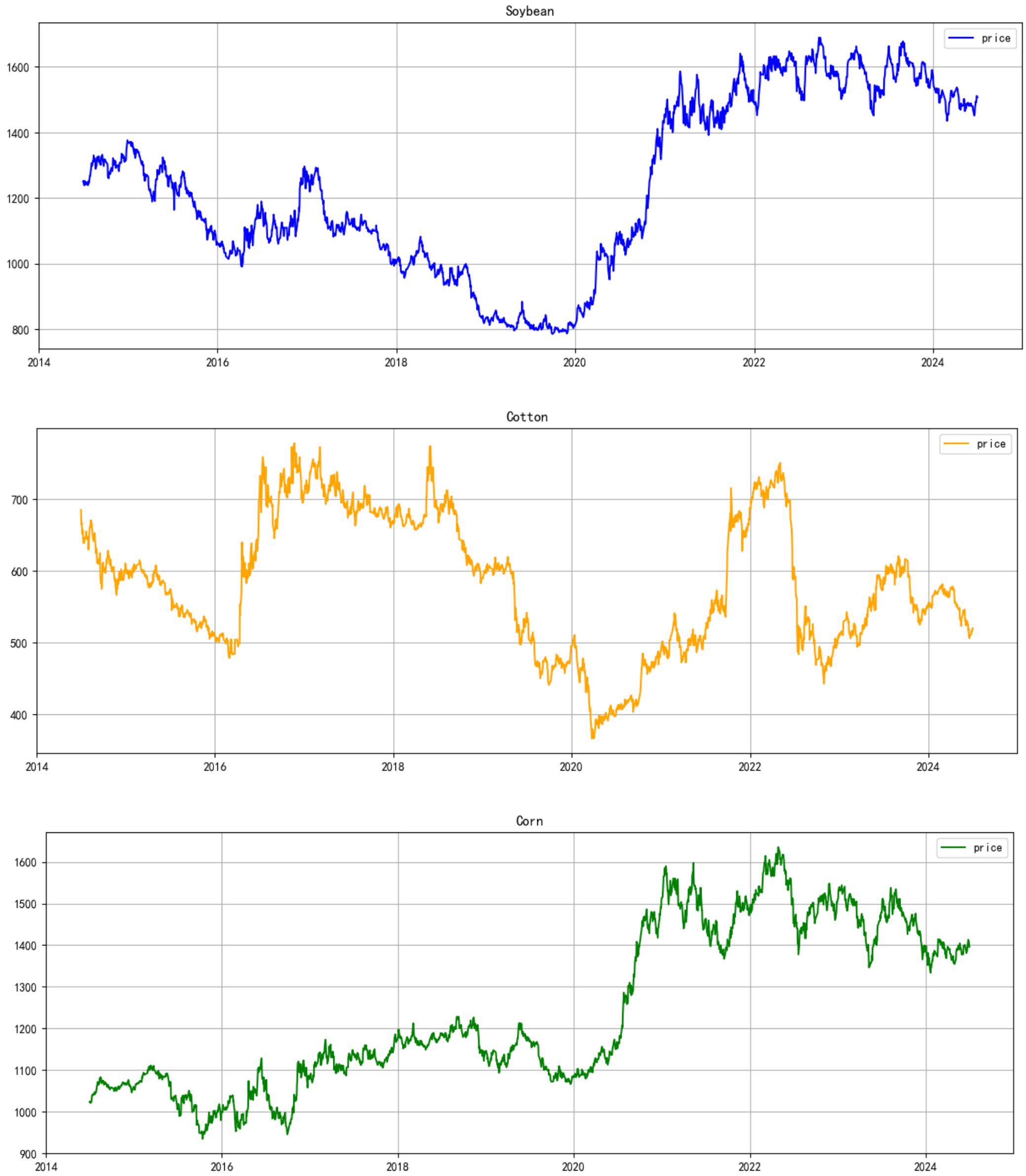

Figure 1. China's soybean, corn, and cotton futures price trends from 2014–2024.

The agricultural futures price data is subjected to CEEMDAN (Complete Ensemble Empirical Mode Decomposition with Adaptive Noise) to obtain a series of Intrinsic Mode Functions (IMFs).

Step 2: Sample Entropy Calculation and Clustering Integration

The sample entropy of the aforementioned IMF components is calculated, and clustering integration is performed based on the entropy values. Then the IMF components are categorized into high-frequency data Co-IMF0, mid-frequency sequence Co-IMF1, and low-frequency sequence Co-IMF2.

Step 3: High-frequency Sequence Processing and Forecasting

High-frequency data Co-IMF0 is subjected to Variational Mode Decomposition (VMD) to further decompose its frequency components. The VMD decomposition results are then input into the LSTM forecasting model, yielding evaluation metrics.

Step 4: Mid and Low-frequency Sequence Processing and Forecasting

The LSTM model is applied to forecast the decomposed low-frequency sequences. Vectors of Co-IMF1 and Co-IMF2 are input into the LSTM for forecasting, resulting in predicted outcomes and evaluation metrics. The experimental workflow is shown in Figure 2.

Step 5: Model Results

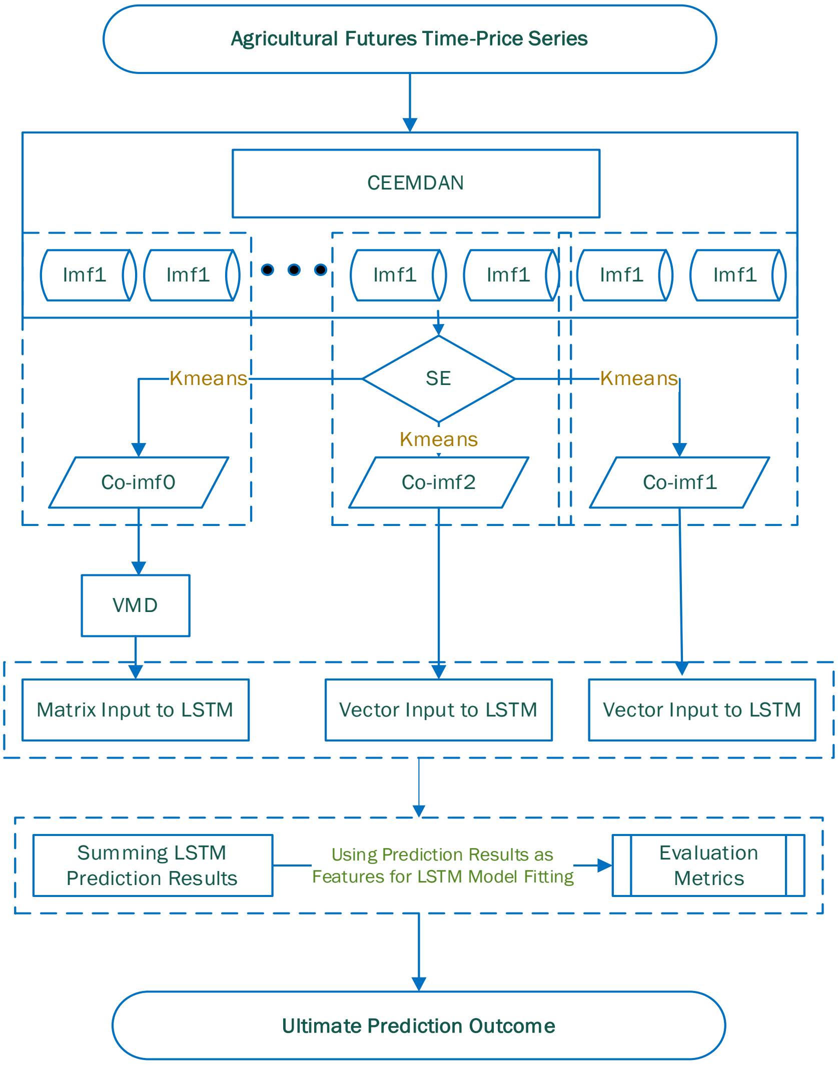

Figure 2. Framework of the proposed model.

The performance of the CEEMDAN-SE-VMD-LSTM model is evaluated by calculating the Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R-squared (R^2).

2.3. Decomposition Algorithm

2.3.1. CEEMDAN

The CEEMDAN algorithm is an improved version of the EMD. CEEMDAN enhances the EMD decomposition by introducing adaptive noise and ensemble averaging technique, addressing issues such as modal aliasing, endpoint effects, and decomposition instability. EMD serves as the foundation for signal time-frequency analysis, capable of decomposing signals without presetting basis functions. Through iteration, EMD strips the signal into multiple Intrinsic Mode Functions (IMFs) and a residual term. Then the IMFs capture local characteristics at different time scales, simplifying the signal structure and providing a solid foundation for subsequent processing. The related steps are as follows:

Step 1: The original signal $X(t)$ is traversed to identify all the extrema, which are typically the local maximum and minimum values of the signal. Subsequently, cubic spline interpolation is used to connect the maxima and minima, forming two smooth curves. The maxima are connected to form the upper envelope curve, denoted as $X_{max}(t)$; the minima are connected to form the lower envelope curve, denoted as $X_{min}(t)$.

Step 2: Interpolation calculation. For each time point ‘t’, calculate the values of the upper envelope curve $X_{max}(t)$ and the lower envelope curve $X_{min}(t)$ at time ‘t’. Compute the average of the upper and lower envelope values $X_{avr}(t)$:

$$X_{avr}(t) = \frac{X_{max}(t) + X_{min}(t)}{2} \tag{1}$$

Step 3: The component obtained from the i decomposition is denoted as $m_i(t)$ and the component obtained from the first decomposition is represented as $m_1(t)$, which is the difference between the original function $X(t)$ and the average of the upper and lower envelope curves $X_{avr}(t)$.

$$m_1(t) = X(t) - X_{avr}(t)$$

Step 4: Assess the IMF conditions. Analyze $m_1(t)$ based on two conditions for an IMF: (a) the number of extrema and the number of zero crossings must be equal or differ by at most one over the entire dataset, and (b) at any point in time, the mean value of the upper and lower envelopes defined by the local maxima and minima must be zero. If these conditions are met, designate $m_1(t) = \mathrm{IMF}_1$ and end the current iteration. If the conditions are not met, repeat steps 1 to 3 for $m_1(t)$, i.e., find the upper $m_{1max}(t)$ and lower envelopes $m_{1min}(t)$ and calculate the new difference $m_{11}(t)$.

$$m_{1avr}(t) = \frac{m_{1max}(t) - m_{1min}(t)}{2} \tag{2}$$

$$m_{11}(t) = m_1(t) - m_{1avr}(t)$$

Step 5: Repeat the aforementioned steps and continue iterating until the obtained component meets the conditions for an Intrinsic Mode Function (IMF). Let the number of repetitions be denoted as ‘k’; then the difference after the $m_{1k}(t)$ repetition is the first IMF, denoted as $imf_1(t)$. IMF_1 represents the IMF with the highest frequency in the decomposition of the original data. Subsequently, subtract $imf_1(t)$ from the original signal sequence $X(t)$.

$$m_2(t) = X(t) - imf_1(t) \tag{3}$$

Repeat the process of $m_2(t)$ to extract the next IMF, continuing until the remaining signal $m_{\mathrm{n}}(t)$ becomes a monotonic function or no more IMFs can be decomposed.

Step 6: When the $\mathrm{n_{th}}$ decomposition is reached, a residual component $res(t)$ with special properties is left behind. This residual component exhibits a monotonically increasing or decreasing characteristic, essentially capturing and reflecting the overall trend of the original data sequence. The final decomposition formula of EMD is as follows:

$$\begin{aligned} X(t) &= imf_1(t) + imf_2(t) + \ldots + imf_n(t) + res(t) \\ &= \sum\nolimits_{i=1}^{N} imf_i(t) + res(t), i = 1, 2, \ldots, N \end{aligned} \tag{4}$$

CEEMDAN is an improved algorithm of EMD decomposition, which aims to address issues such as modal aliasing, endpoint effects, and instability in decomposition that exist in EMD by introducing adaptive noise and ensemble averaging technique. Through multiple iterations, CEEMDAN gradually extracts the Intrinsic Mode Functions (IMFs) of the signal, ultimately obtaining a set of stable and meaningful decomposition results.

The CEEMDAN decomposition performs EMD on the original signal to obtain the first Intrinsic Mode Function (IMF) component. In each iteration, adaptive noise is added to the remaining signal, followed by EMD decomposition. The newly decomposed IMF components are combined with the results from the previous iteration. The number of iterations is determined based on preset stopping criteria. Then an overall average of all IMF components obtained from the iterations is performed to eliminate the noise impact. The related steps are as follows:

Step 1: Add white noise and perform preliminary decomposition. Randomly generated Gaussian white noise is added to the original data *X(t)*, resulting in *M* noisy datasets:

$$X_j(t) = X(t) + \varepsilon_1 E_1\left(\omega_j(t)\right), j = 1, 2, \ldots, M \tag{5}$$

In the process, *X(t)* represents the original data sequence, $\varepsilon_i \omega_j(t) (i = 1, 2, \ldots, \mathrm{N}; j = 1, 2, \ldots, \mathrm{M})$ denotes the Gaussian white noise, which follows a normal distribution. The term ε_i indicates the weight coefficient of the Gaussian white noise, i is the current processing iteration. $\omega_j(t)$ represents the Gaussian white noise added, j is the number of times white noise has been added, M is the maximum number of noise processing iterations. $E(\sim)$ stands for the Empirical Mode Decomposition process.

Step 2: Perform EMD decomposition on these M noisy datasets, and extract the first Intrinsic Mode Function (IMF) from each sequence, denoted as imf_1^j.

$$imf_1(t) = \frac{1}{M}\sum_{j=1}^{M} imf_1^j(t), j = 1, 2, \ldots, M \tag{6}$$

Step 3: Calculate the difference between the original data and the first IMF to obtain the residual $r_1(t)$ after the first decomposition:

$$r_1(t) = X(t) - imf_1(t) \tag{7}$$

Repeat the process of adding white noise and performing EMD decomposition on the residual $r_1(t)$ to obtain the second IMF sequence $imf_2(t)$:

$$imf_2(t) = \frac{1}{M}\sum_{j=1}^{M} E_2\Big(r_1(t) + \varepsilon_2 E_2\big(\omega_j(t)\big)\Big), j = 1, 2, \ldots, M \tag{8}$$

Step 4: Proceeding in this manner, iterate through the calculations to obtain subsequent IMF sequences $imf_i(t)$ and residuals $r_i(t)$ ($I = 1, 2, \cdots, N$):

$$imf_i(t) = \frac{1}{M}\sum_{j=1}^{M} E_i\Big(r_{i-1}(t) + \varepsilon_i E_i\big(\omega_j(t)\big)\Big), j = 1, 2, \ldots, M \tag{9}$$

$$r_i(t) = r_{i-1}(t) - imf_i(t) \tag{10}$$

Step 5: when $i = N$, end iteration. When the final residual $r_N(t) = res(t)$ is obtained, the original data *X(t)* can be expressed as the sum of all IMF components and the trend item:

$$X(t) = \sum_{i=1}^{N} imf_i(t) + res(t), i = 1, 2, \ldots, N \tag{11}$$

CEEMDAN decomposition enhances the adaptability of the decomposition process by introducing adaptive noise and iterative updating, effectively suppressing modal aliasing and improving decomposition accuracy. Compared to EEMD, it reduces unnecessary decomposition iterations, thereby increasing computational efficiency.

2.3.2. VMD

VMD employs the Alternating Direction Method of Multipliers (ADMM), an efficient optimization algorithm, to solve the constructed variational problem. Through continuous iterative optimization, it gradually approaches the optimal solution, thereby accurately extracting the central frequency and bandwidth of each modal function, achieving effective decomposition of the signal. The steps are as follows:

Step 1: Initialize the parameters, $\{\hat{\mu}_k^1\}, \{\omega_k^1\}, \{\hat{\lambda}^1\}, (n \leftarrow 0)$

Step 2: Execute the loop, incrementing $n = n + 1$, and for all $\omega \geqslant 0$, ensure that:

$$\frac{\hat{\mu}_k^{n+1} = \hat{f}(\omega) - \sum_{ik}\hat{\mu}_k^{n+1}(\omega) - \sum_{ik}\hat{\mu}_k^{n}(\omega) + \dfrac{\hat{\lambda}^n(\omega)}{2}}{1 + 2\alpha\left(\omega - \omega_k^n\right)^2} \tag{12}$$

$$\omega_k^{n+1} = \frac{\int_0^\infty \omega \left|\hat{\mu}_k^{n+1}(\omega)\right|^2 dw}{\int_0^\infty \left|\hat{\mu}_k^{n+1}(\omega)\right|^2 dw} \tag{13}$$

Step 3: For all $\omega \geqslant 0$, ensure that condition (2)–(16) is met, where γ is the noise tolerance.

$$\hat{\lambda}^{n+1}(\omega) = \widehat{\lambda^n}(\omega) + \gamma\left[\hat{f}(\omega) - \sum_k \hat{\mu}_k^{n+1}(\omega)\right] \tag{14}$$

Step 4: Pre-set the judgment precision and repeat steps 2 and 3 until the constraint conditions are satisfied.

$$\frac{\sum_k \left\| \hat{\mu}_k^{n+1} - \hat{\mu}_k^n \right\|_2^2}{\left\| \hat{\mu}_k^n \right\|_2^2} \varepsilon \tag{15}$$

VMD employs an iterative process to dynamically update the central frequencies and bandwidths of the IMF components until the preset termination conditions are met. By adaptively dividing the frequency bands according to the spectral characteristics of the signal, VMD precisely extracts k IMF components, effectively avoiding modal aliasing and enhancing the precision of the decomposition.

2.4. Forecasting Algorithm: LSTM Neural Network

The LSTM network is utilized to process sequential data, leveraging its complex internal cell states and three gating mechanisms (forget gate, input gate, and output gate) to achieve selective transmission and memory control of information. The LSTM determines the retention of old information through the forget gate, adds new information through the input gate, and controls the output of information to the hidden state through the output gate. This effectively addresses the gradient problem associated with RNNs and is suitable for data prediction tasks involving long-term dependencies (As shown in Figure 3).

(1) The Forget Gate determines how much information from the previous cell state needs to be forgotten or retained. It receives the hidden state h from the previous moment and the input x from the current moment, and outputs a value between 0 and 1 through a sigmoid function $\sigma(x) = (1 + e^{-x})^{-1}$. A value of 0 indicates complete forgetting of information, while a value of 1 indicates complete retention of information. The sigmoid function's output value is multiplied by the cell state C_{t-1}, thereby deciding how much of the old information is retained.

(2) The Input Gate (Update Gate) determines how much new information should be added to the cell state at the current moment, operating in parallel with the Forget Gate. It also receives the previous hidden state h_{t-1} and the current input $X(t)$, and decides on an update proportion through a sigmoid function. Concurrently, another input transformation through the tanh function generates new candidate information, which is then multiplied by the output of the Input Gate to obtain the actual amount of new information to be added.

(3) Output Gate determines how much information from the current cell state should be output to the hidden state h_t. The output gate, based on the previous hidden state h_{t-1} and the current input $X(t)$, uses a sigmoid function to decide the output proportion. Concurrently, the cell state undergoes a nonlinear transformation through a tanh function, which is then multiplied by the output of the output gate to obtain the final hidden state output.

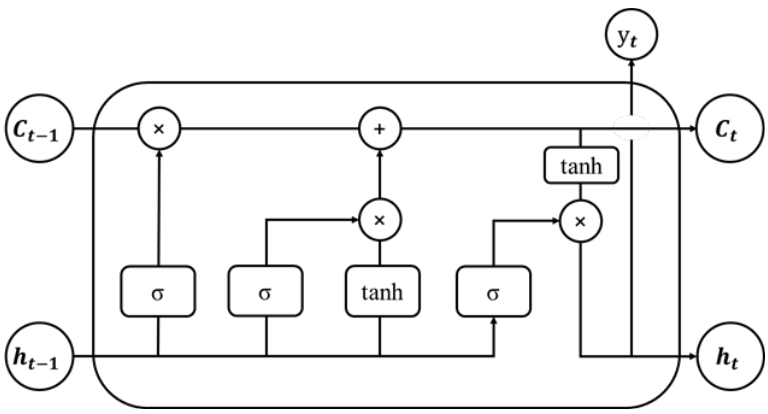

Figure 3. The Internal Structure of an LSTM Model Neuron.

2.5. Model Training and Algorithm Implementation

2.5.1. Model Parameter Configuration

The empirical study employs Python language to implement all algorithms (CEEMDAN, VMD, SE, and LSTM) in the VSCode compiler. The machine learning section utilizes Scikit-learn, which is a collection of high-quality machine learning algorithms featuring a variety of functionalities required throughout the entire machine learning process, from data preprocessing to model training, evaluation, and parameter tuning. The deep learning section employs Keras, a high-level neural networks API that is solidly built upon the TensorFlow backend, providing robust support for the rapid construction and training of deep learning models. Detailed development environment as shown in Table 1.

Table 1. Development Environment.

Component	Description
CPU	AMD Ryzen 7 7735H
RAM	16 GB
Operating System	Windows 11 64-bit
Programming Language	Python 3.11
IDE	Visual Studio Code (VSCode)
Libraries/Frameworks	PyEMD, sampen, vmdpy, Keras

In the CEEMDAN decomposition, the number of trials adding white noise is set to trials = 10, the number of clusters in the cluster analysis is num_clusters = 3, the sample entropy length is mm = 1, the similarity tolerance is r = 0.1, and the number of clusters for k-means is kmeans_num = 3, representing high, medium, and low frequencies, respectively. For the high-frequency sequence VMD, the bandwidth parameter is alpha = 2000, the noise tolerance parameter is tau = 0, the number of IMFs obtained from VMD decomposition is K = 10, and the convergence tolerance is tol = 1×10^{-7}.

The LSTM network comprises three layers, with each layer containing 128, 64, and 32 neurons, respectively. Except for the last layer, each LSTM layer returns the output for the entire sequence. All LSTM layers utilize the hyperbolic tangent (tanh) activation function, which outputs values between −1 and 1. Following each LSTM layer, a Dropout layer is added with a dropout rate of 0.2 to prevent overfitting of the model. The Dense fully connected layer employs the tanh activation function. The initial learning rate of the model is set to 0.001, with adam as the optimizer, and Mean Squared Error (MSE) is used as the loss function.

To effectively prevent model overfitting, this study employs the EarlyStopping callback function mechanism. Within this mechanism, the patience value is meticulously set to one-tenth of the total number of training epochs, and then multiplied by 5 to establish the threshold for early stopping. If the validation loss does not show a significant decrease over 5 consecutive training epochs defined by the patience value, the training process will be prematurely terminated to avoid the model from becoming overly fitted to the training data.

Additionally, this study introduces the learning rate decay callback function to address the issue of the validation loss becoming stagnant. Unlike the patience value setting in early stopping, the learning rate decay section directly sets the patience value to one-tenth of the total number of training epochs. When the validation loss does not show improvement within one-tenth of the total number of epochs, the learning rate decay callback function will automatically trigger, reducing the learning rate by a preset proportion. This adjustment strategy helps the model break free from the constraints of potential local optima, thereby promoting further model optimization and exploring superior performance outcomes.

2.5.2. Assessment of Accuracy

The performance of the model is evaluated using four key metrics: the Root Mean Square Error (RMSE), the Mean Absolute Percentage Error (MAPE), the Mean Absolute Error (MAE), and the Coefficient of Determination (R^2). The smaller the values of RMSE, MAPE, and MAE, the better the fitting effect of the model. The R^2 reflects the goodness of fit of the model, with values closer to 1 indicating a better fit.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2}{n}} \tag{16}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \tag{17}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right| \tag{18}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2}{\sum_{i=1}^{n}\left(\bar{y}_i - y_i\right)^2} \tag{19}$$

In the context, y_i represents the true value of the sequence, $\hat{y}_i$ represents the predicted value of the sequence, $\bar{y}_i$ represents the mean of the sequence, and n represents the total number of forecasted data points.

3. Results

3.1. Descriptive Statistics of Agricultural Product Prices

It can be seen from the above Table 2 that the sample size for each of the three agricultural products is 2432. The maximum value of soybean futures prices is 1689.47, with a minimum value of 785.81, indicating a relatively large price fluctuation range. The maximum value of corn futures prices is 1653.80, but the minimum value is relatively higher at 934.56, with a smaller price fluctuation range compared to soybeans. The maximum value of cotton futures prices is 777.80, and the minimum value is 366.27, which has the smallest price fluctuation range among the three. The standard deviations for soybeans, corn, and cotton are 264.70, 191.01, and 93.94, respectively. The kurtosis values for all three agricultural products are less than 3, suggesting that their price distributions are flatter than the normal distribution, with fewer extreme values in the data. The skewness values for soybeans and cotton are close to 0, indicating that their price distributions are relatively symmetric. The skewness value for corn is 0.374, with a slight right skew, suggesting that there are relatively more high-value numbers in its price distribution.

Table 2. Statistical Characteristics of Agricultural Product Futures Prices.

	Sample Size	**Maximum**	**Minimum**	**Standard Deviation**	**Kurtosis**	**Skewness**
Soybeans	2432	1689.47	785.81	264.70	−1.26	−0.005
Corn	2432	1653.80	934.56	191.01	−1.37	0.374
Cotton	2432	777.80	366.27	93.94	−0.94	−0.015

3.2. Decomposition Results

Taking the soybean futures price data as an example, the original data was decomposed using the CEEMDAN method, employing the pyEMD package in Python for the decomposition. The time series decomposition of the soybean futures price yielded 8 Intrinsic Mode Functions (imf0–imf7) and 1 residual term (imf8), as shown in Figure 4. The CEEMDAN decomposition results indicate that no obvious mode mixing is present in the individual Intrinsic Mode Functions.

The high-frequency Intrinsic Mode Functions (IMFs) concentrate more details of the soybean futures time series, including noise, making them more complex and less predictable than the low-frequency IMFs. Cluster analysis is performed on imf0–imf7 using Symbolic Entropy (SE). Figure 5 and Table 3 display a series of clustering results. Each component is assigned a cluster number, with imf0 and imf1 assigned to Cluster 0, imf2 to Cluster 2, and imf3 to imf7 to Cluster 1. Through K-means clustering of the original data, where each component represents a specific dimension of the data, Cluster result 0 indicates that the component is assigned to the high-frequency sequence during the clustering process, Cluster result 1 indicates that the component is assigned to the low-frequency sequence, and Cluster result 2 indicates that the component is assigned to the mid-frequency sequence.

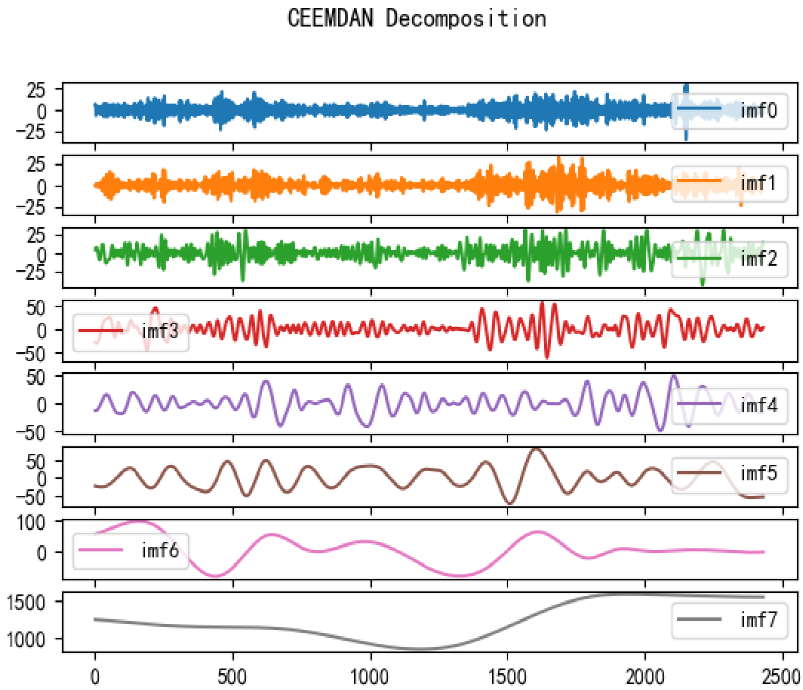

Figure 4. The CEEMDAN Decomposition Results of Soybean Futures Price-Time Series.

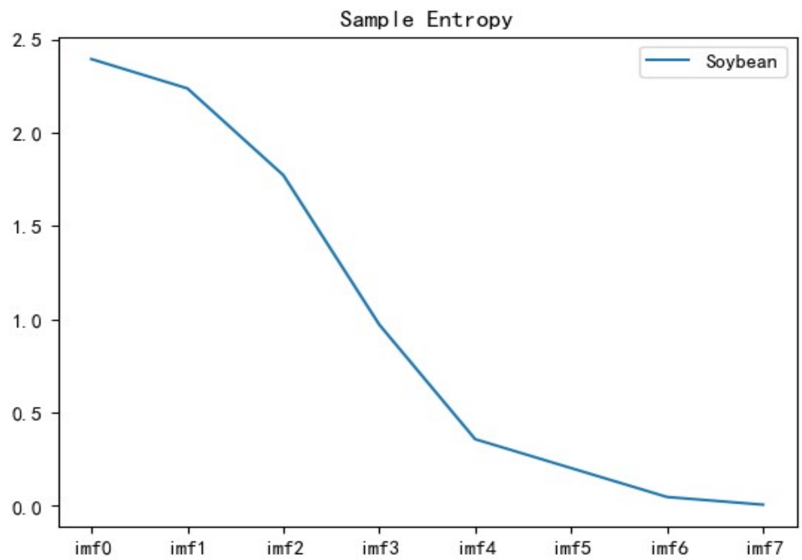

Figure 5. Sample entropy and clustering results.

Table 3. Sample Entropy and Clustering Results.

Component	Clustering Result
imf0	0
imf1	0
imf2	2
imf3	1
imf4	1
imf5	1
imf6	1
imf7	1

Based on the clustering results, sequences of different frequencies are integrated, with the integration results shown in Figure 6. co-imf0 represents the high-frequency sequence, co-imf1 represents the mid-frequency sequence, and co-imf2 represents the low-frequency sequence. The high-frequency sequence co-imf0 is subjected to Variational Mode Decomposition (VMD), which further decomposes and reduces complexity, eliminating noise. In the VMD decomposition process, the parameter K = 10 indicates that the maximum number of VMD results is expected to be 10, as shown in Figure 7. The decomposed Intrinsic Mode Functions (IMFs) are reshaped and input as matrices into the LSTM model, while the vectors of co-imf1 and co-imf2 are also input into the LSTM to obtain the final forecast results.

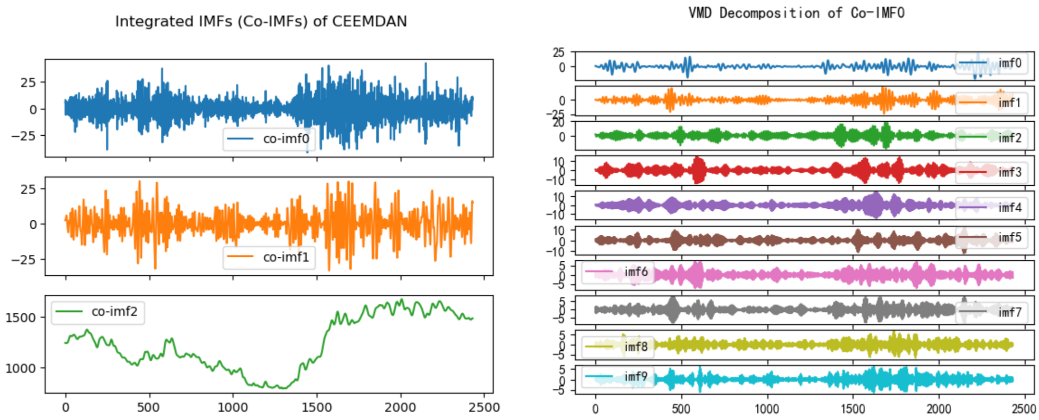

Figure 6. co-imf0 Decomposition Results.

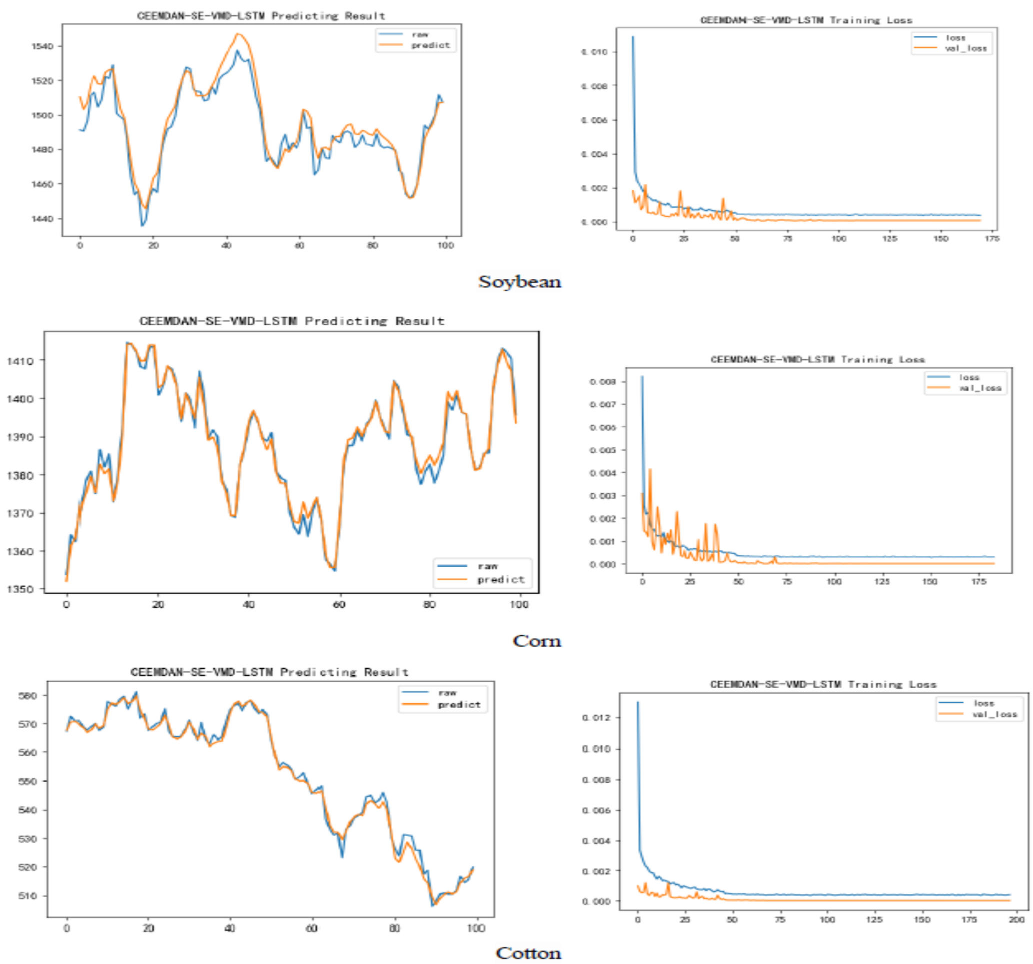

Figure 7. Predicting Result and Training Loss.

3.3. Forecasting Results of Modes

The input of co-imf0, co-imf1, and co-imf2 into the LSTM model yields the forecast results, which are then summed to obtain the predictive outcomes of the CEEMDAN-SE-VMD-LSTM model. The CEEMDAN-SE-VMD-LSTM model's predictions for three agricultural products (soybeans, corn, and cotton) include metrics such as the Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Coefficient of Determination (R^2), with the results presented in Table 4, and the forecast outcomes and loss rates depicted in Figure 7.

For soybeans, the Mean Absolute Percentage Error (MAPE) is 0.0063, the Mean Absolute Error (MAE) is 0.0049, the Root Mean Square Error (RMSE) is 0.0057, and the Coefficient of Determination (R^2) is 0.9495.

For corn, the predictive performance is even more outstanding, with a MAPE of only 0.0029, an MAE of 0.0019, an RMSE of 0.0025, and an R^2 as high as 0.9864, indicating a very strong model fit and predictive capability for corn data.

For cotton, the MAPE is 0.0082, the MAE is 0.0035, the RMSE is 0.0047, and the R^2 reaches 0.9923, which is also a very high value, suggesting that the model also performs excellently in processing cotton data.

Therefore, the CEEMDAN-SE-VMD-LSTM model has achieved good performance metrics in the forecasting of these three agricultural products, demonstrating the effectiveness and accuracy of the model.

Table 4. predictions for three agricultural products.

Agricultural Products	Indicators	CEEMDAN-SE-VMD-LSTM
Soybeans	MAPE	0.0063
	MAE	0.0049
	RMSE	0.0057
	R^2	0.9495
Corn	MAPE	0.0029
	MAE	0.0019
	RMSE	0.0025
	R^2	0.9864
Cotton	MAPE	0.0082
	MAE	0.0035
	RMSE	0.0047
	R^2	0.9923

4. Discussion

To validate the performance of the quadratic decomposition deep learning model in forecasting the prices of three agricultural product futures, the predictive performance of the non-decomposed LSTM, CEEMDAN-LSTM, VMD-LSTM, and CEEMDAN-SE-VMD-LSTM models are compared using four different metrics to measure the models' forecasting effectiveness. The results are presented in Table 5. Figure 8 contrasts the fit quality R^2 of the three agricultural product predictions by the five models.

In all metrics, the LSTM model performed the worst, with errors in MAPE, MAE, and RMSE significantly greater than those of other models. And a negative R^2 indicates that the fit of the standalone LSTM model is very poor, especially in the prediction of soybean prices. To further verify the predictive effect of the LSTM model on agricultural products, a Convolutional Neural Network (CNN) and Long Short-Term Memory Network (LSTM) combined model, CNN-LSTM, is applied to the price prediction of three agricultural products. The results from the table show that CNN-LSTM has made improvements in the prediction of corn and cotton, but the outcomes are still not as good as those of the subsequent decomposition models.

Table 5. Comparison of Model Prediction Results.

Agricultural Products	Indicators	LSTM	CNN-LSTM	CEEMDAN-LSTM	VMD-LSTM	CEEMDAN-SE-VMD-LSTM
Soybean	MAPE	15.2905	10.7965	0.0848	0.0250	0.0063
	MAE	0.1342	170.2799	0.0587	0.0225	0.0049
	RMSE	0.01416	176.1069	0.0670	0.0296	0.0057
	R^2	−3.9463	−8.3669	0.6175	0.7821	0.9495
Corn	MAPE	4.0259	2.1155	0.0448	0.0311	0.0029
	MAE	0.0291	30.4035	0.0313	0.0219	0.0019
	RMSE	0.0354	36.3886	0.0367	0.0259	0.0025
	R^2	0.7822	0.5325	0.7666	0.8828	0.9864
Cotton	MAPE	8.4403	2.4148	0.0860	0.0253	0.0082
	MAE	0.0336	12.9731	0.0319	0.0102	0.0035
	RMSE	0.0438	17.0694	0.0420	0.0135	0.0047
	R^2	0.7870	0.8092	0.8064	0.9800	0.9923

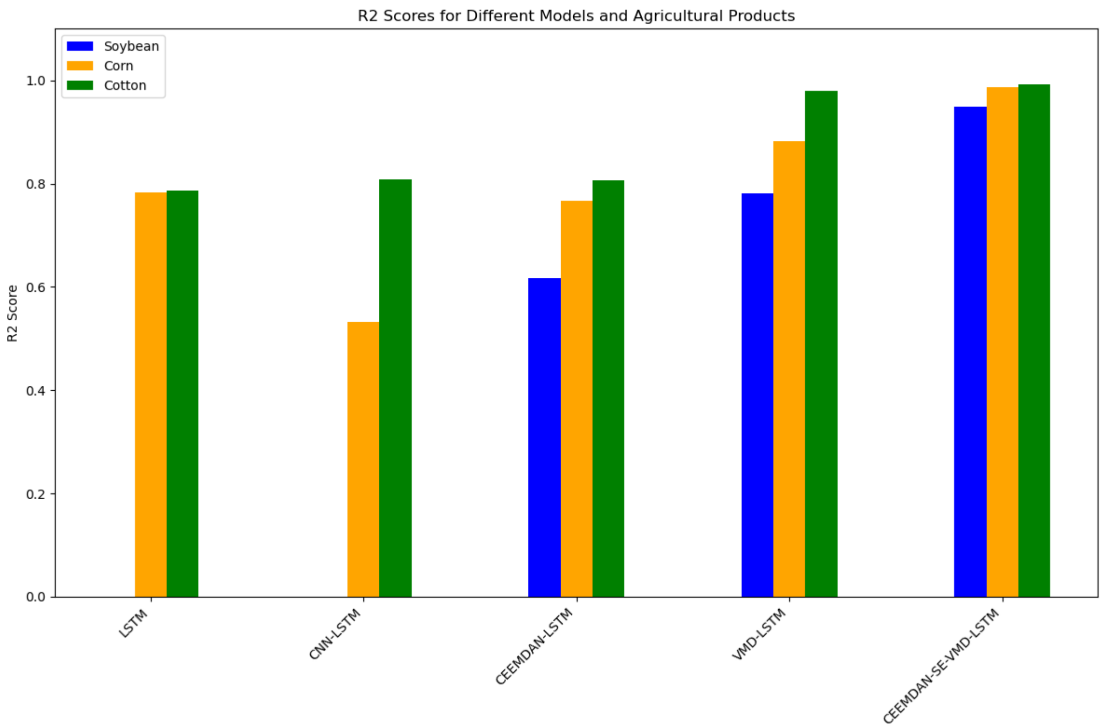

Figure 8. R^2 Scores for Different Models and Agricultural Products.

In the primary decomposition models, the performance of the CEEMDAN-LSTM model is significantly enhanced, particularly in terms of MAPE, MAE, and RMSE. The VMD decomposition also notably improves the performance of the LSTM model, outperforming the CEEMDAN-LSTM model across all four metrics: MAPE, MAE, RMSE, and R^2. It is evident that for non-linear and non-stationary price series of agricultural products, VMD decomposition is markedly more effective than CEEMDAN decomposition in primary decomposition models. The quadratic decomposition model, which combines CEEMDAN and VMD decompositions, achieves the best performance in all metrics. Then the CEEMDAN-SE-VMD-LSTM model demonstrates a significant improvement in evaluation metrics for the price prediction of soybeans, corn, and cotton compared to the two primary decomposition models. Specifically, for soybeans, the R^2 increases from 0.6175 and 0.7821 to 0.9495, representing a precision enhancement of 53.76% and 21.66%, respectively; for corn, the R^2 increases from 0.7666 and 0.8828 to 0.9864, indicating a precision enhancement of 28.67% and 11.74%, respectively; and for cotton, the R^2 increases from 0.8064 and 0.9800 to 0.9923, showing a precision enhancement of 23.03% and 1.26%, respectively. The CEEMDAN-SE-VMD-LSTM model has exhibited extremely high predictive accuracy in the forecasting of agricultural product futures prices.

5. Conclusion

This study addresses the forecasting issue of futures prices for three agricultural products: soybeans, corn, and cotton, by proposing the CEEMDAN-SE-VMD-LSTM hybrid model. By integrating CEEMDAN and VMD signal decomposition techniques, the model effectively overcomes the limitations inherent in the original LSTM model when dealing with complex time series data. Experimental results demonstrate that the hybrid model has achieved significant effectiveness in forecasting the futures prices of the three agricultural products, with R^2 values exceeding 94%, showing higher predictive accuracy compared to other comparative models.

The introduction of CEEMDAN and VMD decomposition techniques successfully extracts valuable information from time series data and effectively reduces noise and redundant components, providing clearer and more effective input features for the LSTM model's accurate predictions. Moreover, the combination of quadratic decomposition algorithms with deep learning technology further enhances the model's predictive accuracy and robustness. The CEEMDAN-SE-VMD-LSTM model proposed in this study exhibits exceptional forecasting performance and practical value in the prediction of agricultural product futures prices, offering strong support for risk management and investment decision-making in the agricultural product market.

Funding

This research received no external funding.

Author Contributions

Y. J.: conceptualization, methodology; Q. M.: data curation, writing—original draft preparation; M. T.: visualization, investigation. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

Not applicable.

Conflicts of Interest

The authors declare no conflict of interest.

References

1 Huang NE, Zheng S, Steven R, *et al*. The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis. *Mathematical Physical and Engineering Sciences* 1998; **454**: 1971.

2 Yeh JR, Shieh JS, Huang NE. Complementary Ensemble Empirical Mode Decomposition: A Novel Noise Enhanced DATA analysis Method. *Advances in Adaptive Data Analysis* 2010; **2**: 135–156.

3 Torres ME, Colominas MA, Schlotthauer G, *et al.* A Complete Ensemble Empirical Mode Decomposition with Adaptive Noise. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and

Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 4144–4147.

4 Dragomiretskiy K, Zosso D, Bertozzi AL, *et al*. Two-Dimensional Compact Variational Mode Decomposition. *Journal of Mathematical Imaging and Vision* 2017; **58**: 294–320.

5 Ling X. Fluctuation Law and Cause Analysis of Hog Spot Price in China. *Hans Journal of Agricultural Science* 2021; **11**(**8**): 755–765.

6 Yang K, Huang YP, Tian FP. Multiscale Co-Movement between Agricultural Futures Market and Other Financial Markets in China. *Systems Engineering Theory Practice* 2022; **42**(**5**): 1172–1184.

7 Liu YH, Liu H, Shang JP. Research on Price Prediction of Agricultural Futures Using Deep Learning Techniques. *Henan Science* 2024; **42**(**03**): 430–439.

8 Feng Y, Wang ZH, Li QY. Agricultural Product Price Prediction Model Based on CEEMD-LSTM under the Digital Background. *Prices Monthly* 2024; **7**: 1–8.

9 Niu D, Ji Z, Li W, *et al*. Research and Application of a Hybrid Model for Midterm Power Demand Forecasting Based on Secondary Decomposition and Interval Optimization. *Energy* 2021; **234**: 121145.

10 Liu H, Mi X, Li Y. Smart Multi-Step Deep Learning Model for Wind Speed Forecasting Based on Variational Mode Decomposition, Singular Spectrum Analysis, LSTM Network and ELM. *Energy Conversion and Management* 2018; **159**: 54–64.

11 Yan ZY, Mu GN. UHF Financial Time Series Predicting Based on CEEMDAN-VMD-LSTM. *Computer Era* 2023; **5**: 102–108.

12 Zhang X. Carbon Emission Rights Trading Price Combined Forecasting in China-Analysis Based Quadratic Decomposition and Machine Learning. *Price Theory Practice* 2023; **9**: 142–145.

13 Zhu JX, Zhang SY, Zhang T. Air Quality Predication Based on Two-Layer Decomposition and Improved Sand Cat Swarm Optimization. *Foreign Electronic Measurement Technology* 2024; **43**: 190–200.

14 Fan C. Outpatient Volume Prediction Model Based on CEEMDAN-VMD-SSA-LSTM. *Microcomputer Application* 2024; **40**: 214–218.