

## Article

# Can Denoising Enhance Prediction Accuracy of Learning Models? A Case of Wavelet Decomposition Approach

C. Tamilselvi <sup>1</sup>, Md Yeasin <sup>2</sup>, Ranjit Kumar Paul <sup>2,\*</sup> and Amrit Kumar Paul <sup>2</sup>

<sup>1</sup> The Graduate School, ICAR-Indian Agricultural Research Institute, New Delhi 110012, India; tamilselvi\_11991@iari.res.in

<sup>2</sup> ICAR-Indian Agricultural Statistics Research Institute, New Delhi 110012, India; md.yeasin@icar.gov.in (M.Y.); amrit.paul@icar.gov.in (A.K.P.)

\* Correspondence: ranjit.paul@icar.gov.in or ranjitstat@gmail.com

**Abstract:** Denoising is an integral part of the data pre-processing pipeline that often works in conjunction with model development for enhancing the quality of data, improving model accuracy, preventing overfitting, and contributing to the overall robustness of predictive models. Algorithms based on a combination of wavelet with deep learning, machine learning, and stochastic model have been proposed. The denoised series are fitted with various benchmark models, including long short-term memory (LSTM), support vector regression (SVR), artificial neural network (ANN), and autoregressive integrated moving average (ARIMA) models. The effectiveness of a wavelet-based denoising approach was investigated on monthly wholesale price data for three major spices (turmeric, coriander, and cumin) for various markets in India. The predictive performance of these models is assessed using root mean square error (RMSE), mean absolute percentage error (MAPE), and mean absolute error (MAE). The wavelet LSTM model with Haar filter at level 6 emerged as a robust choice for accurate price predictions across all spices. It was found that the wavelet LSTM model had a significant gain in accuracy than the LSTM model by more than 30% across all accuracy metrics. The results clearly highlighted the efficacy of a wavelet-based denoising approach in enhancing the accuracy of price forecasting.

**Keywords:** accuracy metrics; denoising; price forecasting; machine learning; LSTM; wavelet decomposition



**Citation:** Tamilselvi, C.; Yeasin, M.; Paul, R.K.; Paul, A.K. Can Denoising Enhance Prediction Accuracy of Learning Models? A Case of Wavelet Decomposition Approach. *Forecasting* **2024**, *6*, 81–99. <https://doi.org/10.3390/forecast6010005>

Academic Editor: Michał Rubaszek

Received: 17 November 2023

Revised: 11 January 2024

Accepted: 12 January 2024

Published: 16 January 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Price forecasting has become a crucial tool to navigate the changes in market dynamics. It empowers stakeholders, including farmers, traders, and businesses, to make well-informed decisions in response to these price fluctuations [1]. With accurate forecasting, one can anticipate future market trends and align production, procurement, and trading strategies accordingly. The evolution of statistical and machine learning techniques has significantly enhanced the precision of forecasting prices in various fields such as finance, agriculture, economics, and business [2]. The learning process and the effectiveness of these techniques are compromised by redundant, noisy, or unreliable information. Assuming the data adhere to a systematic pattern with random noise, a successful denoising algorithm facilitates a profound grasp of the data generation process, resulting in more accurate forecasts [3]. The wavelet transform method stands out as a prospective signal processing technique, offering simultaneous analysis in both the time domain and frequency domain [4]. This transformation enhances the forecasting model's capacity by capturing valuable information across multiple resolution levels.

In this paper, we explore the efficiency of wavelet-based denoising techniques on the time series data of price of important spices. Spices are natural plant substances that enhance the flavor, aroma, and color of food and drinks. They hold a rich history

in culinary, medicinal, and cultural traditions. Common spices like cinnamon, cumin, paprika, turmeric, cloves, and black pepper offer unique tastes and health benefits, with anti-inflammatory, antioxidant, and antimicrobial properties [5]. India has maintained a renowned status as the “land of spices” for centuries, with the scents and tastes of a diverse array of spices profoundly influencing its culinary heritage. The spice market in India is of great importance for local consumption and serves as a significant contributor to the country’s export industry [6]. According to data from the Indian Brand Equity Foundation (IBEF), India occupies the top position as the producer, consumer, and exporter of spices, with a remarkable production of 10.87 million tonnes in the 2021–2022 period. The surge in demand for spices from the food and beverages (F&B) sector, the widespread application of spices for medicinal uses, government-driven initiatives, and the promotion of sustainable sourcing are the leading catalysts behind the expansion of the India Spice Market [7]. There has been a substantial increase of nearly 40 percent in the wholesale prices of spices at the Agricultural Produce Market Committee (APMC), Vashi. This rise in prices is indicative of significant changes in the market dynamics. So, this dataset holds significant potential for examining the impact of wavelet denoising techniques.

To assess the effectiveness of wavelet-based denoising, we employed predictive models, including autoregressive integrated moving average (ARIMA), artificial neural networks (ANN), support vector regression (SVR), and long short-term memory (LSTM). Each decomposed component, obtained through wavelet transforms, was subjected to these benchmark models, with hyperparameter optimization performed to fine-tune model performance. The level of decomposition in wavelet-based denoising acts as a critical parameter, influencing the trade-off between capturing intricate details and avoiding noise. The effectiveness of this technique hinges on finding the optimal level that enhances predictive accuracy by selectively filtering out noise while preserving the essential characteristics of the signal. This study not only explores the broader effectiveness of wavelet-based denoising but also accentuates the crucial role of identifying the optimal level of decomposition for this dataset.

The rest of this paper is arranged as follows. Section 2 provides a comprehensive review of the existing literature, and the current research regarding wavelet-based hybrid models for price forecasting. The algorithms employed in the proposed forecasting model are explained in Section 3. Subsequently, Section 4 deals with the experimental analysis of proposed models and results. Finally, Section 5 deals with the results of the real dataset and their conclusions along with references.

## 2. Background

In this section, we begin with a brief review on selected works that have proposed benchmark forecasting techniques and hybrid models integrating wavelet analysis for the purpose of price forecasting. Extensive research has been conducted in the realm of time series forecasting, leading to the proposal and evaluation of numerous modeling techniques [8,9]. The autoregressive integrated moving average (ARIMA) methodology has emerged as the most widely employed linear technique in time series analysis [10]. Mao et al. [11] assessed the price fluctuations of vegetables during COVID-19 in 2020. They employed a web-crawling technique to gather price data for three distinct categories of vegetables, i.e., leafy vegetables, root vegetables, and solanaceous fruits. Subsequently, an ARIMA model was applied for forecasting the prices in the short-term. However, due to its basic assumption of linearity, ARIMA fails to capture the volatility changes that signify the intricacies of time series data.

Machine learning has experienced groundbreaking developments in recent decades, particularly in the field of intelligent prediction technology. The capacity of machine learning algorithms to model the complex, non-linear associations between variables, provide a valuable tool for identifying patterns that conventional statistical methods may struggle to detect [12]. The most widely used machine learning techniques are artificial neural networks (ANN), support vector regression (SVR), random forest (RF), decision trees,

k-nearest neighbors (KNN), and gradient boosting methods (e.g., XGBoost), etc. Mahto et al. [13] utilized ANN for predicting the seed prices of sunflower and soyabean from Akola market, Maharashtra and Kadari market, and Andhra Pradesh, respectively. Astudillo et al. [14] investigated the potential of SVR with external recurrences to forecast copper closing prices at the London Metal Exchange for various future time horizons, including 5, 10, 15, 20, and 30 days. Jeong et al. [15] applied an SVR model to predict the onion prices in South Korea. Zhang et al. [16] forecasted corn, bean, and grain products using various intelligent models such as ANN, SVR, and extreme machine learning (ELM). Paul et al. [17] attempted to examine the efficiency of different machine learning algorithms such as generalized neural network (GRNN), SVR, RF, and gradient boosting machine (GBM) for predicting the wholesale price of Brinjal across 17 primary markets in Odisha, India.

Deep learning, a subset of machine learning, showcases cutting-edge performance in forecasting intricate time series data [18]. Enhanced understanding of the present often stems from past information, which lacks in conventional ANN. Recurrent neural networks (RNNs) utilize a feedback loop to retain and incorporate previous information, allowing for more precise predictions and decisions [19,20]. Different approaches relying on RNN have been implemented to forecast the prices of agricultural products. Then, long short-term memory (LSTM) [21] was first introduced by Hochreiter and Schmidhuber in 1997, as an extension of RNN that solves the problem of exploding and vanishing gradients more effectively than the conventional RNN models. Chen et al. [22] developed a web-based automated system for predicting agriculture commodity prices. Their study revealed that the LSTM model demonstrated lower error rates compared to other machine learning models, particularly when handling extensive historical data from Malaysia. Gu et al. [23] proposed dual input attention long short-term memory (DIA-LSTM) for the efficient prediction of agricultural commodity prices like cabbage and radish in the South Korean market.

Cakici et al. [24] found that equity anomalies can predict aggregate market returns, as their predictability, if any, is confined to specific anomalies and methodological choices. Dong et al. [25] applied an array of shrinkage methods, incorporating machine learning, forecast combination, and dimension reduction, to effectively capture predictive signals in a high-dimensional environment. Incorporating wavelets as a preprocessing tool has provided a new perspective on the analysis of data characterized by noise [26]. The denoising process, one of the earliest applications of the discrete wavelet transform (DWT), is designed to eliminate small segments of the signal identified as noise [27]. Wavelet-based approaches are commonly used as versatile tools for both the analysis and synthesis of multicomponent signals, particularly for tasks such as noise removal. Removing noise from the signal not only expedites the processing of analysis but also enhances the efficiency of the model [28]. Paul et al. [29] conducted a comparative analysis of various wavelet functions, including Haar, Daubechies (D4), D6, and LA8, to assess their performance in forecasting the prices of tomatoes. Shabri et al. [30] suggested that the augmentation of the wavelet technique in the SVM model results in improved forecasting performance compared to classical SVM and demonstrated its superiority over ANN. Garai et al. [31] presented a methodology that combines stochastic and machine learning models, alongside the utilization of wavelet analysis, for the prediction of agricultural prices.

Chen et al. [32] used the wavelet analysis as a denoising tool along with LSTM for the prediction the prices of Agricultural products in China. Liang et al. [33] put forth a novel threshold-denoising function with the aim of decreasing the distortion levels in signal reconstruction. The outcomes unequivocally demonstrated that the proposed function, when combined with the LSTM model, surpassed the performance of other conventional models. Zhou et al. [34] evaluated the deep neural network (DNN) models for equity-premium forecasting and compared them with ordinary least squares (OLS) and historical average (HA) models. They found that augmenting DNN models with 14 additional variables enhanced their forecasting performance, showcasing their adaptability and superiority in equity-premium prediction. Jaseena and Kovoor [35] utilized wavelet transform to decompose

wind speed data into high and low frequency subseries. Subsequently, they forecasted the low and high frequency subseries using LSTM and SVR, respectively. Peng et al. [36] made an attempt to predict the stock movement for the next 11 days (medium-term) by using multiresolution wavelet reconstruction and RNN. Singla et al. [37] proposed an ensemble model to forecast the 24-h ahead solar GHI for the location of Ahmadabad, Gujarat, India, by combining wavelet and BiLSTM networks (WT-BiLSTM (CF)). They also compared the forecasting performance of WT-BiLSTM (CF) with unidirectional LSTM, unidirectional GRU, BiLSTM, and wavelet-based BiLSTM models. Yeasin and Paul [38] proposed an ensemble model consisting of 13 forecasting models including five deep learning, five machine learning, and three stochastic models for forecasting vegetable prices in India. Liang et al. [39] constructed an Internet-based consumer price index (ICPI) from Baidu and Google searches using principal component analysis. Transfer entropy quantifies information flow between online behavior and future markets. The GWO-CNN-LSTM model, incorporating ICPI and transfer entropy, forecasts daily prices of corn, soybean, PVC, egg, and rebar futures. Cai et al. [40] had proposed a variational mode decomposition method along with deep learning model to forecast the hourly PM2.5 concentration. Deng et al. [41] used multivariate empirical mode decomposition along with LSTM while dealing with forecasting of multi-step-ahead stock prices. Lin et al. [42] decomposed the crude oil price data using wavelet transform and fed into a BiLSTM-Attention-CNN model for predicting the future price.

However, there exists a very low number of studies related to the gain in prediction accuracy of the models based on wavelet decomposed series compared to usual benchmark models. The present study illustrates this in detail, with application of machine learning, deep learning, and stochastic models in conjunction with wavelet denoising.

### 3. Methodology

#### 3.1. ARIMA

ARIMA is often considered the foundational cornerstone in the realm of predictive modeling, which was given by Box and Jenkins in 1970 [43]. The ARIMA modeling process revolves around four key steps. These are the identification of the model, estimation of the parameters, diagnostic checking, and forecasting. This model class relies on the fundamental assumption that time series should exhibit stationarity and the future values of a time series result from a linear combination of previous observations and white noise components. Then, ARMA ( $p, q$ ) can be written as:

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + \alpha_p y_{t-p} - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \dots - \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (1)$$

or equivalently (using backshift operator  $B$ ) by:

$$\alpha(B)y_t = \theta(B)\varepsilon_t \quad (2)$$

where:

$$\alpha(B) = 1 - \alpha_1 B - \alpha_2 B^2 - \dots - \alpha_p B^p \quad (3)$$

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_p B^p \quad (4)$$

ARMA models require stationary data, but real-world data in various fields is often non-stationary. By applying differencing, data can be made stationary, leading to the development of ARIMA models [39], represented as ARIMA ( $p, d, q$ ). Then, the model is written as:

$$\alpha(B)(1 - B)^d y_t = \theta(B)\varepsilon_t \quad (5)$$

where  $\varepsilon_t$  is identically and independently distributed (IID) as  $N(0, \sigma^2)$ . The parameters ranging from  $\alpha_1$  to  $\alpha_p$  represent the autoregressive (AR) coefficients, while  $\theta_1$  to  $\theta_q$  denote the moving average (MA) coefficients.

### 3.2. Artificial Neural Networks

ANN models are built on the concept of neurons that interact by transmitting signals through weighted connections to one another [44]. These networks emulate the way biological neurons exchange information in the brain [45]. In ANN models, each neuron is connected to all preceding neurons and passes the layers through links. In the input layer, each input value is treated as a neuron. To enable the success of an ANN, all input values are initially weighted, and the weighted values are then processed within the hidden layers, where each neuron generates output values. This process allows ANNs to learn and make predictions or classifications based on the weighted inputs and the network’s architecture. The output value of the neuron can be represented by the formula:

$$O = \theta \left( \sum_{j=1}^n w_j x_j + w_0 \right) \tag{6}$$

where  $O$  and  $x_j$  represent the output and input values to the neuron, respectively,  $w_0$  is the bias term,  $w_j$ ’s is the weightage related to each input, and  $\theta$  represents the activation function. Figure 1 explained the basic architecture of ANN model. The flexibility of an artificial neural network (ANN) lies in its adaptable architecture, allowing easy adjustments to the number of layers and neurons in each layer. Unlike certain modeling approaches, ANN does not demand prior assumptions like data stationarity during the model-building process. As a result, the network’s structure is primarily guided by the unique characteristics of the data being analyzed.

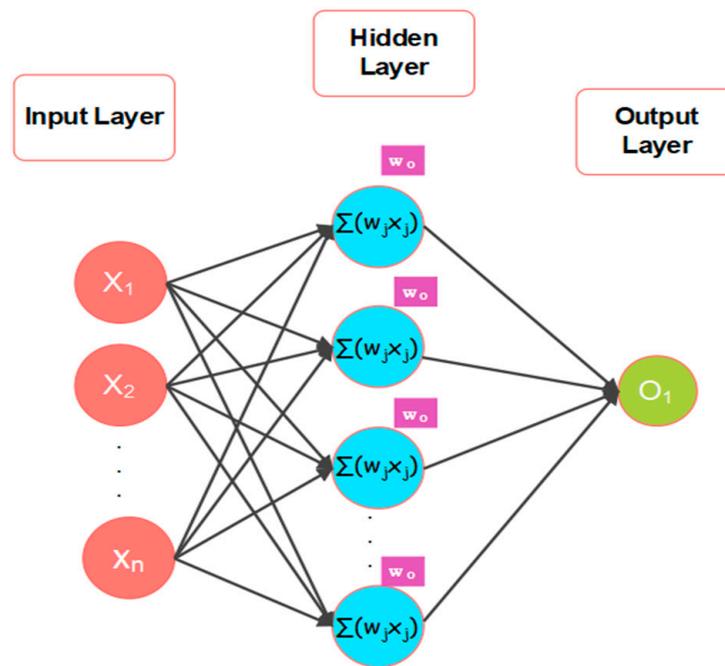


Figure 1. Basic architecture of ANN (Source: own illustration).

### 3.3. Support Vector Regression (SVR)

SVR is centered on the principles of statistical learning theory and the rule of structural risk minimization [41]. It has proven to be highly effective for modeling and predicting non-linear systems by minimizing the upper limit of generalization error [16]. The SVR model can be expressed mathematically as follows:

$$f(x) = (z.\mathcal{O}(x)) + b \text{ with } w \in \mathbb{R}^N, b \in \mathbb{R} \tag{7}$$

where  $z$  stands for the weight vector, while  $\mathcal{O}(x)$  denotes the high-dimensional feature space resulting from a non-linear transformation of the input space and  $b$  is the bias. The estimation of the weight vector  $z$  and bias  $b$  involves the minimization of the regularized risk function:

$$R(C) = \frac{1}{2}|z|^2 + C\frac{1}{p} \sum_{i=1}^t L_\epsilon(f(x_i), y_i) \tag{8}$$

Within the regularized risk function, the initial term  $\frac{1}{2}|z|^2$  serves as the regularization component, regulating the model's capacity. The second term,  $\frac{1}{p} \sum_{i=1}^t L_\epsilon(f(x_i), y_i)$ , represents the empirical error. The constant 'C' plays a crucial role as the regularization parameter, influencing the balance between empirical risk and regularization within the model. In the case of the SVR model,  $\epsilon$ -insensitivity loss function is used and can be expressed as:

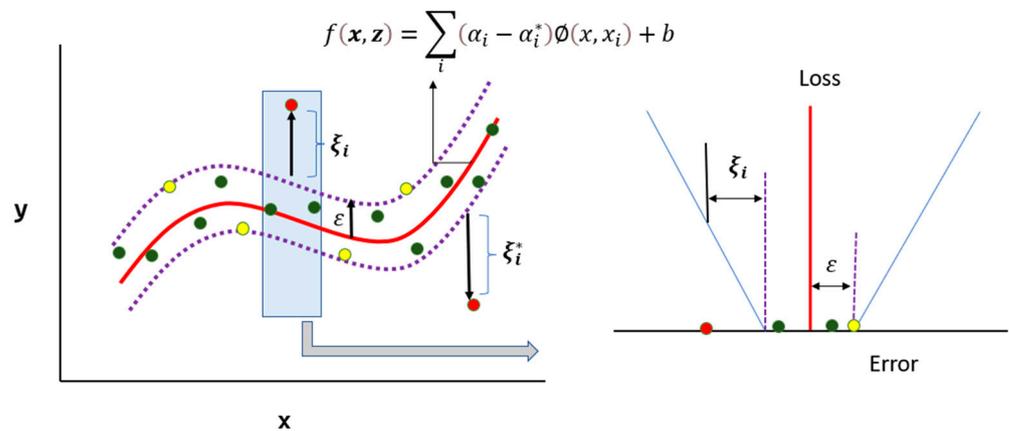
$$L_\epsilon(f(x_i), y_i) = \begin{cases} |f(x_i) - y_i| - \epsilon, & |f(x_i) - y_i| \geq \epsilon \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where  $\epsilon$  denotes the tube size. Minimizing Equation (7) can be expressed as solving the following the primal optimization problem:

$\xi_i$  and  $\xi_i^*$  are positive slack variables that quantify the gap between actual values and the respective boundary values within the  $\epsilon$  tube. Equation (7) can also be written as:

$$f(x, z) = f(x, \alpha, \alpha^*) = \sum_i (\alpha_i - \alpha_i^*) \mathcal{O}(x, x_i) + b \tag{10}$$

The kernel function  $\mathcal{O}(\cdot)$  is employed to transform the non-linear dataset into a higher-dimensional feature space, where it is assumed to be linear. For a function to be considered a kernel function, it must satisfy Mercer's condition, as outlined by Mercer in 1909. The data calibration of SVR model are presented in the Figure 2.



**Figure 2.** Data calibration in the support vector regression model (Source: own illustration).

### 3.4. Long Short Term Memory

The LSTM incorporates a cell state that traverses the network, allowing information to be added or removed through gating mechanisms. Thus, the LSTM model is structured around three vital gate mechanisms: the forget gate, input gate, and output gate [24]. These gate structures serve as the fundamental building blocks of the LSTM, empowering it to effectively process and forecast important information, even when dealing with long-term time intervals and temporal delays in time series data [46]. The architecture of the single LSTM cell is presented in Figure 3.

**Forget Gate:** It involves a sigmoid layer that decides which information needs to be discarded from the cell state. It accomplishes this by merging the current input  $x_t$  with the output of the preceding LSTM unit,  $h_{t-1}$ .

$$f_t = \sigma(W^f x_t + U^f h_{t-1} + b^f) \tag{11}$$

**Input Gate:** It consists of two components. The first layer is a sigmoid layer, responsible for making decision regarding value updates. The following layer is a tanh layer, generating a vector of new candidate values, denoted as  $\tilde{C}_t$ , which can be integrated into the cell state.

$$i_t = \sigma(W^i x_t + U^i h_{t-1} + b^i) \tag{12}$$

$$\tilde{C}_t = \tanh(W^c x_t + U^c h_{t-1} + b^c) \tag{13}$$

Then, we need to update the old cell state  $C_{t-1}$  into new state  $C_t$ . Update includes forgetting the old information by multiply the  $f_t$  with  $C_{t-1}$  and adding the new information by multiplying  $i_t$  with  $\tilde{C}_t$ .

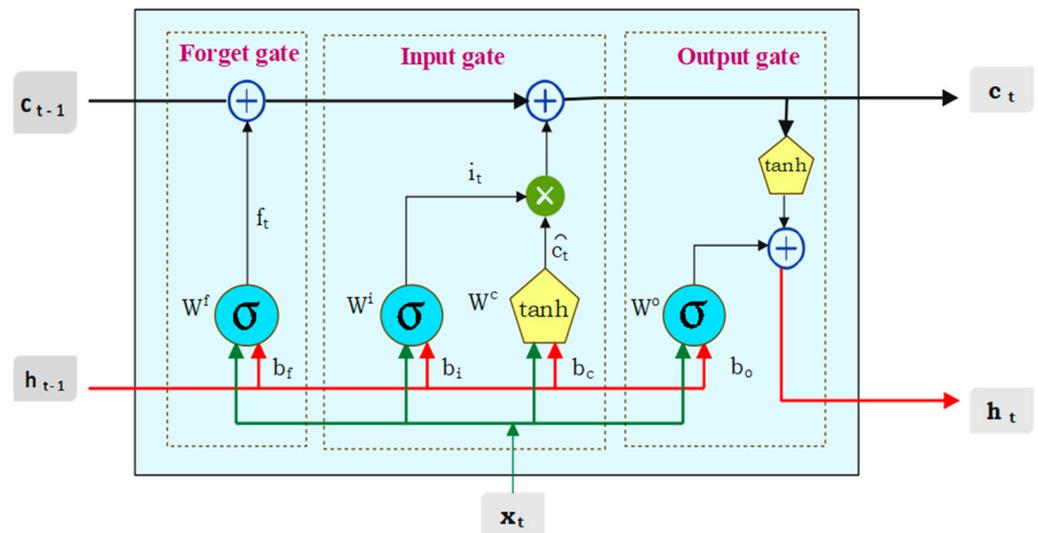
$$C_t = (f_t \times C_{t-1} + i_t \times \tilde{C}_t) \tag{14}$$

**Output gate:** The output gate is instrumental in defining the value of the next hidden state  $h_t$ , which captures information from prior inputs. It initiates by taking both the current state and the previous hidden state values into a sigmoid function. Subsequently, the new cell state  $C_t$ , which is derived from the existing cell state, undergoes additional processing through the tanh function.

$$O_t = \sigma(W^o x_t + U^o h_{t-1} + b^o) \tag{15}$$

$$h_t = O_t \times \tanh(C_t) \tag{16}$$

where  $W^*$  and  $U^*$  represent the weight of current input  $x_t$  and output of previous LSTM unit  $h_{t-1}$ , respectively, while  $b^*$  is bias vector and  $*$  represents the corresponding gates.



**Figure 3.** The architecture of the single LSTM cell (Source: own illustration).

### 3.5. Wavelet-Based Denoising

The main concept behind using wavelet transforms and their inverses is that it allows for a multi-resolution analysis of a signal. Different wavelet coefficients represent different frequency components of the signal at various resolutions or scales. This is particularly

useful in applications like signal processing, image compression, and denoising. The denoising process can be defined as the removal of noise while preserving and without distorting the quality of the processed signal or image. Wavelet transform helps in denoising by decomposing the signal into different levels. The general procedure of noise removal takes place by three main steps. The initial step involves decomposition, wherein the noisy data undergoes transformation into an orthogonal domain. The second step entails various processing operations on the obtained coefficients. The final step involves the data being transformed back into its original domain through a reconstruction process. The time series data are decomposed by passing through the high pass and low pass filter. The high pass filter produces the high-frequency detail components, while the low-pass filter generates the low-frequency approximate components of the signal. Both components of the filter have an equal bandwidth. Subsequently, the next step involves iterating the aforementioned process for the low-frequency component to acquire the two decomposed components.

Step 1: The time series  $x(t)$  under wavelet transform is expressed as:

$$x(t) = \sum_j a_{i,j} \omega_{i,j}(t) + \sum_j d_{i,j} \psi_{i,j}(t) + \sum_j d_{i-1,j} \psi_{i,j}(t) + \dots + \sum_j d_{1,j} \psi_{i,j}(t) \quad (17)$$

$$s_i(t) = \sum_k s_{i,j} \omega_{i,j}(t) \quad (18)$$

$$D_i(t) = \sum_k d_{i,j} \psi_{i,j}(t) \quad (19)$$

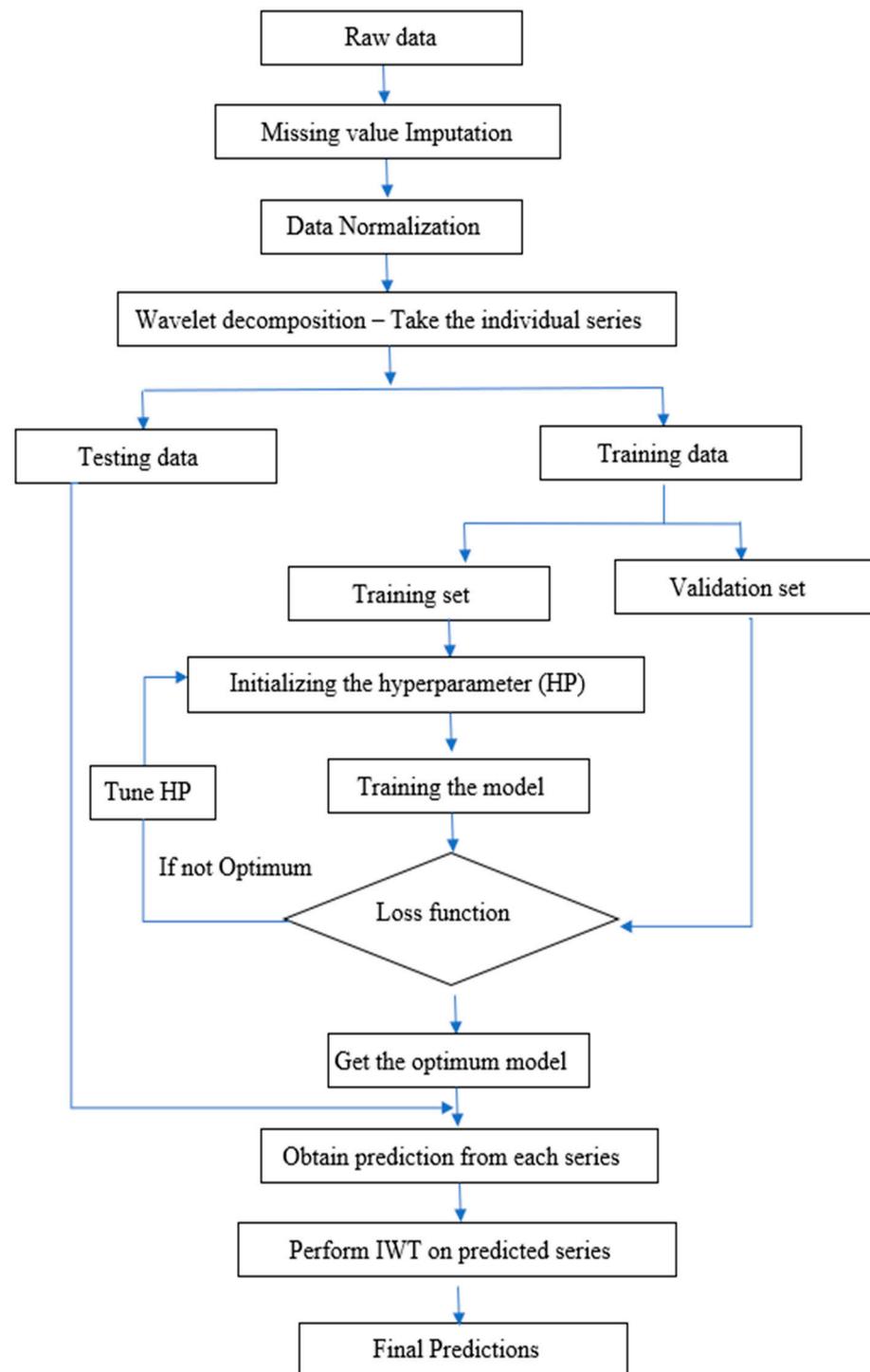
The original signal can be represented as the sum of detailed and approximate components as follows:

$$x(t) = \sum_{j=1}^i D_i(t) + S_i(t) \quad (20)$$

where  $J$  denotes the level of decomposition, ranging from 1 to  $i$ ;  $j$  is the translation parameter;  $\psi_{i,j}(t)$  and  $\omega_{i,j}(t)$  represent the parent wavelet pairs;  $s_{i,j}$  denotes the scaling coefficient of father wavelet  $\omega_{i,j}(t)$  and  $d_{i,j}$  is the detail coefficient of mother wavelet  $\psi_{i,j}(t)$ ;  $D_i(t)$  represent the high-frequency component signal; and  $S_i(t)$  is the low-frequency component signal.

Step 2: Each decomposed component separately is fed into the benchmark models. Optimal hyperparameters for the model are chosen by examining its performance on a distinct validation set. After fine-tuning, the model undergoes the evaluation on test data by subjecting each series to the LSTM model, leading to the accurate prediction of price series.

Step 3: The inverse wavelet transform (IWT) is the mathematical operation that reconstructs a signal from its wavelet coefficients. The prediction obtained from each series coefficient is subjected to IWT for wavelet reconstruction. The outlook of analysis performed is presented in Figure 4.



**Figure 4.** Flowchart of the proposed algorithm (Source: own illustration).

## 4. Result and Discussion

### 4.1. Data Description

The current study focused on the monthly wholesale price data of three important spices (i.e., turmeric, coriander, and cumin) from different markets of India, where data series were collected from the AGMARKNET Portal <http://agmarknet.gov.in/> (accessed on 3 July 2023) for the period of January 2010 to December 2022. Turmeric data were gathered from 18 distinct markets, while coriander and cumin data were obtained from 27 markets each. The choice of markets and commodities was determined by their significant market share and their representative characteristics within their respective categories. The overall

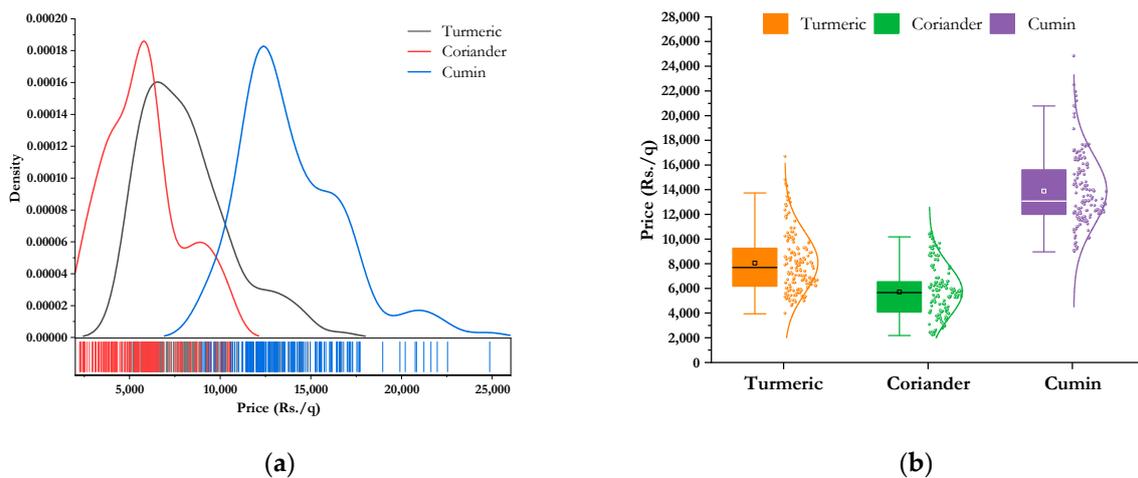
price for each commodity was calculated as the weighted average price across all markets for that specific commodity. The weight assigned to each market was determined by the inverse of the arrival quantity:

$$aggregated\ price_t = \frac{\sum_{i=1}^N quantity_{i,t} \times price_{i,t}}{\sum_{i=1}^N quantity_{i,t}} \tag{21}$$

where  $N$  is the number of markets in that commodity and  $t$  is time period. The price patterns of three spices are illustrated in Figure 4. The dataset consists of a total of 156 observations across all commodities. The final 29 observations were set aside for testing and post-sample prediction, leaving 127 observations for model development in each case. Table 1 represents the descriptive statistics of different commodities. Table 1 indicates that all the prices series were non-normal and have high kurtosis. The Jarque–Bera test and Shapiro–Wilk’s test are statistical methods employed to evaluate whether a dataset follows normal distribution. The null hypothesis of both the tests are that data follow a normal distribution. It is inferred from the table that all the three commodities are non-normal. The variability in price is less for cumin as compared to other two spices. The kernel density and the box-plot of different price series are plotted in Figure 5. Figure 5 also demonstrates high kurtosis and positive skew.

**Table 1.** The descriptive statistics of the price of spices.

Descriptive Statistics	Turmeric (Rs/qtl)	Coriander (Rs/qtl)	Cumin (Rs/qtl)
Mean	7846.2	5093.23	13,756.65
Median	7236.83	4832.49	12,844.92
Minimum	3745.76	2265.09	8826.91
Maximum	16,119.79	9484.08	24,672.25
Standard Deviation	2633.53	1753.48	2841.91
Kurtosis	4.13	2.71	4.22
Skewness	1.29	0.64	1.02
C.V. (%)	33.56	34.43	20.66
Jarque-Bera test	51.39 (<0.01)	6.50 (<0.01)	36.76 (<0.01)
Shapiro-Wilk’s test	0.87 (<0.01)	0.96 (<0.01)	0.93 (<0.01)



**Figure 5.** (a) Kernel density and (b) box-plot of different spices.

**4.2. Application of Benchmark Models**

The primary contenders in the field of predictive modelling start with ARIMA. The process initiates with checking the stationarity of the underlying data through an augmented Dickey–Fuller (ADF) test and the results are outlined in Table 2. All the three series

exhibited non-stationarity which is converted into stationarity by differencing. The order of the model was tentatively identified using the autocorrelation function (ACF) and partial autocorrelation function (PACF). Ultimately, the selection of the best model for individual commodity prices was performed based on the minimum values of Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC).

**Table 2.** The stationarity test results of price series.

Data	Augmented Dickey–Fuller Test		Kwiatkowski–Phillips–Schmidt–Shin		Remarks
	Statistic	<i>p</i> -Value	Statistic	<i>p</i> -Value	
Turmeric (Rs/qtl)	−2.87	0.21	0.47	0.04	Non-Stationary
Coriander (Rs/qtl)	−2.04	0.55	0.75	0.01	Non-Stationary
Cumin (Rs/qtl)	−1.61	0.73	1.08	0.01	Non-Stationary

Before proceeding for the application of machine learning and deep learning techniques, the presence of volatility clustering was tested by means of the autoregressive conditional heteroscedastic-Lagrange multiplier (ARCH-LM) test. It is found that for all three series, the test is not significant indicating absence of any volatility clustering.

The individual price series was subjected to ANN, SVR and LSTM model. In the process of developing an ANN, selection of input lag value is a crucial step. Here, the input lag value was selected as 12 with the help of ACF. The ANN model used in this study is the standard two-layer feed forward network. The efficiency of the model can be improved by finding the best set of hyperparameter values for the training dataset. The hyperparametric search space for ANN were the number of nodes per layer = {6, 7, 8, 9, 10, 11, 12}, number of epochs = {10 to 100 with step 10}, and batch size = {16, 32, 64, 128}. The hyperparameters were optimized by using grid search method.

For building the SVR model, the actual value  $y_t$  was assumed to be a function of its previous lag values. The model uses an RBF kernel, and the hyperparameters  $C$  (Regularization parameter),  $\epsilon$  (epsilon), and  $\gamma$  (kernel width) were optimized using grid search over the following ranges:  $C = \{1, 10, \dots, 1000 \text{ with step } 10\}$ ,  $\epsilon = \{0.01, 0.02, \dots, 0.30\}$ ,  $\gamma = \{3, 4, \dots, 15\}$ . In order to build LSTM model, lag values were established at 12. The Hyperparameter search space for LSTM were the number of LSTM units = {32, 64, 128}, number of epochs = {50, 100, 200}, and batch size = {16, 32, 64, 128}.

The best hyperparameters are selected based on the performance of the model on a hold-out validation set. Once the hyperparameters have been tuned, the model is evaluated on a test set to assess its performance on unseen data. The model's performance on the test data can be evaluated using a variety of metrics, such as the mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE).

Table 3 provides a clear overview of various accuracy measures of different models, including LSTM, SVR, ANN, and ARIMA, across the crops turmeric, coriander, and cumin. For turmeric, LSTM demonstrated a substantial 21.4% reduction in RMSE compared to SVR. In the case of coriander, LSTM achieved an impressive 40.7% reduction in RMSE relative to SVR. For cumin, LSTM showcased a significant 37.3% decrease in RMSE compared to SVR. For all the three crops (turmeric, coriander, cumin), the LSTM model consistently outperformed other models in terms of RMSE, MAPE, and MAE, suggesting they are generally better at capturing the underlying patterns in the data.

**Table 3.** Accuracy measures for different models for test data (bold denotes the lowest of each accuracy metrics).

Crop	Model	RMSE	MAPE	MAE
Turmeric	LSTM	<b>1683.450</b>	<b>0.153</b>	<b>1157.435</b>
	SVR	2141.529	0.182	1694.249
	ANN	2180.729	0.221	1612.335
	ARIMA	1910.800	0.120	1122.188
Coriander	LSTM	<b>1283.670</b>	<b>0.155</b>	<b>1095.620</b>
	SVR	1754.718	0.181	1244.712
	ANN	1637.166	0.161	1207.808
	ARIMA	2930.291	0.247	2216.242
Cumin	LSTM	<b>2374.435</b>	<b>0.102</b>	<b>1725.857</b>
	SVR	3790.348	0.127	2476.648
	ANN	3963.81	0.210	3767.492
	ARIMA	5283.088	0.239	3523.121

#### 4.3. Denoising Using Wavelet

While there are numerous methods available for predicting prices series, they do not consistently meet expected performance levels, as each method comes with its own set of advantages and disadvantages. Denoising helps us to filter out this extraneous noise, enabling the model to focus on the relevant patterns and relationships in the data. So, the original series were denoised using wavelet transform before applying the model. The Haar wavelet filter was used for wavelet transformation. The maximum ( $J_1 = \log_2 N$ ) and minimum ( $J_0 = \log N$ ) levels of decomposition for price series were 7 and 5, respectively. So, each series was decomposed at all three levels and fed as input to ARIMA, ANN, SVR, and LSTM models. The hyperparameters of the models were tuned using a grid search over a range of possible values as mentioned before. Tables 4–6 represent the accuracy measure of various models on the test data with different levels of decomposition (bold denotes the lowest of each accuracy metrics). Here, H5, H6, and H7 represent the wavelet decomposition with the Haar filter and level of decomposition as 5, 6, and 7, respectively.

**Table 4.** Accuracy measures of different models on the decomposed test series (turmeric).

Turmeric Prices	Decomposition Level	RMSE	MAPE	MAE
Wavelet LSTM	WLSTM-H5	1211.138	0.076	683.887
	WLSTM-H6	<b>1090.606</b>	<b>0.069</b>	<b>662.225</b>
	WLSTM-H7	1143.348	0.076	695.462
Wavelet SVR	WSVR-H5	1489.157	<b>0.120</b>	<b>1054.334</b>
	WSVR-H6	1513.634	0.123	1074.154
	WSVR-H7	<b>1458.960</b>	0.159	1234.963
Wavelet ANN	WANN-H5	1939.706	0.212	1612.938
	WANN-H6	1603.803	<b>0.171</b>	1364.857
	WANN-H7	<b>1522.334</b>	0.179	<b>1314.672</b>
Wavelet ARIMA	WARIMA-H5	<b>1221.804</b>	<b>0.092</b>	<b>802.592</b>
	WARIMA-H6	1257.048	0.097	839.862
	WARIMA-H7	1223.024	0.098	835.978

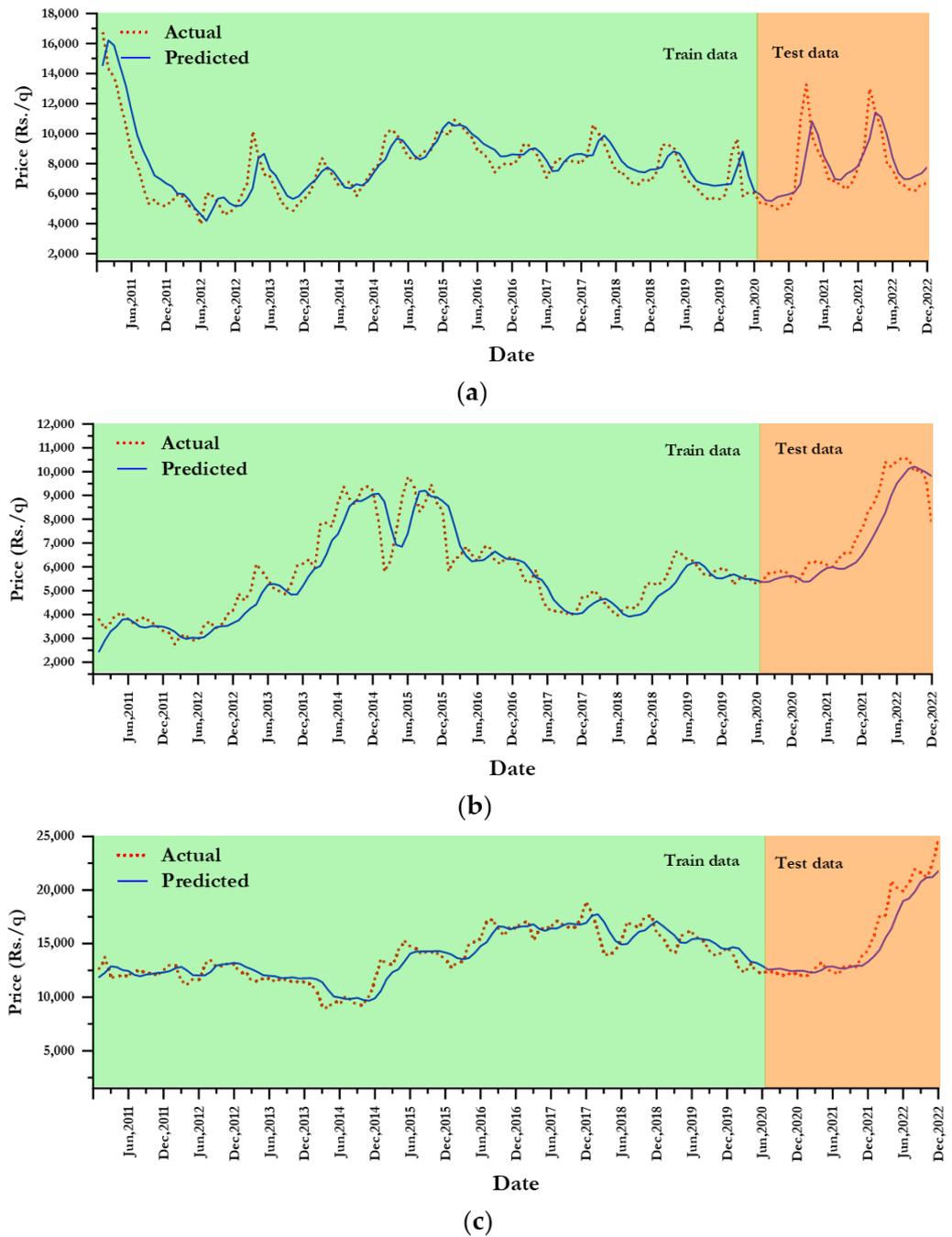
**Table 5.** Accuracy measures of different models on the decomposed test series (coriander).

Coriander Prices	Decomposition Level	RMSE	MAPE	MAE
<b>Wavelet LSTM</b>	WLSTM-H5	915.072	0.093	726.147
	WLSTM-H6	<b>739.661</b>	<b>0.066</b>	<b>523.609</b>
	WLSTM-H7	753.984	0.068	548.466
<b>Wavelet SVR</b>	WSVR-H5	1687.121	0.138	1133.342
	WSVR-H6	<b>1374.417</b>	<b>0.130</b>	<b>1130.952</b>
	WSVR-H7	2109.378	0.180	1562.385
<b>Wavelet ANN</b>	WANN-H5	1359.134	0.167	1133.214
	WANN-H6	<b>1009.808</b>	<b>0.117</b>	<b>780.252</b>
	WANN-H7	1292.371	0.154	1067.117
<b>Wavelet ARIMA</b>	WARIMA-H5	<b>2543.002</b>	<b>0.206</b>	<b>1872.235</b>
	WARIMA-H6	2638.109	0.216	1958.972
	WARIMA-H7	2714.987	0.224	2028.906

**Table 6.** Accuracy measures of different models on the decomposed test series (cumin).

Cumin Prices	Decomposition Level	RMSE	MAPE	MAE
<b>Wavelet LSTM</b>	WLSTM-H5	1988.514	0.083	1478.416
	WLSTM-H6	1676.699	0.070	1227.387
	WLSTM-H7	<b>1549.431</b>	<b>0.069</b>	<b>1182.181</b>
<b>Wavelet SVR</b>	WSVR-H5	2749.753	0.100	1915.539
	WSVR-H6	<b>2703.444</b>	<b>0.093</b>	<b>1814.939</b>
	WSVR-H7	2895.154	0.122	2220.628
<b>Wavelet ANN</b>	WANN-H5	4140.580	0.179	3216.344
	WANN-H6	<b>3447.018</b>	<b>0.189</b>	<b>3000.730</b>
	WANN-H7	3602.930	0.201	3179.977
<b>Wavelet ARIMA</b>	WARIMA-H5	<b>4986.682</b>	<b>0.171</b>	<b>2964.718</b>
	WARIMA-H6	5201.544	0.173	3275.703
	WARIMA-H7	5003.212	0.171	3053.424

For turmeric, decomposition level H6 was optimal for WLSTM, offering the highest accuracy across all metrics, whereas WSVR and WARIMA performed best at H7 and H5 decomposition levels, respectively. For coriander, decomposition level H6 exhibited the lowest RMSE, MAPE, and MAE, indicating better predictive accuracy compared to other levels across all models (WLSTM, WSVR, and WANN), except WARIMA. For cumin prices, the most effective decomposition level was H7 and H5 for WLSTM and WARIMA, respectively, presenting the lowest RMSE, MAPE, and MAE. Across all models in cumin, decomposition level H6 tended to consistently offer the best predictive accuracy, as it frequently displayed the lowest values for RMSE, MAPE, and MAE. The WLSTM model outperformed the other models (WSVR, WANN, and WARIMA) on all three datasets (turmeric, coriander, and cumin). The performance of the best fitted model, i.e., WLSTM, for both training and testing sets is presented in Figure 6.



**Figure 6.** Actual and predicted values of the best fitted model for price series of (a) turmeric, (b) coriander, and (c) cumin.

4.4. Impact of Denoising

Table 7 explains the influence of denoising in the current dataset. Among the different levels of decomposition, the optimum level of decomposition was selected based on the minimum values of accuracy measures and subsequently compared with a benchmark model. The percentage decrease (%↓) in error was calculated for finding the improvement achieved by the wavelet-based denoising model (WLSTM, WSVR, WANN, WARIMA) over their respective baseline models. In all three commodities, percentage of error reduction due to denoising was greater than 30% when compared to traditional LSTM. For turmeric, WLSTM achieved a significant 35.22% reduction in RMSE compared to LSTM. Coriander showed an even more substantial reduction of 42.38% in RMSE with WLSTM. Cumin also

exhibited 34.75% reduction in RMSE when using WLSTM. WSVR demonstrated superior performance compared to SVR, achieving substantial reductions in RMSE of 30.46%, 21.67%, and 28.68% for turmeric, coriander, and cumin, respectively. Similarly, wavelet-based ANN and ARIMA also had a significant reduction in error when compared to benchmark models.

**Table 7.** Comparison of wavelet-based denoising model with the traditional model.

	Turmeric			Coriander			Cumin		
	LSTM	WLSTM	%↓	LSTM	WLSTM	%↓	LSTM	WLSTM	%↓
RMSE	1683.45	1090.61	35.22	1283.67	739.66	42.38	2374.44	1549.43	34.75
MAPE	0.15	0.07	54.92	0.15	0.07	57.35	0.10	0.07	32.60
MAE	1157.44	662.23	42.79	1095.62	523.61	52.21	1725.86	1182.18	31.50
	SVR	WSVR	%↓	SVR	WSVR	%↓	SVR	WSVR	%↓
RMSE	2141.53	1489.16	30.46	1754.72	1374.42	21.67	3790.35	2703.44	28.68
MAPE	0.18	0.12	34.16	0.18	0.13	27.98	0.13	0.09	26.63
MAE	1694.25	1054.33	37.77	1244.71	1130.95	9.14	2476.65	1814.94	26.72
	ANN	WANN	%↓	ANN	WANN	%↓	ANN	WANN	%↓
RMSE	2180.73	1522.33	30.19	1637.17	1009.81	38.32	3963.81	3447.02	13.04
MAPE	0.22	0.18	19.04	0.16	0.12	27.27	0.21	0.19	10.07
MAE	1612.34	1314.67	18.46	1207.81	780.25	35.40	3767.49	3000.73	20.35
	ARIMA	WARIMA	%↓	ARIMA	WARIMA	%↓	ARIMA	WARIMA	%↓
RMSE	1910.80	1221.80	36.06	2930.29	2543.00	13.22	5283.09	4986.68	5.61
MAPE	0.12	0.09	23.52	0.25	0.21	16.43	0.24	0.17	28.44
MAE	1122.19	802.59	28.48	2216.24	1872.24	15.52	3523.12	2964.72	15.85

The technique for order preference by similarity to ideal solution (TOPSIS) has been applied for finding the best model with respect to each of the series under consideration. Equal weightage has been given to each of the performance measures, i.e., RMSE, MAPE, and MAE. The TOPSIS scores and ranks were computed for wavelet-based denoising models and the same is presented in Table 8. Table 8 indicates that the WLSTM occupied the top three ranks in all the price series. Moreover, the Diebold–Mariano (DM) test was applied to see the significant differences in predictive accuracy between the two models. The result of the DM test is reported in Table 9, which indicates the significant differences between the wavelet-based denoising model and the individual benchmark model. Overall, the results clearly demonstrated the effectiveness of wavelet-based denoising in reducing errors across the different spice crops and forecasting techniques.

**Table 8.** TOPSIS score and rank of wavelet-based denoising models.

Models	Decomposition Level	Turmeric		Coriander		Cumin	
		Score	Rank	Score	Rank	Score	Rank
Wavelet LSTM	WLSTM-H5	0.867	3	0.644	3	0.702	3
	WLSTM-H6	1	1	1	1	0.925	2
	WLSTM-H7	0.884	2	0.954	2	1	1
Wavelet SVR	WSVR-H5	0.372	7	0.256	8	0.431	5
	WSVR-H6	0.351	8	0.313	5	0.476	4
	WSVR-H7	0.23	9	0.104	9	0.318	6

Table 8. Cont.

Models	Decomposition Level	Turmeric		Coriander		Cumin	
		Score	Rank	Score	Rank	Score	Rank
Wavelet ANN	WANN-H5	0	12	0.271	7	0.076	9
	WANN-H6	0.15	11	0.521	4	0.136	7
	WANN-H7	0.171	10	0.31	6	0.114	8
Wavelet ARIMA	WARIMA-H5	0.671	4	0.031	10	0.06	10
	WARIMA-H6	0.613	6	0.013	11	0.045	12
	WARIMA-H7	0.616	5	0	12	0.055	11

Table 9. DM test result.

Commodity	Wavelet-Based Denoising Model	Benchmark Model	DM Test Statistic	<i>p</i> -Value
Turmeric	WLSTM	LSTM	−9.71	<0.01
	WSVR	SVR	−7.45	<0.01
	WANN	ANN	−4.56	<0.01
	WARIMA	ARIMA	−8.14	<0.01
Coriander	WLSTM	LSTM	−10.62	<0.01
	WSVR	SVR	−7.11	<0.01
	WANN	ANN	−6.87	<0.01
	WARIMA	ARIMA	−3.18	<0.01
Cumin	WLSTM	LSTM	−7.51	<0.01
	WSVR	SVR	−5.78	<0.01
	WANN	ANN	−2.89	<0.01
	WARIMA	ARIMA	−1.15	<0.01

## 5. Conclusions

In the current study, the effect of wavelet-based denoising was illustrated on the monthly wholesale prices of three pivotal spices—turmeric, coriander, and cumin, collected from the diverse markets across India. Wavelet denoising involves the application of methods or algorithms to filter out the extraneous noise, enabling the model to focus on the relevant patterns and relationships in the data. The various levels of decomposition were taken into account for studying the effectiveness of denoising. The benchmark modeling phase commenced with the deployment of the ARIMA model, a widely recognized time series forecasting method. The modeling spectrum expanded to include artificial neural networks (ANN), support vector regression (SVR), and long short-term memory (LSTM) models. These models were subjected to hyperparameter optimization through a grid search method, aiming to fine-tune the parameters for optimal performance. Also, the efficacy of wavelet-based denoising was assessed by comparing it with a benchmark model using three accuracy metrics, namely, RMSE, MAPE, and MAE. The results on the test set unequivocally indicated the consistent superiority of LSTM across all three spices, showcasing its adeptness in capturing the intricate patterns inherent in the data. The performance of each model may vary with different decomposition levels, emphasizing the importance of carefully selecting the appropriate level based on specific modeling objectives and metric priorities. The comparative analysis between the wavelet-based denoising models (WLSTM, WSVR, WANN, WARIMA) and their traditional counterparts provided resounding affirmation regarding the substantial enhancement in predictive

accuracy introduced by wavelet transforms. Across all three spices, denoising consistently led to significant reductions in error metrics, with percentage decreases ranging from 30% to over 40%.

In general, Wavelet LSTM at H6 appears to be a robust choice for accurate price predictions across all spices. The order of preference for model performance, based on the provided metrics, could be WLSTM > WARIMA > WSVR > WANN for turmeric, WLSTM > WANN > WSVR > WARIMA for coriander, and WLSTM > WSVR > WANN > WARIMA for cumin. The final step in the comparative analysis involved the application of the TOPSIS method. This method offered a comprehensive evaluation, considering all three performance metrics with equal weightage. The TOPSIS scores and ranks reaffirmed WLSTM's superiority, consistently placing it at the forefront across all decomposition levels and spices. The DM test also resulted in significant differences in prediction accuracy between wavelet-based models with that of the usual model. Indeed, the computational intensity of wavelet-based denoising, particularly with sophisticated algorithms, is a notable limitation. Processing large datasets or implementing real-time denoising may pose challenges in terms of computational resources. The effectiveness of wavelet-based denoising is highly dependent on the choice of the wavelet function. The other wavelet filters may be explored in future research to gain a clear idea about the selection of suitable filter and level for a specific pattern existed in the time series.

**Author Contributions:** Conceptualization, R.K.P. and C.T.; methodology, C.T., R.K.P. and M.Y.; software, C.T. and M.Y.; validation, C.T., M.Y. and A.K.P.; formal analysis, C.T.; investigation, C.T. and R.K.P.; resources, R.K.P.; data curation, C.T.; writing—original draft preparation, C.T.; writing—review and editing, C.T., M.Y. and R.K.P.; visualization, C.T. and A.K.P.; supervision, R.K.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Data will be available on reasonable request.

**Acknowledgments:** The authors are thankful to The Graduate School, ICAR-Indian Agricultural Research Institute, New Delhi, and Director, ICAR-Indian Agricultural Statistics Research Institute, New Delhi, for providing the required facility.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Oktoviany, P.; Knobloch, R.; Korn, R. A machine learning-based price state prediction model for agricultural commodities using external factors. *Decis. Econ. Financ.* **2021**, *44*, 1063–1085. [[CrossRef](#)]
2. Cerqueira, V.; Torgo, L.; Soares, C. A case study comparing machine learning with statistical methods for time series forecasting: Size matters. *J. Intell. Inf. Syst.* **2022**, *59*, 415–433. [[CrossRef](#)]
3. Khan, F.; Urooj, A.; Khan, S.A.; Alsubie, A.; Almaspoor, Z.; Muhammadullah, S. Comparing the Forecast Performance of Advanced Statistical and Machine Learning Techniques Using Huge Big Data: Evidence from Monte Carlo Experiments. *Complexity* **2021**, *2021*, 6117513. [[CrossRef](#)]
4. Rioul, O.; Vetterli, M. Wavelets and Signal Processing. *IEEE Signal Process Mag.* **1991**, *8*, 14–38. [[CrossRef](#)]
5. Gidwani, B.; Bhattacharya, R.; Shukla, S.S.; Pandey, R.K. Indian spices: Past, present and future challenges as the engine for bio-enhancement of drugs: Impact of COVID-19. *J. Sci. Food Agric.* **2022**, *102*, 3065–3077. [[CrossRef](#)] [[PubMed](#)]
6. Sharangi, A.B.; Pandit, M.K. *Indian Spices: The Legacy, Production and Processing of India's Treasured Export*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 341–357.
7. Valasan, A. Spices Export from Kerala Current Trends & Opportunities Ahead. *IRA-Int. J. Manag. Soc. Sci.* **2016**, *5*, 54–65.
8. Srinu, N.; Bindu, B.H. A Review on Machine Learning and Deep Learning based Rainfall Prediction Methods. In Proceedings of the 3rd International Conference on Power, Energy, Control and Transmission Systems, Chennai, India, 8–9 December 2022.
9. Ashok, S.P.; Pekkat, S. A systematic quantitative review on the performance of some of the recent short-term rainfall forecasting techniques. *J. Water Clim. Change* **2022**, *13*, 3004. [[CrossRef](#)]
10. Kaur, J.; Parmar, K.S.; Singh, S. Autoregressive models in environmental forecasting time series: A theoretical and application review. *Environ. Sci. Pollut. Res.* **2023**, *30*, 19617–19641. [[CrossRef](#)]
11. Mao, L.; Huang, Y.; Zhang, X.; Li, S.; Huang, X. ARIMA model forecasting analysis of the prices of multiple vegetables under the impact of the COVID-19. *PLoS ONE* **2022**, *17*, e0271594. [[CrossRef](#)]

12. Liakos, K.G.; Busato, P.; Moshou, D.; Pearson, S.; Bochtis, D. Machine Learning in Agriculture: A Review. *Sensors* **2018**, *18*, 2674. [[CrossRef](#)]
13. Mahto, A.K.; Alam, M.A.; Biswas, R.; Ahmad, J.; Alam, S.I. Short-Term Forecasting of Agriculture Commodities in Context of Indian Market for Sustainable Agriculture by Using the Artificial Neural Network. *J. Food Qual.* **2021**, *2021*, 1–13. [[CrossRef](#)]
14. Astudillo, G.; Carrasco, R.; Fernández-Campusano, C.; Chacón, M. Copper price prediction using support vector regression technique. *Appl. Sci.* **2020**, *10*, 6648. [[CrossRef](#)]
15. Jeong, M.; Lee, Y.J.; Choe, Y. Forecasting Agricultural Commodity Price: The Case of onion. *Quest J. J. Res. Humanit. Soc. Sci.* **2017**, *5*, 78–81.
16. Zhang, D.; Chen, S.; Liwen, L.; Xia, Q. Forecasting Agricultural Commodity Prices Using Model Selection Framework with Time Series Features and Forecast Horizons. *IEEE Access* **2020**, *8*, 28197–28209. [[CrossRef](#)]
17. Paul, R.K.; Yeasin, M.; Kumar, P.; Kumar, P.; Balasubramanian, M.; Roy, H.S.; Paul, A.K.; Gupta, A. Machine learning techniques for forecasting agricultural prices: A case of brinjal in Odisha, India. *PLoS ONE* **2022**, *17*, e0270553. [[CrossRef](#)]
18. Paul, R.K.; Yeasin, M.; Kumar, P.; Paul, A.K.; Roy, H.S. Deep Learning Technique for Forecasting Price of Cauliflower. *Curr. Sci.* **2023**, *124*, 1065–1073.
19. Liu, M.; Qin, H.; Cao, R.; Deng, S. Short-Term Load Forecasting Based on Improved TCN and DenseNet. *IEEE Access* **2022**, *10*, 115945–115957. [[CrossRef](#)]
20. Kapoor, A.; Pathiraja, S.; Marshall, L.; Chandra, R. DeepGR4J: A deep learning hybridization approach for conceptual rainfall-runoff modelling. *Environ. Model. Softw.* **2023**, *169*, 105831. [[CrossRef](#)]
21. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
22. Chen, X.; He, G.; Chen, Y.; Zhang, S.; Chen, J.; Qian, J.; Yu, H. Notice of Retraction: Short-term and local rainfall probability prediction based on a dislocation support vector machine model using satellite and in-situ observational data. *IEEE Access* **2021**, *1*. [[CrossRef](#)]
23. Gu, Y.H.; Jin, D.; Yin, H.; Zheng, R.; Piao, X.; Yoo, S.J. Forecasting Agricultural Commodity Prices Using Dual Input Attention LSTM. *Agriculture* **2022**, *12*, 256. [[CrossRef](#)]
24. Cakici, N.; Fieberg, C.; Metko, D.; Zaremba, A. Do anomalies really predict market returns? New data and new evidence. *Rev. Financ.* **2023**, rfad025. [[CrossRef](#)]
25. Dong, X.; Li, Y.; Rapach, D.E.; Zhou, G. Anomalies and the expected market return. *J. Financ.* **2022**, *77*, 639–681. [[CrossRef](#)]
26. Paul, R.K.; Garai, S. Wavelets Based Artificial Neural Network Technique for Forecasting Agricultural Prices. *J. Indian Soc. Probab. Stat.* **2022**, *23*, 47–61. [[CrossRef](#)]
27. Lang, M.; Guo, H.; Odegard, J.E.; Burrus, C.S.; Wells, R.O. Noise reduction using an undecimated discrete wavelet transform. *Signal Process. Lett. IEEE* **1996**, *3*, 10–12. [[CrossRef](#)]
28. Boto-Giralda, D.; Díaz-Pernas, F.J.; González-Ortega, D.; Díez-Higuera, J.F.; Antón-Rodríguez, M.; Martínez-Zarzuela, M.; Torre-Díez, I. Wavelet-Based Denoising for Traffic Volume Time Series Forecasting with Self-Organizing Neural Networks. *Comput. Aided Civ. Infrastruct. Eng.* **2010**, *25*, 530–545. [[CrossRef](#)]
29. Paul, R.K.; Garai, S. Performance comparison of wavelets-based machine learning technique for forecasting agricultural commodity prices. *Soft Comput.* **2021**, *25*, 12857–12873. [[CrossRef](#)]
30. Shabri, A.; Hamid, M.F.A. Wavelet-support vector machine for forecasting palm oil price. *Malays. J. Fundam. Appl. Sci.* **2019**, *15*, 398–406. [[CrossRef](#)]
31. Garai, S.; Paul, R.K.; Rakshit, D.; Yeasin, M.; Emam, W.; Tashkandy, Y.; Chesneau, C. Wavelets in Combination with Stochastic and Machine Learning Models to Predict Agricultural Prices. *Mathematics* **2023**, *11*, 2896. [[CrossRef](#)]
32. Chen, Q.; Lin, X.; Zhong, Y.; Xie, Z. Price prediction of agricultural products based on wavelet analysis-LSTM. In Proceedings of the 2019 IEEE International Conference on Parallel and Distributed Processing with Applications, Big Data and Cloud Computing, Sustainable Computing and Communications, Social Computing and Networking, ISPA/BDCloud/SustainCom/SocialCom (IEEE), Xiamen, China, 16–18 December 2019; pp. 984–990.
33. Liang, X.; Ge, Z.; Sun, L.; He, M.; Chen, H. LSTM with wavelet transform based data preprocessing for stock price prediction. *Math. Probl. Eng.* **2019**, *2019*, 1340174. [[CrossRef](#)]
34. Zhou, X.; Zhou, H.; Long, H. Forecasting the equity premium: Do deep neural network models work? *Mod. Financ.* **2023**, *1*, 1–11. [[CrossRef](#)]
35. Jaseena, K.U.; Koor, B.C. Decomposition-based hybrid wind speed forecasting model using deep bidirectional LSTM networks. *Energy Convers. Manag.* **2021**, *234*, 113956. [[CrossRef](#)]
36. Peng, L.; Chen, K.; Li, N. Predicting Stock Movements: Using Multiresolution Wavelet Reconstruction and Deep Learning in Neural Networks. *Information* **2021**, *12*, 388. [[CrossRef](#)]
37. Singla, P.; Duhan, M.; Saroha, S. An ensemble method to forecast 24-h ahead solar irradiance using wavelet decomposition and BiLSTM deep learning network. *Earth Sci. Inform.* **2022**, *15*, 291–306. [[CrossRef](#)]
38. Yeasin, M.; Paul, R.K. OptiSembleForecasting: Optimization based ensemble forecasting using MCS algorithm and PCA based error index. *J. Supercomput.* **2023**, *80*, 1568–1597. [[CrossRef](#)]
39. Liang, J.; Jia, G. China futures price forecasting based on online search and information transfer. *Data Sci. Manag.* **2022**, *5*, 187–198. [[CrossRef](#)]

40. Cai, P.; Zhang, C.; Chai, J. Forecasting hourly PM 2.5 concentrations based on decomposition-ensemble-reconstruction framework incorporating deep learning algorithms. *Data Sci. Manag.* **2023**, *6*, 46–54. [[CrossRef](#)]
41. Deng, C.; Huang, Y.; Hasan, N.; Bao, Y. Multi-step-ahead stock price index forecasting using long short-term memory model with multivariate empirical mode decomposition. *Inf. Sci.* **2022**, *607*, 297–321. [[CrossRef](#)]
42. Lin, Y.; Chen, K.; Zhang, X.; Tan, B.; Lu, Q. Forecasting crude oil futures prices using BiLSTM-Attention-CNN model with Wavelet transform. *Appl. Soft Comput.* **2022**, *130*, 109723. [[CrossRef](#)]
43. Makridakis, S.; Wheelwright, S.C.; Hyndman, R.J. *Forecasting Methods and Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2008.
44. Feng, F.; Na, W.; Jin, J.; Zhang, J.; Zhang, W.; Zhang, Q.J. Artificial neural networks for microwave computer-aided design: The state of the art. *IEEE Trans. Microw. Theory Tech.* **2022**, *70*, 4597–4619. [[CrossRef](#)]
45. Valiant, L.G. A theory of the learnable. *Commun. ACM* **1984**, *27*, 1134–1142. [[CrossRef](#)]
46. Feng, G.; He, J.; Polson, N.G. Deep learning for predicting asset returns. *arXiv* **2018**, arXiv:1804.09314.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.