Deep learning technique for forecasting the price of cauliflower

Ranjit Kumar Paul[1,*], Md Yeasin[1], Pramod Kumar[2], A. K. Paul[1] and H. S. Roy[1]

[1]ICAR-Indian Agricultural Statistics Research Institute, New Delhi 110 012, India
[2]ICAR-Indian Agricultural Research Institute, New Delhi 110 012, India

Vegetables are the staple food in our diets. Vegetable prices are difficult to forecast because they are influenced by a variety of factors, including weather, demand and supply chain, Government policies, etc. and exhibit volatile fluctuations. Marketing of vegetables is complex, especially because of their perishability, seasonality and bulkiness. An accurate and timely forecast of vegetables is essential to help its stakeholders. Previous studies observed that traditional statistical models are unable to capture the complex behaviour of vegetable markets. In this study, a comparative assessment has been carried out among the traditional time-series model, machine learning and deep learning techniques in order to find the best-suited model. For empirical illustration, cauliflower markets have been chosen as it is one of India's most important and popular winter. In order to identify the complexity in the price of cauliflower, the machine learning technique, i.e. artificial neural network and deep learning technique, i.e. long short-term memory model have been implemented. In addition, the traditional stochastic time-series model, i.e. autoregressive integrated moving average model, was used to compare the prediction accuracy of the above models. To this end, the moving window forecast approach was also implemented to evaluate the sensitivity of these models with respect to forecast length. It can be concluded that the deep learning model outperforms the traditional time-series model and the machine learning technique for both short- and long-term forecasting.

Keywords: Cauliflower, deep learning technique, machine learning, statistical models, vegetable prices.

VEGETABLES are an important source of several vital nutrients, including potassium, dietary fibre, vitamin A and vitamin C (ref. 1). In order to raise awareness regarding the nutritional and health benefits of fruits and vegetables, the United Nations (UN) General Assembly declared 2021 as the 'International Year of Fruits and Vegetables'. The vegetable sector is regarded as critical because it generates high revenue and employment, enhances nutrition and protects and conserves the environment[2]. Due to its diverse climate and ecology, India is the world's second largest producer of fresh vegetables[3]. Also, vegetables account for a substantial share of the market in India for daily consumption, and their prices significantly impact consumer spending and farm household income. The major drawbacks of vegetables are that they are highly perishable and show high volatility in prices due to demand and production inconsistency[4]. When there is an oversupply of vegetables, prices fall, causing financial losses to agricultural households; however, when there is an undersupply, prices rise, putting a burden on consumers. As supply and demand imbalance affects producers and consumers, the Government must balance these variables well. As a result, accurate forecasting of the vegetable market price is critical because prior information can aid in the formulation of a well-planned management strategy, reduce risk and ultimately contribute to the stability of the demand–supply channel. Modelling and forecasting vegetable prices are complex as a large number of market factors affect them. In the past, several studies attempted to forecast the market price of vegetables using different parametric, non-parametric and machine learning models[5–9]. The effectiveness of parametric models and a non-parametric model (spectral analysis) for forecasting vegetable prices were studied by Dieng[10]. For predicting vegetable prices, Luo *et al.*[11] introduced four models using the machine learning technique based on neural networks and backpropagation. Nasira *et al.*[12] presented a data mining classification model to predict the complex behaviour of vegetables. Xiong *et al.*[13] developed a hybrid model for forecasting seasonal vegetable prices that combines seasonal-trend decomposition techniques based on loess (STL) and extreme learning machines (ELMs). Kyriazi *et al.*[14] introduced a new forecasting methodology for modelling agricultural commodity prices, such as vegetable prices, using adaptive learning forecasting. Although stochastic and machine learning methods have been widely employed in vegetable price forecasting, they cannot effectively model the complex behaviour of vegetable prices in India due to high price fluctuation in the markets. Deep learning algorithms are a new breed of potential price-predicting methodologies. The long short-term memory (LSTM) model, a variation of the recurrent neural network (RNN), can efficiently utilize both long and short term information from time-series data[15]. Unlike other approaches, its feedback connection provides a better understanding of the developed patterns of data series by implementing the backpropagation of current historical prices and present

*For correspondence. (e-mail: ranjitstat@gmail.com)

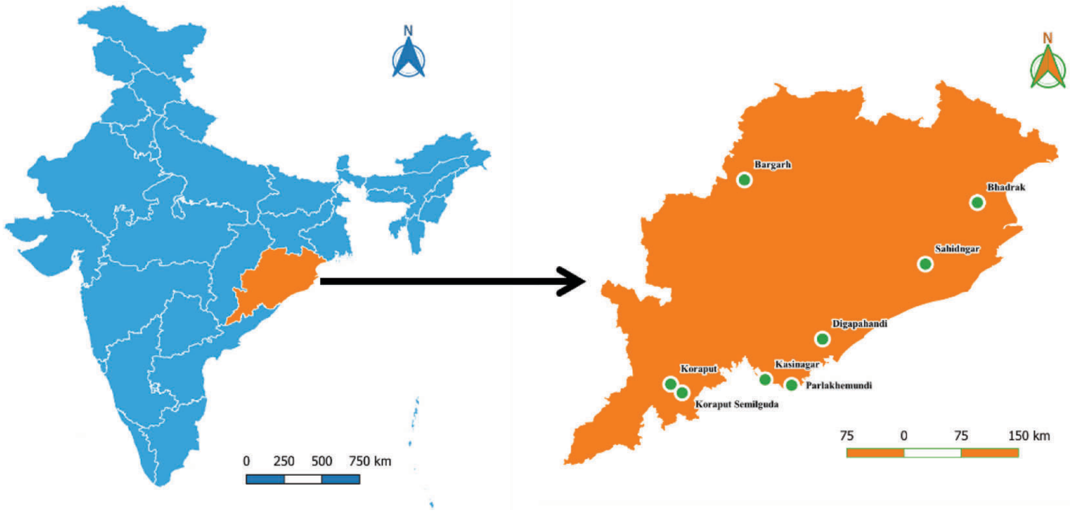

Figure 1. Study markets.

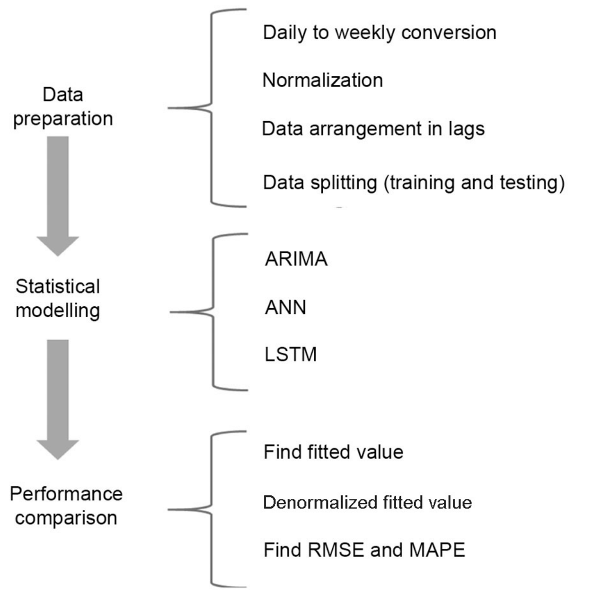

Figure 2. Research methodology.

prices[16]. Chen *et al.*[17] introduced the wavelet-based LSTM model, while Yin *et al.*[18] proposed the STL–attention-based LSTM for forecasting vegetable prices using various types of information. In the Indian context, deep learning has hardly been used for agricultural price prediction. The aim of the present study is to examine the potential of deep learning techniques in forecasting vegetable market prices. The most versatile and popular vegetable, cauliflower (*Brassica oleracea* var. *botrytis*), has been chosen for the study. The cauliflower market price data were collected from eight markets (Digapahandi, Bargarh, Bhadrak, Kasinagar, Koraput, Koraput Semilguda, Parlakhemundi and Sahidngar) in Odisha, as it is one of the largest vegetable-consuming states. The markets were chosen based on the total arrival. Figure 1 shows the major markets considered for the present study.

Methodology

The research methodology can be divided into three phases, namely data preparation, statistical modelling and performance comparison (Figure 2).

ARIMA model

The autoregressive integrated moving average (ARIMA) model is a function of past values and past error terms. An ARIMA (p, d, q) can be written as follows

$$\varphi(B)(1-B)^d y_t = \theta(B)\varepsilon_t, \tag{1}$$

where $\varphi(B) = 1 - \varphi_1 B - \varphi_2 B^2 - \ldots - \varphi_p B^p$ is the autoregressive polynomial of order p; $\varphi_1, \varphi_2, \ldots, \varphi_p$ are autoregressive parameters; $\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q$ is the moving average polynomial of order q; $\theta_1, \theta_2, \ldots, \theta_q$ are the moving average parameters; d is the differencing operator; B is the backshift operator on y_t defined as $B^i(y_t) = y_{t-i}$ and ε_t is the white noise such that $\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$. ARIMA ($p$, d, q) model building has different steps (Figure 3). These are as follows:

(i) Ensure stationarity: Using unit root tests such as the augmented Dickey-Fuller (ADF) test, Philips Peron test and KPSS test, the stationarity of the series is tested, and the value of d is determined.

(ii) Identification: Order of moving average (q) and order of autoregression (p) are identified in this stage by observing the autocorrelation function (ACF) and partial autocorrelation function (PACF) respectively.

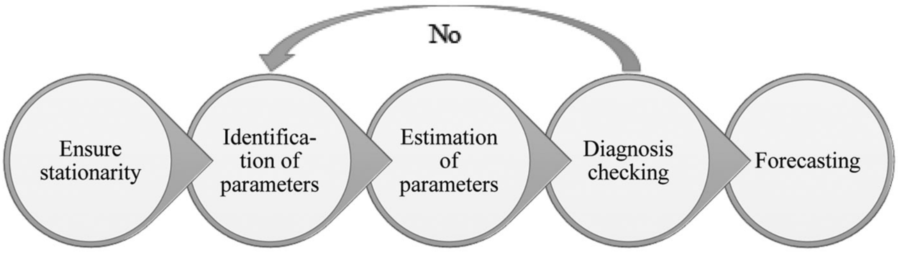

Figure 3. ARIMA process.

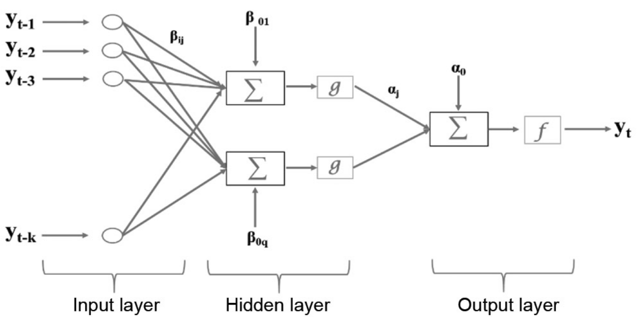

Figure 4. ANN architecture.

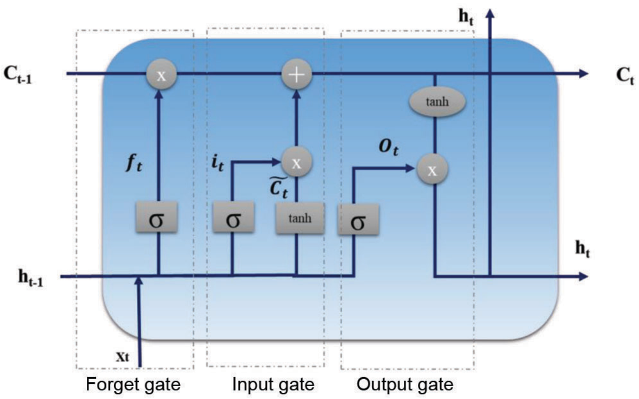

Figure 5. LSTM structure.

(iii) Estimation: Once the order is identified, the estimation of unknown parameters is done by means of the non-linear least square method.

(iv) Diagnostic checking: This is an important step of model-building where the residuals obtained from the developed model are examined, and the adequacy of the model is checked.

(v) Forecasting: The forecast of out-of-sample observations is done in this step.

Artificial neural network model

Artificial neural network (ANN) is a popular machine learning technique suitable for modelling over a wide range of applications due to its flexible architecture. The ANN architecture is mainly composed of three layers[19]. These are the input layer, single hidden layer and output layer (Figure 4). The number of neurons in the input or hidden layer is flexible.

In Figure 4, y_t is considered as a nonlinear function of $y_{t-1}, y_{t-2}, \ldots, y_{t-k}$ past observations and weights.

$$y_t = f(y_{t-1}, y_{t-2}, \ldots, y_{t-k}, w) + \varepsilon_t, \tag{2}$$

where w is a class of all parameters of the model and f is a nonlinear function.

y_t can be expressed as follows

$$y_t = a_0 + \sum_{j=1}^{q} a_j g\left(\beta_{0j} + \sum_{i=1}^{k} \beta_{ij} t_{t-i}\right) + \varepsilon_t, \tag{3}$$

where a_j ($j = 0, 1, 2, \ldots, q$) and β_{ij} ($j = 0, 1, 2, \ldots, q$; $i = 1, 2, \ldots, k$) are parameters of the model, often called the connection weights; k the number of input nodes and q is the number of hidden nodes. The logistic function is most commonly used transfer layer

$$g(x) = \frac{1}{1+\exp(-x)}. \tag{4}$$

For training the multi-layer ANN model, mainly backpropagation, a supervised learning algorithm is used[20]. In this study, we propose an ANN model with 20 input neurons, one hidden layer with 10 neurons and 1 output neuron.

Long short-term memory model

In 1997, Hochreiter and Schmidhuber[21] introduced the LSTM neural network to include the benefits of addressing long-term data dependencies. Because of long-term dependency, the LSTM model was highly suitable for financial high-frequency time-series data. It can also solve the problems of RNN models, gradient expansion and gradient disappearance[22]. The LSTM model has three memory modules: input gate, output gate, and forget gate (Figure 5). The main functions of these three gates are retaining important information and discarding irrelevant information from the units. A variety of LSTM models are available in the literature; we used Hochreiter and Schmidhuber's LSTM model in our study.

The operational premise of the LSTM model is to analyse the information at time t. First, it discards unnecessary information using the forget gate. Then it filters useful information with a given probability using the input gate and ultimately extracts useful information using the output gate, which participates in the next LSTM unit. The selection of the activation function is an important step in the LSTM process. Here we have used the standard sigmoid function and the tanh function as activation functions. The LSTM process can be summarized in five steps.

Step 1: The output value of the previous unit and input value of the current unit are integrated into the forget gate. The output value of the forget gate is calculated as

$$f_t = \sigma\{W_f * (h_{t-1} * x_t)\} + b_f, \tag{5}$$

where W_f is the weight of the forget gate, b_f the bias, x_t the input value and h_{t-1} is the output value of the prior unit.

Step 2: The output value of the prior unit and the input value of the current time are incorporated into the input gate. The output value and candidate cell state values are computed as

$$i_t = \sigma\{W_i * (h_{t-1} * x_t)\} + b_i, \tag{6}$$

$$\tilde{C}_t = \tanh\{W_c * (h_{t-1} * x_t)\} + b_c, \tag{7}$$

where W_i and b_i are the weight and bias of the input gate and W_c and b_c are the weight and bias of the candidate input respectively.

Step 3: Updation of the current cell is done using the formula

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t, \tag{8}$$

Step 4: the output gate takes h_{t-1} and x_t as input values, and its output is calculated using the formula

$$o_t = \sigma\{W_o * (h_{t-1} * x_t)\} + b_o, \tag{9}$$

where W_i and b_i are the weight and bias of this gate respectively.

Step 5: The final output of the LSTM unit is generated by computing the output of the output gate and the cell state, as follows

$$h_t = o_t * \tanh(C_t), \tag{10}$$

Performance comparison

To demonstrate the accuracy of different techniques, two performance metrics, i.e. root mean square error (RMSE) and mean absolute percentage error (MAPE) were used.

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

$$\text{MAPE (\%)} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| * 100, \tag{12}$$

where y_i and $\hat{y}_i$ denotes actual and predicted values of observation at the ith time point respectively, and n is the number of observations used for validating the models.

Results and discussion

Data preparation

Daily wholesale price data of cauliflower for the eight major markets in Odisha were collected from the AGMARKNET portal (https://agmarknet.gov.in/) from January 2015 to February 2021. The missing values were imputed by means of the kernel imputation technique. Data preparation comprised four steps; daily to weekly conversion, normalization, arrangement of data in lag and data splitting (training and testing) (Figure 6).

Step 1: The original data (y') were converted into weekly data (y^{w}) by taking the average price of seven days.

Step 2: Normalizing data is important before introducing it into a machine learning or deep learning model. Data normalization helps reduce the overall training time. Various techniques of data normalization are available in the literature, such as Z-score normalization, minimax sigmoid, etc.[23]. We have used minimax normalization in the present study

$$y_i = \frac{y_i^{w} - y_{\min}^{w}}{y_{\min}^{w} - y_{\max}^{w}}, \tag{13}$$

where y^{w} is the weekly data; $y_{\min}^{w}$ and $y_{\max}^{w}$ the minimum and maximum values of the weekly data respectively, and y_i are the normalized data.

Step 3: The single-series datasets were converted into a matrix of lag values. Here we have used a maximum of 20 lag values for each series. The number of lags was selected based on the ACF of the original series.

Step 4: In this step, the data was split in the 70: 30 ratio into training and testing sets. Therefore, in the present study, the training set consists of 224 observations, whereas the testing set consists of 96 observations.

Summary statistics

Table 1 presents the summary statistics of the series for each market. A perusal of Table 1 indicates that the average

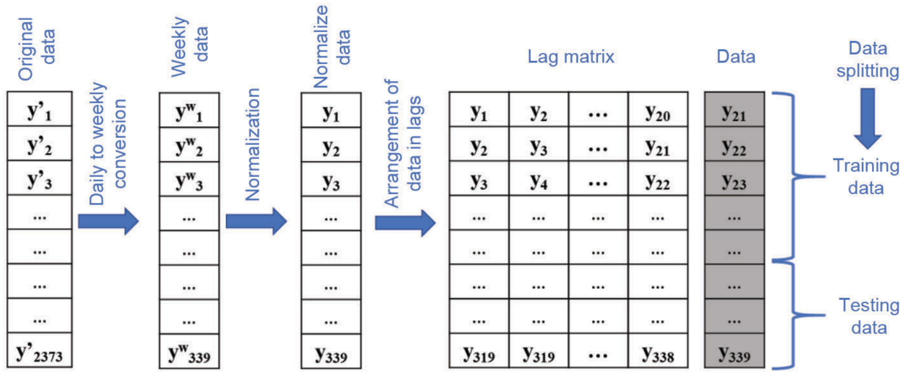

Figure 6. Data preparation.

Table 1. Summary statistics

Statistics	Digapahandi	Bargarh	Bhadrak	Kasinagar	Koraput	Koraput Semilguda	Parlakhemundi	Sahidnagar
Mean	3,496.97	1,541.09	2,554.22	4,093.27	2,505.47	2,498.24	3,869.85	2,568.20
SE	28.91	21.13	25.81	39.16	21.20	21.67	39.28	28.35
Median	3,615.80	1,200.00	2,500.00	4,000.00	2,450.00	2,483.33	3,164.06	2,084.57
Mode	2,700.00	533.33	2,500.00	2,000.00	3,100.00	4,100.00	2,000.00	4,500.00
SD	1,408.29	1,029.26	1,257.33	1,907.82	1,032.79	1,055.81	1,913.30	1,380.79
CV (%)	40.27	66.79	49.23	46.61	41.22	42.26	49.44	53.76
Kurtosis	2.17	2.90	2.06	4.51	2.22	2.18	2.17	2.06
Skewness	–0.03	0.97	0.37	0.87	0.12	0.00	0.40	0.41
Minimum	550	400	500	500	550	500	600	500
Maximum	7,500	6,100	8,000	13,500	5,600	5,600	10,700	7,500
Jarque–Bera statistics	368.79	142.27	69.14	521.97	65.75	131.13	154.22	67.08
P-value	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001

SD, Standard deviation; SE, Standard error; CV, Coefficient of variation.

price remains high in the Kasinagar market, whereas a lower average price is observed in Bargarh. The price variability measured in terms of coefficient of variation (CV) is observed to be highest in the Bargarh market, followed by Sahidnagar, whereas the lowest CV is reported in Digapahandi market. The skewness and kurtosis of all the price series (except skewness in Semilguda market which indicates that the distribution is symmetric) along with the Jarque–Bera test results indicate that the series are deviating from normality. To visualize the distribution of the series, kernel density estimates have been plotted in Figure 7, which also reveals that the probability distribution of each of the series has deviated from normality. This finding indicates that no parametric time-series model is appropriate. Therefore, nonparametric and nonlinear machine learning techniques adequately represented the pattern in the series. The inappropriateness and inadequacy of the parametric time series model, i.e. ARIMA model, are shown from its prediction performance and residual diagnostic point of view.

Fitting of models

Before the application of any model, the dataset is split into training and testing sets with a ratio of 70 : 30. On the training set, the model is trained, and the parameters and hyper-parameters of machine learning and deep learning techniques are tuned to get the optimized value. The best ARIMA model for the data under consideration was chosen based on log-likelihood and AIC (Table 2). Table 2 shows the values of optimized parameters and hyper-parameters of LSTM and ANN. In LSTM, a dense layer with one unit as the output layer has been considered. Softmax activation function was used with loss function as MAE and optimizer as Adam. The total number of epochs was considered as 50. In ANN, the logistic function was used as the activation function with the loss function as MAE and the optimization technique as backpropagation. Table 2 also shows the learning rate and number of iterations for ANN to reach convergence.

The ARIMA, ANN and LSTM model are implemented individually for forecasting vegetable price.

Table 3 shows the prediction accuracy of the three above-mentioned models. The LSTM model has the lowest MAPE and RMSE throughout all datasets, followed by ANN and ARIMA, except for Bhadrak (MAPE), where the ANN model has higher prediction accuracy than the LSTM model. Based on Table 3, we can conclude that the LSTM model outperforms the other two models in modelling vegetable

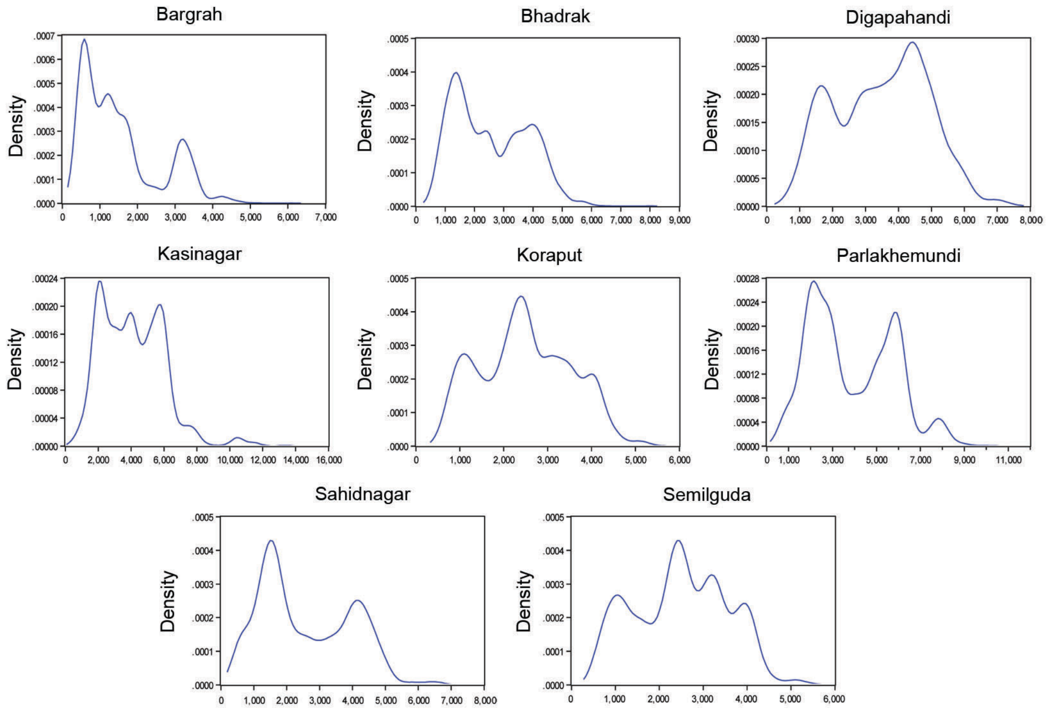

Figure 7. Kernel density estimates of price series.

Table 2. Optimum value of parameters and hyper-parameters of different techniques

Market	LSTM lags	Layers	IS	Units	ANN #HL	LR	#I	ARIMA order (p, d, q)	LL	AIC
Digapahandi	19	LSTM	(19,1)	25	11	0.0099	44,770	(4, 1, 2)	257.13	–500.26
		Dense	(, 25)	1						
Bargarh	22	LSTM	(22,1)	29	13	0.0085	67,432	(1, 0, 0)	236.31	–470.62
		Dense	(, 29)	1						
Bhadrak	20	LSTM	(20,1)	25	11	0.0093	30,102	(1, 0, 1)	333.67	–659.34
		Dense	(, 25)	1						
Kasinagar	22	LSTM	(22,1)	30	12	0.0098	23,922	(1, 0, 0)	236.31	–470.62
		Dense	(, 30)	1						
Koraput	18	LSTM	(18,1)	23	9	0.0099	22,359	(1, 0, 1)	326.93	–645.86
		Dense	(, 23)	1						
Koraput Semilguda	22	LSTM	(22,1)	25	11	0.0098	40,060	(4, 1, 0)	262.20	–514.40
		Dense	(, 25)	1						
Parlakhemundi	21	LSTM	(21,1)	27	9	0.0096	14,451	(2, 1, 0)	327.36	–648.71
		Dense	(, 27)	1						
Sahidngar	19	LSTM	(19,1)	21	10	0.0099	12,771	(3, 0, 0)	244.53	–479.05
		Dense	(, 21)	1						

IS, Input size; #HL, No. of nodes in hidden layer; LR, Learning rate; #I, No. of iteration to converge; LL, Log likelihood; p, Autoregressive parameters; d, Differencing levels; q, Moving average parameters.

Table 3. Prediction performance metrics

	MAPE (%)			RMSE		
Market	ARIMA	ANN	LSTM	ARIMA	ANN	LSTM
Digapahandi	28.06	23.82	13.06	1018.43	1133.42	621.73
Bargarh	60.88	35.23	17.12	1734.65	1006.91	419.97
Bhadrak	50.88	21.51	23.03	1643.22	1051.93	1012.19
Kasinagar	37.22	18.99	15.25	1462.56	1002.65	740.34
Koraput	45.25	34.18	12.57	1135.98	1036.59	467.10
Koraput Semilguda	48.25	44.51	16.03	1075.44	1030.49	456.27
Parlakhemundi	34.42	25.54	16.50	1371.95	994.57	746.32
Sahidngar	50.58	33.10	23.75	1706.68	1236.70	793.84

Table 4. DM test results

Markets	ARIMA vs ANN		ARIMA vs LSTM		ANN vs LSTM	
	Test statistic	P-value	Test statistic	P-value	Test statistic	P-value
Digapahandi	–0.86	0.80	3.58	0.00	4.86	0.00
Bargarh	7.68	0.00	9.77	0.00	5.38	0.00
Bhadrak	5.38	0.00	6.89	0.00	0.43	0.33
Kasinagar	4.58	0.00	5.58	0.00	2.35	0.01
Koraput	2.23	0.01	7.48	0.00	7.14	0.00
Koraput Semilguda	1.17	0.12	5.68	0.00	6.85	0.00
Parlakhemundi	5.91	0.00	6.64	0.00	3.85	0.00
Sahidngar	4.22	0.00	6.36	0.00	3.94	0.00

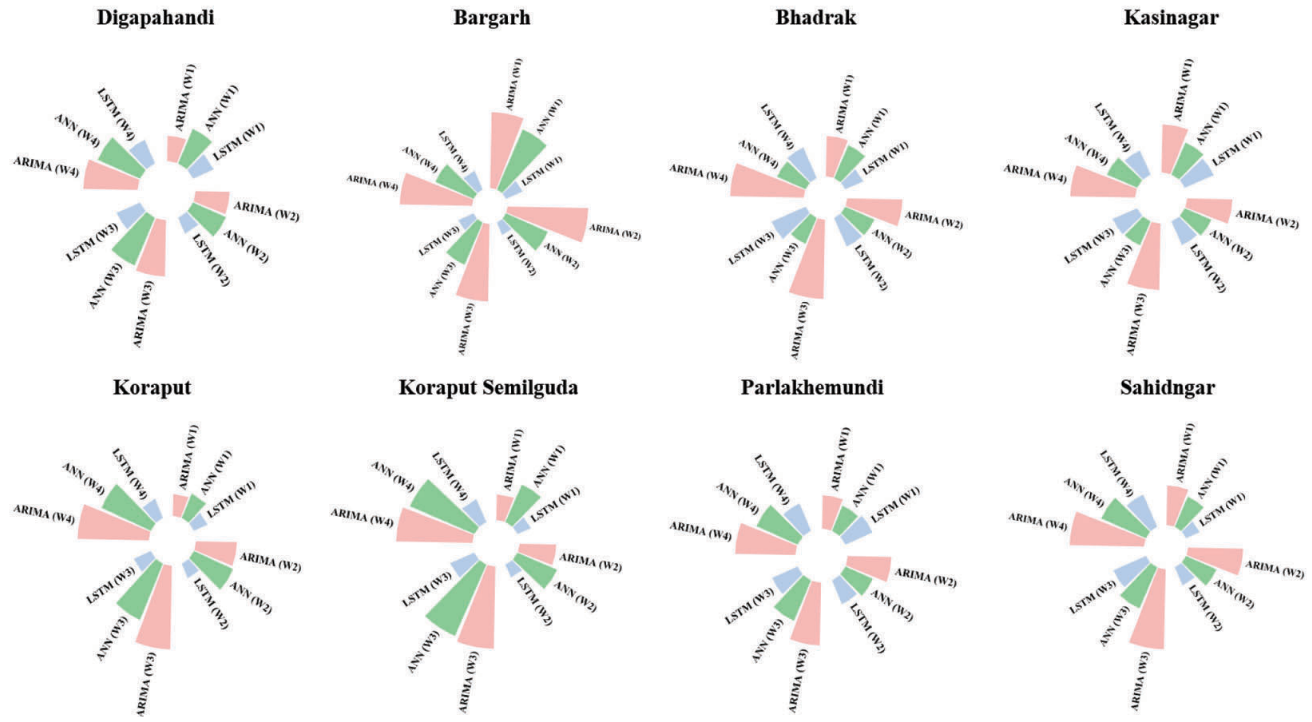

Figure 8. Moving window forecast performance (in terms of RMSE).

prices. The Diebold–Mariano (DM)[24] test was applied to check for significant differences among the competing models. The null and alternative hypotheses of the test are as follows: H_0: Prediction accuracy of the two models is the same, and H_1: Prediction accuracy of the second model is better than the first model. Table 4 shows results of the DM test. A perusal of the table indicates that ANN is superior to ARIMA in all markets except Digapahandi, where both models perform at par. LSTM is superior to ARIMA in all the studied markets. Also, LSTM is superior to ANN in all the markets except Bhadrak, where there is no significant difference in the prediction accuracy between ANN and LSTM.

The features of vegetable prices are difficult to comprehend because they exhibit volatile movements (Figure 8). The main aim of introducing the deep learning model in vegetable price forecasting is to learn the complex features present in the data. To evaluate the prediction performance of these models in short- and long-term forecasting, the moving window forecast approach was used. Four moving windows have been defined, viz. W1: 16 steps ahead forecast; W2: 32 steps ahead forecast; W3: 48 steps ahead forecast and W4: 64 steps ahead forecast, i.e. overall test forecast. RMSE for these windows was computed and has been presented in Figure 8. The figure shows that with a few exceptions, the LSTM model has the lowest RMSE value for all market windows. LSTM is less efficient than ANN in W3 and W4 of Bhadrak and W1 of Parlakhemundi and equally efficient with ANN in W1 of Bhadrak and W3 of Kasinagar. The ANN model is superior to the ARIMA model in most windows, with a few exceptions. In W2 of Digapahandi and W2 of Koraput, ARIMA and

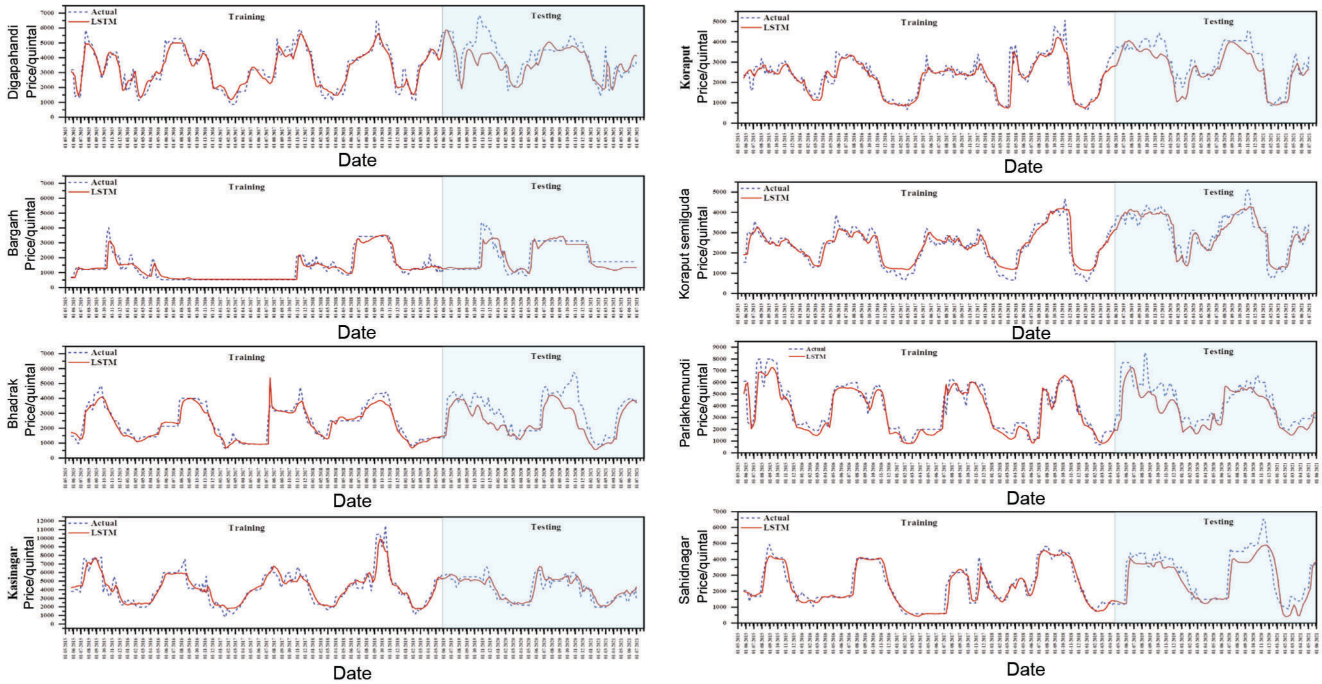

Figure 9. Actual versus predicted price plots.

ANN are equally precise, and in W1 of Digapahandi, W1 of Koraput, and W1 and W2 of Koraput Semilguda, ARIMA outperforms the ANN model. Digapahandi, Koraput and Koraput Semilguda markets witnessed the supremacy and inferiority of ARIMA over ANN in short-term and long-term forecasting respectively. The remaining markets support the fact that ANN is superior to ARIMA for both short- and long-term forecasting. All the markets support the efficiency of the LSTM model over ANN and ARIMA models for both short- and long-term forecasting. The effect of forecast length is low in the LSTM model, modest in the ANN model and very high in the ARIMA model. It is fair to conclude that the ARIMA model may be preferred over the ANN model for short-term forecasting, but it is not recommended for long-term forecasting. It can also be inferred that LSTM achieves higher accuracy irrespective of prediction length.

The deep learning model (LSTM model) effectively captures the movements of vegetable prices across all markets, whereas the machine learning model (ANN model) fails to provide the expected results. Figure 9 presents the fitted series obtained from the LSTM model and the original series.

Conclusion

Vegetable price prediction is important as vegetables are part of our daily diet and significantly impact farmers' income due to their perishable nature and high volatile price. Cauliflower prices are predicted in this study using ARIMA, ANN and LSTM models. We applied the proposed models in eight major cauliflower markets in Odisha. The sensitivity of these three models with respect to forecasting length was analysed using a moving window forecast. According to the results, the LSTM model outperforms the ARIMA and ANN models for both short- and long-term forecasting. The effect of forecast size is relatively low in the LSTM model but high in the ARIMA and ANN models. The ARIMA model is preferred over the ANN model for short-term but not long-term forecasting. Based on this comparative study, we can conclude that the deep learning model captures the complex pattern of vegetable prices efficiently, obtains the highest prediction accuracy and outperforms the traditional time series model (ARIMA) and the machine learning technique (ANN). It has the potential to improve agricultural market price prediction accuracy significantly. More research is needed to improve prediction accuracy by incorporating other important factors into the models.

Conflict of interest: The authors declare that there is no conflict of interest.

1. Dias, J. S., Nutritional quality and effect on disease prevention of vegetables. *Food Nutr. Sci.*, 2019, **10**(4), 369–402.
2. Kumar, V., Sharma, H. R. and Singh, K., Behaviour of market arrivals and prices of selected vegetable crops: a study of four metropolitan markets. *Agric. Econ. Res. Rev.*, 2005, **18**(2), 271–290.
3. https://www.fao.org/faostat/es/#home (accessed on 13 January 2022).
4. Paul, R. K., Das, T. and Yeasin, M., Ensemble of time series and machine learning model for forecasting volatility in agricultural prices. *Natl. Acad. Sci. Lett.*, 2023; https://doi.org/10.1007/s40009-023-01218-x

5. Paul, R. K. and Garai, S., Wavelets based artificial neural network technique for forecasting agricultural prices. *J. Indian Soc. Prob. Stat.*, 2022, **23**(1), 47–61.
6. Paul, R. K., Rana, S. and Saxena, R., Effectiveness of price forecasting techniques for capturing asymmetric volatility for onion in selected markets of Delhi. *Indian J. Agric. Sci.*, 2016, **86**(3), 303–309.
7. Rakshit, D., Paul, R. K. and Sanjeev, P., Asymmetric price volatility of onion in India. *Indian J. Agric. Econ.*, 2021, **76**(2), 245–260.
8. Paul, R. K., Yeasin, M. and Paul, A. K., The volatility spillover of potato prices in different markets of India. *Curr. Sci.*, 2022, **123**(3), 482–487.
9. Paul, R. K. *et al.*, Machine learning techniques for forecasting agricultural prices: a case of brinjal in Odisha, India. *PLoS ONE*, 2022, **17**(7), e0270553.
10. Dieng, A., Alternative forecasting techniques for vegetable, *Rev. Sénégalais Rec. Agric. Agroallementalress*, 2008, **1**(3), 5–10.
11. Luo, C., Wei, Q., Zhou, L., Zhang, J. and Sun, S., Prediction of vegetable price based on neural network and genetic algorithm. *IFIP Adv. Infor. Commun. Technol.*, 2011, **346**, 672–681.
12. Nasira, G. M., Professor, A. and Hemageetha, N., Forecasting model for vegetable price using back propagation neural network. *Int. J. Comput. Intel. Informat.*, 2012, **2**(2), 110–115.
13. Xiong, T., Li, C. and Bao, Y., Seasonal forecasting of agricultural commodity price using a hybrid STL and ELM method: evidence from the vegetable market in China. *Neurocomputing*, 2018, **275**, 2831–2844.
14. Kyriazi, F., Thomakos, D. D. and Guerard, J. B., Adaptive learning forecasting, with applications in forecasting agricultural prices. *Int. J. Forecast.*, 2019, **35**(4), 1356–1369.
15. Hua, Y., Zhao, Z., Li, R., Chen, X., Liu, Z. and Zhang, H., Deep learning with long short-term memory for time series prediction. *IEEE Commun. Mag.*, 2019, **57**(6), 114–119.
16. Ma, Q., Comparison of ARIMA, ANN and LSTM for stock price prediction. *E3S Web of Conf.*, 2020, **218**.
17. Chen, Q., Lin, X., Zhong, Y. and Xie, Z., Price prediction of agricultural products based on wavelet analysis – LSTM. IN Proceedings – 2019 IEEE International Conference on Parallel and Distributed Processing with Applications, Big Data and Cloud Computing, Sustainable Computing and Communications, Social Computing and Networking, Singapore, 2019, pp. 984–990.
18. Yin, H., Jin, D., Gu, Y. H., Park, C. J., Han, S. K. and Yoo, S. J., STL-ATTLSTM: vegetable price forecasting using STL and attention mechanism-based LSTM. *Agriculture*, 2020, **10**(12), 612.
19. Zhang, G., Eddy Patuwo, B. and Hu, Y. M., Forecasting with artificial neural networks: the state of the art. *Int. J. Forecast.*, 1998, **14**(1), 35–62.
20. Rumelhart, D. E., Hinton, G. E. and Williams, R. J., *Learning Internal Representations by Error Propagation*, California University, San Diego, La Jolla Institute for Cognitive Science, USA, 1985.
21. Hochreiter, S. and Schmidhuber, J., Long short-term memory. *Neural Comput.*, 1997, **9**(8), 1735–1780.
22. Ta, V. D., Liu, C. M. and Tadesse, D. A., Portfolio optimization-based stock prediction using long-short term memory network in quantitative trading. *Appl. Sci.*, 2020, **10**(2), 437.
23. Han, J., Kamber, M. and Pei, J., Data mining: concepts and techniques. *Data Mining: Concepts and Techniques*, Elsevier, New York, 2012.
24. Diebold, F. X. and Mariano, R. S., Comparing predictive accuracy. *J. Bus. Econ. Stat.*, 1995, **13**, 253–263.

Received 4 October 2022; revised accepted 2 February 2023

doi: 10.18520/cs/v124/i9/1065-1073