



Deep Learning Approaches for Potato Price Forecasting: Comparative Analysis of LSTM, Bi-LSTM, and AM-LSTM Models

A Praveenkumar^{1,2} · Girish Kumar Jha¹ · Sharanbasappa D. Madival¹ · Achal Lama¹ · Rajeev Ranjan Kumar¹



Received: 24 September 2024 / Accepted: 9 October 2024

© The Author(s), under exclusive licence to European Association for Potato Research 2024

Abstract

Accurate potato price forecasting is crucial for managing market volatility, optimizing supply chains, and improving decision-making for farmers and policymakers. This study compares the forecasting performance of six models: autoregressive integrated moving average (ARIMA), recurrent neural network (RNN), gated recurrent unit (GRU), long short-term memory (LSTM), bidirectional long short-term memory (Bi-LSTM), and attention mechanism-based LSTM (AM-LSTM) to predict weekly potato prices in India from January 2006 to December 2023. The AM-LSTM model outperformed all others, achieving the lowest RMSE of 95.50, MAE of 59.90, and MAPE of 8.95%, along with the highest R^2 of 0.85. This superior performance is attributed to the attention mechanism, which enables the model to dynamically focus on the most relevant time steps, enhancing both prediction accuracy and model interpretability. The results highlight that advanced deep learning models, particularly those integrating memory and attention mechanisms, significantly outperform traditional statistical methods for forecasting agricultural prices. Future research may explore hybrid models and real-time forecasting frameworks to further improve predictive accuracy and adaptability in volatile agricultural markets.

Keywords Attention mechanism · Deep learning · Gated recurrent unit · Long short-term memory · Potato price forecasting

✉ Girish Kumar Jha
girish.jha@icar.gov.in

¹ ICAR-Indian Agricultural Statistics Research Institute, New Delhi 110012, India

² The Graduate School, ICAR-Indian Agricultural Research Institute, New Delhi 110012, India

Introduction

Agricultural price forecasting plays a pivotal role in enhancing food security, stabilizing supply chains, and supporting policy-making, particularly in economies like India, where agriculture accounts for over 15% of the gross domestic product (GDP) and provides employment to nearly 50% of the population (Dev and Rao 2010). The volatile nature of agricultural markets, driven by weather conditions, market disruptions, and policy changes, can cause extreme price fluctuations, which negatively impact both producers and consumers. For farmers, price volatility creates uncertainty, complicating decisions on planting, harvesting, and storage, while for policymakers, it creates challenges in managing food inflation and ensuring market stability (Giller et al. 2021). Hence, reliable price forecasting becomes a critical tool for mitigating risk and ensuring socio-economic well-being in rural areas.

One of the key crops in India's agricultural landscape is the potato. As the second-largest producer of potatoes globally, with a production of nearly 53 Mmt in 2020 (www.fao.org), India relies heavily on potato farming for both domestic consumption and export. Potatoes are a staple food across various regions, playing an important role in household diets, particularly for lower-income groups (Alzakari et al. 2024). Additionally, they contribute significantly to India's agricultural exports, further underscoring the importance of potato price stability. However, potato prices are notoriously volatile, with fluctuations driven by multiple factors including climatic conditions, supply chain disruptions, pest infestations, and changes in demand. For instance, in 2020, disruptions caused by the COVID-19 pandemic, coupled with excessive rainfall, resulted in potato prices surging by over 400% in some markets (Kumari et al. 2024). Such extreme price swings highlight the necessity for reliable forecasting models to stabilize markets and protect farmers and consumers from the adverse impacts of volatility (Dunis and Huang 2002; Lee and Soo 2018).

Historically, agricultural price forecasting has relied heavily on traditional statistical models like the autoregressive integrated moving average (ARIMA) model (Box et al. 1970). ARIMA models have long been a cornerstone in time series forecasting due to their simplicity and ability to model linear relationships in stationary data. They are particularly effective in capturing short-term autocorrelations in datasets, making them useful for predicting short-term trends in relatively stable markets. However, despite ARIMA's widespread use, it suffers from significant limitations, especially in the context of agricultural price forecasting (Ray et al. 2023). First, ARIMA assumes that the underlying data is stationary, meaning that it expects constant variance and mean over time (Zhou et al. 2020). In agricultural markets, prices are often influenced by external shocks such as weather anomalies, pest infestations, or policy changes that lead to non-stationary, volatile data (Jha and Sinha 2014). Furthermore, ARIMA models are limited by their linearity assumption, which restricts their ability to capture non-linear patterns in complex datasets. The model's rigid structure makes it ill-suited for handling the sudden and often unpredictable shifts that characterize agricultural

markets. Consequently, while ARIMA models can provide baseline forecasts, they often fail to capture the more complex dynamics required for accurate long-term predictions in volatile markets like potatoes (Anjoy and Paul 2017).

Given the limitations of traditional statistical models like ARIMA, researchers have increasingly turned to machine learning (ML) techniques to improve forecasting accuracy in agricultural markets. Machine learning models, such as decision trees, support vector machines (SVM) (Kirange and Deshmukh 2016), and ensemble methods (e.g., random forests and gradient boosting), offer several advantages over traditional approaches. These models can handle non-linear relationships in data and do not require strong assumptions about the underlying data structure, making them more flexible for time series forecasting (Zhang et al. 2018). Additionally, ML models are capable of capturing complex interactions between multiple variables, providing a richer and more nuanced understanding of the factors influencing price movements (Jha and Sinha 2013).

However, despite their improved performance over traditional models, machine learning techniques face several challenges. ML models often require extensive feature engineering, where domain knowledge is needed to identify the most relevant variables for forecasting. Moreover, they can struggle with time dependencies in sequential data, as many machine learning models are not inherently designed to handle temporal patterns. While ensemble methods and SVMs have demonstrated some success in capturing non-linearity in agricultural price data, they lack the ability to learn long-term dependencies that are crucial in forecasting datasets with seasonal or cyclical trends, such as agricultural commodity prices (Makridakis et al. 2023).

To address these challenges, deep learning (DL) models, particularly recurrent neural networks (RNNs), have emerged as promising solutions for time series forecasting. Unlike traditional ML models, RNNs are designed to handle sequential data, making them inherently suited for tasks involving temporal dependencies, such as agricultural price forecasting (Jaiswal et al. 2022). RNNs maintain a hidden state that updates with each time step, enabling the model to remember information from previous time steps and use it to make predictions about future values. This ability to model sequential dependencies allows RNNs to capture both short-term and long-term patterns in time series data (Nayak et al. 2024b).

Despite their advantages, RNNs face a significant limitation: the vanishing gradient problem (Chung et al. 2014). When training RNNs over long sequences, the gradients used to update the model parameters become increasingly small, making it difficult for the model to learn long-term dependencies. This issue often results in suboptimal performance when applied to complex datasets with significant temporal depth (Gao et al. 2020). To overcome the limitations of standard RNNs, gated recurrent units (GRUs) (Lawi et al. 2022) and long short-term memory (LSTM) (Hochreiter and Schmidhuber 1997) networks were developed. GRUs simplify the RNN architecture by using gating mechanisms to control the flow of information through the network, allowing the model to selectively retain or discard information as needed (Nayak et al. 2024a, b). This not only addresses the vanishing gradient problem but also makes GRUs computationally more efficient than standard RNNs. LSTM, on the other hand, introduces memory

cells that explicitly manage the flow of information over long sequences, enabling the model to learn long-term dependencies without the vanishing gradient issue (Ismael and Rahaman 2020). Both GRU and LSTM have been successfully applied to a wide range of time series forecasting tasks, including financial and agricultural markets, where they consistently outperform traditional models in terms of accuracy and robustness (Kuber et al. 2022).

While LSTM and GRU networks offer significant improvements over traditional models, they still treat all time steps in a sequence equally. In datasets where certain time periods are more important than others, this uniform treatment can lead to suboptimal predictions. To address this, bidirectional LSTM (Bi-LSTM) was introduced, which processes the input sequence in both forward and backward directions, capturing dependencies from both the past and the future (Lu et al. 2021). This bidirectional approach allows Bi-LSTM to improve forecasting accuracy by providing the model with a more holistic view of the data. However, even Bi-LSTM can struggle with highly volatile datasets where specific time points disproportionately influence future values. This limitation led to the development of attention mechanism-based LSTM (AM-LSTM) models, which dynamically weigh the importance of different time steps in a sequence (Vaswani 2017). The attention mechanism allows the model to selectively focus on the most relevant parts of the input sequence, assigning greater importance to key time steps that have a stronger impact on future predictions. This makes AM-LSTM particularly well suited for datasets like agricultural prices, where external shocks such as weather events or policy changes can have outsized effects on prices (Chen and Ge 2019). Studies in financial and commodity markets have shown that AM-LSTM models consistently outperform traditional LSTM models by identifying the most relevant time steps in noisy datasets (Cui et al. 2023).

Despite the success of advanced models in various forecasting domains, there remains a significant gap in the literature concerning the comparative performance of traditional time series models (such as ARIMA) and deep learning models (including RNN, GRU, LSTM, Bi-LSTM, and AM-LSTM) specifically within the context of agricultural price forecasting. Furthermore, few studies have delved into the use of attention-based models like AM-LSTM for forecasting agricultural commodities, which are often characterized by high volatility and non-linearity, such as potato prices. This research aims to address this gap by providing a comprehensive comparative analysis of these models for potato price forecasting in India.

A primary contribution of this study is the detailed evaluation of advanced deep learning models, namely, LSTM, Bi-LSTM, and AM-LSTM, for forecasting potato prices. While prior research has employed machine learning models for agricultural forecasting, this study extends the current body of knowledge by offering a focused assessment of these three models, each distinguished by varying levels of complexity and capacity to handle large, non-linear datasets. The novelty of this research is reflected in the application of the AM-LSTM (attention mechanism long short-term memory) model, which has seen limited exploration in agricultural forecasting, particularly in commodity price prediction. By incorporating an attention mechanism, the AM-LSTM model dynamically assigns varying degrees of importance to

different time steps in the data sequence, resulting in improved predictive performance for highly volatile and non-linear price series, such as those of potatoes.

Additionally, this research highlights the presence of non-linear trends commonly found in agricultural price data and demonstrates how deep learning models, particularly Bi-LSTM and AM-LSTM, significantly outperform traditional methods such as ARIMA in addressing these complexities. The findings of this study provide new insights into model selection for forecasting applications, underscoring the critical role of advanced architectures like AM-LSTM in capturing long-range dependencies and subtle temporal relationships within noisy, volatile datasets. The contributions of this research offer practical implications for both researchers and policymakers looking to apply machine learning models for agricultural price forecasting. By illustrating the superior performance of the AM-LSTM model in a real-world application, this study paves the way for future exploration of attention-based models in the agricultural sector and beyond. The key contributions of this study are.

1. This study offered a comprehensive comparison between traditional (ARIMA) and advanced deep learning models (RNN, GRU, LSTM, Bi-LSTM, AM-LSTM) for potato price prediction, highlighting the strengths and limitations of each in capturing non-linear trends and seasonality in agricultural prices.
2. The study confirmed that the attention mechanism in LSTM models (AM-LSTM) significantly improves forecast accuracy.
3. By applying these models to potato price forecasting in India, this research advances the use of AI and machine learning for enhancing agricultural market stability and food security.

The structure of the paper is as follows: First, we provide an overview of the theoretical background, discussing the models utilized in the study, including ARIMA, RNN, GRU, LSTM, Bi-LSTM, and AM-LSTM. This is followed by the experimental analysis section, where we assess the performance of these models through rigorous evaluation and comparison against other forecasting techniques. The results from the real dataset are then presented in the “Experiment and Results” section, with a detailed discussion of the key findings and their implications. The paper concludes by summarizing the outcomes in the “Conclusion” section and offers a comprehensive list of references, which provide a strong foundation for future research in this field.

Theoretical Background

Autoregressive Integrated Moving Average (ARIMA) Model

The autoregressive integrated moving average (ARIMA) model is a widely used statistical approach for forecasting time series data, which assumes that future values are linear functions of past observations and random disturbances. For the ARIMA

model to be effective, it requires the time series data to be stationary, meaning the statistical properties such as mean and variance should be constant over time (Adebisi et al. 2014; Jha and Sinha 2014). If the data is not stationary, differencing or other transformations are applied to achieve stationarity. An ARIMA model is specified as ARIMA (p, d, q), where p is order of AR model, d is degree of differencing, and q is order of MA model. The ARIMA model can be mathematically expressed as follows:

$$\phi_p(B)\nabla^d Y_t = \theta_q(B)\varepsilon_t$$

where $\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p$ is the autoregressive operator of order p , $\phi_1, \phi_2, \dots, \phi_p$ is the corresponding autoregressive parameters, $\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q$ is the moving average operator of order q , $\theta_1, \theta_2, \dots, \theta_q$ is the associated moving average parameters, and $\nabla^d = (1 - B)^d$ is the differencing operator in order d to produce stationarity of the d th differenced data; in the above model equation, B is used as a backshift operator on Y_t and is defined as $B^i(Y_t) = Y_{t-i}$.

The process of fitting an ARIMA (p, d, q) model consists of several steps:

- i. Stationarity testing: Use statistical tests like augmented Dickey-Fuller (ADF) or Phillips-Perron (PP) to check for stationarity. If non-stationary, apply differencing for the mean or transformations for variance.
- ii. Model identification: Analyze autocorrelation (ACF) and partial autocorrelation (PACF) plots to determine AR (p) and MA (q) orders.
- iii. Parameter estimation: Estimate AR and MA parameters using maximum likelihood estimation (MLE) or least squares, optimizing to minimize forecast errors.
- iv. Diagnostic checking: Assess model fit using the Akaike information criterion (AIC) or Bayesian information criterion (BIC), and verify residuals are white noise via the Ljung-Box Q test.
- v. Forecasting: Apply the selected ARIMA model to predict future values and generate confidence intervals to measure forecast uncertainty.

Recurrent Neural Networks (RNNs)

Recurrent neural networks (RNNs) are a class of neural networks designed to handle sequential data, making them particularly effective for time series, speech recognition, natural language processing, and financial forecasting tasks. Unlike traditional feed-forward neural networks, which process inputs independently, RNNs feature recurrent connections that allow them to maintain a hidden state, capturing the sequential and contextual relationships between data points over time (Dunis and Huang 2002). This unique capability enables RNNs to process data with temporal dependencies, making them invaluable for tasks where context is critical.

Mathematically, the hidden state at time t , denoted as h_t , is a function of both the current input x_t and the hidden state from the previous time step h_{t-1} . This recursive process can be expressed as follows:

$$h_t = \sigma(W_h x_t + U_h h_{t-1} + b_h)$$

where W_h is the weight matrix applied to the current input x_t , U_h is the weight matrix applied to the previous hidden state h_{t-1} , and b_h is the bias term. The activation function σ , typically a *sigmoid* or *tanh*, introduces non-linearity to the network. However, RNNs face significant challenges such as the vanishing and exploding gradient problems, which occur during backpropagation through time (BPTT). These issues arise when the gradients either shrink or grow exponentially as they are propagated through many layers, making it difficult for the model to learn long-term dependencies in the data (Chen et al. 2023). To address these limitations, advanced architectures such as long short-term memory (LSTM) and gated recurrent units (GRUs) were developed, offering more robust solutions for long-term sequential modeling.

Long Short-Term Memory (LSTM)

The long short-term memory (LSTM) architecture was specifically designed to overcome the vanishing gradient problem, making it highly effective at capturing long-term dependencies in sequential data. LSTM networks feature a more complex cell structure that includes memory cells and multiple gating mechanisms to regulate the flow of information, enabling the network to retain relevant information over longer periods while forgetting irrelevant details (Chung et al. 2014). The key components of an LSTM cell are.

- Forget gate (f_t): Decide what information from the previous cell state should be discarded.
- Input gate (i_t): Determine which new information should be stored in the current cell state.
- Output gate (o_t): Control the output from the current cell state to the hidden state.

Mathematically, the operations in an LSTM cell are defined as follows:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$

$$h_t = o_t \odot \tanh(C_t)$$

where C_t is the cell state at time t and h_t is the hidden state. The gating mechanisms are controlled by weight matrices W , U , and bias terms b , while σ denotes the sigmoid function and $\odot$ represents element-wise multiplication. This architecture enables LSTMs to effectively learn and retain long-term dependencies, making them highly suitable for complex time series forecasting tasks with large datasets. However, LSTMs can be computationally expensive due to the multiple gates and memory cells involved. The architecture of the LSTM neural network is displayed in Fig. 1.

Gated Recurrent Unit (GRU)

The gated recurrent unit (GRU) is a streamlined variant of the LSTM, introduced to simplify the architecture while maintaining the ability to capture long-term dependencies. GRUs reduce the complexity of LSTMs by combining forget and input gates into a single update gate, and the cell state and hidden state are merged into a single state which is shown in Fig. 2. This simplification results in fewer parameters, which reduces the computational cost and memory usage, making GRUs faster to train than LSTMs (Kumar and Abirami 2021).

The operations within a GRU cell are defined as follows:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z)$$

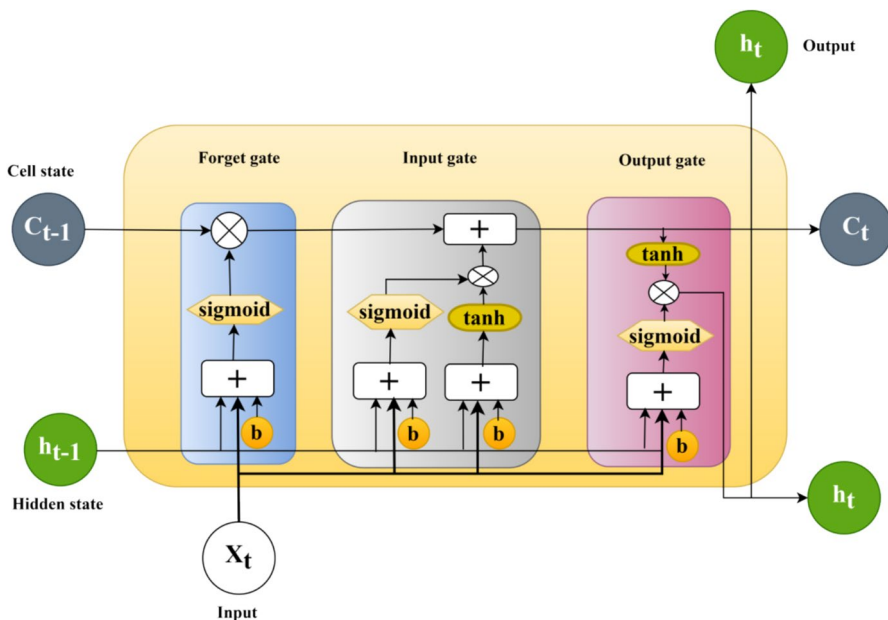


Fig. 1 Long short-term memory (LSTM) model architecture

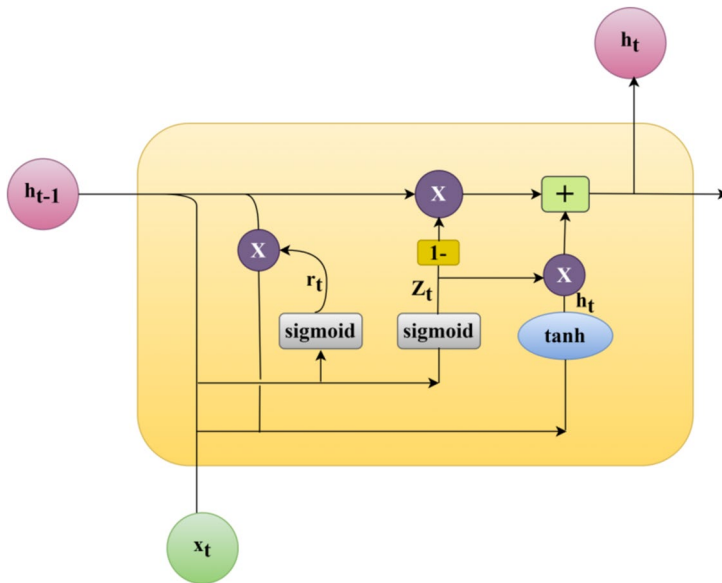


Fig. 2 Gated recurrent unit (GRU) model architecture

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r)$$

$$\tilde{h}_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

where z_t is the update gate that controls how much of the past information is retained and r_t is the reset gate, which determines how much past information is forgotten. The hidden state at time t , h_t , is a combination of the previous hidden state h_{t-1} and the candidate hidden state $\tilde{h}_t$. The weight matrices W and U , along with the bias terms b , control the flow of information. GRUs have been shown to perform comparably to LSTMs in many tasks but are computationally more efficient due to their simplified structure. While GRUs are generally faster and use less memory, LSTMs may offer slightly better accuracy on complex datasets, given their additional gating mechanisms (Chung et al. 2014). The choice between LSTM and GRU often depends on the specific application and the trade-off between computational efficiency and model complexity.

Bidirectional Long Short-Term Memory (Bi-LSTM)

The bidirectional long short-term memory (Bi-LSTM) model is an extension of the standard LSTM architecture designed to improve the learning of both past and future

context in sequential data. While traditional LSTM networks process data in one direction—either forward (from past to future) or backward (from future to past)—Bi-LSTM networks consist of two LSTM layers: one processing the sequence in the forward direction and another in the backward direction which is presented in Fig. 3. By combining the outputs from both directions, Bi-LSTM captures a more comprehensive understanding of the sequence, making it particularly useful for tasks that benefit from contextual information from both the past and future (Gomez et al. 2023).

In Bi-LSTM, each input sequence is passed through two LSTMs:

Forward LSTM: Process the sequence in the standard left-to-right direction.

Backward LSTM: Process the sequence in reverse, from right to left.

The final hidden state at each time step is a concatenation of the hidden states from both the forward and backward LSTMs. This allows the model to leverage information from both directions, improving the ability to capture long-term dependencies that may be missed in unidirectional models. This capability is particularly advantageous in tasks such as machine translation, speech recognition, and time series forecasting, where both past and future contexts are critical.

Mathematically, for each time step t , the forward LSTM computes:

$$\vec{h}_t = \text{LSTM}(x_t, \vec{h}_{t-1}, \vec{C}_{t-1})$$

While the backward LSTM computes:

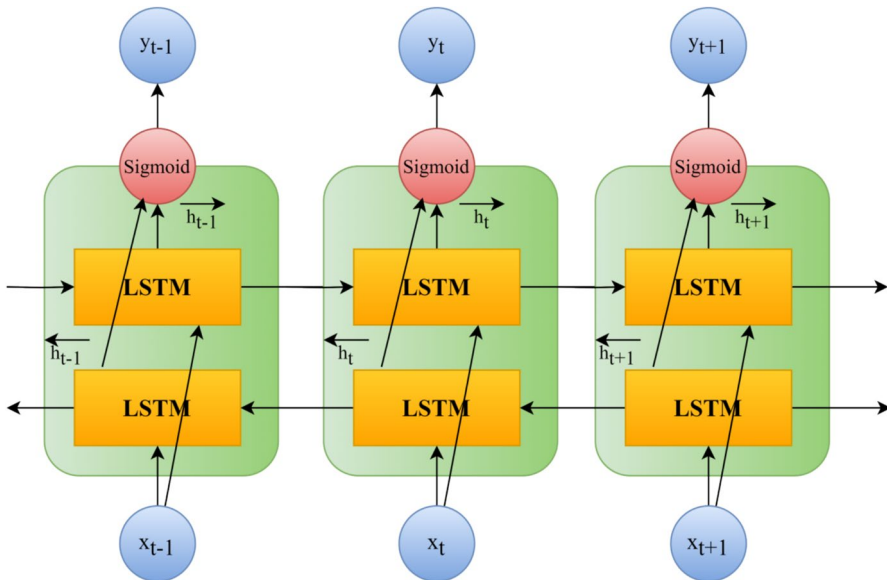


Fig. 3 Bidirectional long short-term memory (Bi-LSTM) model architecture

$$\overrightarrow{h}_t = \text{LSTM}(x_t, \overrightarrow{h}_{t+1}, \overrightarrow{C}_{t+1})$$

where x_t is the input at time t , $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$ are the hidden states for the forward and backward LSTMs, and $\overrightarrow{C}_{t-1}$ and $\overleftarrow{C}_{t+1}$ are the cell states for each direction. The final hidden state at each time step t is obtained by concatenating the forward and backward hidden states:

$$h_t = [\overrightarrow{h}_t; \overleftarrow{h}_t]$$

This concatenated hidden state contains information from both the past and future, improving the model's ability to capture dependencies in both directions.

Attention Mechanism Based Long Short-Term Memory (AM-LSTM)

In time series forecasting, particularly in financial markets, traditional long short-term memory (LSTM) networks often encounter challenges in effectively capturing the varying importance of different inputs over time. This limitation stems from LSTM's inherent tendency to treat all inputs equally, regardless of their individual significance to the prediction task. To overcome this, the integration of an attention mechanism into LSTM models has emerged as a significant improvement. The attention mechanism enables the model to focus selectively on the most relevant parts of the input sequence, thereby enhancing both predictive accuracy and interpretability. LSTM networks, a variant of recurrent neural networks (RNN), are designed to model both short-term and long-term dependencies in sequential data. LSTM achieves this by utilizing memory cells and gating mechanisms that regulate information flow across time steps, thus effectively addressing the vanishing gradient problem that hinders traditional RNN (Gu et al. 2022). However, despite their ability to manage long-term dependencies, LSTMs typically apply uniform weights across all time steps, limiting their effectiveness in domains such as financial time series, where the relevance of inputs can vary significantly. In these contexts, certain time steps may contribute more critically to the predictive outcome, necessitating a more flexible approach.

The attention mechanism was introduced to address this limitation by allowing the model to differentiate between important and less important time steps, rather than treating all hidden states equally. This mechanism assigns varying weights, or *attention scores*, to each time step, enabling the model to focus on the most informative parts of the sequence. By doing so, it ensures that crucial information is prioritized, which is particularly beneficial in time series forecasting, where external factors such as market volatility can cause certain time intervals to have a disproportionate impact on the prediction (Qiu et al. 2020; Peng et al. 2022).

In the attention mechanism, a weight α_t is assigned to each hidden state h_t , and a context vector c_t is produced as a weighted sum of the hidden states:

$$c_t = \sum_{i=1}^T \alpha_i h_i$$

The attention weights α_t , which reflect the relevance of each hidden state, are computed using an alignment score e_t :

$$e_t = v^T \tanh(W_a \cdot [h_t, h_{t-1}] + b_a)$$

$$\alpha_t = \frac{\exp(e_t)}{\sum_{k=1}^T \exp(e_k)}$$

In this formulation, e_t represents the alignment score, which quantifies the relationship between the current hidden state and the input at time step t . The softmax function normalizes these weights, ensuring that they sum to 1 and represent a probability distribution over the input time steps. This allows the model to focus dynamically on the most important time steps when generating predictions, enhancing both accuracy and interpretability. In an attention mechanism-based LSTM model, the attention layer is placed on top of the LSTM layers, allowing the model to weigh the contribution of each hidden state dynamically which is displayed in Fig. 4. The context vector c_t produced by the attention mechanism, along with the LSTM's hidden state h_t , is then used to generate the final output y_t :

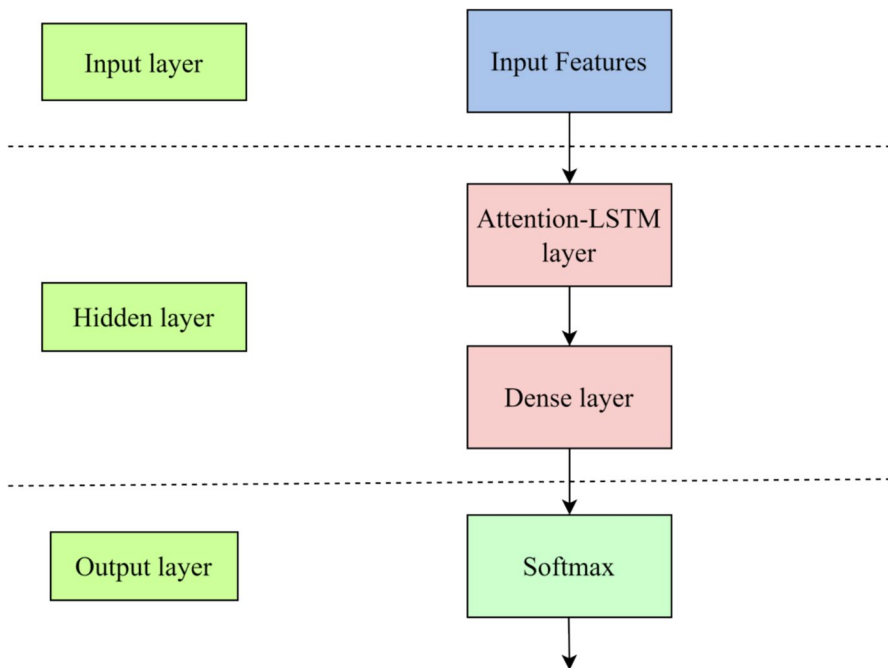


Fig. 4 Attention mechanism-based long short-term memory (AM-LSTM) model architecture

$$y_t = \text{softmax}(W_y \cdot [c_t, h_t] + b_y)$$

By incorporating the attention mechanism, the LSTM model is significantly enhanced. Rather than treating all inputs equally, the model learns to prioritize the most relevant inputs, which is crucial in complex time series data such as financial markets where conditions change frequently. Moreover, the attention mechanism improves the model's interpretability by assigning attention scores, which provide insights into which time steps (past events) are most influential in the prediction. This interpretability is particularly valuable for decision-makers in financial markets, as it allows for a deeper understanding of which factors are driving the forecasts, enhancing trust in the model's outputs.

Evaluation Criteria

The forecasting performance of the models will be evaluated using metrics such as mean absolute error (MAE), mean absolute percentage error (MAPE), root-mean-squared error (RMSE), and R^2 (Coefficient of determination). These performance measures are defined as follows:

$$\text{MAE} = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t|$$

$$\text{MAPE} = \frac{1}{h} \sum_{t=1}^h |e_t|/y_t \times 100$$

$$\text{RMSE} = \sqrt{\frac{1}{h} \sum_{t=1}^h (e_t)^2}$$

$$R^2 = \frac{\sum_{t=1}^n (\hat{y}(t) - \bar{y})^2}{\sum_{t=1}^n (y(t) - \bar{y})^2}$$

where n is the number of observations, y_t is observed value at time t , $\hat{y}_t$ is the predicted value at time t , h is the forecast horizon, $\bar{y}$ is the mean value of $y(t)$, and e_t is the residuals of the time series $e_t = y_t - \hat{y}_t$.

Experiments and Results

Dataset Description

In this study, the weekly potato price series data of the Agra market, Uttar Pradesh is used as experimental datasets. The price series was expressed in Rs per Quintal

(1 Quintal = 100 kg), spans from January 2006 to December 2023, and was obtained from the “Agmarknet” website (<https://agmarknet.gov.in/>), a reliable source of agricultural market information in India. In total, the dataset comprises 864 weekly observations. Table 1 provides the descriptive statistics for the potato price series, while Fig. 3 presents the corresponding time series plots, visually illustrating the non-stationary and non-linear behavior of the data. To further validate these properties, formal statistical tests for stationarity and linearity were conducted, with results presented in Tables 2 and 3.

Summary Statistics

The potato price series exhibits substantial variability, with prices ranging from Rs 205.83/Quintal to Rs 2835.00/Quintal. The standard deviation, a rudimentary

Table 1 Descriptive statistics of potato price series (Rs/Quintal)

Descriptive statistics	Potato prices series (Rs/Quintal)
Mean	824.24
Minimum	205.83
Maximum	2835.00
Skewness	1.38
Kurtosis	3.12
SD	407.54
CV	49.44
Jarque–Bera	626.74
1 Quintal = 100 kg	
SD standard deviation, CV coefficient of variation	

Table 2 Stationarity test of weekly potato price series

Price series	Augmented Dickey-Fuller test		Phillips-Perron test		Remarks
	Test statistic	<i>p</i> -value	Test statistic	<i>p</i> -value	
Potato	− 4.69	0.16	− 18.29	0.27	Non-stationary

Table 3 Non-linearity test of potato price series

Epsilon for close points	Potato price series		<i>p</i> -value
	Embedding dimensions		
	2	3	
0.5 σ	171.90	133.08	<0.001
1.0 σ	57.04	68.95	<0.001
1.5 σ	49.75	49.22	<0.001
2.0 σ	38.27	35.78	<0.001

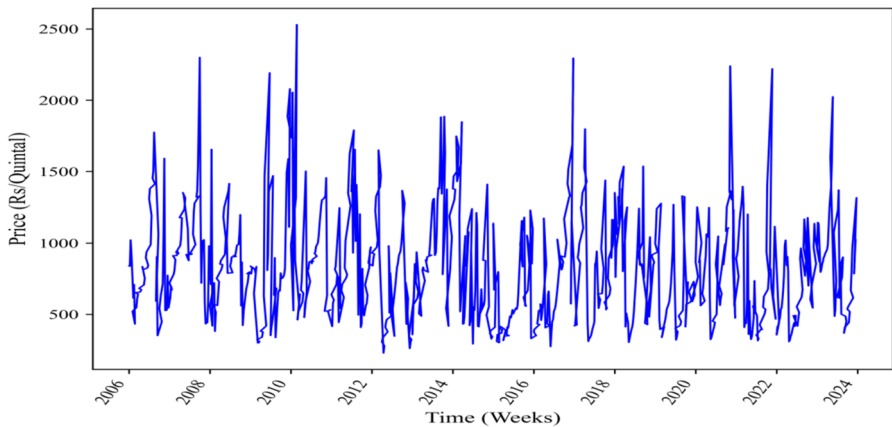


Fig. 5 Time series plot of weekly potato price series

measure of volatility, highlights the inherent price fluctuations in the data, suggesting a highly volatile market environment can be evident from Table 1 and Fig. 5. Moreover, the skewness and kurtosis values indicate that the series is positively skewed and leptokurtic, implying deviations from normality. This observation is statistically confirmed by the Jarque–Bera test, which rejects the null hypothesis of normal distribution, affirming the non-normal nature of the price series. Given the large dataset, a systematic approach is used to divide the series into training, validation, and testing subsets for model development and evaluation. Specifically, 90% of the observations are allocated to the training set, while the remaining 10% are reserved for testing. Additionally, 10% of the training data is further split for validation purposes, ensuring robust model tuning and preventing overfitting. This careful division facilitates the development of reliable predictive models, allowing for consistent evaluation across all stages of the study.

The primary objective of this study is to evaluate and compare the performance of traditional statistical models and advanced deep learning techniques for forecasting agricultural commodity prices, specifically potato prices. The focus is on assessing the effectiveness of ARIMA, RNN, GRU, LSTM, Bi-LSTM, and attention mechanism-based LSTM (AM-LSTM) models in capturing non-linearity, volatility, and long-term dependencies inherent in time series data. The study aims to identify the most suitable model for predicting volatile agricultural prices, providing insights into the advantages of incorporating memory and attention mechanisms in time series forecasting. The study was conducted using Python 3.9 within an environment optimized for machine learning and time series forecasting. Key libraries included *NumPy* for numerical computations, *Pandas* for data manipulation, *Scikit-learn* for pre-processing and evaluation, *TensorFlow/Keras* for building deep learning models, *Statsmodels* for ARIMA, and *Matplotlib/Seaborn* for data visualization. The models were developed and trained on a system with an Intel Core i7 processor, 16 GB RAM, and an NVIDIA GeForce GTX GPU for accelerated computations. This configuration ensured efficient model execution, particularly for deep learning

architectures like LSTM and AM-LSTM, enabling faster training and better handling of large datasets.

Test for Stationarity

To assess the stationarity of the potato price series, we applied the augmented Dickey-Fuller (ADF) (Dickey and Fuller 1979) and Phillips-Perron (PP) (Phillips and Perron 1988) tests. Stationary series are characterized by a constant mean and variance over time. The null hypothesis of the ADF test posits the presence of a unit root, implying that the series is non-stationary. A more negative ADF statistic indicates stronger evidence against the null hypothesis. Similarly, the null hypothesis of the PP test assumes that the series is integrated of order 1 (i.e. non-stationary). The results from both tests, summarized in Table 2, confirm the non-stationarity of the potato price series.

Test for Non-linearity

In this study, the Brock–Dechert–Scheinkman (BDS) test is employed to examine the non-linearity of the potato price series. The null hypothesis of the nonparametric BDS test is that the series is independently and identically distributed (i.i.d.). As shown in Table 3, the test results, calculated for embedding dimensions (lags) 2 and 3 and probability values from 0.5σ to 2.0σ , strongly indicate non-linearity in the series.

Data Pre-processing and Normalization

The dataset used in this study contains no missing values, eliminating the need for imputation. However, given the range of the data, as highlighted by the descriptive statistics, normalization is essential for ensuring that neural network models can be effectively trained and generalized well without introducing bias. Normalization rescales the data values between 0 and 1, preserving the underlying structure of the series while allowing for more stable and efficient model training. The normalization is conducted using the following transformation:

$$X_t' = \frac{X_t - X_{\min}}{X_{\max} - X_{\min}}$$

where X_t is the observed value at time t , $X_{\min}$ and $X_{\max}$ are the minimum and maximum values of the series, and X_t' is the normalized value. This method ensures that all data points are rescaled within the range $[0, 1]$, facilitating better convergence during model fitting. In this study, normalization was implemented using the Min-MaxScaler function from the *Scikit-learn* package in Python, ensuring consistency and accuracy in the data pre-processing phase.

Implementation of Forecasting Models

After confirming the non-stationarity and non-linearity of the potato price series and applying normalization, six models namely ARIMA, RNN, GRU, LSTM, Bi-LSTM, and AM-LSTM were implemented to forecast prices. The data was normalized using the MinMaxScaler to scale values between 0 and 1, ensuring optimal performance for neural network models. The dataset was split into training and testing sets in a 90:10 ratio, with 10% of the training data set aside for validation purposes. The ARIMA model, selected based on the lowest Akaike information criterion (AIC) and Bayesian information criterion (BIC) values, served as a baseline for comparison. ARIMA (2,1,2) was chosen as the best-fitting model, representing a traditional approach for forecasting time series data. However, ARIMA's inability to capture non-linearities in complex datasets required the use of more advanced deep learning models.

For the neural network models RNN, GRU, LSTM, Bi-LSTM, and AM-LSTM, the data was reshaped into a format that is necessary for processing sequential data. Hyperparameter tuning was conducted using grid search, optimizing key parameters such as the number of units in the recurrent layers, batch sizes (32, 64, and 128), and early stopping criteria with the patience of 10 epochs to prevent overfitting was presented in Table 4. All models were trained for a maximum of 100 epochs using the Adam optimizer and mean squared error (MSE) as the loss function. The RNN model was optimized with 64 units in the SimpleRNN layer and a batch size of 64. The model was trained with early stopping and was evaluated using RMSE, MAE, MAPE, and R^2 . Although RNNs can capture temporal dependencies, they are prone to vanishing gradient problems when dealing with long sequences, limiting their effectiveness in complex datasets like the potato price series. The GRU model was optimized with 50 units and a batch size of 128, improving computational efficiency while addressing the vanishing gradient problem more effectively than RNN. GRU's gating mechanisms allow the model to retain important information over long sequences, enhancing its ability to forecast price movements. The LSTM model, with 100 units in the LSTM layer and a batch size of 64, showed even better performance by leveraging its memory cells to capture long-term dependencies, crucial for agricultural price forecasting where past trends can significantly influence future prices. LSTM models are

Table 4 Hyperparameters of different models used for comparison

Hyperparameter	RNN	GRU	LSTM	Bi-LSTM	AM-LSTM
Inputs	128	128	128	128	128
Epochs	72	58	76	68	71
Batch size	64	64	64	128	64
Loss function	MSE	MSE	MSE	MSE	MSE
Activation function	ReLU	ReLU	ReLU	ReLU	ReLU
Optimizer	Adam	Adam	Adam	Adam	Adam

particularly effective for non-linear and volatile datasets, making them well suited for time series forecasting.

For the Bi-LSTM model, grid search identified an optimal configuration of 64 units in the bidirectional LSTM layer and a batch size of 32. Bi-LSTM extends the LSTM architecture by processing sequences in both forward and backward directions, allowing the model to capture dependencies from both past and future data points, which is especially valuable for forecasting in complex and volatile datasets. The AM-LSTM model, which combines LSTM with an attention mechanism, was optimized with 32 units and a batch size of 128. The attention mechanism dynamically assigns different weights to each time step, allowing the model to focus on the most relevant periods in the sequence, particularly during volatile price swings. This feature significantly improved both the predictive accuracy and interpretability of the model, making AM-LSTM the best-performing model in the study.

All models were evaluated using standard performance metrics, including root-mean-squared error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and R^2 . Among the models, AM-LSTM achieved the lowest RMSE, MAE, and MAPE, with the highest R^2 , confirming that integrating attention mechanisms allows for more accurate and interpretable price predictions. The Bi-LSTM and LSTM models also demonstrated strong performance, outperforming RNN and GRU in capturing the long-term dependencies and non-linear dynamics of the dataset. These findings highlight the advantages of deep learning models, particularly those incorporating bidirectionality and attention mechanisms, over traditional models like ARIMA in forecasting complex time series data.

The results in Table 5 clearly indicate that advanced deep learning models significantly outperform the traditional ARIMA model in forecasting accuracy. ARIMA, with a high RMSE of 160.50, MAE of 95.30, and MAPE of 29.25%, shows limited ability to capture the complexity and volatility of the dataset, reflected in its relatively low R^2 of 0.71. In contrast, deep learning models, particularly those with memory and attention mechanisms, exhibit superior performance. The LSTM-based models demonstrate a marked improvement in accuracy, with the RMSE progressively decreasing from 122.10 in RNN to 95.50 in AM-LSTM. AM-LSTM, the best-performing model, achieves the lowest MAE of 59.90, MAPE of 8.95%, and the highest R^2 of 0.85, owing to its ability to dynamically focus on the most relevant time steps via the attention mechanism. The reduction in error metrics from RNN

Table 5 Accuracy measures of various models on the testing dataset of potato price series

Models	MAE	MAPE	RMSE	R^2
ARIMA	95.30	29.25%	160.50	0.71
RNN	80.20	10.85%	122.10	0.75
GRU	75.40	9.95%	118.50	0.78
LSTM	70.80	9.60%	105.00	0.79
Bi-LSTM	65.78	9.38%	101.80	0.80
AM-LSTM	59.90	8.95%	95.50	0.85

The values in the bold indicate the best-fitted model

to AM-LSTM highlights the effectiveness of incorporating advanced features like attention mechanisms and bidirectional processing in time series forecasting. Overall, these findings confirm that deep learning models, especially those integrating attention mechanisms like AM-LSTM, provide significant improvements in both accuracy and robustness when dealing with complex, non-linear agricultural price data.

Discussion

This study provides a comprehensive comparison between traditional time series models and advanced deep learning techniques for forecasting agricultural commodity prices, specifically focusing on potato prices. The models tested include ARIMA, RNN, GRU, LSTM, Bi-LSTM, and AM-LSTM, each with varying capabilities in handling non-linearity, long-term dependencies, and volatility. Traditional models like ARIMA, while effective for linear and stationary datasets, often struggle with the non-stationary and non-linear nature of agricultural price data. This limitation becomes apparent in their inability to accurately capture the complex temporal patterns and volatile fluctuations present in such series. Despite its simplicity and ease of interpretation, ARIMA's reliance on linear assumptions restricts its predictive power in more dynamic environments.

Recurrent neural networks (RNNs), designed to capture sequential dependencies, offer improvements over ARIMA by modeling temporal relationships within the data. However, RNNs face challenges such as the vanishing gradient problem, which limits their capacity to learn long-term dependencies. This issue is addressed by the more advanced GRU and LSTM models. GRU, with its simplified architecture, balances efficiency and accuracy, making it well suited for modeling moderately complex time series. LSTM, with its memory cell structure, extends this capacity further by retaining long-term dependencies, thus offering significant improvements in forecasting accuracy, especially for volatile time series. The bidirectional LSTM (Bi-LSTM) further enhances performance by processing sequences in both forward and backward directions, allowing the model to capture information from both past and future time steps. This bidirectional approach leads to a better representation of the temporal dynamics, particularly in datasets with intricate, time-dependent relationships.

The most significant advancement in this study is observed with the attention mechanism-based LSTM (AM-LSTM). By integrating an attention layer, this model dynamically allocates different weights to time steps, enabling it to focus on the most relevant parts of the input sequence. This approach not only enhances predictive accuracy but also improves the interpretability of the model by highlighting which time steps contribute most to the forecast as shown in Figs. 6 and 7. The attention mechanism is particularly valuable in volatile markets, where specific periods, such as seasonal shifts or sudden market disruptions, may disproportionately influence future prices. Overall, the results demonstrate that while traditional models offer baseline performance, deep learning models especially

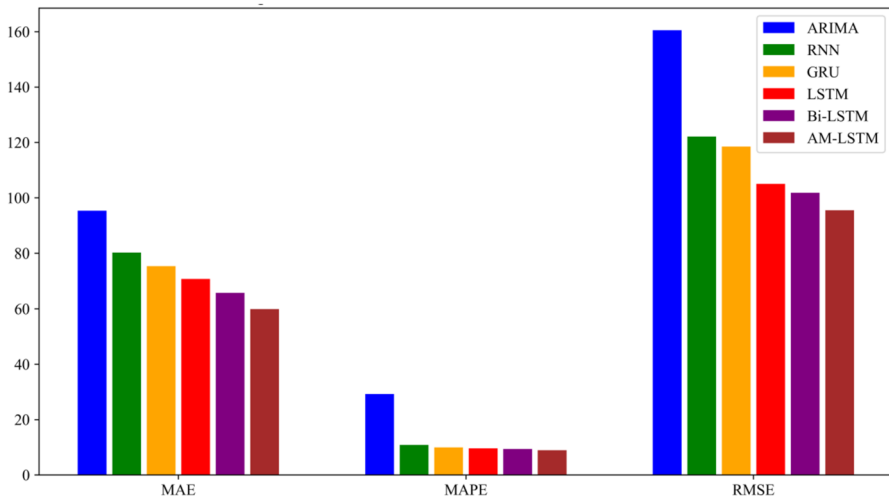


Fig. 6 Visualization of performance of models used in this study

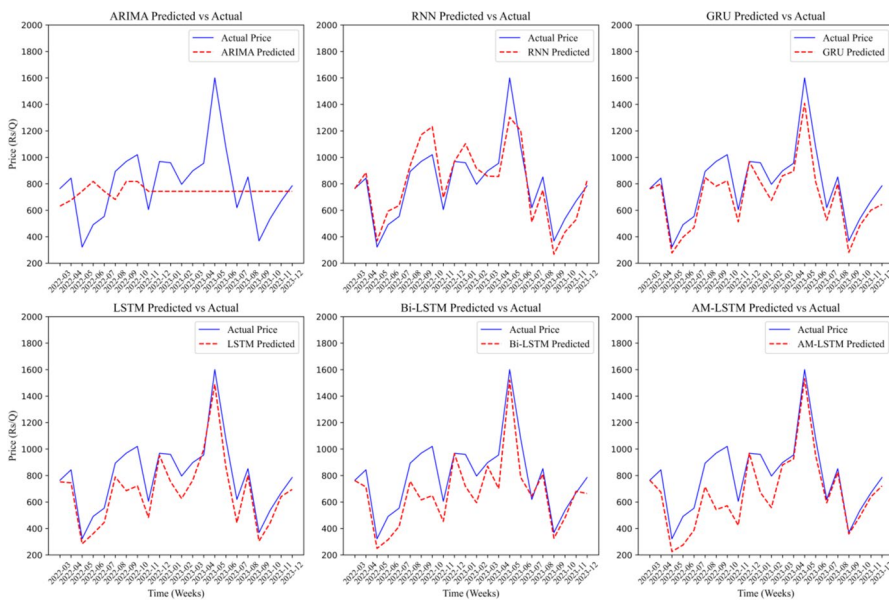


Fig. 7 Actual versus predicted values on testing data using different architectures

those incorporating memory and attention mechanisms are far more capable of capturing the complex, non-linear, and volatile nature of agricultural price data. These findings suggest that for forecasting in such contexts, advanced machine learning techniques provide a substantial advantage over classical methods.

Conclusions

This study highlights the superiority of deep learning models, particularly LSTM-based architectures, over traditional time series methods like ARIMA in forecasting volatile agricultural prices. While ARIMA struggles with the non-linearity and long-term dependencies present in the data, advanced models such as GRU, LSTM, Bi-LSTM, and AM-LSTM demonstrate significantly improved accuracy and robustness. The AM-LSTM with its integrated attention mechanism emerges as the most effective model, providing not only superior predictive performance but also enhanced interpretability by focusing on the most relevant time steps. Future research can explore hybrid models that combine deep learning approaches with econometric methods to further enhance forecasting accuracy in agricultural markets. Additionally, expanding the dataset to include more features, such as weather patterns, policy changes, and market conditions, could improve model performance. Further refinement of attention mechanisms could enhance the interpretability of models, aiding decision-making processes for stakeholders.

Acknowledgements The first author is grateful to the University Grants Commission (UGC) for offering the financial assistance and also to The Graduate School, ICAR-Indian Agricultural Research Institute, New Delhi for providing the requisite facilities to carry out this study.

Author Contribution All authors contributed to the study conception and design. Material preparation, data collection, and analysis were performed by Praveenkumar A, Girish Kumar Jha, and Sharanbasappa D Madival. The first draft of the manuscript was written by Praveenkumar A, Sharanbasappa D Madival, and Achal Lama, and Rajeew Ranjan Kumar commented on its improvement. Girish Kumar Jha reviewed the paper. All authors read and approved the final manuscript.

Data Availability The data can be accessed through the “AgMarknet” website.

Declarations

Ethics Approval and Consent to Participate The manuscript does not report on or involve the use of any animal or human data.

Consent for Publication The manuscript does not report on or involve the use of any animal or human data.

Conflict of Interest The authors declare no competing interests.

References

- Adebiyi AA, Adewumi AO, Ayo CK (2014) Comparison of ARIMA and artificial neural networks models for stock price prediction. *J Appl Math* 2014:614342. <https://doi.org/10.1155/2014/614342>
- Alzakari SA, Alhussan AA, Qenawy A-ST et al (2024) An enhanced long short-term memory recurrent neural network deep learning model for potato price prediction. *Potato Res*. <https://doi.org/10.1007/s11540-024-09744-x>
- Anjoy P, Paul RK (2017) Wavelet based hybrid approach for forecasting volatile potato price. *J Indian Soc Agric Stat* 71:7–14
- Box GEP, Jenkins GM, Reinsel G (1970) *Times series analysis forecasting and control*. Holden-Day San Francisco

- Chen C, Xue L, Xing W (2023) Research on improved GRU-based stock price prediction method. *Appl Sci* 13:8813. <https://doi.org/10.3390/app13158813>
- Chen S, Ge L (2019) Exploring the attention mechanism in LSTM-based Hong Kong stock price movement prediction. *Quant Financ* 19:1507–1515. <https://doi.org/10.1080/14697688.2019.1622287>
- Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. 1–9. Preprint at <https://arxiv.org/abs/1412.3555>
- Cui Z, Guo S, Zhou Y, Wang J (2023) Exploration of dual-attention mechanism-based deep learning for multi-step-ahead flood probabilistic forecasting. *J Hydrol* 622:129688
- Dev SM, Rao NC (2010) Agricultural price policy, farm profitability and food security. *Econ Polit Wkly* 45:174–182. <https://www.jstor.org/stable/40736698>
- Dickey DA, Fuller WA (1979) Distribution of the estimators for autoregressive time series with a unit root. *J Am Stat Assoc* 74:427–431. <https://doi.org/10.1080/01621459.1979.10482531>
- Dunis CL, Huang X (2002) Forecasting and trading currency volatility: an application of recurrent neural regression and model combination. *J Forecast* 21:317–354. <https://doi.org/10.1002/for.833>
- Gao S, Huang Y, Zhang S et al (2020) Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation. *J Hydrol* 589:125188. <https://doi.org/10.1016/j.jhydrol.2020.125188>
- Giller KE, Delaune T, Silva JV et al (2021) The future of farming: who will produce our food? *Food Secur* 13:1073–1099. <https://doi.org/10.1007/s12571-021-01184-6>
- Gomez W, Wang F-K, Amogne ZE (2023) Electricity load and price forecasting using a hybrid method based bidirectional long short-term memory with attention mechanism model. *Int J Energy Res* 2023:3815063
- Gu YH, Jin D, Yin H et al (2022) Forecasting agricultural commodity prices using dual input attention LSTM. *Agriculture* 12(2):256. <https://doi.org/10.3390/agriculture12020256>
- Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Ismael OM, Rahaman M (2020) Stock price trend forecasting using long short term memory recurrent neural networks. *Int J Sci Res Comput Sci Eng Inf Technol* 6(4):468–474. <https://doi.org/10.32628/cseit206474>
- Jaiswal R, Jha GK, Kumar RR, Choudhary K (2022) Deep long short-term memory based model for agricultural price forecasting. *Neural Comput Appl* 34:4661–4676. <https://doi.org/10.1007/s00521-021-06621-3>
- Jha GK, Sinha K (2014) Time-delay neural networks for time series prediction: an application to the monthly wholesale price of oilseeds in India. *Neural Comput Appl* 24:563–571. <https://doi.org/10.1007/s00521-012-1264-z>
- Jha GK, Sinha K (2013) Agricultural price forecasting using neural network model: an innovative information delivery system. *Agric Econ Res Rev* 26:229–239
- Kirange DK, Deshmukh RR (2016) Sentiment analysis of news headlines for stock price prediction. *An Int J Adv Comput Technol* 5(3):2080–2084. <https://doi.org/10.13140/RG.2.1.4606.3765>
- Kuber V, Yadav D, Yadav AK (2022) Univariate and multivariate LSTM model for short-term stock market prediction. Preprint at <https://arxiv.org/abs/2205.06673>
- Kumar JA, Abirami S (2021) Ensemble application of bidirectional LSTM and GRU for aspect category detection with imbalanced data. *Neural Comput Appl* 33:14603–14621. <https://doi.org/10.1007/s00521-021-06100-9>
- Kumari P, Vekariya P, Kujur SN et al (2024) Predicting potato prices in Agra, UP, India: an H2O AutoML approach. *Potato Res*. <https://doi.org/10.1007/s11540-024-09726-z>
- Lawi A, Mesra H, Amir S (2022) Implementation of long short-term memory and gated recurrent units on grouped time-series data to predict stock prices accurately. *J Big Data* 9:89. <https://doi.org/10.1186/s40537-022-00597-0>
- Lee CY, Soo VW (2018) Predict stock price with financial news based on recurrent convolutional neural networks. In: 2017 conference on Technologies and Applications of Artificial Intelligence (TAAI). IEEE, pp 160–165. <https://doi.org/10.1109/TAAI.2017.27>
- Lu W, Li J, Wang J, Qin L (2021) A CNN-BiLSTM-AM method for stock price prediction. *Neural Comput Appl* 33:4741–4753. <https://doi.org/10.1007/s00521-020-05532-z>
- Makridakis S, Spiliotis E, Assimakopoulos V et al (2023) Statistical, machine learning and deep learning forecasting methods: comparisons and ways forward. *J Oper Res Soc* 74:840–859. <https://doi.org/10.1080/01605682.2022.2118629>

- Nayak GHH, Alam MW, Avinash G et al (2024a) N-BEATS deep learning architecture for agricultural commodity price forecasting. *Potato Res.* <https://doi.org/10.1007/s11540-024-09789-y>
- Nayak GHH, Alam W, Singh KN et al (2024b) Modelling monthly rainfall of India through transformer-based deep learning architecture. *Model Earth Syst Environ.* <https://doi.org/10.1007/s40808-023-01944-7>
- Peng L, Wang L, Xia D, Gao Q (2022) Effective energy consumption forecasting using empirical wavelet transform and long short-term memory. *Energy* 238. <https://doi.org/10.1016/j.energy.2021.121756>
- Phillips PCB, Perron P (1988) Testing for a unit root in time series regression. *Biometrika* 75:335–346. <https://doi.org/10.2307/2336182>
- Qiu J, Wang B, Zhou C (2020) Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PLoS ONE* 15:1–15. <https://doi.org/10.1371/journal.pone.0227222>
- Ray S, Lama A, Mishra P et al (2023) An ARIMA-LSTM model for predicting volatile agricultural price series with random forest technique. *Appl Soft Comput* 149:110939. <https://doi.org/10.1016/j.asoc.2023.110939>
- Vaswani A (2017) Attention is all you need. *Adv Neural Inf Process Syst* 30. <https://doi.org/10.48550/arXiv.1706.03762>
- Zhang L, Wang S, Liu B (2018) Deep learning for sentiment analysis: a survey. *Data Min Knowl Discov* 8. <https://doi.org/10.1002/widm.1253>
- Zhou K, Wang WY, Hu T, Wu CH (2020) Comparison of time series forecasting based on statistical ARIMA model and LSTM with attention mechanism. *J Phys* 1631:12141. <https://iopscience.iop.org/article/10.1088/1742-6596/1631/1/012141>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.