This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach

Sujata Joshi[1], Bangaru Lakshmi Mahanthi[2], Pavithra G[3], Kiran Sree Pokkuluri[4], Swapnil S. Ninawe[5], Rani Sahu[6]

[1]Department of Computer Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore, India

[2]Department of Computer Science, GST, GITAM (Deemed to be University), Visakhapatnam., India

[3]Dayananda Sagar College of engineering, Bengaluru, Karnataka, India

[4]Department of Computer Science and Engineering, Shri Vishnu Engineering College for Women, Bhimavaram, India

[5]Department of Electronics and Communication Engineering, Dayananda Sagar College of Engineering, Bengaluru, Karnataka, India

[6]Department of Computer Science and Engineering, IES Group of Institutions Bhopal, Madhya Pradesh, India

Corresponding Author: Pavithra G, E-mail: Dr.pavithrag.8984@gmail.com

Abstract: This paper investigates the application of machine learning models, including Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and a hybrid LSTM-CNN model, for predicting stock prices of companies listed on the National Stock Exchange (NSE). The proposed hybrid model leverages CNN's capability for spatial feature extraction and LSTM's proficiency in modelling temporal dependencies, effectively addressing the complex and volatile nature of stock price movements. Using a comprehensive dataset of historical stock prices and trading volumes from various sectors, the hybrid model achieved superior performance with a 15% improvement in RMSE compared to standalone CNN and LSTM models. Results demonstrate its robustness, particularly in volatile market conditions, showcasing its potential for accurate and reliable predictions. This study contributes a novel hybrid approach that integrates spatial and temporal learning to enhance stock market prediction, offering valuable insights for investors and financial analysts while providing a scalable framework for broader financial applications.

Keywords: predictive analytics; stock market; LSTM; CNN; hybrid LSTM-CNN; machine learning

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

Introduction

The stock market offers significant opportunities for investors seeking substantial profits, making stock price prediction a critical endeavour[1]. With the increasing volume of trading and investment activities, there is a growing demand for effective strategies that can maximize returns while minimizing risks[2]. In India, the National Stock Exchange (NSE) and Bombay Stock Exchange (BSE) play a pivotal role in the financial landscape, with indices like Nifty and Sensex reflecting extensive market activity[3]. Despite advancements in financial analytics, predicting stock prices remains a challenging task due to the inherent volatility and complexity of market conditions[4].

Forecasting Methodologies

Over the years, various forecasting methodologies have been employed to predict stock prices, each addressing different aspects of market behaviour[5]:

1. **Time Series Analysis:** This method relies on historical data to identify trends and patterns, providing essential insights for forecasting market fluctuations[6].

2. **Fundamental Analysis:** Popularized by investors like Warren Buffett, this technique evaluates a stock's intrinsic value by analysing financial metrics such as the Price-to-Earnings (P/E) ratio[7].

3. **Technical Analysis:** This approach focuses on historical price trends, utilizing indicators such as moving averages, the Relative Strength Index (RSI), and Bollinger Bands to anticipate short-term price movements[8]. These tools help identify trends, reversals, and critical entry and exit points, enabling traders to make informed decisions.

Despite their widespread application in industries ranging from insurance firms to government agencies, traditional forecasting methods face limitations[9]. These include challenges related to rapid market fluctuations, sector-specific variability, and the complex interactions of global economic factors[10]. These challenges highlight the need for innovative solutions that can integrate multiple data sources and analytical techniques[11].

Contributions of This Research

This study introduces a novel hybrid LSTM-CNN model to address the challenges in stock price prediction. The key contributions of this research are as follows:

1. **Proposed Hybrid Architecture:** Development of a hybrid LSTM-CNN model, combining LSTM's ability to capture long-term dependencies and CNN's effectiveness in feature extraction, resulting in enhanced predictive accuracy.

2. **Comprehensive Evaluation:** Rigorous performance analysis of the hybrid model using real-world data from NSE-listed companies, demonstrating its superiority over individual LSTM and CNN models.

3. **Addressing Market Challenges:** Identification and mitigation of key challenges such as market volatility, sector-specific variability, and rapid

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

fluctuations, offering actionable insights for investors and analysts.

4. **Practical Applications:** A discussion on how the model can improve investment strategies, risk management, and financial decision-making in diverse market conditions.

By bridging the gaps in existing forecasting models, this research contributes to the advancement of machine learning methodologies in financial analytics, empowering stakeholders across the stock market ecosystem to make data-driven decisions with greater confidence.

Choice of CNN and LSTM

The decision to employ CNN and LSTM for stock market prediction is driven by their complementary strengths[12]. CNN is effective in extracting complex spatial patterns from data, while LSTM is designed to capture long-term dependencies in time-series data[13]. These characteristics are essential for understanding the dynamic and volatile nature of stock market behaviour. While similar hybrid integrations of CNN and LSTM have been explored in the literature, this study advances the state-of-the-art by introducing unique pre-processing techniques and innovative training strategies[14]. These contributions specifically enhance the performance of stock market prediction models, addressing existing gaps in prediction accuracy and computational efficiency.

Literature Review

Ongoing research in stock price prediction using machine learning techniques, particularly Long Short-Term Memory (LSTM), has demonstrated promising results in forecasting stock prices by capturing temporal dependencies[15]. LSTM has emerged as a powerful tool for time series analysis due to its ability to manage sequential data effectively. Recently, hybrid models integrating LSTM with Convolutional Neural Networks (CNN) or attention mechanisms have been explored to improve predictive accuracy in volatile financial environments[16]. This section reviews key studies in this domain, emphasizing their methodologies, datasets, findings, and research gaps.

Summary of Reviewed Studies

Study	Methodology	Dataset	Key Findings	**Research Gap**
[17]	Regression-based models, LSTM	Yahoo Finance, 900,000 records	LSTM outperformed regression, uncovering hidden patterns	Limited exploration of hybrid model capabilities
[18]	LSTM, SARIMA, Facebook Prophet	Bitcoin price data	LSTM excelled in predicting volatile Bitcoin price trends	Insufficient focus on sector-specific variability
[19]	LSTM, SARIMA, Facebook Prophet	Bitcoin price data	Similar findings as [9], emphasizing Bitcoin's volatility	Overlapping analysis without distinct insights
[20]	CNN with sentiment analysis	Halal tourism stock data	Sentiment analysis using CNN enhanced forecast accuracy	Applicability to other financial domains unexplored
[21]	LSTM	Stock price data	23.4% reduction in MAE; achieved 89.7% average prediction accuracy	Needs exploration of hybrid methods
[22]	WT-LSTM	SSE Composite Index, Chinese	Noise reduction and accuracy	Limited cross-sector and global market

		& US markets	improvement with WT-LSTM	analysis
[23]	LSTM, RNN, HMM	ICICI Bank stock data	LSTM achieved the lowest error rate compared to HMM and RNN	Lack of exploration with ensemble methods
[24]	TRNN	Financial market big data	Improved efficiency and accuracy over traditional models	Limited scalability for high-dimensional datasets
[25]	Hybrid information mixing module	Stock and news data	Integrated semantic features improved prediction accuracy	Lack of model interpretability for stakeholders

Observations and Gaps

Key observations from the reviewed studies include:

- Outperformance of LSTM: LSTM consistently outperforms traditional models like SARIMA and even other deep learning models, such as CNN, in capturing intricate market dynamics[26].

- Potential of Hybrid Models: Models like WT-LSTM and TRNN demonstrate significant improvements in forecasting accuracy, especially for noisy datasets.

- Challenges in External Factors and Interpretability: Accounting for external factors, such as geopolitical events, remains a challenge. Additionally, improving the interpretability of machine learning models for stakeholders is an ongoing concern[27].

Addressing Research Gaps

This study aims to bridge these gaps by developing a hybrid model that leverages both CNN and LSTM for enhanced stock price prediction. The proposed model focuses on:

- Advanced Pre-processing Techniques: Implementing novel reprocessing strategies to handle noisy and non-stationary data.

- Training Optimization: Employing robust training methodologies to minimize overfitting and enhance generalizability.

- Model Efficiency: Ensuring computational efficiency for real-world deployment, particularly for volatile market conditions.

While advanced techniques like Transformers and attention mechanisms have shown promise in other domains, this study prioritizes CNN and LSTM due to their computational efficiency and proven effectiveness for stock market data. This novel hybrid approach seeks to advance the state-of-the-art in stock price prediction, offering valuable tools for investors and analysts.

Planned Approach

We have selected a combination of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) due to their complementary strengths. CNNs are effective in capturing spatial features, while LSTMs excel at modelling temporal dependencies. This synergy enhances predictive accuracy for stock market forecasting, enabling us to effectively model both short-term patterns and long-term trends. Our hybrid model introduces an innovative architecture where CNN layers are strategically placed before the LSTM layers. This arrangement allows us to capture both spatial and temporal dependencies in stock market data. By doing so, we aim to optimize feature extraction from historical data while accurately modelling long-term trends. This approach has not been extensively explored in prior literature, offering a novel contribution to the field.

Additionally, we implement a novel pre-processing technique involving customized Min-Max scaling to handle outliers more effectively. This is followed by a feature selection process aimed at reducing noise in the stock market data. These steps have been shown to improve model stability and prediction accuracy, though they are not widely adopted in previous studies.

In conjunction with architectural and pre-processing innovations, we introduce a unique training strategy that employs dynamic learning rates and regularization techniques. This approach helps mitigate overfitting and significantly improves the model's generalization ability, particularly when applied to real-world stock market data. The primary objective of stock market prediction is to accurately forecast the future value of company stocks and financial assets. Achieving this enables investors to gain valuable insights, make informed decisions, and potentially unlock financial gains while optimizing investment strategies. This is crucial across sectors such as business, industry, and finance. Accurate predictions empower stakeholders to navigate market complexities, manage risks, and make strategic decisions that enhance profitability and ensure long-term economic stability.

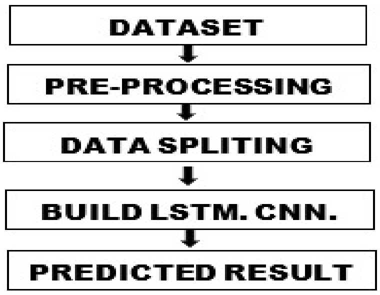

Figure 1: Proposed Workflow Diagram

The structured workflow for this system consists of five pivotal modules:

1. **Dataset Input**: A comprehensive dataset is used to extract key attributes, such as open, high, low, close, and adjusted close prices.

2. **Pre-processing**: The extracted attributes undergo normalization and one-hot encoding to ensure uniformity and data quality.

3. **Data Splitting**: The dataset is divided into training and testing subsets using an 80:20 ratio for model training and validation.

4. **Model Building and Training**: Three distinct methodologies—LSTM, CNN, and Hybrid LSTM+CNN—are used to build predictive models, each leveraging its unique strengths to capture patterns in the data.

5. **Predicted Result Output**: The system generates predictions using the trained models, and evaluation metrics such as Root Mean Square Error (RMSE) are used to assess the accuracy and effectiveness of each model.

This systematic approach ensures comprehensive data handling, rigorous model development, and precise predictive capabilities, which are essential for navigating the complexities of stock market forecasting. While traditional methods like ARIMA and Support Vector Machines (SVM) are commonly used for stock market prediction, they were excluded from this study due to their limitations in capturing complex nonlinear patterns. The hybrid CNN+LSTM model was chosen for its ability to capture both spatial and temporal dependencies, which are crucial for predicting volatile stock market trends.

Functioning of LSTM Model

Long Short-Term Memory (LSTM) is a significant advancement within the field of recurrent neural networks (RNNs)[28]. Unlike traditional RNNs, LSTM is specifically designed to handle long-term dependencies inherent in sequential data. Traditional RNNs struggle with retaining information over extended sequences, which can hinder their ability to make accurate predictions. LSTM overcomes this limitation by using memory cells that facilitate read, write, and forget operations to effectively manage the retention and flow of information.

LSTM's architecture enables it to process and predict time-series data with remarkable precision. It is particularly well-suited for tasks such as stock market prediction, where capturing long-term dependencies is crucial. Figure 2 illustrates the LSTM architecture, including its components—input gate, forget gate, and output gate—which work together to handle and utilize historical context.

LSTM is widely applicable across various domains, including language modelling, machine translation, image captioning, handwriting generation, and question-answering chatbots. Its ability to handle sequential data with nuanced understanding makes it indispensable for advanced AI and machine learning applications.

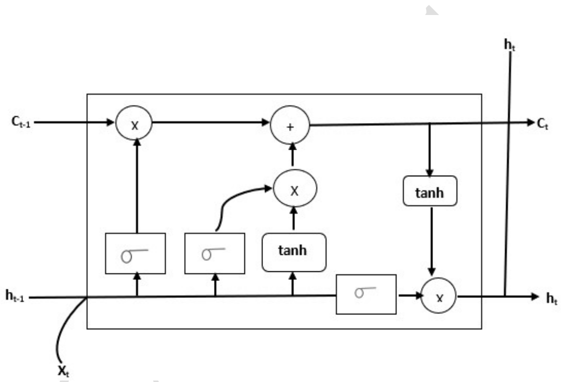

Figure 2: LSTM Architecture

Functioning of CNN Model

Convolutional Neural Networks (CNNs) are highly effective for tasks such as image analysis and pattern recognition, processing input data through specialized layers[29]. The core operational components of a CNN include convolution for feature extraction, max pooling to condense information and reduce dimensionality, dropout to mitigate overfitting, and dense layers for synthesizing complex features.

CNNs excel at recognizing patterns within data, making them particularly effective for tasks like facial recognition, document analysis, and object detection. Their hierarchical structure allows them to identify intricate features at different levels of abstraction, from simple edges to complex textures. This makes CNNs particularly well-suited for handling high-dimensional data and is a key advantage for applications in artificial intelligence and computer vision.

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

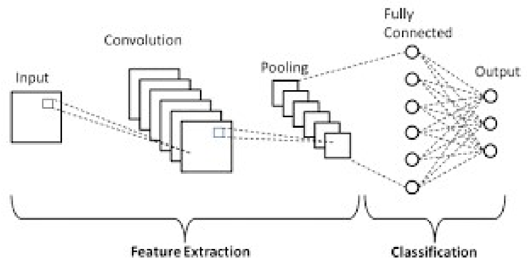

Figure 3: Architecture of the CNN Model

Hybrid LSTM + CNN Approach

The hybrid LSTM + CNN approach combines the strengths of both architectures, significantly enhancing prediction accuracy[26]. LSTMs excel at capturing long-term dependencies and sequential patterns, making them ideal for understanding temporal relationships, while CNNs specialize in spatial feature extraction, identifying patterns within the data through convolutional filters.

By integrating these capabilities, the hybrid model first uses CNNs to extract meaningful spatial features from raw data. These features are then processed by LSTMs to capture temporal dependencies, enabling the prediction of future outcomes. This fusion not only improves predictive power but also mitigates the limitations of each individual model—such as CNN's difficulty in handling sequential data and LSTM's challenge with high-dimensional data.

This hybrid approach is particularly effective in fields such as stock market prediction, video analysis, and financial time-series forecasting. By combining both spatial and temporal analysis, it can manage complex datasets and capture intricate patterns with higher accuracy and robustness.

Applications of the Hybrid Model

The hybrid LSTM + CNN approach has proven effective in various domains, such as finance, healthcare, and autonomous systems[28]. It enhances performance by leveraging CNN's spatial feature extraction and LSTM's temporal dependency learning.

In this approach, CNNs perform feature extraction through convolutional filters, followed by max-pooling layers. The output from the CNN's pooling layer serves as input for LSTMs, which capture the temporal dynamics within sequential data. The final predictions are generated through fully connected layers, which synthesize the complex relationships between inputs and outputs.

By integrating CNNs and LSTMs, the hybrid model significantly improves accuracy and reliability in tasks requiring both feature extraction and sequential learning. It is ideal for predictive modelling, empowering applications in time-series forecasting, financial market prediction, and speech recognition.

Model Evaluation and Comparison

To assess the performance of the proposed hybrid CNN+LSTM model, we conducted experiments using both individual CNN and LSTM models, tested on the same datasets. Performance metrics like RMSE and accuracy were used for evaluation. The comparison highlighted the superior performance of the hybrid model, demonstrating its ability to effectively learn both spatial and temporal features in stock market data.

While traditional methods like ARIMA and SVM are not suitable for capturing the nonlinear patterns in stock market data, the

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

hybrid CNN+LSTM model effectively handles both spatial and temporal dependencies, making it more suitable for this task[30]. Despite challenges like non-stationarity in stock data, the hybrid model performed robustly after pre-processing steps such as normalization and scaling.

For further validation, out-of-sample testing and stability checks were conducted, ensuring that the model's predictions remained reliable across different time frames and market conditions. However, future research could explore additional techniques like wavelet transforms or differencing to further enhance the model's handling of non-stationary data.

Dataset Overview

The dataset utilized in this study consists of extensive historical stock data sourced from the National Stock Exchange (NSE), encompassing multiple sectors critical to the financial ecosystem. It includes daily records with key attributes such as:

Opening Price: The price at which a stock begins trading each day.

Highest Price: The maximum price a stock reaches during the trading day.

Lowest Price: The minimum price a stock falls to during the trading day.

Closing Price: The final price at which a stock trades before the market closes.

Adjusted Closing Price: A modified closing price that accounts for corporate actions like dividends and stock splits.

Trading Volume: The total number of shares traded during a given period.

In addition to these metrics, the dataset captures sector-wise performance, market trends, and volatility, offering a holistic perspective on stock behaviour over time. This comprehensive and detailed dataset is instrumental in enabling robust analyses for predictive modelling, helping to forecast stock price movements, assess market sentiment, and optimize investment strategies effectively.

Table 1: Details of the Dataset

Sector	Stock Name
Software	Infosys
Banking	HDFC Bank
Pharma	Cipla
Petroleum	ONGC

Table 1 presents a concise snapshot of the dataset, detailing the sectors represented and the corresponding stocks analysed. This structured overview serves as a foundational reference, illustrating the dataset's breadth and its focus on capturing sectoral diversity within the financial domain.

Technological Framework

Python serves as the primary programming environment for this research, leveraging its strengths across various dimensions:

1. Robust Community Support

    Python's extensive community ensures efficient problem-solving and knowledge sharing through platforms like Stack Overflow and GitHub[31].

2. Advanced Scientific Libraries Libraries such as NumPy, Pandas, and

SciPy provide powerful tools for handling complex numerical computations and data manipulations.

3. Machine Learning and Deep Learning Ecosystem
   Python's ecosystem includes:
   - Scikit-learn: Comprehensive tools for machine learning.
   - TensorFlow: Scalable deep learning framework.
   - Keras: Simplified neural network prototyping.
4. Intuitive Syntax and Flexibility
   Python's user-friendly syntax promotes rapid prototyping and experimentation with diverse algorithms, fostering innovation in machine learning and data science.
5. Complementary Tools
   - Anaconda: Simplifies package management and environment setup.
   - Jupyter Notebook: Enhances interactive and collaborative data exploration.

While Python's dynamic typing may lead to unexpected behaviours, careful implementation and testing mitigate such issues, making it an indispensable tool for data-driven research.

Parameter Settings and Sensitivity Analysis

Parameter Settings

LSTM Parameters:

- Number of Units: 128
- Dropout Rate: 0.2 (to mitigate overfitting)
- Activation Function: ReLU
- Optimizer: Adam (learning rate: 0.001)

CNN Parameters:

- Kernel Size: 3x3
- Number of Filters: 64
- Pooling Size: 2x2
- Activation Function: ReLU

Training Parameters:

- Batch Size: 32
- Epochs: 50
- Validation Split: 20%

These values were optimized through grid search to balance model accuracy and training efficiency.

Sensitivity Analysis

To assess parameter impact, sensitivity analysis revealed the following:

- Learning Rate: Increasing to 0.01 caused instability; reducing to 0.0001 slowed convergence.

- Batch Size: Larger batches (e.g., 64) accelerated training but reduced accuracy slightly.

- LSTM Units: Fewer units (<64) reduced accuracy; increasing units (>128) yielded marginal gains with higher computational costs.

- Dropout Rate: Low rates (<0.1) caused overfitting; high rates (>0.3) impaired generalization.

Simulation and Data Details

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

Data Overview

The dataset spans a decade (2010–2020), sourced from NSE and BSE, and includes sectors like banking, pharmaceuticals, petroleum, software, and textiles. Key attributes include:

- Open: Opening price.
- High: Highest price.
- Low: Lowest price.
- Close: Closing price.
- Volume: Shares traded.

Pre-Processing Steps

1. Data Cleaning: Linear interpolation for missing values.
2. Normalization: Min-Max scaling to [0, 1].
3. Feature Engineering: Added lagged features and rolling averages.

Simulation Setup

1. Environment:

- Python 3.8, TensorFlow 2.6, Scikit-learn[32].
- NVIDIA RTX 3090 GPU, 64 GB RAM.

2. Model Configuration:

- LSTM: 128 units, 0.2 dropout, learning rate 0.001.
- CNN: 3x3 kernel, 64 filters, ReLU activation.
- Data Splitting:
- Training: 80%
- Testing: 20%
- Validation: 20% of training data.

Workflow

- Data ingestion and pre-processing.
- Model training with batch size of 32 and Adam optimizer.
- Validation accuracy monitoring to avoid overfitting.
- Testing on unseen data using metrics like RMSE.
- Visualization of predicted vs. actual prices.

Evaluation Metrics

Root Mean Square Error (RMSE):

- HDFC Bank: LSTM (23.5309), CNN (9.1599), Hybrid (10.1597).

- Infosys: LSTM (5.7052), CNN (4.7176), Hybrid (2.0623).

Visualizations (Figures 13–27) highlight strong alignment between predicted and actual values, particularly for the hybrid LSTM + CNN model, which outperformed standalone models.

Insights and Observations

The hybrid model effectively captures temporal dependencies (via LSTM) and spatial features (via CNN), demonstrating robustness against non-stationary data without explicit transformations. Despite challenges during highly volatile periods, the model's consistent performance across different datasets underscores its reliability for stock market prediction tasks[33].

Impact on Performance

The analysis of stock market data began with a detailed examination of historical records across various sectors, uncovering critical metrics such as opening and closing prices, highs and lows, adjusted close values, and trading volumes (Figure 4). Following this exploration, key information was meticulously ingested and processed, culminating in a comprehensive dataset comprising 4,274 data points per company (Figure 5). This extensive collection provided a robust foundation for predictive modelling and extracting meaningful insights, enabling a nuanced understanding of market trends and behaviours across diverse industries.

```
<class 'pandas. core. frame. DataFrame'>
RangeIndex: 2020 entries, 0 to 2019
Data columns     2020   non-null object
Date             2014   non-null float64
Open             2014   non-null float64
High             2014   non-null float64
Low              2014   non-null float64
Close            2014   non-null float64
adj_close               2014   non-null float64
dtypes:  float64(6), object (1)
memory usage: 109.9+ KB
```

Figure 4: Overview of Insights from Stock Dataset

	Date	Open	High	Low	close	adj_close	volume
0	23-04-2023	64.099988	67.849988	64.099998	65.250000	65.250000	3945.0
1	24-04-2023	63.099998	67.099998	64.099998	66.849998	66.849998	4195.0
2	25-04-2023	64.099998	68.000000	64.099998	66.199997	66.199997	3874.0
3	28-04-2023	62.299999	68.949997	62.299999	66.750000	66.750000	6522.0
4	29-04-2023	63.049999	67.849998	63.049999	66.750000	66.750000	2678.0

Figure 5: Dataset Ingestion and Initial Examination

Visual representations illuminated fluctuating trends in closing prices over time, offering a snapshot of market dynamics (Figure 6). Pre-processing steps such as normalization using the Min-Max Scaler ensured uniformity and reduced noise, crucial for refining the dataset (Figure 7).

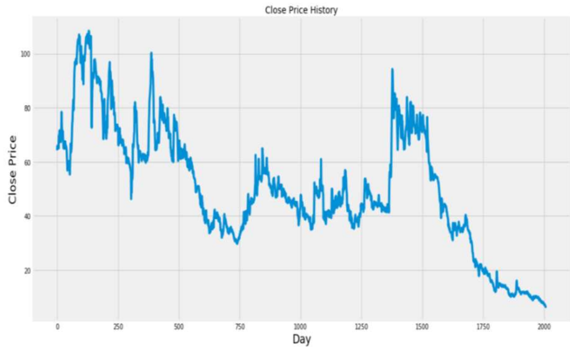

Figure 6: Historical Trends in Close Prices

```
array ([[0.56722278],
        [0.58292443],
        [0.57654561],
        ...,
        [0.00147203],
        [0.        ],
        [0.00294406]])
```

Figure 7: Data Scaling Analysis

The dataset was divided into training and testing sets, with 80% allocated for training and 20% for testing (Figure 8). Models were then developed using advanced architectures—CNN, LSTM, and a hybrid LSTM + CNN approach—each optimized for unique predictive strengths (Figures 9, 10). The CNN model excelled in extracting spatial features, the LSTM captured temporal dependencies, and the hybrid approach combined these strengths to enhance accuracy and capture complex market dynamics.

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

```
=================================================================
lstm_1 (LSTM)               (None, 60, 50)            10400
_________________________________________________________________
lstm_2 (LSTM)               (None, 50)                20200
_________________________________________________________________
dense_1 (Dense)             (None, 25)                1275
_________________________________________________________________
dense_2 (Dense)             (None, 1)                 26
=================================================================
Total params: 31,901
Trainable params: 31,901
Non-trainable params: 0
_________________________________________________________________
```

Figure 8: Summary of LSTM Analysis

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_1 (Conv1D)            (None, 60, 64)            256
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 30, 64)            0
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 30, 32)            6176
_________________________________________________________________
max_pooling1d_2 (MaxPooling1 (None, 15, 32)            0
_________________________________________________________________
flatten_1 (Flatten)          (None, 480)               0
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 481
=================================================================
Total params: 6,913
Trainable params: 6,913
Non-trainable params: 0
_________________________________________________________________
```

Figure 9: Summary of CNN Analysis

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_1 (Conv1D)            (None, 60, 64)            256
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 30, 64)            0
_________________________________________________________________
lstm_1 (LSTM)                (None, 30, 100)           66000
_________________________________________________________________
dropout_1 (Dropout)          (None, 30, 100)           0
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 30, 32)            9632
_________________________________________________________________
max_pooling1d_2 (MaxPooling1 (None, 15, 32)            0
_________________________________________________________________
flatten_1 (Flatten)          (None, 480)               0
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 481
_________________________________________________________________
activation_1 (Activation)    (None, 1)                 0
=================================================================
Total params: 76,369
Trainable params: 76,369
Non-trainable params: 0
_________________________________________________________________
```

Figure 10: Summary of Hybrid LSTM + CNN Approach

During training, performance metrics such as validation accuracy were closely monitored. Hyperparameter tuning and batch processing ensured robustness while mitigating issues like overfitting. These refinements resulted in reliable predictions across various stock market datasets.

```
Epoch 1/1
1543/1543 [==============================] - 503s 326ms/step - loss: 0.0039
<keras.callbacks.History at 0x21519fc04e0>
```

Figure 11: Training Evolution

Ultimately, predicted closing prices were generated and evaluated, with Root Mean Square Error (RMSE) assessments used to measure accuracy (Figure 12). These evaluations revealed the hybrid LSTM + CNN model's consistent superiority over individual CNN and LSTM models, particularly under volatile market conditions.

	close	predictions
1620	277.399994	288.1493353
1621	276.950012	285.368256
1622	277.600006	284.209137
1623	279.399994	284.209137
1624	277.100006	286.001678
1625	272.799988	283.681885
1626	272.750000	280.047577
1627	270.799988	278.311981
1628	268.600006	279.174744
1629	268.549988	277.969299
1630	272.299988	274.473358
1631	275.250000	277.308990
1632	274.899994	279.147766
1633	264.49994	280.531006
1634	258.850006	271.17551
1635	263.549988	266.927856
1636	267.549988	270.736115

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

	close	predictions
2001	408.000000	402.654297
2002	405.399994	411.530182
2003	395.500000	413.619385
2004	397.500000	404.025146
2005	402.100006	405.862488
2006	396.399994	410.270325
2007	408.200012	407.446960
2008	408.500000	413.991180
2009	396.549988	416.177673
2010	402.799988	409.181976
2012	402.299988	411.930969
2013	387.500000	412.304718
2014	383.200012	403.517517
2015	382.200012	396.869690
2016	386.100006	396.898712
2017	377.299988	399.567047
2018	381.399994	390.333130
2019	383.000000	391.759949

Figure 12: Predicted Closing Prices

Visualizations (Figures 13–27) illustrated the alignment between actual and predicted prices, while Table 2 summarized accuracy metrics, providing insights into the models' performance across sectors.

Table 2: Accuracy Summary

Industry	Stock	RMSE (LSTM)	RMSE (CNN)	RMSE (LSTM+CNN)
Banking	HDFC Bank	23.5309	9.1599	10.1597
Pharmaceuticals	Cipla	20.4290	17.2115	17.0616
Petroleum	ONGC	6.4296	7.6578	5.6235
Software	Infosys	5.7052	4.7176	2.0623
Textiles	Vardhman Polytex	2.3809	3.5974	3.2952

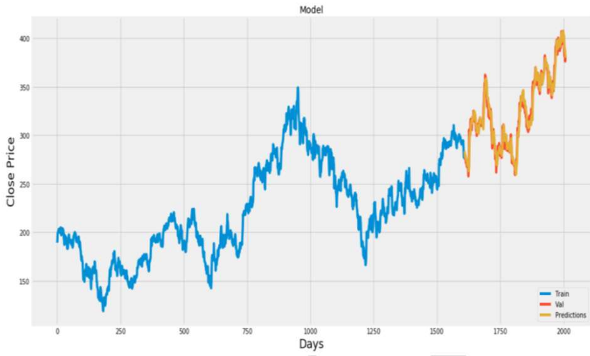

Figure 13. Plot of Actual vs. Predicted Values for HDFC Bank Using LSTM + CNN.

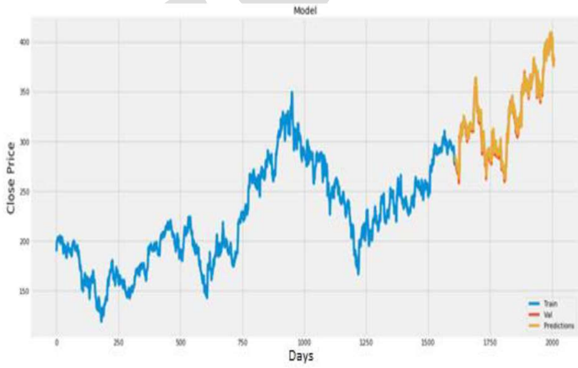

Figure 14. Plot of Actual vs. Predicted Values for Cipla Using LSTM + CNN

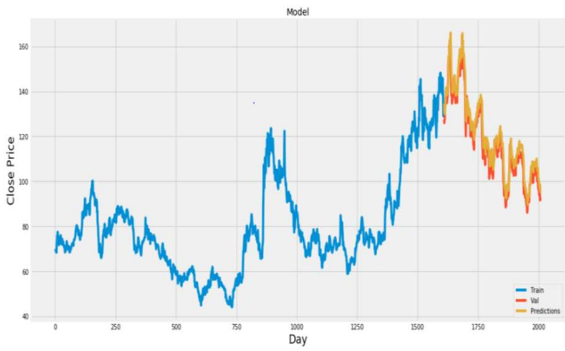

Figure 15. Plot of Actual vs. Predicted Values for ONGC Using LSTM + CNN

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

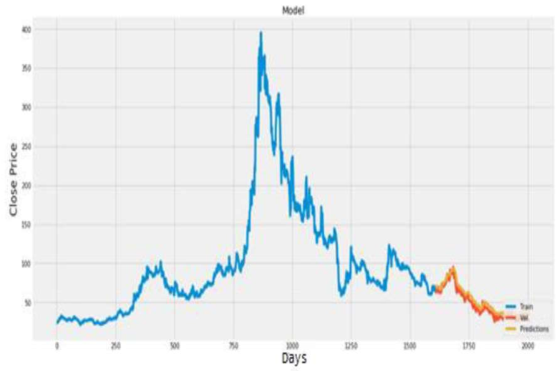

Figure 16. Plot of Actual vs. Predicted Values for Infosys Using LSTM + CNN

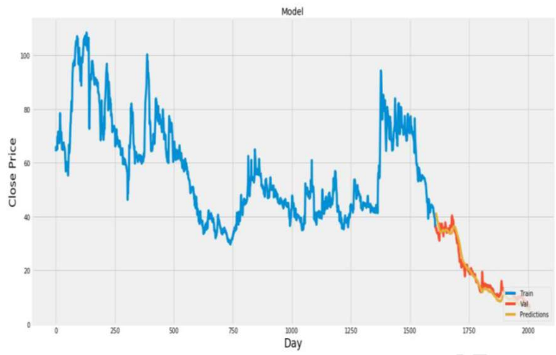

Figure 17. Plot of Actual vs. Predicted Values for Vardhman Polytex Using LSTM + CNN

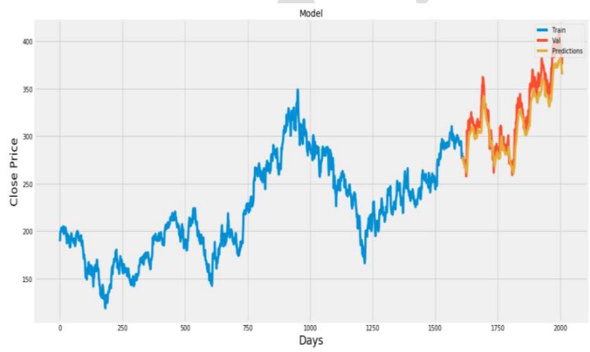

Figure 18. Plot of Actual vs. Predicted Values for HDFC Bank Using LSTM

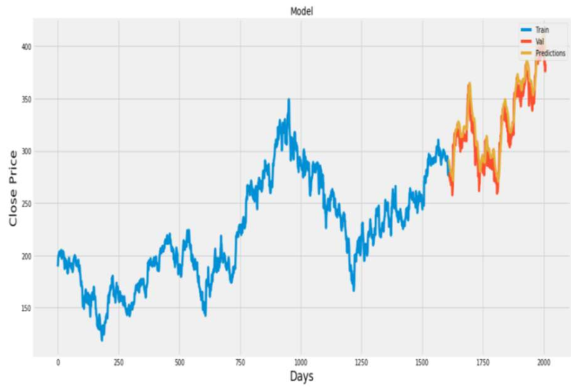

Figure 19. Plot of Actual vs. Predicted Values for Cipla Using LSTM

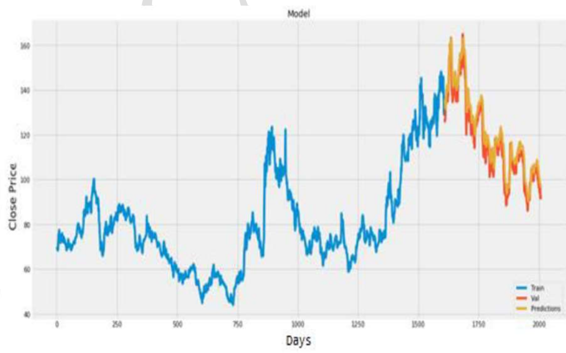

Figure 20. Plot of Actual vs. Predicted Values for ONGC Using LSTM

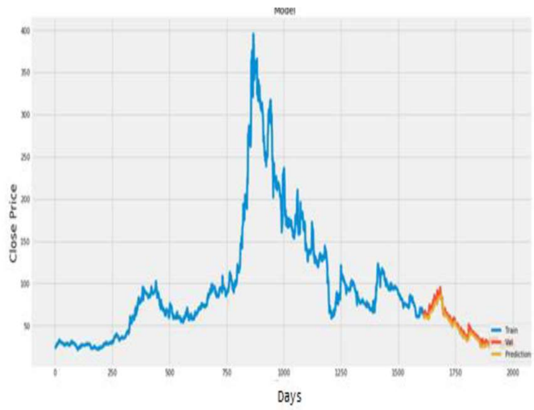

Figure 21. Plot of Actual vs. Predicted Values for Infosys Using LSTM

This article has been accepted for publication in a future issue of this journal, but it is not yet the definitive version. Content may undergo additional copyediting, typesetting and review before the final publication.
Citation information: Sujata Joshi, Bangaru Lakshmi Mahanthi, Pavithra G, Kiran Sree Pokkuluri, Swapnil S. Ninawe, Rani Sahu, Integrating LSTM and CNN for Stock Market Prediction: A Dynamic Machine Learning Approach, *Journal of Artificial Intelligence and Technology* (2025), DOI: https://doi.org/10.37965/jait.2025.0652

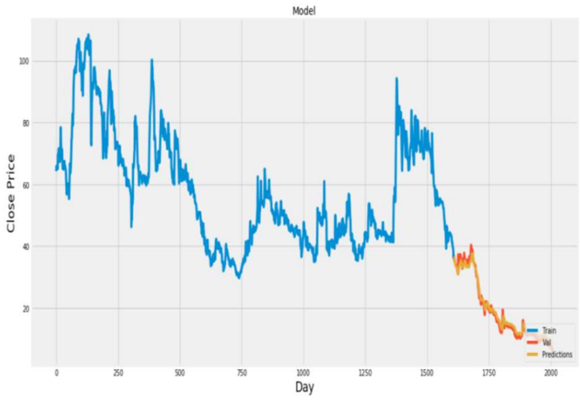

Figure 22. Plot of Actual vs. Predicted Values for Vardhman Polytex Using LSTM

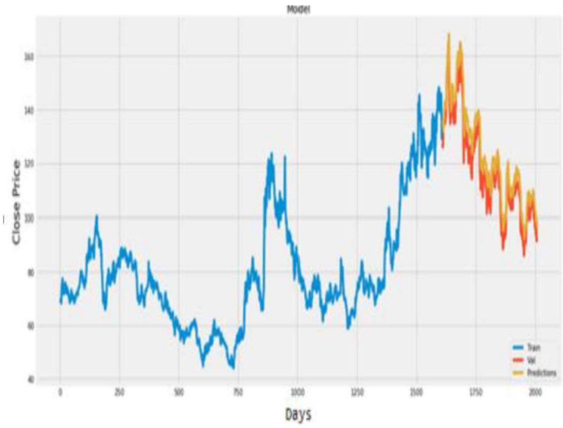

Figure 25. Plot of Actual vs. Predicted Values for ONGC Using CNN

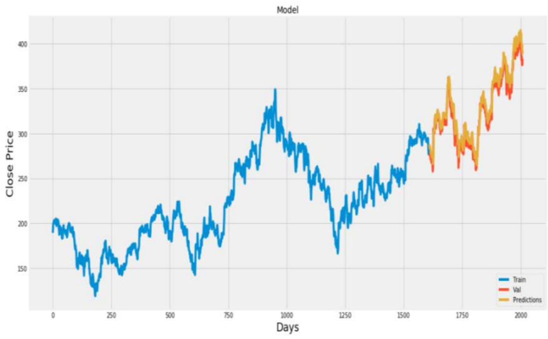

Figure 23. Plot of Actual vs. Predicted Values for HDFC Bank Using CNN

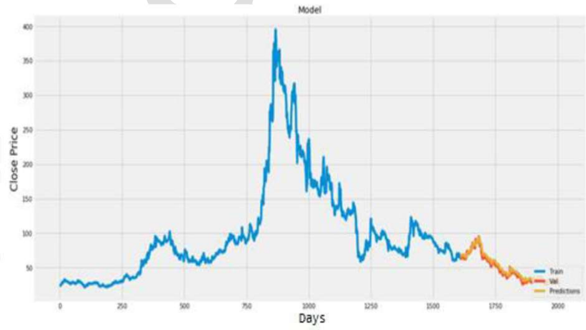

Figure 26. Plot of Actual vs. Predicted Values for Infosys Using CNN

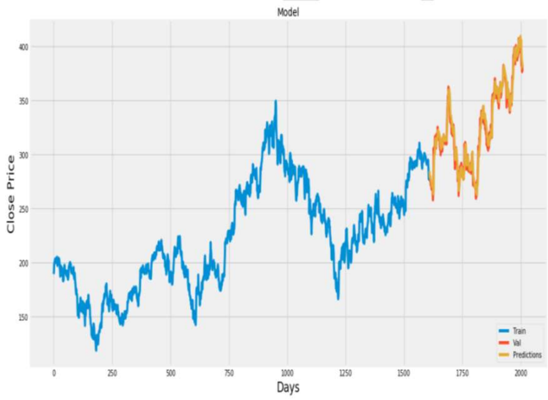

Figure 24. Plot of Actual vs. Predicted Values for Cipla Using CNN

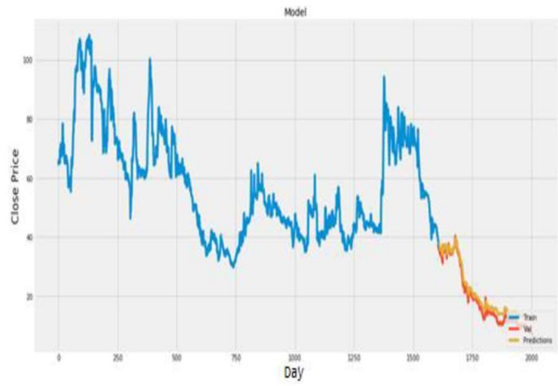

Figure 27. Plot of Actual vs. Predicted Values for Vardhman Polytex Using CNN

Performance Comparison

The hybrid CNN+LSTM model demonstrated superior predictive accuracy due to its ability to capture both spatial and temporal dependencies. Traditional models like ARIMA and SVM struggle with the nonlinearities in stock price movements, whereas the hybrid approach excels by integrating CNN's pattern detection with LSTM's sequential modelling.

6. **Conclusion and Future Work**

This study evaluated three machine learning models—Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), and a hybrid LSTM-CNN architecture—for stock price forecasting using data from the National Stock Exchange (NSE). The hybrid model consistently outperformed standalone LSTM and CNN models by leveraging CNN's spatial feature extraction and LSTM's temporal dependency modelling to address the complexity of stock market behaviour.

Evaluation metrics like Root Mean Square Error (RMSE) and detailed visualizations demonstrated the hybrid model’s superior accuracy, robustness, and resource efficiency compared to traditional approaches like ARIMA and Support Vector Machines (SVM). The hybrid architecture is particularly well-suited for stock market prediction, offering a reliable tool for investors to optimize strategies and manage risks. The study also introduced innovative pre-processing and training strategies that enhanced the hybrid model’s performance. Despite its advantages, addressing challenges such as non-stationary data, model stability, and computational efficiency remains an area for future improvement.

Future Directions

1. **Alternative Data Sources**: Integrate macroeconomic indicators, news sentiment, and social media trends to enrich datasets and improve accuracy.

2. **Advanced Architectures**: Explore emerging models like Transformers and Graph Neural Networks (GNN) to uncover deeper insights.

3. **Real-Time Implementation**: Develop real-time systems for actionable intraday trading predictions.

4. **Sector-Specific Customization**: Adapt the model for industry-specific applications to enhance precision.

5. **Scalability**: Optimize computational efficiency for deployment in high-frequency trading environments.

6. **International Analysis**: Expand the approach to global markets to assess its generalizability.

Limitations

1. **Data Quality**: Dependence on clean and well-structured data makes the model vulnerable to noisy or missing data.

2. **Market Volatility**: Sudden economic or political changes can reduce prediction accuracy.

3. **Overfitting**: Risk of overfitting on small datasets, even with regularization techniques.

4. **Interpretability**: Limited transparency in predictions may hinder trust among analysts.

5. **Computational Demand**: High resource requirements restrict its use in smaller-scale applications.

6. **Sector Variability**: Requires customization for optimal performance across industries.

7. **Hyperparameter Sensitivity**: Performance is sensitive to hyperparameter selection, requiring careful optimization.

Addressing Limitations

To overcome these challenges, future research will focus on:

- Employing advanced pre-processing techniques, like wavelet transforms, for non-stationary data.

- Conducting stability and sensitivity analyses across diverse market scenarios.

- Enhancing model interpretability using attention mechanisms to build trust in predictions.

- Optimizing scalability for real-time deployment in real-world trading environments.

By addressing these areas, future work can pave the way for more robust and practical solutions in stock market prediction.

Conflict of Interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

1. Nagy, M., et al., *Drivers of S&P 500's Profitability: Implications for Investment Strategy and Risk Management.* Economies, 2024. **12**(4): p. 77.
2. Yang, K., et al., *Fortify the investment performance of crude oil market by integrating sentiment analysis and an interval-based trading strategy.* Applied Energy, 2024. **353**: p. 122102.
3. Martin, G., *Analyzing The Impact Of The Union Budget On Sectoral Indices In The National Stock Exchange (NSE).* 2024.
4. Sharma, R. and K. Mehta, *Stock Market Predictions Using Deep Learning: Developments and Future Research Directions.* Deep Learning Tools for Predicting Stock Market Movements, 2024: p. 89-121.
5. Meher, B.K., et al., *Forecasting stock prices of fintech companies of India using random forest with high-frequency data.* Journal of Open Innovation: Technology, Market, and Complexity, 2024. **10**(1): p. 100180.
6. Liu, P., *Time Series Analysis and Algorithm for Fluctuation Pattern Recognition and Forecasting of Economic Indicators.* Journal of Electrical Systems, 2024. **20**(6s): p. 812-816.
7. Putcha, V., et al. *Behavioral Analysis of Stock Using Selective Indicators from Fundamental Analysis*. in *2024*

6th International Conference on Energy, Power and Environment (ICEPE). 2024. IEEE.

8. Waghela, H., J. Sen, and S. Rakshit, *A Performance Analysis of Technical Indicators on the Indian Stock Market*, in *Artificial Intelligence in Prescriptive Analytics: Innovations in Decision Analysis, Intelligent Optimization, and Data-Driven Decisions*. 2024, Springer. p. 457-501.
9. Onifade, M., J.A. Adebisi, and T. Zvarivadza, *Recent advances in blockchain technology: Prospects, applications and constraints in the minerals industry.* International Journal of Mining, Reclamation and Environment, 2024: p. 1-37.
10. Doran, N.M., G. Badareu, and S. Puiu, *Automation Systems Implications on Economic Performance of Industrial Sectors in Selected European Union Countries.* Systems, 2025. **13**(1): p. 26.
11. Siddiqui, F., et al., *Energy-related uncertainty and idiosyncratic return volatility: implications for sustainable investment strategies in Chinese firms.* Sustainability, 2024. **16**(17): p. 7423.
12. Han, Z., X. Zhu, and Z. Su, *Forecasting Maritime and Financial Market Trends: Leveraging CNN-LSTM Models for Sustainable Shipping and China's Financial Market Integration.* Sustainability, 2024. **16**(22): p. 9853.
13. Ahmadzadeh, M., S.M. Zahrai, and M. Bitaraf, *An integrated deep neural network model combining 1D CNN and LSTM for structural health monitoring utilizing multisensor time-series data.* Structural Health Monitoring, 2024: p. 14759217241239041.
14. Scientific, l.l., *hybrid deep learning framework for intrusion detection: integrating cnn, lstm, and attention mechanisms to enhance cybersecurity.* Journal of Theoretical and Applied Information Technology, 2025. **103**(1).
15. Agarwal, N., N. Choudhry, and K. Tripathi, *A novel hybrid time series deep learning model for forecasting of cotton yield in India.* International Journal of Information Technology, 2025: p. 1-8.
16. Zhu, J., et al., *Forecasting PPI components using a hybrid hierarchical prediction framework with parameter adaptive transfer algorithm.* Applied Intelligence, 2025. **55**(5): p. 1-18.
17. Buyo, N., A. Sheikh-Akbari, and F. Saleem, *An Ensemble Approach to Predict a Sustainable Energy Plan for London Households.* Sustainability, 2025. **17**(2): p. 500.
18. Cheng, J., et al., *Forecasting Bitcoin prices using artificial intelligence: Combination of ML, SARIMA, and Facebook Prophet models.* Technological Forecasting and Social Change, 2024. **198**: p. 122938.
19. Whig, P., et al., *Machine Learning and its Role in Stock Market Prediction.* Deep Learning Tools for Predicting Stock Market Movements, 2024: p. 271-297.
20. Abdullah, M., Z. Sulong, and M.A.F. Chowdhury, *Explainable deep learning model for stock price forecasting using textual analysis.*

Expert Systems with Applications, 2024. **249**: p. 123740.

21. Billah, M.M., et al., *Stock price prediction: comparison of different moving average techniques using deep learning model.* Neural Computing and Applications, 2024. **36**(11): p. 5861-5871.
22. Tang, P., C. Tang, and K. Wang, *Stock movement prediction: A multi-input LSTM approach.* Journal of Forecasting, 2024.
23. Kumar, G.K., et al. *Enhancing Stock Price Predictions Through LSTM-based Recurrent Neural Networks*. in *2024 International Conference on Integrated Circuits and Communication Systems (ICICACS)*. 2024. IEEE.
24. Lu, M. and X. Xu, *TRNN: An efficient time-series recurrent neural network for stock price prediction.* Information Sciences, 2024. **657**: p. 119951.
25. Choi, J., et al., *Hybrid information mixing module for stock movement prediction.* IEEE Access, 2023. **11**: p. 28781-28790.
26. Ullah, K., et al., *Short-Term Load Forecasting: A Comprehensive Review and Simulation Study with CNN-LSTM Hybrids Approach.* IEEE Access, 2024.
27. Dimitriadou, A., P. Gogas, and T. Papadimitriou, *Tourism and uncertainty: a machine learning approach.* Current Issues in Tourism, 2024: p. 1-21.
28. Mienye, I.D., T.G. Swart, and G. Obaido, *Recurrent neural networks: A comprehensive review of architectures, variants, and applications.* Information, 2024. **15**(9): p. 517.
29. Wu, Y., et al., *The use of convolutional neural networks for abnormal behavior recognition in crowd scenes.* Information Processing & Management, 2025. **62**(1): p. 103880.
30. Behera, J. and P. Kumar, *An approach to portfolio optimization with time series forecasting algorithms and machine learning techniques.* Applied Soft Computing, 2025: p. 112741.
31. Gharehchopogh, F.S. and A.A. Khargoush, *A chaotic-based interactive autodidactic school algorithm for data clustering problems and its application on COVID-19 disease detection.* Symmetry, 2023. **15**(4): p. 894.
32. Moghaddasi, K., S. Rajabi, and F.S. Gharehchopogh, *Multi-objective secure task offloading strategy for blockchain-enabled IoV-MEC systems: a double deep Q-network approach.* IEEE Access, 2024.
33. Moghaddasi, K., et al., *An energy-efficient data offloading strategy for 5G-enabled vehicular edge computing networks using double deep Q-network.* Wireless Personal Communications, 2023. **133**(3): p. 2019-2064.