StockSense: AI-Powered Prediction with Real-Time News Intelligence

Aashish Bagmar
Department of Artificial Intelligence and Data Science
Anantrao Pawar College of Engineering & Research Pune, India

Mangal Wagh
Department of Artificial Intelligence and Data Science
Anantrao Pawar College of Engineering & Research Pune, India

Jitendra Musale, PhD
Department of Artificial Intelligence and Data Science
Anantrao Pawar College of Engineering & Research Pune, India

Eshwari Bhandkar
Department of Artificial Intelligence and Data Science
Anantrao Pawar College of Engineering & Research Pune, India

Simrah Awati
Department of Artificial Intelligence and Data Science
Anantrao Pawar College of Engineering & Research Pune, India

Aarya Jadhav
Department of Artificial Intelligence and Data Science
Anantrao Pawar College of Engineering & Research Pune, India

ABSTRACT
Stock market forecasting is a difficult task because of the volatility and unpredictable nature of financial data. To address this is-sue, we present an AI-based Stock Prediction Model that predicts future stock prices by analyzing historical data, trading volume, and price movements. Our model captures temporal dependence and finds connections within financial time series using machine learning techniques. Additionally, a sentiment analysis module processes financial news and classifies public sentiment as positive, negative, or neutral, which provides context to the market. Our model is designed as a web application using Python and Flask, making it user-friendly and allowing for analysis and visualization of continuously updated data. Our experimental results show that the model connects past market behavior to future trends, enabling better-informed decisions. In summary, our work emphasizes the value of AI-based predictive analytics in finance for making informed and timely investment choices.

General Terms
Artificial Intelligence, Machine Learning, Financial Analytics

Keywords
Stock Market Prediction, LSTM, Sentiment Analysis, FinBERT, Financial Forecasting

1. INTRODUCTION
The stock market plays a key role in today's world. It shows both an organization's financial health and investor trust [1]. However, predicting stock prices is one of the toughest challenges in financial analysis. This difficulty comes from price volatility, unpredictable behavior, and sensitivity to many outside factors such as political events, economic policies, and public opinion [2], [3]. For investors, traders, and policymakers, predicting price changes accurately is essential for making timely decisions, managing risk, and maximizing returns [4].

Stock prediction models use traditional methods to generate fore-casts, including statistical and econometric models. These meth-ods depend on historical numerical data. While they help us see trends, they do not capture important and complex time-based relationships in financial data. Using Artificial Intelligence (AI) and Machine Learning (ML) represents the next step in tackling these challenges [5]. These technologies allow systems to learn from large, complex data sets and reveal hidden relationships that explain market behavior [6]. This work aims to create an AI-based Stock Prediction Model that analyzes historical stock data, such as opening and closing prices, trading volume, and volatility [7]. It predicts whether the market will trend upward or downward.

Additionally, the system includes a sentiment analysis module that processes live financial news and media [8]. This approach merges quantitative information from historical stock trends with qualitative insights from current market sentiment, forming a two-layered structure [9].

The entire model can be built into a web application using Python and Flask [10]. This design supports real-time data collection, auto-mated analysis, and visual dashboards, making it easy for users. By combining sentiment analysis with predictive modeling, the system aims to serve as a smart decision support tool for investors. It enhances their understanding of the market, helps with short-term predictions, and boosts confidence in their strategies [11].

2. RELATED WORK
Technology plays an essential role in modern financial analysis and decision-making [1]. The rapid growth of data from global markets, social media, and financial news has made manual analysis inefficient and biased [2]. Artificial Intelligence (AI) and Machine Learning (ML) enable automated, data-driven analysis at scale, im-proving traditional forecasting methods [3], [4].

AI algorithms can process large volumes of historical and real-time stock data to identify hidden patterns and predict market movements [5], [6]. Machine learning models such as regression models, recurrent neural networks (RNNs), and transformer-based models capture complex temporal patterns that are often missed by traditional statistical techniques [7], [8].

Natural Language Processing (NLP) enhances stock market analysis by interpreting financial news and investor sentiment [9], [10]. Integrating sentiment analysis with price prediction helps capture psychological and emotional factors that influence market behavior [11].

Traditional forecasting systems mainly rely on statistical models like linear regression and ARIMA, which depend on historical numerical data and fail to adapt to dynamic market conditions influenced by external events [1], [2]. Although some machine learning-based systems improve accuracy, many still ignore real-time qualitative information or analyze sentiment separately, resulting in in-complete predictions [3], [4]. Furthermore, most existing systems lack real-time prediction, visualization, and interactive web-based interfaces [5]. These limitations highlight the need for an AI-driven framework that combines historical price analysis with real-time sentiment analysis in a unified and scalable system [6].

3. LITERATURE REVIEW

Recently, researchers have shown increasing interest in Artificial Intelligence (AI) and Machine Learning (ML) for financial analytics [1]. Many studies indicate that data-based predictive systems outperform traditional models in identifying patterns and trends in stock markets [2].

Zhang et al. [3] studied deep learning frameworks such as Long Short-Term Memory (LSTM) networks for predicting stock prices. They found that these models effectively manage temporal dependencies in sequential financial data. Patel et al. [4] explored various machine learning approaches, without focusing on algorithm mechanics. They compared Support Vector Machines (SVM), Random Forest, and Artificial Neural Networks (ANN). They created stock forecasts using hybrid ensemble methods that improved predictive performance. In addition to numerical data, some studies looked at the impact of investor sentiment and financial news [5]. For instance, Bollen et al. [6] showed that public mood captured from social media platforms like Twitter is closely related to stock price movements. Furthermore, Kumar et al. [7] investigated a sentiment-aware price prediction model that used natural language processing to assess sentiment credibility. They discovered that the tone of news coverage correlated with market volatility patterns and influenced short-term trading strategies.

Recent research has also examined transformer-based models, like BERT and FinBERT, which perform sentiment analysis with context awareness in financial texts [8]. These models demonstrate greater effectiveness and accuracy in understanding financial terms compared to traditional lexicon based methods [9].

Current literature aids in understanding and analyzing components like price prediction and sentiment analysis. However, there remains a gap in combining these elements into a single system that merges both quantitative and qualitative data streams [10]. The pro-posed work seeks to build on past studies to develop an AI-guided predictive model. This model will base its assessments on historical market data while creating an “anchoring” system, supported by a live sentiment analysis component. This approach will enhance interpretability and provide a clearer view of market activity [11]

4. PROPOSED SYSTEM

The proposed system aims to fix the issues with current stock prediction models by combining quantitative and qualitative data analysis into a single AI-driven framework [4], [5]. It merges historical stock market data with real-time sentiment analysis of financial news to provide a more accurate and context-aware prediction of market movements [6]. The system has two main modules: the Stock Prediction Module and the Sentiment Analysis Module. The Stock Prediction Module uses machine learning algorithms to process and analyze historical data, such as opening and closing prices, trading volume, and volatility [7]. It finds patterns in these datasets to predict whether the market trend is likely to go up or down.

The Sentiment Analysis Module examines the qualitative side of financial data. It collects live news articles, headlines, and media updates, then processes them with natural language techniques to evaluate the overall market sentiment as positive, negative, or neutral [8]. This sentiment data combines with the numerical predictions to improve the final output’s accuracy [9].

The system’s architecture supports smooth interaction between these modules. Data gathered from APIs, including financial feeds and news sources, is first preprocessed and then sent through their respective modules. The outputs are merged and analyzed by a decision-making layer that interprets both data streams to create a complete market forecast [10].

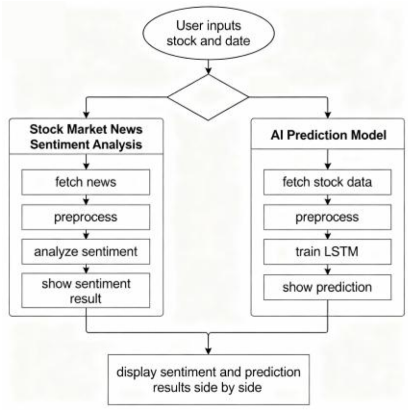
To enhance usability, the entire system is available as a web-based application built with Python and Flask [11]. This interface offers real-time visualizations, dynamic charts, and dashboards that display both market trends and sentiment indicators. Users can monitor predictions continuously and gain valuable insights through an interactive and user-friendly platform [12].

Overall, the proposed system addresses the gaps in existing forecasting methods by combining predictive analytics with sentiment analysis [13]. This integration allows for more reliable, data-driven, and timely investment decisions, helping investors and analysts understand market behavior from both numerical and emotional perspectives [14].

Fig. 1 illustrates the complete workflow of the proposed AI-based stock prediction and sentiment analysis system [1].

The system consists of two parallel modules: the Stock Market News Sentiment Analysis module and the AI Prediction Model module [2], [3]. Both modules start their operation based on user input, where the user specifies the stock symbol to be analyzed.

In the Sentiment Analysis branch, the system fetches the latest financial news related to the selected stock. The collected news data is then preprocessed to remove irrelevant or noisy text through tokenization, stop-word removal, and normalization [4]. Next, the system analyzes the sentiment of each headline using a transformer-based Natural Language Processing model like BERT or FinBERT [5]. The sentiment results are classified as positive, negative, or neutral [6]. These results reflect the overall market mood for that particular stock. Finally, the system shows the sentiment result to the user.



In the AI Prediction Model branch, the system pulls historical stock data from sources like Yahoo Finance through the yfinance API [7]. It preprocesses the data using statistical methods to eliminate inconsistencies and normalize the values [8]. A trained model that uses an algorithm like LSTM or Random Forest predicts the future stock price trend for the specified period [9],

[10]. Once processed, the system presents the prediction result to the user.

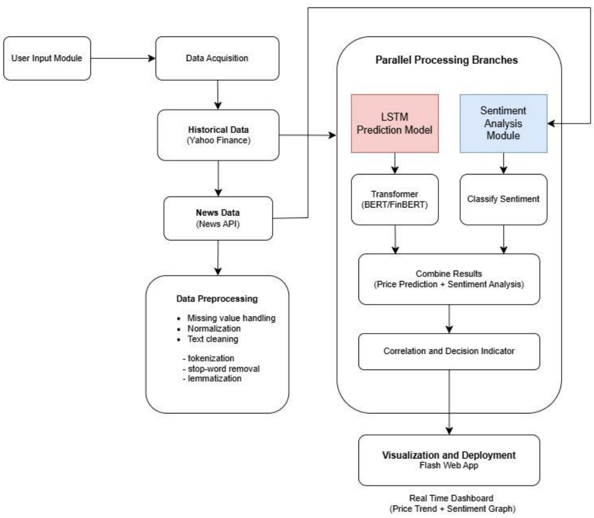
Both module outputs are displayed side by side on the web interface [11]. The user can view the predicted price trend along with the result of the news sentiment analysis. This side-by-side representation helps investors connect public sentiment to potential stock performance, supporting more informed decision-making [12].

5. RESEARCH METHODOLOGY

The research methodology details the entire process, starting with designing and developing an AI-based system to predict stock prices using historical data. It also involves analyzing live financial news for sentiment trends [1]. The proposed methodology has five main stages: data collection, preprocessing, model development, prediction, and deployment. An overview of the system architecture is provided [2].

Overview of the Project The goal of this research is to create a web-based platform with two main functions: The AI Stock Prediction Model predicts future stock values based on historical price data [3]. The Sentiment Analysis Module categorizes real-time financial news as positive, negative, or neutral to show overall market sentiment [4]. Both components work independently, but together help users understand market trends and price changes more clearly.

Data Collection and Preprocessing Data comes from two main sources: Historical Data: Extracted from APIs such as yfinance and Alpha Vantage, this includes open, close, high, low, and trading volume information [5]. News Data: Collected through financial news APIs for real-time sentiment analysis [6]. Preprocessing involves handling missing values, normalizing numeric fields, and converting time-series data into a suitable format for the prediction model. The text data is processed through tokenization, stop-word removal, and lemmatization using NLTK and Transformers libraries.



A. LSTM-Based Prediction Model We use the LSTM model for its ability to learn patterns in sequences [7]. Unlike other regression models, LSTM networks have internal states that allow them to re-member past information for an extended time, which is crucial for forecasting financial time series. The steps to build the prediction model include:

Splitting the dataset into training and testing sets. Normalizing stock price data for stable gradient updates.

Feeding sequences of past stock prices into the LSTM network. Using the trained model to predict future prices for specific time intervals.

Assessing model accuracy with metrics like MAE and RMSE [8]. The model provides dynamic short-term price predictions that update with new data.

B. Sentiment Analysis Module This module works alongside the LSTM model and uses pre-trained transformer models like Fin-BERT or DistilBERT to process real-time financial headlines [9]. The model classifies sentiment as positive, negative, or neutral, pro-viding a qualitative view of the market. These classifications are then visualized with predicted prices to help users connect public sentiment with actual market movements [10].

C. System Deployment The complete system is deployed as a web application using Flask as the backend framework [11]. The de-ployment process includes:

Integrating the trained LSTM and sentiment models into the web server.

Setting API calls to pull data in real time and update automatically. Providing real-time stock predictions, sentiment graphs, and correlation results within an interactive dashboard. This ensures users have the latest market predictions and sentiment analysis for informed decision-making [12].

6. ALGORITHM

The proposed system uses a structured pipeline that combines data preprocessing, machine learning, and natural language processing techniques [1].It gathers historical stock data through APIs, cleans it, and inputs it intoan LSTM for time series prediction [2]. At the same time, it analyzes live financial news to gauge market mood by using FinBERT-based sentiment classification [3]. The system then combines these two outputs and displays them through a Flask web dashboard for real-time trend forecasting and support for investor decisions [4].

Table 1: Algorithm for Proposed AI-Based Stock Prediction and Sentiment Analysis System

Step	Description
1	Initiate the process and set up the necessary Python li- braries, including NumPy, Pandas, Scikit-learn, TensorFlow, and Flask.
2	Gather historical stock data using the yfinance API. Collect financial news data in real-time from a reliable API or web scraping tool like BeautifulSoup or Scrapy.
3	Preprocess each dataset by handling missing values, normaliz- ing numbers, and cleaning text data through tokenization and lemmatization.
4	Generate relevant features such as moving averages, daily re- turns, and volatility. Calculate correlation among stock indica-tors.
5	Train the AI prediction model such as LSTM, Random Forest, or Linear Regression using the preprocessed historical data.
6	Evaluate model performance using metrics like MSE, RMSE,

	and R2 Score.
7	Run the NLP-based sentiment analysis model (BERT/FinBERT) on financial news headlines and clas-sify them as Positive, Negative, or Neutral.
8	Combine results from both modules for overall analysis and generate predictions.
9	Visualize results including stock trends, sentiment charts, and correlation heatmaps through the Flask web dashboard.
10	Terminate the process.

Each vector $\mathbf{x}_t$ represents features such as open, close, high, low, and trading volume. For each time step t, the LSTM cell performs:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \quad (2)$$

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f), \quad (3)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \quad (4)$$

$$\tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c), \quad (5)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \quad (6)$$

$$h_t = o_t \odot \tanh(c_t), \quad (7)$$

where i , f , o denote input, forget, and output gates; c is the cell t t t t state;

7. MATHEMATICAL MODEL

7.1 Notation

Let:

—t : Discrete time (trading days).

—H : Prediction horizon (e.g., $H = 1$ for next-day prediction).

—$\mathbf{x} = [x^{(1)}, x^{(2)}, \ldots, x^{(m)}]^\top$: Feature vector of m stock indica-

h_t is the hidden state; $\sigma(\cdot)$ is the sigmoid function; and $\odot$ denotes element-wise multiplication.

In (2)–(7), i_t, f_t, and o_t represent the input, forget, and output gates of the LSTM cell, respectively. These gates control the amount of information written, retained, and exposed from the memory cell. The term $\tilde{c}_t$ denotes the candidate cell state computed from the current input x_t and the previous hidden state h_{t-1}. The cell state c_t stores the long-term memory, while h_t is the hidden state output at time step t. The matrices W_i, W_f, W_o, W_c and bias vectors b_i, b_f, b_o, b_c are trainable parameters learned during model optimization. The final hidden state h_T is passed through a fully connected layer to predict the stock value:

$$\hat{y}_{t+H} = W_y h_T + b_y. \quad (8)$$

Model parameters $\theta = \{W_i, W_f, W_o, W_c, W_y, b_i, b_f, b_o, b_c, b_y\}$ are optimized using the Mean Squared Error (MSE) loss:

This enables the network to learn nonlinear relationships and temporal dynamics within stock prices.

7.2 Sentiment Classification Model

Each financial news headline $n_{t,i}$ is encoded into a semantic vector using a transformer encoder $E(\cdot)$:

$$\mathbf{z}_{t,i} = E(n_{t,i}). \quad (9)$$

A sentiment classifier g_ϕ maps $\mathbf{z}_{t,i}$ to probabilities over sentiment classes $C = \{\text{pos, neu, neg}\}$:

$$\mathbf{p}_{t,i} = g_\phi(\mathbf{z}_{t,i}) = \text{softmax}(W\mathbf{z}_{t,i} + b). \quad (10)$$

Training minimizes the cross-entropy loss:

The aggregate daily sentiment score S_t is obtained as:

t t t t

In (**??**), $S_t \in [-1, 1]$ denotes the aggregated daily sentiment score

tors at time t.

—$N_t = \{n_{t,1}, n_{t,2}, \ldots, n_{t,k_t}\}$: Set of k_t financial news headlines observed on day t.

—y_t : Target stock value (e.g., closing price or log-return).

—$\hat{y}_{t+H}$: Predicted stock value for time $t + H$.

—S_t : Aggregate sentiment score at time t.

—θ, ϕ : Parameters of the prediction and sentiment models respec-tively.

7.3 Prediction Model (LSTM-Based Time-Series Forecasting)

The proposed prediction component utilizes a Long Short-Term Memory (LSTM) network to capture temporal dependencies in historical stock data. Given an input sequence of T feature vectors:

$$\mathbf{X}_t = [\mathbf{x}_{t-T+1}, \mathbf{x}_{t-T+2}, \ldots, \mathbf{x}_t] \in \mathbb{R}^{m \times T}. \quad (1)$$

derived from k_t financial news articles. The terms represent the predicted probabilities of positive and negative sentiment for the i-th news item at time t. A positive S_t indicates optimistic market mood, while a negative value reflects pessimistic sentiment.

7.4 Correlation Analysis

To identify relationships between sentiment and market movement, the Pearson correlation is computed between sentiment scores $\{S_t\}$ and returns $\{r_t\}$:

Lagged correlations $\rho_{S,r}(\tau)$ are also measured to determine whether sentiment at time t affects returns at $t + \tau$:

$$\rho_{S,r}(\tau) = \text{corr}(S_t, r_{t+\tau}), \qquad \tau \geq 0. \quad (11)$$

7.5 Performance Metrics

The performance of the LSTM prediction model is evaluated using: MSE, RMSE, R^2, Accuracy

7.6 Decision Rule

The outputs of the Stock Prediction Module and the Sentiment Analysis Module are integrated through a weighted fusion strategy. The LSTM model produces a predicted return or price value $\hat{r}_{t+H}$, which represents the expected stock movement at time $t+H$

based on historical market data. In parallel, the sentiment analysis module generates an aggregate daily sentiment score $S_t \in [-1, 1]$, where positive values indicate optimistic market mood and negative values indicate pessimistic sentiment derived from financial news. These two outputs are combined using the decision function

$$C_t = \alpha \hat{r}_{t+H} + \beta S_t \qquad (12)$$

where α and β are weighting parameters that control the relative influence of numerical price trends and qualitative sentiment in-formation. In this work, the values of α and β are selected empirically through validation experiments to balance prediction ac-curacy with sentiment impact. This fusion enables the system to produce context-aware forecasts by jointly considering quantitative price behavior and qualitative market emotion.

In (12), $\hat{r}_{t+H}$ is the predicted return (or price change) at horizon H generated by the LSTM model, and S_t is the sentiment score from the text analysis module. The parameters α and β control the rel-ative influence of numerical price trends and qualitative sentiment information, respectively. This weighted fusion integrates both sig-nals into a single decision score C_t.

7.7 Summary

The proposed mathematical model combines the LSTM-based pre-diction framework with transformer-based sentiment analysis to provide a comprehensive understanding of stock market behavior [1], [2], [3]. The LSTM model captures temporal patterns from his-torical stock prices, while the transformer-based models analyze real-time financial news to classify sentiment as positive, negative, or neutral [2], [3]. By integrating these two approaches, the sys-tem correlates quantitative price trends with qualitative sentiment insights. Performance metrics and correlation analysis are used to evaluate the accuracy and reliability of the predictions, enabling the delivery of real-time, actionable financial forecasts [4].

8. SYSTEM IMPLEMENTATION

The proposed system is implemented in Python due to its strong support for data processing and machine learning. The implemen-tation includes model training, backend services, and a web-based interface.

Key libraries used include NumPy, Pandas, Scikit-learn, Tensor-Flow/Keras, NLTK, Transformers, Matplotlib, and Flask. The system is developed and tested on a standard machine with an Intel Core i5 processor, 8 GB RAM, and 256 GB SSD, running on Win-dows/Linux OS. Model training is performed using Jupyter Note-book or Google Colab, while deployment is handled the Flask framework.

Both the stock prediction and sentiment models are trained offline and integrated into the Flask server. Real-time stock data and financial news are fetched dynamically through APIs based on user input, enabling continuous updates and predictions.

9. EXPERIMENTAL SETUP

The system was evaluated using multiple stocks from different sec-tors to ensure robustness and generalization. The dataset includes historical OHLCV stock price data and financial news headlines. Data was split into 80

9.1 DATASET DESCRIPTION

The dataset consists of both numerical stock data and textual financial news. Historical OHLCV data for companies across sectors such as Banking, IT, and Pharmaceuticals were collected using the yfinance API. This data was used to train and evaluate the prediction model and capture market trends.

Table 2: Overview of the Dataset

Data Type	Source	Attributes
Stock Prices	yfinance API	Open, High, Low, Close, Volume
News Headlines	Financial News API / Web-scraping	Title, Publish Date
Technical Indicators	Calculated	Volatility

During deployment, real-time stock data and financial news are dynamically fetched based on user input to generate live predictions and sentiment analysis. Approximately 15,000–18,000 financial news headlines were collected using APIs and web scraping and classified into positive, negative, and neutral categories.

Additionally, technical indicators such as volatility were computed to improve prediction performance. The dataset split ensures unbiased evaluation of both prediction and sentiment models.

9.2 EVALUATION METRICS

Table 3: Evaluation Metrics Used in the Proposed System

Metric	Description
RMSE	Root Mean Squared Error; measures the square root of average squared differ-ences between predicted and actual val-ues. Lower is better.
MAE	Mean Absolute Error; computes the av- erage absolute difference between pre-dicted and actual values.
R^2 Score	Coefficient of Determination; indicates how well the model explains variance. Values closer to 1 are better.
Precision	Measures correctly predicted positive samples out of all predicted positives.
Recall	Measures correctly predicted positive samples out of all actual positives.
F1-Score	Harmonic mean of Precision and Recall; balances both metrics.

10. RESULTS AND DISCUSSION

The proposed AI-based stock prediction and sentiment analysis system was evaluated using historical stock data and real-time financial news across multiple sectors to ensure robustness and generalization [1]. The system was implemented using Python and deployed via Flask for real-time interaction [2].

A. Experimental Setup and Metrics
Historical stock data (OHLCV) was collected using the yfinance API and split into 80

B. Prediction Results

Table 4: Comparison of Prediction Models

Model	RMSE	MAE	R^2
Linear Regression	2.94	2.18	0.78
Random Forest	2.31	1.76	0.85
LSTM (Proposed)	1.68	1.32	0.90

The LSTM model achieved strong predictive performance with RMSE of 1.68, MAE of 1.32, and R^2 of 0.90, indicating high accuracy in capturing stock price trends. It outperformed traditional models such as Linear Regression and Random Forest, demonstrating its effectiveness in modeling temporal dependencies.

C. Sentiment Analysis Results

The sentiment analysis module achieved an accuracy of 0.92 and an F1-score of 0.89, showing reliable classification of financial news into positive, negative, and neutral categories. Results indicate that sentiment trends are strongly correlated with short-term market movements.

Table 5: Evaluation Metrics for Sentiment Analysis

Metric	Value
Accuracy	0.92
Precision	0.90
Recall	0.88
F1-Score	0.89

D. Integrated System Performance

The integration of prediction and sentiment modules improved overall system performance. The inclusion of sentiment data enhanced prediction reliability, especially during volatile market conditions. The web-based interface enabled real-time visualization of predictions, sentiment trends, and correlation insights, supporting better decision-making.

11. CONCLUSION

This study presents an AI-based framework that integrates historical stock data analysis with real-time financial news sentiment to improve market understanding [1],[3]. The system combines ma-chine learning and natural language processing to provide more re-liable predictions compared to traditional models [4], [5].

The LSTM model effectively captures temporal dependencies in stock data, while the sentiment analysis module identifies market mood from financial news, enhancing prediction accuracy [4], [5]. The web-based implementation ensures scalability and user-friendly access for real-time monitoring and decision-making [2]. Overall, the proposed framework offers a unified and data-driven approach for stock forecasting and sentiment analysis, supporting informed investment decisions. It also provides a foundation for future research in intelligent financial analytics and decision support systems [1], [4], [5].

12. FUTURE SCOPE

The proposed system demonstrates strong performance in stock prediction and real-time sentiment analysis, with scope for further enhancement. Future work can incorporate advanced deep learning models such as hybrid LSTM-CNN and transformer-based architectures to improve multi-step forecasting accuracy.

Expanding data sources to include social media platforms like Twitter and Reddit can enhance sentiment understanding and capture behavioral market trends. Additionally, integrating reinforcement learning can enable the system to adapt dynamically to chang-ing market conditions.

The system can be extended to mobile platforms for real-time alerts and interactive dashboards, improving user accessibility. Further scalability can be achieved by supporting multiple financial markets such as forex, commodities, and cryptocurrencies, along with integration into automated trading systems for portfolio management.

Overall, these enhancements can strengthen the system's adaptability, accuracy, and real-world applicability in intelligent financial forecasting.

13. REFERENCES

[1] N. Zhang, "Stock Market Prediction with Deep Learning Models," *Procedia Computer Science*, vol. 224, pp. 1139–1146, 2023.

[2] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting Stock Market Index Using Machine Learning Techniques," *Expert Systems with Applications*, vol. 42, no. 4, pp. 2162–2172, 2022.

[3] Musale, J. C., Zad, P., Nawale, S. J., Kokare, G., & Niture, S. "Stock Market Prediction Using Machine Learning". In Multifaceted Approaches for Data Acquisition, Processing and Communication, CRC Press, 2024, pp. 248–256. DOI: 10.1201/9781003470939-32

[4] S. Kumar and R. Roy, "Financial Sentiment Analysis and Its Impact on Stock Prediction," *IEEE Access*, vol. 11, pp. 54120–54135, 2023.

[5] Y. Liu, I. Lee, and H. Wang, "FinBERT: A Pretrained Language Model for Financial Sentiment Analysis," *arXiv preprint arXiv:2006.08097*, 2020.

[6] H. Nelson and J. Kim, "Stock Price Prediction Using LSTM Recurrent Neural Networks," *IEEE Access*, vol. 10, pp. 54021–54032, 2022.

[7] S. Li, X. Li, and J. Liu, "Integrating Sentiment Analysis with Stock Market Prediction Using Deep Learning," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 2, pp. 411–422, 2022.

[8] Musale, J. C., et al. "Enhancing Web Traffic Security: A Deep Learning Approach to Defend Against Traffic Analysis Attack". IEEE Conference, 2024. DOI: 10.1109/emergin63207.2024.10961562

[9] T. Wang and Y. Xu, "Multimodal Deep Learning for Stock Prediction Based on News and Price Data," *Neural Computing and Applications*, vol. 35, pp. 1783–1797, 2023.

[10] M. Gupta, N. Jain, and R. Mehta, "AI-Driven Financial Analytics: Correlation of Market Indicators with News Sentiment," in *Proc. IEEE ICCIDS*, pp. 1–6, 2023.

[11] S. Dutta and M. Das, "Sentiment Analysis of Financial News Using Transformer-Based Deep Learning Models," *International Journal of Intelligent Engineering and Systems*, vol. 15, no. 6, pp. 245–253, 2022.

[12] R. Thakkar and P. Chaudhari, "Fusion of News Sentiment and Historical Data for Stock Market Forecasting Using Machine Learning," *Journal of King Saud University – Computer and Information Sciences*, vol. 34, no. , pp. 2718–2728, 2022.