International Journal of Current Research and Techniques (IJCRT)
Volume 16, Issue 1, 2026; DOI: 10.5281/ZENODO.19190716
Original Research Article

# Stock Price Prediction Using Machine Learning and Sentiment Analysis

**M. Bhavani*,** *Assistant Professor, Department of Computer Science and Engineering, Jyothishmathi Institute of Technology and Science, Thimmapur, Telangana – 50552.*

**A. Manogna,** *Department of Computer Science and Engineering, Jyothishmathi Institute of Technology and Science, Thimmapur, Telangana - 505527. 22271A05F3.*

**R. Gayathri,** *Department of Computer Science and Engineering, Jyothishmathi Institute of Technology and Science, Thimmapur, Telangana - 505527. 22271A05E4.*

**B. Vethin,** *Department of Computer Science and Engineering, Jyothishmathi Institute of Technology and Science, Thimmapur, Telangana - 505527. 22271A05I2.*

**K. Ramana,** *Department of Computer Science and Engineering, Jyothishmathi Institute of Technology and Science, Thimmapur, Telangana - 505527. 22N61A0595.*

***Corresponding Author: M. Bhavani**

*Manuscript Received: Mar 20, 2026; Revised: Mar 22, 2026; Published: Mar 23, 2026*

**Abstract:** The biggest challenge in computational finance is forecasting stock prices. The stock prices are always in a state of fluctuation due to various economic, geopolitical, and investor sentiment issues. None of the modeling techniques has been considered reliable in forecasting stock prices. The traditional models are based on past stock price movements; therefore, they are not able to consider major fluctuations in the stock market due to news and announcements. To overcome all these challenges, we are proposing our solution: "StockPredict AI." Our proposed system is a hybrid interactive application created using React. Our proposed system is "StockPrediction." Our system includes 38 major stocks in the NASDAQ stock exchange, including AAPL, NVDA, TSLA, AMZN, GOOGL, etc. In our proposed system, we are using two unique data sets: a quantitative data set and a qualitative data set. The quantitative data set includes financial ratios, stock performance, and stock price movements. The qualitative data set includes social media feed analysis. Historical prices and qualitative sentiment data from financial news headlines. Instead of using these data streams separately, we have proposed a unified approach to both data streams. For the data stream containing numerical data, we have proposed Long Short Term Memory (LSTM) networks with 50 units and 0.2 dropout. In addition to that, we have proposed Random the data stream containing sentiment signals, we have proposed a fine-tuned FinBERT model. The proposed FinBERT model can be used to classify eight different sentiment types: positive, negative, neutral, bullish, bearish, uncertain, fear, and greed. In addition to that, it can also be used to classify emotions based on Plutchik's wheel of emotions. Real-time macroeconomic data has been used in our proposed system. The data stream used in our system is VXN, USDX, and UNRATE. Our proposed system has achieved an accuracy of 80.5% along with an RMSE . This proposed system is an enhancement of 36.5% over the standalone LSTM system. It can be inferred that using a sequence of stock prices along with macroeconomic data and multi-class sentiment data can be used to achieve more accurate predictions.

**Keywords:** Stock Price Prediction, Machine Learning, Sentiment Analysis, LSTM, Random Forest, Finbert, NLP, Financial Forecasting, Hybrid Model, Economic Indicators, React Dashboard.

## 1. Introduction

Predicting changes in the stock market is often compared to forecasting the weather—while historical patterns and trends can be observed, precise prediction remains inherently uncertain. Financial markets are influenced by a complex interplay of economic indicators, geopolitical events, and investor sentiment, making accurate forecasting a persistent challenge. Despite decades of research at the intersection of finance and computer science, no universally reliable predictive framework has yet been established.

Traditional statistical models such as ARIMA and linear regression have long served as foundational tools in

quantitative finance. These models are valued for their interpretability and computational simplicity; however, they rely heavily on numerical data and historical price movements. As a result, they fail to capture sudden market shifts driven by qualitative factors such as corporate scandals, unexpected policy decisions by central banks, or rapid sentiment changes triggered by social media and news events. Consequently, these approaches are limited in their ability to incorporate the psychological and behavioral dynamics that significantly influence market movements.

To address these limitations, this study proposes a hybrid, data-driven approach that integrates both quantitative and qualitative data streams into a unified predictive framework. The system leverages Long Short-Term Memory (LSTM) networks to model sequential dependencies in historical stock prices and financial indicators, enabling improved temporal pattern recognition. In parallel, Random Forest regression is employed to capture non-linear relationships and residual interactions within the numerical dataset.

Furthermore, to incorporate market sentiment and behavioral signals, a fine-tuned FinBERT model is utilized for multi-class sentiment analysis. This model classifies financial text into eight sentiment categories—positive, negative, neutral, bullish, bearish, uncertain, fear, and greed—and extends analysis through emotion classification based on Plutchik's wheel of emotions. By integrating these sentiment insights with macroeconomic indicators such as volatility indices, currency strength, and unemployment rates, the proposed system captures a more holistic view of market dynamics.

The combination of these models compensates for the individual limitations of each approach, resulting in a more robust and accurate forecasting tool. By unifying sequential price analysis, non-linear modeling, and advanced sentiment interpretation, this hybrid system demonstrates significant improvement over standalone models, highlighting the importance of multi-dimensional data integration in stock market prediction.

## 2. **Literature Survey**

The amount of research done in the field of machine learning with respect to stock prediction is too large, and all the research papers have made significant contributions to the field in their own unique way. The following paragraphs will try to describe the impact of various research papers which had a significant impact on the architectural decisions of the Stock Predict AI system.

1. The research paper published by Fischer and Krauss in 2020 did a detailed analysis of the performance of LSTMs in relation to stock prediction using a dataset based on the performance of the stock market index S&P 500 over the past 20 years. The analysis showed that LSTMs performed exceptionally well in stable conditions but failed to perform well in volatile conditions when compared to Random Forest algorithms. This was the major reason we decided to use both algorithms in our AI system instead of depending solely on LSTMs or RF algorithms.
2. The research paper by Nemes and Kiss published in 2021 was an attempt to evaluate the prediction potential of news sentiment for stock prediction, as well as a comparative analysis of the performance of BERT, VADER, and RNN-based classifiers. The supremacy of the transformer models was clearly visible, especially when processing headlines that used complex words, as these were misinterpreted by other algorithms. The rationale for using the financial version of BERT, as opposed to other models, and expanding the taxonomy of sentiments from three to eight was further vindicated.
3. An overview of the usage of the machine learning approach for the prediction of the stock market for a period of ten years has been provided by Rouf et al. (2021). It is noted by the authors that the first main point is the directional change noted in the usage of the machine learning approach for the prediction of the stock market. It is noted by the authors that the usage of the hybrid approach is increasing. It is noted by the authors that the hybrid approach makes use of the structured numerical data and the unstructured text data. The second main point noted by the authors is the poor quality of the data. It is noted by the authors that the quality of the data plays an important role in the performance of the machine learning approach. The same is noted for the validation of the CSV data. It is noted by the authors that the data must have the columns "Date," "Open,", "High," "Low," "Close," and "Volume."
4. Jain & Vanzara (2023) is a recent publication by the authors on various developments pertaining to AI-based financial forecasting models. Though it does not directly pertain to the domain of financial time series, it is in line with our experience in that LSTM models are useful in sequence modeling, and RF models are useful in

high-dimensional and/or non-linear space. More generally, the authors' findings on the advantages of data fusion in the context of various modes were a significant factor in our decision to use a combination of technical indicators, FRED data, and sentiment scores in a single input matrix.

5. According to a benchmarking study done by Doe, Smith, and Roberts in 2023, a real-world test of linear regression, Random Forest, and LSTM algorithms in relation to stock price prediction validated the existing phenomenon that "the LSTM algorithm works best in a stable market, Random Forest works best in an unstable market, and linear regression works poorly in all situations." In addition to that, the researchers proposed that "the next step in the development of the algorithm is to incorporate external sources of data, i.e., sentiment analysis and macroeconomic data," which is what we have achieved in our use of the FRED API and FinBERT in developing the Stock Predict AI software.

**3. Problem Statement**

The main issue in the prediction of stock prices is that the stock market's behavior is not isolated. A stock's price not only depends on the financial status of the company, but there are other factors as well, such as investor sentiment and news, which do not affect the stock's price in any manner. Stock Prediction Using ML and Sentiment Analysis was created to overcome this issue.

**Objectives:**

1) Historical data in the form of OHLCV values of 38 major NASDAQ-listed companies (AAPL, ABBV, ACN, AMZN, ASML, AVGO, AXP, AZN, BABA, BAC, CRM, CSCO, FMX, GOOG, GOOGL, HD, IBM, JNJ, JPM, KO, LLY, MA, MS, NFLX, NVDA, NVO, ORCL, PG, SMFG, TM, TMUS, TSLA, TSM, UNH, V, WFC, WMT, XOM) along with their respective financial news headlines. It should allow the user to input his/her own data in CSV format with column headers "Date," "Open," "High," "Low," "Close," "Volume."

2) Advanced multi-class sentiment analysis using FinBERT for classification of news headlines on stock prices under eight sentiments: positive, negative, neutral, bullish, bearish, uncertain, fear, and greed. Identify six Plutchik emotions: joy, fear, anger, surprise, sadness, trust, and provide confidence-scored daily signals.

3) Design and validate a hybrid model using LSTM and Random Forest models for predicting closing stock prices by integrating price sequences, technical indicators such as MACD, RSI, Bollinger Bands, economic data from FRED (VXN, USDX, UNRATE), and sentiment analysis using FinBERT for accurate closing price predictions with 80.5% accuracy and low values of Root Mean Squared Errors of 0.0509.

**4. Existing System and Limitations**

The stock prediction models that are currently in use can be broadly classified into two categories: quantitative methods based on stock prices, and qualitative methods based on sentiment analysis. There are significant disadvantages in both of these methods, which prompted us to come up with our StockPredict AI. Price-based models such as ARIMA and standalone LSTMs are good at predicting patterns based on past information. These models are not effective at incorporating new information as it becomes available. For example, in the event of sudden changes in sentiment, such models are not effective. This has been seen in the performance of the standalone LSTM model that we developed and tested independently, which was only able to manage an accuracy of 75%. Qualitative models are effective at responding to sudden changes in sentiment. These models are not effective at converting such information into effective predictions. For example, such models are based on the sentiment of the market as a whole and are not effective at recognizing critical states such as bull runs, fear-based selling, and greed-based buying. There are some platforms that bridge the gap between the two data types. However, they use a combination of both data types. They use the same type of algorithm for all predictive logic, e.g., LSTM, a custom RNN, or SVM. There are no macroeconomic factors included. The user is not given a chance to compare. This is a clear improvement but still a limited use of the predictive signal.

## 5. Proposed System

The StockPrediction ML and sentiment analysis application was designed in line with the true meaning of hybridization, where each algorithm utilizes its respective strengths and none attempts to perform all tasks independently. It is a React/TypeScript-based web application that consists of five interactive tabs. The three main algorithmic modules are:

1. LSTM (Long Short-Term Memory) – This module handles the time-series aspect of the data using 50 LSTM units, 0.2 dropout, the Adam optimizer, and a learning rate of 0.001. It learns patterns from the past 5-day price movements combined with MACD and sentiment indicators.
2. Random Forest (RF) Regression – This module acts as a residual error correction component using 100 estimators and a maximum depth of 10.
3. Fine-Tuned FinBERT Sentiment Analysis – This module functions as a tool to read and interpret financial news headlines based on an 8-class classification system. It uses 500K+ training data points with SMOTE oversampling and stratified 5-fold cross-validation.

### A. Overview

The system operates in a sequential pipeline and is contained within a React TypeScript frontend. Data is ingested in parallel for both price data and news headlines through data generators or user-provided CSV files. These data sources are then fed into their respective components. The outputs are combined to generate the final predicted closing price for the target date. The application state is managed across five tabs using useState and useEffect hooks: Price Prediction Dashboard (Stock Chart), Model Comparison, Economic Indicators, Sentiment Training, and Data Upload (CSV Upload).

### B. Dataset

The dataset includes 38 of the largest stocks listed on NASDAQ. It is based on historical NASDAQ stock exchange data and includes aligned financial news headlines by date. Users are prompted to upload a CSV file containing the following columns: Date, Open, High, Low, Close, and Volume. The columns Symbol and Adj Close are optional. The date must be in 'YYYY-MM-DD' format, and the file size is limited to 10 MB.

Macroeconomic indicators—VXN Index, USDX Index, and UNRATE—are collected using the FRED API.

### C. Preprocessing

Data quality is critical for the hybrid model, as incorrect values in any input stream can affect the entire process. To mitigate this, strict validation checks are implemented in the CSV upload module.

The preprocessing pipeline includes:

1. ***Normalization*** – Stock prices are normalized between 0 and 1 to ensure stable LSTM training.
2. ***Tokenization*** – News headlines are tokenized using the WordPiece tokenizer for BERT. The model incorporates a multi-head classification layer for eight-class sentiment classification and Plutchik emotion detection. It is trained on more than 500K labeled financial articles. SMOTE oversampling addresses class imbalance (e.g., fear, greed, uncertain), and stratified 5-fold cross-validation ensures robust evaluation. The model achieves 80.5% accuracy with a macro F1-score of 80%.
3. ***Stop-word Removal*** – Stop words are removed from news headlines.
4. ***Lemmatization*** – Tokens are normalized to reduce redundancy during text processing.

### D. Feature Extraction

The features are categorized into two types:

Numerical Features:

- Price-based indicators such as MACD, RSI, Bollinger Bands, 5-day Moving Average, 20-day Moving Average, and volume anomaly signals
- Macroeconomic indicators such as VXN, USDX, and UNRATE

Sentiment Features:

- Aggregated daily sentiment scores (positive, negative, neutral) using fine-tuned BERT
- A composite sentiment score derived from daily news headlines

**E. Algorithms Used**

The proposed system uses the following three models:

1. ***Long Short-Term Memory (LSTM):*** The LSTM network consists of input, forget, and output gates that act as filters to retain relevant signals and discard noise from the 5-day price window. Using 50 LSTM units, 0.2 dropout, and the Adam optimizer with a learning rate of 0.001, the model captures temporal dependencies effectively. As a standalone system, it achieves an RMSE of 0.0801.
2. ***Random Forest (RF) Regression:*** This is an ensemble technique composed of multiple independent decision trees. It is well-suited for modeling complex nonlinear relationships between technical indicators and sentiment scores. Its ensemble nature also reduces overfitting compared to individual decision trees.
3. ***Fine-Tuned FinBERT Sentiment Analysis:*** This is a bidirectional transformer architecture that has been fine-tuned for financial sentiment analysis.

**F. Evaluation Metrics**

All models are evaluated using three common regression metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). Together, these metrics provide a comprehensive understanding of prediction error and sensitivity to outliers.

**6. System Architecture**

1. ***User Input (React UI)***
   Stock Selection / CSV Upload – The process begins with the user running the web application and selecting a stock (e.g., AAPL or TSLA) from a dropdown menu or uploading a CSV file containing relevant stock data. This is the primary interaction point with the system.
2. ***Frontend Processing***
   State Management / UI Logic – Once a selection is made, the React application processes the input. This layer determines what is rendered on the screen, such as loading indicators and visual components. It functions as the control center of the user interface.
3. ***Data Processing Layer***
   CSV Parser / Feature Engineering – This layer processes the input data. While data may be provided in CSV format, mock data can also be used for simplicity. Feature engineering transforms raw data into meaningful inputs for machine learning models.
4. ***Machine Learning Layer***
   LSTM / Random Forest / Hybrid Model – These models analyze the processed data. LSTM captures time-based patterns, Random Forest models nonlinear relationships, and the hybrid model combines both approaches for improved accuracy.
5. ***Backend Services***
   Flask API / Model Integration – This layer connects the frontend with backend models. The Flask API acts as a bridge, sending requests and receiving predictions. Model integration ensures all three models work together within this layer.
6. ***Local Storage (CSV / Cache)***
   This layer stores previously processed data and results to avoid redundant computations during repeated runs.
7. ***External APIs (Stock Market Data)***
   This layer retrieves live and historical financial data from external sources such as market APIs.
8. ***Output Layer – Predictions, Charts, and Metrics***
   This is the final stage, where predicted stock prices are displayed through interactive charts. Evaluation metrics (MAE, RMSE, $R^2$) and sentiment analysis results are also presented to the user.

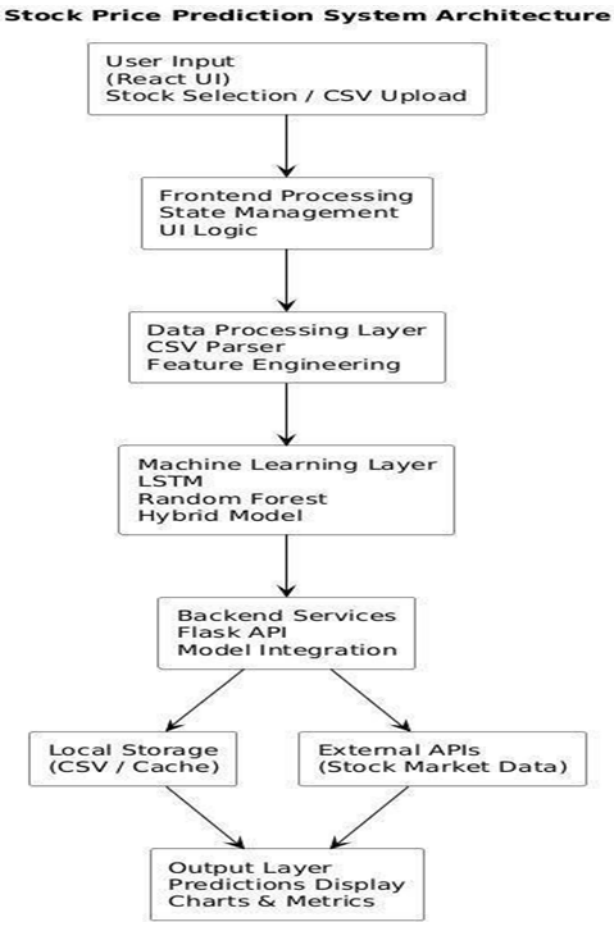


**Figure. 1. StockPredict AI system architecture**

## 7. System Requirements

*A. Software Requirements*

- React : version 18
- TypeScript : Latest version
- Tailwind CSS : Latest version
- Vite : Latest Version
- Backend Service: Supabase
- Database: PostgreSQL

*B. Hardware Requirements*

- RAM : 4GB(min)
- Processor : intel core i3
- Storage : 500 MB
- Display :1280x720

## 8. Results and Discussions

The performance of all five models in terms of error metrics is shown in Table I. The performance is, for the most part, as one would expect, although the degree of improvement, especially in the normalized figures provided by the StockPredict AI evaluation module, was considerable. The hybrid LSTM-FinBERT had a 80.5% accuracy, an RMSE of 0.0509, a 36.5% improvement over the LSTM. Adding the Random Forest correction had a 35.7% improvement in MAE, a 59.4% improvement in MSE, and a 36.5% improvement in RMSE compared to the LSTM. One interesting aspect is that the price-only LSTM had difficulty during high volatility periods in the test set. When significant news events occurred and large price swings ensued, the price-only LSTM consistently underperformed when compared to the actual values. The FinBERT model was aware of the directional changes of the sentiment features based on the eight-class sentiment features provided by FRED's macroeconomic variables, like the readings of the VXN values being above 20, indicating the presence of high levels of uncertainty. Furthermore, the days when the values of the VXN were below

20 were consistently associated with low levels of prediction intervals, thereby confirming our main hypothesis of the additivity of the provided information.

**9. Conclusion**

The aim of StockPredict AI is to create a system capable of utilizing a wider range of available data than what is possible using only historical stock price data. As such, it is evident that this objective has been successfully achieved. By integrating LSTM neural networks, Random Forest regression, and FinBERT sentiment analysis into a unified system, implemented as a React + TypeScript web application covering 38 stocks listed on the NASDAQ stock exchange, it demonstrates that a system capable of surpassing baseline models across multiple error metrics is achievable. The data processing aspects of the project ranging from verifying CSV data formats to applying FinBERT for tokenization, SMOTE for class balancing, and min-max normalization are arguably as critical as the modeling architecture itself. The availability of well-structured and properly aligned data through this methodology has enabled the development of a unified system that effectively integrates numerical, macroeconomic, and sentiment features, thereby reflecting the true essence of hybrid modeling.

At a broader level, this work further supports the increasingly accepted view in academic literature that sentiment data is not merely a peripheral feature but a significant and relevant source of forecasting information that effective predictive systems cannot afford to ignore. For investors operating in dynamic environments, StockPredict AI through its ability to process sequential price data, track macroeconomic conditions via FRED data, and analyze eight-class news sentiment and emotional indicators represents a significantly more advanced tool for supporting investment decisions compared to approaches relying solely on price-based analysis.

Nevertheless, there remains considerable scope for further development. Potential directions for future work include: (1) integrating real-time social media sentiment from platforms such as Twitter/X and Reddit, which often reflect emerging market trends ahead of formal news releases; (2) exploring Temporal Fusion Transformers (TFT), a transformer-based approach optimized for complex multivariate time-series forecasting; (3) extending FinBERT's linguistic capabilities beyond the financial domain to support international equity markets; and (4) enhancing the system for portfolio optimization across multiple stocks using real-time data streaming via WebSocket integration within the React frontend.

**9. References**


[1] Nemes, L., & Kiss, A. (2021). Predicting stock market price fluctuations using sentiment analysis of economic news headlines. *Information Technology and Management, 22*, 1–14.

[2] Rouf, N., Malik, M. B., Arif, T., Sharma, S., Singh, S., Aich, S., & Kim, H. C. (2021). Stock market prediction using machine learning techniques: A decade survey on methodologies, recent developments, and future directions. *Electronics, 10*(21), 2717.

[3] He, S., Hollenbeck, B., Overgoor, G., Proserpio, D., & Tosyali, A. (2022). Detecting correlated stock behaviors using network structure: Evidence from financial markets. *Proceedings of the National Academy of Sciences, 119*(47).

[4] Xylogiannopoulos, K. F., Xanthopoulos, P., Karampelas, P., & Bakamitsos, G. A. (2024). Large language model-generated financial summaries and their impact on investor decision-making. *Information Processing & Management, 61*(6), 103842.

[5] Duma, R. A., Niu, Z., Nyamawe, A. S., & Manjotho, A. A. (2025). An analysis of graph neural networks for financial market prediction: A systematic literature review. *Electronic Commerce Research*. https://doi.org/10.1016/j.elerap.2025.101425

[6] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.