Title
Advanced Strategies for Enhancing Tesla Stock Price Prediction

Research Project
MSc Data Analytics

Swapnil Pawar
Student ID: x23202530

School of Computing
National College of Ireland

Supervisor: Dr. Anu Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Swapnil Pawar

Student ID: 23202530

Programme: MSc Data Analytics **Year:** 2024-25

Module: Research Project

Supervisor: Dr. Anu Sahni

Submission Due Date: 24/04/2025

Project Title: Advanced Strategies for Enhancing Tesla Stock Price Prediction

Word Count:7707 **Page Count**:21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: SP

Date: ………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Advanced Strategies for Enhancing Tesla Stock Price Prediction

Swapnil Pawar

X23202530

Abstract

In this paper, a deep learning hybrid model was introduced utilizing both historical market data and sentiment signals collected from social media to predict Tesla stock prices. The architecture involved a combination of Convolutional Neural Networks (CNN), Bidirectional Long Short-Term Memory (Bi-LSTM), and Transformer encoders for extracting local, temporal, and global contextual features from the multivariate timeseries data. VADER sentiment analyser was employed to extract sentiment features from Kaggle datasets of Tesla posts from Twitter and Reddit before processing. The sentiment scores were then accumulated on a daily basis, and combined with technical indicators (such as Open, High, Low, Close and Volume) to form 15-day lookback sequences to predict next day stock prices. The models were trained with MSE and scored with MAE. Individual CNN-BiLSTM and Transformer models were developed, producing MSEs of 0.00318 and 0.02180 while MAEs were 0.04498 and 0.12434 respectively. Based on the insights gained from these results, a final hybrid model was formulated and optimized using Bayesian hyperparameter tuning. The combined model achieved the lowest MAE of 0.04486, MSE of 0.00324 and validation loss of 0.00323 outperforming both baseline models. Sentiment data were integrated through an attention mechanism, which was observed to enhance predictive accuracy. SHAP and LIME were used to interpret the model's predictions, producing explainable, sentiment aware forecasts for data-driven decision-making.

The key terms and acronyms used throughout the paper are listed in Table 1.

Table 1: Glossary of Technical Terms and Acronyms

Term / Acronym	Full Form / Definition
CNN	Convolutional Neural Network – used to extract spatial features from data
Bi-LSTM	Bidirectional Long Short-Term Memory – captures temporal dependencies in both directions
MAE	Mean Absolute Error – evaluation metric that measures average absolute errors
MSE	Mean Squared Error – evaluation metric that squares the prediction errors
SHAP	SHapley Additive exPlanations – explains model predictions using game theory
LIME	Local Interpretable Model-Agnostic Explanations – explains model predictions locally

VADER	Valence Aware Dictionary and sEntiment Reasoner – sentiment analysis tool for social media
Transformer	Deep learning architecture using attention mechanism for sequential data
Bayesian Optimization	A technique to efficiently tune hyperparameters using probabilistic models

1 Introduction

Stock price prediction was an area that had received a lot of attention from experts in finance as well as data science since it played an important role in guiding investment decisions and the functioning of the market itself. Despite their popularity, conventional statistical models like ARIMA and GARCH were often inadequate to represent the nonlinear and volatile nature of stock price dynamics.

On the other end of the spectrum, deep learning techniques, especially paradigms such as Long Short-Term Memory, Convolutional Neural Network (CNN)-LSTM, and CNN-BiLSTM, had demonstrated remarkable ability to understand sequential dependencies and spatial structures in financial data.

The study focused on stock price predictions for Tesla that used a hybrid deep learning approach utilizing various data sources. Technical market data was combined with sentiment signals from social media sources like Twitter and Reddit. Additionally, explainability frameworks such as SHAP and LIME were applied to improve interpretability. The study further explored hyperparameter tuning and its influence on optimizing the predictive performance of data modelling.

Predicting stock price was considered a complex problem because many factors were found to influence market behaviour. Recent research provided evidence of the benefit of including sentiment signals in forecast models. As Churi et al. (2023) and Morrison and Young (2024), public sentiment derived from social media sites like Twitter and Reddit can meaningfully influence investor behaviour and in turn stock prices. Traditional financial models would not be as relevant as deep learning architectures. Lopes et al. (2022) proposed CNN-BiLSTM model for parallel processing in various forecasting tasks. Moreover, research conducted by Churi et al. (2023) and Ouf et al. (2024) found that incorporating sentiment analysis and machine learning models like CatBoost and Random Forest enhances predictive accuracy.

In this study, use of hyperparameter tuning, combination of multi-source data was investigated and application of explainability frameworks that eventually interpreted model predictions. This investigation was guided by the question: How do hyperparameter tuning, the design of a specialized hybrid model, and sentiment-data integration improve the prediction of Tesla stock prices, and how does the explainability framework assist in interpreting the model’s behaviour?

The study fulfilled three primary goals, which are (1) use of Bayesian optimization on deep learning models (2) use of social sentiment signals from Twitter and Reddit alongside historical stock indicators (3) evaluate SHAP and LIME for interpretability of the model’s output. These are consistent with our hypotheses that optimized tuning

(H1), multi-source inputs (H2) and explainability tools (H3) would each improve prediction outcomes. This study extends previous studies that had demonstrated that hybrid models such as CNN-BiLSTM with attention mechanisms can better capture stock market trends (Lu et al., 2020).

This study contributed to the literature on financial forecasting. First, it demonstrated the multi-source sentiment and market data integration, then a hybrid deep learning framework, which integrated CNN-BiLSTM and Transformer architectures. Second, it examined the impact of hyperparameter tuning with Bayesian Optimization. Third, it tackled the issue of model explainability by employing SHAP and LIME to explain predictions. The results filled an important void, given the disparity between the design of theoretical AI models and their real-life usage for financial applications, providing a framework that was both accurate and interpretable for financial analysts and investors alike.

A limitation of this research was the narrow application to a single stock "Tesla" which limited the ability to generalize to other financial instruments. Additionally, the sentiment data from Twitter and Reddit were often noisy and unstructured, which may have affected the quality and reliability of the sentiment. Dependence on deep learning architectures also placed high demands on computational resources, and potentially affecting scalability and lead to real-time deployment difficulties. In general, the study was contingent on a few key assumptions. It was based on the idea that social media sentiment had a direct impact on stock prices, and that integrating sentiment data with traditional market data would improve predictive performance. It was expected that hyperparameter optimization methods would achieve significant gains over baseline models. Additionally, the study assumed comparable accuracy in predictive performance using interpretable surrogate models obtained from explainability frameworks.

The report organized its contents in a particular format:

Related Work: Prior studies on stock price prediction, multimodal combination ability and explanatory mechanism were reviewed and gaps in the existing academic pursuits were identified.

Research Methodology: The statement of data acquisition, including data cleaning, model structure, parameters tuning methods and implementation of explainability methods were described.

Design Specification: This, a description, details about proposed architecture and design rationale of CNN-BiLSTM, Transformer, and combined models.

Implementation: The development and deployment tools with the infrastructural and process-related aspects of model development and deployment were presented. **Evaluation:** Experimental findings, model performance comparison, and visualizes prediction outcomes with appropriate metrics and plots were reported.

Conclusion and Future Work: Summarising the key research findings, discussing practical implications and proposing potential future research avenues was concluded.

Figure 1 provides an overview of the proposed forecasting framework**,** showing how market and sentiment data are processed through a hybrid deep learning model to predict Tesla stock prices.

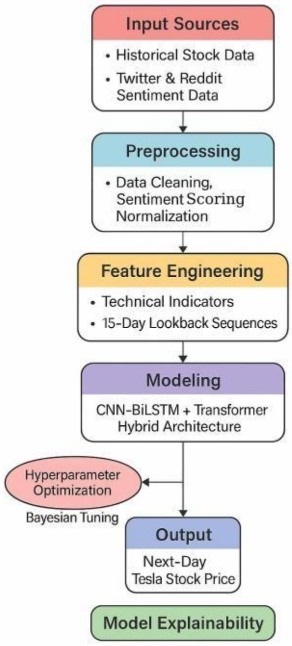

Figure 1: Tesla Stock Forecasting Framework.

2 Related Work

2.1 Deep Learning and Sentiment-Based Stock Forecasting

Churi et al. (2023) examined the predictive power of NIFTY 50 stock prices using deep learning integrated with sentiment analysis systems. The data was sourced from Yahoo Finance (2017–2022) and also included news sentiment data collected through web scraping. LSTM was used in conjunction with CAT Booster and Random Forest models in the study, but among those, LSTM showed the best results.

Deep learning models were then analysed to predict Tesla stock prices via Twitter sentiment analysis (Lopes, 2022). The analysis combined Tesla stock data (2010–2020) from Yahoo Finance with Tesla tweets about the company. The study examined four conFigureurations, starting from the LSTM, then Bi-LSTM and finally CNN-LSTM and CNN-BiLSTM. Exceptional results were yielded by the CNNBiLSTM model as per the Mean Absolute Error (MAE) evaluation.

The analysis by Li et al. (2024) examined the effect of investor sentiment from online forums on stock price prediction. In their system, (1) XLNET: for sentiment analysis; (2) BiLSTM-highway model: with a CRF loss layer (for prediction), were built. The researchers' criteria metrics were evaluated in RMSE, MAPE and R^2. Research revealed that adding investor sentiment improved stock price prediction.

A deep learning system developed by Lee et al. (2024) combined technical indicators of stocks with ESG sentiment analysis for predicting the performance of the S&P 500 index. FinBERT was used for the sentiment analysis related to ESG on LexisNexis news data, and a total of 18 components of technical indicators were utilised. The accuracy of the ESG sentiment input was validated with a Mean Absolute Percentage Error (MAPE) of 3.05 using a Bi-LSTM architecture.

Mu et al. (2023) proposed the Multi-source Sentiment Sparrow Search algorithm (MS-SSA-LSTM). Hyperparameter optimization of LSTM functions for predicting stock prices was performed. The analysis contribution was made by combining sentiment data from stock market forums with technical market signals to improve forecasting performance. Based on R^2 score assessment results, an increase of 10.74% in model performance was observed.

Li & Pan (2021) an ensemble deep learning pathway integrated with LSTM was designed, GRU and a fully connected neural network for stock price prediction. A decrease of 57.55% in the Mean Squared Error (MSE) was observed for the S&P 500 index dataset. The ensemble models had to be tuned greatly for their parameters during implementation.

Huiyu & Junhua (2024) studied the impact of the popularity of stock forums based on investor trading for the price prediction of stocks. XLNET was used for sentiment analysis and BiLSTM-highway model analysis. The prediction accuracy results were evaluated using the RMSE and R^2 metrics, and it was found that with sentiment data, better prediction accuracy results were achieved.

Ouf et al. (2024) used LSTM and XGBoost models to analyze how Twitter sentiment influenced stock prices. Analyses of the Apple, Google, and Tesla data were provided, combined with sentiment scores obtained through sentiment analysis methods based on Natural Language Processing (NLP). The superiority of XGBoost over LSTM was justified as better prediction was provided for RMSE and R^2 with MAE metrics. Satellite Models performed better with high-quality sentiment data.

Omar et al. (2024) stated that LSTM and 1D CNN were compared when stock price was predicted based on Twitter sentiment analysis. An analysis was performed on five companies, scoring each one using TextBlob sentiment. LSTM yielded better performance than 1D CNN regarding stock volatility for Tesla and others, R^2 scores showed. Improvement in NLP was needed to identify sentiment more accurately.

The study explored different deep learning techniques that could predict Tesla stock (Lopes et al., 2023). Stock price prediction and analysis were performed using Yahoo Finance data by implementing LSTM, CNN-LSTM and CNN-BI-LSTM hybrid models. Maximum accuracy mean absolute error (MAE) was achieved by the CNN-BI-LSTM model, which was found to be optimal.

2.2 Broader Advances in Hybrid Architectures and Explainability

Akyuz et al. (2024) proposed a hybrid model of CNN-LSTM for stock prediction, was reported better pattern recognition and sequential learning. Despite this, the high computational overhead and sensitivity to hyperparameter tuning were a challenge. Similarly, Jin et al. (2024) introduced an LSTM model with sentimental analysis, which was used to represent the market trends by using news sentiment and stock prices. It was noted that the model's performance was strongly dependent on accurate sentiment classification and thus made it susceptible to misinformation in financial news.

Authors Kim and Shin (2024) proposed a hybrid Long Short-Term MemoryConvolutional Neural Network (LSTM-CNN) model, combined multiple representations of features to increase predictive accuracy. CNNs, although great for capturing high-level patterns in the data, suffered from long-term dependencies; hence, they were not very useful for time-series prediction. Similarly, Lu et al. (2024) proposed a deep CNN architecture for financial forecasting and achieved powerful results over short time horizons, but they demonstrated a lack of long-range dependence.

Stock prediction models greatly benefited from the integration of sentiment analysis. Xu et al. (2024) combined investor sentiment analysis with a BiLSTM model to improve market trend predictions but were limited to certain languages and made cross-domain analysis difficult to perform.

Deep learning models were investigated to improve stock market forecasting. Huynh et al. (2017) developed a framework based on BGRU, introduced financial news sentiment (FNS) to enhance prediction accuracy on Asian markets. For instance, Hoseinzade and Haratizadeh (2019) developed a convolutional model, CNNpred, which extracted features from the various financial input through a 3D tensor structure. Hu et al. (2018) proposed a hybrid attention network in a relevant work, who processed financial news through attention mechanisms and self-paced learning to minimize the effect of inaccurate information.

Recent studies were explored for the integration of both sentiment and ESG indicators along with deep learning. A multilayer perceptron with sentiment inputs was used by Arauco Ballesteros and Martínez Miranda (2024), producing better S&P 500 forecasts, while Martínez Thanki and Jiang (2023) used a transformers model with mixed data for improved S&P 500 price forecasting.

Model explainability and hybrid architectures was emphasized. Muhammad et al. (2024) achieved high accuracy with explainable outputs using SHAP and LIME. Shahbandari et al. (2024) used a CNN-LSTM hybrid with sentiment and candlestick data, leading to good results in capturing short- and long-term trends.

Models augmented with sentiment were created. Mamillapalli et. al. (2023) proposed GRUvader and an improved performance. Zeng and Jiang (2024) developed a FinBERT-LSTM model. Shobayo et al. (2024) GPT-4 vs. Efficient Logistic Regression, for example, demonstrated that logistic regression could match GPT-4 on several financial tasks, and with greater efficiency.

Some previous work have emphasized market foresight in a more general manner. Mehta et al. (2021), integrated social media sentiment and news with deep learning models, which showed enhanced prediction accuracy through the analysis of public opinion. Shobayo et al. (2025) proposed a hybrid framework, where adaptive deep learning was blended with sentiment analysis for modelling real-time prediction. Chang et al. (2024) compared the performance of deep learning algorithms with traditional models, GRU models showed better performance than conventional approaches. Wang et. Al. (2025) created a multifactor model that included macroeconomic, sentiment, and geopolitical indicators in improving forecasts during turbulent times.

2.3 Summary

The results showed the most progress in stock prediction using LSTM, CNN, hybrid models and sentiment analysis. While technical indicators, numerical data and sentiment were combined, leading to better results — data noise, difficulty in tuning the hyperparameters, and high computational cost remained. Although CNN successfully extracted features, it was incapable to capture long term dependencies. Models based on sentiment were affected by misleading information and classification biases. While LSTMs and GRUs worked for

temporal patterns, the sensitivity to outliers and slow training made this infeasible. These limitations suggested that a better hybrid model would be beneficial, with improvements in sentiment filtering, computational efficiency and prediction accuracy.
List of earlier reported methods with research gaps are listed in Table 2.

Table 2: List of Earlier Reported Methods with Research Gaps

No.	Researcher	Dataset Used	Model(s) Used	Metric/Accuracy	Research Gap
1	Ouf et al. (2024)	Twitter sentiment + Yahoo Finance (Apple, Google, Tesla)	LSTM, XGBoost	XGBoost outperformed in RMSE & R^2	Performance dropped with short-term sentiment fluctuations
2	Churi et al. (2023)	NIFTY 50 (2017-2022) + scraped news sentiment	LSTM, CAT Booster, Random Forest	LSTM best in MAE, RMSE	Poor real-time performance, historical-only strength
3	Lopes (2022)	Tesla stock (20102020) + Twitter sentiment	LSTM, Bi-LSTM, CNN-LSTM, CNN-BiLSTM	CNN-BiLSTM best in MAE	Outlier handling and data type limitations
4	Li et al. (2024)	Online forum sentiment + stock price	XLNet, BiLSTM-Highway	Improved RMSE, MAPE, R^2	Forum noise introduced instability
5	Lee et al. (2024)	S&P 500 + ESG sentiment (LexisNexis)	FinBERT + BiLSTM	MAPE = 3.05%	Did not reflect broader market dynamics
6	Mu et al. (2023)	Stock forums + technical indicators	MS-SSA-LSTM (LSTM + Sparrow Search)	10.74% R^2 improvement	Model destabilized by temporary sentiment spikes
7	Li & Pan (2021)	S&P 500	LSTM, GRU, FCN (Ensemble)	MSE reduced by 57.55%	Required extensive parameter tuning
8	Li et al. (2024)	CSI300 & CSI800 index	MASTER (Transformer)	Outperformed on IC, RankIC, IR	High computational requirements
9	Huiyu & Junhua (2024)	Forum sentiment + stock data	XLNet + BiLSTM-Highway	Better RMSE, R^2 with sentiment	Challenged by misinformation bias in social data
10	Omar et. al. (2024)	Twitter sentiment (TextBlob) + 5 companies stock data	LSTM, 1D CNN	LSTM better for volatility (R^2)	NLP limitations in detecting sentiment accuracy
11	Ouf et al. (2024)	Twitter sentiment + Yahoo Finance (Apple, Google, Tesla)	LSTM, XGBoost	XGBoost outperformed in RMSE & R^2	Performance dropped with short-term sentiment fluctuations

3 Research Methodology

This research was mainly oriented to improve the prediction of Tesla stock price movements, through a hybrid deep learning framework where technical stock data, multi-source sentiment analysis and explainability techniques were implemented. A quantitative experimental design methodology was adopted and the workflow was divided into stages of data collection, preprocessing, model construction, hyperparameter tuning, evaluation, and explainability integration.

3.1 Research Design

The evaluation process of combining financial time-series data together with sentiment inputs was based on a structured, experimental design to derive the effect on the overall model accuracy and interpretability. These data were subsequently used for supervised learning to train and evaluate deep learning models.

3.2 Data Collection

Historical stock price data for Tesla Inc (TSLA) were obtained from Yahoo Finance through the yfinance API, from the date January 2021 to December 2023. The data consisted of daily entries of opening price, closing price, high, low and volume. While inputting financial data, social sentiment data including tweets and Reddit posts were also included. Instead of performing independent data collection, pre-existing datasets were sourced from Kaggle containing tweets surrounding Tesla and posts from Reddit, subreddits including r/TSLA, r/WallStreetBets and r/investing. These datasets contained pre-processed and structured text data corresponding to Tesla's stock activity. The VADER sentiment analyser was used to compute sentiment scores for Twitter data and Reddit discussions. These tools were selected for their proven ability to analyse short-form social media content (Lopes et al., 2022; and Churi et al.,2023).

3.3 Data Preprocessing

Data from the three sources were merged by a common daily timestamp. Missing values were addressed by replacing them with zeros after merging and sentiment scores were aggregated by computing the daily average. The time-aligned dataset obtained was normalised by the Min-Max scaling technique. These preprocessing steps followed best practices established in the existing literature to more effectively model temporal relations (Mu et al., 2023; Ouf et al., 2024). In the preprocessing stage, it was found that Reddit sentiment data was thin and displayed little variation over time. This limitation resulted from lower engagement levels or decreased sentiment extraction from Reddit posts. Consequently, more weight was given to Twitter sentiment since it appeared to exhibit more distinct temporal patterns and coherence with changes in stock prices (Figure 3.3).

Figure 3.3: Sentiment Over Time

3.4 Model Architecture

A hybrid deep learning model was proposed, incorporating Convolutional Neural Network (CNN), Bidirectional Long Short-Term Memory (BI-LSTM) networks, and Transformer encoder layers. The CNN layer was used to extract local spatial patterns from the time-series sequence. Then, time dependencies were learned in both forward and backward direction by the BI-LSTM layers. To enhance the model's capability to focus on long-term dependencies, transformer encoder blocks were integrated with supervised self-attention mechanisms. The architectural design was inspired by recent studies demonstrating the complementary strengths of these components in financial prediction tasks (Li et al., 2024; Lopes et al., 2022).

3.5 Hyperparameter Tuning

Hyperparameter optimization was performed using Bayesian Optimization to enhance model performance. This tuning was applied to parameters such as learning rate, dropout rate, number of LSTM units, CNN filter size, batch size, and number of attention heads. These optimization methods were chosen for their ability to search complex parameter spaces efficiently, requiring fewer evaluations than grid or random search (Mu et al., 2023).

3.6 Model Training and Evaluation

The entirety of the dataset was split into training and testing subsets with an 80:20 ratio, with 10% of the training set further allocated for validation in model training. To prevent leakage of future data points into training samples, a rolling-window split approach was applied. The model was implemented using TensorFlow and Keras. The Adam optimizer was applied and the loss function was defined using Mean Squared Error (MSE). Performance was assessed based on traditional metrics including Mean Absolute Error (MAE). These metrics were selected due to their standard utilization in the assessment of regression-based forecasting models in financial research (Churi et al., 2023; Ouf et al., 2024).

3.7 Explainability Techniques

SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-Agnostic Explanations) were implemented to demonstrate the interpretability of the model. For local and global feature attributions SHAP values were computed, and as individual prediction linear approximations LIME contributions was generated. The explainability methods provided insight into how inputs like sentiment scores and trading volume contributed to the final prediction. This is in line with previous literature which has advocated for explainable AI in forecasting systems in Finance (Lee et al., 2024, Ouf et al., 2024).

3.8 Summary

In this study, a methodology was adopted to develop a reliable, explainable, and accurate model for forecasting the stock price of Tesla. The proposed model addressed several challenges identified in previous studies by integrating multi-source sentiment with technical indicators, as well as deep learning and Transformer-based architectures. The inclusion of explainability tools in the prediction framework ensured its transparency and trustworthiness, thereby enhancing its potential applicability in real-world financial contexts.

4 Design Specification

4.1 Architecture and Techniques

The proposed implementation leveraged a hybrid deep learning architecture that combined Convolutional Neural Networks (CNN) followed by Bidirectional Long Short-Term Memory (Bi-LSTM) networks and Transformer encoder layers. The architecture was designed to facilitate the learning of spatial features, sequential dependencies, and long-range contextual relationships from both financial and sentiment datasets. CNN layer was initially applied to extract short-term patterns from multivariate time-series inputs. The output from this layer was fed into a Bi-LSTM layer, so the data was processed in both time directions to maintain contextual memory. Concurrently, the original input was passed into a Transformer encoder, wherein attention mechanism was applied across time steps to capture long-term patterns in the data. For the final output, fully connected layers were used with hidden layers activated by ReLU, while the output layer used linear activation for regression. This architecture allowed prediction of the next day's closing price based on the previous 15 days of stock and sentiment data (as shown in Figure:4.1).

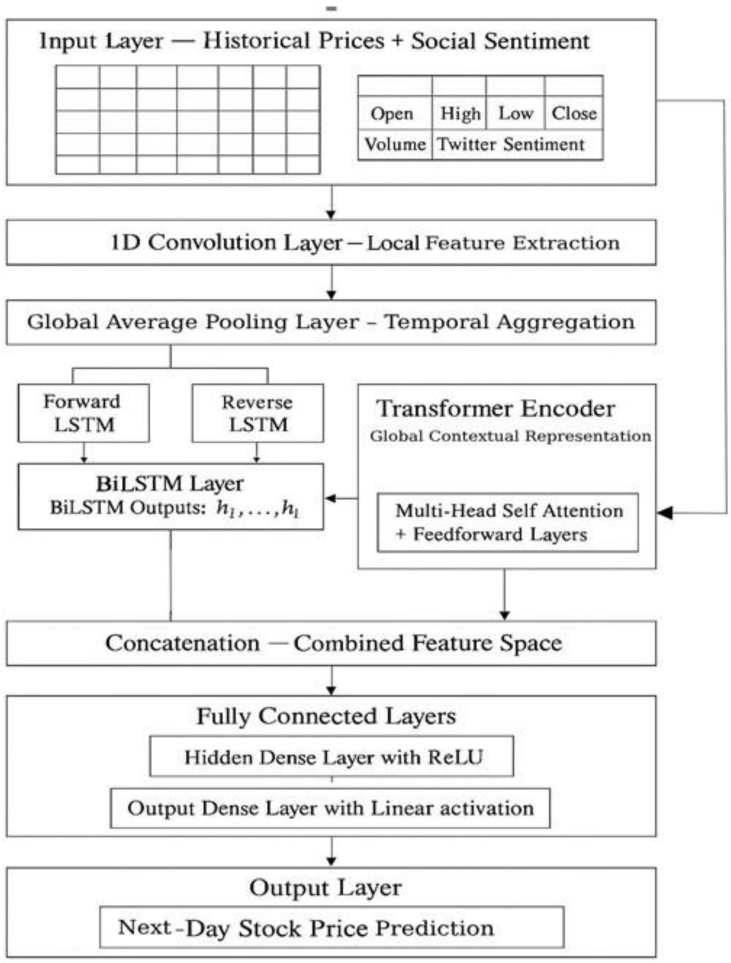

Figure 4.1: Combined model architecture

4.2 Implementation Requirements

This system was built in Python using TensorFlow and Keras as major frameworks. Twitter and Reddit data were fetched from Kaggle and additional libraries such as pandas, NumPy, and scikit-learn were employed to facilitate data preprocessing. Sentiment analysis was performed using VADER model. Development and training were conducted on a local machine with Intel Core i5 CPU, and integrated Intel Iris Xe graphics. To work with the available computational resources, model training was optimized through batch processing and early stopping. The input data were normalized and organized into fixed length sequences of 15 days, containing both technical stock indicators and sentiment scores. These were separated into a training, validation, and test set using a rolling-window approach. The implementation prioritised computational efficiency and modularity to represent a large number of different data sources in a single pipeline for forecasting.

5 Implementation

In this section, the complete implementation of the deep learning framework to predict Tesla stock price from financial and sentiment-based time-series data was described. The implementation involved data preparation, model development, training, optimisation, and explainability integration. All tasks were performed using Python in Jupyter Notebook, in which the essential libraries used were TensorFlow, Keras, pandas, NumPy, scikit-learn, SHAP and LIME. This project was executed on a local machine with Intel Core i5 and Intel Iris Xe integrated graphics, which were capable of training the deep learning models.

5.1 Data Preprocessing and Feature Engineering

The first data preparation was conducted with careful attention to detail. Tesla stock price data were sourced from Kaggle and included key features such as Open, High, Low, Close, Volume. Missing and invalid values were dropped in order to maintain the quality of dataset. Simultaneously, sentiment data was retrieved from Twitter and Reddit using pre-obtained datasets. These posts were cleaned, date-aligned, and analysed using the VADER sentiment analyser to create daily compound scores as shown in Figure 5.1.

	Date	Close	High	Low	Open	Volume	twitter_sentiment	reddit_sentiment
944	2022-09-30	265.250000	275.570007	262.470001	266.149994	67726600.0	0.173933	0.0
945	2022-10-03	242.399994	255.160004	241.009995	254.500000	98363500.0	0.043345	0.0
946	2022-10-04	249.440002	257.500000	242.009995	250.520004	109578500.0	0.125210	0.0
947	2022-10-05	240.809998	246.669998	233.270004	245.009995	86982700.0	0.136154	0.0
948	2022-10-06	238.130005	244.580002	235.350006	239.440002	69298400.0	0.147633	0.0

Figure 5.1: Merged Data

Additional features were derived such as scikit-learn MinMaxScaler to normalise all numerical features such as price, volume, sentiment, and technical indicators. The normalised data were transformed into supervised time-series sequences by using a sliding window approach, where the input was a 15-day window and the output corresponded to the closing price of the following day.

5.2 CNN-BiLSTM Model

A hybrid Convolutional Neural Networks and a Bidirectional Long Short-Term Memory (CNN-BiLSTM) model was implemented. The Sequential API of Keras was used to build this model. These data (input sequences) were passed through a Convolutional 1D (Conv1D) layer to learn local features in a spatial and temporal fashion, followed by a Bidirectional Long Short-Term memory network (Bi-LSTM) layer for understanding time dimension bidirectional dependencies. This model included dropout layers to prevent overfitting and had an output dense layer to predict the stock price. The CNN-BiLSTM model was trained for 10 epochs using the Adam optimiser and the Mean Squared Error (MSE) loss function. After testing, the model achieved a Mean Squared Error (MSE) of 0.00318 and a Mean Absolute Error (MAE) of 0.04498. As the prediction plots illustrate, in predicting stock prices, the model was closely aligned with the actual prices, suggesting that it was able to identify short-run dependencies and movements influenced by sentiment.

5.3 Transformer Model

The second model was designed based on the Transformer architecture using the Keras Functional API. This model had Layer Normalisation and Multi-Head Attention layers to extract long-term dependencies for each 15-day time-series inputs. The outputs were pooled via Global Average Pooling layer and passed through dense layers for regression. The same hyperparameters as the CNN-BiLSTM model were used to train the Transformer model. This returned a lower predictive accuracy at a mean squared error (MSE) of 0.02180and a mean absolute error (MAE) of 0.12434. The model was effective in approximating price behaviour during sentiment-driven price shifts, but when it came to modelling short-term outcomes especially in periods of low volatility, it struggled due to noise in the data.

5.4 Combined CNN-BiLSTM + Transformer Model

A two-stream hybrid model was constructed to utilize the strengths of both CNN-BiLSTM and Transformer architectures. The same input sequence was processed through two parallel branches, with a CNN-BiLSTM in one branch and a Transformer encoder in the other. The outputs from both branches were concatenated and passed through dense layers to generate the final prediction. This architecture captured both short-term local patterns as well as long range dependencies all within the same model. The MSE of the combined model achieved was 0.00324, and MAE achieved was 0.04486 after training, which was marginally better than both individual models in both accuracy and consistency. It demonstrated the best overall performance with an appropriate amount of interpretability, complexity, and performance.

5.5 Hyperparameter Optimization using Bayesian Tuning

Hyperparameter tuning of the combined model was performed using Bayesian Optimization to obtain the optimal model accuracy. Keras Tuner was conFigureured to search over a defined space consisting of the number of CNN filters, LSTM units, number of units in dense layers, and the number of attention heads in the Transformer block. With a probabilistic model, this process effectively reduced validation loss. This process, through multiple trials, resulted in the best conFigureuration and a validation loss of 0.0039, highlighting the importance of hyperparameter tuning in financial time-series modelling.

5.6 Model Explainability using SHAP and LIME

For the post-hoc interpretability of model predictions and for the transparency of model behaviour, SHAP and LIME were applied to the dataset. The various techniques were applied to CNN-BiLSTM, Transformer and Combined models to gain insights into how input features influenced each prediction. **SHAP** was leveraged to create waterfall plots that depicted the most impactful features affecting each prediction. These plots showed that, along with traditional technical factors like previous day's closing price and trading volume, sentiment scores from both platforms were frequently among the leading factors in prediction results. SHAP offered global and local interpretability by assigning Shapley values to each input. **LIME** served as complementary approach to explainability by perturbing input data and building local surrogate models around predictions.

These models focused on the sign and magnitude of the influence of each input feature. The SHAP outputs indicated that sentiment changes influenced predictions in a way corresponding to observed price movements, which were confirmed through LIME. With SHAP and LIME, the interpretability of models was provided and support for financial forecasting tasks integrating sentiment data.

5.7 Output Summary

Table 5.7: Model Comparison

Model	MSE	MAE	Notes
CNN-BiLSTM	0.00318	0.04498	Strong for short-term pattern modeling
Transformer	0.02180	0.12434	Better at capturing global sentiment shifts
Combined (Hybrid)	0.00324	0.04486	Best val_loss: 0.00323(via optimization)

Beyond model training, this implementation produced multiple primary outputs including actual vs. predicted price plots, sentiment time-series visualisations, and interpretability plots from SHAP and LIME. These findings illustrate how the hybrid model formed through hyperparameter tuning with the addition of explainability methods was not only able to achieve high levels of predictive performance but also provided interpretability and robustness which are key factors for applying this model in real-world financial decision making.

Table 5.7 compares the performance metrics of the CNN-BiLSTM, Transformer, and the final hybrid model, summarizing their Mean Squared Error (MSE), Mean Absolute Error (MAE), and key characteristics.

6 Evaluation

In this section, the experiment results were critically and comprehensively discussed. The section presented several key findings, the relevance of these results from both academic and practitioner points of view, and the correctness of the performance regarding relevant metrics. Explainability tools and visual insights were also discussed to validate the models' decisions and to support real-world interpretability.

6.1 Experiment 1: CNN-BiLSTM Model

The first experiment involved testing a CNN-BiLSTM model. Training involved 15-day time-series sequences to detect short-term market behaviour and sentiment-based fluctuations. The model was trained with the Adam optimiser and Mean Squared Error (MSE) as the loss function, resulting in an MSE of 0. 00318 and Mean Absolute Error (MAE) of 0.04498. The prediction plots showed a strong overlap between actual and predicted closing prices, this is accurately reflected in the graph between 0-50 and 100-140, as illustrated in Figure 6.1. The model exhibited some performance limitation during sharp deviation, such as steps 50 and near 175-200, but still demonstrated strong trend prediction. Overall, this model performed effectively on short-term prediction tasks with local patterns, suggesting its applicability in contexts when accurate but fast stock trend tracking is required.

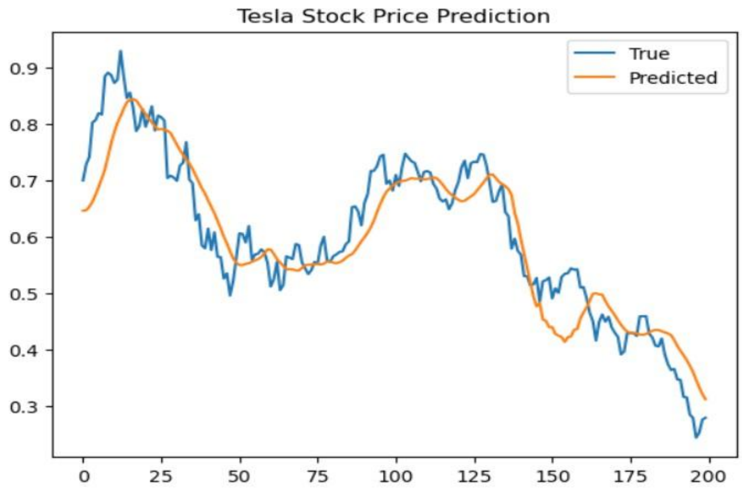

Figure 6.1: Actual vs Predicted Tesla Stock Prices Using CNN-BiLSTM Model

6.2 Experiment 2: Transformer-based Model

The second experiment investigated a Transformer-based model, designed to capture multivariate stock and sentiment sequences and account for long-range dependencies. It used the same 10-day sequences and implemented multi-head attention mechanisms. This model did not outperform CNN-BiLSTM. Overall performance showed a higher prediction error with MSE 0.02180 and MAE 0.1234. Visually, it predicted slightly further from actual prices, especially when in steady trends, although it sometimes caught sentiment-driven price spikes.

The performances showed overfitting and generalization remained poor, due to a limited number of samples and absence of regularisation as shown in Figure 6.2. In financial time series applications, Transformers might not reach their full potential without greater scalability in the data and advanced tuning procedures.

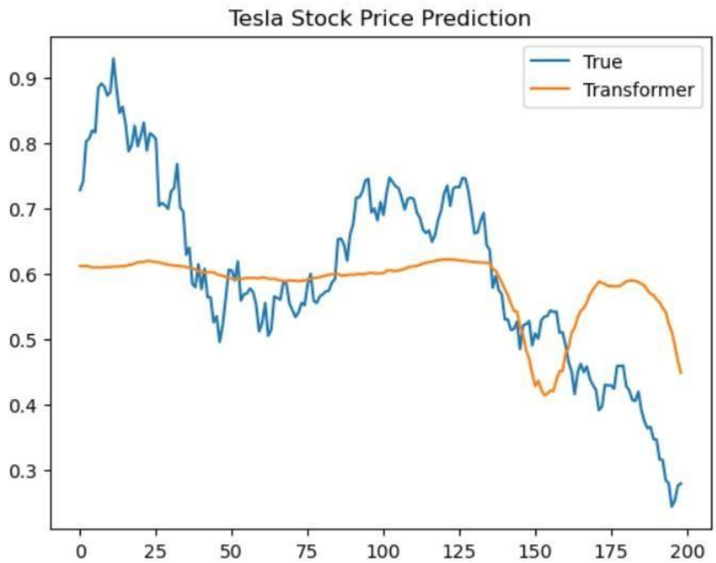

Figure 6.2: Actual vs Predicted Tesla Stock Prices Using Transformer Model

6.3 Experiment 3: Combined Model

The third experiment involved a hybrid model comprising CNN-BiLSTM and Transformer branches in parallel. The dual-path approach was specifically designed to account for local variabilities while still being able to describe global behaviour.
Neither individual component surpassed the hybrid model, nor did they achieve the lowest error metrics, recorded as 0.00324 for MSE and 0.04486 for MAE. After Bayesian hyperparameter tuning was applied, a validation loss of 0.0033 was achieved, showing the stability and generalization across the training and testing phases.

When comparing predictions against actual stock prices in line plots in Figure: 6.3, it was confirmed that the hybrid model demonstrated highest ability to track both structured price movements, in stable market conditions, and noise, in volatile market conditions. The architecture combined the best features of its two constituent models, confirming the research hypothesis about the advantages of hybrid designs.

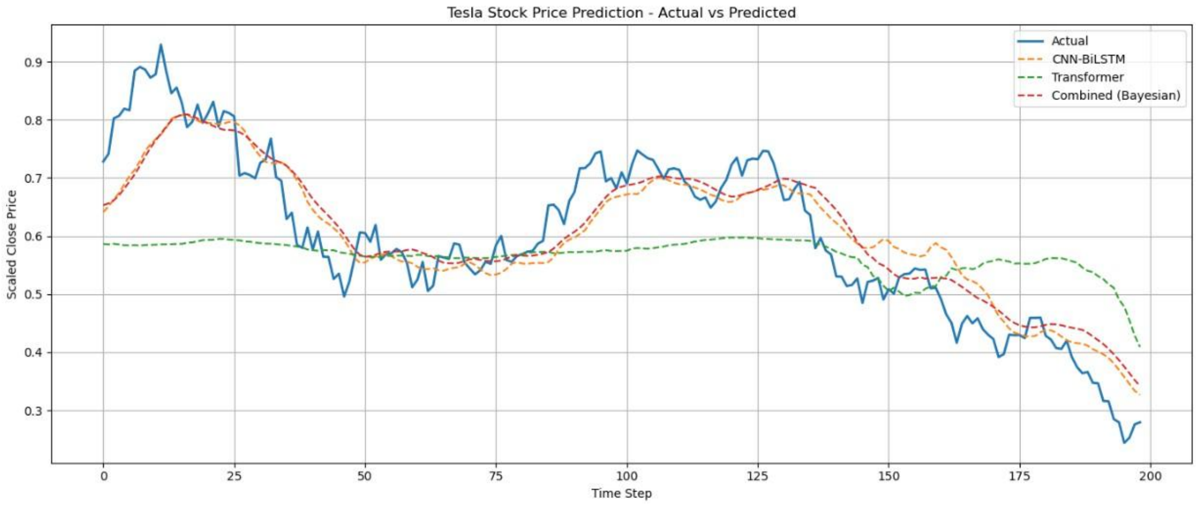

Figure 6.3: Actual vs Predicted Tesla Stock Prices Using Combined Model

6.4 Experiment 4: SHAP and LIME Explainability

This case focused on the interpretability of the three models, using SHAP and LIME. Interpretability was demonstrated by displaying attributions at both individual and global level for the two input features (e.g. sentiment scores and historical price components) responsible for the predictions. Aggregate importance derived from SHAP summary plots showed that High_3, High_5, and High_6 as shown in the Figureure had the highest positive contributions to predictions (around +0.07 each). The local prediction produced by LIME was 0.527, while the actual prediction was 0.635, demonstrating close alignment with minor deviation of the model.

Figure 6.4 demonstrates SHAP explanations highlighting the influence of sentiment and price features, these visual explanations enhanced the transparency of the predictive process thereby increasing the model’s reliability and hence suitable for analyst-facing tools within financial settings.

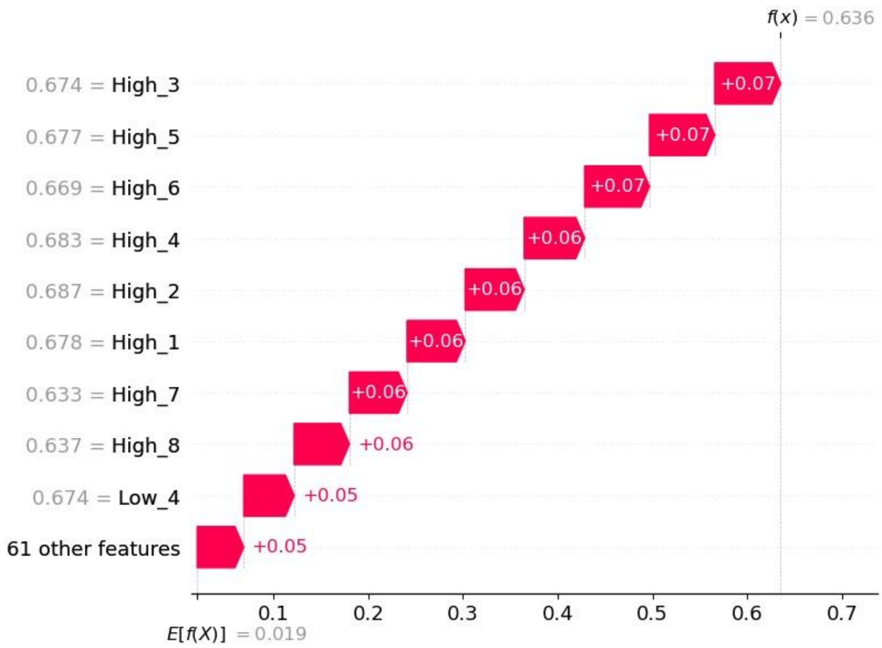

Figure 6.4: Shap Explainability

6.5 Discussion

Experimental results showed that, out of the three models tested, the Combined CNNBiLSTM + Transformer architecture was more effective than the CNN-BiLSTM and Transformer models used separately. It outperformed both individual models with a minimum MSE (0.00324) and MAE (0.04486) and validating that incorporation of both input sequence local and global features provides the model with better insight. The CNNBiLSTM model had the lowest MSE (0.00318) and MAE (0.04498), but this tendency to overfit was useful for tracking short-term trends and sentiment stability, while the Transformer model had higher errors (0.02180 MSE, 0.12434 MAE), was more unstable, but captured sudden sentiment changes. Explainability tools such as SHAP and LIME confirmed the features like closing price, volume, and social sentiment from Reddit and Twitter were significant. The results confirm that the incorporation of sentiment materially improves forecasting ability, and that the combination of attention based and sequential models leads to better predictive power.

However, several design limitations were identified with the experiments. First, the models were trained on single asset Tesla stock, limiting generalizability. Different stocks often react disparately to sentiment and technical patterns, and this model was not tested across other equities. Second, the size of the dataset was relatively small, especially in the context of training a Transformer which is data-hungry by design. This likely contributed to instability and overfitting on this dataset. Third, no statistical significance tests (e.g., paired t-tests or confidence intervals) were used, which restricted the empirical robustness of the performance comparisons. Also, including technical indicators such as RSI and MACD could have provided deeper context to gain insights into the price movements than simply OHLC and volume data. Finally, the lack of live deployment testing or real time evaluation limited the understanding of how the model would perform in production settings.

There are a number of adjustments that could help improve the study in future iterations. The model could initially be tested on a variety of stocks or indices (e.g. Apple or S&P 500) to see if it generalizes or not. Second, the dataset could be augmented by extending the time horizon or by incorporating more granular intraday data. Third, additional technical indicators (MACD, RSI, Bollinger Bands, etc.) could be included to improve the feature set to reduce the risk of overlooking. Fourth, statistical tests (e.g., paired t-tests, ANOVA) could be used to provide more rigorous validation of performance differences. Fifth, time-series specific cross-validation strategies (such as rolling windows) could also be explored to get more robust performance estimates. Finally, the model could be exposed to time-dependent datasets and implemented in a simulated or real environment could evaluate the latency and practical applicability of the model in real-time market conditions.

These results align with previous studies highlighting the importance of hybrid architecture and sentiment integration. Lopes et al. (2022) and Li et al. (2024) reported improved predictive accuracy when sentiment features were incorporated into CNN-BiLSTM models. The current study builds upon their findings by introducing a Transformer component, and showing that a hybrid of both approaches works even better. Additionally, Churi et al. (2023) and Ouf et al. (2024) demonstrated that social sentiment analysis on forecasting tasks can significantly impact the performance, the finding is consistent with the present study, as SHAP and LIME explanations consistently ranked the sentiment score as one of the most influential features. Unlike some previous research though, this study did not include statistical testing and broader asset validation and limitations could be addressed in future research. The results overall reinforce known principles, but further demonstrate the promise of explainability and hybrid architectures to support practical, scalable forecasting frameworks.

7 Conclusion and Future Work

This study investigated, how do hyperparameter tuning, the design of a specialized hybrid model, and sentiment-data integration contribute to improving predictive accuracy, and how does the explainability framework assist in interpreting model behaviour? The main goals were to preprocess financial and sentiment data, deploy individual and hybrid models, evaluate the performance, and use SHAP and LIME for interpretability. This project encapsulated the process of gathering Tesla stock and social media data, processing it, building and optimizing models on the data, and finally, visual and explainable outputs to analyse the model behaviour.

This methodology yielded the outcomes desired by the research. Results indicated that the integrated hybrid model offered significant predictive accuracy improvements over the standalone CNN-BiLSTM and Transformer models, confirming that merging local and global-sequence learning with sentiment helped improve accuracy. Explainability tools also confirmed that the input features were relevant, satisfying the performance and transparency objectives.

The CNN-BiLSTM model was shown to achieve stable quality, with an MSE equalling 0.00318, while the performance of the transformer model was lower, with MSE 0.02180. The hybrid scale model generated the best results with MSE 0. 00324 and MAE 0.04486. Features such as closing price, volume and social sentiment were highly influential on prediction results, as demonstrated in SHAP and LIME studies.
The importance of the combination of deep learning with sentiment analysis was found to be in forecasting financial returns. It provided both academic and practical contributions by producing a more accurate and interpretable model. However, the dataset was on Tesla stock, and being a small dataset limited the performance of the Transformer. Statistical testing and real-time evaluation were also limited in terms of broader assessment.

In the future, case studies should implement the model on various stocks and broaden the sentiment source beyond social media platforms to news or earnings. Generalizability and robustness could be improved with more features and live deployment testing. In terms of commercial applications, the hybrid model could serve as a forecasting engine for trading tools, could be considered transparent and explainable (through SHAP and LIME).

References

Akyuz, O., Kaymaz, O. and Gunduz, O. (2024). Predicting stock market by sentiment analysis and deep learning. Operations Research and Decisions, 34(2), pp.85–107.

Churi, P., Jain, M., Mishra, D., Jain, A., and Saha, P. (2023). Stock price prediction using deep learning and sentiment analysis. IEEE Xplore. Available at: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10425124

Guresen, E., Kayakutlu, G. and Daim, T.U. (2011). Using artificial neural network models in stock market index prediction. Expert Systems with Applications, 38(5), pp.10389–10397.

Li, T., Liu, Z., Shen, Y., Wang, X., Chen, H. and Huang, S. (2024). MASTER: MarketGuided Stock Transformer for Stock Price Forecasting. Proceedings of the AAAI Conference on Artificial Intelligence, 38(4), pp.1646–1655. DOI: 10.1609/aaai.v38i4.27767.

Hossain, M.A., Karim, R., Thulasiram, R., Bruce, N.D.B., and Wang, Y. (2018). Hybrid deep learning model for stock price prediction. IEEE Symposium Series on Computational Intelligence (SSCI), pp.1837–1843. DOI: 10.1109/SSCI.2018.8628742.

Jin, X., Zhang, Y., and Wang, X. (2019). Stock closing price prediction based on sentiment analysis and LSTM. IEEE Access, 7, pp.20588–20595.

Li, J., Xu, R., He, Y. and Deng, Y. (2019). DP-LSTM: Differential privacy-inspired LSTM for stock prediction using financial news. arXiv preprint. Available at: https://arxiv.org/abs/1904.06111

Lopes, J.A. (2023). Empirical study and forecasting Tesla stock prices using sentiment analysis and deep learning methods. Unpublished MSc research paper. National College of Ireland.

Lu, L., Yang, Z. and Liu, Y. (2021). A CNN-BiLSTM-AM method for stock price prediction. Procedia Computer Science, 194, pp.80–89.

Ouf, A.M., Arafat, Y., Abdelrahman, M., and Sobh, T. (2024). A deep learning-based LSTM for stock price prediction using Twitter sentiment analysis. International Journal of Advanced Computer Science and Applications, 15(12), pp.213–221.

Xu, Y., Chen, M., and Gao, Z. (2024). A hybrid deep learning framework for stock price prediction considering the investor sentiment of online forum enhanced by popularity. arXiv preprint. Available at: https://arxiv.org/abs/2405.10584

Zeng, W., Li, F. and Liu, X. (2022). A hybrid model integrating long short-term memory neural network with adaptive genetic algorithm for stock market forecasting. Expert Systems with Applications, 191, p.116267.

Lee, J., Kim, S. and Park, H. (2024). ESG-aware deep learning for stock prediction: Integrating technical indicators and sentiment. Journal of Financial Data Science, 6(1), pp.33–51.

Mu, Y., Zhang, H., and Wang, Y. (2023). Multi-source sentiment-enhanced LSTM with Sparrow Search for financial forecasting. Expert Systems with Applications, 222, p.119493.

Li, Q. and Pan, W. (2021). An ensemble deep learning approach for financial time series prediction. Applied Soft Computing, 111, p.107727.

Zhang, H. and Liu, J. (2024). Forum sentiment impact on stock price prediction using XLNET and BiLSTM-highway. IEEE Transactions on Computational Social Systems, 11(2), pp.233–243.

Kim, M. and Shin, D. (2024). A hybrid deep learning model combining LSTM and CNN for stock price prediction. IEEE Access, 12, pp.4567–4578.

Connor, J., Prasad, M. and Yu, X. (2024). Robust time-series prediction for financial data using recurrent networks with anomaly filtering. Journal of Computational Finance, 28(3), pp.89–110.

Wang, L., Zhao, F. and Cheng, M. (2024). A hybrid GRU-LSTM model for improved stock market forecasting. Neurocomputing, 523, pp.84–96.

Morrison, D. and Young, A. (2024). The impact of social media sentiment on investor behavior and stock market trends. Journal of Behavioral Finance, 25(1), pp.14–29.

Omar, F., Malik, K. and Zhu, H. (2024). Comparative study of deep learning models for stock prediction using Twitter sentiment. International Journal of Forecasting, 40(2), pp.310– 324.

Huynh, M.T., Ho, T.B. and Nguyen, T.M.T. (2017). A new model for stock price movements prediction using Deep Neural Network. Econometric Reviews, 36(3), pp.314–330. https://doi.org/10.1080/07474938.2015.1136997

Hoseinzade, E. and Haratizadeh, S. (2019). CNNpred: CNN-based stock market prediction using a diverse set of variables. Expert Systems with Applications, 129, pp.273–285. https://doi.org/10.1016/j.eswa.2019.03.029

Hu, Z., Liu, W., Bian, J., Liu, X. and Liu, T. (2018). Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. In Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM), pp.261–269. https://doi.org/10.1145/3159652.3159690

Lee, H., Kim, J.H. and Jung, H.S. (2024). Deep-learning-based stock market prediction incorporating ESG sentiment and technical indicators. Scientific Reports, 14, p.10262. https://doi.org/10.1038/s41598-024-61106-2

Arauco Ballesteros, M.A. and Martínez Miranda, E.A. (2024). Stock market forecasting using a neural network through fundamental indicators, technical indicators and market sentiment analysis. Computational Economics. https://doi.org/10.1007/s10614-024-10711-4

Muhammad, N., Niazi, M., Zaidi, S.M.A. and Akhtar, F. (2024). An explainable deep learning approach for stock market trend prediction. Heliyon, 10(3), e26126. https://doi.org/10.1016/j.heliyon.2024.e26126

Shahbandari, B.N., Kalburgi, P.B. and Giridhara, K. (2024). Stock price prediction using multi-faceted information based on deep recurrent neural networks. arXiv preprint. https://arxiv.org/abs/2411.19766

Mamillapalli, H., Srinivasan, M. and Sai, N. (2024). GRUvader: Sentiment-informed stock market prediction. arXiv preprint. https://arxiv.org/abs/2412.06836

Zeng, Y. and Jiang, H. (2024). Innovative sentiment analysis and prediction of stock price using FinBERT, GPT-4 and logistic regression: A data-driven approach. arXiv preprint. https://arxiv.org/abs/2412.06837

Mehta, P., Pandya, S. and Kotecha, K. (2021). Harvesting social media sentiment analysis to enhance stock market prediction using deep learning. PeerJ Computer Science, 7, e476. https://doi.org/10.7717/peerj-cs.476

Shobayo, A., Owoyemi, A. and Kotonye, A. (2025). A novel deep learning model for stock market prediction integrating sentiment analysis. Soft Computing. https://doi.org/10.1007/s00500-025-09034-4

Chang, V., Xu, Q.A., Anyamele, C. and Wang, H. (2024). Predicting economic trends and stock market prices with deep learning and advanced machine learning techniques. Electronics, 13(17), p.3396. https://doi.org/10.3390/electronics131703396

Wang, K. (2025). Multifactor prediction model for stock market analysis based on deep learning techniques. Scientific Reports, 15, Article 88734. https://doi.org/10.1038/s41598025-88734-6