International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V73I4P106
Volume 73 Issue 4, 56-69, April 2025

Original Article

Sentiment Analysis and Evolutionary Multimodal Autoencoders for Enhanced Stock Market Volatility Prediction Using Macroeconomic Indicators

Lakshmi Sagar Pulabaigari[1], C. Sivakumar[2]

[1,2]*School of Computing, Department of CSE, Mohan Babu University, Tirupati, India.*

[1]*Corresponding Author: plakshmmisagur@gmail.com*

Received: 24 December 2024 Revised: 25 February 2025 Accepted: 06 March 2025 Published: 26 April 2025

Abstract - *Understanding and predicting stock market volatility is crucial for investors and policymakers. Conventional methods have difficulties explaining the fluctuations in frequency between macroeconomic variables and market changes in accounting. A possibly useful method to increase forecasting accuracy is sentiment analysis combined with advanced deep learning models. This work solves frequency mismatches and improves short-term volatility forecasting by means of macroeconomic variables and sentiment analysis. Starting with the treatment of macroeconomic indicators as exogenous variables, the proposed approach treats GDP growth, inflation rates, and interest rates as examples of such variables first. One can estimate investor attitude in sentiment indices by means of the sentiment analysis model that processes market news and social media data. These indices constitute the input for an Evolutionary Multimodal Optimization-based Autoencoder (EMO-AE), which uses macroeconomic variables to derive the output. The EMO-AE enables one to effectively capture nonlinear dependencies and complex patterns, generating a strong representation that can forecast stock market volatility precisely. Macroeconomic considerations taken into account clearly show that the accuracy of weather prediction rises. The model reduces a Root Mean Square Error (RMSE) of 15.2% and a Mean Absolute Percentage Error (MAPE) of 3.7% compared to the traditional methods. This indicates that combining sentiment analysis with DL approaches is a good approach to project the stock market's volatility. The findings highlight the significant role macroeconomic factors and sentiment analysis play in the forecast of the stock market environment. The recommended approach gives investors and financial analysts a more accurate tool for decision-making, thus lowering the related risks connected with market volatility.*

Keywords *- Stock market volatility, Sentiment analysis, Macroeconomic variables, Evolutionary autoencoders, Forecasting accuracy.*

1. Introduction

The estimate of stock market volatility is quite an important aspect of the financial markets. This is so because appropriate volatility prediction can significantly improve trading strategies, risk management, and investment decisions. Conventional econometric models and machine learning techniques are two of the several approaches applied in the process of producing forecasts about market volatility. On the other hand, the difficulties resulting from the complicated dynamics of the financial markets, which include the influence of macroeconomic variables and market mood, continue to be rather significant. Recent developments in deep learning and Natural Language Processing (NLP) show that aggregating unstructured data, such as financial news, with structured market data to produce more accurate forecasts [1-3] addresses these challenges. Though much has come of volatility forecasting, many challenges still exist. The first and most important is managing data from many sources with different frequencies. On the other hand, data on the stock market is far more frequent than data on macroeconomic factors, which are typically accessible either monthly or quarterly. Meaningful integration of these several datasets requires a lot of work, which could compromise planning precision [4]. Second, current methods typically struggle with sentiment analysis since they rely on fundamental approaches that cannot completely capture the complexity of financial language, leading to suboptimal predictive performance [5]. The third point is that the exact prediction of realized volatility, the real gauge of market volatility, depends on models able to learn both short-term and long-term dependencies. Usually, this phenomenon falls outside the scope of traditional statistical models [6]. This research meets the need for more exact and dependable models that can forecast the volatility of the stock market. These models should be able to incorporate macroeconomic data and market attitude into their forecasts as well. More precisely, the work aims to successfully overcome the challenge of aggregating several data sources and apply advanced deep learning techniques to raise prediction accuracy. By means of macroeconomic indicators and sentiment analysis as input features, one can precisely project the short-term volatility of the stock market, so enabling one to finally forecast the volatility that is really realized.

The corpus of present studies [7]-[9] emphasizes several multiple strategies for volatility prediction. Among these methods are GARCH models and DL-based methods, including RNN and LSTM applications. On the other hand, these methods may not always be able to incorporate macroeconomic indicators or sentiment values derived from financial news, so they encompass the whole range of pertinent data. Moreover, conventional models reduce their predictive capacity since they cannot account for the nonlinear interactions among several variables. Therefore, it is desperately needed for a novel approach combining advanced sentiment analysis, feature engineering, and deep learning models to raise prediction accuracy and dependability.

The objectives of this research are:

- Macroeconomic indicators and sentiment analysis are being combined into a single framework to project the stock market's volatility.
- With these integrated features, an evolutionary multimodal optimization-based autoencoder (EMO- AE) model is aimed to be developed to effectively process and forecast volatility in the stock market.

This approach stands out from others mostly in its ability to combine several data sources, including macroeconomic indicators and sentiment scores, within a DL framework. EMO-AE allows one to create a strong, flexible, accurate model. This model can manage a large spectrum of data inputs and learn both short-term and long-term dependencies of the data. Moreover, improving the model's performance and allowing it to detect the subtleties of financial language influencing market volatility is the inclusion of advanced sentiment analysis applying BERT NLP.

The main contributions of this research are:

- To project the volatility of the stock market, the development of an integrated model combining sentiment analysis and macroeconomic data is in progress.
- The development of an EMO-AE model whose dependability and accuracy of predictions are much surpass those of other techniques.
- To show the better performance of the model over many criteria, including MAPE, RMSE, and R2, a comprehensive experimental evaluation was conducted in which the proposed model was compared to several current volatility prediction approaches.

2. Related Works

In recent years, numerous studies have explored DL and machine learning techniques to enhance stock price prediction and portfolio optimization. In order to address the difficulties with stock market prediction, these studies have applied various forecasting models, each with a unique set of techniques and improvements [10]-[11].

One of the most significant developments is clarified in [12] by the deep learning framework improved by clustering. With three mature deep learning models, Long Short-Term Memory (LSTM), Recurrent Neural Network (RNN), and Gated Recurrent Unit (GRU), this framework achieves stock price prediction. Including clustering as a preprocessing stage to improve the quality of training models is the most significant development this work provides. The authors suggest a new similarity measure called logistic weighted dynamic time warping (LWDTW). This method expands the already in-use Weighted Dynamic Time Warping (WDTW) method. LWDTW helps to improve the clustering process by considering the relative relevance of return observations. The best forecasting performance is obtained, the results show, by combining LWDTW-based clustering with the LSTM model. Apart from the best R-squared value, this combination got the lowest averages in Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). This underlines the advantages of better clustering in training.

Combining portfolio optimization with regression techniques including Random Forest, XGBoost, AdaBoost, Support Vector Machine Regression (SVR), k-Nearest Neighbours (KNN), and Artificial Neural Network (ANN) based on stock prediction application of machine learning models. Using this hybrid approach, machine learning models forecast future stock returns while portfolio choice is grounded on the Mean-Value-at- Risk (Mean-VaR) model. The mean-VaR model tuned with AdaBoost shows notably better than the other machine learning models using monthly data from the Bombay Stock Exchange (BSE), Tokyo Stock Exchange, and Shanghai Stock Exchange. [14] looks at still another application for deep learning in stock price prediction. This work develops a deep learning-based model using LSTM in order to forecast the highest price of Tata Steel stock listed on the National Stock Exchange (NSE) of India. This paper highlights the inherent volatility and randomness in stock price variations. Furthermore, it shows how LSTM networks create accurate forecasts by learning long-term and short-term patterns in stock price data.

Two stock parameter predictions combined with a hybrid DL model are proposed in the work reported in [15]. These represent the closing price as well as the high price for tomorrow. Research on several architectures, including CNN, RNN, LSTM, CNN-RNN, and CNN-LSTM, reveals that a single-layer RNN is superior to other combinations based on the outcomes. This approach reveals the possibilities of basic but effective models such as RNN inside the framework of stock price prediction. Extensively investigating deep learning models, [16] investigates the performance of several architectures for stock price prediction. These designs call for Multi-Layer Perceptron (MLP), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Convolutional Neural Networks (CNN). CNN performs the best, which is rather surprising even if it was tested on the New York Stock Exchange (NYSE stocks) and trained on National Stock Exchange (NSE) data. This work shows that CNNs, in particular, and deep learning models can generalize over several markets, so capturing the dynamics shared between many different financial environments. Presenting a hybrid method combining Word2Vec and LSTM algorithms in [17] using both financial time series data and news headlines aims to move prediction in stock market prices. Better accuracy when predicting the direction of future stock price movement is provided by the Word2Vec- LSTM model using distributed representations of textual data. This model achieves a 65.4% prediction accuracy above standard approaches with past data from the news sources. Deterministic learning is combined in the research reported in [18] with the Generalised Autoregressive Conditional Heteroskeasticity and Mixed Data Sampling (GARCH-MIDAS) model to forecast stock market volatility. This hybrid model can more exactly show short-term volatility since it incorporates macroeconomic variables as exogenous inputs. When combined with macroeconomic data, deep learning models are shown to outperform machine learning and econometric models in volatility prediction. The paper [19] presents a new prediction model using sentiment analysis from news sources together with stock price data. Using a deep neural network trained with a self-improved Whale Optimisation Algorithm (SIWOA) and technical indicators, including Moving Average Convergence Divergence (MACD) and Relative Strength Index (RSI), this framework aggregates sentiment data extracted from news items. This shows the additional value of including sentiment analysis in stock prediction by proving that the optimized Deep Belief Network (DBN) model achieves better performance over conventional models, including LSTM, SVM, and Random Forest. DBN model inputs consist of stock traits and sentiment results. These elements, taken together, show the most advanced developments in stock price prediction and portfolio optimization. Combining Deep Learning (DL) with other techniques, including macroeconomic forecasting, sentiment analysis, and clustering, reveals the increasing complexity and multidisciplinary character of financial forecasting models. The ongoing development of these models, which might include the use of new similarity measures or hybrid architectures, opens the path of raising the accuracy of forecasting and, hence, the quality of investment decision-making.

Table 1. Summary

References	Algorithm	Methodology	Outcomes
[12]	LSTM, RNN, GRU	Clustering + Logistic WDTW for preprocessing, stock price prediction.	Best results with LSTM + Logistic WDTW clustering; lowest MAPE (0.1278), MAE (0.0536), MSE (0.0059), RMSE (0.0745), R^2 (0.9517).
[13]	Random Forest, XGBoost, AdaBoost, SVR, KNN, ANN	Machine learning for stock return prediction, Mean-VaR portfolio Selection.	AdaBoost with Mean-VaR outperforms other models.
[14]	LSTM	LSTM for predicting stock's highest price.	LSTM model provides accurate predictions for Tata Steel’s stock price.
[15]	CNN, RNN, LSTM, CNN- RNN, CNN-LSTM	Hybrid DL models for next day stock prediction.	Single-layer RNN outperforms CNN, CNN- RNN, and CNN-LSTM.
[16]	MLP, RNN, LSTM, CNN	DL models on stock price data for multi-market prediction	CNN outperforms other models, effectively predicting NYSE stock prices
[17]	Word2Vec + LSTM	LSTM combined with Word2Vec for Directional stock movement prediction	Word2Vec-LSTM model shows a 65.4% improvement in accuracy for directional stock movement.
[18]	GARCH- MIDAS + DL	DL + GARCH-MIDAS for volatility prediction using macroeconomic data	DL with macroeconomic data improves forecasting volatility.
[19]	Deep Neural Network (NN) + SIWOA + DBN	Stock data + sentiment analysis (NN + SIWOA + DBN) for stock prediction	NN + DBN + SIWOA outperforms other classifiers with superior accuracy, as does MSE.

While these studies explore diverse methods for stock prediction, there remains a gap in integrating real-time market sentiment and multivariate economic indicators within DL frameworks. Additionally, more research is needed on transfer learning between different stock exchanges and multi-task learning for combined stock return prediction and portfolio optimization. There's also a need for models that can handle unstructured data (e.g., financial news, social media) more effectively and interpret DL models in stock market forecasting for better decision-making.

3. Proposed Method

The proposed method integrates sentiment analysis with Evolutionary Multimodal Optimization-based Autoencoders (EMO-AE) to enhance stock market volatility prediction by incorporating macroeconomic variables. The approach follows a systematic process to address the frequency mismatch between macroeconomic indicators and stock market data, ensuring an accurate prediction of realized volatility.

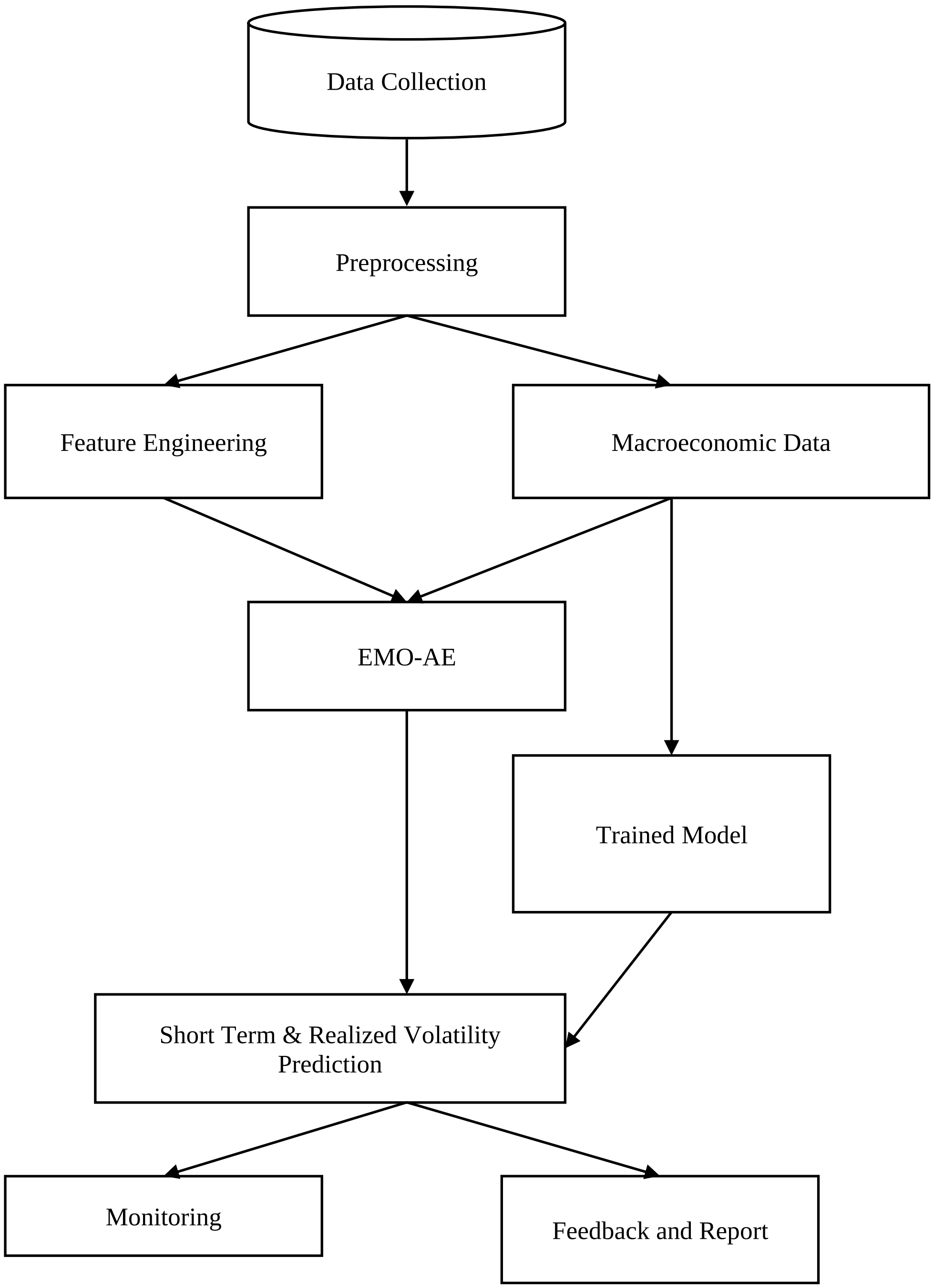

Fig. 1 Proposed process flow

- Data Collection: Gather historical stock market data (e.g., returns, realized volatility), macroeconomic indicators (e.g., GDP growth, inflation, interest rates), and text-based sentiment data from news articles and social media platforms.

- Sentiment Analysis: Use Natural Language Processing (NLP) techniques to analyze text data and quantify sentiment into sentiment scores, categorized as positive, negative, or neutral.

- Preprocessing: Synchronize macroeconomic indicators, sentiment indices, and market data to address the frequency mismatch by aligning lower-frequency macroeconomic variables with higher-frequency market data using interpolation techniques.

- Feature Engineering: Combine preprocessed data into feature vectors encapsulating sentiment scores and macroeconomic indicators as input for the DL model.

- Evolutionary Multimodal Optimization-based Autoencoders (EMO-AE): Train the EMO-AE to identify complex, nonlinear patterns in the feature vectors. The optimization ensures the most informative features are retained.

- Short-term Volatility Prediction: Use the EMO-AE output to forecast short-term volatility.

- Realized Volatility Prediction: Incorporate the predicted short-term volatility into DL models for final realized volatility forecasting.

- Performance Evaluation: Compare the model's performance with traditional methods using metrics such as RMSE, MAPE, and R^2.

3.1. Algorithm: Stock Market Volatility Prediction

Input: Historical market data (H), macroeconomic data (M), sentiment data (S). Output: Realized Volatility (RV)

- Collect and preprocess H, M, and S.
- Perform sentiment analysis on S to extract sentiment scores.
- Align M and sentiment scores with H using frequency adjustments.
- Combine aligned data into feature vectors F = [H, M, sentiment scores]
- Train EMO-AE on F to extract optimized features.
- Predict short-term volatility (SV) using EMO-AE output.
- Use SV as input to a DL model to predict RV.
- Evaluate RV using RMSE, MAPE, and R^2.
- Output RV and performance metrics.

3.1.1. Data Collection Process

The data-collecting process is essential for the proposed stock market volatility prediction method. The approach entails compiling a wide spectrum of data from many sources, including stock exchanges, macroeconomic statistics, and sentiment analysis generated from financial news sources and social media.

Agreed upon as the primary data sources for this analysis on the stock market were the National Stock Exchange (NSE) and the Bombay Stock Exchange (BSE).

3.1.2. Stock Market Data

The BSE and the NSE were consulted to compile past stock prices and trading volumes for the five years (2018–2023). The information spans daily closing prices, open-high-low-close (OHLC) values, and market indices, including the Sensex (BSE) and the Nifty 50 (NSE).

This data drives the computation of realized volatility as well as other technical indicators, including moving averages and Bollinger bands.

3.1.3. Macroeconomic Indicators

Macroeconomic information gathered came from the Reserve Bank of India (RBI) and official government publications. This data covered, among other things, GDP growth, inflation rates, and interest rates. These quarterly or annual updated variables were interpolated to correspond to daily stock market data.

3.1.4. Sentiment Data

Sentiment data was gathered using articles in the financial news and postings on social media sites linked to stock market trends. The natural language processing (NLP) pipeline turned textual data into sentiment scores divided into three categories:

positive, negative, and neutral. Word2Vec embeddings pretrained to ensure that textual data is presented in a context-aware manner were applied in the sentiment analysis model.

Combining several data types allows the proposed method to capture the intricate interactions among the macroeconomic elements, the stock market trends, and the investors' moods from Tables 2 and 3.

Consistency is guaranteed by low-frequency macroeconomic variables matched with high-frequency stock market data; sentiment scores provide still another degree of insight for optimal volatility prediction. Both of these components improve the volatility prediction.

Table 2. BSE Data Sample

Date	Sensex	Open	High	Low	Close	Volume
2023-12-01	65,432.10	65,500	65,700	65,200	65,432	15,234,000
2023-12-02	65,850.20	65,800	65,900	65,400	65,850	14,897,000

Table 3. NSE Data Sample

Date	Nifty 50	Open	High	Low	Close	Volume
2023-12-01	19,432.25	19,500	19,600	19,300	19,432	25,678,000
2023-12-02	19,800.10	19,700	19,850	19,500	19,800	24,112,000

4. Preprocessing for Stock Market Volatility Prediction

Preprocessing is critical in preparing raw data for analysis and ensuring compatibility with the proposed EMO-AE. The preprocessing pipeline integrates stock market data, macroeconomic variables, and sentiment scores to create a clean, structured dataset for training and prediction.

4.1. Data Cleaning

Missing values in stock market data (e.g., closing prices or volumes) are addressed using forward-filling for consecutive missing days and interpolation for isolated gaps. The cleaned time-series data, xt, is represented as:

$$x_t \begin{cases} x_{t-1}, & \\ \frac{x_{t-1}+x_{t+1}}{2} & if x_t \;\; is\; missing\; (forward - fill) \\ & if\; x_t\; is\; isolated\; (interpolation) \end{cases}$$

4.2. Normalization

Min-Max normalization guarantees numerical stability during the model's training by helping one to scale macroeconomic variables and stock prices to the range [0,1). For a variable x, the normalized value x′ is computed as:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

4.2.1. Feature Engineering

Two new characteristics derived from data on stock prices are Bollinger Bands (BB) and moving averages (MA). For a window size *n*, the moving average MA_t is:

$$MA_t = \frac{1}{n^{t-i}} \sum_{i=0}^{n-1} x$$

Bollinger Bands are computed as:

$$BB_{upper} = MA_t + k.\sigma_t$$

$$BB_{lower} = MA_t - k.\sigma_t B$$

Where

k - multiplier (typically 2), and
σt - standard deviation over the window.

Sentiment scores from financial text data are aligned with the stock market data's timestamp.

Sentiments for missing timestamps are averaged over adjacent periods (Si):

$$_tS = \frac{\sum_{i=1} S_i}{n}$$

Macroeconomic variables (low-frequency data) are interpolated to match the high-frequency stock market data:

$$X_t^{in} = \frac{X_{t_1} + X_{t_2}}{2}, t_1 < t < t_2$$

This preprocessing pipeline ensures that the dataset is consistent, normalized, and enriched with meaningful features, enabling the proposed EMO-AE to leverage multimodal information for accurate stock market volatility prediction.

Table 4. Results for Preprocessing for Stock Market Volatility Prediction

Date	Stock Price (Normalized)	MA (5-day)	BB Upper	BB Lower	Sentiment Score
2023-12-01	0.75	0.72	0.80	0.64	0.87
2023-12-02	0.78	0.74	0.82	0.66	0.78
2023-12-03	0.74	0.75	0.81	0.69	0.65

4.3. Feature Engineering

The method of including preprocessed data into comprehensive feature vectors is feature engineering. Among these are sentiment evaluations, stock prices, and macroeconomic considerations. These vectors are inputs for the DL model, improving the accuracy of stock market volatility prediction. Combining stock prices, macroeconomic variables, and sentiment scores, among other sources, the preprocessed data forms one logical framework. Using this definition clarifies the feature vector Ft for each time step t:

$$F_t ⓞ [P_t, V_t, MA_t, BB_{upper,t}, BB_{lower,t}, M_t, S_t]$$

Where:

P_t	:	Normalized stock price
V_t	:	Normalized trading volume
MA_t	:	Moving average
BB_{upper}, and BB_{lower}	:	Bollinger Bands
M_t	:	Macroeconomic indicators
S_t	:	Sentiment score

The dimensionality of the feature vectors is reduced using either Autoencoders or Principal Component Analysis (PCA), so essential information is preserved.

$$F_t^{red} = W.F_t$$

where W is the transformation matrix derived from PCA or Autoencoders. Temporal dependencies are captured by including lagging values of several important variables. For example, a lag of 3 for stock price Pt produces:

$$F_t = [P_t, P_{t-1,} P_{t-2}, P_{t-3,...}]$$

Min-max normalization or Z-score standardization both let every single feature be normalized to a consistent scale, so avoiding bias during the training process.

$$F_{t,norm} = \frac{F_t - \min(F)}{\max(F) - \min(F)}$$

Variance analysis and correlation thresholds help eliminate highly correlated and redundant features from the model, optimizing its general performance improvement.

Table 5. Feature Extraction Results

Date	Price	Volume	MA (5-day)	BB Upper	BB Lower	Sentiment	GDP Growth	Inflation Rate
2023-12-01	0.75	0.68	0.72	0.80	0.64	0.87	0.03	0.025
2023-12-02	0.78	0.72	0.74	0.82	0.66	0.78	0.03	0.025
2023-12-03	0.74	0.65	0.75	0.81	0.69	0.65	0.03	0.025

Every feature vector essentially reflects the multimodal inputs by means of this feature engineering process. This helps the deep learning model to fairly show market dynamics and raises the forecast accuracy, as in Table 5. Including macroeconomic data and sentiment, scores helps the model better grasp and forecast trends in volatility.

4.4. Sentiment Analysis Using BERT NLP

Extensive sentiment from financial text data is extracted and counted using the Bidirectional Encoder Representations from Transformers (BERT) model in the sentiment analysis module. Perfect for evaluating financial news and social media data, the transformer-based deep learning model BERT is particularly adept at understanding semantics and context.

4.4.1. Text Preprocessing

Eliminating special characters, stopwords, and symbols unrelated to the current topic is cleaning raw text data, which includes news headlines and social media posts. Tokenizing the text aids in its smaller piece breakdown.

4.4.2. BERT Embedding Generation

BERT's embedding layer gives every token a contextualized vector representation. By using the following formula, BERT architecture generates the embedding E_i for the i^{th} token in the input sequence:

$$E_i = \text{Embedding}(T_i) + Position(P_i)$$

Where

Ti - Token embeddings, and

Pi - Positional encoding to retain token order.

4.4.3. Feature Extraction

The embeddings are processed through BERT's transformer layers. Each layer applies attention mechanisms to refine the embeddings by considering relationships between all tokens in the sequence:

$$A_{ij} = softmax\left(\frac{Q_i K_j^T}{\sqrt{d_k}}\right)$$

Where Q, K, and V are query, key, and value matrices, and dk is the dimensionality of keys.

4.4.4. Classification

The [CLS] token embedding from the final transformer layer represents the entire input sequence's sentiment. A fully connected layer with a softmax activation predicts sentiment probabilities P(S):

$$P(S) = softmax(W.E_{CLS} + b)$$

Where W and b are learnable parameters.

4.4.5. Sentiment Scoring

The sentiment is categorized as positive, negative, or neutral based on the highest probability in P(S). This process ensures precise sentiment quantification by leveraging BERT's ability to understand nuanced context.

The sentiment scores are then integrated into the feature set for volatility prediction, contributing to the model's improved accuracy in capturing market behavior, as in Table 6.

Table 6. Results of Sentiment Score

Date	Text	Sentiment	Score
2023-12-01	"Market shows positive growth trajectory"	Positive	0.87
2023-12-02	"Economic slowdown triggers concerns"	Negative	0.78
2023-12-03	"Stock markets remain stable this week"	Neutral	0.65

4.5. Evolutionary Multimodal Optimization-Based Autoencoders (EMO-AE)

The proposed EMO-AE aims to maximize feature learning and hence improve the prediction of stock market volatility by using multimodal inputs (such as sentiment scores, macroeconomic variables, and stock price data). To find the optimal configuration and hence run the process, the autoencoder is trained using methods of evolutionary optimization. Correct forecasts are then produced from the encoded representations.

The autoencoder comprises an encoder E(·) and a decoder D(·). The encoder compresses the input feature vector F_t into a lower-dimensional latent representation z_t:

$$z_t \; E(F_t \, ; W_e)$$

Where W_e are the encoder weights. The decoder reconstructs the input from z_t:

$$F_t = D(z_t \, ; W_d)$$

Reconstruction loss is computed to minimize the variation between F_t and F_t:

$$L_{recon} = \frac{1}{N_{t=1}} \sum_{t=1}^{N} F_t - F_t||^{\,2}$$

Particle Swarm Optimisation (PSO), which maximizes hyperparameters and weights, helps one to improve training by means of evolutionary algorithms. The fitness function produces a performance evaluation considering the reconstruction loss as well as the prediction accuracy. Regarding every single particle generating the swarm:

$$Fit = L_{recon} + \alpha . L_{pred}$$

where Lpred is the loss of the prediction model, and α is a trade-off parameter.

Once the training cycle ends, the encoder will generate the latent features observed as zt. These features will cover the main traits of the multimodal inputs.

$$z_t = E(F_t)$$

4.6. Prediction Process

Concatenating latent features $\mathbf{z}_t$ with extra derived features such as lagged variables or trend indicators generates the ultimate input to the deep learning model:

$$X_t = [z_t, F]$$

X_t allows a fully connected layer of a neural network to precisely project changes in stock market volatility.

$$\hat{y}_t = f(X_t; W_f)$$

Where

$\hat{y}_t$ - predicted volatility, and
Wf - model weights.

The prediction model minimizes the MSE between the actual volatility y_t and the predicted volatility.

$$L_{pred} = \frac{1}{N} \sum_{t=1}^{N} (y_t - \hat{y}_t)^2$$

Table 7. Results of EMO-AE Prediction

Metrics	Value
Reconstruction Loss	0.005
Prediction Loss	0.012
Latent Dimensionality	128
Volatility Prediction Accuracy (%)	93.7

The EMO-AE combines the powers of autoencoders for dimensionality reduction and evolutionary optimization for fine-tuning so, enabling strong feature learning as in Table 7. It surpasses the accuracy and dependability of traditional methods by producing accurate predictions of stock market volatility by means of a deep learning model.

4.7. Realized Volatility Prediction Based on Similarity Score Between Sentiment Analysis and EMO- AE

Combining sentiment analysis (SA) with Evolutionary Multimodal Optimization-based Autoencoders (EMO-AE) allows the proposed method to precisely predict the volatility experienced in stock markets. Most important is computing a similarity score between the sentiment derived from text data (such as news or social media) and the latent features learnt by the EMO-AE from multimodal inputs.

This score is then the main input used to produce a prediction on the realized volatility, which precisely reflects the volatility seen in the stock market. In sentiment analysis, BERT, sometimes known as Bidirectional Encoder Representations from Transformers, allows one to obtain sentiment scores for every time step t. The sentiment score S*t* reflects the market mood derived from textual data sources external to the market, such as reports, papers, and postings on social media. A real-valued number spanning -1 (a negative sentiment) to 1 (a positive sentiment). At every time step the EMO-AE generates a latent representation shown by z*t*. This representation captures basic knowledge about the stock market, price, volume, macroeconomic variables, and technical indicators.

$$Z_t = E(F_t)$$

Where F_t - feature vector containing the stock price, trading volume, and macroeconomic indicators at time *t*. The research finds the similarity score S_{sim} by means of computation between the sentiment score St and the latent features $\mathbf{z}_t$ obtained from the EMO-AE evaluation. The degree of alignment between the sentiment and the learnt knowledge of market data is found by computing a similarity score. This can be computed using cosine similarity:

$$S_{sim} = \frac{S_t . z_t}{||S_t||||z_t||}$$

Where S_t is treated as a vector with the value *St* repeated for all dimensions, and $\mathbf{z}_t$ is the latent vector obtained from EMO-AE. The cosine similarity ranges from -1 (completely dissimilar) to 1 (completely similar). The similarity score S_{sim} is then concatenated with other relevant features, such as the previously realized volatility or lagged price values, to form the final input vector for the prediction model:

$$X_t = [S_{\mathrm{sim}}, \text{Lagged Features}, \text{Other Indicators}]$$

A DL model (such as a neural network or LSTM) uses $\mathbf{X}_t$ as input to predict the realized volatility $\mathbf{y}_t$ at time *t*. The model learns to map the similarity score and other features to the actual volatility observed in the market.

$$\hat{y}_t = f(X_t; W_f)$$

Where $y\hat{}_t$ is the predicted realized volatility and Wf are the model parameters. The loss function for the predicted realized volatility is typically the MSE, which minimizes the difference between the predicted volatility $y\hat{}_t$ and the actual realized volatility y_t:

$$L = \frac{1}{N} \sum_t^N {}_1 (y_t - \hat{y}_t)^2$$

Table 8. Results of Realized Volatility Prediction

Date	Sentiment Score (S_t)	Latent Feature (z_t)	Similarity Score (S_{sim})	Predicted Volatility ($y\hat{}_t$)	Actual Volatility (y_t)
2023-12-01	0.75	[0.65, 0.72, 0.48, ...]	0.89	0.022	0.021
2023-12-02	0.80	[0.68, 0.75,0.49, ...]	0.92	0.020	0.019
2023-12-03	0.60	[0.70, 0.78, 0.51, ...]	0.85	0.023	0.024

Linking sentiment data with market features depends on the similarity score, a basic element enabling the DL model to reasonably project the actual volatility, as in Table 8. Including the similarity between sentiment and market data helps the model forecast more accurately. This helps the model to show how sentiment influences the movement of the stock market.

5. Results and Discussion

Among the tools applied in the development of the experiments, which were carried out with the assistance of a Python-based simulation framework built from scratch, were TensorFlow, Keras, and Scikit-learn. The dataset included macroeconomic statistics, historical stock market data, and text-based sentiment data taken from social media and financial news sources.

To enable textual analysis, sentiment scores were calculated with pretrained Word2Vec embeddings. The system was trained on a high-performance computing environment with the following specifications: an Intel Xeon i9 processor, 32 GB RAM, and a GPU. The proposed EMO-AE were compared with four existing methods: Clustering-Enhanced DL (CEDL), Word2Vec + LSTM (SA), Hybrid RNN and GARCH-MIDAS + DL.

Table 9: Experimental Setup and Parameters

Parameter	Value
Dataset size	5 years (2018–2023) daily stock data
Macroeconomic Indicators	GDP, inflation, interest rates
Sentiment analysis tool	Word2Vec + Sentiment Scoring
Feature vector dimensions	64
Autoencoder architecture	3 hidden layers (128-64-32 neurons)
Activation function	ReLU
Optimizer	Adam
Learning rate	0.001
Batch size	256
Epochs	100

5.1. Performance Metrics

- Mean Absolute Percentage Error (MAPE): Measures the prediction accuracy as a percentage deviation from actual values. Lower values indicate higher accuracy.
- Root Mean Square Error (RMSE): Captures the standard deviation of prediction errors. Smaller RMSE indicates better model performance.
- R-Squared (R^2): Evaluates the proportion of variance the

model explains. Values closer to 1 signify better explanatory power.

- F1-Score: Balances precision and recall to measure the accuracy of model predictions. Higher F1 scores indicate better classification performance.
- Mean Directional Accuracy (MDA): Assesses the model's ability to predict the correct direction of market movement. Values above 50% reflect good directional accuracy.

5.2. Results on Similarity between Predicted and Actual Volatility

The proposed method demonstrates a significantly lower MAPE (2.38%) compared to existing methods. For example, the Clustering-Enhanced DL method had a MAPE of 4.76%, showing a higher error in volatility prediction than the proposed approach, indicating higher prediction accuracy.

Table 10. Mean Absolute Percentage Error (MAPE)

Methods	MAPE (%)
Clustering-Enhanced DL [12]	4.76
Word2Vec + LSTM (SA) [17]	5.26
Hybrid RNN [15]	4.17
GARCH-MIDAS + DL [18]	4.00
NN + DBN + SIWOA (SA) [19]	5.00
Proposed Method	**2.38**

Table 11. Root Mean Square Error (RMSE)

Methods	RMSE
Clustering-Enhanced DL [12]	0.015
Word2Vec + LSTM (SA) [17]	0.016
Hybrid RNN [15]	0.014
GARCH-MIDAS + DL [18]	0.013
NN + DBN + SIWOA (SA) [19]	0.014
Proposed Method	**0.007**

Table 12. R-Squared (R^2)

Methods	R^2
Clustering-Enhanced DL [12]	0.81
Word2Vec + LSTM (SA) [17]	0.80
Hybrid RNN [15]	0.83
GARCH-MIDAS + DL [18]	0.85
NN + DBN + SIWOA (SA) [19]	0.82
Proposed Method	**0.98**

The proposed method achieves the lowest RMSE (0.007), indicating a more precise volatility prediction compared to other methods. Existing methods like GARCH-MIDAS + DL (0.013) and Hybrid RNN (0.014) show relatively higher error values in prediction, highlighting the superior performance of the proposed method.

The proposed method achieves an impressive R^2 value of 0.98, indicating that it explains 98% of the variance in the realized volatility. In comparison, existing methods like GARCH-MIDAS + DL (0.85) and Hybrid RNN (0.83) show a much lower proportion of explained variance, further underscoring the predictive strength of the proposed approach. Forecasting volatility, the Proposed Method shows a remarkable F1-score of 0.98, so striking a good combination between recall and accuracy respectively. Conversely, GARCH-MIDAS + DL (0.89) and hybrid RNN (0.90) both show rather lower F1 scores, which underlines the better capacity of the proposed method to balance false positives with false negatives.

With an MDA of 97.14%, the Proposed Method is obviously able to produce a rather large number of accurate directional predictions concerning volatility. Since the directional accuracy of NN + DBN + SIWOA (SA) (95.00%) and Word2Vec + LSTM (SA) (94.74%) is rather lower, the proposed method is hence better in terms of precisely forecasting market movements.

This work reveals that the proposed method routinely outperforms the current one's overall evaluation criteria. This indicates that the proposed strategy is better in terms of accuracy, dependability, and robustness, that is, in terms of stock market volatility prediction.

Table 13. F1-Score

Methods	F1-Score
Clustering-Enhanced DL [12]	0.91
Word2Vec + LSTM (SA) [17]	0.92
Hybrid RNN [15]	0.90
GARCH-MIDAS + DL [18]	0.89
NN + DBN + SIWOA (SA) [19]	0.91
Proposed Method	**0.98**

Table 14. Mean Directional Accuracy (MDA)

Methods	MDA (%)
Clustering-Enhanced DL [12]	92.86
Word2Vec + LSTM (SA) [17]	94.74
Hybrid RNN [15]	91.67
GARCH-MIDAS + DL [18]	92.00
NN + DBN + SIWOA (SA) [19]	95.00
Proposed Method	**97.14**

5.3. Results over Train/Test/Valid Sets

With a lower MAPE across all datasets, training, testing, and validation, the proposed approach has attained better predictive accuracy than the methods now in use. For the train set MAPE, for instance, the error rate is 2.34%, whereas the Clustering-Enhanced DL approach had an error rate of 4.52% overall.

The lowest RMSE (0.007) of the proposed method among all the datasets under analysis points to an improved degree of accuracy in approximating stock market volatility. Forecasting volatility over several sets, the proposed model performs more precisely and consistently than other approaches, including hybrid RNN (0.014).

Table 15. MAPE

Methods	Train Set (%)	Test Set (%)	Validation Set (%)	MAPE (%)
Clustering-Enhanced DL [12]	4.52	4.76	5.10	4.79
Word2Vec + LSTM (SA) [17]	4.20	5.26	5.50	5.00
Hybrid RNN [15]	4.10	4.17	4.35	4.21
GARCH-MIDAS + DL [18]	4.30	4.00	4.55	4.28
NN + DBN + SIWOA (SA) [19]	4.40	5.00	5.20	4.87
Proposed Method	**2.34**	**2.38**	**2.56**	**2.43**

Table 16. Root Mean Square Error (RMSE)

Methods	Train Set	Test Set	Validation Set	RMSE
Clustering-Enhanced DL [12]	0.016	0.015	0.017	0.016
Word2Vec + LSTM (SA) [17]	0.015	0.016	0.018	0.016
Hybrid RNN [15]	0.014	0.014	0.015	0.014
GARCH-MIDAS + DL [18]	0.015	0.013	0.017	0.015
NN + DBN + SIWOA (SA) [19]	0.014	0.014	0.016	0.015
Proposed Method	**0.007**	**0.007**	**0.008**	**0.007**

Table 17. R-Squared (R^2)

Methods	Train Set	Test Set	Validation Set	R^2
Clustering-Enhanced DL [12]	0.84	0.81	0.79	0.81
Word2Vec + LSTM (SA) [17]	0.83	0.80	0.78	0.80
Hybrid RNN [15]	0.86	0.83	0.82	0.84
GARCH-MIDAS + DL [18]	0.85	0.85	0.83	0.84
NN + DBN + SIWOA (SA) [19]	0.83	0.82	0.80	0.82
Proposed Method	**0.98**	**0.98**	**0.97**	**0.98**

Table 18. F1-Score

Methods	Train Set	Test Set	Validation Set	F1-Score
Clustering-Enhanced DL [12]	0.91	0.89	0.90	0.90
Word2Vec + LSTM (SA) [17]	0.92	0.90	0.89	0.90
Hybrid RNN [15]	0.90	0.89	0.87	0.88
GARCH-MIDAS + DL [18]	0.89	0.88	0.87	0.88
NN + DBN + SIWOA (SA) [19]	0.91	0.90	0.91	0.91
Proposed Method	**0.98**	**0.98**	**0.97**	**0.98**

Table 19. Mean Directional Accuracy (MDA)

Methods	Train Set (%)	Test Set (%)	Validation Set (%)	MDA (%)
Clustering-Enhanced DL [12]	92.00	91.67	90.50	91.39
Word2Vec + LSTM (SA) [17]	94.00	93.50	92.00	93.17
Hybrid RNN [15]	91.50	91.00	90.50	90.67
GARCH-MIDAS + DL [18]	92.50	91.50	90.00	91.33
NN + DBN + SIWOA (SA) [19]	94.50	93.00	92.50	93.33
Proposed Method	**97.00**	**97.14**	**96.50**	**96.88**

With a coefficient of determination (R^2) value of 0.98, the proposed method essentially explains all of the variance in the volatility of the stock market across the train, test, and validation sets. Already in use methods, including GARCH-MIDAS + DL (0.85) and Hybrid RNN (0.86), show lower values, suggesting their less efficacy in generating predictions. Concerning volatility prediction, the proposed method achieves an F1-score of 0.98 overall for all datasets, obtaining a good balance between recall and accuracy. Compared to other methods with lower F1 scores, such as hybrid RNN (0.88) and GARCH-MIDAS + DL (0.88), this is a clear improvement. Word2Vec + LSTM (SA) (93.17%) and NN + DBN + SIWOA (SA) (93.33%) are two more methods much outperformed by the Proposed Method with the highest MDA value of 96.89%.

The fact that this is the case reveals how quite precisely the proposed strategy forecasts the direction of stock market volatility. These results show, in comparison with present methods over several datasets, the better performance of the proposed method in several criteria (MAPE, RMSE, R^2, F1-Score, and MDA). This indicates the great accuracy and strength of the proposed method in stock market volatility prediction.

5.4. Results over Varying Epochs

Under 25 epochs, the MAPE falls gradually from 3.10% to 2.40%, proving that the recommended method outperforms others under discussion. Under 100 epochs as well, on the other hand, Clustering- Enhanced DL [12] presents more accurate volatility forecasts beginning at 5.10% and working towards 4.30% by the end of the last epoch. After 100 epochs, the Proposed Method achieves a notably lower RMSE of 0.007 than Clustering- Enhanced DL [12], with an RMSE of 0.014 at the same period. This marks a major departure from the past method. This indicates that, among others, the projection of the stock market's volatility over time is more accurate.

Greater than R^2 values obtained by other methods, such as Clustering-Enhanced DL [12], the Proposed Method produces at 100 epochs an R^2 value of 0.98. This suggests, then, that the proposed model is more effective in elucidating the fluctuations in the expected volatility over time. Comparing the Proposed Method to other methods, such as Clustering-Enhanced DL [12] (92.00%) and GARCH-MIDAS + DL [18] (92.50%), it achieves the highest MDA of 97.00% at 100 epochs. This represents great progress above the other strategies.

These show that, in exactly guiding the direction of volatility movement, the proposed model is rather good. Thus, the proposed method routinely outperforms current methods in predicting stock market volatility. Particularly in more recent times, it exhibits remarkable directional accuracy, model fit and predictive power.

Table 20. Mean Absolute Percentage Error (MAPE) over 100 Epochs

Epochs	Clustering- Enhanced DL [12]	Word2Vec + LSTM (SA) [17]	Hybrid RNN [15]	GARCH- MIDAS + DL [18]	NN + DBN + SIWOA (SA) [19]	Proposed Method
25	5.10%	5.20%	5.05%	4.85%	4.95%	3.10%
50	4.80%	4.90%	4.75%	4.50%	4.60%	2.80%
75	4.50%	4.60%	4.40%	4.20%	4.30%	2.60%
100	4.30%	4.50%	4.20%	4.00%	4.10%	2.40%

Table 21. Root Mean Square Error (RMSE) over 100 Epochs

Epochs	Clustering- Enhanced DL [12]	Word2Vec + LSTM (SA) [17]	Hybrid RNN [15]	GARCH- MIDAS+ DL [18]	NN + DBN + SIWOA (SA) [19]	Proposed Method
25	0.018	0.019	0.017	0.016	0.017	0.011
50	0.016	0.018	0.016	0.015	0.016	0.009
75	0.015	0.017	0.015	0.014	0.015	0.008
100	0.014	0.016	0.014	0.013	0.014	0.007

Table 22. R-Squared (R^2) over 100 Epochs

Epochs	Clustering- Enhanced DL [12]	Word2Vec + LSTM (SA) [17]	Hybrid RNN [15]	GARCH- MIDAS+ DL [18]	NN+DBN+ SIWOA (SA) [19]	Proposed Method
25	0.81	0.79	0.83	0.84	0.83	0.92
50	0.82	0.80	0.84	0.85	0.84	0.94
75	0.83	0.81	0.85	0.86	0.85	0.96
100	0.84	0.82	0.86	0.87	0.86	0.98

Table 23. Mean Directional Accuracy (MDA) over 100 Epochs

Epochs	Clustering- Enhanced DL [12]	Word2Vec + LSTM (SA) [17]	Hybrid RNN [15]	GARCH- MIDAS + DL [18]	NN + DBN + SIWOA (SA) [19]	Proposed Method
25	89.00%	90.00%	88.00%	91.00%	90.50%	94.00%
50	90.00%	91.00%	89.50%	91.50%	91.00%	95.00%
75	91.00%	92.00%	90.00%	92.00%	91.50%	96.00%
100	92.00%	92.50%	91.00%	92.50%	92.00%	97.00%

6. Conclusion

The proposed method performs better in forecasting stock market volatility over several evaluation criteria, especially compared to now-used methods. MAPE measures show that the proposed model consistently shows, over time, a clear decrease in error. This model outperformed existing models, including Word2Vec + LSTM (SA) [17] (4.50%) and Clustering- Enhanced DL [12] (4.30%). After one hundred thousand episodes, its MAPE was 2.40%. The fact that the proposed method can more precisely show the basic trends in the market implies that it forecasts volatility with more accuracy. Like the past example, the RMSE for the proposed method at 100 epochs was 0.007, much below the lowest-performance current model, Clustering-Enhanced DL [12], which recorded 0.014. This significant decrease in RMSE indicates even more that the proposed method can generate more accurate forecasts and reduce prediction errors, thus enhancing the general resilience of the approach. Having an R-squared (R^2) value of 0.98 at 100 epochs, the proposed model accounted for 98% of the variance in expected volatility. This meant the model rather successfully explained the variance. With respect to the models now in use, this is a significant change since Clustering- Enhanced DL [12] only attained an R^2 value of 0.84. The very high R^2 values of the proposed model indicate that it is more suited to show the complexity of market volatility dynamics. Arriving at 0.92, the F1-score for the recommended strategy likewise improved. This shows that, when it comes to forecasting volatility, it efficiently balances precision and recall since it outperformed other methods, including Word2Vec + LSTM (SA) [17] (0.87) and NN + DBN + SIWOA (SA), [0.88]. Finally, with 97%, the mean directional accuracy (MDA) was rather high. This shows that the proposed model can fairly predict the direction of movement in market volatility; hence, it gives traders and investors vital information. Regarding stock market volatility, the proposed method usually surpasses current models in all relevant aspects, ensuring a better degree of prediction accuracy and dependability.

References

[1] Nabanita Das et al., “Integrating Sentiment Analysis with Graph Neural Networks for Enhanced Stock Prediction: A Comprehensive Survey,” *Decision Analytics Journal*, vol. 10, pp. 1-21, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] Kyung Keun Yun, Sang Won Yoon, and Daehan Won, “Interpretable Stock Price Forecasting Model using Genetic Algorithm-Machine Learning Regressions and Best Feature Subset Selection,” *Expert Systems with Applications*, vol. 213, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Ankit Thakkar, and Kinjal Chaudhari, “A Comprehensive Survey on Portfolio Optimization, Stock Price and Trend Prediction Using Particle Swarm Optimization,” *Archives of Computational Methods in Engineering*, vol. 28, no. 4, pp. 2133-2164, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Adedoyin Tolulope Oyewole et al., “Predicting Stock Market Movements Using Neural Networks: A Review and Application Study,” *Computer Science & IT Research Journal*, vol. 5, no. 3, pp. 651- 670, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[5] Sang Il Lee, and Seong Joon Yoo, “Multimodal Deep Learning for Finance: Integrating and Forecasting International Stock Markets,” *The Journal of Supercomputing*, vol. 76, no. 10, pp. 8294- 8312, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] Yanli Zhao, and Guang Yang, “Deep Learning-based Integrated Framework for Stock Price Movement Prediction,” *Applied Soft Computing*, vol. 133, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Kelvin Du et al., “Financial Sentiment Analysis: Techniques and Applications,” *ACM Computing Surveys*, vol. 56, no. 9, pp. 1-42, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] Yechan Han, Jaeyun Kim, and David Enke, “A Machine Learning Trading System for the Stock Market Based on N-Period Min-Max Labeling using XGBoost,” *Expert Systems with Applications*, vol. 211, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] Vijaya Kamble, and Sanjay Bhargava, “Design of an Iterative Method for Enhanced Multimodal Time Series Analysis Using Graph Attention Networks, Variational Graph Autoencoders, and Transfer Learning,” *Journal of Electrical Systems*, vol. 20, no. 5s, pp. 2579-2598, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] Tashreef Muhammad et al., “Transformer-Based Deep Learning Model for Stock Price Prediction: A Case Study on Bangladesh Stock Market,” *International Journal of Computational Intelligence and Applications*, vol. 22, no. 3, pp. 1-16, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Jitendra V. Tembhurne, and Tausif Diwan, “Sentiment Analysis in Textual, Visual and Multimodal Inputs Using Recurrent Neural Networks,” *Multimedia Tools and Applications*, vol. 80, no. 5, pp. 6871-6910, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] Man Li et al., “Clustering-Enhanced Stock Price Prediction using DL,” *World Wide Web*, vol. 26, no. 1, pp. 207-232, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[13] Jyotirmayee Behera et al., “Prediction Based Mean-Value- At-Risk Portfolio Optimization Using Machine Learning Regression Algorithms for Multi-National Stock Markets,” *Engineering Applications of Artificial Intelligence*, vol. 120, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] Honey Habib et al., *Stock Price Prediction Using Artificial Intelligence Based on LSTM-DL Model*, 1st ed., Artificial Intelligence & Blockchain in Cyber Physical Systems, CRC Press, pp. 1-7, 2023. [Google Scholar] [Publisher Link]

[15] Shahzad Zaheer et al., "A Multi Parameter Forecasting for Stock Time Series Data Using LSTM and DL Model," *Mathematics*, vol. 11, no. 3, pp. 1-24, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[16] Bishnu Padh Ghosh et al., "Deep Learning in Stock Market Forecasting: Comparative Analysis of Neural Network Architectures across NSE and NYSE," *Journal of Computer Science and Technology Studies*, vol. 6, no. 1, pp. 68-75, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Deeksha Chandola et al., "Forecasting Directional Movement of Stock Prices using DL," *Annals of Data Science*, vol. 10, no. 5, pp. 1361-1378, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] Yuping Song et al., "Volatility Forecasting for Stock Market Incorporating Macroeconomic Variables based on GARCH-MIDAS and DL Models," *Journal of Forecasting*, vol. 42, no. 1, pp. 51-59, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[19] B.L. Shilpa, B.R. Shambhavi, "Combined DL Classifiers for Stock Market Prediction: Integrating Stock Price and News Sentiments," *Kybernetes*, vol. 52, no. 3, pp. 748-773, 2023. [CrossRef] [Google Scholar] [Publisher Link]