Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

An Optimized Deep Learning Model for Stock Price Prediction Incorporating Investor Sentiment

Grk Prasad[1], Goluguri Guna Koushik Reddy[1], Srinivas Bachu[2], Y Dasaratha Rami Reddy[3], Raenu Kolandaisamy[4]

[1]Department of ECE, Koneru Lakshmaiah Education Foundation, Guntur, India. E-mail: ramguda1978@gmail.com

[2]Department of ECE, Siddhartha Institute of Technology & Sciences, Narapalley, Hyderabad, India. E-mail: bachusrinivas@gmail.com

[3]Department of CSE, Chaitanya Bharathi Institute of Technology, Proddatur, Andhra Pradesh, India. E-mail:dasradh@gmail.com

[4]Institute of Computer Science & Digital Innovation, UCSI University, Kuala Lumpur, Malaysia E-mail:raenu@ucsiuniversity.edu.my

ARTICLE INFO

Received: 30 Dec 2024

Revised: 05 Feb 2025

Accepted: 25 Feb 2025

ABSTRACT

The Multimodal Semi- Supervised Attention-based Long Short-Term Memory (MS-SSA-LSTM) model introduces an improvement in stock price forecasting by combining sentiment analysis, swarm intelligence algorithms, deep learning methods, and multi-source data. This model builds a dictionary of sentiments and calculates a sentiment index based on aggregation from East Money forum postings for sentiment analysis. As a result, this kind of study is insightful in providing analyses on how changes in market sentiment influence stock prices.In order to optimize the prediction accuracy, the LSTM hyperparameters are tuned using the Sparrow Search Algorithm (SSA). Empirical findings indicate that the MS-SSA-LSTM model outperforms the other models evaluated. This methodology is an effective approach for generating accurate stock price forecasts. Excellent, the program, which was specially designed for China's erratic financial market, excels in the prediction of short-term stock prices and provides useful information to help investors make an informed decision. A hybrid model of LSTM+GRU has also been introduced for the classification of stock emotion. The approach also utilized a strong ensemble, which consisted of a voting regression for stock price prediction: LinearRegression + RandomForestRegressor + KNeighborsRegressor, and a voting classifier for sentiment analysis: AdaBoost + RandomForest. All these ensembles were combined to enhance the overall predictive performance by being easily integrated with the existing models (MLP, CNN, LSTM, MS-LSTM, and MS-SSA-LSTM). A user-friendly Flask frame work with SQ Lite support was developed to streamline the sign up, sign in, and model assessment procedures and enable user engagement and testing.

Keywords: Deep learning, LSTM model, stock price prediction, sentiment analysis, sentiment dictionary, sparrow search algorithm.

INTRODUCTION

In light of the elucidation regarding the benefits associated with investment, a significant number of individuals have chosen to engage in the financial marketplace, facilitated by the maturation of the Chinese stock market alongside the swift evolution of online financial services. Notwithstanding, the stock market is characterized by its pronounced volatility and an extensive array of information. A majority of individual investors yearn for enhanced analytical competencies in data to attain profitability. Consequently, it follows that with precise forecasts of stock prices, both corporations and investors are positioned to mitigate investment risks and augment profitability.

Through the application of statistical methodologies, early scholars constructed a linear model that corresponds to the temporal patterns of stock price series. Established methodologies encompass GARCH, ARIMA, ARMA, among others. The ARMA model was conceived in an effort to facilitate the analysis of stock time series [1]. Building upon the ARMA framework, the ARIMA model serves to forecast the trajectory of stock price variations [2]. Wavelet analysis constitutes a viable component of the ARIMA model that may be utilized to enhance the fitting precision of the Shanghai Composite Index [3]. The GARCH model introduces innovative techniques for the longitudinal prediction of stock time series [4]. Concurrently, numerous researchers utilized GARCH and ARMA to formulate a novel predictive model that theoretically substantiates multivariate volumetric stock price analysis [5]. These conventional methodologies generally succeed in capturing regular and systematic data. Nonetheless, traditional forecasting techniques presume the existence of variables that are virtually nonexistent in real-world applications. As a result, the modeling of nonlinear financial data via statistical methods proves to be a formidable challenge.

Subsequently, a considerable number of researchers employ machine learning methodologies, including support vector machines (SVMs) and neural networks, to forecast stock market valuations. The fundamental principle of machine learning encompasses the acquisition of data via algorithms, the assimilation of knowledge from this data, and the subsequent forecasting of novel datasets. Due to its distinctive attributes that facilitate effective functioning

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

with limited sample sizes, high-dimensional datasets, and nonlinear scenarios, the majority of researchers implement SVM within their stock market prediction investigations. Hossain and Nasser [6] established that the SVM methodology exhibits superior accuracy in stock prediction in comparison to traditional statistical methods. Chai et al. [7] introduced a hybrid SVM framework aimed at forecasting the volatility of the HS300 index, revealing that the least squares SVM, in conjunction with the Genetic Algorithm (GA), outperforms alternative models; however, the application of SVM necessitates considerable memory resources and computational time when utilized with extensive training datasets, potentially constraining its capacity to predict voluminous stock data. Moreover, challenges pertaining to financial time series are addressed through the utilization of Artificial Neural Networks (ANNs) and multilayer ANNs. Empirical evidence suggests that ANNs possess advantages including heightened accuracy and rapid convergence [8, 9, 10]. Moghaddam and Esfandyari investigated the impact of various forward propagation ANNs on stock price forecasting [11]. Liu and Hou utilized the Bayesian regularization technique to enhance the performance of the BP neural network [12]. Nevertheless, the conventional neural network approach presents certain limitations that necessitate enhancement. The deficient generalization capability frequently culminates in local optima and overfitting. Given the substantial volume of data required for training, it is imperative to develop more robust models to mitigate these challenges.

Next, numerous academic researchers employ machine learning techniques, including support vector machines (SVMs) and neural networks, to forecast stock prices. The fundamental principle of machine learning involves the ingestion of data via algorithms, the extraction of insights from this data, and the subsequent prediction of novel data. Given its distinctive attributes, this article presents the MS-SSA-LSTM, an innovative stock price forecasting model that amalgamates data characteristics from diverse sources with LSTM neural networks and incorporates the Sparrow search algorithm. By facilitating the anticipation of stock prices, the MS-SSA-LSTM model aids investors and traders in enhancing their investment decision-making processes. Investors and traders gather data pertaining to a particular stock they intend to acquire, which encompasses historical trading records and the opinions of market participants, and subsequently input this data into the MS-SSA-LSTM model. The system autonomously generates a stock price trend chart and projects the stock price for the following day.

LITERATURE SURVEY

Utilizing the Autoregressive Moving Average (ARMA) model, analyses of volatility processes and returns within the London Stock Exchange and the S&P 500 reveal the existence and fluctuations of long-term memory characteristics [1]. Recent advancements in multifractal analysis have established it as a pivotal framework for elucidating the intricacies of financial markets, which are often inadequately represented by the linear methodologies inherent in efficient markets theory. According to the weaker interpretations of the efficient markets hypothesis, financial market returns are characterized by sequences of serially uncorrelated variables; consequently, asset prices are anticipated to follow a random walk trajectory. We engage in a comparative examination of various theoretical frameworks that endorse either unifractality or multifractality in relation to the random walk hypothesis. Numerous empirical investigations have demonstrated that stock return volatility displays consistent clustering, possesses heavy-tailed distributions, and manifests long-term dependence. The application of self-similar stochastic processes has been proposed to effectively model both long-term dependence and the presence of large tails in return volatility frameworks. In the present study, the ARMA model is employed to forecast stock returns across monthly and annual time series for both the S&P 500 and the London Stock Exchange [1]. The statistical evaluation of the S&P 500 indicates that the ARMA model for this particular index is capable of forecasting medium- to long-term horizons with known actual values and exhibits superior performance relative to the London Stock Exchange. An analysis of the London Stock Exchange demonstrates that the ARMA model outperforms the annual model regarding monthly stock returns. A comparative analysis between the London Stock Exchange and the S&P 500 suggests that both markets operate efficiently and maintain robust financial health, even amid periods of economic expansion and contraction. Drawing upon historical data spanning from November 2003 to January 2014, this analysis offers insights into the application of the Autoregressive Integrated Moving Average (ARIMA) time series model for forecasting the prospective price of gold within the Indian market, thereby mitigating the risks associated with gold acquisition. Consequently, it serves to inform investors regarding the most advantageous timing for purchasing or selling the precious metal [2]. Its momentum has recently intensified, influenced by factors such as inflationary pressures, shifts in the political climate, and overarching global trends affecting the Indian economy. To mitigate risk and enhance portfolio diversification, investors, speculators, and researchers are increasingly exploring a diverse array of financial instruments.

Theoretical investigations and empirical applications have extensively leveraged the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model along with its various extensions. For the purpose of quasi-

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

maximum likelihood estimation, robust GARCH processes typically exhibit pairwise independence within general innovations, possess a mean of zero, and maintain unit variances [4]. Higher-order dependence structures, predicated on less stringent assumptions (such as weak GARCH and the absence of unconditional correlation), can be utilized to facilitate ex ante predictions concerning GARCH innovations and, consequently, stock returns. In the present study, empirical moving windows of stock returns are implemented to assess the validity of independence tests pertaining to GARCH innovations. This research employs moving windows of empirical stock returns to scrutinize the independence of GARCH innovations. Furthermore, the independence tests produce moving values of serial dependence times, which are instrumental in providing insights regarding the anticipated directions of stock price movements for the subsequent period. Nonparametric innovation forecasts have been observed to gain from ex ante forecasting, particularly when the sign of linear profitability projections or independence diagnostics (p-values) is integrated with the sign of innovation predictors.

These categories of models—Support Vector Machines (SVMs) and Relevance Vector Machines (RVMs)—alongside GARCH-type models, particularly ARMA-GARCH-type models, have been employed extensively in financial forecasting in recent years. In the current study, ARMA-GARCH, recurrent SVMs (RSVMs), and recurrent RVMs (RRVMs) are utilized for the purpose of volatility forecasting. A comparison is conducted between pure GARCH and parametric ARMA-GARCH with the multiperiod forecasting capabilities of two GARCH frameworks based on RSVMs and RRVMs. The efficacy of the models is assessed through four performance metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), Directional Symmetry (DS), and R-squared linear regression.

The BSE SENSEX and the NIKKEI 225 represent the two Asian stock market composite indices for which empirical data were gathered for the purposes of this investigation. This paper delves into a comprehensive analysis of the effects of outliers on volatility modeling and forecasting. Despite their comparable performance, our findings indicate that RSVM and RRVM exhibit superior predictive accuracy in contrast to GARCH models. The ARMA-GARCH model demonstrates enhanced performance relative to pure GARCH, with only the RRVM in conjunction with RSVM maintaining robust predictive attributes.

In this study, an Empirical Mode Decomposition-Least Squares Support Vector Machine (EMD-LSSVM) model has been introduced to analyze the CSI 300 index, accompanied by a framework for evaluating this model in the form of a wavelet-denoising least squares support machine, or WD-LSSVM. These optimization methodologies encompass simplex, grid search (GS), particle swarm optimization (PSO), and genetic algorithm (GA), as parameter selection is deemed critical for the model's efficacy. The experimental findings further illustrate that EMD-LSSVM utilizing the GS algorithm offers superior predictive capabilities compared to alternative methodologies in ascertaining the trajectory of stock movements within the stock market.

METHODOLOGY

i) Proposed Work:

This project presents a state-of-the-art stock price prediction system called MS-SSA-LSTM. Swarm intelligence techniques, sentiment analysis, and multi-source data are all well-integrated in this approach. [14, 15, 16, 30] The Sparrow Search Algorithm is used to optimize LSTM hyperparameters, which allows the system to forecast stock prices with remarkable accuracy. Experimental results are highlighted, showing the ubiquitous application and promise of improvement in the performance of the model for the task of prediction. This model is compared to CNN, MS-LSTM, LSTM, and MLP. Moreover, a hybrid model of LSTM+GRU for classifying stock emotion was proposed. Also, a robust ensemble approach has been used with a voting regression approach for predicting stock prices through LinearRegression, RandomForestRegressor, and KNeighborsRegressor, while the voting classifier is used to determine sentiment, combining AdaBoost and RandomForest.Together, these ensembles enhanced the overall predictive performance by fitting in seamlessly with the existing models (MLP, CNN, LSTM, MS-LSTM, and MS-SSA-LSTM). A user-friendly Flask framework with SQLite support was developed to simplify the signup, signin, and model assessment procedures and allow for user interaction and testing..

ii) System Architecture:

The first thing is importing the datasets, that is, importing the Stock Tweets Dataset, Single Stock Data, and Multi-Source Data, on which basis both sentiment analysis and stock price prediction are derived. Cleaning up the text data of the Stock Tweets Dataset consists of excluding the emojis, HTML tags, punctuations, and URLs. Thus, the task of this stage is to prepare the text, ensuring it will be ready for sentiment analysis. Therefore, both Single Stock Data and Multi-Source Data undergo processing in order to eliminate the null values, duplicates, and scale the data in

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

order to prepare financial data for the forecasting of stock prices. For sentiment classification, a number of models are trained - CNN, MLP, LSTM, MS-LSTM, MS-SSA-LSTM, extensions-Voting Classifier, and LSTM + GRU. In order to determine market mood, they check the cleansed data on tweet. Other models that are trained to predict the stock prices are MLP, CNN, LSTM, MS-LSTM, MS-SSA-LSTM, and extension-Voting Regression. For predicting the stock prices, they use processed financial data. They make predictions with the models once they are trained. Predictions in sentiment analysis highlight the mood of the market. The models predict future stock prices to predict the stock prices. Sentiment analysis and stock price model predictions are very important in helping traders and investors make informed decisions. The combined findings minimize risks, maximize investment returns, and assist consumers in navigating the intricate stock market environment.

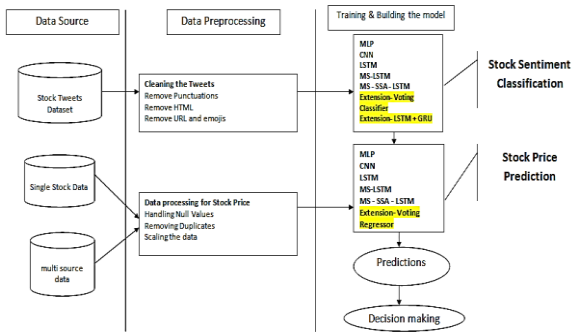

Fig 1 Proposed architecture

iii) Dataset collection:

STOCK TWEETS DATASET

The "Stock Tweets" dataset has tweets pertaining to stocks and financial markets on social media. We used it to understand the feelings and reactions of people on news about the market [1,4,7,8]. This helped us develop the tools for investments and trading stocks. We attempted to understand the way social media impacts movements in the markets and stock prices so that we can help the traders and investors.So, these are the top 5 rows of the dataset

	Text	Sentiment
0	Kickers on my watchlist XIDE TIT SOQ PNK CPW B...	1
1	user: AAP MOVIE. 55% return for the FEA/GEED i...	1
2	user I'd be afraid to short AMZN - they are lo...	1
3	MNTA Over 12.00	1
4	OI Over 21.37	1

Fig 2 Stock tweets dataset

ALL STOCK DATASET

This "All Stock Dataset" is the name given to an extensive compilation of financial data from multiple sources. It offers a multitude of data for in-depth analysis of the stock market. We used this dataset in our project to improve our stock price prediction model. Our objective was to enhance the accuracy of stock price forecasting by using data from various sources, which would eventually benefit businesses and investors.

THIS IS THE SAMPLE DATASET

Date	Open	High	Low	Close	Volume
2012-01-03	325.25	332.83	324.97	663.59	7,380,500
2012-01-04	331.27	333.87	329.08	666.45	5,749,400
2012-01-05	329.83	330.75	326.89	657.21	6,590,300
2012-01-06	328.34	328.77	323.68	648.24	5,405,900
2012-01-09	322.04	322.29	309.46	620.76	11,688,800

Fig 3 All stock datasets

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

iv) Data Processing:

Data processing refers to the process of transforming raw data into useful business information. Data scientists usually work with data collection, organization, cleaning, verification, analysis, and converting it to output formats, such as documents or graphs. Data can be processed through three methods: manual, mechanical, and electronic. This eventually enhances the value of the information and makes decision-making easier. This has enabled businesses to improve their operations and make strategic decisions timely. This is largely because of automated data processing technologies, especially computer software programming. It may help in turning vast volumes of data, such as big data, into knowledge for decision-making and quality control.

v) Feature selection:

Feature selection is defined as finding the most reliable, relevant, and non-redundant features to use in model building. Given the increasing size and diversity of datasets, it is critical to systematically reduce their size. Improving the performance of a predictive model and reducing the computational cost of the modeling process are the most important goals of feature selection.

Feature selection-the act of identifying the most salient features that go into training a machine-learning algorithm-is yet another important technique used in the domain of feature engineering. Methods for feature selection involve removing either noisy or irrelevant features and scaling the feature universe down to focus only on input variables most critical to the training machine-learning algorithm, thereby decreasing the number of input variables.The main benefits of performing feature selection in advance, rather than letting the machine learning model figure out which features are most important.

vi) Algorithms:

A Multilayer Perceptron (MLP) operates by processing data through a series of layers. It begins with an input layer that receives data and proceeds to hidden layers, where each neuron calculates a weighted sum of inputs, applies an activation function for non-linearity, and passes the result to the next layer. These weights between neurons are adjusted during training to optimize the network's ability to learn complex patterns in data. The terminal output layer produces predictions or categorizations. Multi-Layer Perceptrons (MLPs) are employed in an extensive array of applications, ranging from visual recognition to financial prediction, owing to their capacity to represent intricate interrelationships within data.

```
from sklearn.neural_network import MLPClassifier
mlp = MLPClassifier(random_state=1, max_iter=300)
mlp.fit(X_train, y_train)
y_pred = mlp.predict(X_test)
```

Fig 4 MLP

A Convolutional Neural Network (CNN) is a type of deep learning model suitable for various data beyond images. It processes data through layers that apply convolutions and pooling operations, enabling the network to automatically learn relevant patterns or features within the data. This makes CNNs valuable for tasks involving sequential data or grids, such as time series analysis or structured data processing. They excel at capturing intricate relationships and hierarchies, contributing to their versatility in different domains, including natural language processing and financial predictions.

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

```
from tensorflow.keras import Sequential,utils
from tensorflow.keras.layers import Flatten, Dense, Conv1D, MaxPool1D, Dropout

def reg():

    model = Sequential()

    model.add(Conv1D(32, kernel_size=(3,), padding='same', activation='relu', input_shape = (X_train.shape[1],1)))
    model.add(Conv1D(64, kernel_size=(3,), padding='same', activation='relu'))
    model.add(Conv1D(128, kernel_size=(5,), padding='same', activation='relu'))

    model.add(Flatten())

    model.add(Dense(50, activation='relu'))
    model.add(Dense(20, activation='relu'))
    model.add(Dense(units = 1))

    model.compile(loss='mean_squared_error', optimizer='adam')

    return model
```

Fig 5 CNN

A type of recurrent neural network designed for sequential data analysis is known as the Long Short-Term Memory, or LSTM. LSTMs are great for applications involving data points which have intricate distant associations since, unlike traditional RNNs, they are able to collect and hold dependencies for long periods of time. Specifically, to be able to accurately track sequential patterns, LSTMs employ particular memory cells and gates that enable remembering, updating, or forgetting the information. They have been employed in many application domains where capturing past context is essential and, hence, can predict future patterns. Some such applications include the analysis of financial time series, speech recognition, and natural language processing..

```
# Initialising the RNN
regressor = Sequential()
# Adding the first LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True, input_shape = (X_train.shape[1], 1)))
regressor.add(Dropout(0.2))

# Adding a second LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a third LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a fourth LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50))
regressor.add(Dropout(0.2))

# Adding the output layer
regressor.add(Dense(units = 1))
```

Fig 6 LSTM

The Multi-Source Long Short-Term Memory (MS-LSTM) is an extended variant of the traditional LSTM neural network designed to process data from various sources simultaneously. It excels at handling comprehensive information by integrating data inputs from multiple origins, making it particularly valuable for complex tasks such as stock price prediction. [30,32] MS-LSTM enhances the model's capacity to capture and analyze intricate dependencies and patterns by leveraging a broad range of data, thus improving the overall predictive capabilities of the system in scenarios where diverse data sources play a critical role.

```
# Initialising the RNN
regressor = Sequential()
# Adding the first LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True, input_shape = (X_train.shape[1], 1)))
regressor.add(Dropout(0.2))

# Adding a second LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a third LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a fourth LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50))
regressor.add(Dropout(0.2))

# Adding the output layer
regressor.add(Dense(units = 1))
```

```
# Compiling the RNN
regressor.compile(optimizer = 'adam', loss = 'mean_squared_error')

# Fitting the RNN to the Training set
regressor.fit(X_train, y_train, epochs = 100, batch_size = 32)
```

Fig 7 MS-LSTM

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The MS-SSA-LSTM model, or Multi-Source Sparrow Search Algorithm Long Short-Term Memory, represents a sophisticated approach to stock price prediction. It combines multi-source data from various origins, employs sentiment analysis, and optimizes the Long Short-Term Memory (LSTM) network using the Sparrow Search Algorithm (SSA). This advanced model effectively addresses the challenges of financial forecasting by offering a more accurate and robust way to predict stock prices. It outperforms conventional models and holds high universal applicability, making it a valuable tool for investors and enterprises operating in dynamic financial markets.

```
optimizer=SSA()
```

```
# Initialising the RNN
regressor = Sequential()
# Adding the first LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True, input_shape = (X_train.shape[1], 1)))
regressor.add(Dropout(0.2))

# Adding a second LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a third LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a fourth LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50))
regressor.add(Dropout(0.2))

# Adding the output layer
regressor.add(Dense(units = 1))
```

Fig 8 MS-SSA-LSTM

The Voting Regressor is an ensemble machine learning technique that combines the predictions of multiple regression algorithms to improve predictive performance. In this case, it incorporates three diverse regressors: Linear Regression, Random Forest Regressor, and k-Neighbors Regressor. By aggregating their individual predictions, it aims to create a more accurate and robust model for regression tasks. This approach leverages the strengths of each base regressor, such as the linearity of Linear Regression, the adaptability of Random Forest, and the proximity-based learning of k-Neighbors Regression, to enhance overall predictive capabilities.

```
r1 = LinearRegression()
r2 = RandomForestRegressor(n_estimators=10, random_state=1)
r3 = KNeighborsRegressor()

eclf1 = VotingRegressor([('lr', r1), ('rf', r2), ('r3', r3)])
eclf1.fit(X_train, y_train)
y_pred = eclf1.predict(X_train)
```

Fig 9 Voting Regressor

LSTM+GRU represents a sophisticated architecture of recurrent neural networks (RNN) that integrates the functionalities of long short-term memory (LSTM) and gated recurrent unit (GRU) components. This synthesis enhances the model's capacity to discern sequential patterns within datasets by capitalizing on the memory preservation characteristics of LSTM alongside the computational efficacy of GRU. Such an amalgamation proves particularly advantageous for applications pertaining to time series data, natural language processing, and the recognition of sequential patterns, as it mitigates the individual limitations inherent in each cell type, thereby yielding superior performance and augmented training efficiency.

```
model = Sequential()
model.add(Embedding(num_words, embed_dim,input_length = X_train.shape[1]))
model.add(LSTM(64,dropout=0.4, recurrent_dropout=0.4,return_sequences=True))
model.add(GRU(32,dropout=0.5, recurrent_dropout=0.5,return_sequences=False))
model.add(Dense(2,activation='softmax'))
model.compile(loss = 'categorical_crossentropy', optimizer='adam',metrics = ['accuracy',f1_m,recall_m,precision_m])
#print(model.summary())

trained5 = model.fit(X_train, Y_train, epochs = 20, batch_size=batch_size,validation_data=(X_test, Y_test),verbose = 1)
```

Fig 10 LSTM + GRU

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The Voting Classifier is a crucial part of this project's sentiment classification, combining the strengths of Random Forest (RF) and AdaBoost [18, 39]. It utilizes RF's ensemble learning technique, which combines predictions from several decision trees, and AdaBoost's boosting capabilities, which combine several weak learners to create a strong classifier. The Voting Classifier in our research will be a good tool for testing market sentiment as it combines both methods to increase the accuracy and robustness of sentiment classification.

```
from sklearn.ensemble import RandomForestClassifier, VotingClassifier, AdaBoostClassifier
clf1 = AdaBoostClassifier(n_estimators=100, random_state=0)
clf2 = RandomForestClassifier(n_estimators=50, random_state=1)

eclf1 = VotingClassifier(estimators=[('ad', clf1), ('rf', clf2)], voting='soft')
eclf1.fit(X_train, y_train)
y_pred = eclf1.predict(X_test)
```

Fig 11 Voting classifier

RESULTS

Precision: Precision evaluates the proportion of instances or samples accurately categorized within the subset identified as positive. Consequently, the equation utilized for the computation of precision is:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

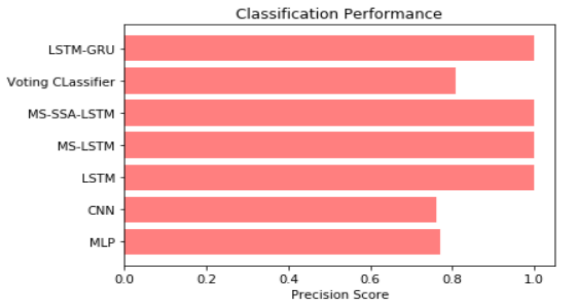

Fig 12 Precision comparison graph

Recall: In machine learning, recall is a measure used to determine how well a model is able to find all the relevant instances of a specific class. It provides information regarding how good a model is at finding instances of a specific class by dividing the number of correctly predicted positive observations with the total number of actual positives.

$$Recall = \frac{TP}{TP + FN}$$

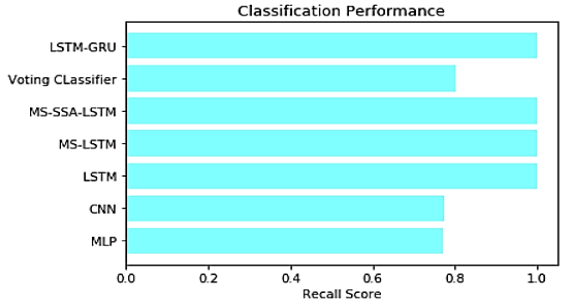

Fig 13 Recall comparison graph

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Accuracy: The percentage of accurate predictions in a classification task is known as accuracy, and it indicates how accurate a model's predictions are overall.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

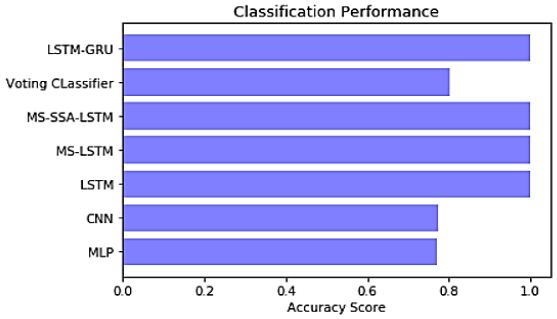

Fig 14 Accuracy graph

F1 Score: The F1 Score is a balanced measure that considers both false negatives and false positives and is suitable for unbalanced datasets. It is the harmonic mean of precision and recall..

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

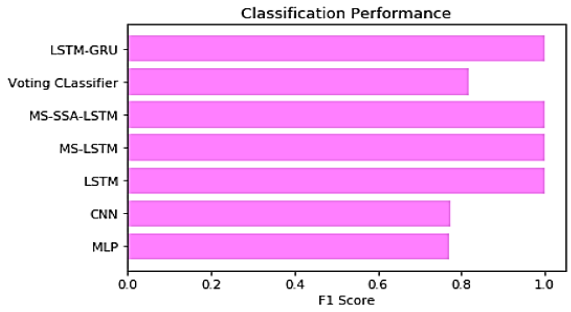

Fig 15 F1Score

	ML Model	Accuracy	Precision	Recall	F1-Score
0	MLP	0.771	0.771	0.771	0.770
1	CNN	0.773	0.761	0.773	0.774
2	LSTM	1.000	1.000	1.000	1.000
3	MS-LSTM	0.998	0.998	0.998	0.998
4	MS-SSA-LSTM	1.000	1.000	1.000	1.000
5	**Extension- Voting Classifier**	**0.803**	**0.808**	**0.803**	**0.819**
6	**Extension- LSTM-GRU**	**1.000**	**1.000**	**1.000**	**1.000**

Fig 16 Performance Evaluation

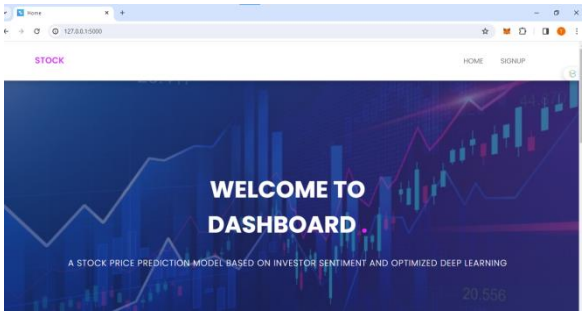

Fig 17 Home page of webinterface

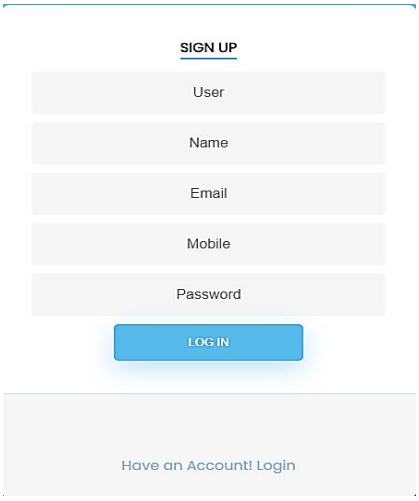

Fig 18Signin page

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

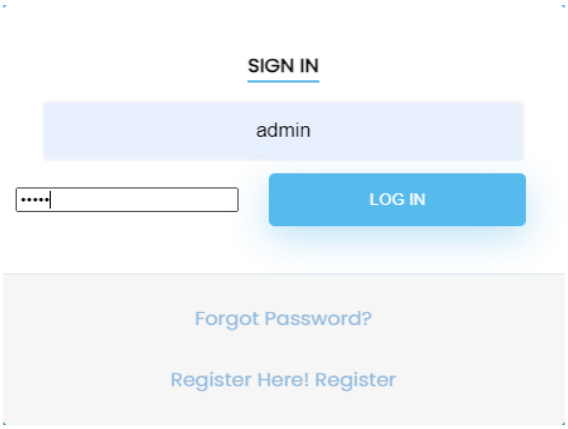

Fig 19 Login page

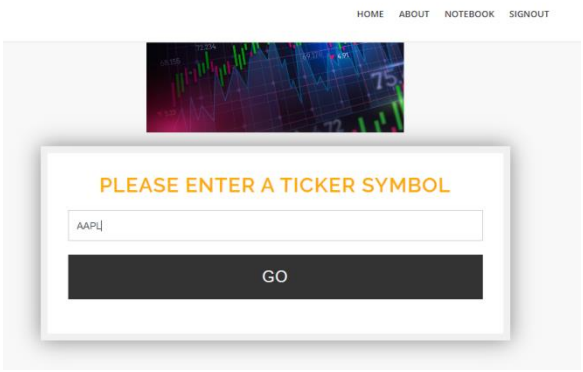

Fig 20User input for web interface

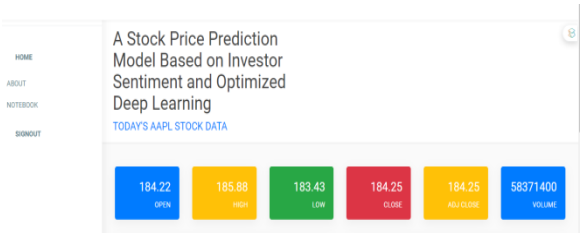

Fig 21Web interface for proposed model

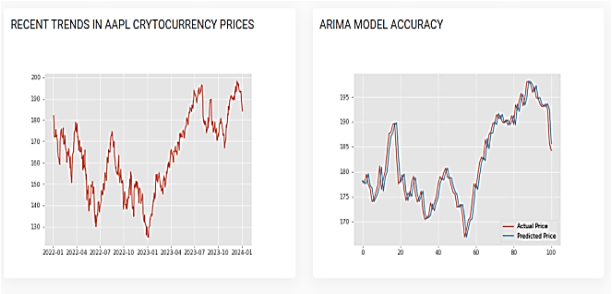

Fig 22 Arima Model Accuracy Graph

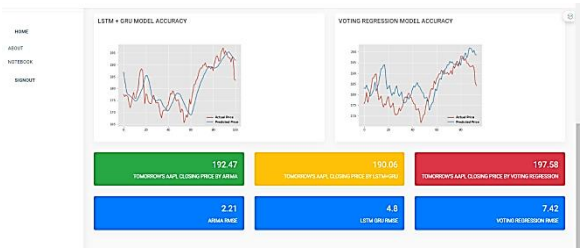

Fig 23 LSTM- GRU Model and Voting Regression Model Accuracy Graph

DISCUSSION

In order to enhance the accuracy of stock market predictions, the initiative was on the MS-SSA-LSTM model. A set of models was studied so that it could ensure a better focus on the context of sentiment analysis along with advanced algorithms for accurate prediction [26]. Since the MS-SSA-LSTM model was capable of both sentiment classification and stock price forecast, it served as the primary focus. It offered a rigorous approach to risk reduction as well as profit generation by using multiple sources of information and leading-edge methodologies. Although MS-SSA-LSTM was superior, the other models, namely MLP, CNN, LSTM, and MS-LSTM, were competent especially in the short-term forecasts of the dynamic market of China. Ensemble models, namely Voting Classifier, LSTM+GRU, and Voting Regressor, were added in the extension phase to the toolkit of the predictor. The Voting Regressor proved to be a reliable alternative, where the performance was better in stock price prediction, and LSTM+GRU performed exceptionally well in sentiment categorization. The Flask addon allowed users to interact more smoothly by enabling them to input ticker symbols for better predictions. The easy deployment of LSTM+GRU for sentiment analysis and Voting Regressor for stock price forecast contributed to improved investor and user access. Strong prediction algorithms and an intuitive interface will be helpful to traders, investors, and companies. In the dynamic Chinese financial market, the MS-SSA-LSTM model, along with its extensions, proves to be useful in providing some insightful information in reducing investment risk and improving the decision-making.

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

FUTURE SCOPE

With an extension of model functionality to deal with real-time data inputs, investors may be able to make judgments at a faster pace. It will also be helpful if data sources offer current information. [34] Further refining the sentiment analysis component could provide a more sophisticated understanding of market sentiment with the incorporation of sentiment-specific machine learning models and NLP techniques. By looking into and combining a number of data sources such as social media, news feeds, and macroeconomic indicators, quite good market understanding may be achieved and possibly more accurate forecasting. The model can be made more transparent and user-friendly by developing features or tools that give justifications for its predictions. Investors will be willing to understand the rationale for certain projections..With risk assessment and portfolio optimization added into the model's capacity, investors will be able to manage their assets more inclusively. This will include taking into consideration risk-adjusted returns and asset diversification.

REFRENCES

[1] M. M. Rounaghi and F. N. Zadeh, "Investigation of market efficiency and financial stability between S&P 500 and London stock exchange: Monthly and yearly forecasting of time series stock returns using ARMA model," Phys. A, Stat. Mech. Appl., vol. 456, pp. 10–21, Aug. 2016, doi: 10.1016/j.physa.2016.03.006.
[2] G. Bandyopadhyay, "Gold price forecasting using ARIMA model," J. Adv. Manage. Sci., vol. 4, no. 2, pp. 117–121, 2016, doi: 10.12720/joams.4.2.117-121.
[3] H. Shi, Z. You, and Z. Chen, "Analysis and prediction of Shanghai composite index by ARIMA model based on wavelet analysis," J. Math. Pract. Theory, vol. 44, no. 23, pp. 66–72, 2014.
[4] H. Herwartz, "Stock return prediction under GARCH—An empirical assessment," Int. J. Forecasting, vol. 33, no. 3, pp. 569–580, Jul. 2017, doi: 10.1016/j.ijforecast.2017.01.002.
[5] H. Mohammadi and L. Su, "International evidence on crude oil price dynamics: Applications of ARIMA-GARCH models," Energy Econ., vol. 32, no. 5, pp. 1001–1008, Sep. 2010, doi: 10.1016/j.eneco.2010.04.009.
[6] A. Hossain and M. Nasser, "Recurrent support and relevance vector machines based model with application to forecasting volatility of financial returns," J. Intell. Learn. Syst. Appl., vol. 3, no. 4, pp. 230–241, 2011, doi: 10.4236/jilsa.2011.34026.
[7] J. Chai, J. Du, K. K. Lai, and Y. P. Lee, "A hybrid least square support vector machine model with parameters optimization for stock forecasting," Math. Problems Eng., vol. 2015, pp. 1–7, Jan. 2015, doi: 10.1155/2015/231394.
[8] A. Murkute and T. Sarode, "Forecasting market price of stock using artificial neural network," Int. J. Comput. Appl., vol. 124, no. 12, pp. 11–15, Aug. 2015, doi: 10.5120/ijca2015905681.
[9] D. Banjade, "Forecasting Bitcoin price using artificial neural network," Jan. 2020, doi: 10.2139/ssrn.3515702.
[10] J. Zahedi and M. M. Rounaghi, "Application of artificial neural network models and principal component analysis method in predicting stock prices on Tehran stock exchange," Phys. A, Stat. Mech. Appl., vol. 438, pp. 178–187, Nov. 2015, doi: 10.1016/j.physa.2015.06.033.
[11] A. H. Moghaddam, M. H. Moghaddam, and M. Esfandyari, "Stock market index prediction using artificial neural network," J. Econ., Finance Administ. Sci., vol. 21, no. 41, pp. 89–93, Dec. 2016, doi: 10.1016/j.jefas.2016.07.002.
[12] H. Liu and Y. Hou, "Application of Bayesian neural network in prediction of stock time series," Comput. Eng. Appl., vol. 55, no. 12, pp. 225–229, 2019.
[13] A. M. Rather, A. Agarwal, and V. N. Sastry, "Recurrent neural network and a hybrid model for prediction of stock returns," Expert Syst. Appl., vol. 42, no. 6, pp. 3234–3241, Apr. 2015, doi: 10.1016/j.eswa.2014.12.003.
[14] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," Phys. D, Nonlinear Phenomena, vol. 404, Mar. 2020, Art. no. 132306, doi: 10.1016/j.physd.2019.132306.
[15] G. Ding and L. Qin, "Study on the prediction of stock price based on the associated network model of LSTM," Int. J. Mach. Learn. Cybern., vol. 11, no. 6, pp. 1307–1317, Nov. 2019, doi: 10.1007/s13042-019-01041-1.
[16] X. Yan, W. Weihan, and M. Chang, "Research on financial assets transaction prediction model based on LSTM neural network," Neural Comput. Appl., vol. 33, no. 1, pp. 257–270, May 2020, doi: 10.1007/s00521-020-04992-7.
[17] M. Nabipour, P. Nayyeri, H. Jabani, A. Mosavi, E. Salwana, and S. Shahab, "Deep learning for stock market prediction," Entropy, vol. 22, no. 8, p. 840, Jul. 2020, doi: 10.3390/e22080840.
[18] Z. D. Aksehir and E. Kiliç, "How to handle data imbalance and feature selection problems in CNN-based stock price forecasting," IEEE Access, vol. 10, pp. 31297–31305, 2022, doi: 10.1109/ACCESS.2022. 3160797.
[19] Y. Ji, A. W. Liew, and L. Yang, "A novel improved particle swarm optimization with long-short term memory hybrid model for stock indices forecast," IEEE Access, vol. 9, pp. 23660–23671, 2021, doi: 10.1109/ACCESS.2021.3056713.

Research Article

[20] X. Zeng, J. Cai, C. Liang, and C. Yuan, "A hybrid model integrating long short-term memory with adaptive genetic algorithm based on individual ranking for stock index prediction," PLoS ONE, vol. 17, no. 8, Aug. 2022, Art. no. e0272637, doi: 10.1371/journal.pone.0272637.
[21] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: Sparrow search algorithm," Syst. Sci. Control Eng., vol. 8, no. 1, pp. 22–34, Jan. 2020, doi: 10.1080/21642583.2019.1708830.
[22] J. Borade, "Stock prediction and simulation of trade using support vector regression," Int. J. Res. Eng. Technol., vol. 7, no. 4, pp. 52–57, Apr. 2018, doi: 10.15623/ijret.2018.0704009.
[23] X. Li and P. Tang, "Stock price prediction based on technical analysis, fundamental analysis and deep learning," Stat. Decis., vol. 38, no. 2, pp. 146–150, 2022, doi: 10.13546/j.cnki.tjyjc.2022.02.029.
[24] J. Heo and J. Y. Yang, "Stock price prediction based on financial statements using SVM," Int. J. Hybrid Inf. Technol., vol. 9, no. 2, pp. 57–66, Feb. 2016, doi: 10.14257/ijhit.2016.9.2.05.
[25] J. B. De Long, A. Shleifer, L. H. Summers, and R. J. Waldmann, "Noise trader risk in financial markets," J. Political Economy, vol. 98, no. 4, pp. 703–738, Aug. 1990, doi: 10.1086/261703.
[26] H. Cui and Y. Zhang, "Does investor sentiment affect stock price crash risk?" Appl. Econ. Lett., vol. 27, no. 7, pp. 564–568, Jul. 2019, doi: 10.1080/13504851.2019.1643448.
[27] R. P. Schumaker, Y. Zhang, C.-N. Huang, and H. Chen, "Evaluating sentiment in financial news articles," Decis. Support Syst., vol. 53, no. 3, pp. 458–464, Jun. 2012, doi: 10.1016/j.dss.2012.03.001.
[28] M. Nofer and O. Hinz, "Using Twitter to predict the stock market," Bus. Inf. Syst. Eng., vol. 57, no. 4, pp. 229–242, Jun. 2015, doi: 10.1007/s12599- 015-0390-4.
[29] P. Fan, Y. Yang, Z. Zhang, and M. Chen, "The relationship between individual stock investor sentiment and stock yield-based on the perspective of stock evaluation information," Math. Pract. Theory, vol. 51, no. 16, pp. 305–320, 2021.
[30] Z. Jin, Y. Yang, and Y. Liu, "Stock closing price prediction based on sentiment analysis and LSTM," Neural Comput. Appl., vol. 32, no. 13, pp. 9713–9729, Sep. 2019, doi: 10.1007/s00521-019-04504-2.
[31] X. Xu and K. Tian, "A novel financial text sentiment analysis-based approach for stock index prediction," J. Quantum Technol. Econ., vol. 38, no. 12, pp. 124–145, 2021, doi: 10.13653/j.cnki.jqte.2021.12.009.
[32] C.-R. Ko and H.-T. Chang, "LSTM-based sentiment analysis for stock price forecast," PeerJComput. Sci., vol. 7, p. e408, Mar. 2021, doi: 10.7717/peerj-cs.408.
[33] Y. Li and Y. Pan, "A novel ensemble deep learning model for stock prediction based on stock prices and news," Int. J. Data Sci. Anal., vol. 13, no. 2, pp. 139–149, Sep. 2021, doi: 10.1007/s41060-021-00279-9.
[34] C. Kearney and S. Liu, "Textual sentiment in finance: A survey of methods and models," Int. Rev. Financial Anal., vol. 33, pp. 171–185, May 2014, doi: 10.1016/j.irfa.2014.02.006.
[35] T. Wang and Z. Zhang, "Research on the construction method of emotional lexicon for movie review," Comput. Digit. Eng., vol. 50, no. 4, pp. 843–848, 2022, doi: 10.3969/j.issn.1672-9722.2022.04.031.
[36] Y. Rao, J. Lei, L. Wenyin, Q. Li, and M. Chen, "Building emotional dictionary for sentiment analysis of online news," World Wide Web, vol. 17, no. 4, pp. 723–742, Jun. 2013, doi: 10.1007/s11280-013-0221-9.
[37] L. Yang and T. Zhai, "Research on sentiment tendency analysis of video reviews based on sentiment dictionary," Netw. Secur. Technol. Appl., vol. 255, no. 3, pp. 53–56, 2022.
[38] A. Fathy, T. M. Alanazi, H. Rezk, and D. Yousri, "Optimal energy management of micro-grid using sparrow search algorithm," Energy Rep., vol. 8, pp. 758–773, Nov. 2022, doi: 10.1016/j.egyr.2021. 12.022.
[39] Y. Chen, Z. Liu, C. Xu, X. Zhao, L. Pang, K. Li, and Y. Shi, "Heavy metal content prediction based on random forest and sparrow search algorithm," J. Chemometrics, vol. 36, no. 10, Sep. 2022, Art. no. e3445, doi: 10.1002/cem.3445.
[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735