IJITCE

ISSN 2347- 3657

International Journal of Information Technology & Computer Engineering

Email : ijitce.editor@gmail.com or editor@ijitce.com

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657
Volume 11, Issue 4, 2023

A Deep Learning-Based Model for Stock Price Forecasting Leveraging Investor Sentiment

Mrs.A.Anuradha[1]., Afshan Fareed [2]

1 Assistant Professor, Department of CSE, Malla Reddy College of Engineering for Women., Maisammaguda., Medchal., TS, India

2, B.Tech CSE (20RG1A0562),

Malla Reddy College of Engineering for Women., Maisammaguda., Medchal., TS, India

Abstract: The research presents the MS-SSA-LSTM model, which integrates multi-source data, sentiment analysis, swarm intelligence algorithms, and deep learning techniques to enhance stock price predictions. This model incorporates sentiment analysis from East Money forum posts, creating a unique sentiment dictionary and calculating a sentiment index. This offers valuable insights into market sentiment's influence on stock prices. The Sparrow Search Algorithm (SSA) is used to fine-tune LSTM hyperparameters, optimizing prediction accuracy. • Experimental results showcase the MS-SSA-LSTM model's superior performance. It's a valuable tool for accurate stock price predictions. Tailored for China's volatile financial market, the model excels in short-term stock price predictions, offering insights for dynamic decision-making by investors.And also, a hybrid LSTM+GRU model was introduced for stock sentiment classification. Additionally, a robust ensemble strategy was adopted, incorporating a Voting Classifier (AdaBoost + RandomForest) for sentiment analysis and a Voting Regressor (LinearRegression + RandomForestRegressor + KNeighborsRegressor) for stock price prediction. These ensembles seamlessly integrated with existing models (MLP, CNN, LSTM, MS-LSTM, MS-SSA-LSTM), collectively enhancing overall predictive performance. To facilitate user interaction and testing, a user-friendly Flask framework with SQLite support was developed, streamlining signup, signin, and model evaluation processes.

Index terms -*Deep learning, LSTM model, stock price prediction, sentiment analysis, sentiment dictionary, sparrow search algorithm.*

1. INTRODUCTION

With the maturity of China’s stock market and the rapid growth of Internet finance, many people realize the importance of investment and choose to enter the financial market. However, the stock market is characterized by massive data and enormous volatility. Many retail investors need more data-mining skills to make money. Therefore, accurate stock price prediction can reduce investment risks and improve investment returns for investors and enterprises.

Early scholars used statistical methods to construct a linear model to fit the stock price time series trend. The traditional methods contain ARMA, ARIMA, GARCH, etc. The ARMA is established to conduct a time series stock analysis [1]. The ARIMA model is developed based on the ARMA and predicts the trend of stock price changes [2]. The ARIMAmodel can also introduce wavelet analysis to improve the fitting accuracy of the Shanghai Composite Index [3]. The GARCH model provides innovative ideas

for stock time series prediction through a time window [4]. At the same time, some scholars have combined ARMA and GARCH to build a new prediction model, which provided theoretical support for the volumetric price analysis of multivariate stocks [5]. Generally speaking, these classical methods only capture regular and structured data. However, traditional forecasting methods require assumptions that are uncommon in real life. Therefore, It is challenging to describe nonlinear financial data using statistical methods.

Subsequently, many researchers attempt to anticipate stock prices using machine learning approaches such as Support Vector Machines (SVM) and Neural Networks. Machine learning's core idea is to use algorithms to parse data, learn from it, and make predictions about new data. Because the SVM shows unique benefits in dealing with limited samples,high-dimensional data, and nonlinear situations, many scholars use it in stock forecasting. Hossain and Nasser [6] found that the SVM method is superior to the statistical ones in stock prediction accuracy. Chai et al. [7] suggested a hybrid SVM model to anticipate the HS300 index's ups and downs and found that the least squares SVM combined with the Genetic Algorithm (GA) performed better. However, when the SVM applies to large-scale training samples, much memory and computing time will be consumed, which may limit its development space in predicting a large amount of stock data. Then, Artificial Neural Networks (ANN) and multi-layer ANN address financial time series issues. According to the experimental data, ANN has the benefits of quick convergence and high accuracy [8], [9], [10]. Moghaddam and Esfandyari [11] evaluated the effect of several feedforward artificial neural networks on the market stock price forecast through experiments. Liu and Hou [12] improved the BP (Back Propagation) neural network using the Bayesian regularization method. Nevertheless, the traditional neural network method has the following areas for improvement. Generalization ability is not strong, quickly leads to overfitting, and falls into local optimization. Since many samples need to be trained, better models must be found to solve these problems.

A new model for predicting stock prices is proposed in this paper (MS-SSA-LSTM), which matches the characteristics of multi-source data with LSTM neural networks and uses the Sparrow Search Algorithm. The MS-SSA-LSTM stock price forecast model can forecast the stock price in advance and help investors and traders make more informed investment decisions. Investors and traders obtain the data of individual stocks they want to invest in, including historical transaction data and comment information of stock market shareholders, and input them into the MS-SSA-LSTM model. The model automatically outputs a stock price trend chart and forecasts the stock price for the next day

2. LITERATURE SURVEY

The presence and changes in, long memory features in the returns and volatility dynamics of S&P 500 and London Stock Exchange using ARMA model [1]. Recently, multifractal analysis has been evolved as an important way to explain the complexity of financial markets which can hardly be described by linear methods of efficient market theory. In financial markets, the weak form of the efficient market hypothesis implies that price returns are serially uncorrelated sequences. In other words, prices should

follow a random walk behavior. The random walk hypothesis is evaluated against alternatives accommodating either unifractality or multifractality. Several studies find that the return volatility of stocks tends to exhibit long-range dependence, heavy tails, and clustering. Because stochastic processes with self-similarity possess long-range dependence and heavy tails, it has been suggested that self-similar processes be employed to capture these characteristics in return volatility modeling. The present study applies monthly and yearly forecasting of Time Series Stock Returns in S&P 500 and London Stock Exchange using ARMA model. [1] The statistical analysis of S&P 500 shows that the ARMA model for S&P 500 outperforms the London stock exchange and it is capable for predicting medium or long horizons using real known values. The statistical analysis in London Stock Exchange shows that the ARMA model for monthly stock returns outperforms the yearly. A comparison between S&P 500 and London Stock Exchange shows that both markets are efficient and have Financial Stability during periods of boom and bust.

The study gives an inside view of the application of ARIMA time series model to forecast the future Gold price in Indian browser based on past data from November 2003 to January 2014 to mitigate the risk in purchases of gold. Hence, to give guideline for the investor when to buy or sell the yellow metal. [2]This financial instrument has gained a lot of momentum in recent past as Indian economy is curbed with factors like changing political scenario, global clues & high inflation etc, so researcher, investors and speculators are in search of different financial instrument to minimize their risk by portfolio diversification. Gold earlier was only purchased at the time of marriage or other rituals in India but now it has gained importance in the eyes of investors also, so it has become necessary to predict the price of Gold with suitable method.

The GARCH model and its numerous variants have been applied widely both in the financial literature and in practice. For purposes of quasi maximum likelihood estimation, innovations to GARCH processes are typically assumed to be identically and independently distributed, with mean zero and unit variance (strong GARCH) [4]. Under less restrictive assumptions (the absence of unconditional correlation, weak GARCH), higher order dependency patterns might be exploited for the ex ante forecasting of GARCH innovations, and hence, stock returns. In this paper, rolling windows of empirical stock returns are used to test the independence of consecutive GARCH innovations. Rolling -values from independence testing reflect the time variation of serial dependence, and provide useful information for signaling one-step-ahead directions of stock price changes. Ex ante forecasting gains are documented for nonparametric innovation predictions, especially if the sign of the innovation predictors is combined with independence diagnostics (-values) and/or the sign of linear return forecasts.

In the recent years, the use of GARCH type (especially, ARMA-GARCH) models and computational-intelligence-based techniques—Support Vector Machine (SVM) and Relevance Vector Machine (RVM) have been successfully used for financial forecasting. [2,6]This paper deals with the application of ARMA-GARCH, recurrent SVM (RSVM) and recurrent RVM (RRVM) in volatility forecasting. Based on RSVM and RRVM, two GARCH methods are used and are compared with parametric

GARCHs (Pure and ARMA-GARCH) in terms of their ability to forecast multi-periodically. These models are evaluated on four performance metrics: MSE, MAE, DS, and linear regression R squared. The real data in this study uses two Asian stock market composite indices of BSE SENSEX and NIKKEI225. This paper also examines the effects of outliers on modeling and forecasting volatility. Our experiment shows that both the RSVM and RRVM perform almost equally, but better than the GARCH type models in forecasting. The ARMA-GARCH model is superior to the pure GARCH and only the RRVM with RSVM hold the robustness properties in forecasting.

This paper proposes an EMD-LSSVM (empirical mode decomposition least squares support vector machine) model to analyze the CSI 300 index. A WD-LSSVM (wavelet denoising least squares support machine) is also proposed as a benchmark to compare with the performance of EMD-LSSVM [7]. Since parameters selection is vital to the performance of the model, different optimization methods are used, including simplex, GS (grid search), PSO (particle swarm optimization), and GA (genetic algorithm). Experimental results show that the EMD-LSSVM model with GS algorithm outperforms other methods in predicting stock market movement direction.

3. METHODOLOGY

i) Proposed Work:

The project introduces the MS-SSA-LSTM model, a cutting-edge system for stock price prediction. This model seamlessly integrates multi-source data, sentiment analysis, and swarm intelligence algorithms. [14,15,16,30] By optimizing LSTM hyperparameters with the Sparrow Search Algorithm, the system excels in forecasting stock prices with exceptional accuracy. Experimental results affirm its superiority over other models, underlining its universal applicability and potential to enhance predictive performance. This model is compared with MLP, CNN, LSTM, MS-LSTM.And also, a hybrid LSTM+GRU model was introduced for stock sentiment classification. Additionally, a robust ensemble strategy was adopted, incorporating a Voting Classifier (AdaBoost + Random Forest) for sentiment analysis and a Voting Regressor (LinearRegression + Random Forest Regressor + K-Neighbors Regressor) for stock price prediction. These ensembles seamlessly integrated with existing models (MLP, CNN, LSTM, MS-LSTM, MS-SSA-LSTM), collectively enhancing overall predictive performance. To facilitate user interaction and testing, a user-friendly Flask framework with SQLite support was developed, streamlining signup, signin, and model evaluation processes.

ii) System Architecture:

The initial step is to import datasets, including the Stock Tweets Dataset, Single Stock Data, and Multi-Source Data. These datasets serve as the foundation for both sentiment analysis and stock price prediction. Text data from the Stock Tweets Dataset undergoes cleaning, which includes removing punctuations, HTML tags, URLs, and emojis. This step ensures the text is ready for sentiment analysis. The Single Stock Data and Multi-Source Data are processed to handle null values, remove duplicates, and scale the data. This prepares the financial data for stock price prediction. Several models, including

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 11, Issue 4, 2023

MLP, CNN, LSTM, MS-LSTM, MS-SSA-LSTM, extensions- Voting Classifier, and LSTM + GRU, are trained for sentiment classification. They analyze the cleaned tweet data to determine market sentiment. Another set of models, including MLP, CNN, LSTM, MS-LSTM, MS-SSA-LSTM, and extension- Voting Regression, are trained for stock price prediction. They utilize processed financial data to forecast stock prices. After the models are trained, they are used to make predictions. In the case of sentiment analysis, predictions provide insights into market sentiment. For stock price prediction, the models forecast future stock prices. The predictions from sentiment analysis and stock price models play a crucial role in aiding investors and traders in making informed decisions. The combined results help users navigate the complex landscape of the stock market, reduce risks, and optimize investment returns.

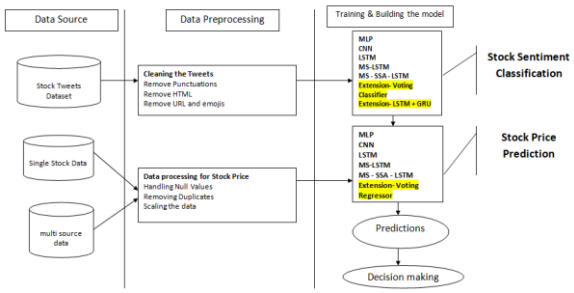


Fig 1 Proposed architecture

iii) Dataset collection:

STOCK TWEETS DATASET

The "Stock Tweets" dataset includes posts about stocks and financial markets on social media. We used it to understand people's feelings and reactions to market news [1,4,7,8]. This helped us create tools for stock trading and investments. We wanted to see how social media affects stock prices and market trends to help investors and traders.

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

The "All Stock Dataset" is a comprehensive collection of financial data from various sources. It provides a wealth of information for in-depth stock market research. In our project, we used this dataset to enhance our stock price prediction model. We aimed to improve the accuracy of stock price forecasts by leveraging diverse data sources, ultimately benefitting investors and businesses.

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 11, Issue 4, 2023

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 dataset

iv) Data Processing:

Data processing involves transforming raw data into valuable information for businesses. Generally, data scientists process data, which includes collecting, organizing, cleaning, verifying, analyzing, and converting it into readable formats such as graphs or documents. Data processing can be done using three methods i.e., manual, mechanical, and electronic. The aim is to increase the value of information and facilitate decision-making. This enables businesses to improve their operations and make timely strategic decisions. Automated data processing solutions, such as computer software programming, play a significant role in this. It can help turn large amounts of data, including big data, into meaningful insights for quality management and decision-making.

v) Feature selection:

Feature selection is the process of isolating the most consistent, non-redundant, and relevant features to use in model construction. Methodically reducing the size of datasets is important as the size and variety of datasets continue to grow. The main goal of feature selection is to improve the performance of a predictive model and reduce the computational cost of modeling.

Feature selection, one of the main components of feature engineering, is the process of selecting the most important features to input in machine learning algorithms. Feature selection techniques are employed to reduce the number of input variables by eliminating redundant or irrelevant features and narrowing down the set of features to those most relevant to the machine learning model. 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 final output layer generates predictions or classifications. MLPs are used in a wide range of applications, from image recognition to financial forecasting, owing to their capacity to model intricate relationships in data.

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 11, Issue 4, 2023

```
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.

```
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 **Long Short-Term Memory** (LSTM) is a type of recurrent neural network (RNN) designed for sequential data analysis. Unlike traditional RNNs, LSTMs are adept at capturing and preserving dependencies over long sequences, making them ideal for tasks where data points have complex, distant relationships. LSTMs utilize specialized memory cells and gates that enable them to remember, update, or forget information, facilitating precise modeling of sequential patterns. This has found applications in various fields, including natural language processing, speech recognition, and financial time series analysis, where understanding historical context and predicting future trends are crucial.

```
# 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

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 11, Issue 4, 2023

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

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

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 11, Issue 4, 2023

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

The **LSTM+GRU** is an advanced recurrent neural network (RNN) architecture that combines the capabilities of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) cells. It enhances the model's ability to capture sequential patterns in data by leveraging both LSTM's memory retention and GRU's computational efficiency. This combination is particularly effective for tasks involving time series data, natural language processing, and sequential pattern recognition, as it addresses the limitations of each cell type individually, resulting in improved performance and 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

The Voting Classifier is a key component for sentiment classification in this project, combining the strengths of AdaBoost and Random Forest (RF) [18,39]. It harnesses AdaBoost's boosting capabilities, where multiple weak learners are combined to form a strong classifier, and RF's ensemble learning approach, which aggregates predictions from multiple decision trees. By integrating these two techniques, the Voting Classifier enhances the accuracy and robustness of sentiment classification, making it a powerful tool for analyzing market sentiment in our research.

```
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

4. EXPERIMENTAL RESULTS

International Journal of Information Technology & Computer Engineering
ISSN 2347-3657
Volume 11, Issue 4, 2023

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

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

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

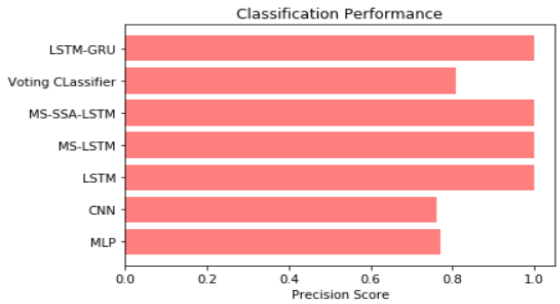


Fig 12 Precision comparison graph

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

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

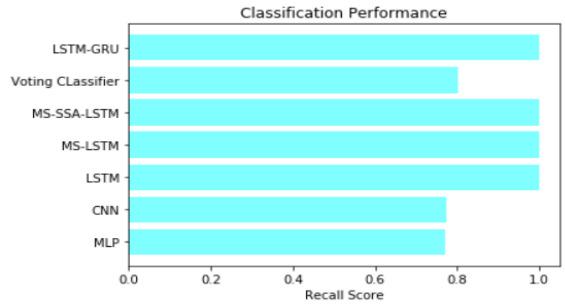


Fig 13 Recall comparison graph

Accuracy: Accuracy is the proportion of correct predictions in a classification task, measuring the overall correctness of a model's predictions.

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 11, Issue 4, 2023

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

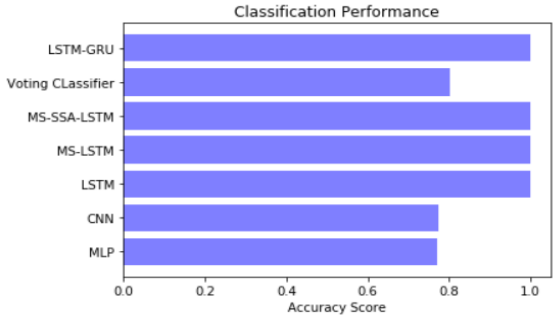


Fig 14 Accuracy graph

F1 Score: The F1 Score is the harmonic mean of precision and recall, offering a balanced measure that considers both false positives and false negatives, making it suitable for imbalanced datasets.

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

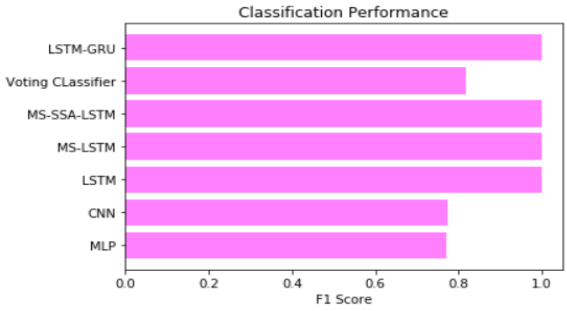


Fig 15 F1Score

	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

International Journal of Information Technology & Computer Engineering
ISSN 2347-3657
Volume 11, Issue 4, 2023

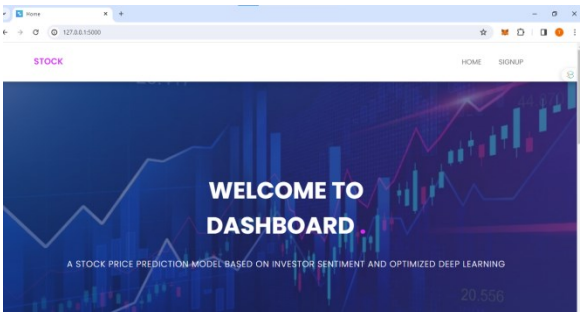


Fig 17 Home page

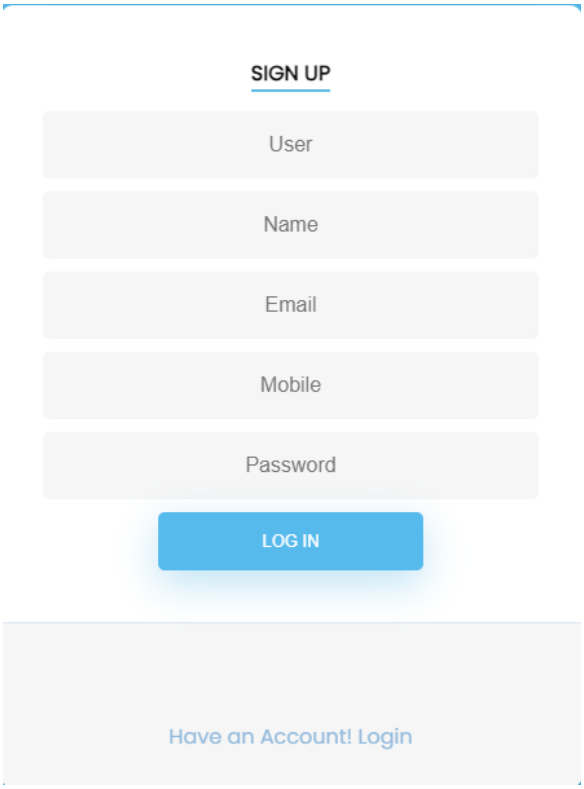
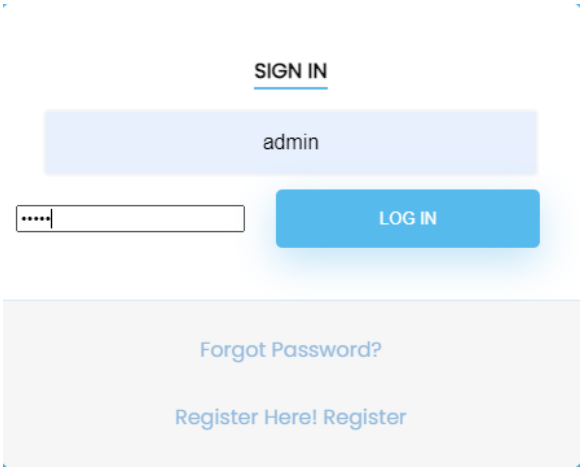


Fig 18 Signin page



International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 11, Issue 4, 2023

Fig 19 Login page

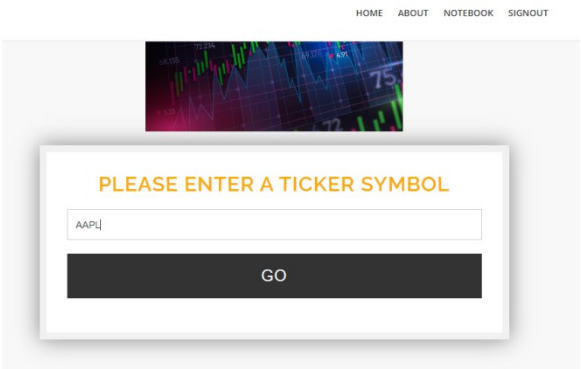


Fig 20User input

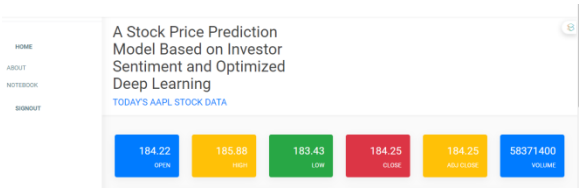


Fig 21Result

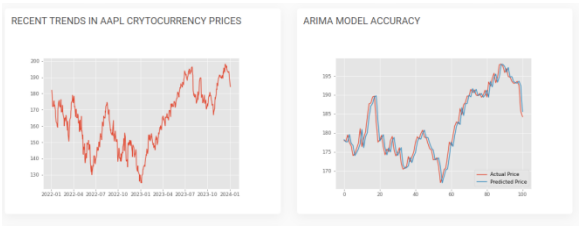


Fig 22 Graphs

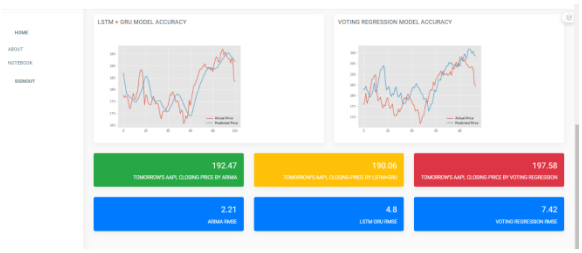


Fig 23 Graphs

5. CONCLUSION

The project aimed to enhance stock market predictions, with a focus on the MS-SSA-LSTM model. The research explored various models, emphasizing the significance of sentiment analysis and innovative algorithms for optimized forecasting [26]. The MS-SSA-LSTM model stood out for its dual

proficiency in stock price prediction and sentiment classification. Leveraging diverse data sources and advanced techniques, it offered a comprehensive approach to risk reduction and improved returns. Existing models (MLP, CNN, LSTM, MS-LSTM) demonstrated competence, while the MS-SSA-LSTM model showcased superiority, particularly in short-term predictions for China's dynamic market. Ensemble models (Voting Classifier, LSTM+GRU, Voting Regressor) introduced in the extension phase expanded the predictive toolkit. LSTM+GRU excelled in sentiment classification, and the Voting Regressor outperformed in stock price prediction, contributing reliable alternatives. The Flask extension facilitated user-friendly interaction, allowing input of ticker symbols for accurate predictions. LSTM+GRU for sentiment and Voting Regressor for stock price predictions were seamlessly deployed, enhancing accessibility for users and investors. Investors, traders, and businesses stand to benefit from the project's robust predictive models and user-friendly interface. The MS-SSA-LSTM model and its extensions offer valuable insights, reducing investment risks, and enhancing decision-making in the dynamic landscape of the Chinese financial market.

6. FUTURE SCOPE

Expanding the model's capabilities to handle real-time data feeds can enable investors to make even more timely decisions. Integrating data sources that provide up-to-the-minute information could be a valuable addition. [34] Further refining the sentiment analysis component by incorporating natural language processing (NLP) techniques and sentiment-specific machine learning models can provide a more nuanced understanding of market sentiment. Exploring and integrating data from diverse sources, such as social media, news feeds, and macroeconomic indicators, can offer a comprehensive view of the market and potentially improve predictive accuracy. Developing tools or features that offer explanations for the model's predictions can make it more transparent and user-friendly. Investors may benefit from understanding the reasons behind specific forecasts. Extending the model's capabilities to include risk assessment and portfolio optimization can provide investors with a holistic approach to managing their investments. This could involve considering the diversification of assets and risk-adjusted returns.

REFERENCES

[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.

[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,’’ PeerJ Comput. 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.