Stock Market Prediction Based On Political Impact Using Zasl-Ecc And Tlsaaf-Bilstm

Narasimha Rao Konangi

Senior Manager, Software Engineering
Capital One
Atlanta, GA.

DOI: 10.29322/IJSRP.13.07.2023.p13931
http://dx.doi.org/10.29322/IJSRP.13.07.2023.p13931

Paper Received Date: 14th June 2023
Paper Acceptance Date: 19th July 2023
Paper Publication Date: 30th July 2023

Abstract: Precise processing of financial big data, which could aid in predicting future Stock Market (SM), can provide insights into market trends. But, the prevailing prediction model lacks in providing secure forecasted outcomes. Hence, this article proposes a secure SM prediction scheme. Primarily, historical SM data along with real-time SM financial data and technical indicators are processed with time series windowing. After that, the time-series features are extracted. In the meantime, for analyzing the political influence in the SM, the political and stock-related tweets are extracted, pre-processed, and grouped grounded on the keyword. From the grouped SM data, the numerical and alpha-numerical data are extracted from which numerical data is given for correlation analysis along with the time series features, and alphanumeric data is given for sentimental analysis. For SM prediction, the political data is embedded and given to Transfer Learning- Shape Autotuning Activation Function-Bidirectional Long Short Term Memory (TLSAAF-BiLSTM) along with the times series features, correlation value, and sentiment score. This predicted outcome is encrypted with Zaslavsky Map-Elliptic Curve Cryptography (Zasl-ECC) and stored in the blockchain. Lastly, to securely retrieve the SM-predicted data, the users are authenticated. The proposed mechanism's security and efficacy are proved by the experimental analysis.

Keywords*: Stock Market (SM), Kernelized Total Bregman Divergence Means Algorithm (KTBDMA), Bidirectional Encoder Representations from Transformers (BERT), Transfer Learning-Shape Autotuning Activation Function-Bidirectional Long Short Term Memory (TLSAAF-BiLSTM), Zaslavsky Map-Elliptic Curve Cryptography (Zasl-ECC).*

1. INTRODUCTION

A huge amount of structured and unstructured information is enclosed in big data (Awan et al., 2021). Its analytics aid to manage enormous volumes of data produced by several businesses and could be wielded for forecasting trends (Gupta & Sharma, 2020). In recent times, the main challenge to investors is SM forecasting. SMs have a vast financial impact on the world economy, which was well evident by the market crashes in 2008 and 2020 (Hoseinzade& Haratizadeh, 2019). It is considered to be a stochastic and challenging real-world environment in which stock price movements are influenced by a significant number of factors, namely social media data, historical pricing, news, government bonds, and national economics (Achyutha et al., 2022) (Thakkar & Chaudhari, 2020). Stock prices are guaranteed to be erratic when these factors are combined. However, precise prediction becomes exceedingly challenging (Aasi et al., 2021).

Time series models, which comprise the Auto-Regressive Integrated Moving Average (Moghar & Hamiche, 2020), Auto-Regressive Conditional Heteroscedastic (ARCH), and Generalized ARCH, are wielded for forecasting SM (Gandhmal & Kumar, 2019). However, the majority of them rely on stationary patterns. Hence, price prediction deals with inherent difficulty (Nabipour et al., 2020). Researchers frequently utilized Support Vector Machines (Rouf et al., 2021), Logistic Regression, Random Forests, et cetera to categorize future price forecasts with the development of Machine Learning (ML). But, the outcomes have not been promising (Jiang, 2021). In SM prediction, deep learning architectures have been quite successful owing to their capacity to extract high-level features (Lin et al., 2022). Among them, LSTM is beneficial for modeling the time series behaviour of SMs since it captures temporal activity (Wu et al., 2022). But, these models could not securely forecast the predictions. Thus, a secured SM prediction model is proposed here.

1.1 Problem Statement

Prevailing SM prediction models had the following shortcomings,

- Existing works did not focus on resolving the security issues of SM prediction.
- Previously, social media data was considered, which is unreliable.
- Existing works did not evaluate past events with recent trends.

Hence, to alleviate these drawbacks, this paper proposes a secured SM prediction framework. Its contributions are,

- To securely forecast the results, the Zasl-ECC is presented.
- To provide reliable results, political news-based tweets and termly financial data are utilized.
- Correlation analysis is performed grounded on the historical and current stock trend data.

The remaining paper is systematized as: Section 2 analyses the associated works. The proposed technique and its outcomes are elucidated in Sections 3 and 4. Lastly, Section 5 ends with a conclusion.

2. LITERATURE SURVEY

(Shah et al., 2022) recommended a system grounded on a Convolutional Neural Network (CNN) merged with LSTM to predict SM. For anticipating the movement of the following day, the system gathered features and wielded time series modeling. An enhancement was shown by the framework. But, it did not consider the policy changes and perceptions of investors.

(Xu et al., 2020) employed an SM Prediction centered on the tweet and historic prices. To provide information, the network combined the local and contextual attention mechanisms. The results demonstrated better performance. Nevertheless, the accuracy level was degraded owing to its low-learning efficiency.

(Jaggi et al., 2021) proffered a strategy grounded on historical stock prices as well as financial messages for SM prediction. The system was developed with Fin-A Lite BERT (FinALBERT) and the dataset was labeled. The superior outcomes were given by the percentage change technique with two labels. Yet, FinALBERT training time was too high.

(Rezaei et al., 2021) established hybrid systems, namely Complete Ensemble Empirical Mode Decomposition based CNN-LSTM and Empirical Mode Decomposition-CNN-LSTM for extracting deep features and time sequences for SM prediction. Outcomes displayed that CNN outperformed with higher accuracy. However, the collaboration procedure consumed more time.

(Deepika & Nirupama Bhat, 2021) propounded an effective SM prediction technique grounded on the Kalman Filter (KF). The KF smoothed the noise; also, the Accelerated Gradient LSTM (AG-LSTM) predicted the SM value. The AG-LSTM with KF achieved higher accuracy. Nevertheless, even with KF, the model could not improve the prediction for the dataset with an outlier.

(Nti et al., 2021) recommended a multi-source information-fusion technique for SM prediction. The fusion model was developed with CNN and LSTM. As per the outcomes, good prediction accuracy was achieved. But, the model consumed more time for training; also, it required more memory for the fusion process.

(Khan et al., 2022) presented an SM prediction model using ML classifiers. For discovering the impact of data on prediction accuracy, social media news, as well as financial news data, were wielded. Outcomes displayed that the highest prediction accuracies were achieved. Yet, it was difficult to afford the prediction model's time overhead.

3. PROPOSED SECURED SM PREDICTION

This work proposes a secured framework for SM prediction. Figure 1 elucidates the steps involved in this framework.

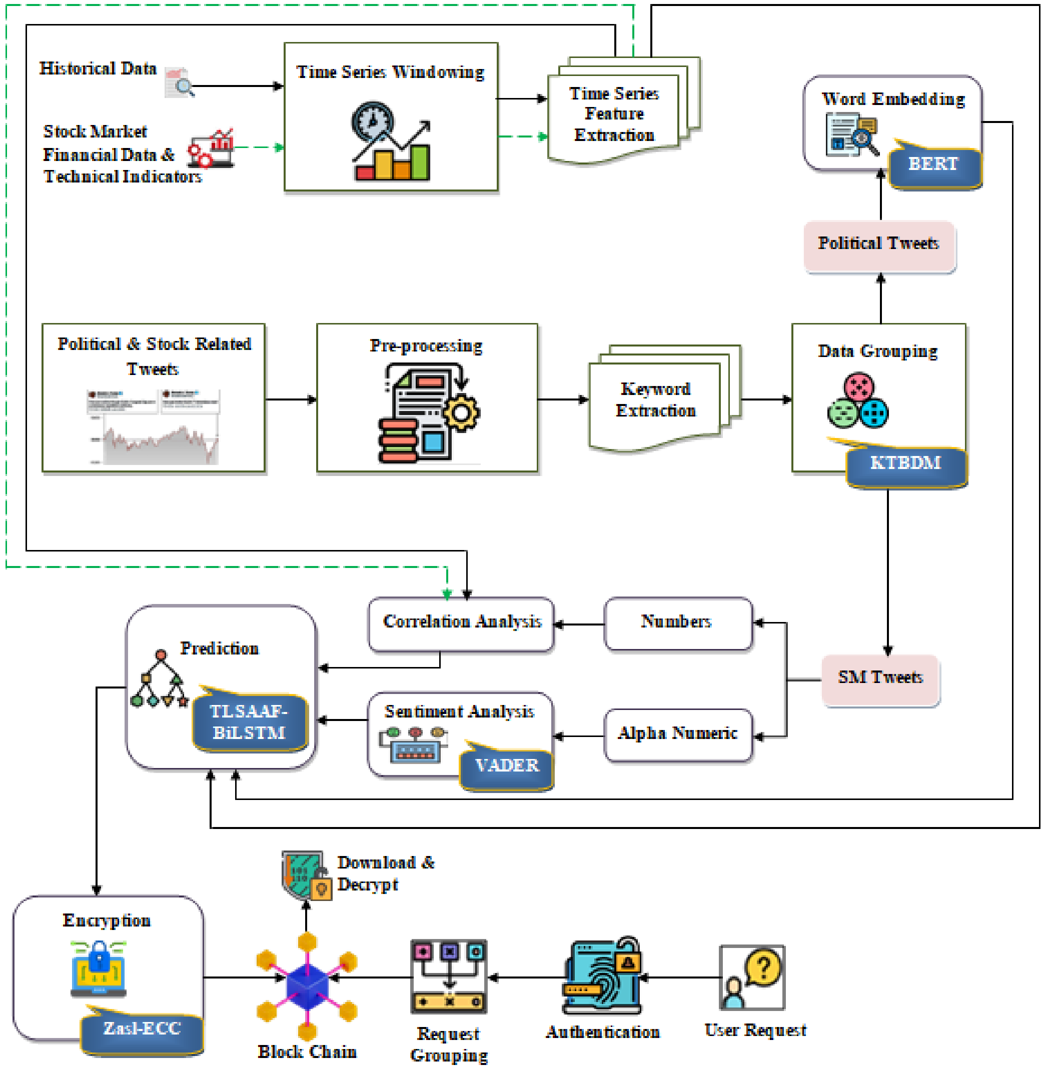


Figure 1: Structural Model of Proposed Methodology

3.1 Time Series Operations

The proposed technique begins by collecting historical data (H) along with real-time SM financial data and technical indicators (D_{tec}^{fin}). Technical indicators like Double Exponential Moving Average, Rate of Change, Average True Range, et cetera are considered. Now, these data undergo time series windowing to split them grounded on quarterly, half-yearly, and annual reports.

Afterward, the features, such as the Number of Peaks, Number of Maximum peaks, Number of Minimum peaks, Median, Mean, et cetera from splitted H and D_{tec}^{fin} are extracted. The feature set $\left(T \in T^{H}, T^{D_{tec}^{fin}}\right)$ is represented as,

$$T^{H} = \{T_1^{H}, T_2^{H}, ..., T_n^{H}\} \tag{1}$$

$$T^{D_{tec}^{fin}} = \{T_1^{D_{tec}^{fin}}, T_2^{D_{tec}^{fin}}, ..., T_n^{D_{tec}^{fin}}\} \tag{2}$$

3.2 Processing of Twitter Data

Here, the recent Twitter data $(\Im)$ comprising SM-centric tweets and political news is collected. By performing symbol removals, missing value replacement, tokenization, stop word removal, and stemming, these data undergo pre-processing. After that, the keywords (κ) from the pre-processed data $(\Im_{pre})$ are extracted.

3.3 Data Grouping

Here, grounded on the keywords, the data $\Im_{pre}$ is grouped using Kernelized Total Bregman Divergence Means Algorithm (KTBDMA). When contrasted with other clustering techniques, tighter clusters are formed with K-means. But, owing to its large amount of computation, the algorithm is too slow to finish the grouping task. Hence, Kernelized Total Bregman Divergence is wielded for increasing the clustering speed.

Primarily, the data $\Im_{pre}$ forms a number of clusters (C), and the cluster centroids (κ) are selected. Now, the sentences (s) in the input data are assigned to the centroids grounded on the similarity betwixt them, which is computed using KTBD $(\wp)$ as,

$$\wp = \min \sum \frac{\varsigma(s_i, s_{i+1}) - 2\varsigma(s_i, \kappa_i) + \varsigma(\kappa_i, \kappa_{i+1})}{\sqrt{1 + 4\varsigma(s_i, s_{i+1})}} \quad (3)$$

Where, ς signifies the kernelized polynomial. Repeat the steps until all the sentences are grouped. Therefore, the final groups are expressed as,

$$C = \{C_{SM}, C_{pol}\} \quad (4)$$

Where, C_{SM} and C_{pol} specify the grouped SM-based tweets and recent Political tweets. Pseudo-code of the KTBDMA is represented further.

Input: Pre-processed data $(\Im_{pre})$
Output: Grouped Tweets (C)

Begin
  Initialize number of clusters, centroids
  For $(i = 1\,to\,n)$ **do**
    Estimate KTBD as,

$$\wp = \min \sum \frac{\varsigma(s_i, s_{i+1}) - 2\varsigma(s_i, \kappa_i) + \varsigma(\kappa_i, \kappa_{i+1})}{\sqrt{1 + 4\varsigma(s_i, s_{i+1})}}$$

    If $sim == Max$
      Assign s_i to cluster
    Else
      Recalculate the similarity for other sentence
    End if
  End for
  Return C
End

3.4 Processing of Grouped Political Data

Here, the grouped political data (C_{pol}) is embedded utilizing BERT, which comprises three layers. Primarily, the tokens for each word in C_{pol} is determined and given to the embedding layer. This layer performs token embedding, segment embedding, and position embedding on the tokens, and is sent to the transformer layer. The encoder encodes the string values, whereas the decoder gives the contextual embedding (ε) as,

$$\varepsilon = \{e_1, e_2, ..., e_q\} \tag{5}$$

Where, e_q symbolizes the contextual embedding of the q^{th} string. The encoded output is given to the output layer, which encompasses a simple classifier model with a fully connected layer along with an activation function. The loss (Ls) in the output is computed as,

$$Ls = \frac{1}{2}\left(T\arg - \sum C_{pol}^{emb}\right) \tag{6}$$

Where, $T\arg$ illustrates the target embedding score, and C_{pol}^{emb} is the output embedding.

3.5 Processing of Grouped SM Data

Here, the alphanumeric characters (α) and the numerical data (β) in C_{SM} are gathered. Thereafter, sentiment analysis is performed on the data α using the Valence Aware Dictionary for Sentiment Reasoning (VADER). VADER makes use of a variety of sentiment lexicon that is often classified as either positive or negative depending on their sentiment score (δ), which is computed as,

$$\delta = \begin{cases} \alpha_+ \rightarrow score = 1, \;\; if \;\; o > 0.001 \\ \alpha_\pm \rightarrow score = 0, \;\; if \;\; o > -0.001, o < 0.001 \\ \alpha_- \rightarrow score = -1, \; if \;\; o < -0.001 \end{cases} \tag{7}$$

Where, $\alpha_+, \alpha_\pm$ and α_- imply the positive, neutral, and negative words, correspondingly, and o is the compound value.

3.6 Correlation Analysis

Here, the time series features $T^{D_{tec}^{fin}}$ and T^H along with the numerical data β are given to Hadoop Distributed File System (HDFS) for performing big data mapping. Hadoop uses a map function and reduces the function. The map function (map) creates several small chunks of data, which are expressed as,

$$M = map(T^{D_{tec}^{fin}}, T^H, \beta) \tag{8}$$

Afterward, the reduce function combines those small chunks (M) and modifies them. After processing, it produces a new set of outputs $(T_{red}^{D_{tec}^{fin}}, T_{red}^H, \beta_{red})$. Now, correlation is calculated between them as,

$$\Psi = \frac{\left(\sum T_{red}^{D_{tec}^{fin}} \cdot T_{red}^H \cdot \beta_{red}\right) - \left(\sum T_{red}^{D_{tec}^{fin}}\right)\left(\sum T_{red}^H\right)\left(\sum \beta_{red}\right)}{\sqrt{\left[\sum\left(T_{red}^{D_{tec}^{fin}}\right)^2 - \left(\sum T_{red}^{D_{tec}^{fin}}\right)^2\right]\left[\sum\left(T_{red}^H\right)^2 - \left(\sum T_{red}^H\right)^2\right]\left[\sum(\beta_{red})^2 - \left(\sum \beta_{red}\right)^2\right]}} \tag{9}$$

Where, Ψ symbolizes the output obtained by correlating the historic and recent data.

3.7 Prediction

Lastly, the correlated output Ψ, sentiment score δ, embedding C_{pol}^{emb}, and time series features $\left(T^{D_{tec}^{fin}}, T^H\right)$ are given to the TLSAAF-BiLSTM for predicting future trends. BiLSTM is capable of utilizing inputs from both sides. However, its activation function suffers from the non-zero mean, negative missing, and unbounded output problems. Thus, to overcome these three challenges, SAAF is wielded. Further, transfer learning is included in the network to enhance the prediction results. Figure 2 displays the architecture of TLSAAF-BiLSTM,

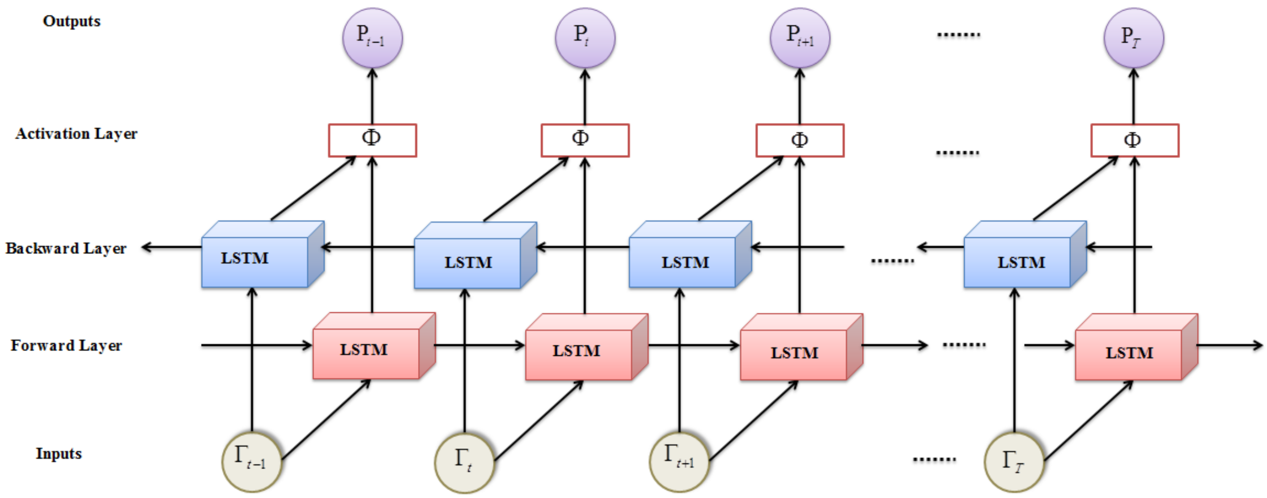


Figure 2: TLSAAF-BiLSTM

Primarily, Transfer Learning (TL) is performed in the input data. In TL, the knowledge about the input data is learned to solve the variations in the recent trends and identify them. Now, this data (Γ) is processed by three gate structures and one cell structure of TLSAAF-BiLSTM. The functions of the gates and the cell state are expressed as,

$$\mathrm{I}_t = \Phi\left(\omega_{\mathrm{I}} * \left[\hbar_{t-1}, \Gamma\right] + B_I\right) \tag{10}$$

$$F_t = \Phi\left(\omega_F * \left[\hbar_{t-1}, \Gamma\right] + B_F\right) \tag{11}$$

$$\Theta_t = \Phi\left(\omega_\Theta * \left[\hbar_{t-1}, \Gamma\right] + B_\Theta\right) \tag{12}$$

$$\eta_t = F_t * \eta_{t-1} + \mathrm{I}_t * \tanh\left(\omega_\eta * \left[\hbar_{t-1}, \Gamma\right] + B_\eta\right) \tag{13}$$

$$\hbar_t = \Theta_t * \tanh\left(\eta_t\right) \tag{14}$$

Where, $\mathrm{I}_t, F_t, \Theta_t$ and η_t symbolize the output of input, Forget, Output Gates, and Cell state at time t, $\hbar_t$ is the hidden layer, ω is the weight, B epitomizes the bias vector, and Φ exemplifies the SAAF.

$$\Phi = \frac{\Gamma}{\frac{\Gamma}{X} + \exp^{\frac{\Gamma}{Y}}} \tag{15}$$

Where, X and Y specify the pair of trainable non-negative parameters. Now, the updated status of the cell in forward $(\hbar_{t-1})$ and backward $(\hbar_{t+1})$ directions are represented as,

$$\hbar_t^{for} = \overrightarrow{\hbar_t}\left(\Gamma, \hbar_{t-1}\right) \tag{16}$$

$$\hbar_t^{bac} = \overrightarrow{\hbar_t}\left(\Gamma, \hbar_{t+1}\right) \tag{17}$$

Lastly, the output predictions (P) are obtained by processing the forward $\left(\hbar_t^{for}\right)$ and backward $\left(\hbar_t^{bac}\right)$ operations.

3.8 Data Security

The predictions (P) are encrypted using the Zasl-ECC algorithm to secure the prediction results from malicious attacks. ECC provides better security for faster computations with less processing power. But, the random generation of public and private keys makes the algorithm susceptible to guessing attacks. Therefore, Zaslavsky Map-based Pseudo Random Number is generated. For the cryptography process, an elliptic curve is utilized, which is defined as,

$$u^3 = a^3 + av + b \tag{18}$$

Where, a, b are integers and u, v signifies the parameters that define the curve. To obtain encryption and decryption, the private (γ_{priv}) and public keys (γ_{pub}) are generated utilizing the Zasl technique as,

$$\gamma_{pub} = \exp^{-rd}\left(\gamma_{priv} + \cos(2\pi\tau)\right) \tag{19}$$

Where, τ exemplifies the curve point and rd depicts the Pseudo Random Number. Now, the prediction data is encrypted. The encrypted data comprises two cipher texts $(\aleph)$, which are defined as,

$$\aleph_1 = (\tau * rd) * \gamma_{pub} \tag{20}$$

$$\aleph_2 = (P + \gamma_{pub} * rd) \tag{21}$$

Hence, the predictions are encrypted and securely stored in the blockchain.

3.9 Data retrieval

Then, for the secure retrieval of future predictions, the users are authenticated when they request for the forecasted results of the stock prices. By verifying the user details, namely valid bank account, email, password, credit/debit card number, and user ID, authentication is performed. If the user is authenticated, they are grouped grounded on the company name they intend to forecast the stock value. Afterward, for the grouped users, the company name is checked in the blockchain and the forecasted stock value is displayed after the internal decryption. In this way, the predicted stock value can be securely accessed without any tampering.

4. RESULTS AND DISCUSSION

Experiments conducted in the working platform of PYTHON are analyzed for demonstrating the advantage of the proposed secure SM prediction model. The proposed technique is trained by integrating different data sources that are publicly available.

4.1 Performance Analysis

Here, the outcomes of the proposed algorithms are analyzed by comparing them with prevailing models.

Table 1: Security Level Analysis

Methods	**Security Level (%)**
Proposed Zasl-ECC	97
ECC	95
Kyber	92

RSA	88
ElGamal	86

Table 1 compares the performance of the proposed Zasl-ECC with the existing ECC, Kyber, Rivest-Shamir-Adleman (RSA), and ElGamal techniques. The proposed method outperformed the existing methods owing to the reason of reducing guessing attacks with the key generation mechanism using the Zasl technique.

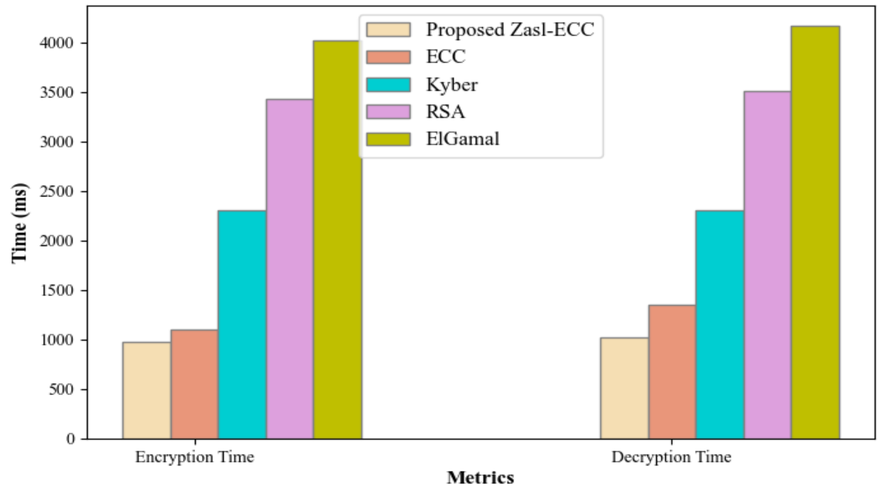


Figure 3: Performance Comparison of Encryption Time (ET) and Decryption Time (DT)

As per Figure 3, ET and DT for the proposed method are less by 127ms and 330ms when contrasted with the existing mechanisms. Therefore, the Zasl technique reduced the delay in offering different encrypted and decrypted messages.

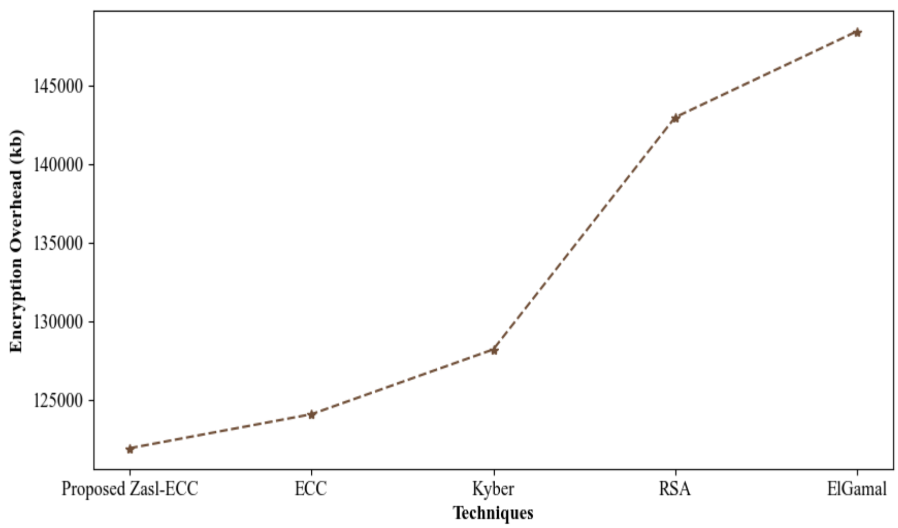


Figure 4: Comparison of Encryption Overhead

Figure 4 displays that the proposed one is stronger with minimum encryption overhead (121944kb) when contrasted with the existing mechanisms. This indicates that the potential replacement of the key generation mechanism provided more flexibility in attaining a reduction in overhead.

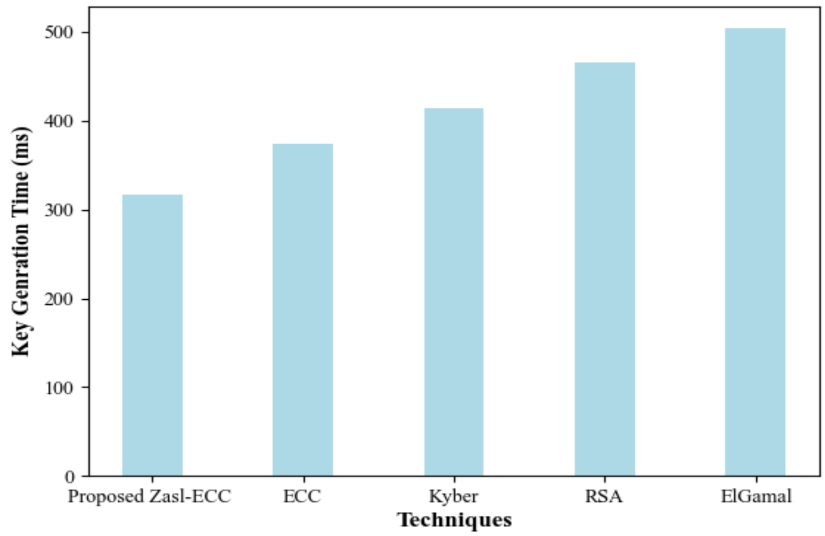


Figure 5: Key Generation Time for Different Models

Figure 5 exhibits that for the key generation, the proposed technique takes just 317ms, which is much lesser than the existing methods. This minimum time consumption ensures that the Zasl-based key generation takes less number of rounds and provides stronger encryption.

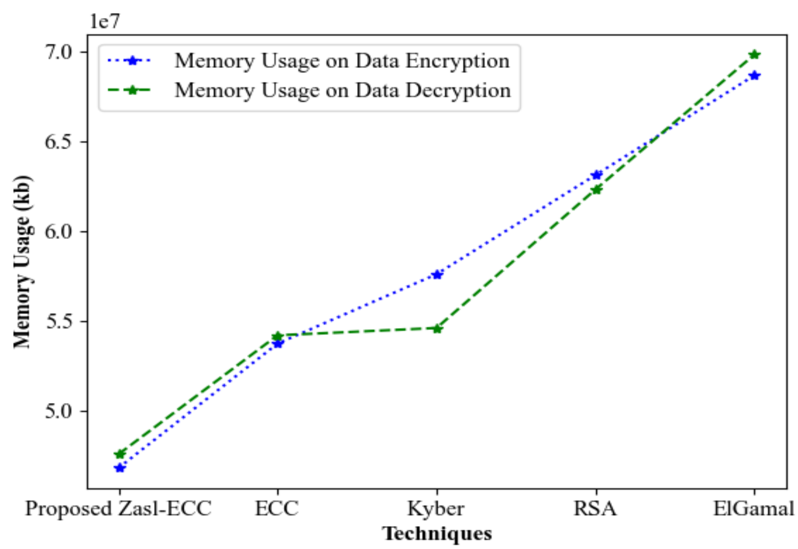
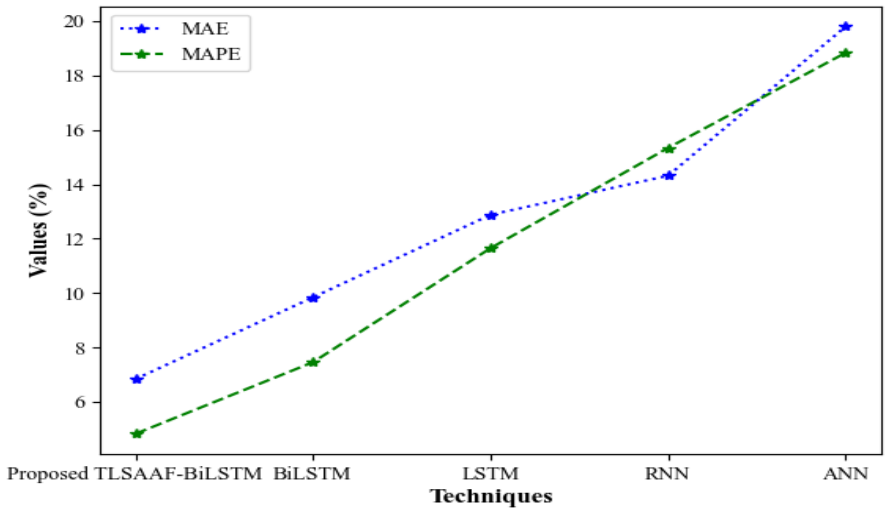
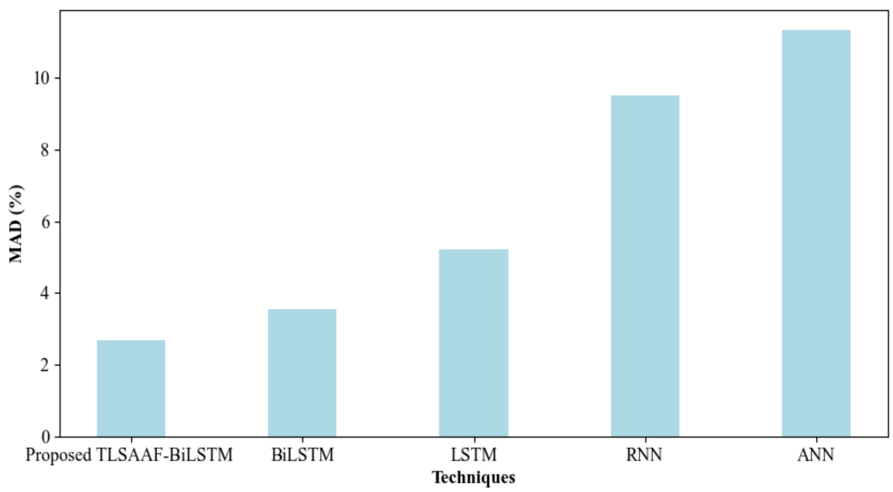


Figure 6: Memory Usage on Encryption and Decryption

Figure 6 elucidates the memory consumption by different encryption algorithms. In comparison, the proposed technique gives a better performance, where its memory consumption on encryption and decryption is lesser (6903899kb and 6604510kb) than the prevailing ECC.



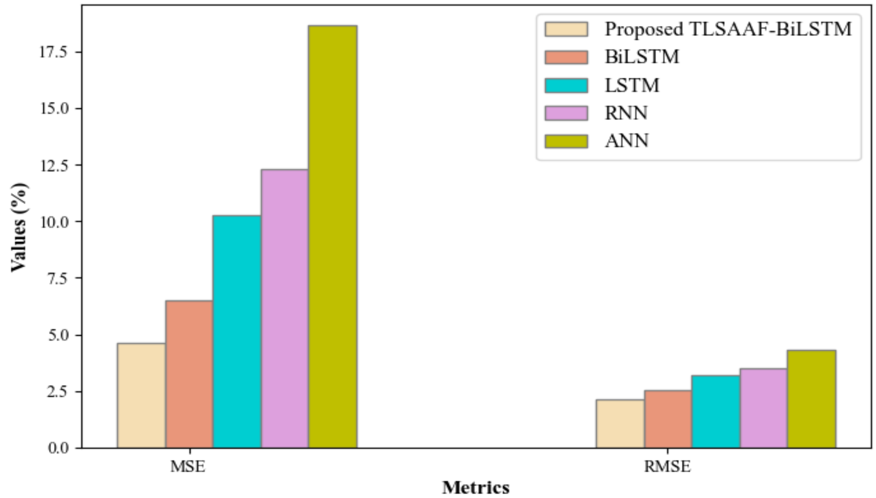
(a)



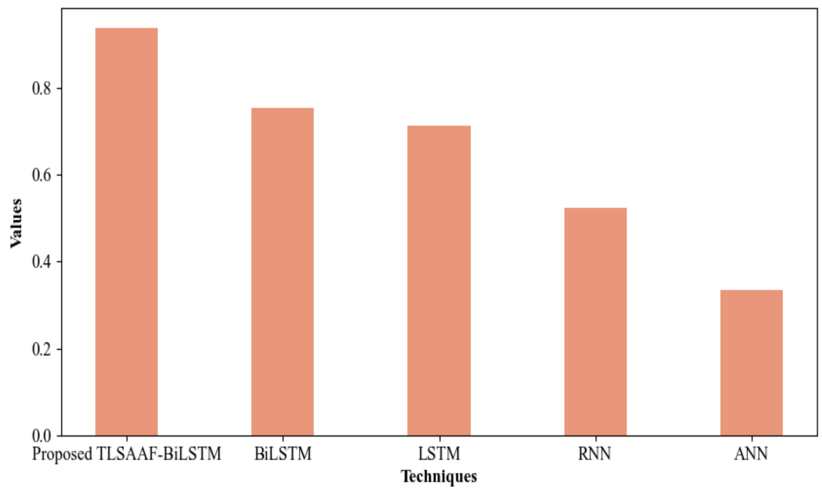
(b)

Figure 7: Comparison of (a) MAE, MAPE, and (b) MAD Results

In Figure 7, the prediction outcomes of the proposed SAAF-BiLSTM and existing BiLSTM, Recurrent Neural Network (RNN), LSTM, and Artificial Neural Network (ANN) are tested on (a) Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) and (b) Mean Absolute Deviation (MAD). In comparison, the proposed model attains a minimum error rate of 6.8356% (MAE), 4.8213% (MAPE), and 2.6837% (MAD) than the existing approaches. Therefore, the inclusion of SAAF has strong potential in improving the performance of the proposed model by providing less variability in prediction results.



(a)



(b)

Figure 8: Error Analysis

The Mean Squared Error (MSE), Root MSE (RMSE), and R-Squared error plots in Figure 8 (a) and (b) specify that with a minimum value of MSE (4.6385%), RMSE (2.1537%), and maximum R-Squared (0.9357), the proposed model forecasts more accurate than the existing techniques. Thus, the adaption of the SAAF technique well diminished the deficiencies in its learning of the proposed TLSAAF-BiLSTM.

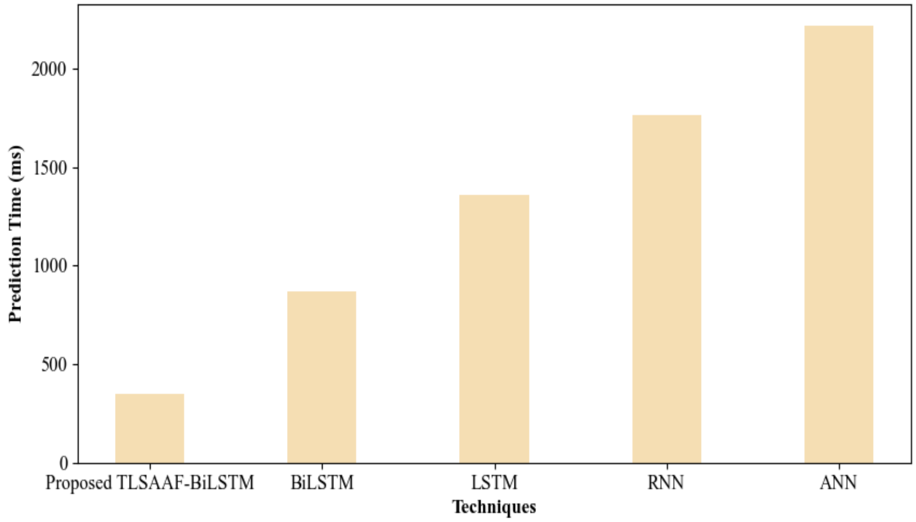


Figure 9: Performance Analysis Regarding Prediction Time

As per Figure 9, the prediction time of the proposed system is 348 ms, which is much lower than the existing systems. Therefore, the integration of SAAF enhanced the learning speed of the prediction through the adjustment of training parameters.

Table 2: Clustering Time Analysis

Methods/Metrics	**Time (ms)**
Proposed KTBDMA	576
KMA	767
K-Medoid	846
FCM	1028
CLARA	1145

Table 2 displays the performance evaluation of proposed and existing KMA, K-Medoid, Fuzzy C-Means (FCM), and Clustering LARge Applications (CLARA) approaches grounded on Clustering time. The proposed method handles grouping much faster in 576ms, while the existing systems are time-consuming. This proves that the KTBD technique aids in attaining rapid execution of grouping.

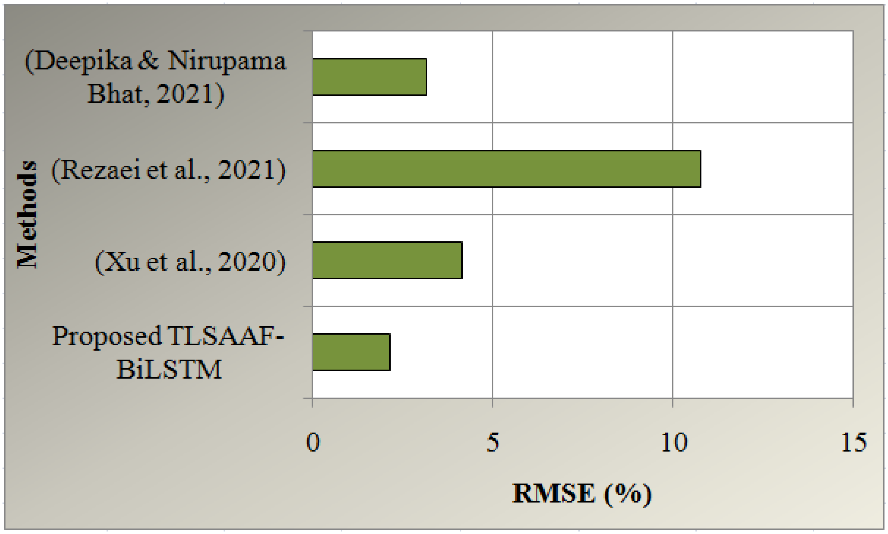


Figure 10: Comparative Analysis

Figure 10 compares the TLSAAF-BiLSTM's performance with the techniques developed by (Xu et al., 2020), (Rezaei et al., 2021), and (Deepika & Nirupama Bhat, 2021) in section 2. From Figure 10, it can be said that the proposed system's performance concerning RMSE is higher than the existing methods. This concludes that even though the existing models were accurate in their prediction, the inclusion of several influential factors together with periodic company reports and stock tend correlations enhanced the proposed technique's superiority.

5. CONCLUSION

A security framework for SM prediction using TLSAAF-BiLSTM and Zasl-ECC methods is proposed in this paper. The proposed model focused on enhancing prediction accuracy by utilizing a novel combination of security and sentiment analysis-based forecasting. Further, the performance of Zasl-ECC and TLSAAF-BiLSTM was weighted against several prevailing techniques for testing the proposed mechanism's efficacy. As per the experimental outcomes, the security level attained by the Zasl-ECC technique is higher than the existing approaches. Moreover, the proposed TLSAAF-BiLSTM method enhances the MAE, MAD, MAPE, MSE, RMSE, and R-Squared; also, it is good enough to apply in different forecasting applications. The work can be extended in the future to concentrate on more significant factors affecting SM and thereby conducting risk level assessments.

REFERENCES

Aasi, B., Imtiaz, S. A., Qadeer, H. A., Singarajah, M., & Kashef, R. (2021). Stock price prediction using a multivariate multistep LSTM: A sentiment and public engagement analysis model. *2021 IEEE International IOT, Electronics and Mechatronics Conference, Proceedings*, *April*. https://doi.org/10.1109/IEMTRONICS52119.2021.9422526

Achyutha, P. N., Chaudhury, S., Bose, S. C., Kler, R., Surve, J., & Kaliyaperumal, K. (2022). User Classification and Stock Market-Based Recommendation Engine Based on Machine Learning and Twitter Analysis. *Mathematical Problems in Engineering*, *2022*. 1-9. https://doi.org/10.1155/2022/4644855

Awan, M. J., Rahim, M. S. M., Nobanee, H., Munawar, A., Yasin, A., & Zain, A. M. (2021). Social Media and Stock Market Prediction: A Big Data Approach. *Computers, Materials and Continua*, *67*(2), 2569–2583. https://doi.org/10.32604/cmc.2021.014253

Deepika, N., & Nirupama Bhat, M. (2021). An Efficient Stock Market Prediction Method Based on Kalman Filter. *Journal of The Institution of Engineers (India): Series B*, *102*(4), 629–644. https://doi.org/10.1007/s40031-021-00583-9

Gandhmal, D. P., & Kumar, K. (2019). Systematic analysis and review of stock market prediction techniques. *Computer Science Review*, *34*, 1-13. https://doi.org/10.1016/j.cosrev.2019.08.001

Gupta, Y. K., & Sharma, N. (2020). Propositional aspect between apache spark and hadoop map-reduce for stock market data. *Proceedings of the 3rd International Conference on Intelligent Sustainable Systems*, 479–483. https://doi.org/10.1109/ICISS49785.2020.9315977

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

Jaggi, M., Mandal, P., Narang, S., Naseem, U., & Khushi, M. (2021). Text mining of stocktwits data for predicting stock prices. *Applied System Innovation*, *4*(1), 1–22. https://doi.org/10.3390/asi4010013

Jiang, W. (2021). Applications of deep learning in stock market prediction: Recent progress. *Expert Systems with Applications*, *184*, 1–97. https://doi.org/10.1016/j.eswa.2021.115537

Khan, W., Ghazanfar, M. A., Azam, M. A., Karami, A., Alyoubi, K. H., & Alfakeeh, A. S. (2022). Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Computing*, *13*(7), 3433–3456. https://doi.org/10.1007/s12652-020-01839-w

Lin, Y.-L., Lai, C.-J., & Pai, P.-F. (2022). Using Deep Learning Techniques in Forecasting Stock Markets by Hybrid Data with Multilingual Sentiment Analysis. *Electronics*, *11*(21), 1-19. https://doi.org/10.3390/electronics11213513

Moghar, A., & Hamiche, M. (2020). Stock Market Prediction Using LSTM Recurrent Neural Network. *Procedia Computer Science*, *170*, 1168–1173. https://doi.org/10.1016/j.procs.2020.03.049

Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., Salwana, E., & Shahab, S. (2020). Deep learning for stock market prediction. *Entropy*, *22*(8).1-23 https://doi.org/10.3390/E22080840

Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2021). A novel multi-source information-fusion predictive framework based on deep neural networks for accuracy enhancement in stock market prediction. *Journal of Big Data*, *8*(1). 1-28. https://doi.org/10.1186/s40537-020-00400-y

Rezaei, H., Faaljou, H., & Mansourfar, G. (2021). Stock price prediction using deep learning and frequency decomposition. *Expert Systems with Applications*, *169*, 1-29. https://doi.org/10.1016/j.eswa.2020.114332

Rouf, N., Malik, M. B., Arif, T., Sharma, S., Singh, S., Aich, S., & Kim, H. C. (2021). Stock market prediction using machine learning techniques: A decade survey on methodologies, recent developments, and future directions. *Electronics (Switzerland)*, *10*(21), 1-25. https://doi.org/10.3390/electronics10212717

Shah, A., Gor, M., Sagar, M., & Shah, M. (2022). A stock market trading framework based on deep learning architectures. *Multimedia Tools and Applications*, *81*(10), 14153–14171. https://doi.org/10.1007/s11042-022-12328-x

Thakkar, A., & Chaudhari, K. (2021). Fusion in stock market prediction : A decade survey on the necessity , recent developments , and potential future directions. *Information Fusion Journal*, 95–107.

Wu, J. M.-T., Sun, L., Srivastava, G., Diaz, V. G., & Lin, J. C.-W. (2022). A Stock Trading Expert System Established by the CNN-GA-Based Collaborative System. *International Journal of Data Warehousing and Mining*, *18*(1), 1–19. https://doi.org/10.4018/ijdwm.309957

Xu, H., Chai, L., Luo, Z., & Li, S. (2020). Stock movement predictive network via incorporative attention mechanisms based on tweet and historical prices. *Neurocomputing*, *418*, 326–339. https://doi.org/10.1016/j.neucom.2020.07.108