Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1, No.4 : 2024
ISSN : **1906-9685**

ENSEMBLE RANDOM FOREST WITH LSTM NEURAL NETWORK (ERF-LSTM) METHOD FOR CLASSIFICATION OF TWITTER SENTIMENT ANALYSIS

K. Brindha, Research Scholar, NGM College, Pollachi, Tamilnadu, India
Dr. E. Ramadevi, Associate Professor, Dept of Computer Science, NGM College, Pollachi, Tamilnadu, India

Abstract - This study presents a novel approach for stock classification by harnessing the power of Yahoo Finance stocks data and Twitter data. We employ a fusion of the Random Forest Classifier and LSTM Neural Network to create an ensemble model capable of enhancing stock classification accuracy. Structured financial data from Yahoo Finance provides a solid foundation, while Twitter data contributes real-time sentiment insights. Weighted averaging optimizes the ensemble's performance by assigning different weights based on historical model accuracy. This approach empowers traders and investors with more reliable and timely stock classifications, facilitating informed decision-making in the ever-evolving world of stock trading.
Keywords: Stock Classification, Yahoo Finance, Twitter Data, Random Forest Classifier, LSTM Neural Network, Ensemble Model;

1. Introduction

In the age of information, financial markets are not only driven by traditional data sources but are also significantly influenced by the vast and dynamic landscape of social media. Among the plethora of social media platforms, Twitter has emerged as a prominent source of real-time financial information and sentiment. Traders, investors, and financial analysts actively engage with Twitter to gain insights into market trends, news, and public sentiment regarding various stocks. This intersection of finance and social media has paved the way for innovative research endeavors, one of which involves harnessing Yahoo Finance stocks' Twitter data for classification and analysis. Yahoo Finance, a renowned platform for real-time financial data and news, provides a rich source of structured data related to stocks, including historical prices, financial statements, and market news. Complementing this, Twitter serves as a treasure trove of unstructured data, filled with discussions, opinions, and news regarding stocks. Integrating these two data sources allows for a holistic understanding of the financial market's pulse, and this fusion has garnered significant attention from researchers, data scientists, and financial professionals alike.

This research embarks on a journey to explore the immense potential that Yahoo Finance stocks' Twitter data offers for classification tasks. The central objective is to leverage this data fusion to enhance research understanding of stock market dynamics and investor sentiment. To achieve this, the study employs a multifaceted approach, encompassing heuristic-based and supervised classifiers, as well as cutting-edge deep learning techniques such as recurrent neural networks (RNNs) specifically Long Short-Term Memory (LSTM) networks. The primary motivation behind this research is the recognition of the inherent complexity and nuances present in Twitter data. Unlike traditional financial data sources, Twitter is characterized by its brevity, informality, and the use of hashtag, mentions, and emojis, which can make sentiment analysis and classification a challenging endeavor. Moreover, Twitter is prone to the emergence of homonyms and collisions, where the same symbol or acronym is used to refer to both company tickers and cryptocurrencies. This ambiguity can lead to significant data distortion and misclassification if not properly addressed.

Data Fusion for Enhanced Insights

The cornerstone of approach is the fusion of structured financial data from Yahoo Finance with unstructured Twitter data. This synergy enables us to capture not only the numerical aspects of stock performance but also the sentiment and opinions expressed by the Twitter community. By merging these diverse data sources, we create a holistic view of stock dynamics, allowing for more comprehensive and informed classification.

Harnessing the Power of Random Forest Classifier

The Random Forest Classifier is a versatile machine learning algorithm known for its accuracy in structured data analysis. In this approach, we leverage this strength by training the classifier with historical stock data as input and generating labels that represent stock movement categories, such as "buy," "sell," or "hold." Cross-validation techniques and performance metrics help evaluate the classifier's accuracy and predictive power, ensuring it contributes significantly to ensemble model.

Classification using Machine Learning

Machine learning methods can develop sentiment classifiers by creating feature vectors through a series of essential steps. This process involves collecting and cleaning data, extracting relevant features, selecting the most pertinent ones, training the classifier using the training dataset, and subsequently assessing its performance. To achieve this, the dataset must be split into two subsets: the training set, which teaches the classifier to recognize text features, and the test set, which evaluates the classifier's effectiveness in sentiment classification.

Classifiers, including the Naïve Bayes, Support Vector Machine, Logistic, and Random Forest classifiers, play a vital role in categorizing text into predefined classes. Machine learning is a widely adopted approach for text classification, and it's essential to note that classifier performance can vary significantly across different types of text. Therefore, it's advisable to train separate feature vectors for each text type to enhance model robustness. One effective technique for handling noise data's impact on classification is employing a two-stage Support Vector Machine classifier. In the subsequent steps, tweets data needs to be vectorizer, and labeled tweets should be split into a training set (80%) and a test set (20%) for training various classification models. Sentiment classification involves predicting tweet sentiments as positive, negative, or neutral based on tweet feature representations. Supervised machine learning methods, like Random Forest, excel at classifying and predicting unlabeled text by leveraging a substantial number of sentiment-labeled tweets.

2. Literature Survey

2.1 Support Vector Machine (SVM)

S. Urolagin (2017) et.al proposed Text Mining of Tweet for Sentiment Classification and Association with Stock Prices. Companies are using sentiment analysis with Naïve Bayes and SVM classifiers on tweets. N-gram-based vectors from tweet keywords are used. Positive, negative, neutral tweets, and total tweet count are features in an SVM model for stock market prediction. Social media platforms like Facebook, Twitter, YouTube, and MySpace provide customer feedback and insights into product improvements.

2.2 Unigram Term Frequencies - Inverse Document Term Frequency (TF-IDF)

M. Qasem (2015) et.al proposed Twitter sentiment classification using machine learning techniques for stock markets. This study assesses the effectiveness of sentiment classification in predicting stock-related tweets using logistic regression and neural networks. It evaluates a dataset of 42,000 tweets related to tech stocks (Twitter, Google, Facebook, and Tesla). Both classifiers achieve 58% accuracy, with Unigram TF-IDF weighting outperforming Bigram TF. Twitter's open nature, accessible API, real-time data, and diverse interactions make it a valuable source for predicting stock market trends.

2.3 Multi-Dimensional and Multi-Level Modeling

B. Wang (2022) et.al proposed A Sentiment Classification Method of Web Social Media Based on Multidimensional and Multilevel Modeling. A novel research approach tackles missing contextual semantics in web social media text sentiment classification. Existing methods mainly consider document-level sentiment features, overlooking sentence-to-sentence and word-to-word interactions.

This study introduces a multi-level sentiment modeling technique, capturing sentiment nuances across words, sentences, and documents. Additionally, it includes emoticons and punctuation symbols for richer sentiment analysis, addressing the challenge of missing text context semantics in web social media.

2.4 Structure-Based Multiclass Classification

W. Weng (2016) et.al proposed a multiclass classification model for stock news based on structured data. This research introduces a data-driven stock news classification model that evaluates news correlations through structured data. It employs parameter fitting and softmax regression to produce probability sequences. A comparison with a standard algorithm highlights its improved effectiveness and accuracy. The study also proposes a custom feature selection method, fusing structured stock data with softmax regression for a distinctive multiclass classification model. Empirical results validate its enhanced accuracy in stock news classification, overcoming shortcomings of conventional methods and pioneering a structure-based approach.

2.5 Heuristic-Based and Supervised Classifiers

A. Fernández Vilas (2020) et.al proposed The Eruption of Crypto currencies into Twitter Cash tags: A Classifying Solution. This research presents heuristic and supervised classifiers for Twitter data analysis, highlighting their strengths and weaknesses, especially in adapting to changing Twitter trends. It identifies data distortion issues when overlapping hashtags are present. Independent models, disconnected from training data, demonstrate cross-market potential. The study also explores recurrent neural networks (LSTM) trained on the top 10,000 terms to establish term importance and relationships.

3. Proposed Methodology

In this context, propose a classification approach that combines the predictive strengths of the Random Forest Classifier and the Long Short-Term Memory (LSTM) Neural Network. This approach capitalizes on the structured financial data from Yahoo Finance and the unstructured Twitter data to create a robust and adaptable stock classification model.

Random Forest Classifier

Training a Random Forest Classifier using historical stock data as input and the generated labels as target variables is a crucial step in building a predictive model for stock classification. This ensemble learning algorithm constructs multiple decision trees to collectively make accurate predictions. Each tree examines subsets of the historical data and votes on the classification, reducing the risk of overfitting. By analyzing past stock price movements, trading volumes, and relevant technical indicators, the Random Forest Classifier learns to recognize patterns and trends that can aid in classifying future stock movements as "buy," "sell," or "hold." Its ability to handle complex data and mitigate overfitting makes it a powerful tool in stock market prediction and trading strategy development.

LSTM Neural Network

To design LSTM-based neural network architecture for processing sequential Twitter data, the key focus is on harnessing the temporal dependencies within the text. This architecture comprises an input layer, followed by one or more LSTM layers that capture sequential information. Preprocessing techniques like tokenization and embedding layers are used to convert text data into numerical format.

The LSTM model is trained with Twitter data as input and stock price movements as target variables. This training process involves optimizing the model's weights through backpropagation, minimizing prediction errors, and maximizing accuracy in forecasting stock price movements. Fine-tuning hyperparameters and optimizing the network's architecture is crucial. Parameters such as the number of LSTM layers, hidden units, batch size, and learning rate are adjusted to enhance the model's predictive power. Techniques like dropout and regularization are employed to prevent overfitting. By iteratively optimizing the architecture and fine-tuning hyperparameters, the LSTM-based neural network becomes adept at capturing nuanced patterns within Twitter data, ultimately improving its ability to predict stock price movements accurately.

3.1 Ensemble Approach

To create an ensemble model, we harness the predictive power of both the Random Forest Classifier and the LSTM Neural Network. This ensemble approach leverages the strengths of each model, aiming to enhance overall performance. To achieve this, we implement a weighted averaging scheme, assigning different weights to the predictions generated by each model. These weights are determined based on the historical performance and accuracy of each model. By assigning greater weight to the model with a stronger track record of accurate predictions, we can optimize the ensemble's predictive capabilities. This weighted averaging technique ensures that the ensemble model benefits from the strengths of both models, resulting in more reliable and accurate stock classification decisions.

Ensemble Model Creation

The core of this approach involves the creation of an ensemble model that amalgamates the strengths of the Random Forest Classifier and the LSTM Neural Network. This proposed methodology combines the strengths of Random Forest Classifier and LSTM Neural Network to create a robust and adaptive system for stock classification using both Yahoo Finance and Twitter data. By fusing structured financial data with unstructured social media data, the model aims to provide more accurate and timely insights for informed trading decisions.

Random Forest Classifier

RFC is an ensemble learning method based on decision tree classifiers. The RFC algorithm constructs multiple decision trees during training and combines their predictions to make more accurate classifications. The equation below represents the RFC prediction for a single decision tree:

$$RFC(x) = \sum_{i=1}^{N} T_i(x)$$

Where:

- $RFC(x)$ is the RFC's prediction for input x.
- $T_i(x)$ is the prediction of the i^{th} decision tree.

The final prediction of the RFC ensemble is obtained through a majority vote or weighted averaging of individual tree predictions.

LSTM Neural Network

LSTM is a type of recurrent neural network (RNN) that is well-suited for sequence data, making it ideal for processing Twitter text data. The key equations for LSTM include:

Forget Gate:

The forget gate determines what information from the previous cell state should be discarded and what new information should be stored.

$$f_t = \sigma(W_f . [h_{t-1}, x_t] + b_f)$$

Where:

- $f_t$is the forget gate at time t.
- σ is the sigmoid activation function.
- W_f is the weight matrix for the forget gate.
- h_{t-1} is the previous hidden state.
- x_t is the input at time t.
- b_f is the bias for the forget gate.

Input Gate:

The input gate determines what new information should be stored in the cell state.

$$i_t = \sigma(W_i . [h_{t-1}, x_t] + b_i)$$
$$\tilde{C}_t = tanh\,(W_C . [h_{t-1}, x_t] + b_C)$$

Where:

- i_t is the input gate at time t.
- $\tilde{C}_t$ is the new candidate values for the cell state.
- $tanh$ is the hyperbolic tangent activation function.
- W_i is the weight matrix for the input gate.
- W_C is the weight matrix for the candidate values.
- b_i And b_C are the biases for the input gate and candidate values.

Cell State Update:

The cell state is updated by combining the previous cell state; forget gate, and new candidate values.

Output Gate:

The output gate determines what information from the cell state should be passed to the hidden state and the final prediction.

$$o_t = \sigma(W_o . [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t . \tanh(C_t)$$

Where:

- o_t is the output gate at time t.
- $h_t$is the hidden state at time t.
- W_o is the weight matrix for the output gate.
- b_o is the bias for the output gate.

Creating a Robust Stock Classification Model

- **Model Building:** Random Forest Classifier and LSTM Neural Network. The core of the hybrid approach lies in model building. A Random Forest Classifier, known for its prowess in handling structured data, is trained using historical stock data. Simultaneously, an LSTM Neural Network, designed for sequential data processing, is constructed to analyze Twitter data. These models are trained and fine-tuned to optimize their predictive capabilities.
- **Weighted Averaging for Enhanced Predictions:** The fusion of the Random Forest Classifier and LSTM Neural Network is achieved through weighted averaging. Different weights are assigned to their predictions based on historical performance. This optimization ensures that the ensemble model benefits from the strengths of each model, resulting in more reliable and accurate stock classification decisions.

Creating an algorithm using a Random Forest Classifier and an LSTM Neural Network for enhanced Yahoo Finance stock Twitter data classification involves several key stages. Here's a detailed algorithm to achieve this:

Combined RF and LSTM Ensemble for Twitter Sentiment Classification II n this algorithm, you load pre-trained RF and LSTM models, pre-process new Twitter data, make predictions using both models, combine the predictions, and finally output the sentiment label. The way you combine the predictions (e.g., weighted averaging, stacking) can be customized based on your specific needs and the performance of each model on your validation data.

Algorithm: Combined RF and LSTM Ensemble for Twitter Sentiment Classification

Input:

- Labeled Twitter data with sentiment labels (positive, negative, neutral)
- Load feature selected data
- RF Model (trained and saved)
- LSTM Model (trained and saved)
- Weights for RF and LSTM models in the ensemble (optional)

Output:

- Sentiment labels for new Twitter data

Step 1: Load the RF and LSTM models

Step 2: Predict Sentiment using the Random Forest (RF) model

Step 3: Apply the RF model to the feature selected data to obtain sentiment probabilities.

Step 4: Predict Sentiment using the LSTM model

Step 5: Apply the LSTM model to the feature selected data to obtain sentiment probabilities.

Step 6: Combine the predictions from RF and LSTM models

Step 7: You can combine the predicted probabilities in various ways, such as weighted averaging or stacking.

Step 8: Weighted Averaging: Calculate a weighted average of the predicted probabilities from both models using predefined weights for RF and LSTM.

Step 9: Stacking: Train a meta-model (e.g., logistic regression) on the predicted probabilities from both models to make a final ensemble prediction.

Step 10: Based on the combined prediction, determine the sentiment label for the new Twitter data (e.g., positive, negative, or neutral).
Step 11: Metrics including accuracy, precision, recall, and F1-score can be used to evaluate the ensemble model's performance.

This algorithm outlines the entire process of combining a Random Forest Classifier and an LSTM Neural Network to enhance Yahoo Finance stock Twitter data classification. It allows traders and investors to make more informed decisions by leveraging the predictive power of both models and adapting to evolving market conditions.

4. Experiment Results

4.1 Accuracy

Accuracy is the degree of closeness between a measurement and its true value. The formula for accuracy is:

$$\boldsymbol{Accuracy} = \frac{(\boldsymbol{true\ value} - \boldsymbol{measured\ value})}{\boldsymbol{true\ value}} * \mathbf{100}$$

Dataset	SVM	Naive Bayes	Proposed ERF-LSTM
100	66	74	91
200	73	77	96
300	78	66	88
400	84	70	100
500	87	63	98

Table 1.Comparison Table of Accuracy

The Comparison table 1 of Accuracy demonstrates the different values of existing SVM, Naive Bayes and Proposed ERF-LSTM. While comparing the Existing algorithm and Proposed ERF-LSTM, provides the better results. The existing algorithm values start from 66 to 87, 63 to 77 and Proposed ERF-LSTM values starts from 88 to 100. The proposed method provides the great results.

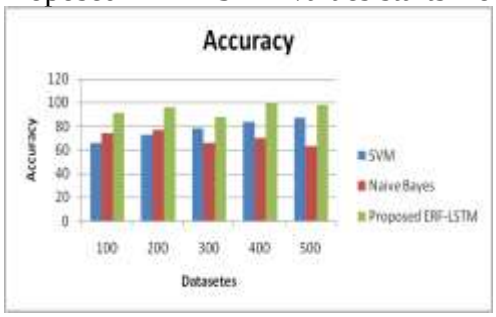

Figure 1.Comparison Chart of Accuracy

The Figure 1 Shows the comparison chart of Accuracy demonstrates the existing SVM, NAIVE BAYES and Proposed ERF-LSTM. X axis denote the Dataset and y axis denotes the Accuracy. The Proposed ERF-LSTM values are better than the existing algorithm. The existing algorithm values start from 66 to 87, 63 to 77 and Proposed ERF-LSTM values starts from 88 to 100. The proposed method provides the great results.

4.2 Precision

Precision is a measure of how well a model can predict a value based on a given input.

$$\boldsymbol{Precision} = \frac{\boldsymbol{true\ positive}}{(\boldsymbol{true\ positive} + \boldsymbol{false\ positive})}$$

Dataset	SVM	Naive Bayes	Proposed ERF-LSTM
100	88.12	82.37	98.67
200	81.69	88.82	97.26

300	75.62	85.54	99.21
400	74.55	83.63	96.58
500	76.94	80.72	91.87

Table 2.Comparison Table of Precision

The Comparison table 2 of Precision demonstrates the different values of existing SVM, Naive Bayes and Proposed ERF-LSTM. While comparing the Existing algorithm and Proposed ERF-LSTM, provides the better results. The existing algorithm values start from 74.55 to 88.12, 80.72 to 88.82 and Proposed ERF-LSTM values starts from 91.87 to 99.21. The proposed method provides the great results.

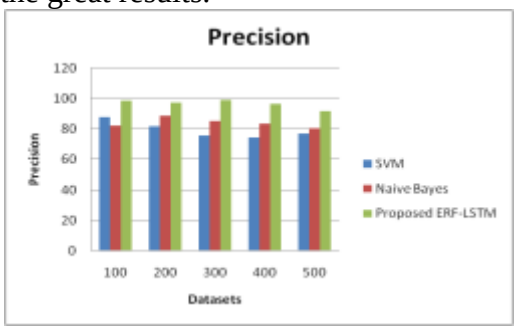

Figure 2.Comparison Chart of Precision

The Figure 2 Shows the comparison chart of Precision demonstrates the existing SVM, Naive Bayes and Proposed ERF-LSTM. X axis denote the Dataset and y axis denotes the Precision ratio. The Proposed ERF-LSTM values are better than the existing algorithm. The existing algorithm values start from 74.55 to 88.12, 80.72 to 88.82 and Proposed ERF-LSTM values starts from 91.87 to 99.21. The proposed method provides the great results.

4.3 Recall

Recall is a measure of a model's ability to correctly identify positive examples from the test set:

$$\boldsymbol{Recall} = \frac{\boldsymbol{True\ Positives}}{(\boldsymbol{True\ Positives} + \boldsymbol{False\ Negatives})}$$

Dataset	SVM	Naive Bayes	Proposed ERF-LSTM
100	0.74	0.82	0.87
200	0.76	0.78	0.94
300	0.83	0.67	0.95
400	0.85	0.77	0.93
500	0.88	0.72	0.97

Table 3.Comparison Table of Recall

The Comparison table 3 of Recall demonstrates the different values of existing SVM, Naive Bayes and Proposed ERF-LSTM. While comparing the Existing algorithm and Proposed ERF-LSTM, provides the better results. The existing algorithm values start from 0.74 to 0.88, 0.67 to 0.82 and Proposed ERF-LSTM values starts from 0.87 to 0.97. The proposed method provides the great results.

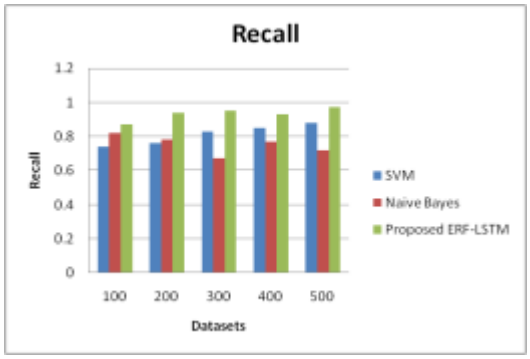

Figure 3.Comparison Chart of Recall

The Figure 3 Shows the comparison chart of Recall demonstrates the existing SVM, Naive Bayes and Proposed ERF-LSTM. X axis denote the Dataset and y axis denotes the Recall ratio. The Proposed ERF-LSTM values are better than the existing algorithm. The existing algorithm values start from 0.74 to 0.88, 0.67 to 0.82 and Proposed ERF-LSTM values starts from 0.87 to 0.97. The proposed method provides the great results.

4.4 F -Measure

F1-measure is a test's accuracy that combines precision and recall. It is calculated by taking the harmonic mean of precision and recall.

$$\boldsymbol{F1 - Measure = \frac{(2 * Precision * Recall)}{(Precision + Recall)}}$$

Dataset	SVM	Naive Bayes	Proposed ERF-LSTM
100	0.85	0.78	0.97
200	0.87	0.76	0.98
300	0.80	0.69	0.96
400	0.79	0.67	0.95
500	0.77	0.68	0.93

Table 4.Comparison Table of F -Measure

The Comparison table 4 of F -Measure Values explains the different values of existing SVM, Naive Bayes and Proposed ERF-LSTM. While comparing the Existing algorithm and Proposed ERF-LSTM, provides the better results. The existing algorithm values start from 0.77 to 0.87, 0.67 to 0.78 and Proposed ERF-LSTM values starts from 0.93 to 0.98. The proposed method provides the great results.

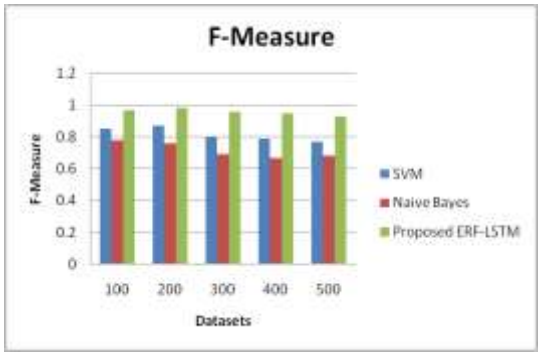

Figure 4.Comparison chart of F -Measure

The Figure 4 Shows the comparison chart of F -Measure demonstrates the existing SVM, Naive Bayes and Proposed ERF-LSTM. X axis denote the Dataset and y axis denotes the F -Measure ratio. The Proposed ERF-LSTM values are better than the existing algorithm. The existing algorithm values start from 0.77 to 0.87, 0.67 to 0.78 and Proposed ERF-LSTM values starts from 0.93 to 0.98. The proposed method provides the great results.

5. Conclusion

In this research of stock trading, timely and accurate information is paramount. This study's innovative approach to stock classification, combining Yahoo Finance stocks data with Twitter data through an ensemble model, represents a significant stride towards informed decision-making. The Ensemble Random Forest Classifier and LSTM Neural Network, when harmonized, create a powerful tool for analyzing structured and unstructured data. Weighted averaging further enhances predictive capabilities. By fusing the strengths of both models, we provide traders and investors with a more reliable and accurate means of classifying stocks, paving the way for more profitable and informed trading strategies in dynamic financial markets.

References

1. A. Fernández Vilas, R. P. Díaz Redondo and A. Lorenzo García, "The Eruption of Cryptocurrencies into Twitter Cashtags: A Classifying Solution," in IEEE Access, vol. 8, pp. 32698-32713, 2020, doi: 10.1109/ACCESS.2020.2973735.
2. B. Wang, D. Shan, A. Fan, L. Liu and J. Gao, "A Sentiment Classification Method of Web Social Media Based on Multidimensional and Multilevel Modeling," in IEEE Transactions on Industrial Informatics, vol. 18, no. 2, pp. 1240-1249, Feb. 2022, doi: 10.1109/TII.2021.3085663.
3. C. Lee and I. Paik, "Stock market analysis from Twitter and news based on streaming big data infrastructure," 2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST), Taichung, Taiwan, 2017, pp. 312-317, doi: 10.1109/ICAwST.2017.8256469.
4. C. Nousi and C. Tjortjis, "A Methodology for Stock Movement Prediction Using Sentiment Analysis on Twitter and StockTwits Data," 2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM), Preveza, Greece, 2021, pp. 1-7, doi: 10.1109/SEEDA-CECNSM53056.2021.9566242.
5. M. Qasem, R. Thulasiram and P. Thulasiram, "Twitter sentiment classification using machine learning techniques for stock markets," 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Kochi, India, 2015, pp. 834-840, doi: 10.1109/ICACCI.2015.7275714.
6. M. Skuza and A. Romanowski, "Sentiment analysis of Twitter data within big data distributed environment for stock prediction," 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), Lodz, Poland, 2015, pp. 1349-1354, doi: 10.15439/2015F230.
7. S. Urolagin, "Text Mining of Tweet for Sentiment Classification and Association with Stock Prices," 2017 International Conference on Computer and Applications (ICCA), Doha, Qatar, 2017, pp. 384-388, doi: 10.1109/COMAPP.2017.8079788.
8. W. Weng, Y. Liu, S. Wang and K. Lei, "A multiclass classification model for stock news based on structured data," 2016 Sixth International Conference on Information Science and Technology (ICIST), Dalian, China, 2016, pp. 72-78, doi: 10.1109/ICIST.2016.7483388.
9. Y. E. Cakra and B. Distiawan Trisedya, "Stock price prediction using linear regression based on sentiment analysis," 2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, Indonesia, 2015, pp. 147-154, doi: 10.1109/ICACSIS.2015.7415179.
10. Z. Zhao, C. Yang and Q. Wu, "TwoWin-SOVA: Two Windows Discrete Cosine Transform and Synthetic Minority Oversampling Technique One-Versus-All Ensemble Classifiers for Imbalanced Hyperspectral Image Explainable Classification," in IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1-16, 2023, Art no. 5520216, doi: 10.1109/TGRS.2023.3307123.