iJRASET
International Journal For Research in Applied Science and Engineering Technology

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 11 **Issue:** V **Month of publication:** May 2023

**DOI:** https://doi.org/10.22214/ijraset.2023.53169

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 11 Issue V May 2023- Available at www.ijraset.com*

# Prediction and Sentiment Analysis of Stock using Machine Learning

Bharath Gowda B K[1], Ashitha B K[2], Shree Sakshi Suresh[3], S Vishal[4], Asst. Prof. Mrs. T Shilpa[5]
[1, 2, 3, 4, 5]*Department of Information Science and Engineering, Bangalore Institute of Technology, VV Puram, Bangalore*

***Abstract:*** ***The stock market is highly volatile and subject to rapid fluctuations and changes, with seemingly insignificant news or rumors potentially impacting the value of a stock. This project aims to assist retail investors in navigating the complex and volatile modern stock market by developing machine learning models for predicting stock market trends and identifying promising stocks. The solution involves components to facilitate data collection through sitemap spider, data processing through coreference resolution technique, model training using random forest, and sentiment analysis for the stock sentiment. The results of sentiment analysis are merged with technical indicators data to create the dataset for training. The proposed system has shown results with 72% accuracy in assisting investment decision making. The project's applications extend to news analysis, investment decision-making, risk management and trading strategies.***
***Keywords: Stock Prediction, Sentiment, Machine Learning, Coreference Resolution, Random Forest, Sitemap Spider.***

## I. INTRODUCTION

Stock is a general term used to describe the ownership certificates of any company. A share, on the other hand, refers to the stock certificate of a particular company. Holding a particular company's share makes the owner a shareholder. Stock market prediction refers to the attempt to forecast the future performance of a company's stock or the overall stock market. The efficient-market hypothesis suggests that stock prices reflect all currently available information and any price changes that are not based on newly revealed information are inherently unpredictable. However, there are many methods and technologies that purport to allow for the prediction of future stock prices.

This project aims to help users analyze news articles online in bulk, which can be a time- consuming task due to the large volume of articles and its amount of content. Articles may also contain personal opinions rather than factual information. To address this issue, the project proposes to web-scrape articles in real-time, summarize, and analyze them. The end product should have fast search times, ideally in a few milliseconds, with the most current information possible.

To achieve the goal of this project, natural language processing (NLP) is used to analyze the sentiments expressed in the news articles. NLP is a field of artificial intelligence that deals with the interaction between computers and humans through the use of natural language. By analyzing the sentiments expressed in the articles. The aim is to gauge the overall sentiment towards a particular stock or the overall market. These sentiments will serve as input features for the machine learning models, which will be trained to predict whether the stock market will trend upwards or downwards and identify specific stocks that are likely to perform well. The models will be updated in real-time as new articles are published.

## II. RELATED WORK

Researchers have proposed a few methods in Machine Learning models and other related methods to predict the direction of the stock and whether the stock value would increase, decrease, or stagnate.

In Otmazgin, Shon, Arie Cattan, and Yoav Goldberg's [1] paper relating to coreference, they had proposed a new coreference model F-COREF that would replace the related pronouns of an entity to the entity name itself. F-COREF is an open source library in python. And that their proposed system would be 29 times faster than the older AllenNLP model. This paper has utilized two complementary directions to obtain a fast and efficient coreference model.

To substantially reduce the size of the s2e model using knowledge distillation from the LINGMESS model. Knowledge Distillation is the process of learning a small student model from a large teacher model. Teacher model uses the state-of-the-art LINGMESS model of the authors Otmazgin et al. Student model will build student model as a variant of the s2e model with fewer layer and parameters. The implementation aims to maximize parallelism via batching while limiting the number of unnecessary computations such as padded tokens.

With Mehta, Yash, Atharva Malhar, and Radha Shankarmani.[2], the researchers focused on looking into different methods of dynamically learning the market and its trends using three models: ARIMA, LSTM and Linear Regression.
ARIMA is best for short term prediction despite its low accuracy and LSTM is best for long term predictions.
Huang, Yuxuan, Luiz Fernando Capretz, and Danny Ho [3] had proposed a system of testing out three models with 22 years worth of financial data. The three models being Feed-forward Neural Network (FNN), Random Forest (RF) and Adaptive Neural Fuzzy Inference System (ANFIS) for stock prediction based on fundamental analysis.
Dhaval Dangaria, Riccardo Giacomelli, Wilfrido Martinez[4] had proposed BigBirdFLY, a BigBird system that was solely focused on financial NLP tasks. BigBird was meant to overcome the drawbacks of BERT being that it has a 512 character limit. BigBird ranks relevance of sections of text and summarized those sections.
Zhou, Zhihan, Liqian Ma, and Han Liu [5] had aimed to develop a system that can automatically detect and classify corporate events, such as earnings releases or mergers and acquisitions, for use in event- driven trading. The authors develop a deep learning-based system that combines natural language processing techniques with financial information to identify and classify corporate events in news articles.
Basak, Suryoday, et al. [6] proposed a paper in which using tree based classifiers, instead of a traditional forecasting style problem, it is used to simply test the direction of the stock value, whether it will increase or decrease. The problem is posed as a classification problem, where the class labels may be ±1, indicating an increase or a decrease in the price of a stock with respect to n days back.
Zhang, Jingqing, et al. [7], in their paper, proposed the system of Pegasus. A new pre training approach for abstractive summarization. The approach is based on the Transformer architecture and uses a new technique for pre-training called "gap-sentence extraction". The gap-sentence extraction technique involves masking out a portion of the input text and then training the model to predict the missing information. This pre-training task forces the model to learn about the relationships between words and phrases in the input text, which is important for understanding the meaning of the text and generating an appropriate summary.
Araci, Dogu [8] has proposed the system of FINBERT, a BERT based language model which is focused on financial tasks. BERT is based on transformers. It is seen that even with a smaller training set and fine tuning only a part of the model, FINBERT outperforms many state-of-the-art technologies.
Vaswani, Ashish, et al. [9] proposed a new architecture for neural machine translation called Transformer, which is based solely on attention mechanisms. The Transformer model consists of multi- head self-attention layers followed by feedforward neural network layers. This allows the model to capture both long-range dependencies and short-range dependencies, which are essential for NLP tasks like translation.
Maini, Sahaj Singh, and K. Govinda [10] proposes the use of machine learning techniques for evaluating data concerned to the stock market using a novel method for prediction of stock prices to minimize the risk of investment in a stock market. Natural Language Processing is used on content derived from news articles and other relevant sources along with an Ensemble learning model called Random Forest model and Support Vector Machine.
In conclusion, usage of systems like FINBERT [8] and Random Forest [10][6] as well as transformers [9] have been useful in predicting the direction of the value of the stock [6]. There have been many drawbacks however, as transformers have a limit in translating number of words leading to context fragmentation [9].

## III. PROBLEM DEFINITION

The modern stock market is highly volatile and subject to rapid changes, with even the smallest news or rumor potentially impacting the value of a stock or bond. This can make it difficult for investors, particularly novice investors, to choose the best stocks to invest in due to the numerous factors that can influence stock performance. One of these factors is the vast amount of financial news articles that are published on a daily basis, which can be challenging to keep up with and analyze for relevant information. In addition, the difficulty of analyzing large amounts of numerical data at various times and dates makes stock market prediction a challenging task. Our proposed solution aims to address these issues by providing a tool that can assist investors in making more informed investment decisions.

## IV. OBJECTIVES

*1)* Develop machine learning models that can accurately predict stock market trends and identify promising stocks based on the analysis of financial news articles.
*2)* Integrate technical analysis (TA) in the stock prediction process by incorporating sentiment analysis of news articles.
*3)* Use computational linguistic and machine learning techniques to efficiently scan through large volumes of text across multiple news channels and identify key opinions and trends that may impact the stock market.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 11 Issue V May 2023- Available at www.ijraset.com*

## V. METHODOLOGY

The system design of the project encompasses a well-structured architecture, incorporating various components and modules to facilitate data collection, data processing, model training, and sentiment analysis for stock sentiment and prediction.

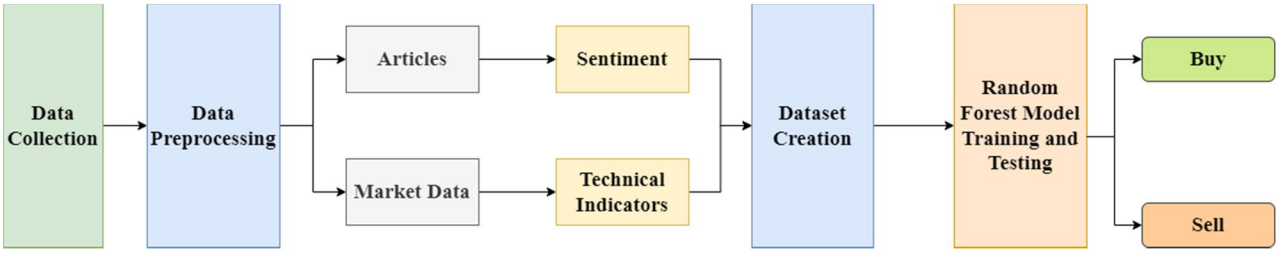



Fig. 1: System Design

In Figure 1, depicts system architecture of the sentiment analysis application. This includes various components for data collection, preprocessing dataset creation, training of ML model and testing.

The first step of the process is the collection of data in the form of news articles about target companies from reputable news sources. The data undergoes preprocessing, and is split into the news articles itself, and the market data. The articles undergo sentiment analysis, while the market data is subjected to the application of technical indicators, the data being calculated under the formulas of the indicators.

The resulting data (the sentiment analyzed data from articles and the market data that was calculated under the technical indicator formulas) is then collected, and undergoes the process of dataset creation, where all the data is manipulated into fitting the created dataset. This resulting dataset, is then utilized for model training and testing of data (the chosen model being Random Forest). Training and testing of the model ensures increased chances of accuracy in prediction and classic.

After training and testing of the Random Forest model, the model can then classify data into classes (those being the 2 classes or 4 classes) to parameterize the prediction of the stocks of a company. From the resulting classes, the result of whether a stock should be bought or sold is then displayed.

### *A. Data Collection*

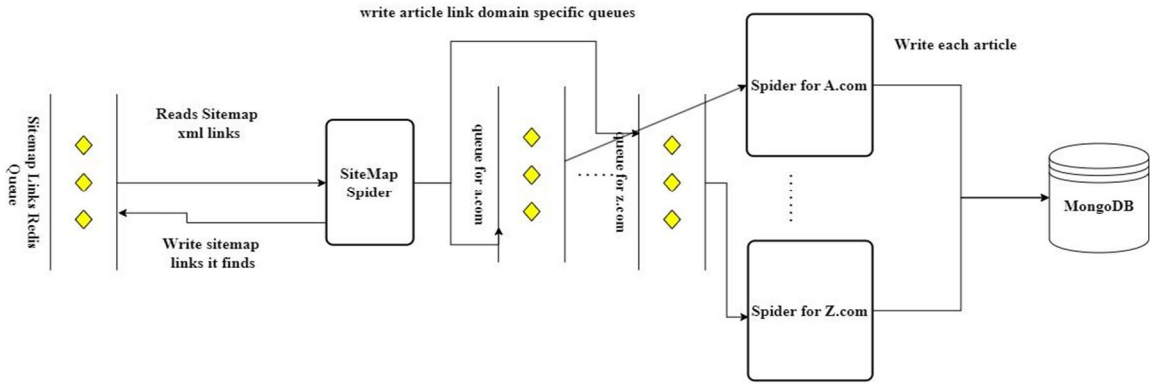



Fig. 2: News article collection Architecture.

Obtaining historical stock prices involves utilizing the Yahoo Finance website, which provides a convenient API. To ensure accuracy, only data from 2017 onwards is considered. Accessing this information is simple and readily available through the platform. Our primary objective is to calculate whether the stock prices of various companies rose, stayed the same, or fell during the given period.

Figure 2 depicts architecture for news article collection.

The key components include SitemapSpider and MongoDB. SitemapSpider is used to crawl a site by discovering the URLs using Sitemaps. The search results are stored in MongoDB a NoSQL database.

The National Stock Exchange (NSE) website serves as a reliable source for obtaining a comprehensive list of companies and their respective symbols. This information is essential for accurately identifying and tracking each company's stock.

Acquiring relevant news articles poses a challenge, but this can be overcome by employing the Scrapy framework. By coding a crawler, navigating a predefined list of reputable websites and extract the desired article content is easy. It is critical to ensure that only trusted and well-structured sources are included in this list to avoid the negative impact of fake news or unstructured data on our model's results.

To overcome Scrapy's single-threaded limitation and enable concurrent data retrieval, Redis is utilized. Redis acts as an in-memory data store, leveraging RAM for storing the collected data. This architectural approach ensures that our system's performance remains unaffected, despite Scrapy's inherent single-threaded nature.

*B. Data Processing*

Once a sufficient number of articles is gathered, the following steps are undertaken. Firstly, a curated list is created, consisting of 48 company names along with their associated common search terms. For example, "Reliance Industries" would be associated with the term "Reliance," and "State Bank of India" with "SBI."

Using this list, a database search is performed to retrieve articles that mention the target companies by utilizing the identified search terms. This enables the extraction of relevant articles pertaining to each company. To enhance the analysis further, the content of each article is processed using coreference resolution techniques. This involves identifying expressions within the text that refer to the same entity, aiding in improving the overall understanding and context of the article content.

Next, the article content is segmented into sentences using the "sent_tokenize" function from the NLTK library. For sentences containing terms related to the target company, they are passed through the FinBERT model. The sentiment of each sentence is then evaluated and recorded, resulting in a sentiment score assigned to each article. By implementing these procedures, the aim is to curate a comprehensive dataset of articles, analyze sentiment at the sentence level, and gain valuable insights into the sentiment surrounding each company.

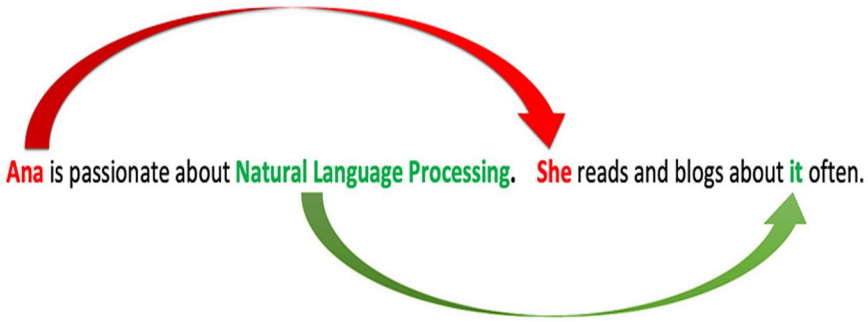



Fig. 3: Example of coreference resolution technique

Figure 3. shows an example of coreference resolution technique. Ana is a Graduate Student at UT Dallas. She loves working in Natural Language Processing at the institute. Her hobbies include blogging, dancing and singing. Here, “Ana”, “Natural Language Processing” and “UT Dallas” are possible entities. “She” and “Her” are references to the entity “Ana” and “the institute” is a reference to the entity “UT Dallas”.

*C. Sentiment Analysis*

Once the articles are grouped by company, obtain the market data spanning from 2017 to 2023 is obtained. This data encompasses various parameters for each trading day, such as the opening and closing prices, trading volume, highest and lowest prices reached during the day.

Utilizing this market data, our next step involves calculating technical indicators and analyzing the changes in stock prices compared to the previous trading day. These indicators provide valuable insights into the market trends and help in assessing the overall market conditions. Concurrently, sentiment data extracted from the articles by their respective publication dates is grouped. This sentiment data, which comprises the sentiment scores obtained from the articles, is averaged for each specific date. This allows us to capture the aggregated sentiment surrounding a company for a particular day.

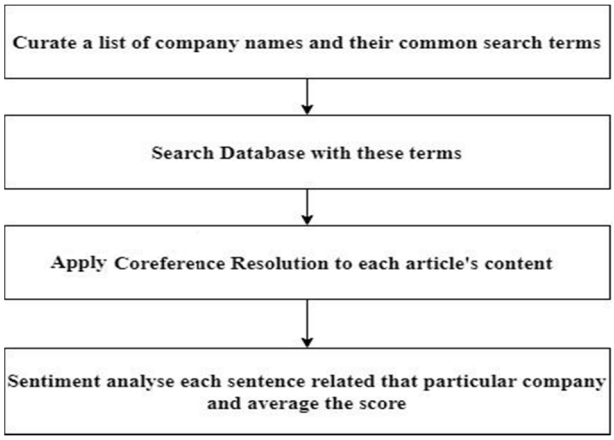



Fig. 4: Sentiment analysis.

Figure 4 shows the entire process of the sentiment analysis. The multiple steps include curating the list of companies, search the database for required information, apply corefernce resolution for the article's content and sentiment analysis to obtain the scores.
By performing these operations, the aim is to combine the market data with sentiment analysis, enabling us to analyze the impact of news sentiment on stock prices. This comprehensive approach provides a deeper understanding of the interplay between market dynamics and news sentiment.
Figure 5 depicts various steps for Grouping articles and merging with market data. Each selected article and article sentiment obtained by yahoo finance are merged by date. Two datasets are created for each article. The first dataset has price and is classified into 4 classes, the second dataset has price change and is classified into 2 different classes.
The Sentiment summarizer component looks at the stock related details on the website. It collects and summarizes the sentiment for individual stocks. It takes pieces of text od website and summarize them into a more manageable paragraph or see the most important sentences in the text. Typical sentiment can include "BUY", "HOLD" or "SELL". This application is developed using Python. The data from Sentiment Summarizer is stored in Mongo DB. Any additional information required for sentiment analysis is also stored in MongoDB.

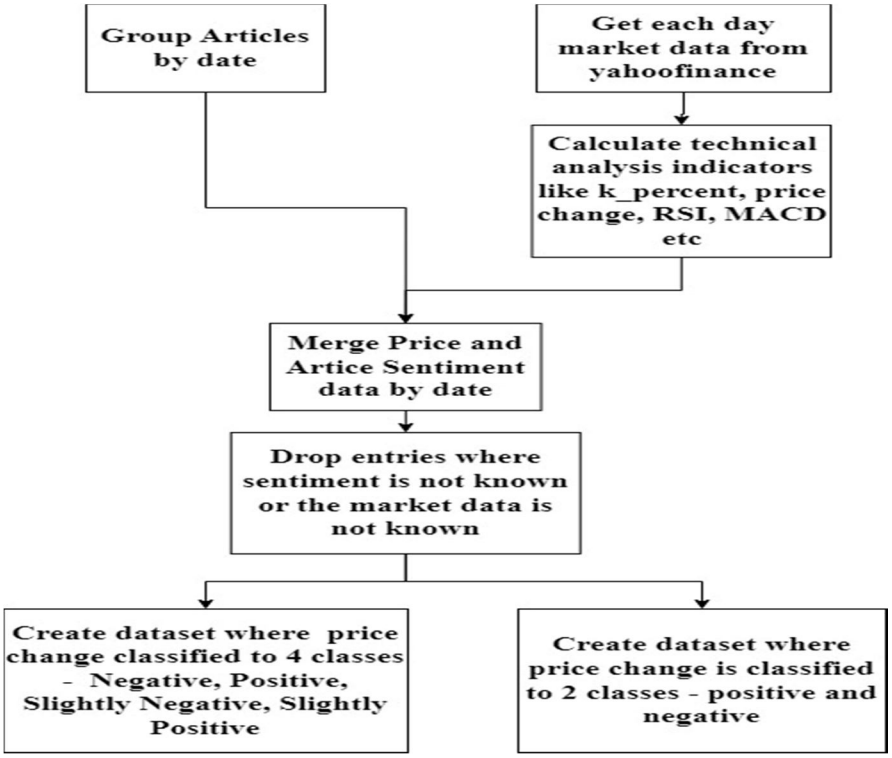



Fig. 5: Grouping articles and merging with market data

*D. Training Model*

To train the models for each dataset, a standardized process is followed. Firstly, the dataset is read using the powerful pandas library, enabling efficient data manipulation and analysis. Necessary features from the dataset are carefully extracted, focusing on key indicators such as RSI, k percent, r percent, Price Rate of Change, MACD, On Balance Volume, sentiment, and prediction. By selectively choosing these features, the aim is to capture the most relevant information for our predictive models.

Next, the dataset is split into separate training and test data subsets. This critical step is facilitated by the widely-used train_test_split() function provided by the sklearn library. By allocating a portion of the data for testing purposes, it is ensured that our models can be evaluated on unseen data, providing a robust assessment of their performance and generalization capabilities.

Moving forward, the formidable RandomForestClassifier class from the sklearn library is employed to create a random forest classifier model. This ensemble learning algorithm combines the predictions of multiple decision trees to enhance accuracy and mitigate overfitting. The model's hyperparameters are carefully adjusted, such as the number of estimators, which determines the number of trees in the model. By fine-tuning these parameters, a balance between model complexity and performance is formed.

Once the model is constructed, the training data fit to the classifier, allowing it to learn patterns and relationships within the features. Subsequently, evaluation the model's performance using the test data is done, employing various metrics to gauge accuracy, precision, recall, and other performance indicators. This comprehensive evaluation provides valuable insights into the model's predictive capabilities and its ability to generalize to new, unseen data.

Through adherence to this systematic approach, the objective is to train robust models that can proficiently analyze and predict stock price movements. This is accomplished by harnessing the power of random forests and utilizing the abundance of features extracted from the datasets. To maximize the effectiveness of our Random Forest classifier, a randomized search technique is employed for hyperparameter tuning. This method enables us to thoroughly explore a broad range of values for different hyperparameters, ensuring that our model is finely tuned to achieve optimal performance. During the randomized search, a range of values for each hyperparameter are considered. For instance, there are various options for the number of trees in the forest (n_estimators), ranging from 200 to 1800 with increments of 200. Similarly, different possibilities for the maximum number of features to consider at each split (max_features), including 'auto', 'sqrt', None, and 'log2'. Additionally, the impact of varying the maximum depth of the trees (max_depth) within a range of values from 10 to 100, as well as the minimum number of samples required to split a node (min_samples_split), ranging from 2 to 40 is examined. The minimum number of samples required at each leaf node (min_samples_leaf) is explored using values such as 1, 2, 7, 12, 14, 16, and 20. Furthermore, an assessment of the effect of using bootstrapping (bootstrap) by evaluating both True and False values is carried out. By comprehensively evaluating this range of hyperparameter combinations, it is ensured that our Random Forest classifier is optimized to its fullest potential. This rigorous search process allows us to identify the most effective hyperparameter values that yield the highest predictive accuracy and enhance the model's ability to generalize well to unseen data. Through the strategic exploration of hyperparameters, there is a strive to create a Random Forest classifier that leverages the full range of possible configurations, resulting in a robust and accurate model for stock price prediction. The random grid is defined using these hyperparameter ranges, and the RandomizedSearchCV function is used to randomly search this grid of hyperparameters to find the best combination for the Random Forest classifier. After defining the random search, the fit() method is called on the RandomizedSearchCV object with the training data to find the best hyperparameters for the Random Forest classifier.

## VI. EVALUATION AND RESULTS

A correlation heatmap is a graphical representation of a correlation matrix representing the correlation between different variables. The value of correlation can take any value from -1 to 1. A correlation heatmap is a heatmap that shows a 2D correlation matrix between two discrete dimensions, using colored cells to represent data from usually a monochromatic scale. The primary purpose of the seaborn heatmap is to show the correlation matrix by data visualization.

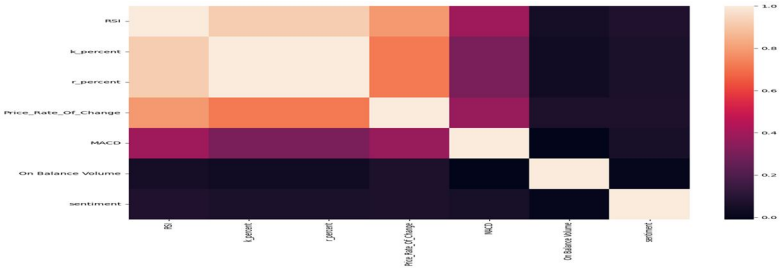



Fig. 6: Feature heat map

Figure 6 is Heat map of correlation of features. It helps to find the correlation between 7 technical indicators (RSI, k_percent. r_percent, price rate of change, MACD, On Balance Volume and sentiment distribution) extracted during the process and also helps to select best features for model building.

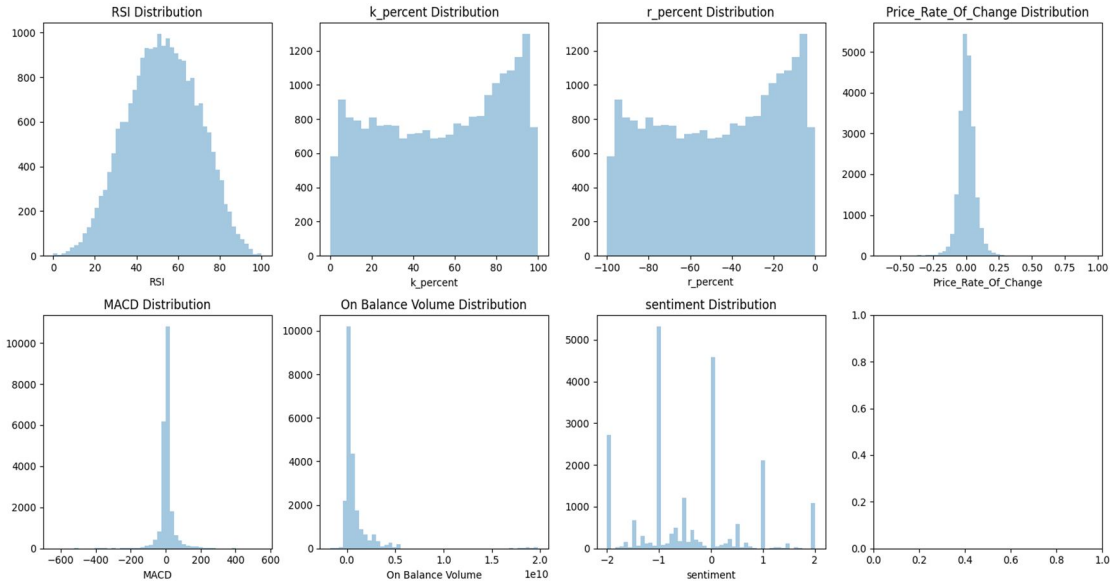



Fig. 7: Feature Distribution

Figure 7 displays the feature distribution graph of 7 technical indicators that include RSI, k_percent. r_percent, price rate of change, MACD, On Balance Volume and sentiment. Feature distribution is the distribution of a feature over its range, with value on the horizontal axis and frequency on the vertical axis. For string and categorical encoded integers, the feature distribution is displayed as a bar chart with the highest, i.e., the label with the highest count, on the left side.

| Model Name | Train Accuracy | Test Accuracy |
|---|---|---|
| 2 Class Model | 76.62 | 70.99 |
| 4 Class Model | 60.98 | 52.15 |

Fig. 8: Models Accuracy Comparison

Figure 8 shows the various accuracies achieved after the training the random forest model.

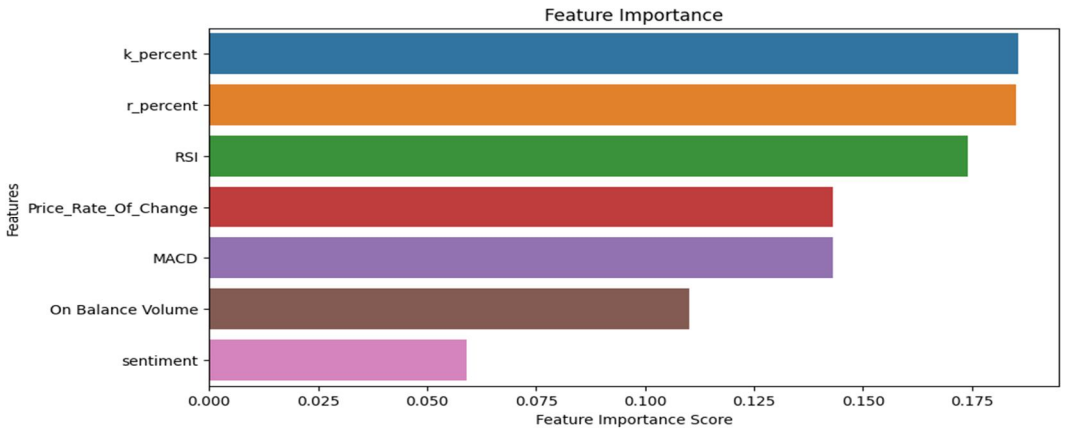



Fig. 9: Feature Importance for 2 class model

Figure 9 displays Feature Importance of a single model of 2 classes. Feature importance assigns the score of input features based on their importance to predict the output. More the feature's will be responsible to predict the output more will be their score.

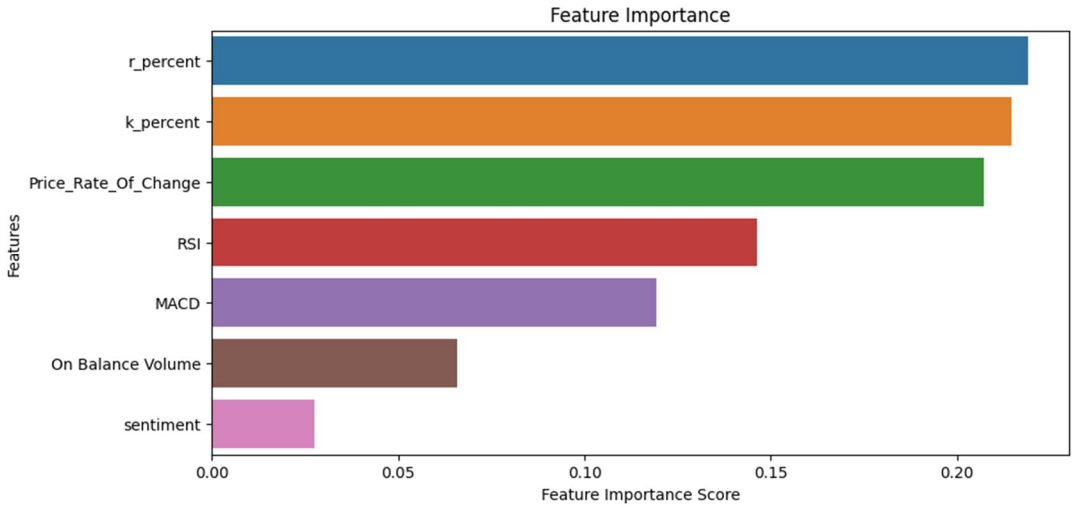



Fig. 10: Feature Importance for 4 class model

Figure 10 displays Feature Importance of a single model of 4 classes. Feature importance assigns the score of input features based on their importance to predict the output. More the feature's will be responsible to predict the output more will be their score.

## VII. CONCLUSION

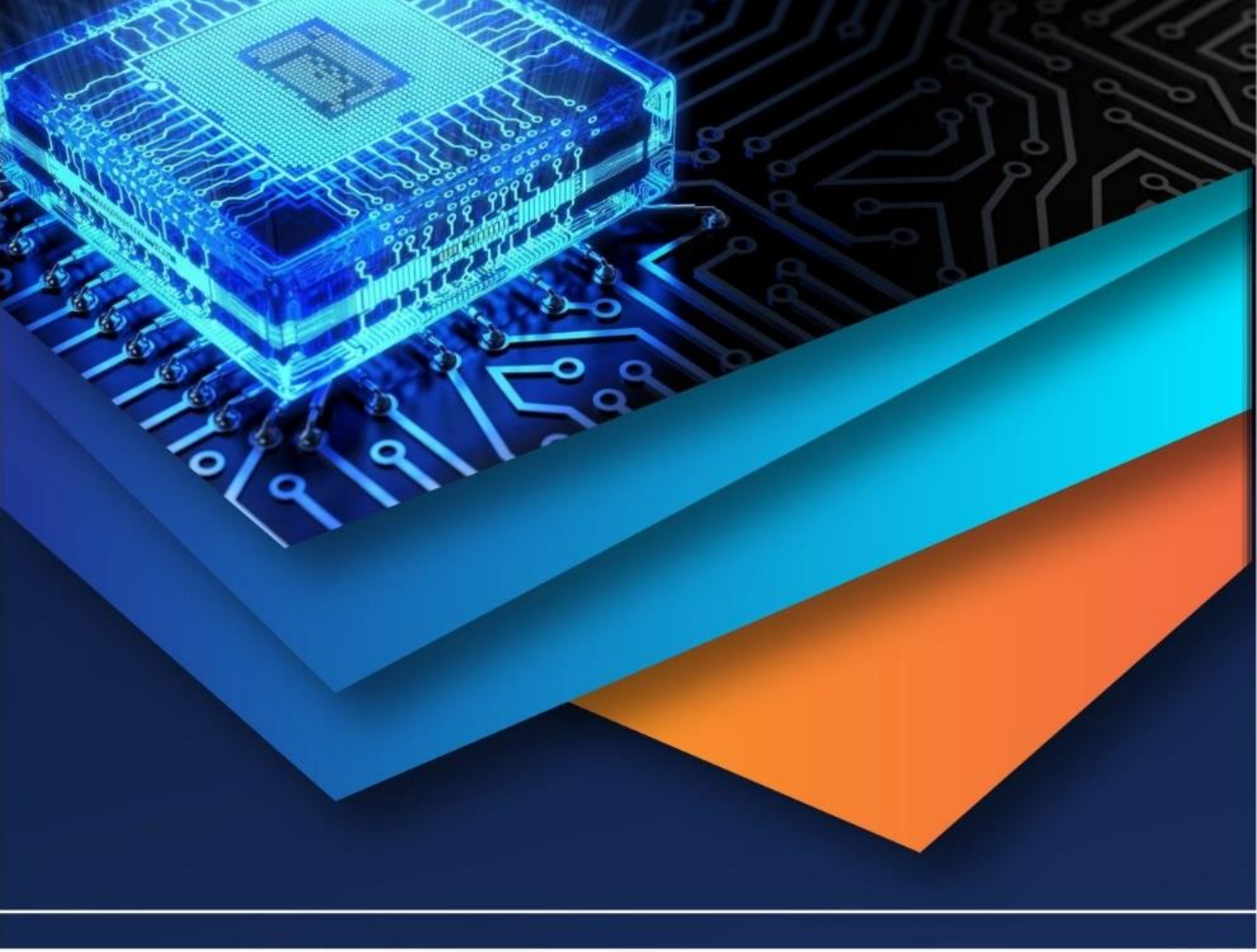
This project aimed to predict the price movement of a particular company by analyzing its news articles sentiment and technical indicators. Through the creation of a dataset containing both positive and negative sentiment articles, and the extraction of relevant features from them, machine learning models were trained and tested for their performance in predicting the next day's price movement.

The results showed that the machine learning models had a significant level of accuracy in predicting the price movements of the company. The project's findings suggest that utilizing sentiment analysis and technical indicators in tandem could provide valuable insights into the future movement of a company's stock price.

Overall, the project's approach of combining machine learning and classification techniques to predict price movements based on sentiment analysis and technical indicators has the potential to be a powerful tool in the financial industry for traders and investors alike.

## REFERENCES

[1] Otmazgin, Shon, Arie Cattan, and Yoav Goldberg. "F-COREF: Fast, Accurate and Easy to Use Coreference Resolution." arXiv preprint arXiv:2209.04280 (2022).
[2] Mehta, Yash, Atharva Malhar, and Radha Shankarmani. "Stock price prediction using machine learning and sentiment analysis." 2021 2nd International Conference for Emerging Technology (INCET). IEEE (2021).
[3] Huang, Yuxuan, Luiz Fernando Capretz, and Danny Ho. "Machine learning for stock prediction based on fundamental analysis." 2021 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE (2021).
[4] Dhaval Dangaria, Riccardo Giacomelli, Wilfrido Martinez; BigBirdFLY: Financial Long Text You can read; (2021).
[5] Zhou, Zhihan, Liqian Ma, and Han Liu. "Trade the event: Corporate events detection for news-basedevent-driven trading." arXiv preprint arXiv:2105.12825 (2021).
[6] Basak, Suryoday, et al. "Predicting the direction of stock market prices using tree based classifiers." The North American Journal of Economics and Finance 47 (2019).
[7] Zhang, Jingqing, et al. "Pegasus: Pre-training with extracted gap-sentences for abstractive summarization." International Conference on Machine Learning. PMLR (2020).
[8] Araci, Dogu. "Finbert: Financial sentiment analysis with pre-trained language models." arXiv preprint arXiv:1908.10063 (2019).
[9] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).
[10] Maini, Sahaj Singh, and K. Govinda. "Stock market prediction using data mining techniques." 2017 International Conference on Intelligent Sustainable Systems (ICISS). IEEE, (2017).



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY