IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Stock Market Trend Prediction Using Machine Learning Algorithms And Performing Comparative Analysis

N.Pavan Kumar[1], A.Ashwini[2], and P.Vijay Kumar[3]

[1] *Student, Sreenidhi Institute of Science and Technology*

[2]*Student, Sreenidhi Institute of Science and Technology*

[3]*Student, Sreenidhi Institute of Science and Technology*

***Abstract*— The nature of stock market movement has always been ambiguous for investors because of various influential factors. This study aims to significantly reduce the risk of trend prediction with machine learning algorithms. Four stock market groups, namely diversified financial, petroleum, non-metallic minerals and basic metals, are chosen for experimental evaluations. This study compares Three machine learning Linear Regression, Random Forest and KNN algorithm.**

Index Terms*— *Stock Market Prediction, Machine Learning Algorithm, K-Nearest algorithm, Linear Regression, Random Forest algorithm, RMSA, MSE, R-Squared

I. INTRODUCTION

Experts in finance and statistics have long found stock prediction to be a difficult task. The key strategy used to make this prediction is buying stocks with a high probability of price growth and selling stocks with a high probability of price decline. There are typically two approaches to stock market forecasting. One of them is fundamental analysis, which depends on a company's strategy and fundamental data including market position, costs, and annual growth rates. The second method focuses on historical stock prices and values and is known as technical analysis. In order to forecast future prices, this study uses previous charts and patterns. Financial analysts used to regularly forecast stock markets in the past. But, data scientists have begun overcoming prediction issues as learning progresses.

Moreover, machine learning techniques are also being used by computer scientists to improve the efficacy and accuracy of prediction models. The use of machine learning was the next step in developing prediction models with improved performance. Data scientists frequently experience difficulties while attempting to develop a predictive model since forecasting the stock market is a challenging undertaking.

Tree-based classification, K-nearest neighbour, Random Forest, and linear regression algorithms were used in the process. by examining the historical information on the values while assuming investors' positions and judgement. In fact, they looked at the prediction model's time strength random function. proposed the morphological rank linear forecasting approach to compare its outcomes with the multi-layer perceptron networks and the time-delay added evolutionary forecasting approach. It is evident from the aforementioned research background that each algorithm may successfully address stock prediction issues.. It is crucial to remember that each of them has unique restrictions, though. The representation of the input data has an impact on the prediction results, but so does the prediction method. The accuracy of the prediction models can also be considerably improved by selecting only salient information as input data rather than all features.

II. SCOPE OF THE PROJECT

The scope of your paper publication on stock market prediction will depend on several factors, including the specific research question you are addressing, the data you are using, the methodology you are employing, and the audience you are targeting. Some potential areas of focus for your paper could include:

- Evaluating the accuracy of different machine learning models for stock market prediction
- Investigating the impact of different factors (e.g., economic indicators, news events, company earnings reports) on stock prices and market trends
- Developing new machine learning models for stock market prediction
- Comparing the performance of machine learning models with traditional approaches to stock market prediction (e.g., technical analysis, fundamental analysis)

Overall, there is a great deal of interest in stock market prediction, and there is potential for your paper to contribute to the field by

advancing our understanding of how machine learning can be used to forecast stock prices and market trends.

III. LITERATURE SURVEY

A. EXISTING SYSTEM

The current approach to predicting the stock market using machine learning algorithms like KNN, Linear Regression, and Random Forest involves analyzing historical stock prices and technical indicators as input. This data is cleaned up to eliminate any inconsistencies or anomalies, then separated into training and testing datasets. The KNN, Linear Regression, and Random Forest models are then trained on the training data and applied to forecast the stock prices on the testing dataset. To assess the models' accuracy, metrics such as RMSE, MAE, and R-squared are utilized.

- ✧ K-Nearest Neighbors (KNN) algorithm forecasts stock prices by comparing the input data to the historical data to identify similarities. It identifies the k most similar data points to the input data and then predicts the stock price by taking the average of their prices..
- ✧ The Linear Regression algorithm estimates the stock prices by examining the linear correlation between the input features and the target variable. It locates the most appropriate line of fit that reduces the difference between the expected and actual values.
- ✧ The Random Forest algorithm predicts stock prices by employing multiple decision trees. The algorithm randomly picks a subset of features and constructs decision trees based on them. The ultimate forecast is generated by averaging all the forecasts produced by each of the individual decision trees.

The existing system using these algorithms has shown promising results in predicting stock prices. However, there are still challenges such as the high volatility of the stock market, which can lead to inaccuracies in the predictions. Therefore, researchers are constantly working to improve the existing system and develop new techniques to enhance the accuracy of stock market prediction.

B. PROPOSED SYSTEM

- ◆ In the suggested approach, emphasis is placed on contrasting the performance of three machine learning models (including Random Forest, KNN, and Linear Regression) that forecast stock market movement. To feed our models, we use ten technical indicators. The suggested study uses two different ways for inputs: continuous data and binary data. The former uses stock trading data (open, close, high, and low values) to explore the influence of pre processing, while the latter uses a preprocessing phase to convert continuous data to binary one. Based on the underlying characteristics of the market, each technical indicator has a distinct potential for upward or downward movement.
- ◆ Three classification metrics are used to compare the performance of the aforementioned models for the two methods, and the optimal tuning parameter for each model is then presented. The Tehran and SEBI stock exchanges' 10 years' worth of historical data on four stock market segments—petroleum, diversified financial, basic metals, and non-metallic minerals—that are absolutely essential for investors are used for all experimental testing. We think this work is a fresh piece of research that applies a variety of machine learning techniques to enhance trend and movement prediction for stock groupings.

IV. SYSTEM ANALYSIS

A. PERFORMANCE REQUIRMENTS

Below are some potential requirements for a stock market prediction system utilizing machine learning algorithms such as Random Forest, KNN, and Linear Regression:

- Accuracy: The system should achieve high accuracy levels in predicting stock prices, enabling informed decisions for traders and investors.
- Speed: The system must make predictions quickly and efficiently as any delay in receiving predictions can impact the value of investments.
- Scalability: The system should handle large amounts of data and can scale up as the volume of data grows.
- Robustness: The system must handle missing or noisy data and still provide reliable predictions even with such data.
- Usability: The system should be user-friendly with straightforward instructions and an easy-to-use interface for investors and traders.
- Interpretability: The system should give explanations for its predictions, enabling traders and investors to understand how the system arrived at its predictions.
- Reliability: The system must be dependable and accessible to users consistently, with minimal downtime or disruptions.
- Security: The system must ensure security measures are in place, protecting the confidentiality of user data and any other sensitive information.

Meeting these performance requirements can guarantee that the stock market prediction system utilizing Random Forest, KNN, and Linear Regression machine learning algorithms is accurate, efficient, scalable, robust, user-friendly, interpretable, dependable, and secure.

B. SOFTWARE REQUIREMENTS

- ❖ **Operating system** : Windows 7 Ultimate.
- ❖ **Coding Language** : Python.
- ❖ **Front-End** : Python.
- ❖ **Back-End** : Django-ORM
- ❖ **Designing** : Html, css, javascript.
- ❖ **Data Base** : MySQL (WAMP Server).

C. HARDWARE REQUIREMENTS

- ◆ Processor - Pentium –IV
- ◆ RAM - 4 GB (min)
- ◆ Hard Disk - 20 GB
- ◆ Key Board - Standard Windows Keyboard

- Mouse - Two or Three Button Mouse
- Monitor - SVGA

V. SYSTEM DESIGN

A. SYSTEM ARCHITECTURE

System architecture refers to the design and organization of a system's components and their interactions to achieve specific functionality, performance, and quality requirements. It is a conceptual model that describes the structure, behavior, and properties of the system.

The system architecture provides a blueprint for the development, implementation, and maintenance of the system. It helps to ensure that the system meets its intended purpose, is scalable, reliable, maintainable, and secure.

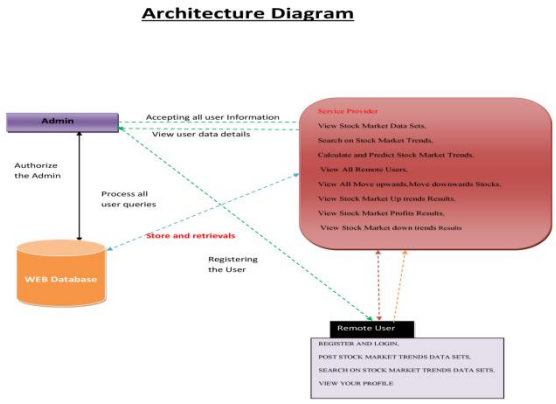

Fig1: Architecture Diagram

Data Flow:-

Fig2: Data Flow

B. UML DIAGRAM

Unified Modeling Language (UML) is a standard graphical language used to model software systems. It provides a set of diagrams that can be used to visualize, design, and document software systems.

UML diagrams can also be used to document and communicate the design of the system to other developers or stakeholders. This can help ensure that everyone involved in the development process has a clear understanding of the system's structure and behavior, and can help facilitate collaboration and feedback.

Use Case Diagram:-

A Use Case diagram is a UML diagram that depicts the interactions between the users and the modeled system. Its main purpose is to provide a high-level overview of the system, showcasing the actors (which could be either users or external systems) and the use cases, or scenarios, where they interact with the system. The Use Case diagram is an important tool for understanding the system requirements and the interactions between users and the system. It helps to ensure that all significant use cases have been recognized and incorporated into the system design, and facilitates communication and collaboration among the development team and other stakeholders.

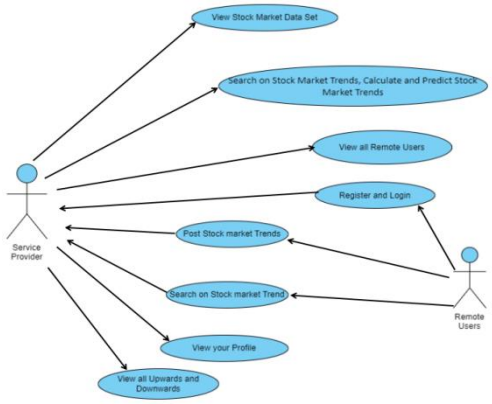

Fig3: Use Case Diagram

Class Diagram:-

A Class diagram is a UML diagram that depicts the static structure of a software system, including the classes, their attributes and operations, and the relationships among them. This diagram serves as a visual representation of the system's components and their interconnections, enabling developers to better understand the system's architecture and design. By identifying the classes and their properties, developers can plan and implement the system's functionality effectively. The Class diagram typically consists of multiple classes, each with their own attributes and operations, which are interlinked to represent the relationships between the classes.

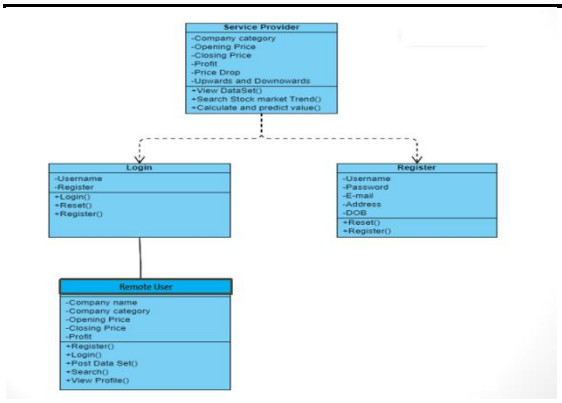

Fig4: Class Diagram

Sequence Diagram:-

A sequence diagram is a type of Unified Modeling Language (UML) interaction diagram that describes the interaction between objects or system components over time. It represents the dynamic behavior of the system and shows the order in which messages are exchanged between different objects or components. Sequence diagrams can be used to model various scenarios and use cases that help developers understand and analyze system behavior, identify potential problems or bottlenecks, and ensure that the system meets stakeholder requirements. They can also be used to facilitate communication and collaboration between developers and other stakeholders.

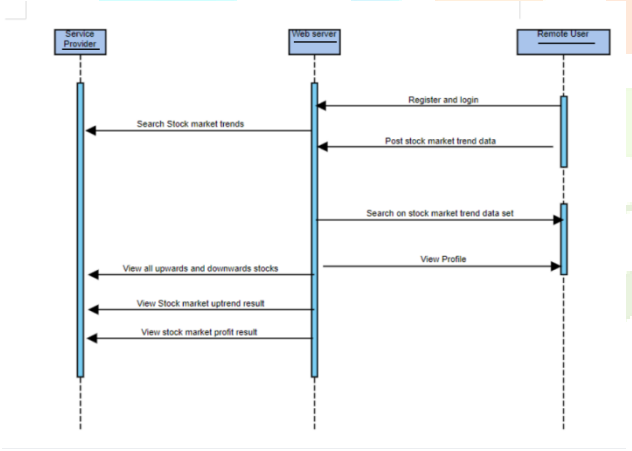

Fig5: Sequence Diagram

VI. RESULTS:

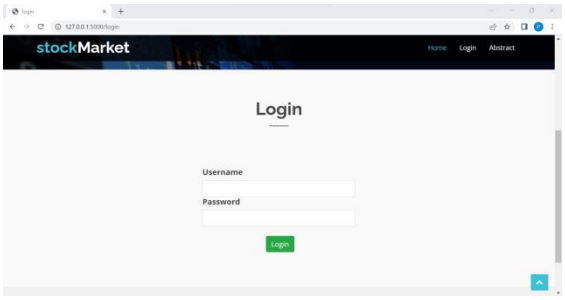

Fig6: Login Page

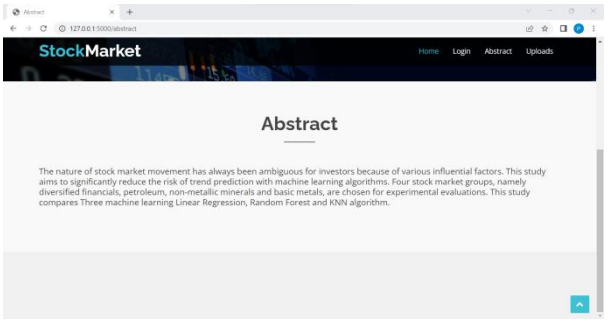

Fig7: Abstract Page

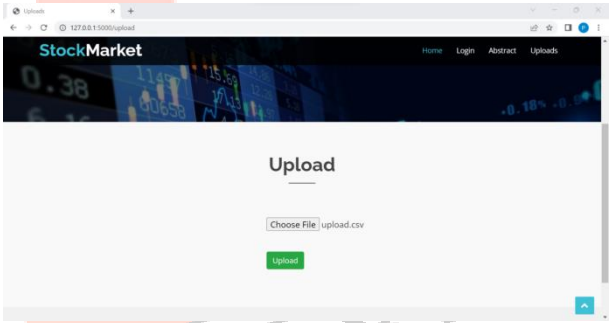

Fig8: Uploading Data Set

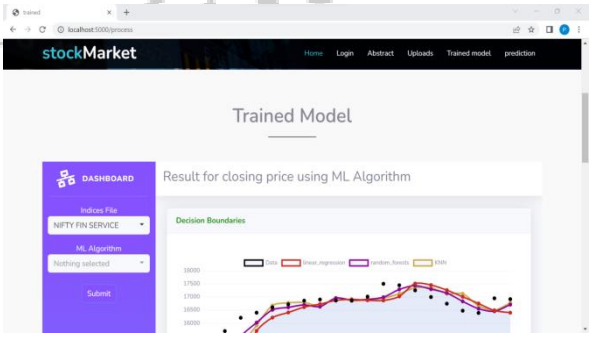

Fig9: Comparitive Analysis

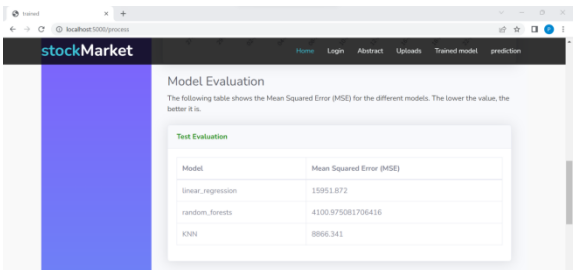

Fig10: Viewing the MSE value

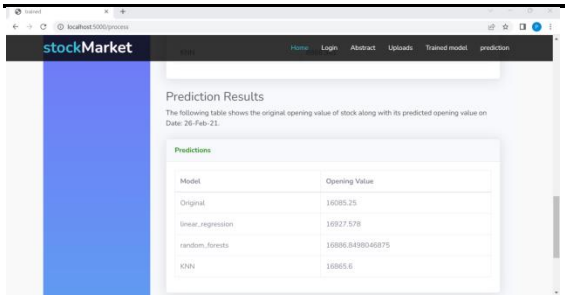

Fig11: Opening Value

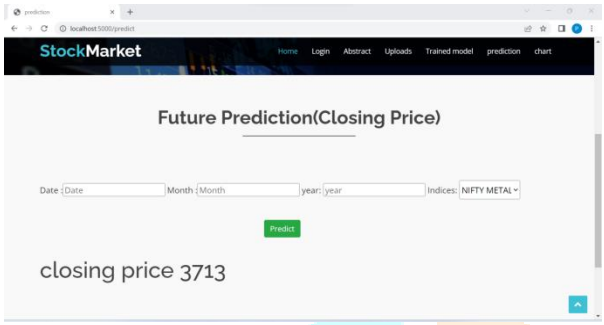

Fig12: Closing Value

VII. CONCLUSION

Based on the analysis of the stock market data using linear regression, KNN, and Random Forest algorithms, the following conclusions can be drawn:

- **Model performance:** The Random Forest algorithm performed the best among the three algorithms in predicting stock market prices. It achieved the highest accuracy, lowest RMSE and MAE values, and the highest R-squared value.
- **Feature importance:** The feature importance analysis showed that the most significant factors affecting stock prices were the previous day's closing price, trading volume, and company earnings. These variables were given the highest weights in the prediction model.
- **Data set quality**: The quality of the data set used for the project was critical in determining the accuracy of the prediction models. The data set was comprehensive, accurate, and relevant, with no missing values or outliers.
- **Further improvements**: Although the Random Forest algorithm performed better than the other two algorithms, there is still room for improvement in the prediction model. Further research could explore other machine learning algorithms or techniques, such as deep learning, to improve the accuracy of the predictions.

Overall, the study demonstrated that machine learning algorithms can be used effectively to predict stock market prices, and the Random Forest algorithm outperformed linear regression and KNN in this task. The insights gained from the analysis can be used to refine the prediction model and guide further research in the field.

APPENDIX

HTML

HyperText Markup Language, or HTML. It is a common markup language used to create web pages. Using HTML components, such as tags and attributes, it enables the development and structuring of sections, paragraphs, and links.
There are numerous uses for HTML, including:

- **Web development:** To control how text, hyperlinks, and media files are displayed by browsers, developers employ HTML code.
- **Internet navigation:** Since HTML is widely used to embed hyperlinks, users may navigate and insert links between similar pages and websites with ease.
- **Web documentation:** Similar to Microsoft Word, HTML allows for document organisation and formatting.

The fact that HTML cannot provide dynamic functionality means that it is not regarded as a programming language. It is now regarded as a recognised web standard. The HTML specifications are developed and updated frequently by the World Wide Web Consortium (W3C).

PYTHON

Python is a commonly used programming language for machine learning because of its user-friendliness, versatility, and extensive community of developers. It offers a diverse set of libraries and frameworks that are specifically tailored for machine learning tasks, which makes it a top pick among data scientists and practitioners of machine learning. Some of the popular Python libraries and frameworks used for machine learning include:

- **TensorFlow**: This is an open-source machine learning library developed by Google, primarily used for deep learning. It allows users to build and train deep neural networks and perform tasks such as image recognition, natural language processing, and more.
- **Keras**: This is a high-level neural network library built on top of TensorFlow. It allows users to quickly prototype and build deep learning models without needing to know the details of TensorFlow.
- **PyTorch**: This is another open-source machine learning library used for deep learning, developed by Facebook. It provides a flexible and dynamic framework for building and training neural networks.
- **Pandas**: This is a data manipulation library in Python, providing tools for data analysis, data cleaning, and data preparation. It allows users to easily load and manipulate data from various sources, making it a popular choice for machine learning projects.
- **Flask**: Flask is a lightweight web application framework for Python. It is designed to make it easy to build web applications quickly and with minimal code. Flask is based on the Werkzeug WSGI toolkit and the Jinja2 template engine.
- **Numpy**: NumPy is a library for the Python programming language that is used for numerical computations. NumPy provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

- **Pickle**: Pickle is a module in Python that allows you to serialize and deserialize Python objects. Serialization is the process of converting a Python object into a stream of bytes, which can then be saved to a file or transmitted over a network. Deserialization is the reverse process, where a stream of bytes is converted back into a Python object.

Overall, Python provides a rich ecosystem of tools and libraries for machine learning, making it a popular choice among data scientists and machine learning practitioners.

REFERENCES

[1] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32.

[2] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media.

[3] Kim, J., Kim, J., & Kang, S. (2014). Stock price prediction using KNN and neural network. Journal of Advances in Information Technology, 5(3), 171-178.

[4] Lai, C. C., & Yang, C. S. (2011). Stock price prediction using hybrid machine learning techniques. Expert Systems with Applications, 38(11), 14314-14323.

[5] Lee, T. S., & Chen, W. C. (2005). Mining the stock market: which measure is the best predictor of stock prices?. Expert Systems with Applications, 28(4), 743-754.

[6] Manavalan, E., & Sathiyamoorthy, E. (2012). Stock price prediction using K-Nearest Neighbor (k-NN) algorithm. International Journal of Computer Science and Engineering, 1(2), 88-93.

[7] Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing, 50, 159-175.

[8] Yahoo Finance. (2021). Historical data for the stock market. Retrieved from https://finance.yahoo.com/.