International Journal of Communication Networks and Information Security
2024,16(4)
ISSN: 2073-607X,2076-0930
https://https://ijcnis.org/
Research Article

Enhancing Stock Price Prediction: Improvising in KNN

Pranit Bari [1*], Lynette D'Mello [1*] , Meet Daftary [2], Param Shah [2], Ansh Bhatt [2]
Harsh Patel [2]

1 Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India

2 Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India

***Corresponding Author:** [1*]Pranit.Bari@djsce.ac.in,

ARTICLE INFO

Received: 10 Aug 2024
Accepted: 14 Sep 2024

ABSTRACT

Stock price prediction is very crucial for informed investment decisions, involving forecasting future stock values which are based on various factors. K-nearest neighbors (KNN) is a machine learning algorithm that can assist in predicting stock prices by identifying patterns and similarities between the target stock and its neighboring data points in a multidimensional feature space. However, traditional KNN algorithms encounter challenges like sensitivity to irrelevant features and outliers, potentially compromising predictive accuracy. To address this, integrating Density-Based Spatial Clustering of Applications with Noise (DBSCAN) before KNN proves effective. DBSCAN identifies and filters out noisy data points and outliers, refining the dataset for subsequent KNN analysis. This integration not only mitigates traditional KNN issues but also uncovers underlying data structures, improving overall predictive power in stock market analysis.

Keywords: Stock Price Prediction, KNN, DBSCAN, Improvised KNN

INTRODUCTION

Stock prices are the current market value of a company's shares and function as the most important indicator for investors and financial analysts. Knowledge of the stock market is essential for making smart investment decisions because they show how the company's financials are, what the market sentiment is, and how the economy is going. Envisioning the future of stocks is key to the investors in order for them to be able to balance their wealth, deal with their hazards, as well as seize investment opportunities.

The prediction of stock prices is essentially done by using machine learning algorithms that can analyze big data, predict future trends and present a forecast of the future market. Therefore, they are important for forecasting stock prices. Machine learning algorithms which are able to analyze data to find trends, predict variables and project forecasts are very important in the stock price prediction. For this research work, we specifically aim at estimating the stock price of Inner Mongolia First Machinery Group Co., Ltd.

[1] Considering the fact that we are sensitive about the correctness of our forecasts, we have decided to include the basic financial data of Open, High, Low, Close, Adjusted Close, and Volume. This data gives us a full insight on the stock's performance during the day. The 'Open' and 'Close' prices are the initial and ending trade prices, respectively, during the day, while 'High' and 'Low' indicate the prices that experienced the uppermost and lowermost extents. 'Adjusted Close' corrects for various corporate actions such as dividends and stock splits, so that it becomes a more reliable gauge of a stock's intrinsic value. 'Volume' is the tracking of the value of the activities performed in the security market. This is a metrics that track the number of shares that were traded,

and is a measure of the activity of the market. In sum, these tools grant invaluable insights into the dynamics of stock prices and they are very much allied to trading techniques that are quantitative in nature.

Our strategy is to implement DBSCAN, a kind of clustering algorithm which depends on the density of the data to identify clusters within the stock price data. DBScan can see through dense data areas and therefore, able to differentiate the outliers and the noise data. By attaching similar data points to the clusters, it is possible to have a more meaningful representation of the dataset and at the same time, we can improve analysis. An alternative paradigm in the financial market whereby K nearest neighbor (KNN) is applied to cluster the data in order to predict the price of stock. KNN conventional algorithms can be in some cases costly especially when the datasets are large.

By ameliorating both prediction accuracy and efficiency, our technique resolves the problem of traditional KNN algorithms. By the initial object, the number of the data points that are needed to be taken into account by KNN is cut down, therefore the computation times are reduced. On top of that, the clustering step not only does the prediction more accurately by grouping the similar data points together but also allows KNN to produce the prediction for the nearby elements within the cluster. In general, this method presents a good solution for stock price prediction as it wraps and consolidates the features of DBSCAN and KNN while lessening the computational difficulties of conventional KNN algorithms.

LITERATURE REVIEW

[2] The research proposes a novel trend in the forecast of financial time series by integrating the strengths of DBSCAN with Support Vector Regression (SVR). This innovative approach highlights that DBSCAN significantly enhances data quality by clustering similar data points together while effectively reducing the presence of noise, which is crucial when dealing with complex financial datasets. Financial data often contain various anomalies and outliers, making it imperative to have a robust method for data cleaning and preprocessing. Following the data processing phase, the concept of SVR is employed to predict future values. By combining these two powerful techniques, the proposed model harnesses the capabilities of DBSCAN to identify and mitigate the impact of outliers while utilizing the predictive strengths of SVR to achieve greater accuracy and stability in forecasts. This integrated methodology not only demonstrates a marked improvement in forecasting performance but also shows its practical applicability in the realm of financial time series prediction, ultimately providing analysts and investors with a more reliable tool for decision-making in volatile markets. The authors' findings indicate that such an approach could significantly enhance the overall effectiveness of financial forecasting models.

[3]The study meticulously investigates how density-based clustering techniques, particularly DBSCAN (Density-Based Spatial Clustering of Applications with Noise), can be effectively applied to analyze stock market data. The authors have demonstrated that by clustering stock data based on their density, researchers can unveil hidden patterns that may not be immediately apparent through traditional analysis methods. This approach not only aids in detecting anomalies within the data but also contributes to a heightened understanding of financial trends and behaviors. Through the application of this innovative method, the analysis of stock market data becomes more robust, allowing for a clearer interpretation of market dynamics. Consequently, this enhanced understanding leads to improved decision-making processes among investors and analysts alike. The research highlights the efficacy of density-based clustering techniques in the realm of financial data analysis, and it strongly recommends the adoption of such methodologies for further advancements in market prediction and analytical accuracy. By leveraging these techniques, stakeholders can gain deeper insights into market movements and ultimately make more informed investment choices.

[4] This research paper rigorously tests the predictive ability of the k-Nearest Neighbor (KNN) algorithm for discerning stock market trends. Specifically, the authors apply the KNN algorithm to a comprehensive set of historical stock data, aiming to predict future movements in the market with a reasonable degree of accuracy. The paper places particular emphasis on the KNN algorithm's effectiveness in trend classification and prediction, utilizing the principles of similarity based on past data points. This methodology represents a straightforward yet powerful approach to analyzing stock market trends, making it accessible for both novice and experienced traders alike. The results from the study indicate that the KNN algorithm is capable of generating highly accurate predictions, which can be invaluable for traders and financial analysts looking to forecast market behavior. Furthermore, the paper underscores the simplicity and interpretability of the KNN model, which contributes to its appeal in practical applications. However, it also candidly discusses the algorithm's limitations and potential drawbacks, encouraging further exploration into enhancements and refinements that could improve its forecasting capabilities in the dynamic and often unpredictable landscape of stock markets. Overall, this research provides insightful contributions to the field of financial analytics, paving the way for future studies to build upon

its findings.

[5] The paper thoroughly discusses the complex task of predicting stock closing prices through the innovative application of machine learning techniques. Several algorithms are explored in depth, including linear regression, support vector machines, decision trees, and neural networks. Each method is analyzed for its strengths and weaknesses in capturing the intricate market behaviors and trends that influence stock prices. The authors emphasize the critical importance of data preprocessing, as well as the necessity of feature engineering to ensure that the models can accurately interpret and leverage the information available. They also stress the significance of proper model evaluation to assess the effectiveness of each algorithm. By demonstrating the performance of various machine learning models, the study convincingly proves that these advanced techniques can significantly enhance the accuracy of stock price forecasting. Furthermore, it highlights potential improvements that could be made to each method, aiming to better manage market volatility and instability, ultimately contributing to more reliable investment strategies and informed decision-making for investors. This comprehensive approach not only reinforces the value of machine learning in finance but also sets the stage for future research and development in this dynamic field.

[6]The paper discusses applying a variety of machine learning algorithms, including linear regression, decision trees, support vector machines, and neural networks, for the purpose of predicting stock prices. It emphasizes the critical importance of data cleaning and feature selection, along with the careful choice of estimation criteria, as these factors significantly contribute to improving the efficiency and accuracy of predictions. By comparing these diverse approaches, the study provides compelling evidence that machine learning can yield more reliable and precise forecasts in the stock market, even amidst the challenges posed by an extremely volatile market and complex non-linear patterns. Furthermore, the paper makes several suggestions for enhancement, such as the implementation of hybrid models that combine different algorithms for better performance and the integration of real-time data analysis, which can help in making stock forecasting more robust and pragmatic. This holistic approach not only underscores the potential of machine learning in financial markets but also encourages continued exploration of innovative methods to refine prediction techniques and adapt to the ever-changing dynamics of the stock market.

[7] The study delves into a comprehensive exploration of a variety of machine learning algorithms that are specifically utilized for predicting stock prices. Among these methods are linear regression, support vector machines, decision trees, and neural networks, each of which has its own strengths and weaknesses in the context of financial forecasting. The research emphasizes the critical importance of data preprocessing, which involves cleaning and organizing the data to enhance the model's performance. Additionally, it highlights the role of feature engineering in selecting and transforming variables to improve predictive power. Evaluation metrics are also discussed as essential tools for assessing the accuracy and reliability of predictions generated by different models. Furthermore, the research conducts a comparative analysis of various algorithms, showcasing their relative effectiveness in managing market volatility and capturing non-linear trends inherent in stock price movements. It identifies the unique capabilities of each model and how they can be leveraged to navigate the complexities of financial markets. To further advance the field, the study suggests potential future enhancements, such as the development of hybrid models that combine the strengths of multiple algorithms and the incorporation of real-time predictive capabilities, which could significantly improve stock price forecasting and provide investors with more accurate information for decision-making.

[8] The text discusses the application of various machine learning algorithms, including linear regression, decision trees, support vector machines, and neural networks, for the purpose of predicting stock prices. It highlights the critical importance of data cleaning and feature selection, emphasizing that these processes are vital for enhancing the accuracy and efficiency of predictions. Additionally, the proper selection of estimation criteria plays a significant role in optimizing the performance of these algorithms. By comparing the effectiveness of these different approaches, the text provides compelling evidence that machine learning can yield more reliable and accurate predictions in the stock market, even amidst the challenges posed by an extremely volatile environment and the presence of non-linear patterns. Furthermore, it makes several suggestions for enhancing predictive models, such as incorporating hybrid models that combine the strengths of various algorithms and leveraging real-time data analysis. These enhancements aim to make stock forecasting not only more robust but also more pragmatic, allowing investors to make informed decisions based on the latest available information. Overall, the discussion underscores the transformative potential of machine learning in the realm of financial forecasting and investment strategies.

[9] The text provides an insightful overview of the Nearest Neighbor (NN) algorithm, exploring its fundamental principles and diverse applications in both learning and classification tasks. It elaborates on the NN principle of data classification, which is based on the concept of proximity to training examples, thus allowing for intuitive decision-making. Additionally, the discussion highlights the evolution of the original NN method into the more sophisticated K-Nearest Neighbor (KNN) algorithm, which aims to enhance accuracy and reliability in classification tasks. Several important enhancements are noted, including the optimization of distance metrics,

the importance of pre-processing data for better results, and techniques for dimensionality reduction that help simplify complex datasets. Moreover, the implementation of efficient data structures such as KD-trees and Ball-Trees is emphasized, showcasing how these tools significantly enhance the scalability of the algorithm. However, the review also identifies critical gaps, including the lack of adaptive algorithms that can effectively handle real-time data streams. This presents a compelling avenue for future research aimed at improving the NN framework to achieve better performance and efficiency in dynamic environments, thereby expanding its applicability in various fields.

[10] The text focuses on the application of the K-Nearest Neighbor (KNN) algorithm in predicting stock market trends, which has become increasingly important in today's fast-paced financial environment. It emphasizes the need for highly accurate forecasts to effectively support finance and risk management decisions, where even minor inaccuracies can lead to significant financial repercussions. The discussion transitions from classical methods of making predictions to KNN, which stands out due to its ability to solve complex, non-linear data patterns by utilizing a voting mechanism based on the proximity of historical data points. Furthermore, the text highlights the importance of optimal parameter selection and the choice of distance metrics, which are crucial for enhancing the practical application of KNN in real-world scenarios. Additionally, it addresses some challenges, such as computational efficiency and issues related to processing real-time data, which can hinder the algorithm's performance. A comparative analysis demonstrates the effectiveness of KNN in predicting market trends, showcasing its advantages over traditional methods. The text concludes with recommendations for further improvements, emphasizing the importance of feature selection and dimensionality reduction techniques to enhance the algorithm's predictive power and overall efficiency. By leveraging these strategies, practitioners can significantly improve the accuracy and reliability of stock market predictions.

[11] The focus of this research is on the enhancement of the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm specifically for location prediction purposes. This contribution significantly advances the field of spatial clustering, particularly within the realms of Geographic Information Systems (GIS), urban planning, and various location-based services. The study elaborates on the unique advantages of DBSCAN, particularly its ability to excel in detecting clusters of varying shapes while effectively handling noise within the data. The paper meticulously details the development of an optimized DBSCAN framework, which incorporates advanced tuning parameters and more refined distance metrics. These enhancements are aimed at increasing both the accuracy and efficiency of clustering, particularly in situations characterized by scale variations and sensitivity to noise. We provide a thorough comparative analysis that demonstrates how our improved DBSCAN method outperforms traditional clustering techniques. In the final section, the paper offers insightful remarks on where future research opportunities exist and outlines potential applications that could further enhance spatial clustering methodologies for location prediction. By addressing these critical areas, we aim to contribute to the ongoing evolution of data analysis techniques in various fields related to urban and regional studies.

[12] The paper introduces a refined K-Nearest Neighbor (KNN) algorithm specifically designed to predict ad hoc carpooling opportunities. This innovative approach addresses the inherent challenges associated with dynamic carpooling systems, where user preferences and availability can change rapidly and unpredictably. By enhancing the traditional KNN framework through several key improvements—namely, parameter selection optimization, feature weighting, and distance metric adjustment—the authors assert that their method achieves higher accuracy and efficiency in real-time carpooling predictions. The research demonstrates that this enhanced KNN not only outperforms the standard version of KNN but also surpasses other baseline models in terms of both accuracy and computational efficiency. The findings suggest that such an improved KNN has the potential to be effectively integrated into real-time carpooling systems, leading to significant enhancements in urban mobility. Ultimately, this advancement could play a crucial role in curbing traffic congestion in densely populated areas, encouraging more individuals to participate in carpooling initiatives, and thus contributing to a more sustainable urban transportation ecosystem.

[13] This paper presents an advanced KNN algorithm specifically designed for stock price forecasting, addressing the intricate challenges posed by financial markets. The research meticulously identifies the weaknesses inherent in traditional stock prediction methods, which often rely on simplistic models that fail to capture the multifaceted nature of market dynamics. By demonstrating how the adoption of more sophisticated machine learning techniques can effectively manage these complexities, the study emphasizes the importance of innovation in financial forecasting. The enhanced model incorporates optimal parameters, appropriate distance metrics, and high-level feature engineering, all of which significantly contribute to improving both the accuracy and efficiency of stock price predictions. A thorough comparative analysis reveals that this improved KNN outperforms not only the conventional KNN but also a variety of other machine learning algorithms in the realm of stock price prediction, particularly in its ability to capture market volatility and non-linearity. Ultimately, the paper concludes that this advanced KNN algorithm has the potential to provide investors and financial analysts with greater reliability and validity in their predictions, thereby enhancing decision-making processes and investment strategies in an increasingly unpredictable market environment.

METHODOLOGY

Traditional K-Nearest Neighbors

The K-Nearest Neighbors (KNN) algorithm is a straightforward, non-parametric, lazy learning method widely utilized for tackling both regression and classification problems. Its core principle is based on the idea that similar data points residing within the same feature space are typically located in close proximity to each other. Essentially, the classification of a new data point hinges on the majority class among its nearest neighbors, which directly influences its predicted classification. A notable advantage of KNN is that it does not make any assumptions about the underlying data distribution; therefore, this learning method can be effectively applied to a diverse array of data formats, including both structured and unstructured data.

However, one of the drawbacks of KNN is the necessity to compute the distance between the test points and all training points. This requirement can lead to significant computational costs, particularly when dealing with very large datasets. To elaborate, the primary process involves calculating the distance between each sample (denoted as Ai) in the testing set (T E = A1, A2, ..., Am) and every sample in the training set. After determining these distances, the k training samples with the shortest distances are selected as the closest neighbors for the testing sample. Once these k neighbors have been identified, they each vote on the class label for the testing sample, and the category that receives the most votes is chosen as the predicted outcome. This voting mechanism is a key feature of KNN, as it emphasizes the influence of the local neighborhood on the classification decision, making it an intuitive yet powerful algorithm in various applications.

Nevertheless, KNN has its limitations, particularly the need to compute the distance between the test points and all training points, which can become computationally expensive for large datasets. This computational burden arises from the fact that the algorithm must calculate the distance for every sample in the test set against all samples in the training set. For example, given a testing set TE=A1,A2,...,AmT_E = A_1, A_2, ..., A_mTE=A1,A2,...,Am, the distance between each test sample AiA_iAi and all training samples is computed. Afterward, the kkk nearest neighbors are selected based on the shortest distances, and each of these neighbors votes on the classification of the test sample. The class with the most votes is then assigned as the predicted label. While this voting mechanism leverages local neighborhood information to make decisions, it also means that KNN is sensitive to noisy or irrelevant features, which can reduce its accuracy in certain cases. Despite these challenges, KNN remains popular due to its simplicity and effectiveness in a wide range of practical applications.

The steps that are followed in the traditional KNN algorithm for a given Training set T = A, b Where every A = (A1, A2, . . . An) with p attributes and b = (b1, b2, ...bn) be the set of labels. As we have continuous labels we use a traditional KNN regressor.

Step 1: Determine the Distance

For each testing sample Ai in the test set T E and each sample Aj in the training set T, calculate the distance Mi = (mi1, mi2, . . . , min). The Euclidean distance is typically used, as shown in Equation 1:

$$m_{ij} = \sqrt{(A_i^T - A_j^T)^2} = \sqrt{\sum_{l=1}^{p} \left(A_{il}^T - A_{jl}^T\right)^2} \tag{1}$$

Step 2: Sort the Distances

Sort the distance set Mi = (mi1, mi2, . . . , min) in ascending order, and record their indices in Mi index = {ji1, ji2, . . . , jin}.

Step 3: Select the k Nearest Neighbors

Select the top k samples that are closest to Ai , i.e., the training samples indexed by {ji1, ji2, . . . , jik}. Then, gather their category labels bik = {bji1 , bji2 , . . . , bjik }.

Step 4: Compute the Final Value

Calculate the average of the values obtained in bik, and assign that value as the final value bi for Ai .

Density-Based Spatial Clustering Of Application With Noise

DBSCAN, which stands for Density-Based Spatial Clustering of Applications with Noise, is a powerful statistical technique that leverages the density of elements within a specific geographical space to identify and form clusters present in a dataset. Unlike K-means, a clustering method that requires the number of clusters to be predetermined, DBSCAN operates without this prerequisite, allowing the algorithm to dynamically determine the number of clusters as it analyzes the data. This flexibility makes DBSCAN particularly effective for identifying clusters of varying shapes and sizes, as well as for pinpointing outliers—data points that do not conform to any established group. Additionally, DBSCAN excels in discovering the densest clusters within the dataset, which can be invaluable for various applications, such as anomaly detection in large datasets.

This clustering process is categorized as an unsupervised learning technique, meaning it does not require labeled input data. The core principle underlying DBSCAN is the preservation of connectivity between samples based on density. This allows the algorithm to explore the potential for additional reachable samples to enhance connectivity even further. If a sample exists within the epsilon (eps) neighborhood of the current sample, it is considered directly density accessible to that sample. Furthermore, samples that can be reached through transitive relations formed by a series of directly density reachable samples are referred to as density reachable samples. This intricate approach enables DBSCAN to effectively navigate complex datasets, making it a valuable tool in data analysis and machine learning.

One of the major advantages of DBSCAN is its ability to handle noise in the data. By classifying data points that do not belong to any cluster as noise, DBSCAN effectively filters out outliers, improving the overall quality of the clustering. This is particularly useful in real-world applications such as geographic data analysis, image processing, and market segmentation, where noisy data can often obscure meaningful patterns. Moreover, DBSCAN is scalable to large datasets, making it an ideal choice for big data scenarios where computational efficiency is key. Its non-linear approach to clustering allows for more nuanced groupings, which is difficult to achieve with traditional methods like K-means.

Step 1: Identify Required Parameters for the DBSCAN Algorithm

- Epsilon (ϵ): It establishes the boundaries of a data point's neighborhood. Two points are considered to be nearby if their separation is less than or equal to ϵ. If ϵ is set too low, a large portion of the data will be considered anomalies. Conversely, if ϵ is too high, clusters will merge, and most data points will belong to the same cluster. The k-distance graph can be used to determine an appropriate ϵ value.

- MinPts: The minimum number of neighbors (data points) required within an ϵ radius. For larger datasets, a larger value of MinPts is required. Generally, MinPts should be greater than or equal to D + 1, where D is the number of dimensions in the dataset.

$$\mathrm{MinPts} \geq D + 1 \tag{2}$$

MinPts must also be at least 3.

Step 2: Core Point Identification

For each point in the dataset, count the number of points within its ϵ-neighborhood. A point is a core point if the number of neighbors is greater than or equal to MinPts. Mathematically, this can be expressed as:

$$|N_\epsilon(p)| \geq \mathrm{MinPts} \tag{3}$$

where |Nϵ(p)| represents the number of points within the ϵ-neighborhood of point p

Step 3: Cluster Formation

For each core point, if it is reachable by another core point, assign it to the same cluster. For each core point and its corresponding cluster, find all points that are density-reachable from these core points. Repeat this process iteratively until all points are assigned to a cluster:

$$\mathrm{Cluster}(p) = \mathrm{Cluster}(q) \text{ if } p \text{ is density-reachable from } q \tag{4}$$

Step 4: Noise Detection

Points that are neither core points nor members of any cluster are considered noise. Additionally, a point is considered noise if it is within ϵ-distance of a core point but does not have enough neighbors to form a cluster:

$$\text{Noise} = \{p \mid p \text{ is not a core point and not part of any cluster}\} \tag{5}$$

Step 5: Cluster Labels

The output of DBSCAN includes a set of clusters and a set of noise points. Add a new column in the database as Cluster to identify which cluster each data point belongs to:

$$\text{Cluster}(p) = \text{label assigned to } p \tag{6}$$

RESULTS AND DISCUSSION

The KNN (K-Nearest Neighbors) and DBSCAN (Density-Based Spatial Clustering of Applications with Noise) are two algorithms that the hybrid model derives significant advantages from. DBSCAN is particularly effective in identifying clusters within similar data points, allowing the resulting dataset to include an additional column that clearly indicates the cluster to which each data point belongs. This clustering technique is beneficial because it can effectively reduce the size of the dataset, leading to quicker response times. Moreover, by focusing on individual clusters, the Root Mean Square Error (RMSE) in predictions can be minimized when KNN is executed solely on the relevant clusters of data. This selective approach not only enhances the model's efficiency but also improves its predictive accuracy. The integration of these two algorithms allows the hybrid model to leverage the strengths of both clustering and nearest neighbor techniques.

The hybrid model of DBSCAN and KNN overcomes some of the most important hurdles in stock price prediction. Anatomy of almost all anticipated stock price methods utilizes impute features of the data therefore, DBSCAN is good here as it is able to filter irrelevant or abnormal extreme conditions that tend to distort conventional expectations of forecast. Using well-designed and precise filters makes it possible for DBSCAN to help build cleaner surfaces on which KNN will operate. Analysis of time series data is also possible in this approach into this study, standards and averages within stock groups can also be established. Generally, the KNN will have a high level of precision when predicting trends since the clusters make sure that such trends are eminent in a particular context. The increase in speed of this hybrid method is also worth mentioning since the time needed for large amounts of financial data to be processed quicker is crucial in trading systems that operate in real time.

By combining the noise-reduction and pattern-recognition capabilities of DBSCAN with the adaptive, instance-based learning of KNN, this hybrid model represents a powerful tool for navigating the complexities of stock market prediction, offering both improved accuracy and enhanced interpretability of results.

The detailed steps for building this hybrid model involve careful data preprocessing, appropriate parameter tuning for both algorithms, and systematic evaluation of performance metrics to ensure optimal results:

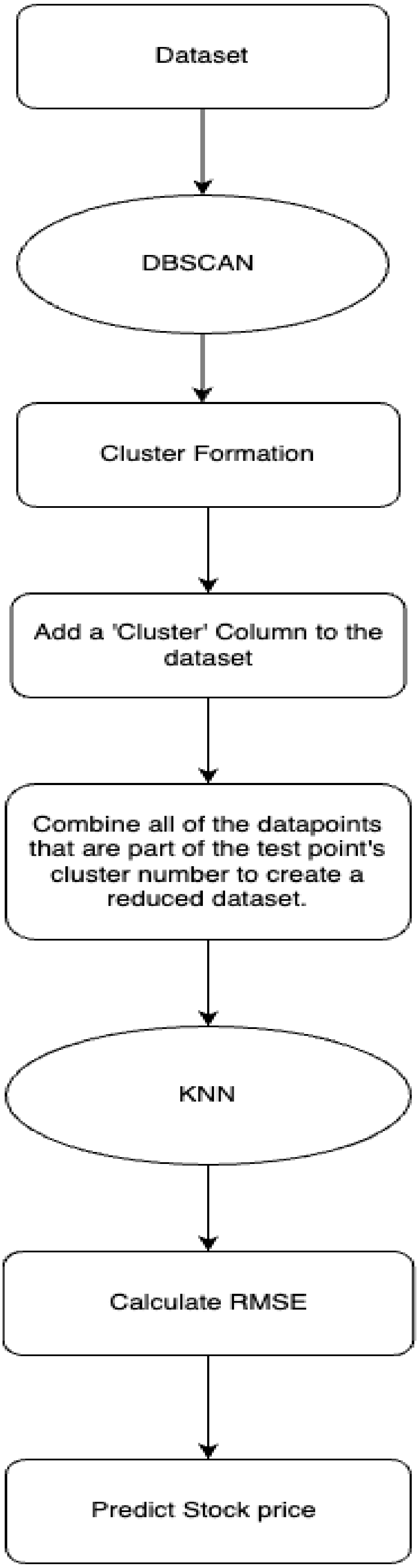

Dataset
DBSCAN
Cluster Formation
Add a 'Cluster' Column to the dataset
Combine all of the datapoints that are part of the test point's cluster number to create a reduced dataset.
KNN
Calculate RMSE
Predict Stock price

Fig. 1 Workflow of the Hybrid Model

Step 1: Apply DBSCAN to Cluster Data

Start with the original dataset containing features Open, High, Low, Adj Close, Volume and target variable Close. Then perform DBSCAN on the dataset using appropriate ’eps’ (epsilon) and ’min samples’ values that were found using GridSearchCV. DBSCAN will group similar data points into clusters based on density and label the noise points as outliers (labeled as -1). The dataset will now be partitioned into clusters.

Step 2: Add a Cluster Column to the Dataset

The dataset had already been divided into clusters by DBSCAN, and then we had a new column ’Cluster’ added to the dataset which includes the cluster label of each data point.The new dataset that the cluster model applied ’DBSCAN’ to all the data points and has all the features of the original dataset and a new column of ’Cluster’ that is depending on which cluster the data point belongs to.

Step 3: Reduce the Dataset Based on the Cluster Number

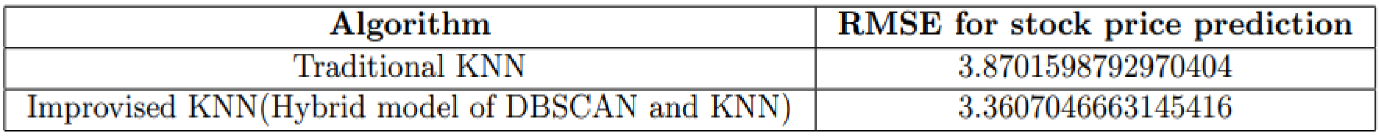
For a given test data point A, KNN can be used to identify which of its neighbors are in the same cluster. But the computation is only done for the relevant cluster. The procedure begins by finding the cluster number of the test data point A and next, you should specify and filter the dataset to strictly include only the samples from that cluster. Hence, the dataset, which is a lesser number of points, contains the information gained from a particular cluster.

Step 4: Apply KNN on the reduced Dataset

Use the reduced dataset obtained after Step3 to apply the KNN algorithm. With the reduced number of data points, KNN is computationally less expensive and can make predictions faster. The prediction is more accurate since it is limited to a more relevant subset of data, potentially reducing the Root Mean Squared Error (RMSE).

The results of our comprehensive study provide compelling evidence for the efficient working of our innovative hybrid model when compared to traditional algorithms in the realm of stock price prediction. By strategically reducing the dataset size through DBSCAN clustering prior to the application of the KNN algorithm, we achieved significantly quicker predictions without sacrificing accuracy. This innovative approach facilitates a more focused and nuanced analysis by concentrating on specific, relevant clusters, which consequently leads to a substantial reduction in Root Mean Square Error (RMSE) prediction errors. The effectiveness of our hybrid model is particularly pronounced when confronted with large, heterogeneous datasets that are characterized by varying densities—a common challenge faced in financial data analysis. Our DBSCAN-KNN hybrid system not only demonstrates remarkable efficiency but also superior accuracy in its predictions, consistently outperforming traditional algorithms that have long been used in the industry.

This improvement is not merely incremental; it signifies a paradigm shift in how we approach stock market forecasting in an era where data is abundant yet complex. By leveraging the strengths of both clustering and nearest neighbor algorithms, our model adeptly navigates the complexities of financial data with greater agility and precision. The ability to quickly process and analyze vast amounts of data while simultaneously maintaining high predictive accuracy addresses a critical need in the fast-paced world of financial markets. Furthermore, this hybrid approach opens up exciting new possibilities for real-time analysis and decision-making in stock trading, potentially revolutionizing how investors and financial institutions operate within the market landscape. As we look to the future, the implications of our findings may extend beyond mere profitability, influencing how financial strategies are formulated and executed in an increasingly data-driven world.



Algorithm	RMSE for stock price prediction
Traditional KNN	3.8701598792970404
Improvised KNN(Hybrid model of DBSCAN and KNN)	3.3607046663145416

Table 1 Comparison among various prediction models

CONCLUSION

In this study, we leveraged the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm to significantly enhance the performance of the classic K-Nearest Neighbors (KNN) algorithm for stock price prediction. By integrating DBSCAN's unique ability to identify and remove noise points and outliers, we were able to improve the overall quality of the dataset. This data cleaning process not only ensured a more robust model but also paved the way for more precise KNN predictions. The improvements were evident in the substantial increase in prediction accuracy, as demonstrated by a remarkable reduction in the Root Mean Square Error (RMSE). When comparing these results with a previous study conducted on a Chinese dataset, which used KNN in isolation, the hybrid DBSCAN-KNN model outperformed it, achieving the lowest RMSE and more reliable results.

One of the key aspects of our approach was the strategic reduction of data predictability to enhance the efficiency of KNN. By removing noise and outliers, we were able to train the KNN model more effectively, enabling it to process data with greater speed and accuracy. This refinement reduced computational costs, making the model more scalable and applicable to larger datasets. Furthermore, the improved capacity for generalization meant that the model could accurately predict phenomena that had not been previously observed in the training data. The reduction in computational overhead, coupled with increased accuracy, demonstrated the efficacy of pairing a clustering algorithm like DBSCAN with traditional machine learning models like KNN in the realm of financial forecasting.

The integration of cluster-based algorithms and traditional machine learning techniques presents a promising future for stock price prediction. This innovative approach opens up new research avenues, allowing for a deeper exploration of how advanced clustering methods can be combined with established algorithms to yield more accurate and reliable results. The strong correlation observed between DBSCAN and KNN in this study provides valuable insights into how financial markets can be better understood and predicted. With enhanced accuracy and efficiency in predicting stock market trends, this model stands to benefit not only financial analysts but also market participants who rely on precise forecasts to make informed investment decisions. This integration marks a significant step forward in the field of financial forecasting and stock price prediction, highlighting the potential for future advancements in this area.

FUTURE SCOPE

The integration of DBSCAN clustering with the K-Nearest Neighbors (KNN) algorithm has the potential to significantly enhance the framework for predictive stock price analysis. By combining these two methods, it becomes possible to identify patterns and trends in financial data more effectively. This hybrid approach could revolutionize traditional forecasting methods, leading to innovative techniques that offer greater accuracy and insight. One of the primary advantages of this model lies in its ability to handle noisy and high-dimensional data, which is prevalent in stock market datasets. As a result, this integration offers promising opportunities for improving prediction outcomes in both short-term and long-term forecasts.

Future research should focus on optimizing the parameters for both DBSCAN and KNN. Fine-tuning these variables can not only improve the precision of predictions but also significantly reduce the computational demands of the model. This would make the approach more scalable, allowing it to be applied to larger datasets or real-time market predictions. Additionally, studying the suitability of this method across various financial markets and asset classes is crucial. Different market conditions, such as volatility or liquidity, could affect the model's performance. Analyzing its efficacy in different environments would help in determining its applicability and generalizability, ensuring that the model can be adapted to various financial contexts.

Furthermore, exploring alternative machine learning or clustering techniques could provide fresh insights into stock price prediction. Incorporating methods like Random Forests, Support Vector Machines, or even deep learning algorithms could lead to the development of more robust models. The combination of DBSCAN, KNN, and these advanced methods could result in a holistic approach to stock price forecasting. Additionally, integrating findings from other analytical methods, such as sentiment analysis or macroeconomic trend assessments, could create a more comprehensive understanding of stock market behavior. By addressing these key areas, researchers have the opportunity to contribute significantly to the future of financial prediction models, ultimately creating systems that are both more effective and more adaptive to changing market dynamics.

REFERENCES

[1] Yahoo Finance: Historical Stock Data for 600967.SS. Accessed: 2024- 08-29 (2020). https://finance.yahoo.com/quote/600967.SS/history/?period1=1084843800&period2=1593475200&guccounter=2

[2] Huang, M., Bao, Q., Zhang, Y., Feng, W.: A hybrid algorithm for forecasting financial time series data based on dbscan and svr. Information 10, 103 (2019) https://doi.org/10.3390/info10030103

[3] Das, T., Halder, A., Saha, G.: Application of density-based clustering approaches for stock market analysis. Applied Artificial Intelligence 38 (2024) https://doi. org/10.1080/08839514.2024.2321550

[4] Reddy, A.S., Praneeth, M., Reddy, K.P., Srinivas Reddy, A.: Stock market trend prediction using k-nearest neighbor (knn) algorithm. International Journal for Innovative Engineering & Management Research 13(5), 8 (2021). Posted: 6 May 2024

[5] Vijh, M., Chandola, D., Tikkiwal, V.A., Kumar, A.: Stock closing price prediction using machine learning techniques. Procedia Computer Science 167, 599–606 10 458 PranitBari /IJCNIS, 16(4), 449-459 (2020) https://doi.org/10.1016/j.procs.2020.03.326 . International Conference on Computational Intelligence and Data Science

[6] Parmar, I., Agarwal, N., Saxena, S., Arora, R., Gupta, S., Dhiman, H., Chouhan, L.: Stock market prediction using machine learning. In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), pp. 574– 576 (2018). https://doi.org/10.1109/ICSCCC.2018.8703332

[7] Kabir, M.H., Sobur, A., Amin, M.R.: Stock price prediction using the machine learning 11, 946–950 (2023) https://doi.org/10.1729/Journal.37948

[8] Suyal, M., Goyal, P.: A review on analysis of k-nearest neighbor classification machine learning algorithms based on supervised learning. International Journal of Engineering Trends and Technology 70(7), 43–48 (2022) https://doi.org/10. 14445/22315381/IJETT-V70I7P205 . This is an open access article under the CC BY-NC-ND license

[9] Taunk, K., De, S., Verma, S., Swetapadma, A.: A brief review of nearest neighbor algorithm for learning and classification. In: 2019 International Conference on Intelligent Computing and Control Systems (ICCS), pp. 1255–1260 (2019). https: //doi.org/10.1109/ICCS45141.2019.9065747

[10] Reddy, A.S., Praneeth, M., Reddy, K.P., Reddy, A.S.: Stock market trend prediction using k-nearest neighbor (knn) algorithm. International Journal for Innovative Engineering & Management Research 13(5), 1–8 (2021). Posted: 6 May 2024

[11] perumal, M., Velumani, B.: Design and development of a spatial dbscan clustering framework for location prediction- an optimization approach. In: 2018 3rd International Conference on Communication and Electronics Systems (ICCES), pp. 942–947 (2018). https://doi.org/10.1109/CESYS.2018.8724094

[12] Piao, Y.: Ad hoc carpooling prediction based on improved knn. In: Proceedings of the 12th International Symposium on Computational Intelligence and Design (ISCID), p. (2019)

[13] Yunneng, Q.: A new stock price prediction model based on improved knn. In: Proceedings of the 7th International Conference on Information Science and Control Engineering (ICISCE), p. (2020)