# Stock Market Time Series Data Prediction Using Sequence Pattern Mining Algorithms

Xia Liu
School of Data and Information，Changjiang Polytechnic, Wuhan, 430074, China
E-mail: jasmine19840000@163.com

**Keywords:** Sequence pattern mining algorithm; Financial stock market; Time series data; Prediction model; K-line

**Received:** June 4, 2025

*With the continuous improvement of China's financial markets, public investment has gradually shifted toward the stock market. However, stock price fluctuations are influenced by multiple factors, making it difficult for the public to make accurate judgments. To address this, the study proposes a stock market sequence prediction model based on Sequence Pattern Mining algorithms. The model introduces K-line indicator data from the stock market for optimization, forming specific similarity sequences for stocks. By constructing optimal K-line patterns and matching them with stock sequences, the model achieves stock price prediction. Experiments using public datasets show that the proposed model achieves the lowest Mean Squared Error of 18% for K-line data prediction in later iterations. After 500 iterations, the Coefficient of Determination increased from 0.38 to 0.79. The recall rate rose to 88% after 150 iterations. In the analysis of a self-built dataset, the model demonstrated the best predictive performance for the fourth sequence group, with a Mean Absolute Percentage Error of 0.94% and a Root Mean Square Error of 0.95%. Among the prediction accuracy rates for all stocks, the proposed model's accuracy mostly fell within the range of 60%-70%. These results indicate that the proposed model can accurately predict trends in complex financial markets, enhance public confidence in China's stock market, and contribute to the development of the financial industry.*

*Povzetek: Opisan je sistem za napovedovanje časovnih vrst delniškega trga z uporabo inteligentnih tehnologij. Algoritem SPM (Sequence Pattern Mining) je optimiziran s podatki K-linij in metodo PCA in izboljšuje napovedi trendov na kompleksnih finančnih trgih.*

## 1 Introduction

With the increasing popularity of intelligent network technologies, smart finance has become a hot research topic [1]. The surplus of assets held by the public has also made the stock market a key investment avenue for profit. How to accurately predict financial stock market trends using intelligent technologies has become a focus of academic research [2-4]. Currently, traditional stock market prediction models estimate stock prices by evaluating a company's market price and earnings per share, enabling predictions under specific conditions. However, these models cannot effectively account for market sentiment, national policies, or international situations, which often interfere with stock price fluctuations, resulting in low prediction accuracy [5]. Today, many scholars have integrated intelligent technologies with the financial sector. The Sequence Pattern Mining (SPM) algorithm can extract valuable features from various unstructured data and images [6-7]. Meanwhile, K-line data in the stock market, represented by charts based on opening, closing, highest, and lowest prices, accurately reflects stock price fluctuations [8]. K-line sequences not only visually depict price trends but also imply market sentiment, revealing potential stock trends. Based on this, the study introduces K-line data to improve SPM algorithm, constructing a comprehensive prediction model. By generating optimal K-line patterns and matching them with stock sequences, the model predicts future trends. The study aims to create an effective prediction model for stock market time series data, establish a robust prediction system, reduce investors' financial losses, and provide the public with reliable investment strategies.

The study is divided into four parts. The first part summarizes and discusses relevant research on SPM algorithm. The second part introduces K-line data to improve the SPM algorithm, generating similarity sequences that match stock price trends and forming an effective stock market time series prediction model. The third part validates the performance of the improved SPM and the stock market time series prediction model. The fourth part concludes the study.

## 2 Related works

With the continuous improvement of databases, data analysis algorithms based on databases have gradually emerged. Among them, SPM algorithm extracts frequent subsequences from sequence sets to analyze target sequences and predict outcomes. This characteristic of SPM algorithm has attracted significant attention from

scholars worldwide. For example, Huang et al. raised a general target SPM model to improve the analysis efficiency of sequential pattern mining algorithms. They designed multiple pruning methods to effectively reduce redundant operations. Experiments showed that the model significantly accelerated the workflow and reduced memory consumption [9]. Wu et al. optimized the one-off strong pattern mining algorithm using backtracking measurements to enhance recognition effectiveness. Experimental results demonstrated that the optimized algorithm improved analysis efficiency, exhibited excellent mining performance, and could be applied to traffic flow prediction [10]. Djenouri et al. raised a novel parallel pattern mining framework for identifying fake and real news. By generating comparison patterns from fake news and matching them with real news data, they constructed an analysis model. Results proved that the model's accuracy significantly outperformed popular existing models [11]. Li et al. employed depth-first search and backtracking strategies to raise a non-overlapping sequential rule mining model, enhancing the algorithm's ability to generate specific patterns. Experimental results showed that the co-occurrence patterns generated by the model had excellent recommendation performance, surpassing similar algorithms [12]. Abbas raised a specific sequence mining framework to improve SPM's fraud detection capabilities. The framework detected fraudulent behaviors in user logs one by one and added personalized transaction patterns. Experiments confirmed that the model could be applied to online banking systems, accurately identifying evolving fraud tactics [13].

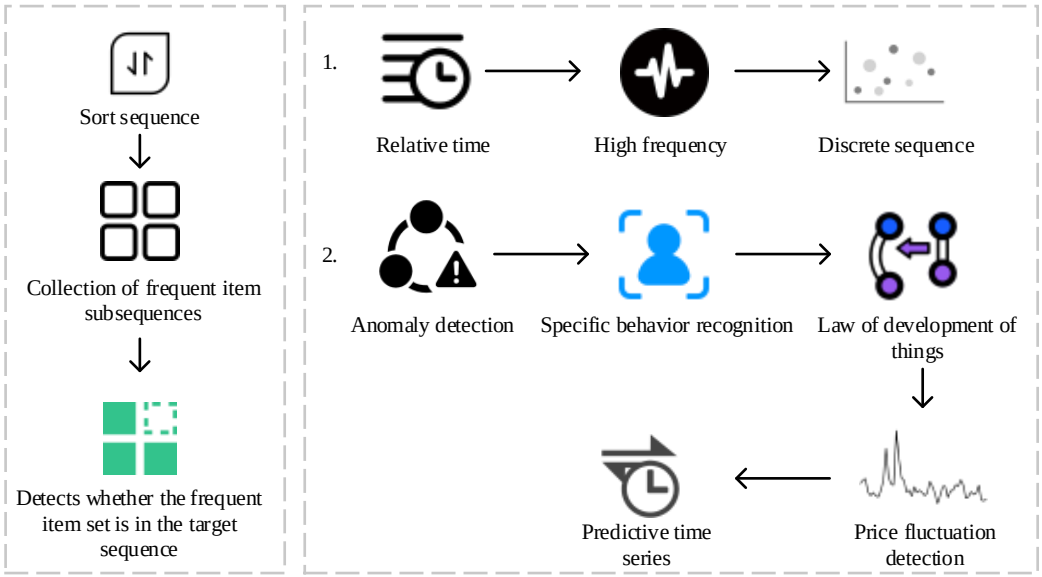
Additionally, with the continuous development of the domestic economy, financial markets have become increasingly prosperous. Scholars worldwide have adopted various cutting-edge technologies to accurately predict trends in the financial industry. For instance, Zheng et al. introduced machine learning models into the financial sector, raising a hybrid model combining convolutional and bidirectional memory neural networks. They optimized its ability to capture nonlinear data from dynamic indicators. Experiments validated the model's predictive capabilities in financial markets, providing theoretical support for economic forecasting [14]. Kumar et al. found that time series prediction models could effectively analyze stock market time series data. They introduced a differential evolution algorithm combined with an artificial bee colony algorithm. Results proved that the model could analyze multi-step time series trends and achieved higher prediction accuracy than similar algorithms [15]. Sisodia et al. raised a stock price prediction model based on long short-term memory neural networks. Using a decade-long historical stock price dataset from India, they compared the model's performance, achieving an accuracy of 83.88%, outperforming comparative models [16]. Kirisci et al. applied convolutional neural networks to financial data prediction. Their model, consisting of three convolutional layers and five fully connected layers, effectively clarified the relationship between input and output values. Results showed that the model's predictive capabilities surpassed those of similar models [17]. Mukherjee et al. raised a convolutional neural network-based model to address the impact of specific events on stock market prices. They created histograms from datasets within specific timeframes. Analyzing the impact of the COVID-19 pandemic on stock markets, the model achieved a prediction accuracy of 91% [18].

In summary, existing research on SPM algorithm still faces challenges such as insufficient prediction accuracy, low operational efficiency, and limited prediction types. This study focuses on addressing these issues by introducing SPM algorithm into the financial stock market for in-depth research. By integrating K-line data unique to stock markets, the study aims to optimize the algorithm and create a robust stock investment environment.

# 3 Stock market time prediction model based on SPM algorithms

## 2.1 Optimization design of SPM algorithm with K-line data

SPM algorithm can extract valuable information from various unstructured data and images, enabling predictions for target objects [19]. Many scholars have applied this algorithm to the financial sector, analyzing transaction data from multiple stocks and exchange rate fluctuations to uncover potential trading patterns and trends [20]. The operational mechanism and application scenarios of SPM are shown in Figure 1.




(a) Mechanisms of operation of SPM (b) Application scenarios of SPM

Figure 1: The operation mechanism and application scenarios of the SPM algorithm

As shown in Figure 1(a), the SPM algorithm arranges elements within sequences to extract frequent subsequences and detects frequent sets in target sequences. From Figure 1(b), the algorithm performs best on discrete sequences, primarily mining sequences with high relative time and frequency. It can also be used for anomaly detection, identifying specific features, analyzing patterns of change, detecting price data variations, and predicting time series. The traditional SPM mines frequent sequences by strategy, but it cannot be applied in time-series continuous features in finance. Therefore, the study introduces sliding window dynamic alignment and multi-scale feature fusion mechanism to optimize the processing. The study first preprocesses the original data, splits the data into multiple subsequences, and zero-mean unit variance is applied to each feature of the subsequence, while the absolute ups and downs in the subsequence are expressed as shown in Equation (1).

$$A=(P_{t+1}-P_t)/P_t \quad (1)$$

In Equation (1), $A$ represents the absolute price change, $P_{t+1}$ denotes the subsequence price at time $t+1$, and $P_t$ denotes the subsequence price at time $t$. Next, the raw time series is transformed, and the time difference between two endpoints is expressed as Equation (2).

$$T_s=T_{t+1}-T_t \quad (2)$$

In Equation (2), $T_s$ represents the time difference between adjacent extremes, $T_{t+1}$ denotes the specific time at $t+1$, and $T_t$ denotes the specific time at $t$. After preprocessing the raw data, the study used a sliding-alignment similarity matching method to compare the input sequences with a predefined sequence of K-line patterns, as shown in Equation (3).

$$S(Z,U)=\begin{cases}\underset{1\le t\le m-n+1}{Max}\sum_{i=1}^{n}sim(Z_{t+i-1},U_i), m\ge n\\ \frac{1}{n-m+1}\underset{1\le t\le n-m+1}{Max}\sum_{i=1}^{m}sim(Z_i,U_{t+i-1}), m<n\end{cases} \quad (3)$$

In Equation (3), $S(Z,U)$ is the degree of matching under optimal alignment, $Z$ is the input sequence, $U$ is the K-line pattern sequence, $sim$ is the similarity function, $Max$ is the maximum value in the set, $m$ is the length of the sequence $Z$, $n$ and is the length of the sequence $U$. Using this method overcomes the problem that the traditional Euclidean distance is sensitive to the length of the sequence and is more applicable to the matching process of financial time series data. After defining the SPM's operational mechanism, the study introduces K-line data to optimize the design process of SPM algorithm as shown in Figure 2.

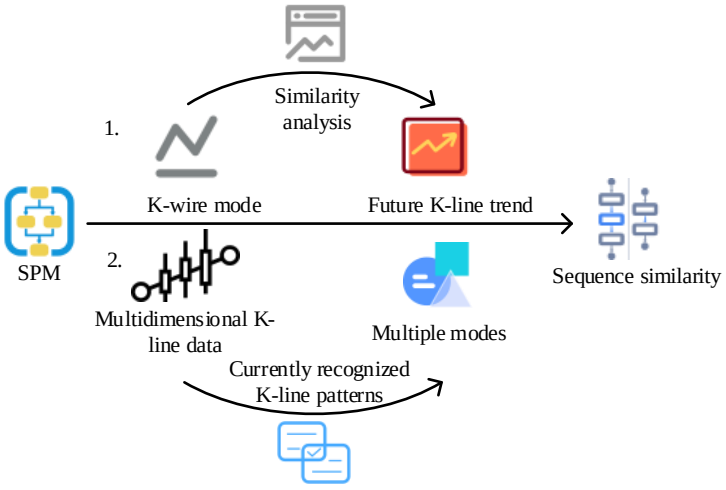



Figure 2: Diagram of the optimization process of the algorithm based on K-line data

As shown in Figure 2, the SPM algorithm reads historical K-line data to construct specific patterns. It then compares multiple generated K-line patterns with predicted future K-line trends for correlation analysis. Additionally, it matches multiple historical K-line sequences with the currently identified K-line patterns to construct feasible similarity sequences. The confidence of the input historical data in the SPM algorithm is calculated by measuring specific preconditions in association rules to determine the probability of subsequent conditions, as shown in Equation (4).

$$C(A \Rightarrow B) = P(B|A) = \frac{support\_cout(A \cup B)}{support\_cout(A)} \quad (4)$$

In Equation (4), $A$ and $B$ are two subsets of itemsets under different conditions, and $support\_cout$ is the number of transactions. When frequent itemsets are identified in both subsets, strong association rules are generated, as expressed in Equation (5).

$$\frac{support\_cout(l)}{support\_cout(s)} \geq \min\_conf \quad (5)$$

In Equation (5), $l$ represents the frequent itemset, $s$ is each non-empty subset of the frequent itemset, and $\min\_conf$ is the minimum confidence threshold. When $s$ meets or exceeds the minimum confidence threshold, the output is expressed as Equation (6).

$$s \Rightarrow (l - s) \quad (6)$$

In Equation (6), once the frequent itemset is generated, the corresponding specific K-line pattern is obtained. The operational process of the improved SPM algorithm with K-line data is shown in Figure 3.

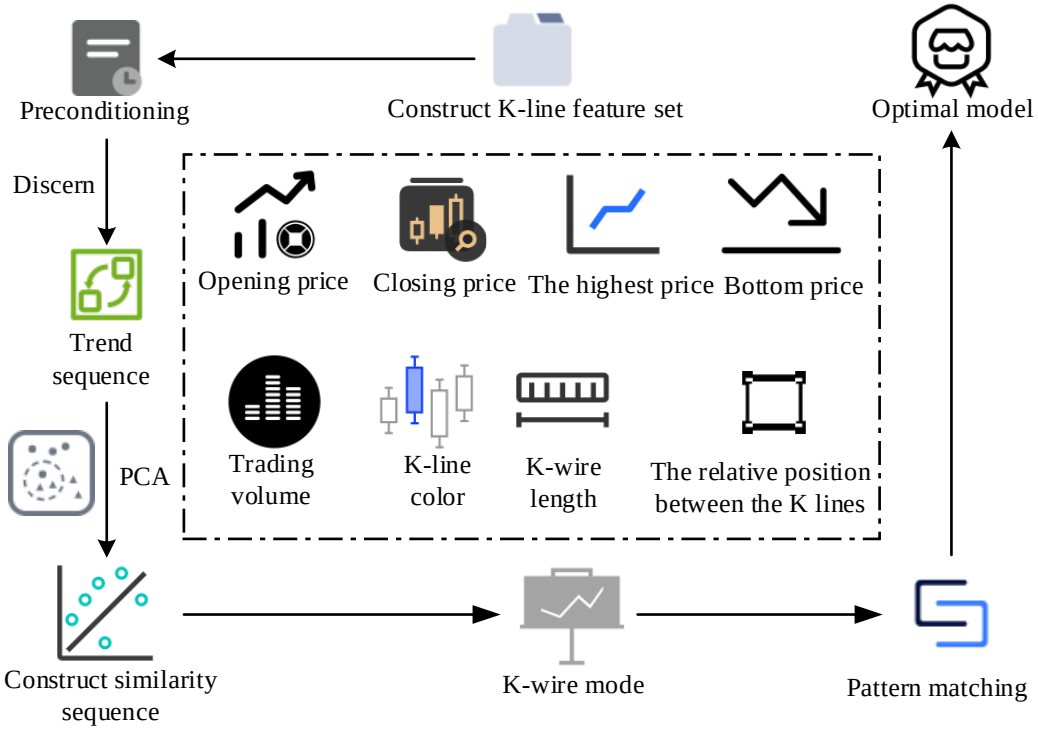



Figure 3: The flow chart of SPM algorithm operation is improved

As shown in Figure 3, the study constructs a K-line feature set, including daily opening price, closing price, highest price, lowest price, trading volume, K-line color, K-line length, and relative length between K-lines. These eight specific features are preprocessed to identify sequences with specific trends. Using Principal Component Analysis (PCA), a new similarity sequence is constructed, forming a complete K-line pattern. This pattern is matched with the current sequence to optimize and obtain the best K-line pattern.

## 2.2 Construction of stock market time prediction model using sequence mining algorithms

The study improves the SPM algorithm by introducing K-line data and applying PCA to extract principal components, constructing new similarity sequences to generate optimal K-line patterns. To apply the optimized algorithm in stock market scenarios, a targeted prediction model is built. First, the time series data in the stock market is defined, and related influencing factors are listed, as shown in Figure 4 [21].

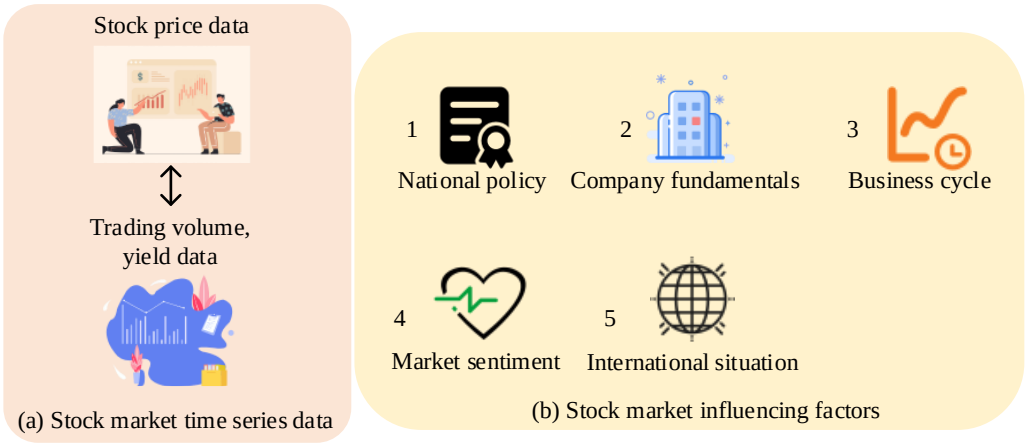



Figure 4: Schematic diagram of the impact of related factors on time series data

As shown in Figure 4, stock market time series data include stock price data, trading volume, and return data. The stock market is influenced by factors such as company fundamentals, national policies, international situations, market sentiment, and economic cycles. For example, continuous corporate profitability and high public expectations can drive stock prices up, while poor corporate performance, low public expectations, and unfavorable international situations can lead to declines. Stock market changes often exhibit nonlinear and complex curves. On different trading days, stock trends are unpredictable due to market fluctuations. The definition of stock price changes is expressed as Equation (7).

$$l(x) = -\log_2 p(x_i) \quad (7)$$

In Equation (7), $l(x)$ represents the price change value, and $p(x_i)$ is the probability of a specific price change category on a trading day for the $i$-th stock. The entropy of price changes on a trading day is expressed as Equation (8) [22].

$$H = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) \quad (8)$$

In Equation (8), $H$ represents the entropy of price changes on a trading day. A higher entropy value indicates greater uncertainty in the stock market [23-24]. To address the impact of uncertainties on the stock market, the study uses PCA to process stock market data, as shown in Figure 5.

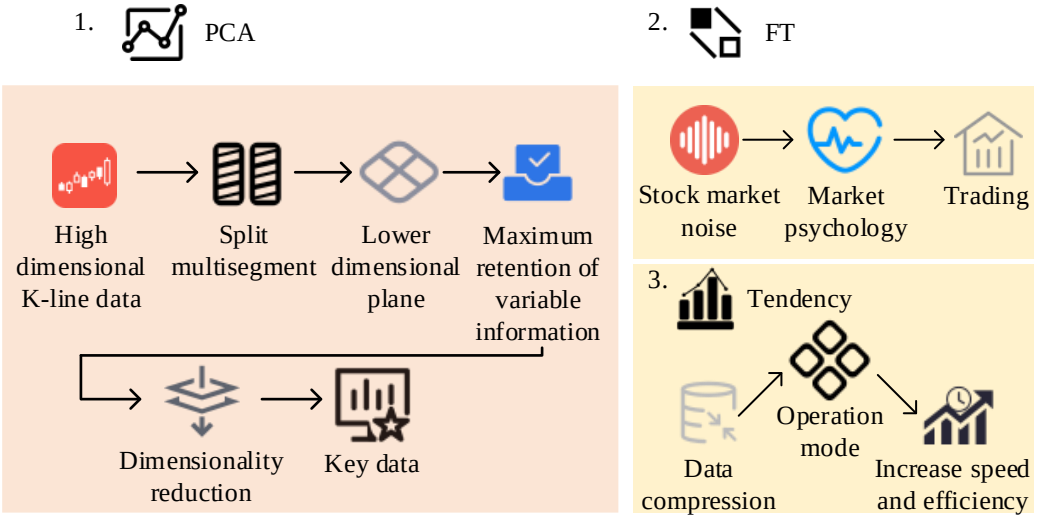



Figure 5: Construct a schematic diagram of how to process stock market data

As shown in Figure 5, PCA divides high-dimensional K-line data into multiple subsequences and analyzes the data on a low-dimensional plane. During dimensionality reduction, key principle components are retained, replacing numerous original variables while preserving most of the information. To mitigate noise from market psychology and multi-party trading factors, the study introduces Fourier Transform (FT) to remove noise. Compressed stock market data accelerates the prediction process, improving model speed and efficiency. After defining the preprocessing steps, the study builds a prediction model for future stock trends. First, historical stock data is collected to record specific K-line indicators, as expressed in Equation (9).

$$J = \left\{ J_{t_1}, J_{t_2}, \ldots J_{t_i} \ldots \right\} \quad (9)$$

In Equation (9), $t_i$ represents time, and $J_{t_i}$ denotes all K-line indicators for the $t_i$-th day. The K-line indicator set for a specific day is expressed as Equation (10).

$$J_{ti} = \{t_i, A_1, ...A_m ...A_n\} \quad (10)$$

In Equation (10), $n$ represents the total number of stocks, and $A_m$ denotes the K-line feature set for stock $m$ at the current time. To analyze periodic patterns in stock market time series and remove noise, Fourier Transform is applied, as expressed in Equation (11).

$$\hat{f}(\xi) = \int_{R^n} f(x) e^{-i(\xi \cdot x)} dx \quad (11)$$

In Equation (11), $i$ represents the imaginary unit, and $\hat{f}(\xi)$ denotes the Fourier Transform function. The study compares the proposed model with other models using the same dataset, calculating the average accuracy as expressed in Equation (12).

$$A = \frac{(\sum_{i=1}^{k} P_{A_i})}{k} \quad (12)$$

In Equation (12), $P_{A_i}$ represents the prediction probability for the SS-th stock, and DD is the total number of stocks in the experiment. After defining the accuracy comparison method, the complete stock market prediction model is built, as shown in Figure 6.

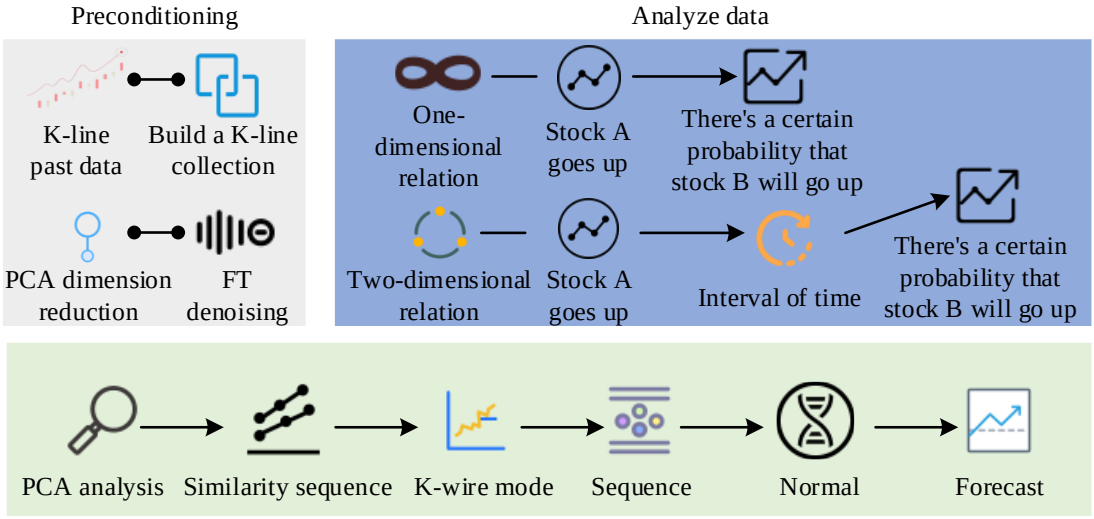


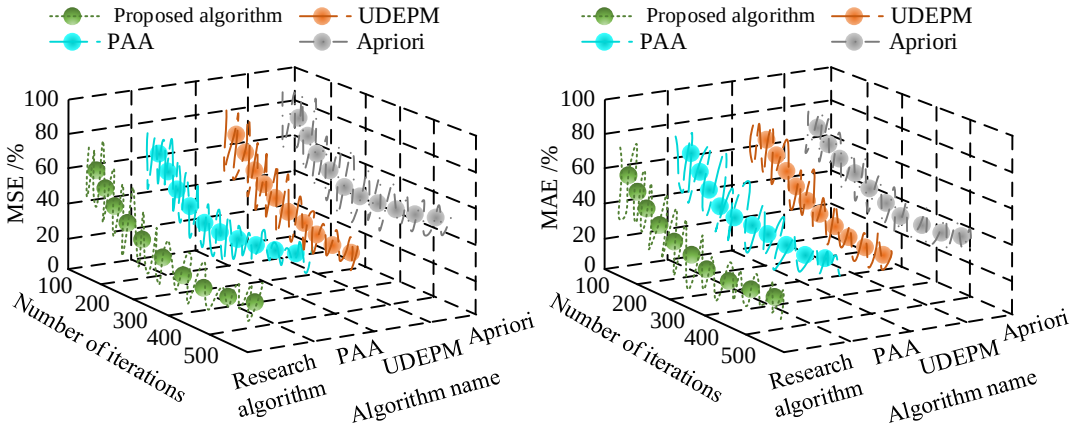

Figure 6: Schematic diagram of a complete stock market forecasting model

As shown in Figure 6, the stock market time prediction model consists of three parts. The first part aggregates historical K-line data, constructs a K-line dataset, and preprocesses the data using PCA and FT for dimensionality reduction and noise removal. The second part divides sequence relationships based on association rules into two types: one-dimensional relationships (e.g., if stock A rises, stock B may also rise) and two-dimensional relationships (e.g., if stock A rises, stock B may rise after a certain period). The data is analyzed to distinguish between these relationships. The third part extracts key information using PCA, constructs similarity sequences, and forms K-line patterns. These patterns are matched with current sequences to obtain the most effective K-line pattern, enabling accurate predictions of future trends.

# 4 Performance verification of stock market time prediction model

## 3.1 Performance verification of improved SPM algorithm

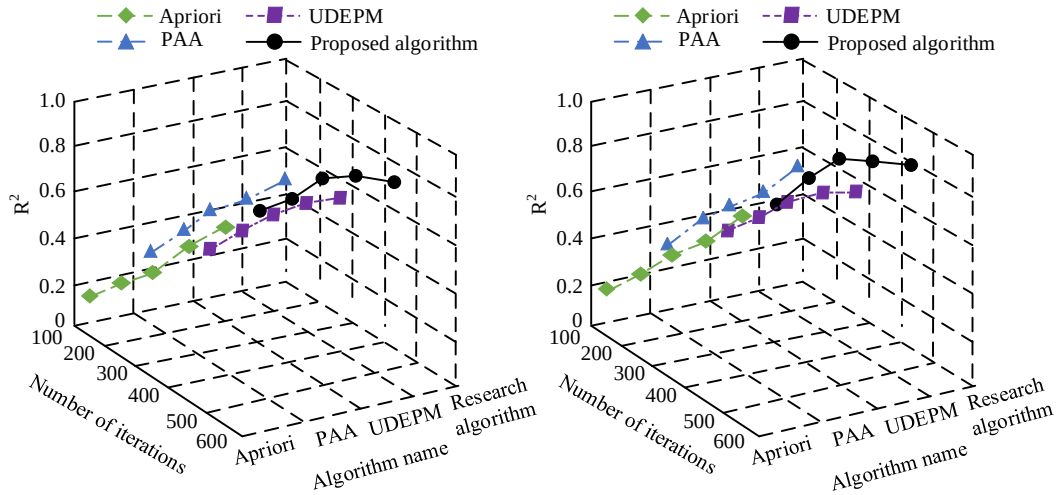
To validate the superiority of the improved SPM algorithm proposed in this study, the Ultimate Stock Prediction Machine Learning Training Dataset was selected for model validation. The dataset includes daily K-line data for all stocks up to 2019. Sixty percent of the dataset was used as the training set, 20% as the validation set, and 20% as the test set. The experimental platform was built on Windows 10 with an Intel-Xeon Gold 6132 CPU and software configurations including Pytorch 2.2, Python 3.10, and CUDA 11.2. Three comparative algorithms—Uniform Design Extreme Point Method (UDEPM), Piecewise Aggregate Approximation (PAA), and Apriori—were selected for analysis alongside the proposed algorithm. The performance of the four algorithms in stock K-line data prediction is first analyzed using the MSE and MAE metrics (the unit of error is expressed in the form of %, reflecting the relative deviation between the predicted price and the actual price), and the comparison is shown in Figure 7.




(a) Comparison of four algorithms under MSE index (b) Comparison of four algorithms under MAE index

Figure 7: Comparison of four algorithms under MSE and MAE indexes

As shown in Figure 7(a), Apriori has the largest initial value of error under the MSE indicator, with an error value of 76% at 100 iterations, indicating a large error between its predicted value and the actual K closing price. UDEPM's error decreased to 21% at 500 iterations, while PAA's error was 25% at the same iteration count. The research algorithm has the largest decreasing gradient of error value during 100-200 iterations. At 500 iterations, the MSE value of the predicted price to the actual price is 18%. In Figure 7(b), Apriori exhibited the largest MAE reduction gradient between 100 and 200 iterations, dropping from 75% to 48%. UDEPM's MAE decreased from 68% to 23%, while PAA's MAE stabilized at 26% in later iterations, slightly higher than UDEPM. The MAE curve of the research algorithm stabilizes during 400-500 iterations, indicating that the MAE value of the predicted price versus the actual price is 17%. Overall, the proposed algorithm outperformed the comparative algorithms in predicting K-line feature sets, demonstrating the best prediction performance. For the results of the validation of the two metrics, MSE and MAE, the study further performs paired t-tests with 95% confidence interval analysis to verify the statistical significance of the performance differences. The difference in MSE between the study algorithm and the second best UDEPM model is highly significant ($p < 0.01$), and the 95% confidence interval of the difference in MSE between the two is [1.8%,4.6%], which indicates that the relative reduction in MSE of the study algorithm fluctuates within this interval. As for the difference comparison on MAE, the paired t-test of the two models is $p<0.05$, which is significant, while the 95% confidence interval is [0.9%,10.5%], which further verifies the superiority of the research algorithm on the MAE index. After verifying the advantage of the research algorithm in these two indicators, R2 was selected as a comparison indicator to compare its iterative change process in the training and validation sets, and the comparison is shown in Figure 8.

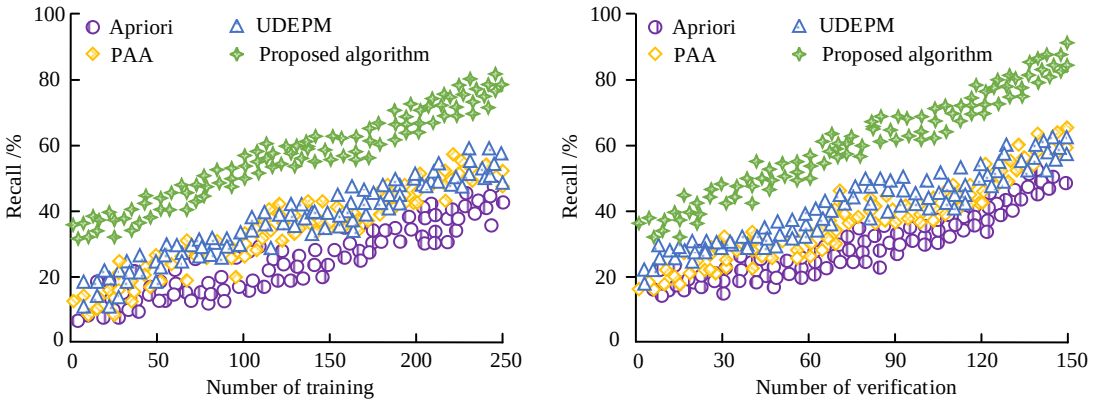



(a) Comparison of $R^2$ in the training set (b) Comparison of $R^2$ in verification set

Figure 8: Comparison of R2 of the four algorithms in the data set

As shown in Figure 8(a), Apriori's $R^2$ was only 0.18 in the early iterations. UDEPM's initial $R^2$ was 0.23, lower than PAA's. By 600 iterations, UDEPM's $R^2$ increased to 0.53. The proposed algorithm's $R^2$ rose from 0.38 to 0.73 over 500 iterations, with the most significant increase (0.38 to 0.62) occurring between 200 and 300 iterations.

In Figure 8(b), Apriori's $R^2$ showed a noticeable upward trend during iterations, with a more pronounced change than in the training set. PAA's $R^2$ increased from 0.36 to 0.53 over 500 iterations. UDEPM's $R^2$ rose significantly in the early iterations but only reached 0.54 by 600 iterations, showing no significant improvement over the training set. The proposed algorithm's $R^2$ increased notably between 100 and 300 iterations, reaching 0.79 by 600 iterations. Overall, the proposed algorithm performed better in the $R^2$ metric compared to the other algorithms. To analyze the operational performance of the four algorithms, recall rates were compared, as shown in Figure 9.




(a) Comparison of Recall in training set (b) Comparison of Recall in verification set

Figure 9: Comparison of recall rates of the four algorithms in the data set

As shown in Figure 9(a), Apriori's recall rate fluctuated significantly in the training set, reaching 43% at 250 iterations. UDEPM and PAA showed similar recall rate trends, with UDEPM achieving 59% and PAA 57% by the final iteration. The proposed algorithm's recall rate improved significantly, rising from 37% to 79% over 250 iterations. In Figure 9(b), PAA initially underperformed UDEPM but surpassed it after 150 iterations, achieving a recall rate of 62% compared to UDEPM's 59%. By the final iteration, UDEPM's recall rate was 60%. The proposed algorithm's recall rate improved notably in the validation set, rising from 36% to 88% over 150 iterations. Overall, the proposed algorithm outperformed the comparative algorithms in recall rate across both training and validation sets.

*3.2 Verification of the practical application effect of stock market time prediction model*

To validate the effectiveness of the proposed stock market time prediction model, transaction data from all Chinese stocks between 2020 and 2023 was used, totaling 2,696,545 records. Fifty percent of the dataset was used as the training set, 30% as the validation set, and 20% as the test set. Three comparative models—Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting (FEDformer), Temporal Fusion Transformer (TFT), and traditional SPM—were selected for analysis alongside the proposed model. The experimental environment was built on Ubuntu 16.04 with an Intel-Xeon Gold 6226R CPU, 256GB of memory, and an NVIDIA GeForce GTX 1080Ti GPU. Software configurations included Scikit-learn 1.6.1, Pytorch 1.10, Python 3.8, and CUDA 12.7. Then the key parameters of the research models are set, and the specific parameter settings are shown in Table 1.

Table 1: List of key parameter settings for the stock market time prediction model

| Key parameters | Project name | Model-specific settings |
|---|---|---|
| SPM hyperparameters | Sliding window length | 20 |
| | Pattern sequence length | 10 |
| | Minimum support | 0.05 |
| | Minimum confidence | 0.6 |
| PAC | Dimension of the original feature space | 8 |
| | Dimension of the principal component space | 3 |

Based on the key parameters set in Table 1, the performance of each model is performance verified. In order to comprehensively assess the performance of each model in different markets, the study divides all the rise sequences into three types according to volatility: low volatility, medium volatility and high volatility, and randomly selects two consecutive rise sub-sequences of 30 days length in each type as the six groups of rise sequences for performance validation, and carries out the error analysis of the accuracy of the predicted value of the stock K-line data of the four models, which is shown in Table 2.

Table 2: The error analysis results of the predicted value of the candlestick data

| Rise series | FEDformer | | TFT | | Traditional SPM | | Proposed model | |
|---|---|---|---|---|---|---|---|---|
| | MAPE | RMSE | MAPE | RMSE | MAPE | RMSE | MAPE | RMSE |
| 1 | 6.95 | 4.53 | 10.54 | 9.69 | 7.85 | 6.23 | 3.05 | 1.53 |
| 2 | 8.49 | 3.25 | 12.59 | 8.65 | 9.09 | 4.95 | 2.41 | 1.25 |
| 3 | 8.26 | 5.15 | 18.24 | 9.57 | 8.24 | 7.05 | 4.16 | 1.26 |
| 4 | 7.29 | 6.26 | 16.51 | 7.29 | 6.42 | 5.24 | 0.94 | 0.95 |
| 5 | 6.49 | 5.12 | 17.13 | 7.46 | 5.43 | 6.02 | 4.19 | 0.89 |
| 6 | 9.15 | 3.19 | 14.32 | 10.29 | 10.05 | 5.28 | 3.69 | 0.78 |

As shown in Table 2, traditional TFT performed the worst among the four models, with a Mean Absolute Percentage Error (MAPE) of 18.24% for the third sequence and a Root Mean Square Error (RMSE) of 10.29% for the sixth sequence. FEDformer performed poorly on the fifth sequence, with an MAPE of 6.49% and an RMSE of 5.12%. The traditional SPM algorithm performed better on the fifth sequence, with an MAPE of 5.43% and an RMSE of 6.02%. The proposed model performed best across all sequences, achieving an MAPE of 0.94% and an RMSE of 0.95% for the fourth sequence. Although its performance on the third sequence was weaker (MAPE: 4.16%, RMSE: 1.26%), it still outperformed the comparative models. After analyzing the prediction errors, the accuracy rates of the four models were compared using frequency histograms, as shown in Figure 10.

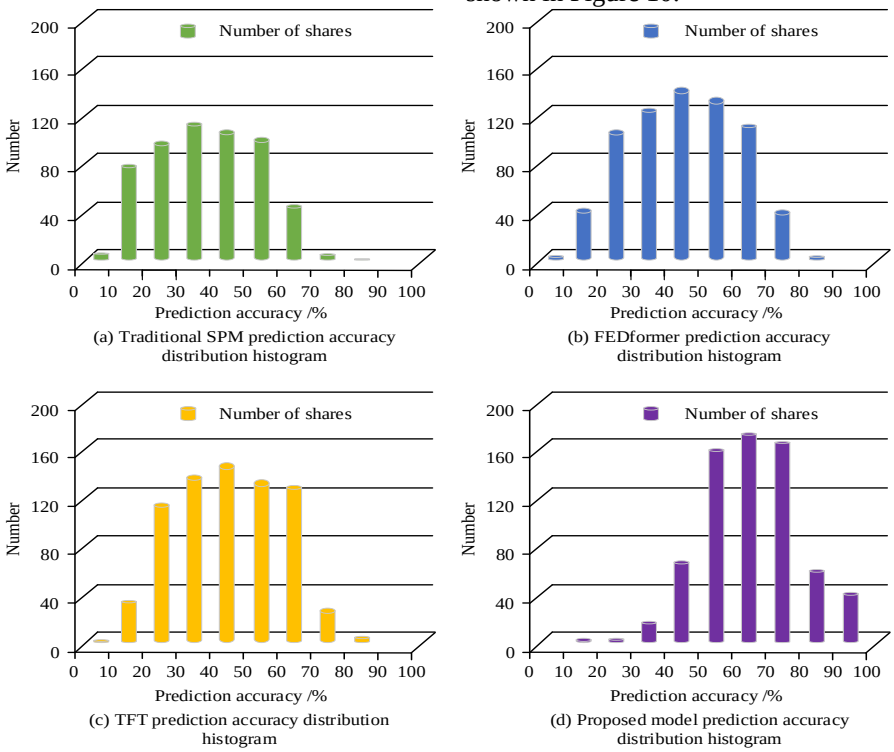
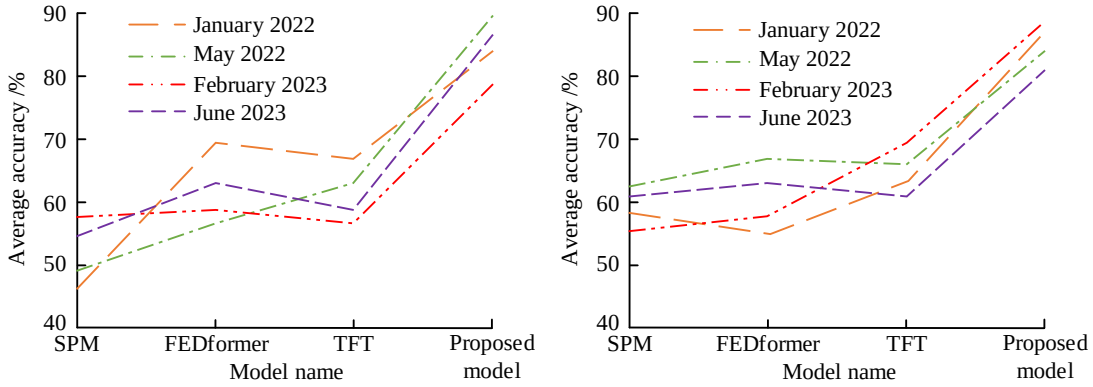



Figure 10: Histogram of accuracy distribution of the four models for stock prediction

As shown in Figure 10(a), the traditional SPM algorithm's predictions were most concentrated in the 30%-40% accuracy range, with significantly fewer stocks achieving above 60% accuracy. In Figure 10(b), FEDformer's predictions were most frequent in the 40%-50% accuracy range, with 135 stocks, outperforming the traditional SPM algorithm. In Figure 10(c), TFT's predictions were least frequent in the 0-10%, 80%-90%, and 90%-100% accuracy ranges, with only 2, 7, and 0 stocks, respectively. In Figure 10(d), the proposed model achieved the highest number of stocks in the 60%-70% accuracy range (162 stocks), followed by the 70%-80% range (158 stocks). Overall, the proposed model demonstrated the best prediction performance. Finally, the average prediction accuracy of the four models for multiple stocks was compared using data from the CSI 500 and CSI 300 indices at four time points, as shown in Figure 11.




(a) The average accuracy of CSI 500 (b) The average accuracy of CSI 300

Figure 11: Average accuracy of the four models for CSI 500 and CSI 300

As shown in Figure 11(a), the traditional SPM algorithm achieved its highest average accuracy (58%) for the CSI 500 in February 2023. FEDformer achieved an average accuracy of 69% in January 2022 and 63% in June 2023, outperforming SVM except in May 2022. The proposed model achieved its highest average accuracy (90%) in May 2022. In Figure 11(b), FEDformer's average accuracy for the CSI 300 in January 2022 and February 2023 was slightly lower than the traditional SPM algorithm. TFT achieved its highest average accuracy (69%) for the CSI 300 in February 2023. The proposed model consistently achieved over 80% average accuracy for the CSI 300 across all four time points, with 81% accuracy in June 2023, still outperforming the comparative models. To verify the performance advantages of the research model, four cutting-edge time series network prediction models were selected for the study, including two models: TFT and FEDformer. In addition, the Decomposition Transformers with Auto-Correlation for Long-term Series Forecasting were selected. Autoformer and Beyond Efficient Transformer for Long Sequence Time-Series Forecasting (Informer). The operating mechanisms and comparisons of advantages and disadvantages of each model are shown in Table 3.

Table 3: Comparison list of characteristics of different time series prediction models

| Model name | Operating mechanism | Advantages | Disadvantages |
|---|---|---|---|
| TFT | Variable selection network and multi-head attention | Interpretability and dynamic feature screening | Large number of parameters, long training period, and sensitivity to small samples |
| FEDformer | Frequency domain decomposition and time domain attention | The periodic signal prediction accuracy is high and it can model both frequency-domain and time-domain signals simultaneously | High complexity |
| Autoformer | Sequence decomposition and auto-correlation attention | Cyclical and trend advantages | Lack of performance in dealing with non-periodic data |
| Informer | ProbSparse self-attention and multi-scale convolution | Efficiently handle extremely long sequences and have high operational efficiency | The sparsity parameter needs to be adjusted |
| Research model | Improve the extraction of similarity sequences and sliding similarity matching by SPM and PCA | Take into account both interpretability and the balance of numerical performance and computational complexity | Relying on predefined pattern libraries and lacking real-time online update capabilities |

From Table 3, it can be seen that the research model is to construct similarity sequences by means of dimensionality reduction through principal component analysis, however, there is an over-reliance on predefined pattern libraries, which needs to be strengthened for online updating in extreme mutation scenarios.The Informer model, on the other hand, has the advantage of dealing with ultra-long sequences, however, it is more sensitive to sparsity parameter, which needs to be tuned in a fine-grained way.

# 5 Discussion

In order to improve the accuracy of financial forecasting model for stock market prediction, the study proposes a stock market time series data prediction model with improved SPM algorithm. The model introduces the stock market K-line sequence into SPM in which the similarity sequence is constructed and the optimal K-line pattern is dynamically generated, which is slidingly matched with the current subsequence, thus realizing the accurate price prediction of the stock market. The experimental results show that the MSE of the research model is only 18% after 500 iterations, which is substantially lower than the MAE value range of the comparison model. And the R2 of the research model increases from 0.38 to 0.73 in the test set and validation set, indicating that the model achieves good fitting results for both the historical stock market prediction trend and unknown samples. Meanwhile, its recall rate rapidly increases from 36% to 88% at 150 iterations, indicating that the model possesses high sensitivity in the upward space, which further reduces the riskiness of missed detection. Through the high recall and R2, it significantly reflects that the research model is able to balance the two objectives of detecting real signals and accurately restoring prices. In the self-constructed dataset, the MAPE can reach only 0.94% and the RMSE is 0.95%, which verifies its adaptability in dealing with different stock markets. Meanwhile, in the analysis of the full-sample histogram, the prediction accuracy is concentrated at 60%-70%, which is better than the distribution peak of the comparison model. In the face of CSI 500 large-cap stocks, the average prediction accuracy of the research model can reach up to 90%, indicating its potential application on highly liquid, large-cap stock samples. Compared with the running algorithms based on the Transformer architecture, the computational complexity of the research model is moderate, which can meet the demand of online backtesting, and it can visualize the optimal K-line pattern, which enhances the interpretability of the model. In terms of data use, the study strictly follows privacy regulations to ensure legally compliant collection, storage and processing of data. It also establishes complete input and output logs with interpretable weight records to provide a basis for regulatory audits. When deployed to real financial environments, the risk of market manipulation needs to be guarded against, by combining real-time monitoring and risk control strategies in order to strengthen the response to the volatility of the financial market. In summary, the stock market prediction model constructed based on the improved SPM algorithm can accurately analyze stock time series data efficiently and precisely, provide investors with reliable decision support, and help to enhance investment returns and reduce trading risks.

# 6 Conclusion

With the rapid development of the financial industry, the public's demand for investing in financial stock markets has increased. However, traditional stock market prediction models cannot analyze influencing factors from a global perspective, making it difficult to accurately predict stock market trends. To address this issue, the study introduced K-line sequences to optimize the SPM algorithm, constructing similarity sequences and generating optimal K-line patterns. These patterns were matched with current sequences to achieve stock price prediction. It was found that in the validation of the public dataset, the research model had an MSE of 18% at 500 iterations. Meanwhile, in the analysis of the self-constructed dataset, the accuracy of the research model falls between 60% and 70% at most in the histogram analysis of the prediction accuracy of all stocks, and the average accuracy of the research model reaches up to 90% for all stocks of the CSI 500. These results indicate that the proposed model has excellent prediction accuracy, enabling precise analysis of individual and multiple stocks while maintaining low error rates. However, this study focused only on time series data of stocks. Future research could expand the application scope by incorporating mechanisms for screening multiple stocks and developing investment recommendation strategies.

# LIST OF ABBREVIATIONS

SPM: Sequence Pattern Mining
K-line: Candlestick Chart
UDEPM: Uniform Design Extreme Point Method
PAA: Piecewise Aggregate Approximation
PCA: Piecewise Aggregate Approximation
FT: Fourier Transform
TFT: Temporal Fusion Transformer
Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting
Autoformer: Decomposition Transformers with Auto-Correlation for Long-term Series Forecasting
FEDformer: Frequency Enhanced Decomposed Transformer
MSE: Mean Squared Error
MAE: Mean Absolute Error
MAPE: Mean Absolute Percentage Error
RMSE: Root Mean Squared Error
$R^2$: Coefficient of Determination
$\bar{A}$: the absolute price change
$P_{t+1}$: denotes the subsequence price at time $t+1$
$P_t$: denotes the subsequence price at time $t$

$T_s$ : the time difference between adjacent extremes

$T_{t+1}$ : denotes the specific time at $t+1$

$T_t$ : denotes the specific time at $t$

$S(Z,U)$ : the degree of matching under optimal alignment

$Z$ : the input sequence

$U$ : the K-line pattern sequence

$sim$ : the similarity function

$Max$ : the maximum value in the set

$m$ : the length of the sequence $Z$

$n$ : the length of the sequence $U$

$A$ 、$B$ : two subsets of itemsets under different conditions

$support\_cout$ : the number of transactions

$l$ : the frequent itemset

$s$ : each non-empty subset of the frequent itemset

$\min\_conf$ : the minimum confidence threshold

$l(x)$ : the price change value

$p(x_i)$ : the probability of a specific price change category on a trading day for the $i$ -th stock

$H$ : the entropy of price changes on a trading day

$t_i$ : time

$J_{t_i}$ : all K-line indicators for the $t_i$ -th day

$n$ : the total number of stocks

$A_m$ : the K-line feature set for stock $m$ at the current time

$i$ : the imaginary unit

$\hat{f}(\xi)$ : the Fourier Transform function

$P_{A_i}$ : the prediction probability for the SS-th stock

$k$ : the total number of stocks in the experiment

# References


[1] Younis H, Sundarakani B, Alsharairi M. Applications of artificial intelligence and machine learning within supply chains: systematic review and future research directions. Journal of Modelling in Management, 2022, 17(3): 916-940. https://doi.org/10.1108/JM2-12-2020-0322

[2] Xiao Hu, Zhenghua Deng. Impact of Digital Finance On Technological Efficiency Of Creative Enterprises: Evidence From Chinese Capital Market. Malaysian E Commerce Journal. 2023, 7(1): 42-45. https://doi.org/10.26480/mecj.01.2023.42.45

[3] Feizabadi J. Machine learning demand forecasting and supply chain performance. International Journal of Logistics Research and Applications, 2022, 25(2): 119-142. https://doi.org/10.1080/13675567.2020.1803246

[4] Wu S, Liu Y, Zou Z, Weng T H. S_I_LSTM: stock price prediction based on multiple data sources and sentiment analysis. Connection Science, 2022, 34(1): 44-62. https://doi.org/10.1080/09540091.2021.1940101

[5] Chan Wen, Ziqi Li, Xiaomin Zhou. China's economy based on grey forecast model and k-means clustering algorithm. Advances In Industrial Engineering And Management, 2023, 13(2)

[6] Shawkat M, Badawi M, El-ghamrawy S, Arnous R, El-desoky A. An optimized FP-growth algorithm for discovery of association rules. The Journal of Supercomputing, 2022, 78(4): 5479-5506. https://doi.org/10.1007/s11227-021-04066-y

[7] Girang Permata Gusti. Analysis of Indonesian Citizens' Financial Behavior In Online Shopping: A Study of Average Expenditures and Influence Factors. Malaysian E Commerce Journal. 2024, 8(1): 17-22. https://doi.org/10.26480/mecj.01.2024.17.22

[8] Tanimu Mohammed, Yahaya Haruna Umar, Samuel Olorunfemi Adams. Modeling the volatility for some selected beverages stock returns in nigeria (2012-2021): a garch model approach. Matrix Science Mathematic, 2022, 6(2):41-51. Https://doi.org/10.26480/msmk.02.2022.41.51

[9] Karavias Y, Narayan P K, Westerlund J. Structural breaks in interactive effects panels and the stock market reaction to COVID-19. Journal of Business & Economic Statistics, 2023, 41(3): 653-666. https://doi.org/10.1080/07350015.2022.2053690

[10] Huang G, Gan W, Yu P S. TaSPM: Targeted sequential pattern mining. ACM Transactions on Knowledge Discovery from Data, 2024, 18(5): 1-18. https://doi.org/10.1145/3639827

[11] Wu Y, Chen M, Li Y, Liu J, Li Z, Li J, Wu X. ONP-Miner: One-off negative sequential pattern mining. ACM Transactions on Knowledge Discovery from Data, 2023, 17(3): 1-24. https://doi.org/10.1145/3549940

[12] Djenouri Y, Belhadi A, Srivastava G, Lin J C W. Advanced pattern-mining system for fake news analysis. IEEE Transactions on Computational Social Systems, 2023, 10(6): 2949-2958. https://doi.org/10.1109/TCSS.2022.3233408

[13] Li Y, Zhang C, Li J, Song W, Qi Z, Wu Y, Wu X. MCoR-Miner: Maximal co-occurrence nonoverlapping sequential rule mining. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(9): 9531-9546. https://doi.org/10.1109/TKDE.2023.3241213

[14] Abbas G. A Sequential Pattern Mining Method for the Individualized Detection of Online Banking Fraudulent Transactions. PatternIQ Mining, 2024, 1(1): 34-44. https://doi.org/10.70023/piqm244

[15] Zheng H, Wu J, Song R, Guo L, Xu Z. Predicting financial enterprise stocks and economic data trends using machine learning time series analysis. Applied and Computational Engineering, 2024, 87(5): 26-32. https://doi.org/10.54254/2755-2721/87/20241562

[16] Kumar R, Kumar P, Kumar Y. Multi-step time series analysis and forecasting strategy using ARIMA and evolutionary algorithms. International Journal of Information Technology, 2022, 14(1): 359-373. https://doi.org/10.1007/s41870-021-00741-8

[17] Sisodia P S, Gupta A, Kumar Y, Ameta G K. Stock market analysis and prediction for NIFTY50 using LSTM Deep Learning Approach//2022 2nd international conference on innovative practices in technology and management (ICIPTM). IEEE, 2022, 2(3): 156-161. https://doi.org/10.1109/ICIPTM54933.2022.9754148

[18] Kirisci M, Cagcag Yolcu O. A new CNN-based model for financial time series: TAIEX and FTSE stocks forecasting. Neural Processing Letters, 2022, 54(4): 3357-3374. https://doi.org/10.1007/s11063-022-10767-z

[19] Mukherjee S, Sadhukhan B, Sarkar N, Roy D, De S. Stock market prediction using deep learning algorithms. CAAI Transactions on Intelligence Technology, 2023, 8(1): 82-94. https://doi.org/10.1049/cit2.12059

[20] Kumar S, Mohbey K K. A utility-based distributed pattern mining algorithm with reduced shuffle overhead. IEEE Transactions on Parallel and Distributed Systems, 2022, 34(1): 416-428. https://doi.org/10.1109/TPDS.2022.3221210

[21] Yan D, Qu W, Guo G, Wang X, Zhou Y. PrefixFPM: a parallel framework for general-purpose mining of frequent and closed patterns. The VLDB Journal, 2022, 31(2): 253-286. https://doi.org/10.1007/s00778-021-00687-0

[22] Kumar D, Sarangi P K, Verma R. A systematic review of stock market prediction using machine learning and statistical techniques. Materials Today: Proceedings, 2022, 49(4): 3187-3191. https://doi.org/10.1016/j.matpr.2020.11.399

[23] Javed F, Mustafa G, Fatima G, Maurya S K, Alshehri M H, Mubeen I. Joule-Thomson expansion for charged-AdS black hole with nonlinear electrodynamics and thermal fluctuations by using Barrow entropy. Journal of High Energy Astrophysics, 2024, 44(5): 60-73. https://doi.org/10.1016/j.jheap.2024.09.003

[24] Wu Y, Zhang F, Li F, Yang Y, Zhu J, Wu H H, Lu Z. Local chemical fluctuation mediated ultra-sluggish martensitic transformation in high-entropy intermetallics. Materials Horizons, 2022, 9(2): 804-814. https://doi.org/10.1039/D1MH01612A