CIT. Journal of Computing and Information Technology, Vol. 33, No. 1, March 2025, 43–56
doi: 10.20532/cit.2025.1005939

Characteristic Representation of Stock Time Series Data Based on Trend Extreme Points of K-line Combinations

Lei Han[1], Xuedong Gao[1] and Haining Yang[2]

[1]School of Economics and Management, University of Science and Technology Beijing, Beijing, China
[2]Linyi Vocational College, Linyi, China

The study of stock time series has been an important area of research in economics, finance, and management. Time series feature representation serves as the primary approach for studying time series. The K-line chart is a common representation of stock time series data. This study proposes a segmented representation method KCTEP based on the extreme points of stock K-line portfolio trend for K-line chart data, which is validated in sequence compression rate and distance metric. The experimental results show that the KCTEP method significantly improves the trend description by 8.63% and the compression rate by 2.95% when compared with the uniform extreme point representation method and the piecewise aggregation approximation method (PAA).The results lead to significant enhancement of the trend description effect and reduction of the distance metric error.

ACM CCS (2012) Classification: Theory of computation → Design and analysis of algorithms → Algorithm design techniques → Preconditioning
Applied computing → Law, social and behavioral sciences → Economics

Keywords: stock time series, K-line chart, segmented representation, trend analysis

1. Introduction

Time series data is a class of high-dimensional data and is an important research element in data mining. It is widely used in intelligent manufacturing [1][2], e-commerce [3] and other fields. The stock market is a typical application of time series data. Compared with traditional economic data, stock prices change more drastically and at shorter time intervals, which makes it more challenging to analyze and forecast stock time series data. How to model and analyze stock time series data is of great significance in predicting stock price trends and formulating investment strategies.

Characteristic representation of stock time series data is a key issue in the study of stock time series. Usually, stock time series data are characterized by high dimensionality, large scale, non-stationarity and noise interference. Through the feature representation of stock time series data, the potential laws and patterns in the data can be discovered, and the interpretability and prediction accuracy of the data can be improved.

Segmented linear representation, pole extraction and trend analysis have been of great interest when dealing with the problem of characterizing non-stationary time series. In a complex stock market, stock prices are affected by a variety of uncertainties and high levels of noise, resulting in sharp price fluctuations. In order to better understand and analyze the trend characteristics of stock time series, most of the existing work uses linear transform methods such as Fourier transform, discrete cosine transform, wavelet transform and PAA. However, the effectiveness of these methods depends on the match between the data and the assumptions of the selected transforms and is dependent on set thresholds, which may have some limitations.

Referring to the Japanese trader who plotted the price of rice to explore the pattern of its rise and fall and formed the now widely used K-line chart, K-line chart is one of the most common forms of stock time series data. It reflects the change of the stock price by displaying the four key prices, *i.e.*, the opening price, the closing price, the high price and the low price, in each time period. There are many variations of negative and positive lines in a K-line chart, and various combinations of K-lines represent different meanings. Therefore, how to mine and utilize K-line patterns for stock price prediction has become a major research component of K-line technical analysis. Observation of a stock's K-line chart reveals that trend extreme points in K-line changes are usually of great significance. These trend extreme points reflect the overall trend and major characteristic patterns of the stock time series. Trend extremes can be viewed as inflection points in a stock's price action, signaling a change in price from up to down or from down to up. They are usually characterized by highs and lows in the K-chart and represent key moments and important turning points in the market.

Stock price forecasting using trend extreme points is an important research direction. By identifying and analyzing trend extreme points in K-charts, it is possible to reveal the underlying trend and future movements of stock prices. This provides investors with an important basis for decision making and helps them make more informed investment decisions. This study proposes a segmented representation of stock time series data based on K-line chart data, combined with the trend extreme point extraction method, which extracts the trend extreme points by re-representing the K-line chart to better represent the stock time series data.

This study is structured as follows. Chapter 2 introduces related research work and progress. Chapter 3 introduces the algorithmic flow of the proposed method in this study. Chapter 4 introduces the experimental process and evaluates and discusses the experimental results. Chapter 5 summarizes the conclusions drawn from this study.

2. Related Research

The study of the representation of time series features began in the field of engineering, and along with the further development of information technology and machine learning, the study of time series patterns has increasingly migrated to other fields, such as finance, economics, and medicine. With the increasing size of time series and the increasing degree of interference in various application scenarios, it is very difficult to model and analyze with raw data, so it is especially important to transform and reduce the dimensionality of the raw data appropriately. The transformation of time series data refers to the extraction of features such as mean, variance, frequency and magnitude of the original time series, and then mapping these features into a new vector space to transform them into data, called the construction of the feature space, so as to achieve the purpose of data compression and reduce the cost of computation, which is known as the method of time series feature representation. For time series characterization, it is crucial to strike a balance between downscaling and preserving important trend features embedded in the original series. For time series feature representation methods, the following conditions usually need to be satisfied:

- the ability to significantly reduce the dimensionality of time series data;
- effective retention of the overall trend and local characteristics of the time series;
- efficient characterization of the time series data with high representation accuracy.

2.1. Segmentation-based Representation

In 2004, H. Wu *et al.* [4] proposed Piecewise Linear Representation (PLR) in stock forecasting and utilized the results of PLR to investigate its data similarity and stock price prediction. On the basis of PLR, scholars have proposed segmented bottom-up linear approximation to represent time series algorithm (PLR-BU) and partial linear top-down algorithm (PLR-TD). P. Jia *et al.* [5] proposed how to use the fitting error to find different δ in each line segment separately, and the result improved the fineness of

PLR and made the overall error smaller, but the computing time spent increased.

Keogh *et al.* [6] proposed the Piecewise Aggregate Approximation (PAA), which can divide the original time series equidistantly and calculate the average value of the subsequence based on the pre and post order of the division. However, since this method does not take into account the fact that there are similarity differences between neighboring subsamples, its results are often inaccurate. For this reason, researchers have carried out a series of improvement operations on the PAA method. On the one hand, Huang *et al.* [7] used data features such as slope and variance to enhance the characterization of the original set of means, and on the other hand, Keogh *et al.* [8] proposed an adaptive segmentation constant approximation method (APCA). This method is based on the idea of dynamic programming. According to the characteristics of the time series data, it conducts an optimal segmentation on the time series, thus obtaining a collection of time series segments with different lengths.

Liu *et al.* [9] proposed a time series trend extraction algorithm based on turning points and trend segments on the basis of segmented linear representation of time series. The experimental results show that the method not only has good noise immunity but also has higher fitting accuracy under the same compression rate, which can provide better features for subsequent data mining. Li *et al.* [10] discovered the fluctuating characteristics of time series by studying the trend of time series. They proposed to divide the trend change of time series into the upper and lower layers, and eliminate the trend while keeping points in the upper and lower layers, respectively. The experimental results show that the segmentation method has low time complexity and is easy to implement, and on the basis of maintaining the trend characteristics of the time series, the fitting error obtained is smaller.

In order to enhance the fast dimensionality reduction effect as well as to reflect the overall trend of the time series, researchers have proposed the Symbolic Aggregate approXimation (SAX) method, which is based on the PAA method and transforms the time series into symbolic characters according to the normal distribution [11]. Since SAX is a symbolic representation based on the PAA method, SAX also suffers from some defects similar to the PAA method, *i.e.*, SAX can only represent the average value of the segmented series with symbols, and it is easy to ignore the important data features that reflect the trend of the series. To solve the above problems, researchers have used different strategies to improve the SAX method [12][13][14]. LLkhagva *et al.* [15] proposed the ESAX (Extended SAX) algorithm, which enhances the classification results by fusing the maximum, minimum, and mean values of the segments into a symbol aggregation algorithm. Although the ESAX algorithm can effectively improve the classification accuracy and reduce the computation, it cannot well describe the continuously changing dataset, *i.e.*, it lacks good dynamics. Sun *et al.* [16] proposed an approximate representation for describing sequence segments using their mean and trend distances (Symbolic Aggregate approXimation based on Trend Distance, SAX_TD), which can better characterize the overall trend of sequence segments. Zhang *et al.* [17] proposed an approximate representation of time series symbol aggregation based on trend features, in which the trend distance factor and the trend pattern factor of the segments are used to jointly describe the trend features of the sequences, in addition to retaining the mean value feature of each sequence segment. Although the above feature representation method locally improves the accuracy of time series pattern recognition, it still has the problem of losing the trend features inside the sequence segments.

2.2. Representation Based on Spatial Transformations

Representation based on spatial transformation refers to the transformation of a time series into another data space, and the transformation process and the selection of feature coefficients are independent of the data itself. Specifically, the time series can be transformed from the original time-domain space to the frequency-domain space, signal spectrum space, *etc.*, in which the characterization can be completed. The commonly used methods are Discrete Fourier Transform (DFT) [18][19] and Discrete

Wavelet Transform (DWT) [20][21][22], where DWT is an important transform.

The DFT is a method of transforming a time domain signal into the frequency domain, which can transform the time domain information in the original time series into the frequency domain information, using a set of complex numbers to represent the periodicity and frequency components of the original series. In DFT, the number of data points before and after the transform are the same, which is suitable for smoother time series [23]. For time series with large ups and downs, this algorithm is not able to show local variations efficiently and is prone to ignore local features. Agrawal *et al.* [24] proposed a method for initial transformation of time series using DFT, which utilizes trigonometric basis functions to achieve the mapping of time series to a lower dimensional space in the process of finding the similarity of the time series, and introduces a new binary tree to understand the similarity between layers based on new sequences. This method was shown to be efficient for large-scale datasets.

DWT can make up for the disadvantage of poor performance of local features of DFT. It can take into account the local features in the time domain and frequency domain and use signal processing methods to transform the time domain data into wavelet coefficients with the same length but with different values to effectively grasp the local information, which is better to application to the non-stationary time series. Most of the time series with financial backgrounds are non-stationary, and the application of DWT is more valuable in this field than DFT [25].

2.3. Model-based Representations

Model-based feature representation is a method that describes the original time series data in a simplified way using some mathematical model. The method assumes that the original time series data is generated based on some mathematical model, and therefore the feature information in the time series can be constructed and extracted based on this mathematical model. Model-based feature representation methods can be applied to various types of time series data, and different mathematical models can be selected for feature representation according to practical needs. Commonly used model-based time series feature representation methods include methods based on statistical models and methods based on neural network models.

The main statistical model-based methods are based on Autoregressive Mode (ARM) [26], Orthogonal Polynomial Regression Analysis (OPRA) [27], Hidden Markov Model (HMM) [28], Kernel Model (KM) based feature representation [29], *etc.* In addition, the feature representation based on the Hidden Markov Model treats the time series as a sequence of observations generated by a Hidden Markov Chain. The time series are represented by modeling the Markov chain and based on the transfer and emission matrices. Statistical model-based feature representation can help people better understand and analyze time series data and provide a basis for subsequent data processing and application. In the face of massive, high-dimensional and multi-source time series data, how to construct corresponding time series models for different time series data often requires the experience and knowledge of domain experts. At the same time, the time series data itself has certain special characteristics, such as long-term dependence, periodicity, stochasticity, *etc.*, and these characteristics also need to be fully considered in the process of data model construction. The model-based time series feature representation method extracts the features of time series data by building a mathematical model, which has the advantages of predictive ability and data smoothing. However, the method requires significant data assumptions, and parameter selection and model assumptions may affect the accuracy of the results, while the computational complexity is high. Therefore, such methods have some limitations.

With the development of computers, researchers have respectively proposed a series of neural network model-based methods for time series feature representation. For example, Wang *et al.* [30] utilized a variety of typical neural networks respectively to effectively perceive the temporal correlations embedded in the time series data and complete the temporal characterization. Lin *et al.* [31] combined a traditional convolutional neural network model with a recurrent neural network in order to achieve an effective understanding of the main temporal

trends and temporal contextual semantic correlations and to further enhance the model's representational capabilities. Zheng *et al.* [32] constructed a multi-channel deep convolutional neural networks (MC-DCNN) for multivariate time series feature representation, and utilized a multilayer perceptron for classification based on the results to further improve the classification of multivariate time series. Wang *et al.* [33] proposed the DAFA-BiLSTM model, which can learn both linear and nonlinear features and learn nonlinear feature information from different directions while generating hierarchical feature representations. Although deep learning methods can learn features from large-scale data, they have relatively poor interpretability and transparency, require large computational overhead, are difficult to meet the traditional sense of downscaling the representation of time series, and have poor generalization ability, which is a limitation for financial time series.

The above research methodology still faces some problems and challenges in the segmented representation of stock time series data. Traditional linear models and algorithms are often difficult to capture the nonlinear characteristics and time variability of stock time series data well, resulting in less accurate analysis and forecasting results that cannot fully reflect the trend characteristics of stock prices. At the same time, stock time series data often contain a large number of data points, resulting in high computational complexity of some algorithms, which makes it difficult to handle large-scale data, thus affecting the efficiency of analysis and forecasting.

3. Algorithmic Process

3.1. Re-representation and Combination of Stock Time Series K-lines

A stock K-line chart is a type of chart that graphs the trend of stock prices by plotting data such as opening prices, closing prices, highs and lows over a certain period of time using elements such as rectangles and sub-sequences as the basic plotting unit. A K-line is a representation of how the price of a stock fluctuates at certain time intervals, as shown in Figure 1.

In past studies, scholars were mostly accustomed to using the most deterministic closing price for connecting the dots to derive a stock's movement. But from the basic meaning of the K-line, the opening and closing prices only have time significance, that is, a period of time at the beginning and at the end of the price of the stock, and the highest and lowest prices in a period of price fluctuations represent the maximum force of buyers and sellers, or "Force Index". Benjamin Graham, known as the "father of security analysis" in the stock market, described stock prices as "short-term voters, long-term weighing machines". Along these lines, the high and low prices represent the largest number of votes in both directions, making them even more interesting to study. Therefore, in this study, the identification of extreme points is done by focusing more on the maximum and minimum price of the K-line. The K-line can be simplified into two patterns, up and down, as shown in Figure 2.

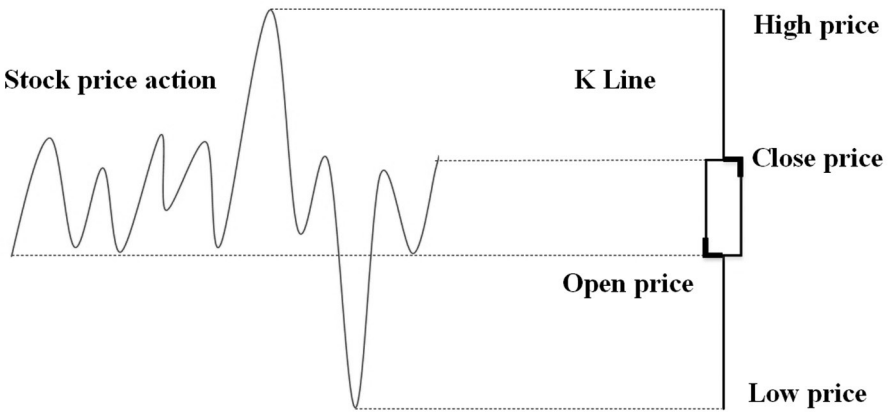

Figure 1. K-line expression of stock price.

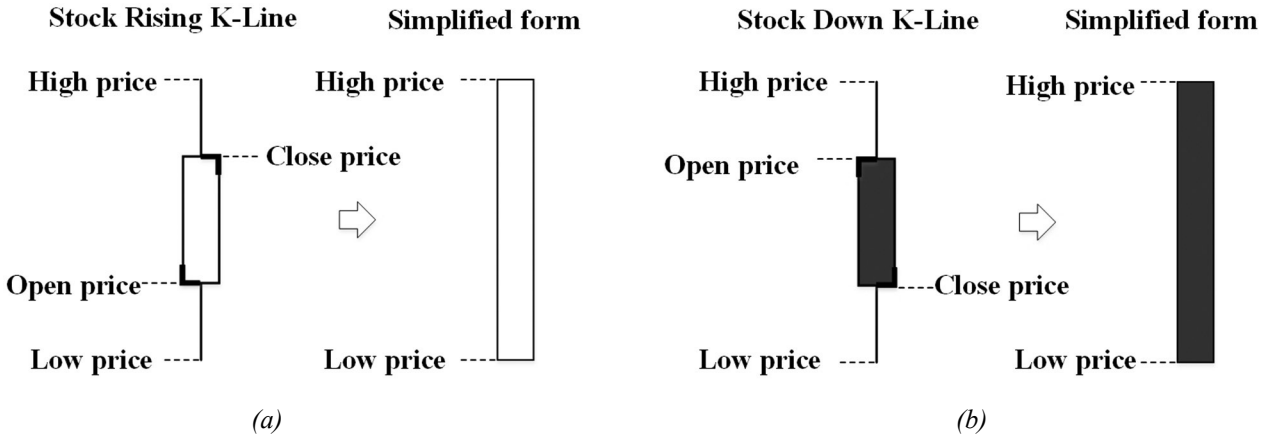

Figure 2. K-line representation.

Definition 1: After the K-lines have been re-represented, it is necessary to make a judgment about the positional relationship of adjacent K-lines. For two adjacent K-lines K_{i-1}, K_i, if it meets the requirements of $K_{i-1\_max} < K_{i\_max}$, $K_{i-1\_min} > K_{i\_min}$ or $K_{i-1\_max} > K_{i\_max}$, $K_{i-1\_min} < K_{i\_min}$, then there exists a composition relation. If $K_{i-1\_max} > K_{i-2\_max}$, $K_{i-1\_min} > K_{i-2\_min}$, then recombine K_{i-1}, K_i into a new K-line K'_i, use the larger highs and lows of K_{i-1}, K_i to correspond to K'_{i_max}, K'_{i_min}; If $K_{i-1\_max} < K_{i-2\_max}$, $K_{i-1\_min} < K_{i-2\_min}$, then recombine K_{i-1}, K_i into a new K-line K'_i, use the smaller highs and lows of K_{i-1}, K_i to correspond to K'_{i_max}, K'_{i_min}.

According to Definition 1, four positional relationships are obtained by combining the K-lines, as shown in Figure 3.

3.2. K-line Sequence Extreme Point Extraction

The purpose of extreme point extraction is to accurately identify extreme highs and lows of a stock trend in time series data in conjunction with stock market theory. These extreme points represent the highest and lowest levels of a stock price or index over a period of time.

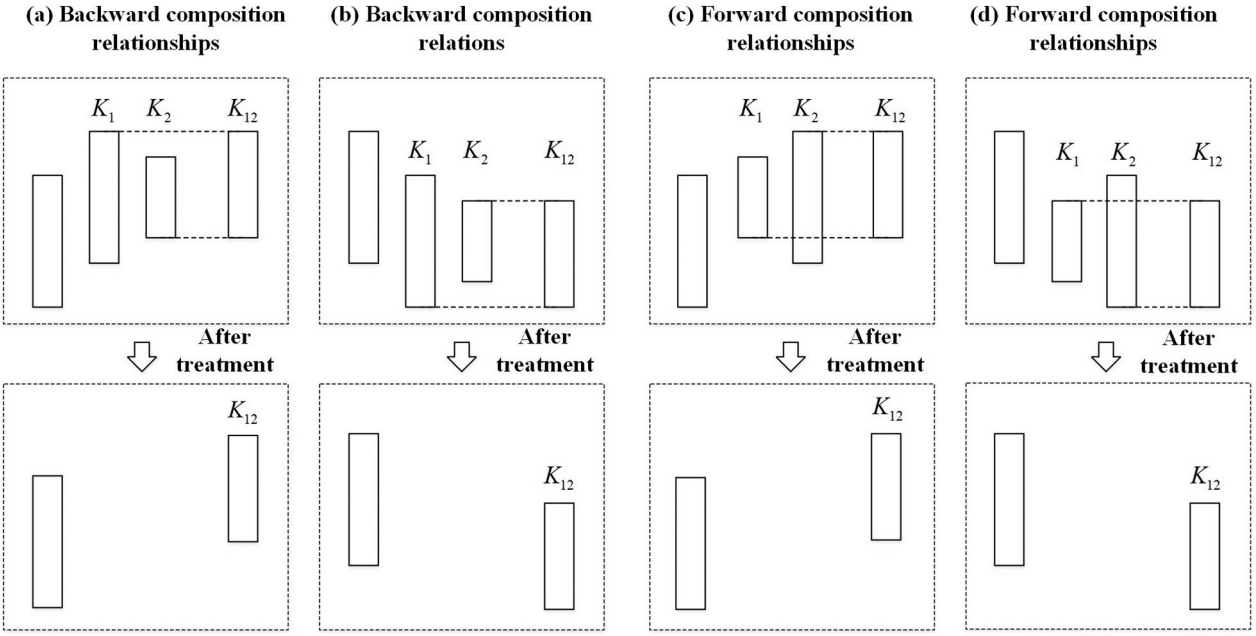

Figure 3. Processing inclusion relationship diagram.

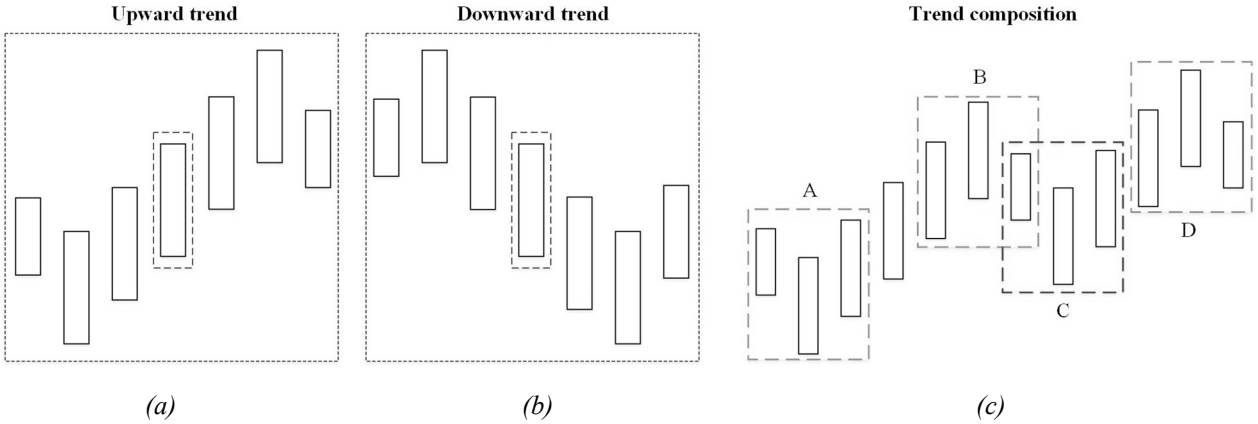

Figure 4. Trend Structure and Composition.

Based on the fractal market theory of stocks, we can apply the following logic to determine extreme points in stock time series data: 1) For the top fractal, observe the highs in the three neighboring K-lines, the middle one should have the highest high and the highest low, forming a top pattern, indicating that the uptrend may be ending. 2) For the bottom fractal, observe the lows in the three adjacent K-lines, the middle one should have the lowest low and the lowest high to form a bottom pattern, indicating that the downtrend may be over. Therefore, this study proposes definition 2.

Definition 2: For a given piece of K-line sequence data, if there are three consecutive K-lines K_{i-1}, K_i, K_{i+1}. If $K_{i-1\_max} < K_{i\_max}$, $K_{i+1\_max} < K_{i\_max}$ and $K_{i-1\_min} < K_{i\_min}$, $K_{i+1\_min} < K_{i\_min}$, then it is regarded as the extreme high point; if $K_{i-1\_max} > K_{i\_max}$, $K_{i+1\_max} > K_{i\_max}$ and $K_{i-1\_min} > K_{i\_min}$, $K_{i+1\_min} > K_{i\_min}$, then it is regarded as the extreme low point.

Definition 3: K-Line sequence trend: There needs to be at least three separate K-lines between the extremes, with no extremes higher or lower than the starting extreme in between.

A stock price can only be called a trend if it exhibits a characteristic of inertia, and a standard uptrend needs to fulfill two characteristics: The first is to start with an extreme low and end with an extreme high; the independent K-line that is neither of the three K-lines to determine the extreme high nor the three K-lines to determine the extreme low, and the uptrend is the connecting line from the extreme low to the extreme high; downtrend is to extreme highs to start, to extreme lows to end, at least one independent K-line between the two points, downtrend is the line from the extreme highs to the extreme lows, that is, the extreme highs to the extreme lows need to have at least five K-lines continue to run in one direction to be called trend, as shown in Figure 4 (a, b).

Based on the above description, it is clear that dealing with trends is similar to dealing with K-line composition relationships. An uptrend is inevitably followed by a downtrend. However, simple upward and downward movements do not fully satisfy the requirements of a trend. This study entails classifying the extreme highs and extreme lows of a trend into a standard uptrend or downtrend. This division can be viewed as a compositional relationship of trends, as shown in Figure 4(c).

Based on the constituent characteristics of a trend, we infer that an important sign in determining whether a trend is over is the emergence of a new trend, and that intervening K-lines that do not fall within the principles of the judgment do not constitute a trend.

Definition 4: Extreme low combination K_{i-1_min} to extreme high combination K_{i-1_max} constitutes an uptrend, but the next extreme low combination K_{i_min} does not constitute a downtrend, and then the highest point of the extreme high combination K_{i_max} is higher than the highest point of the extreme high combination K_{i-1_max}. Then it is called the extreme high combination K_{i_max} which contains the extreme high combination K_{i-1_max}, and the uptrend becomes an uptrend from K_{i-1_min} to K_{i_max}. Instead, it is called

extreme low combination $K_{i\_\min}$ which contains extreme low combination $K_{i-1\_\min}$ to form a downtrend from $K_{i-1\_\max}$ to $K_{i\_\min}$.

Based on the constituent features of a trend, this study can infer that an important indicator of the end of a trend is the emergence of a new trend. In stock time series analysis, accurately identifying the extreme points of a trend is a key task. In practice, common methods include the use of technical indicators, trend lines, moving averages and other tools to identify and confirm trend extremes. However, due to the complexity and volatility of the market, identifying the extreme points of a trend is not always an easy task and requires a combination of analytical tools and empirical judgment. Synthesizing the above definitions, this study proposes the segmentation representation method KCTEP based on the extreme points of the trend of a stock K-line portfolio, as shown in Algorithm 1.

The implementation process of the KCTEP algorithm consists of the following steps:

Step 1. The stock time series data are smoothed to remove the effects of noise and mutation points.

Step 2. The extreme points of the trend are identified by calculating the local maxima and local minima within a certain period.

Step 3. Turning points of the trend and the continuity of the trend are identified by comparing neighboring extreme points.

Step 4. Based on the identified trend extremes, trend lines are plotted or other technical indicators are used for further trend analysis.

In order to show the principle of KCTEP algorithm more clearly, this study obtained the K-line data of the SSE index from May 19,

Algorithm 1. Pseudo-code of the KCTEP segmentation representation method.

```
Input: a set of stock time series data D
D = D1, D2, …, Di
traversal round T, T = 1, 2, …, i
define variable dir, Used to indicate the direction of the current trend, Initialized to 0.
define functions find_extrema(data), where data is the input data array.
define an empty array extrema, used to store found extreme points.
Output: All trend extremes in D.
Iterate through the data array, for each data point data[i]:
  if i = 0
    (i, data[i]) add to the extrema array and set dir to 1.
  then break
  if i is the last element of the array,
    (i, data[i]) add to the extrema array.
  if dir = 1,
    if data[i] <= price at previous extreme point,
      removes the previous extreme point from the extrema array and sets dir to −1.
    else
      add (i, data[i]) to the extrema array.
  else dir = −1,
      if data[i] >= price at previous extreme point,
        removes the previous extreme point from the extrema array and sets dir to 1.
      else
        add (i, data[i]) to the extrema array.
return extrema array
```

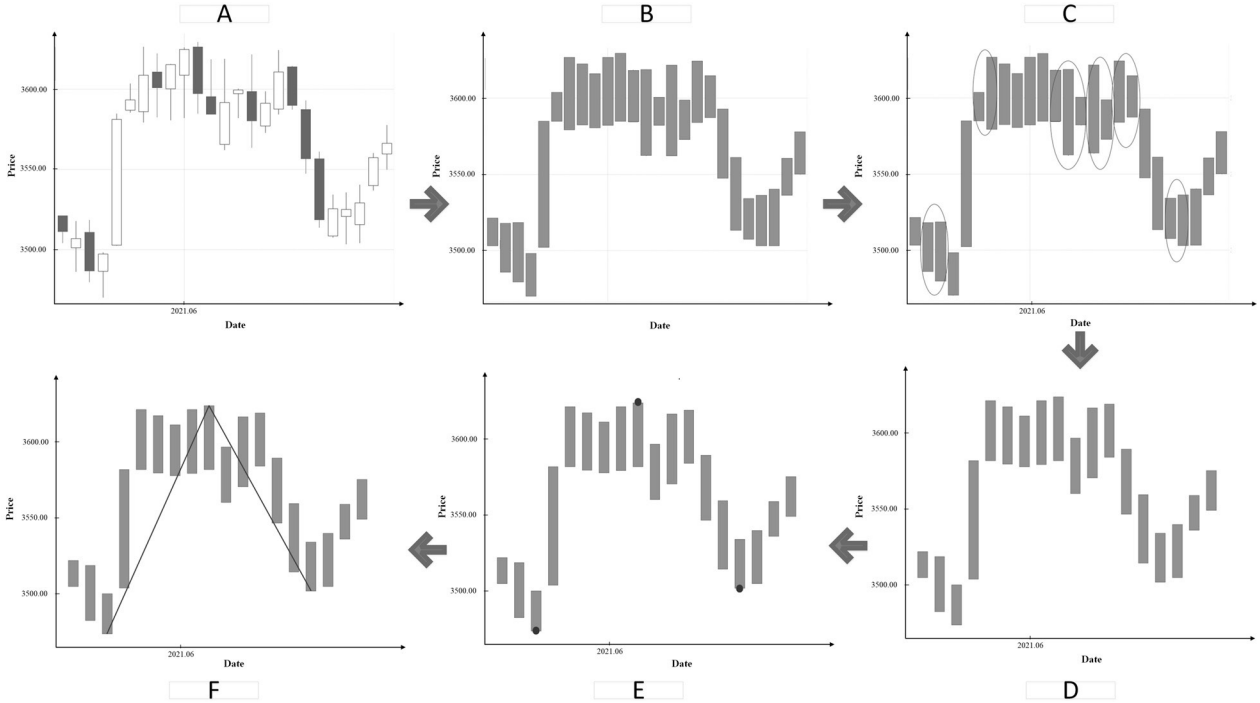

Figure 5. Trend extreme point extraction process.

2021 to June 23, 2021 from Wind, with a total of 25 K-lines. Figure 5 shows the process of identifying the trend extreme points.

As can be seen in Figure 5, the KCTEP algorithm is utilized to simplify the 25 K-lines in the time period into a 3-point representation, which greatly reduces the amount of data in the stock time series and lays the groundwork for other subsequent applications.

4. Analysis and Evaluation of Experimental Results

4.1. Experimental Process

4.1.1. Experimental Environment

In implementing the KCTEP algorithm, we chose a series of development components, and the specific choices of these components are shown in Table 1.

Table 1. Experimental environment setting.

Name	Configurations
Processor CPU	Intel(R) Core(TM) i7-8550U CPU @ 1.80 GHz 2.00 GHz
Memory RAM	DDR4 8 GB 2400MHz
Hard disk	128 GB Solid State Drive (Toshiba)
Operating System OS	Windows 10 Home Edition 19043.1889
Programming Language	Python3.5.0
IDE	JetBrains PyCharm 2018.3.2
Toolkits	Pandas NumPy et.al

4.1.2. Dataset

The FTSE China A50 Index contains the 50 largest companies in China's A-share market by market capitalization, and its total market capitalization accounts for about 35% of the total market capitalization of A-shares. Many investment institutions consider the FTSE China A50 Index to be the most representative index of China's A-share market. The stocks included in the FTSE China A50 Index are representative of the Chinese stock market and more fully reflect the specifics of each sector. All daily K-line price data of the 50 stocks of FTSE China A50 Index are exported using Wind to form a dataset from December 31, 1996 to December 31, 2021, with a total of 321,761 pieces of data, and each piece of data includes the opening price, the closing price, the high price and the low price.

4.1.3. Experimental Program

Experiment 1: In order to easily see the effect of extreme point extraction, the dataset is compared with the uniform extreme point method and the segmented aggregation approximation (PAA) extraction method using compression rate as an evaluation criterion using long time data and interval effects with different trends.

Experiment 2: The dataset was segmented using uniform extreme point segmentation representation, segmentation aggregation approximation (PAA), APCA method and the KCTEP method proposed in this paper, respectively, and the segmentation error was calculated using three distance metrics: Euclidean distance, plumb line distance and orthogonal distance, respectively, based on the number of trend extreme point segments.

4.2. Analysis of Experimental Results

4.2.1. Experiment 1 Results

This study uses K-line data from stock data for the experiment. The original dataset totaled 1,247,044 price data. At 5% level of significance, 143,535 extreme points were statistically obtained using the uniform extreme point method, 72,653 extreme points were found using the PAA method while 35,914 key points were found using the KCTEP method. In order to measure the degree of dimensionality reduction for the original data, the data compression rate is chosen as the evaluation index and the calculation is shown in Equation 1:

$$CR(\%) = \frac{w}{m} \cdot 100\% \tag{1}$$

where m denotes the amount of data in the original time series, w denotes the number of key points after compression, and obviously, the smaller the compression rate the greater the degree of dimensionality reduction for the data. The compression rate was 11.51% for the uniform extreme point method, 5.83% for the PAA method, and 2.88% for the trend extreme point method used in this study. Figure 6 shows a schematic diagram of the data after approximation for the fourth quarter of 2021, (b) shows the original data, (a) shows the approximation results using the PAA algorithm, (c) shows the approximation results using the Absolute Extreme Points algorithm, and (d) shows the approximation results of the KCTEP algorithm proposed in this paper. From the figure, it can be seen that the KCTEP segmentation representation method can better remove the noise data in the stock time series, and can retain the trend characteristics of the original data as well as describe the trend changes of the stock time series data with the minimum segmentation height, making it visually consistent with the trend of the original time series data, which proves the effectiveness of the KCTEP method.

4.2.2. Experiment 2 Results

Time series distance metric can intuitively reflect the degree of similarity between the sequence after feature representation and the original sequence, the smaller the distance metric, the better the fit between the sequence after feature representation and the original sequence, the better the original information of the time series can be retained. This experiment compares the advantages and disadvantages of the segmented representation of uniform extreme points, the segmented aggregation approximation (PAA) method, the APCA method,

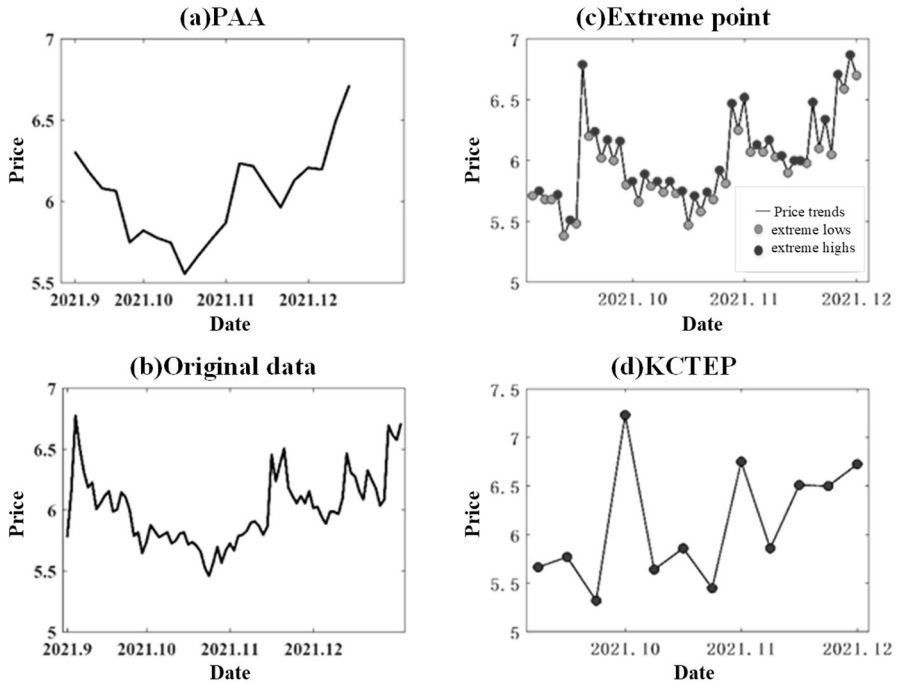

Figure 6. Comparison of experimental results.

and the three different distance measures (Euclidean, orthogonal, and plumb line distances) in the KCTEP method proposed in this chapter. The experiment uses the number of KCTEP method segments as the standard and calculates the average error on this number of segments as the basis for evaluation. The data of 000058. SZ from September 1, 2021 to December 31, 2021 is selected as the original data set for the experiment. Table 2 demonstrates the results of the quantitative comparison of the four methods in terms of distance. It can be seen that the KCTEP method outperforms the uniform extreme point method, the PAA method and the APCA method in terms of vertical, Euclidean and orthogonal distance metrics.

Table 2. Distance metric results.

	Euclidean distance	**Vertical distance**	**Orthogonal distance**
The uniform extreme point method	890.2479	9.4815	10.1210
PAA method	896.4368	8.9652	11.3649
APCA method	889.6124	8.4286	9.2657
KCTEP method	867.9871	7.5140	7.7596

As can be seen in Table 2, the Euclidean distance is much higher than the plumb line distance and the orthogonal distance. The reason for this phenomenon is that in the calculation of the distance from a point to a straight line, the Euclidean distance is primarily concerned with the sum of the distances to the two endpoints of the line and is affected by the length of the segment. Specifically, Euclidean distance is calculated by measuring the distance by calculating the Euclidean distance from a point to a straight line. However, since the Euclidean distance takes into account the sum of the distances of the two endpoints on a straight line, the Euclidean distance may be greater in the case of longer segment lengths. In contrast, perpendicular and orthogonal distances focus more on the perpendicular distance from a point to a straight line and are not affected by segment length.

5. Conclusion

In this study, we found the trend extreme points based on K-line combinations by re-representing stock K-lines and re-combining them using the relationship between K-line charts. Based on this, we proposed a segmentation representation method of stock time series data based on the trend extreme points of K-line combinations, KCTEP, and compared it with the uniform extreme point representation method, the PAA method, and the APCA method in terms of the evaluation indexes of the data compression rate and the segmentation error. KCTEP was compared with the uniform extreme point representation, PAA method and APCA method in terms of data compression rate and segmentation error. The study reached the following conclusions:

- The segmented representation of stock time series data based on K-line combination trend extreme points (KCTEP) proposed in this study significantly outperforms the uniform extreme points method, the PAA method, and the APCA method in all indicators. This means that the method proposed in this paper is able to compress the time series data while maintaining a low loss of information and is able to more accurately characterize the trend of the stock time series.
- The KCTEP method proposed in this study can capture trend characteristics more accurately in stock time series analysis, providing investors with a more reliable and precise basis for decision-making. The results of this research are of great significance to the understanding of the stock market and investment decisions.

However, it is important to recognize the limitations of this study. Our algorithm focuses on stock time series data, with the addition of adjustments and combinations of K-plots relative to other methods, with relatively high computational complexity, and the computational complexity of large datasets needs to be further investigated. In addition, while our method improves under different stock trend identification, its performance in extreme cases needs to be tested more extensively.

Future research directions include further testing of this method on high-frequency data or other stock datasets and indices to improve the generalizability of the performance method. In addition, integrating the method with machine learning models for trend prediction or improving its robustness to noisy markets are also important directions for future research.

Declaration of Competing Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This research work was not funded by any program.

Data availability

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

References

[1] X. Wang *et al.*, "Data-driven and Knowledge-based Predictive Maintenance Method for Industrial Robots for the Production Stability of Intelligent Manufacturing", *Expert Systems with Application*, vol. 234, pp. 121136.1–121136.20, 2023.
https://doi.org/10.1016/j.eswa.2023.121136

[2] B. Zhou *et al.*, "Semantic-aware Event Link Reasoning Over Industrial Knowledge Graph Embedding Time Series Data", *International Journal of Production Research*, pp. 4117–4134, 2023.
https://doi.org/10.1080/00207543.2021.2022803

[3] M. Dasoomi *et al.*, "Predicting the Choice of Online or Offline Shop** Trips Using a Deep Neural Network Model and Time Series Data: A Case Study of Tehran, Iran", *Sustainability*, vol. 15, no. 20, p. 14764, 2023.
https://doi.org/10.3390/su152014764

[4] H. Wu *et al.*, "Online Event-driven Subsequence Matching Over Financial Data Streams", in *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data, Paris France*, 2004, pp. 23–34.
https://doi.org/10.1145/1007568.1007574

[5] J. Pengtao *et al.*, "Error Restricted Piecewise Linear Representation of Time Series Based on Special Points", in *Proceedings of the 2008 7th World Congress on Intelligent Control and Automation, Chongqing*, 2008, pp. 2059–2064.
http://dx.doi.org/10.1109/WCICA.2008.4593241

[6] J. Lin *et al.*, "A Symbolic Representation of Time Series, with Implications for Streaming Algorithms", in *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, San Diego California*, 2003.
https://doi.org/10.1145/882082.882086

[7] N. Q. V. Hung and D. T. Anh, "An Improvement of PAA for Dimensionality Reduction in Large Time Series Databases", in *Proceedings of the PRICAI 2008: Trends in Artificial Intelligence, Lecture Notes in Computer Science*, 2008, pp. 698–707.
http://dx.doi.org/10.1007/978-3-540-89197-0_64

[8] E. Keogh, "Fast Similarity Search in the Presence of Longitudinal Scaling in Time Series Databases", in *Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence, Newport Beach, CA, USA*, 2002.
http://dx.doi.org/10.1109/TAI.1997.632306

[9] Y Y. Liu *et al.*, " Trend Feature Extraction Method for Time Series Based on Turning Point and Trend Segment", *Journal of Computer Applications*, vol. 40, pp. 92–97, 2020.
http://dx.doi.org/10.15888/j.cnki.csa.007978

[10] Y. Li *et al.*, "Piecewise Linear Representation Based on Time Series Volatility", *Computer Systems & Applications*, vol. 30, no. 6, pp. 300–305, 2021.
http://dx.doi.org/10.15888/j.cnki.csa.007978

[11] J. Lin *et al.*, "A Symbolic Representation of Time Series, with Implications for Streaming Algorithms", in *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, San Diego California*, 2003.
https://doi.org/10.1145/882082.882086

[12] Y. Sun *et al.*, "An Improvement of Symbolic Aggregate Approximation Distance Measure for Time Series", *Neurocomputing*, vol. 138, pp. 189–198, 2014.
https://doi.org/10.1016/j.neucom.2014.01.045

[13] C. Ji *et al.*, "A Piecewise Linear Representation Method Based on Importance Data Points for Time Series Data", in *Proceedings of the 2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Nanchang, China*, 2016, pp. 111–116.
http://dx.doi.org/10.1109/CSCWD.2016.7565973

[14] Y. Hu *et al.*, "A Novel Segmentation and Representation Approach for Streaming Time Series", *IEEE Access*, vol. 7, pp. 184423–184437, 2019.
http://dx.doi.org/10.1109/ACCESS.2018.2828320

[15] B. Lkhagva *et al.*, "New Time Series Data Representation ESAX for Financial Applications", in *Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06), Atlanta, GA, USA*, 2006.
http://dx.doi.org/10.1109/ICDEW.2006.99

[16] Y. Sun *et al.*, "An Improvement of Symbolic Aggregate Approximation Distance Measure for Time Series", *Neurocomputing*, pp. 189–198, 2014.
https://doi.org/10.1016/j.neucom.2014.01.045

[17] B. Zhang *et al.*, "Novel Symbolic Aggregate Approximation Approach for Time Series Based on Trend Features", *Journal of Computer Applications*, vol. 42, pp. 123–129, 2022.

[18] T. Pavlidis, "Waveform Segmentation Through Functional Approximation", *IEEE Transactions on Computers*, vol. C–22, no. 7, pp. 689–697, 1973.
http://dx.doi.org/10.1109/TC.1973.5009136

[19] E. Keogh *et al.*, "Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases", *ACM SIGMOD Record*, pp. 151–162, 2001.
https://doi.org/10.1145/375663.375680

[20] G. Reinert *et al.*, "Probabilistic and Statistical Properties of Words: An Overview", *Journal of Computational Biology*, vol. 7, no. 1–2, pp. 1–46, 2000.
https://doi.org/10.1089/10665270050081360

[21] R. Staden, "Methods for Discovering Novel Motifs in Nucleic Acid Sequences", *Bioinformatics*, vol. 5, no. 4, pp. 293–298, 1989. https://doi.org/10.1093/ bioinformatics/5.4.293

[22] J. Buhler and M. Tompa, "Finding Motifs Using Random Projections", in *Proceedings of the 5th Annual International Conference on Computational Biology, Montreal Quebec Canada*, 2001, pp. 69–76. https://doi.org/10.1145/369133.369172

[23] A. Bagnall and G. Janacek, "A Run Length Transformation for Discriminating Between Auto Regressive Time Series", *Journal of Classification*, vol. 31, no. 2, pp. 154–178, 2014. http://dx.doi.org/10.1007/s00357-013-9135-6

[24] R. Agrawal *et al.*, "Efficient Similarity Search in Sequence Databases", in *Foundations of Data Organization and Algorithms,Lecture Notes in Computer Science*, 1993, pp. 69–84. http://dx.doi.org/10.1007/3-540-57301-1_5

[25] A. G. Li and Z. Tan, "Dimensionality Reduction and Similarity Search in Large Time Series Databases", *Chinese Journal of Computers*, vol. 9, pp. 1467–1475, 2005.

[26] E. Fuchs *et al.*, "On-line Motif Detection in Time Series with SwiftMotif", *Pattern Recognition*, vol. 42, no. 11, pp. 3015–3031, 2009. https://doi.org/10.1016/j.patcog.2009.05.004

[27] E. Fuchs *et al.*, "Temporal Data Mining Using Shape Space Representations of Time Series", *Neurocomputing*, vol. 74, no. 1–3, pp. 379–393, 2010. https://doi.org/10.1016/j.neucom.2010.03.022

[28] P. Smyth, "Clustering Sequences with Hidden Markov Models", *Neural Information Processing Systems,Neural Information Processing Systems*, 1996.

[29] H. Chen *et al.*, "Model-based Kernel for Efficient Time Series Analysis", in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago Illinois USA*, 2013. https://doi.org/10.1145/2487575.2487700

[30] Z. Wang *et al.*, "Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline", in *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA*, 2017. http://dx.doi.org/10.1109/ijcnn.2017.7966039

[31] S. Lin and G. C. Runger, "GCRNN: Group-Constrained Convolutional Recurrent Neural Network", *IEEE Transactions on Neural Networks and Learning Systems*, pp. 4709–4718, 2018. http://dx.doi.org/10.1109/TNNLS.2017.2772336

[32] Y. Zheng *et al.*, "Exploiting Multi-channels Deep Convolutional Neural Networks for Multivariate Time Series Classification", *Frontiers of Computer Science*, pp. 96–112, 2016. http://dx.doi.org/10.1007/s11704-015-4478-2

[33] H. Wang *et al.*, "DAFA-BiLSTM: Deep Autoregression Feature Augmented Bidirectional LSTM Network for Time Series Prediction", *Neural Networks*, vol. 157, pp. 240–256, 2023. https://doi.org/10.1016/j.neunet.2022.10.009

Received: December 2024
Revised: December 2024
Accepted: December 2024

Contact addresses:
Lei Han*
School of Economics and Management
University of Science and Technology Beijing
Beijing
China
e-mail: D202210486@xs.ustb.edu.cn
*Corresponding author

Xuedong Gao
School of Economics and Management
University of Science and Technology Beijing
Beijing
China
e-mail: gaoxuedong@manage.ustb.edu.cn

Haining Yang
Linyi Vocational College
Linyi
China
e-mail: B20180394@xs.ustb.edu.cn

Lei Han is a current PhD student at the School of Economics and Management, University of Science and Technology Beijing, with research interests in financial time series analysis and data mining.

Xuedong Gao is a professor at the School of Economics and Management, University of Science and Technology Beijing, with research interests in management process optimization, financial time series analysis, and data mining.

Haining Yang received her Ph.D. in Management from University of Science and Technology Beijing in June 2023 and is currently a lecturer at Linyi Vocational University. His research interests are financial time series analysis and data mining.