Soft Computing (2023) 27:6329–6344
https://doi.org/10.1007/s00500-023-07915-5

DATA ANALYTICS AND MACHINE LEARNING

Conceptual-temporal graph convolutional neural network model for stock price movement prediction and application

Zhang Fuping[1]

Accepted: 6 February 2023 / Published online: 24 February 2023

Abstract
Stock price movement prediction is an important problem for trading decision-making. But it is a challenging task due to the nonlinearity and complexity of the stock trading data. This paper analyzes the linkage effect of price movement among stocks with the same concept segment through the dissipative structure theory, which is one of the major drivers for stock price movement. Considering the time-dimensional and concept-dimensional characteristics of the stock price movement, the stock conceptual-temporal network is constructed and the conceptual-temporal graph convolutional neural network model (CT-GCNN) is designed to map the linkage effect and predict the stock price movement. The experiment is conducted to validate the proposed model by utilizing the Chinese stock trading market data, which shows that CT-GCNN model outperforms the baseline deep learning models. Ten application cases are designed according to the conceptual quantity. The monthly highest stock yield is up to 16.276 and the lowest stock yield is 5.083, which reveals the stability and superiority of CT-GCNN model.

Keywords Conceptual properties · Temporal characteristics · Stocks price movement prediction · Conceptual-temporal graph convolutional neural network · Application

1 Introduction

The prediction of the stock price movement has become an important interest for investors due to the potential of obtaining a high return in the stock trading market (Chen et al. 2017). Stock price movements are influenced by both the exogenous information (the state information presented by the trading market, e.g. macro policies, hot events) and endogenous information (the operating state of the stock) (Li et al. 2014; Zhou et al. 2017). The series of complex and variable influence factors make stock price movement data present obvious nonlinearity, time-varying, and high-volatility characteristics, which pose a huge challenge to the task of stock price movement prediction.

With the rapid update of online investment information in Internet finance, it is easy for investors to obtain exogenous information, which could attract investors' attention to the relevant stocks (Zhao and Yang 2022; Milad and Seyed 2021). Due to the differences in investors' understanding of exogenous information, the stock trading decisions are not always consistent, leading to a disorganized operating state of stocks. The investors' trading decisions are ultimately determined by the endogenous information on the trading market, especially for short-term investors, who are more sensitive to the current state of operation for the stock.

Concept stocks are a category of stocks with special meaning and representativeness. The stocks with the same concept mean that all stocks are related to a specific event or industry. The concept stocks cover a much smaller area than the industry stocks. The conceptual quantity of one stock is determined by the corporate scope of the company and corresponding exogenous information. The impact of different concept attributes on the price movement of one stock tends to have a long continuity. Exogenous information would trigger a change in the operating state for the corresponding concept stocks, which has a demonstration effect. For example, stocks with semiconductor concept segment would have the same positive price movement due

✉ Zhang Fuping
zhangfuping@sdust.edu.cn

[1] College of Economics and Management, Shandong University of Science and Technology, Shandong, China

to macro policies supporting the development of the semiconductor industry. Similarly, endogenous information will also trigger the same price movement of stocks under a strong demonstration effect, where the leading stocks in the concept segment tend to bring the price movement of other stocks with the same concept to change in the same direction.

Conceptual attributes of stocks can trigger a linkage effect of the price movements among stocks within the same concept segment. However, at a certain period, the impact of different concept attributes on the price movement of one stock would be inconsistent. Because some concept segments are rising and have a positive demonstration effect on the stock, while other concept segments are falling and have a dampening effect on the stock. Given the limited attention of investors, it is worthwhile to examine how investors develop effective trading strategies based on the conceptual quantity of stocks, the characteristics of the conceptual attributes, and the current operating state of the stock.

Many researchers construct stock price movement trend-prediction models based on historical stock operating state information or exogenous information (Thi-Thu and Seokhoon 2019; Aliu et al. 2021; Ho and Huang 2021), which ignores the linkage effect of the price movement among stocks with the same concept segments. In this paper, we analyze the dissipative structure and characteristics of stock price movement trend operation to clarify the linkage relationship among stocks. Besides, considering the information of temporal dimension for the stock price operating state, a new deep learning architecture, the Conceptual-Temporal Graph Convolutional Neural Network (CT-GNN), is constructed, which could capture the relationship between the conceptual attributes of stocks, stock operating state, and stock price movement. Finally, the proposed CT-GCNN model is validated by utilizing Chinese stock trading market data. Our contributions are:

- The dissipative structure and characteristics of stock price movement trend operation are analyzed, which explains that one of the major drivers for the stock price movement is the linkage effect of the price movement among stocks with the same concept segment.
- The stock conceptual-temporal network is designed to portray the conceptual-graph structured interaction among stocks and express the cross-fertilization of exogenous information and endogenous information that influence the stock price movement.
- A novel deep learning architecture is proposed, which combines graph convolution with a sequence learning convolution network to predict the stock price movement by employing the stock conceptual-temporal network. The Chinese stock trading market is utilized to validate the effectiveness of the CT-GCNN model.

The remainder of this paper is organized as follows: Section 2 reviews the related work. And we propose the dissipative structure and characteristics of stock price movement trend operation in Sect. 3. Section 4 presents the CT-GNN model to predict the stock price movement. Section 5 describes the datasets and experiments to validate the effectiveness of the proposed model. In Sect. 6, one application case is conducted to further assess the role of the conceptual quantity on the proposed model, followed by a conclusion in Sect. 7.

2 Literature review

Stock price stands as a crucial instrument that generates signals for the trading strategies of investors. Stock price movement prediction is studied widely in many pieces of research, which could help investors make a trading decision appropriately and reduce the risk.

Traditional research employs information about the historical operating state of stocks and constructs the autoregressive integrated moving average (ARIMA), generalized autoregressive conditional heteroskedasticity (GARCH) volatility (Philip and Hendrik 1999), and the smooth transition autoregressive model (STAR) (Nicholas 2001) to predict the stock price movement (Fama and French 1996; Naranjo et al. 1998; Lettau and Ludvigson 2005; Santos and Veronesi 2006). Besides, the time-series models are proposed to learn the sequential data, which utilize a pre-selected mathematical model structure and existing sample data to fit the data distribution (Ticknor 2013; Zbikowski 2015; Patel et al. 2015; Rather and Agarwal 2015). Unfortunately, these models depend on the assumption of smoothness of time and need to select enough factors to calculate the intrinsic value of a target index, which are difficult to portray the nonlinear characteristics of the stock price movement, resulting in poor performance of these models.

In recent years, with the application of machine learning to the field of finance, machine learning models have been utilized to construct intelligent quantitative trading models (Mevlut and Imran 2006). The machine learning methods, such as SVM (Krizhevsky et al. 2017); fuzzy system (Kyunghyun et al. 2014); random forest (Basak et al. 2019); hybrid methods (Qing et al. 2005; Dennis and Charles 2003); 2-stage, end-to-end model (Zhou et al. 2019); and robust kernel extreme learning machine (Bisoi et al. 2019), have been widely employed in stock price movement prediction, which could achieve non-stationary data analysis. Chandar (2021) proposes a gray wolf optimization-Elman neural network model to predict future stock price movement. But these methods are difficult to model high-level feature representation and portray the intrinsic patterns of stock price movement.

Deep learning has achieved exciting performance in image recognition tasks and natural language processing tasks (Chen et al. 2019a; Zhipeng et al. 2022; Vasanthi and Seetharaman 2021; Venkatesh et al. 2021). Stock data are time-series multidimensional vectors, and more appropriate approaches should be taken in processing this special type of data. Many recent studies use deep neural networks for stock price movement prediction (Naik and Mohan 2020). Considering LSTMs perform well in time-series problems, they yield better advancement in stock price movement prediction than conventional ones (Nelson et al. 2017; Chen et al. 2019b; Zhao et al. 2020; Wu et al. 2020). LSTM model is good at capturing the serial characteristics of stock price movements in the temporal dimension, but it is difficult to learn the characteristics of price relationships, the spatial structure of endogenous factors.

As the CNN model could extract the implicit information, many researchers transform stock data into images and then extract multi-scale localized spatial features using CNN to predict the stock price movement (Zhang and Cai 2021; Hao and Gao 2020; Ding et al. 2015; Joshua et al. 2019). Long et al. (2020) construct three matrices to represent trading behavior patterns and to extract deep features by the CNN model. Omer and Ahmet (2018) convert stock technical indicators into 2D images and propose a novel algorithmic trading model based on CNN to predict the stock price movement. Wen et al. (2019) simplify noisy-filled financial temporal series via sequence reconstruction by leveraging motifs and then utilize CNN to capture the spatial structure of time series. Omer and Ahmet (2020) propose a 2D convolutional neural network to fit stock price movement by only using the 2D stock bar chart images. Silvio et al. (2020) generate Gramian angular field images from time series and exploit an ensemble of CNNs to predict the future price movement of stocks. CNN can only handle the data defined on regular grids and cannot be used to capture the complex topological structure (Zhou et al. 2019; Zhang et al. 2019).

Attempting to overcome these shortages, Kipf and Welling (2016) propose a novel deep learning method, named graph convolutional network (GCN). The interrelationship among the stocks or factors could contain valuable features, which is significant in predicting the stock price movement. Several studies have attempted to apply GCN models for the task of stock price movement prediction. Feng et al. (2019) capture the sector–industry relations and the wiki company-based relations and then propose the relational stock ranking method to obtain the sequential embeddings and the temporal graph convolution to account for the relational embeddings to rank the stock. Long et al. (2020) construct the companies' knowledge graph to capture the correlation between stocks. They utilize an attention-based bidirectional long short-term memory network (BiLSTM) to predict the stock price movement. Wu et al. (2020) utilize stock-related leading indicators to extract the time-series feature and construct a short form of stock sequence array convolutional neural network to predict the stock price movement. Based on the correlations between stocks, Chen et al. (2021) build stock market networks to represent the stock market information and propose a graph convolutional network and a dual-CNN to fit the stock price movement by capturing the stock market features and individual stock features. Hou et al. (2021) propose a hybrid deep learning pipeline VAE-GCN-LSTM model to incorporate the graph-structured relationship among firms to predict the stock price movement. Feng et al. (2022) put forward the detrended cross-correlation analysis coefficient to capture the relationship in the stock market and propose the relation-aware dynamic attributed graph attention network to rank the return ratio of stocks. Wu et al. (2022) convert market price series into graphs, which has the temporal points and the node weights. And then the structural information in the price graphs is embedded to predict the stock price movement. Cheng et al. (2022) capture the relationships from the financial events, news, and so on, and construct the graph structure. And they propose a multi-modality graph neural network to predict the stock price movement.

The above stock price movement prediction models have a more homogeneous perspective, which utilize various exogenous information and endogenous information to learn the characteristics of stock movements from temporal or spatial dimensions. For the research of linkage effect among stocks, recent research has constructed the knowledge graph according to the same-industry sector to learn the interaction linkage relationship among stocks. Nevertheless, the industry attributes are so broad that result in delayed reactions to information about macro-policies, hot events, demonstration effects, and other exogenous information. The knowledge graph expresses the weak linkage effect among stocks, so that it is difficult to portray the relationships and characteristics among stocks at a fine-grained level for these models.

The conceptual relationship among stocks is more segmented than the industry attributes, so that it is direct for the response of the conceptual segments to the exogenous information and it is readily apparent for the linkage effect among stocks within the conceptual segment. In this paper, we attempt to build the stock conceptual networks based on the conceptual relationship with the stocks as nodes and the concepts as edges. Motivated by graph CNN and convolutional sequence learning, we propose a novel deep learning architecture, the conceptual-temporal graph convolutional neural network (CT-GCNN) to predict the stock price movement through the conceptual-temporal network, which handles the graph-structured datasets and extracts the hidden information to retrieve the relationship with the neighbors. Through the conceptual convolution layer, the spatial structural characteristics of the stock conceptual relationship are extracted. And the temporal convolution layer is designed to combine efficient information about the features of the time sequence.

3 The dissipative structure and characteristics of stock price movement trend operation

Prigogine and Lefever (1968) propose dissipative structure theory to reflect a nonlinear open system far from an equilibrium state that continuously exchanges matter and energy with the external environment. When the system conditions change to a certain threshold, the system transforms from the previous disordered state to a new state that is temporally, spatially, and functionally more ordered, and which also needs to constantly exchange materials or energy with the outside world to maintain its structure.

The price movement trend of stocks indicates the reaction of the trading market to variations in the level of investor attention to the information. Due to the limited attention of investors, the investors reduce attention constraints through individual categorical thinking, one of the behavioral characteristics, which could help investors make the scarce resource of attention gain maximum benefit. Investors confirm the trading target though focusing on the conceptual properties of stocks, which is the behavioral characteristic that stems from individual categorical thinking. With the attention of investors flows, gathers, and removes among the different concept segments, the price movement trend of the stocks experiences a process of shocks, rises, falls, and then shocks. The stock trading market has a structure of investor attention dissipation clearly, which is mainly characterized by:

1. Openness. Different stocks with concept segments attract the attention of investors with different investment preferences. Based on the operating state of stocks, investors could make the buy–sell decision for any stock. The concept segments and the corresponding stocks keep releasing endogenous information to the outside world to gain the attention resources of investors and exchange the information energy with the outside world.
2. Far from equilibrium. As the difference of conceptual attributes and conceptual quantity for stocks, there is a significantly different in the investor attention that the stocks gained at a certain period. The imbalance of power between the long and short sides of the underlying stock results in stock price movement operating in disequilibrium without seasonal fluctuations.
3. Nonlinearity. The endogenous and exogenous factors influence the stock price movement nonlinearly. These factors interact and integrate in a nonlinear manner, which constructs a network of mutual influences and constraints between stocks and concept segments. The nonlinear mechanism of operation contributes to the basis of the evolution of the stock trading market.
4. Rising and falling characteristics. The corresponding concept segments and stocks related to the hot events would obtain more investor attention for a certain period, which results in significantly positive pressure on the price movement trend of these stocks in a short time. And the positive pressure would disappear or operate in reverse until the profit-taking or other hot events happen and the dissipation of investors attention on the concepts. The rising or falling of the stock price is both the way where the stock trading market system exists and the trigger of the system. The dissipative structure of the stock price movement trend operation is shown in Fig. 1.

The investor attention flows between different concepts, which would result in the corresponding operating directions of stock price movement. The linkage effect of the operating state among stocks with the same concept segment, the systematic dynamics of stock price movements, provides significant information for predicting the stock price movement. This paper would construct a novel deep learning method to predict the stock price movement by learning the mapping of the linkage effect.

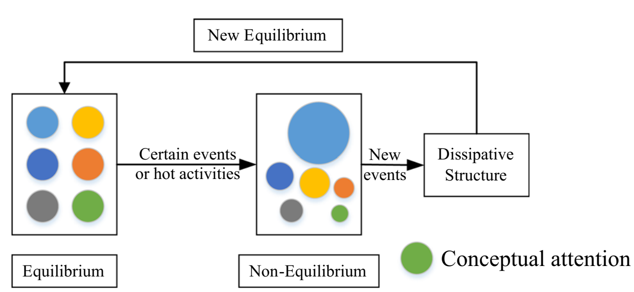


Fig. 1 The dissipative structure of stocks price movement trend operation

4 The conceptual-temporal graph convolutional neural network

4.1 The stock conceptual-temporal network

Consider S stocks, each with N trading variables of the stock market, including opening price, closing price, exchange rate, high price, low price, trading volume, and so on. The trading variables of stock i can be represented by a vector:

$$x_i = [TrdOpen_i,\ Trdclose_i, ExchangeRate_i, TrdHigh_i, TrdLow_i, TrdVol_i]. \tag{1}$$

Moreover, the variables could be portrayed from the N dimensions in M previous time steps, which can be regarded as the form of a vector $X{=}[x_1, x_2, \ldots, x_M]^{\mathrm{T}}$. To describe the conceptual relationship between neighboring stocks from the stock conceptual-temporal network, we introduce an undirected graph $G = (V, E, A)$ utilizing the trading variables, where V is the nodes of stocks, $|V| = S$, E represents the edges, indicating the conceptual relationship between the stocks, A is the adjacency matrix of G, and X is the features of nodes. Therefore, the whole stock data could be defined as G, which includes the M graph-structured data frame. And the stock conceptual-temporal network is shown in Fig. 2. It is different for the conceptual quantity that the two connected stocks belong to, which is reflected through the thickness of the conceptual relation edge.

Our goal is to predict the stock price movement, *uprange*, with the trading variables and the graph. With both conceptual and temporal features ready, we design a conceptual-temporal graph neural network model for the price movement of the stock predicting problem. And the model can be formulated as:

$$\widehat{uprange}_{M+T} = \arg\max p(uprange|x_1, x_2, \ldots, x_M, G), \tag{2}$$

where T is the predicting time step.

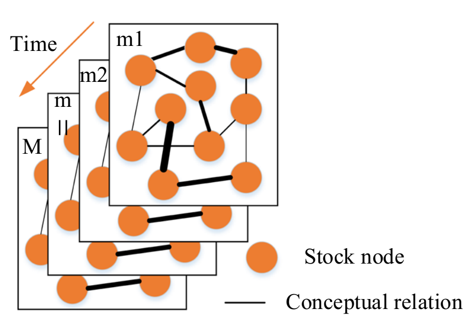


Fig. 2 The stock conceptual-temporal network

4.2 The Conceptual-Temporal Graph Convolutional Neural Network

CT-GCNN model is composed of one conceptual convolutional layer, one temporal convolutional layer, and one fully connected layer. The conceptual graph convolutional layer is utilized to learn the conceptual features among the stocks. The temporal convolutional layer is employed to extract the temporal dependence among the trading features of stocks. The fully connected layer is applied to integrate the comprehensive features to obtain the final output, the price movement of the stock, $\widehat{uprange}$. The input of the graph neural network is converted from the vector X into a 3-dimension tensor with the size of $[M, N, 1]$. And the framework of the model is shown in Fig. 3.

4.2.1 The conceptual graph convolutional layer

The layer is constructed to capture the hidden linkage information among stocks. The graph is $G = (V, E, A)$ with S nodes. The characteristics of stock nodes are $X = [X_1, X_2, \ldots, X_M]$, where $X_M{=}[x_M^1, x_M^2, \ldots, x_M^{C_{in}}]$. To obtain the overall information of conceptual relationships among stocks, we define two trainable kernels θ_h, θ_f of size $[C_{\text{in}}, C_{\text{out}}, K_s]$, where the K_s is the kernel size of graph convolution, C_{in} indicates the number of channels of input, C_{out} represents the number of channels of outputs, and X_M indicates the size of $S \times C_{\text{in}}$. Therefore, the convolution operation with a kernel tensor θ_h on X is shown as:

$$H_m = \left[h_m^1, h_m^2, \ldots, h_m^{C_{\text{out}}}\right], \quad \text{where} \quad h_m^j = \sum_{i,k} \theta_h^{ijk} L^k x_m^i. \tag{3}$$

And the convolution operation with a kernel tensor θ_f on X is shown as:

$$F_m = \left[f_m^1, f_m^2, \ldots, f_m^{C_{\text{out}}}\right], \quad \text{where} \quad f_m^j = \sum_{i,k} \theta_f^{ijk} L^k x_m^i \tag{4}$$

where $i = 1, 2, \ldots, C_{\text{in}}$, $j = 1, 2, \ldots, C_{\text{out}}$, $k = 0, 1, \ldots, K_s$, and $m = 1, 2, \ldots, M$. And the L^k represents the Laplacian of graph G. And $L = D - A$, the eigen decomposition of $L = U\Lambda U^{\mathrm{T}}$, where D is the diagonal degree matrix $D_{ii} = \sum_j A_{ij}$, and U is an orthogonal matrix.

Consequently, the final output of the conceptual convolution layer is represented as:

$$conceptual_{\text{out}} = [conceptual_1, conceptual_2, \ldots, conceptual_M], \tag{5}$$

$$conceptual_m = relu(H_m) \odot sigma(F_m) + X_m \odot (1 - sigma(F_m)), \tag{6}$$

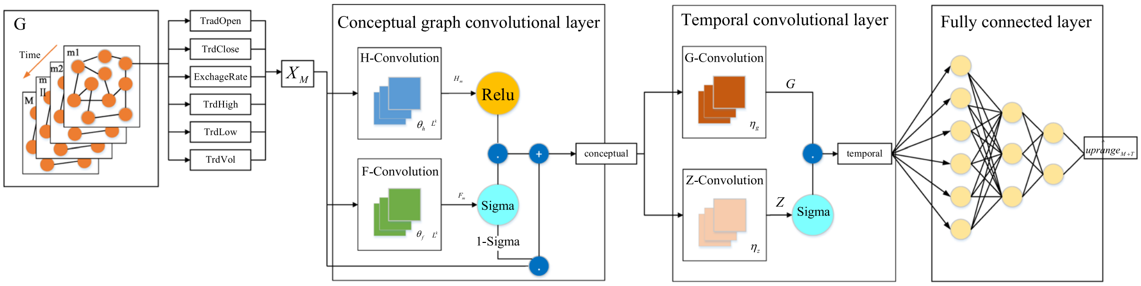


Fig. 3 The framework of the CT-GCNN model

where *relu* is the rectified linear units function, $\odot$ is the element-wise Hadamard product.

4.2.2 The temporal convolutional layer

The N variables in stock data are a time sequence. Motivated by the convolutional architecture from Facebook with the attention mechanism to tackle the sequences (Gehring et al. 2017), which avoids the ordered computational operation and achieves customized training procedures, we design the temporal convolutional layer to capture the temporal characteristics utilizing the gated linear units. The input of the temporal convolution layer is a 3-dimension tensor $conceptual_{out}$ with the size $[M, S, C_{\text{in}}]$, which stands for M time steps, S nodes, and C_{in} channels individually. Furthermore, we define the two training convolution kernels η_g, η_z of size $[K_t, 1, C_{\text{in}}, C_{\text{out}}]$ to exclusively apply on each time axis, where the K_{t} indicates the kernel size of graph convolution, C_{in} indicates the number of channels of input, and C_{out} represents the number of channels of outputs. Therefore, employing the kernels, the two convolution operations to the input:

$$conceptual = \left[conceptual^1, conceptual^2, \ldots, conceptual^{C_{\text{in}}}\right] \tag{7}$$

could be computed as:

$$g^j = \sum_i G = \left[g^1, g^2, \ldots, g^{C_{\text{out}}}\right], \quad conceptual^i \times \eta_g^{ij}, \tag{8}$$

$$Z = \left[z^1, z^2, \ldots, z^{C_{\text{out}}}\right], \quad z^j = \sum_i conceptual^i \times \eta_z^{ij} \tag{9}$$

where $i = 1, 2, \ldots, C_{\text{in}}$ and $j = 1, 2, \ldots, C_{\text{out}}$. Therefore, the output of the temporal convolution layer could be computed as:

$$temporal = G \odot sigma(Z). \tag{10}$$

4.2.3 The fully connected layer

According to the above conceptual and temporal convolution layer, we could utilize the historical features of stocks to capture the hidden information:

$$conceptual^M = \text{CN}\left(X^{M-T}, \theta_h, \theta_f\right), \tag{11}$$

$$temporal^{M+T} = \text{TM}\left(conceptual^M, \eta_g, \eta_z\right), \tag{12}$$

where CN indicates the conceptual convolution operation and TM represents the temporal convolution operation.

The fully connected layer is designed to compute the price movement of all stocks nodes, which is computed as:

$$\widehat{uprange}_{M+T} = temporal \times \omega + b, \tag{13}$$

where ω is the weight matrix and b is a bias.

The loss function of the CT-GCNN model for the next T time step could be computed as:

$$L\left(\{\text{X}\}_{i=1}^{M}; W\right) = \sum_{X_i} \left\| \widehat{uprange}_i, M + T\left(x_{i,1}, x_{i,2}, \ldots, x_{i,M}, W\right) - uprange_{i,M+T} \right\|^2, \tag{14}$$

where W are the whole trainable variables in the CT-GCNN model, $uprange_{i,M+T}$ is the real price movement of the stocks, and $\widehat{uprange}_i$ denotes the model-predicted stock price movement.

4.3 The evaluation of model

We utilize MSE and MAPE to evaluate the performance of the CT-GCNN model:

MSE(Mean Square Error): MSE

$$= \frac{1}{S} \sum_{i=1}^{S} \left(uprange_{i,M+T} - \widehat{uprange}_i\right)^2. \tag{15}$$

MAPE (Mean absolute percentage error): MAPE

$$= \frac{1}{S} \sum_{i=1}^{S} \left| \frac{uprange_{i,M+T} - \widehat{uprange}_i}{uprange_{i,M+T}} \right|. \tag{16}$$

4.4 The implementation using CT-GCNN model

In the CT-GCNN model, the conceptual linkage relationship and temporal features are captured to predict the stock price movement. The CT-GCNN model is trained by minimizing the loss function. Figure 4 presents the flowchart of the CT-GCNN model, whose steps are as follows.

Step 1 Data preprocessing. The stock conceptual-temporal network is constructed first, which includes the nodes, edges, and the features of nodes and edges. And then the data are preprocessed to meet the input and output format requirements of the model, which includes data normalization and the division of training data, validation data, and test data.

Step 2 CT-GCNN model initialization. Set the proper parameters and hyperparameters. Select the optional activation function of the model for the three layers.

Step 3 Training and validation. The stock price movement is computed through the initial CT-GCNN. Through calculating the loss function between the real value and the prediction value, the errors are backpropagation to update the weights. The validation data are utilized to obtain the rational parameters.

Step 4 Final CT-GCNN model performance test. After Step 3 is completed, the prediction performance of CT-GCNN model is evaluated using the test dataset.

5 Experiment

To the best of our knowledge, this work is the first one to incorporate the conceptual and temporal relationships among stocks to predict stock price movement.

5.1 Data description

The ordinary A-shares stocks in the Chinese stock trading market are utilized to conduct the experiments. Stocks and the conceptual properties data (the trading day from 01/01/2021 to 04/30/2022) are obtained as sample data from the Eastmoney Trading System (https://www.eastmoney.com/). The price movement trend of the ordinary A-shares stocks is limited between -10% and 10% daily. To maintain the consistency of the data, the stocks with abnormal trading data are excluded. Finally, there are 673, 600 data with 2, 105 stocks and 226 conceptual properties. To verify the generalization ability of the model, the data are divided into training data, validation data, and test data according to the ratio of 7:2:1.

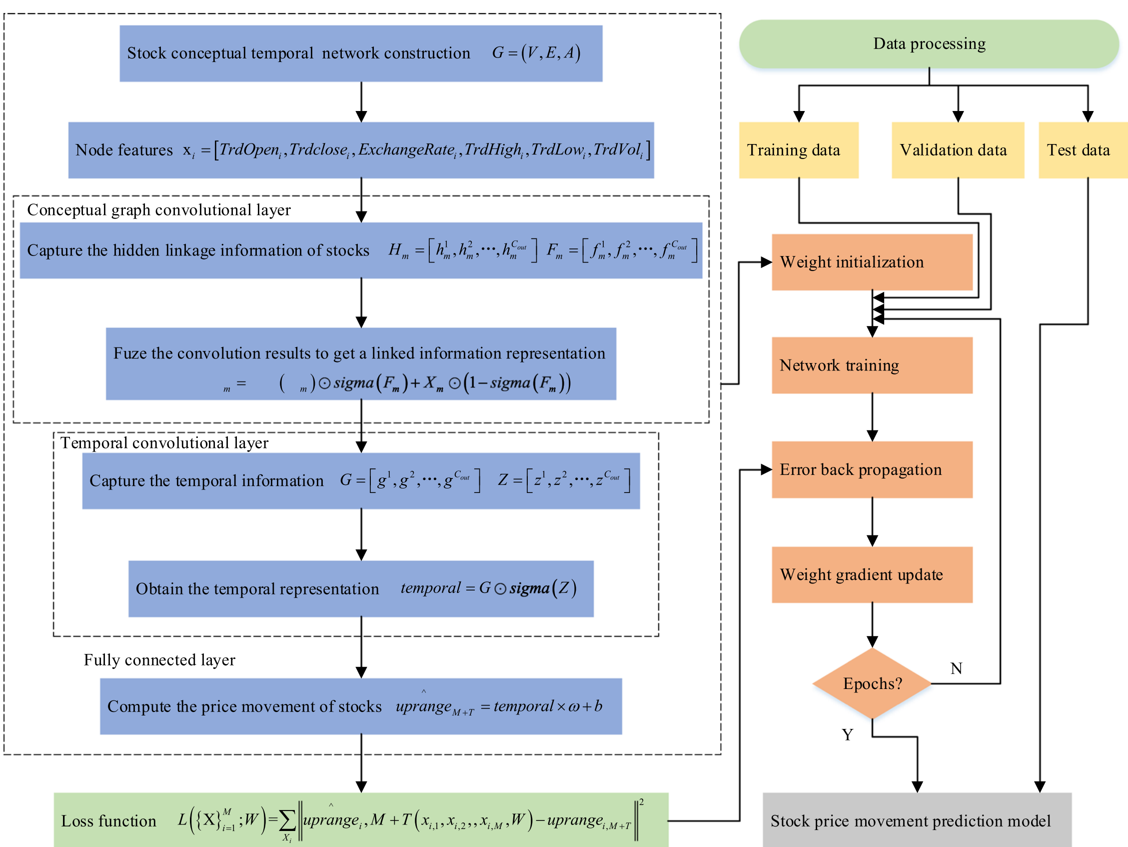


Fig. 4 The flowchart of CT-GCNN model

5.2 Data preprocess

The stock conceptual-temporal network is constructed by connecting stocks of the same concept according to the conceptual category from the Eastmoney Trading System. The stocks are the nodes and the conceptual quantity between the two stocks belongs to the edge. Then the constituted stock conceptual-temporal network at a certain time is shown in Fig. 5.

From the stock conceptual-temporal network, the adjacency matrix is obtained for the input. To learn the temporal features, the data are prepared in the time-sequential format where the time step is 7.

5.3 The initialization and training of hyperparameters for the CT-GCNN model

For the conceptual graph convolutional layer, two convolutional kernels are set up to explore the conceptual features. The 6-D features are learned to 64-D. The output is input in the temporal convolutional layer; the kernel size is set up as (1,2). Through the 2-D convolution, the temporal features are extracted, and the output is the 64-D representation. The fully connected layer has 3 outputs, which are the stock price movement of stocks in the future 3 trading days. The model hyperparameters are determined by the experimental trial-and-error method. The accuracy rate on the validation set is compared with different values of hyperparameters. The value with the highest accuracy rate is employed as the hyperparameter value for the training of the CT-GCNN model to assess the quality of user creative ideas. The learning rate is set as 0.001, the model is trained 50,000 times, and Adam is chosen as the optimizer.

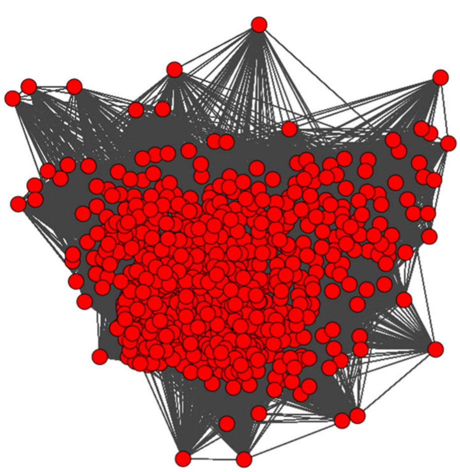


Fig. 5 The stock conceptual-temporal network

5.4 The comparison with other deep learning models

We compare the following stock price movement predicting models with the CT-GCNN model:

BP (Back Propagation) (Zhou et al. 1997): this model can learn the relationship among the inputs and outputs by the error backpropagation algorithm. We construct the compared model based on the BP model with four layers. And the number of neurons in the hidden layer is 64, 128, and 64, respectively. The input is the sequential feature dataset, and the output is the price movement of stocks.

CNN (convolutional neural network) (Gonzalez 2018): the model could regard the picture as 3D data with width, height, and depth. We consider the sequential feature as the 3D data, where the time step is the depth, the feature dimension is the width, and the time series is the height. Therefore, we construct the model based on CNN to predict the price movement of stocks. We set three convolutional layers to capture the hidden information of features. And the channel is 64.

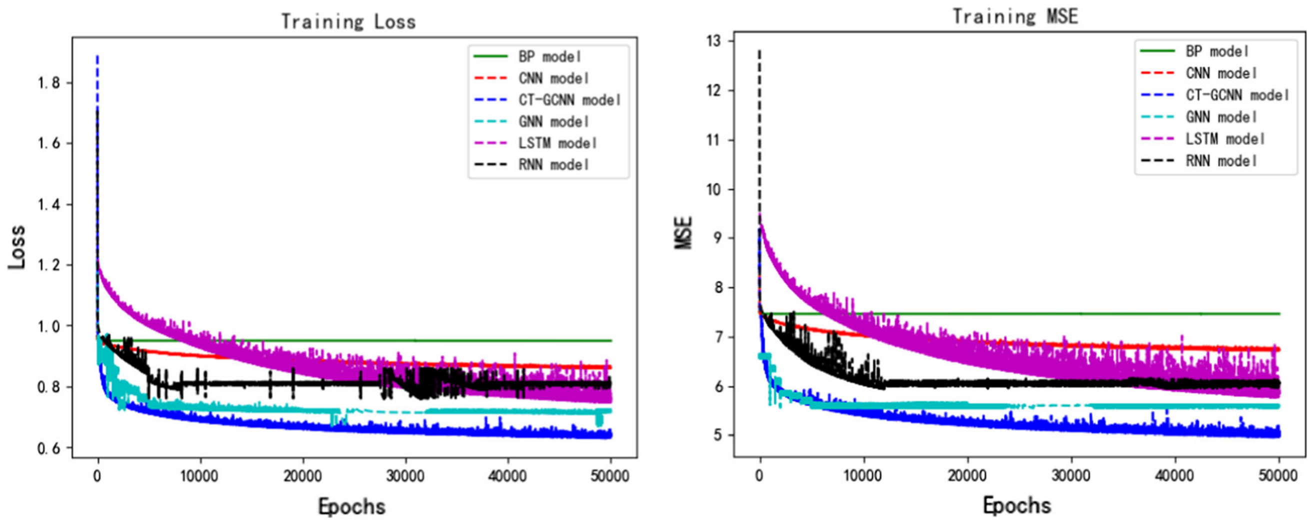
RNN (recurrent neural network) (Wojciech et al. 2014): the model takes the sequence data as input, recursion is performed in the evolutionary direction of the sequence and all recurrent units are connected in a chain, which is the performance for the non-linear characteristics. Therefore, the RNN model is built to predict while considering the features as sequential data. The number of features in the hidden state is 64 and the number of features in the hidden state is 2.

LSTM (long short-term memory) (Hochreiter et al. 1997): the model could solve the problem of long-term dependency on sequence data, where the gate structure allows information to be passed on in a sequence chain. The LSTM model is constructed to predict the price movement of stocks for comparison. The number of features in the hidden state is 64 and the number of features in the hidden state is 2.

GNN (graph neural network) (Kipf and Welling 2016): the model can use neural networks to learn graph-structured data, extract and uncover features and patterns in graph-structured data. We view the sequence feature as the features of nodes.

Figure 6 illustrates the change of loss function for the six models in the training procedure.

From Fig. 6, we have the following observations: On the training set, the CT-GCNN model has the lowest loss error (0.6449) and the lowest MSE (5.0773926), followed by the GNN model (loss: 0.7231 and MSE: 5.5989), followed by the LSTM model (loss: 0.7555 and MSE: 5.8298067),



(a) the change of loss function

(b) the change of MSE

Fig. 6 The training procedure of the six models

followed by the RNN model (loss: 0.8053 and MSE: 6.00564), followed by the CNN model without the conceptual relations among the stocks (loss: 0.8616 and MSE: 6.7282867), followed by the BP model (loss: 0.9500 and MSE: 7.13). The loss of the CT-GCNN model stabilizes to 0.6 and the MSE stabilizes to 5 at around 45,000 times.

To further validate the performance of the six models, the MSE, MAPE, and time calculated in the test data are shown in Table 1.

From the comparison, the CT-GCNN model outperforms others in test data. Compared with the GNN model, the CT-GCNN model could capture the temporal features of stocks, the MSE and MAPE have been reduced considerably, which reveals that the temporal features play an important role in predicting the price movement of stocks. The LSTM model and the RNN model can learn the temporal features but cannot learn the expression of conceptual relationships among stocks, thus performing worse than the CT-GCNN model. And both the CNN model and the BP model convert the temporal features into simple input, which could not represent the original sequential feature.

Table 1 Performance of the compared models in the test data

Model	MSE	MAPE	Time
CT-GCNN	0.405855888	0.47298035	21 s
GNN	5.641970921	3.642262768	20 s
LSTM	1.801380356	1.578209955	19 s
RNN	1.487050373	1.409820062	21 s
CNN	1.223643301	1.234103029	16 s
BP	1.148404236	1.087239931	13 s

Therefore, both the conceptual relationship among the stocks and the temporal features are significant factors to predict the stock price movement. And CT-GCNN model performs at its best considering both features. From the time, in the same hardware environment, the more complex the model, the more time it takes. The CT-GCNN model takes 21 s to run one time. These compared models are performed by GPU, the hardware is 1 NVIDIA A10 graphics processor with 24G graphics memory, 2 Intel Xeon Gold 5218 CPUs, 256G Memory, and 1 T SSD Fast Drive. And the development environment is built by Anaconda for Python programming experiments. The Anaconda version is 4.10, and the Python version is 3.8. The deep learning framework is torch 1.11.

According to Li et al. (2022), we conduct Wilcoxon signed-rank test and Friedman test to ensure the significance of the proposed model's contribution, where the former is used to make simple pairwise comparisons and the latter is effective for making multiple comparisons (Derrac et al. 2011). These two one-tail tests are both performed with a significance level of $\alpha = 0.005$ (Zhang and Hong 2021). The test dataset is utilized in the significance test. In the Wilcoxon signed-rank test, the original hypothesis is that no significant difference exists between the predictive accuracies of the two models. In the Friedman test, the original hypothesis is that no significant variation exists among the predictive accuracies of all compared models. While the p-value of the tests is less than 0.05, the original hypothesis is rejected. Table 2 presents the results of the Wilcoxon signed-rank test and Friedman test.

In Table 2, based on the Friedman test, the proposed CT-GCNN model receives the significance than other

Table 2 Results of the Wilcoxon signed-rank test and Friedman test

Compared models	Wilcoxon signed-rank test		Friedman test
	W	*p* value	*p* value
CT-GCNN vs. GNN	265,554,595.000	0.000	H_0:$e_1 = e_2 = e_3 = e_4 = e_5$
CT-GCNN vs. RNN	261,854,705.000	0.000	$F = 912.688$,
CT-GCNN vs. LSTM	267,207,249.500	0.000	$p = 0.000$***,
CT-GCNN vs. CNN	244,327,833.500	0.000	(reject H_0)
CT-GCNN vs. BP	254,842,225.500	0.000	

alternative models. Based on the Wilcoxon signed-rank tests, the proposed CT-GCNN model outperforms the other compared models.

6 Application cases

The strength of the relationship between the concepts of stocks is uneven. There is different influence relation between the conceptual quantity and the stock price movement. To discuss the problem and investigate the application situation of CT-GCNN model, the application cases are conducted by utilizing the trading data from 05/01/2022 to 06/30/2022. The whole conceptual quantity of stocks is attempted to discuss. To compare and analyze, the impact of two terms is investigated, including the conceptual quantity greater than the specified value and equal to the specified value. While the conceptual quantity is greater than 4, there are 738 stocks. And there are 139 stocks when the conceptual quantity is greater than 5. It is obvious that the larger the conceptual number is, the smaller the number of stocks that fall into this category. While the number of concepts is greater than 5, the number of stocks is too small, which would result in the model's lack of representativeness and practical significance. The conceptual quantity of stocks is set lower than 5.

The wonderful model could bring a larger stock yield. To validate and compare the different stock yields of stocks with different conceptual quantities, this part sets up the buy–hold–sell strategy for the trading date of two months, which would provide the basis for developing trading strategies. For the two terms, we design five models, respectively, which are trained with the same parameters and hyperparameters. And the 10 models are shown in Table 3.

6.1 The buy–hold–sell strategy

On each trading day M + T, we simulate an investor utilizing the stock price movement that the model predicted to decide the trading strategy in the following way:

1. buy while the market at the end of trading day M: The investor employs the method to obtain the predicted value for the price movement of stocks and sorting the stocks in descending order by the positive predicted value. Considering the transaction cost factor, we buy the stocks with a predicted price movement greater than 1% and a closing increase of less than 9.9% at the end of the trading day.
2. sell while the market at trading day $M + T$: the investor sells the stock purchased on trading day M. We design the 10 selling ways to calculate the stock yield, which include: (a) the stocks are sold on the closing price at the end of trading day $M + T$, which ignores the rate of price movement; (b) the stocks are prioritized selling while the price rises to the specified value at the trading period $M + T$, otherwise are sold on the closing price at the end of trading day $M + T$. According to the risk appetite of the investment, the specified value is 1%, 2%, 3%, 4%, 5%, 6%, 7%, 8%, and 9%, respectively.

6.2 The quantity of conceptual relation and the specified value

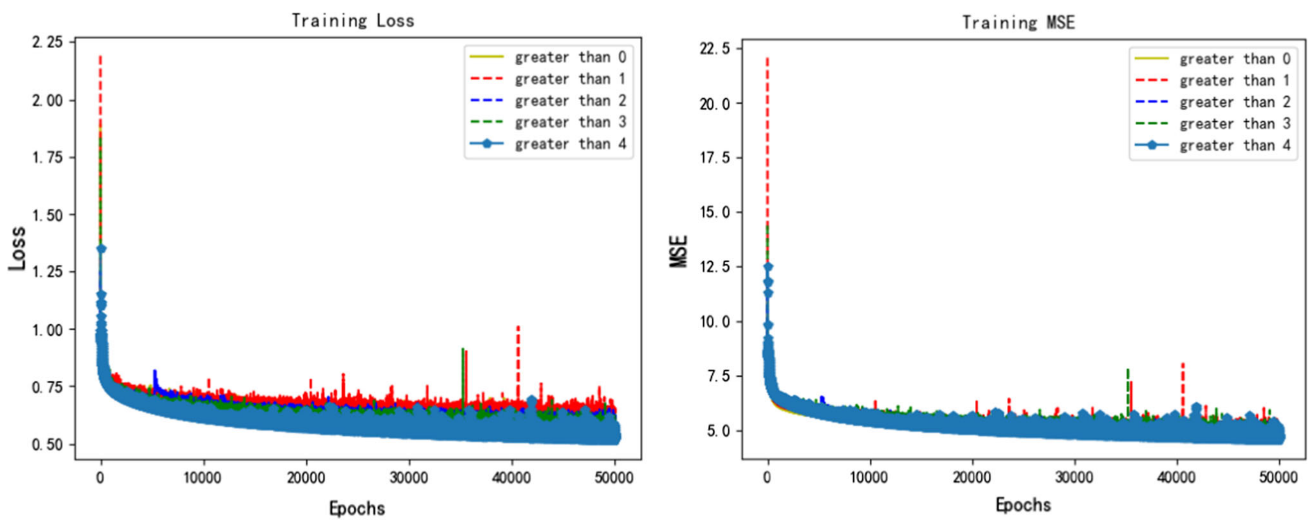
We train the model with the conceptual quantity of stocks greater than 0 to 4 and equal to 1 to 5, respectively. The change of loss function and MSE for five Greater Models in the training procedure are shown in Fig. 7 and five Equal Models are shown in Fig. 8. The ten ways of stock yield calculation for five Greater Models are shown in Table 4 and five Equal Models are shown in Table 5, where the best-performing results are highlighted with boldface. The change in stock yield for five Greater Models can be seen in Fig. 9 and for five Equal Models is seen in Fig. 10.

From the performance of the six models, we can conclude that:

1. For five Greater Models, Greater Model 3 and Greater Model 5 outperform others. In May, the highest stock yield is 11.376 and the lowest stock yield is 7.108. In June, the highest stock yield is 12.148 and the lowest stock yield is 8.725.

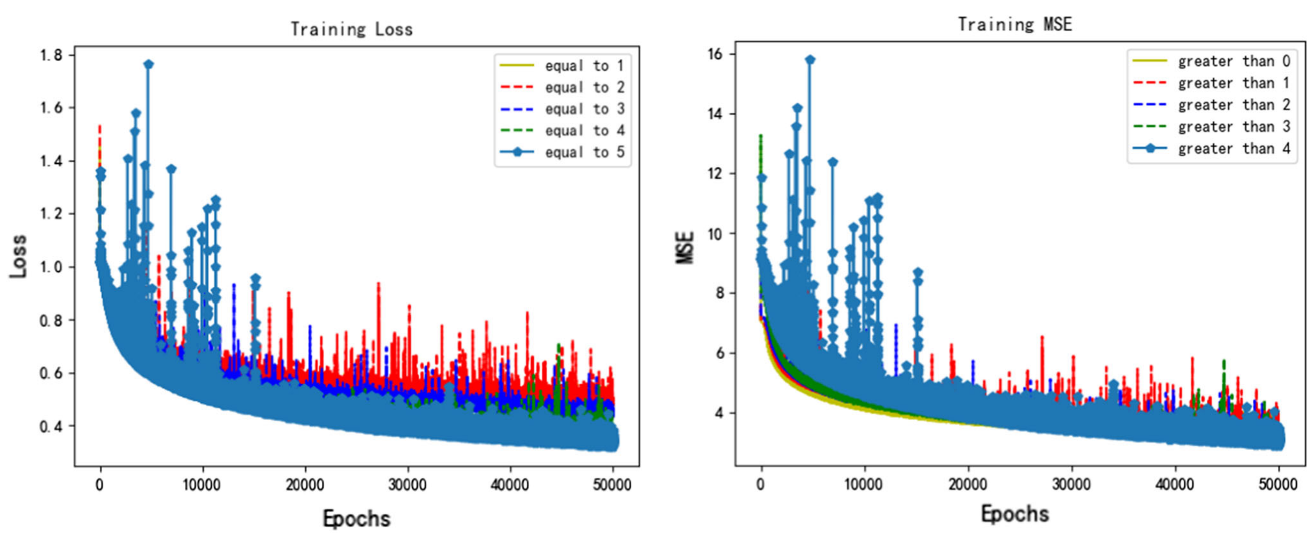
Table 3 The designed models and meanings

Model	Meanings
Greater Model 1	The conceptual quantity of the stocks is greater than 0
Greater Model 2	The conceptual quantity of the stocks is greater than 1
Greater Model 3	The conceptual quantity of the stocks is greater than 2
Greater Model 4	The conceptual quantity of the stocks is greater than 3
Greater Model 5	The conceptual quantity of the stocks is greater than 4
Equal Model 1	The conceptual quantity of the stocks is 1
Equal Model 2	The conceptual quantity of the stocks is 2
Equal Model 3	The conceptual quantity of the stocks is 3
Equal Model 4	The conceptual quantity of the stocks is 4
Equal Model 5	The conceptual quantity of the stocks is 5



(a) the change of loss function (b) the change of MSE

Fig. 7 The training procedure of five Greater Models



(a) the change of loss function (b) the change of MSE

Fig. 8 The training procedure of five Equal Models

Table 4 The stock yield of different selling strategies for five Greater Models

Time	May					June				
Greater than	0	1	2	3	4	0	1	2	3	4
Rise 1% yield	1.420	1.789	3.903	3.047	4.414	1.433	1.009	-0.237	1.513	0.227
Rise 2% yield	4.198	4.873	7.700	6.499	7.678	4.708	4.799	3.521	5.127	4.814
Rise 3% yield	5.236	6.068	9.072	6.336	7.922	6.696	7.141	5.344	7.209	7.220
Rise 4% yield	5.800	6.952	9.707	6.954	8.098	7.861	8.312	6.336	8.454	8.675
Rise 5% yield	6.112	7.525	10.246	7.289	8.260	8.399	8.648	6.820	8.935	9.539
Rise 6% yield	6.538	8.093	10.680	7.908	8.842	9.040	9.473	7.753	9.857	10.389
Rise 7% yield	6.868	8.495	11.058	8.277	9.216	9.366	10.005	8.193	10.068	11.093
Rise 8% yield	7.101	**8.845**	11.306	**8.625**	**9.248**	9.586	10.256	8.462	10.386	11.749
Rise 9% yield	**7.108**	8.647	**11.376**	8.470	9.053	**9.645**	**10.462**	**8.725**	**10.594**	**12.148**
Closing yield	6.542	7.941	10.774	7.977	8.466	8.978	9.786	8.206	9.895	11.434

Table 5 The stock yield of different selling strategies for five Equal Models

Time	May					June				
Equal to	1	2	3	4	5	1	2	3	4	5
Rise 1% yield	1.599	2.813	2.816	1.282	2.516	3.242	3.277	1.377	0.361	3.232
Rise 2% yield	4.785	5.736	6.370	4.628	5.682	4.628	7.657	6.150	4.920	8.472
Rise 3% yield	6.934	4.292	8.047	5.684	6.867	5.687	10.199	7.742	7.684	11.636
Rise 4% yield	7.531	4.314	9.081	6.583	7.092	7.404	10.792	9.432	9.118	13.579
Rise 5% yield	8.187	4.566	9.602	7.371	9.013	**8.632**	11.298	9.278	10.466	14.007
Rise 6% yield	8.615	4.822	9.851	8.060	**10.270**	8.291	11.871	9.686	12.030	15.308
Rise 7% yield	9.060	4.916	9.876	8.090	9.689	8.502	12.240	**10.229**	12.741	15.786
Rise 8% yield	9.264	**5.083**	10.201	8.164	9.771	8.411	12.476	10.006	12.514	**16.276**
Rise 9% yield	**9.283**	4.903	**10.540**	**8.435**	9.214	8.540	**12.664**	9.490	**12.774**	16.071
Closing yield	8.720	4.554	9.784	7.947	8.115	8.068	12.002	8.815	12.408	15.676

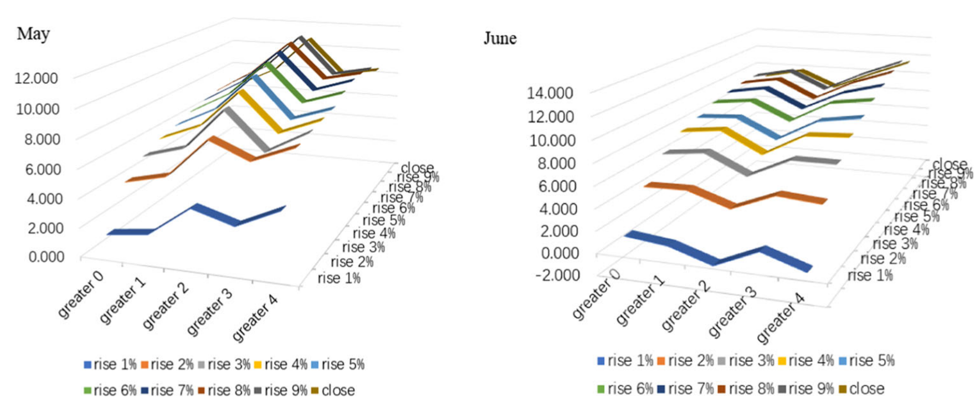


Fig. 9 The change in stock yield for five Greater Models

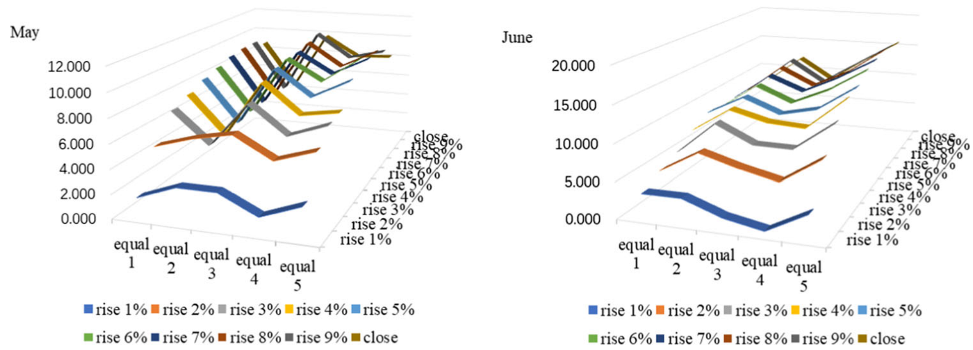


Fig. 10 The change in stock yield for five Equal Models

2. For five Equal Models, Equal Model 3 and Equal Model 5 outperform the others. In May, the highest stock yield is 10.540 and the lowest stock yield is 5.083. In June, the highest stock yield is 16.276 and the lowest stock yield is 8.632. It should be noted that the broad market trend in May is 4.89 and in June is − 0.67. Both the highest stick yield in May and June are similar, which indicates the CT-GCNN model is stable whether the stock trading environment is positive or negative.
3. Five Greater Models perform the highest stock yield, while the price rises to 9% in both May and June. And five Equal Models perform the highest stock yield, while the price rises to 9% in May and 8% in June. Besides, while selling at the close price, the stock yield would be the middle between the highest and lowest.

6.3 The Conclusion

The above applications prove the stability and superiority of CT-GCNN model again through the perspective of stock yield. The following are the specific findings:

1. The stock yield of the model with the conceptual quantity relation equal to the specified value is slightly higher than the model with the conceptual quantity greater than the specified value, which could be contributed to the unstable operating state of stock price movement. Usually, the smaller the conceptual quantity for the targeted stock, the smaller the impact of the price movement of other stocks on it. While the conceptual quantity increases, the impact of the price movement of other conceptual-connected stocks on it in the trading market is more and more. In this situation, some conceptual relations have a positive impact on the price movement of the targeted stock, some are negative, which would lead to partly mutual offset and make the price movement of the targeted stock more directionless. Therefore, the more the conceptual quantity that the stock belongs to, it would be inconsistent with the operating direction of different concept segments, which would aggravate the instability in stock price movements.
2. For different selling strategies, with the increase in the specific value that the price rises to in the trading day, the stock yield is increasing at the same time. Therefore, in terms of investment sub-risk preferences, a higher risk is associated with higher returns, and the stock yield is roughly in the middle for the sell-on-close strategy.
3. The stock price movement is driven by the operating direction of concept segments. While the concept is recognized by the market, the stocks with the concept may appear to have a certain increase, and this increase would have a certain degree of sustainability in the short term.

7 Conclusion

In this paper, we propose a conceptual-temporal graph convolutional neural network model (CT-GCNN) to predict stock price movement. Firstly, we analyze the dissipative structure and characteristics of stock price movement trend operation, which explores how the conceptual characteristics of the stocks affect the stock price movement. Then, we design the stock conceptual-temporal network to portray the stocks. Through the conceptual-temporal network, we propose a deep learning framework model, a conceptual-temporal graph convolutional neural network model (CT-GCNN) to predict the stock price movement, which includes three layers, the conceptual graph convolutional layer to capture the hidden conceptual information, the temporal convolution layer to learn the sequential information of the features, and the fully connected layer to integrate the conceptual graph convolutional layer and the temporal convolutional layer to output the price movement of stocks. We employ the Chinese stock trading market data to validate the effectiveness of the CT-GCNN model. CT-GCNN model is compared with the other deep learning-based model such as BP, CNN, RNN, LSTM, and GNN. The CT-GCNNA model has the lowest MSE and loss function in both training data and test data, which indicates the superiority of CT-GCNN model. The conceptual relationships among the stocks could support the prediction of price movement, and CT-GCNN model could represent and learn the conceptual relationship and the hidden temporal information.

Besides, we design two aspect application cases to discuss the conceptual quantity and CT-GCNN model, including the conceptual quantity greater than the specified valve and equal to the specified value. Eleven selling strategies are made to experiment with the stock yield of these models. We find that the five Equal Models outperform the five Greater Models, which reveals that with the increase of conceptual quantity that the stocks belong to, the inconsistency and the instability in stock price movements would be aggravated. Whether the broad market trend is rising or falling, the calculated stock yields are similar based on the predicted stock price movement. This discovery indicates the stability of CT-GCNN model.

Because any stock trading market is influenced by national policies, hot events, and other factors, the stock trading market exists a linkage effect among stocks, the operating state of concept segments would drive the stock price movement. In the future, we would further research the following aspects: (a). the purity of a specific stock in a conceptual relationship network can be considered. If the main business of one stock belongs to a concept, and this concept is recognized by the market; then, the stock is easily recognized by the market. However, if the little business of one stock belongs to the concept, which means the concept that the stock belongs to is not pure and the linkage with the price movement of the concept is not strong. (b). The concept index can be used as the base data as the input feature of the model. (c). The stocks in the different stock trading markets would be employed to further validate and improve the proposed model.

Author contributions All authors contributed to the study conception and design.

Funding The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data availability The datasets are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors state that there is no conflict of interest with the publication of this paper.

Ethical approval This article does not contain any studies with human participants performed by any of the authors.

References

Aliu F, Nadirov O, Nuhiu A (2021) Elements indicating stock price movements: the case of the companies listed on the V4 stock exchanges. J Bus Econ Manag 22(2):503–517

Basak S, Kar S, Saha S, Khaidem L, Dey SR (2019) Predicting the direction of stock market prices using tree-based classifiers. North Am J Econ Finance 47:552–567

Bisoi R, Dash PK, Parida AK (2019) Hybrid variational mode decomposition and evolutionary robust kernel extreme learning machine for stock price and movement prediction on daily basis. Appl Soft Comput 74:652–678

Chandar SK (2021) Grey Wolf optimization-Elman neural network model for stock price prediction. Soft Comput 25(1):649–658

Chen YJ, Chen YM, Lu C (2017) Enhancement of stock market forecasting using an improved fundamental analysis-based approach. Soft Comput 21(13):3735–3757

Chen L, Hu X, Tian W, Wang H, Cao D, Wang FY (2019a) Parallel planning: a new motion planning framework for autonomous driving. CAA J Autom Sin 6(1):236–246

Chen MY, Liao CH, Hsieh RP (2019b) Modeling public mood and emotion: stock market trend prediction with anticipatory computing approach. Comput Hum Behav 101:402–408

Chen W, Jiang M, Zhang WG, Chen Z (2021) A novel graph convolutional feature based convolutional neural network for stock trend prediction. Inf Sci 556:67–94

Cheng D, Yang F, Xiang S, Liu J (2022) Financial time series forecasting with multi-modality graph neural network. Pattern Recogn 121:108218

Dennis O, Charles M (2003) Neural network forecasts of Canadian stock returns using accounting ratios. Int J Forecast 19(3):453–465

Derrac J, Garcia S, Molina D (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18

Ding X, Zhang Y, Liu T, Duan J (2015) Deep learning for event-driven stock prediction. In: Proceedings of the twenty-fourth international joint conference on artificial intelligence (IJCAI), pp 2327–2333

Fama EF, French KR (1996) Multifactor Explan Asset Pricing Anom. J Financd 51(1):55–84

Feng FL, He XN, Wang X, Luo C, Liu YQ, Chua TS (2019) Temporal relational ranking for stock prediction. ACM Trans Inf Syst 37(2):1–30

Feng S, Xu C, Zuo Y, Chen G, Lin F, XiaHou J (2022) Relation-aware dynamic attributed graph attention network for stocks recommendation. Pattern Recogn. https://doi.org/10.1016/j.patcog.2021.108119

Gehring J, Auli M, Grangier D, Yarats D, Dauphin YN (2017) Convolutional sequence to sequence learning. arXiv:1705.03122

Gonzalez RC (2018) Deep convolutional neural networks. IEEE Signal Process Mag 35(6):79–87

Hao Y, Gao Q (2020) Predicting the trend of stock market index using the hybrid neural network based on multiple time scale feature learning. Appl Sci 10(11):3961

Ho TT, Huang Y (2021) Stock price movement prediction using sentiment analysis and candleStick chart representation. Sens-Basel 21(23):7957

Hochreiter S, Technische U, Munchen F, Informatik MG, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

Hou X, Wang K, Zhong C, Zhi W (2021) ST-trader: a spatial-temporal deep neural network for modeling stock market movement. IEEE/CAA J Autom Sin 8(5):1015–1024
Joshua ZGH, Huang X, Mou H, Li D, Wu Q, Xu YB (2019) BERT-based financial sentiment index and lstm-based stock return predictability. Quant Financ
Kipf TN, Welling M (2016) Semi-supervised classification with graph convolutional networks. arXiv: 1609.02907
Krizhevsky A, Sutskever I, Hinton GE (2017) ImageNet classification with deep convolutional neural networks. Commun ACM 60(6):84–90
Kyunghyun C, Bartvan M, Caglar G, Fethi B, Holger S, Yoshua B (2014) Learning phrase representations using rnn encoder-decoder for statistical machine translation. https://arxiv.org/pdf/1406.1078v1.pdf
Lettau M, Ludvigson SC (2005) Expected returns and expected dividend growth. J Financ Econ 76(3):583–626
Li Q, Wang TJ, Li P, Liu L, Gong QX, Chen YZ (2014) The effect of news and public mood on stock movements. Inf Sci 278:826–840
Li MW, Xu DY, Geng J, Hong WC (2022) A hybrid approach for forecasting ship motion using CNN–GRU–AM and GCWOA. Appl Soft Comput. https://doi.org/10.1016/j.asoc.2021.108084
Long JW, Chen ZP, He WB, Wu TY, Ren JT (2020) An integrated framework of deep learning and knowledge graph for prediction of stock price trend: an application in Chinese stock exchange market. Appl Soft Comput 91:106205
Mevlut T, Imran K (2006) Comparison of four different time series methods to forecast hepatitis A virus infection. Expert Syst Appl 31(1):41–46
Milad SF, Seyed HRH (2021) Forecasting stock price using integrated artificial neural network and metaheuristic algorithms compared to time series models. Soft Comput 25(13):8483–8513
Naik N, Mohan BR (2020) Intraday stock prediction based on deep neural network. Natl Acad Sci Lett 43(3):241–246
Naranjo A, Nimalendran M, Ryngaert M (1998) Stock returns, dividend yields, and Taxes. J Finance 53(6):2029–2057
Nelson DMQ, Pereira ACM, Oliveira RA (2017) Stock market's price movement prediction with LSTM neural networks. Int Joint Conf Neural Netw (IJCNN) 2017:1419–1426
Nicholas S (2001) Nonlinearities, cyclical behaviour and predictability in stock markets: international evidence. Int J Forecast 17(3):459–482
Omer BS, Ahmet MO (2018) Algorithmic financial trading with deep convolutional neural networks: time series to image conversion approach. Appl Soft Comput 70:525–538
Omer BS, Ahmet MO (2020) Financial trading model with stock bar chart image time series with deep convolutional neural networks. Intell Autom Soft Comput 26(2):323–334
Patel J, Shah S, Thakkar P (2015) Predicting stock market index using fusion of machine learning techniques. Expert Syst Appl 42(4):2162–2172
Philip HF, Hendrik G (1999) Additive outliers, GARCH and forecasting volatility. Int J Forecast 15(1):1–9
Prigogine I, Lefever R (1968) Symmetry breaking instabilities in dissipative systems. II. J Chem Phys 48(4):1695
Qing C, Karyl B, Leggio MJS (2005) A comparison between Fama and French's model and artificial neural networks in predicting the Chinese stock market. Comput Oper Res 32(10):2499–2512
Rather AM, Agarwal ASVN (2015) Recurrent neural network and a hybrid model for prediction of stock returns. Expert Syst Appl 42(6):3234–3241
Santos T, Veronesi P (2006) Labor income and predictable stock returns. CFA Digest 36(3):38–39
Silvio B, Salvatore MC, Andrea C (2020) Deep learning and time series-to-image encoding for financial forecasting. IEEE/CAA J Autom Sin 7(3):683–692
Thi-Thu N, Seokhoon Y (2019) A novel approach to short-term stock price movement prediction using transfer learning. Appl Sci 9(22):4745
Ticknor JL (2013) A Bayesian regularized artificial neural network for stock market forecasting. Expert Syst Appl 40(14):5501–5506
Vasanthi M, Seetharaman K (2021) A hybrid method for biometric authentication-oriented face detection using autoregressive model with Bayes Backpropagation Neural Network. Soft Comput 25(2):1659–1680
Venkatesh R, Balasubramanian C, Kaliappan M (2021) Rainfall prediction using generative adversarial networks with convolution neural network. Soft Comput 25(6):4725–4738
Wen M, Li P, Zhang LF, Chen Y (2019) Stock market trend prediction using high-order information of time series. IEEE Access 7:28299–28308
Wojciech Z, Ilya S, Oriol V (2014) Recurrent neural network regularization. https://arxiv.org/pdf/1409.2329v5.pdf
Wu D, Wang X, Su J, Tang B, Wu S (2020) A labeling method for financial time series prediction based on trends. Entropy 22(10):1162
Wu J, Xu K, Chen X, Li S, Zhao J (2022) Price graphs: Utilizing the structural information of financial time series for stock prediction. Inf Sci Int J 588(C):405–424
Zbikowski K (2015) Using volume weighted support vector machines with walk forward testing and feature selection for the purpose of creating stock trading strategy. Expert Syst Appl 42(4):1797–1805
Zhang D, Cai E (2021) Improving stock price forecasting using a large volume of news headline text. CMC-Comput Mater Contin 69(3):3931–3943
Zhang ZC, Hong WC (2021) Application of variational mode decomposition and chaotic grey wolf optimizer with support vector regression for forecasting electric loads. Knowl-Based Syst. https://doi.org/10.1016/j.knosys.2021.107297
Zhang Q, Chang JL, Meng GF, Xu SB, Xiang SM, Pan CH (2019) Learning graph structure via graph convolutional networks. Pattern Recogn 95:308–318
Zhao YL, Yang G (2022) Deep Learning-based Integrated Framework for stock price movement prediction. Appl Soft Comput 133:109921
Zhao JH, Zeng DL, Liang S, Kang HL, Liu QM (2020) Prediction model for stock price trend based on recurrent neural network. J Ambient Intell Humaniz Comput 12(1):745–753
Zhipeng T, Jing C, Qi K, Mengchu Z, Abdullah A, Khaled S (2022) Dynamic embedding projection-gated convolutional neural networks for text classification. IEEE Trans Neural Netw Learnng Syst 33(3):973–982
Zhou SQ, Ding LX, Zhang J et al (1997) Linearization learning method of BP neural networks. Wuhan Univ J Nat Sci 2(1):35–39
Zhou LG, Si YW, Fujita H (2017) Predicting the listing statuses of Chinese-listed companies using decision trees combined with an improved filter feature selection method. Knowl-Based Syst 128(6):93–101
Zhou F, Zhou HM, Yang ZH, Yang LH (2019) EMD2FNN: a strategy combining empirical mode decomposition and factorization

machine based neural network for stock market trend prediction. Expert Syst Appl 115:136–151