Applied Soft Computing Journal 163 (2024) 111847

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Algorithmic stock trading based on ensemble deep neural networks trained with time graph

Muhammed Yilmaz, Mustafa Mert Keskin, Ahmet Murat Ozbayoglu *

TOBB University of Economics and Technology, Ankara, 06560, Turkey

ARTICLE INFO

Keywords:
Financial forecasting
Stock market
Graphs
Deep learning
Deep neural networks
Convolutional neural networks
Ensemble models

ABSTRACT

Financial forecasting is generally implemented by analyzing the time series data related to the stock. This is accomplished widely with deep neural networks (DNNs) since DNNs can directly extract the related information that is otherwise hard to obtain. Time series is the core data representation of financial forecasting problem since it comes naturally. However recent studies show that even if time series representation is necessary, it still lacks certain aspects related to the problem. One of them is the relationship between the stocks of the market which can be captured through graph representation. Therefore, DNNs might solve the financial forecasting problem better when graph and time series representations are combined. In this study, we present different graph representations that can be used for this purpose. We also present an ensemble network that gives an investment strategy related to the stock market from stock predictions. Our proposed model returns an average of 20.09% annual profit on DOW30 dataset through daily buy–sell decisions based on close prices. Therefore, it can serve as a daily financial investment strategy, offering higher annual returns than conventional heuristic approaches.

1. Introduction

DNNs are used widely for financial forecasting since they can extract the information related to the problem from raw data [1]. However, data representation still plays a crucial role in deep learning since information that can be extracted is strictly correlated to the way the information is represented. Different representations can capture different aspects of the problem even if they belong to the same data. Graphs are best suited to represent the relational information of the data which can be significant in a dataset composed of dependent entities. Time series representation can capture the behavior of the entity with respect to time but it cannot capture its relation to other entities. Therefore, combining two representations can give information about entities with respect to both time and each other.

Graph representations are applied in several financial forecasting studies since they can capture the relationship between stocks. Analysis of graph representation of the market can lead to several inferences such as correlation coefficient estimation between stocks, critical stock analysis or general market behavior prediction. Combining these features with time series data leads to more successful and robust financial forecasting. In a previous study, it was demonstrated that DNNs trained with both graphs and time series return a higher average annual profit than some baseline models and the average market growth rate [2]. However, this study's further analysis indicates that DNNs do not yield a higher average annual profit than some heuristic strategies, indicating the need for further improvement to make them suitable for actual trading. This study provides additional graph representations to achieve this goal. DNNs trained with each representation perform similarly, showing that graph representation methods can vary. This study also introduces an ensemble model that outperforms the market average in annual return and introduces some heuristic strategies.

The key contributions of this paper are:

- To the best of our knowledge, for the first time in the literature, various graph representations, in total, six types of graphs (undirected Pearson, Spearman, Euclidean, and directed Pearson, Spearman, Euclidean) are generated and trained with CNNs and DNNs in a study, with their corresponding performance comparison.
- Combining various graph representations (defined in the previous six graphs) and training a single CNN model, named Graph Combined, is one of the other contributions, and we have also shown that it enhances the performance for all six graph models.
- We employ an ensemble model with graphs in order to outperform graph-based models and provide more profit. To best of our knowledge, ensemble models were not used by graph based

* Corresponding author.
E-mail addresses: muhyilmaz@etu.edu.tr (M. Yilmaz), m.keskin@etu.edu.tr (M.M. Keskin), mozbayoglu@etu.edu.tr (A.M. Ozbayoglu).

https://doi.org/10.1016/j.asoc.2024.111847
Received 10 January 2024; Received in revised form 30 April 2024; Accepted 28 May 2024
Available online 21 June 2024

models in finance. Therefore, using this ensemble model with graphs is a novel approach.
- We exhibit the effectiveness of our graph-based and ensemble graph-based models in analyzing the DOW30 stocks. Our approach involves the use of objective and equitable measurements, which include financial evaluation metrics, baseline techniques, and both MLP and MLP-Multi models.
- Our ensemble model offers a strategy that reduces risk while maximizing returns over the long haul when implementing the algorithmic trading system.

The rest of the paper is organized as follows. Section 2 explains other studies that use graph representations for financial forecasting. Section 3 provides several graph representations of the stock market and introduces ensemble neural network for financial forecasting. Section 4 compares the performance of DNNs trained with different graph representations. It also evaluates the performance of ensemble model. Finally, Section 5 summarizes and concludes the study.

2. Related works

In the literature, there are lots of studies that analyze the graph representation of the market in order to make inference about stocks or the stock market [3–10]. There are significant number of studies that combine the graph representation and DNN since DNNs are generally very successful to extract features from the data. For example, some studies train support vector machine (SVM), multi-layer perceptron (MLP), convolutional neural network (CNN) or long-short term memory (LSTM) with graph representation of the stock market [11–15]. In the literature, there are some models that specialize for graphs such as graph neural network (GNN) and graph convolutional network (GCN). They are also used for analysis of graph representation of the stock market [4,14,16–24]. GNN and GCN are feed with graphs directly. Therefore, they are more successful compared to earlier mentioned DNNs for financial analysis. However, they also require more data for training.

Table 1 provides an overview of some recently investigated graph-based models found in the literature. It details characteristics such as dataset, period, time resolution, method, and performance criteria. These studies have enhanced traditional graph models like GCN and GNN by altering their design or graph representation. They have been tested across various datasets using a range of evaluation metrics, including accuracy, precision, and recall for classification, as well as MAE, MSE, and RMSE for regression. Other metrics like Sharpe ratio, maximum drawdown, cumulative return, and annual return are also examined. These models have demonstrated superior performance compared to traditional machine learning, deep learning, and classical graph models.

On a separate note, ensemble models enhance the performance of individual weak learners trained separately by using techniques such as bagging, boosting, or stacking. These models have been extensively researched in the context of financial issues, particularly in addressing challenges in portfolio optimization. They analyze multiple stock prices and develop distinct models for each, making the use of ensemble models in finance highly beneficial. Table 2 highlights recent successful applications of ensemble models in finance. The research includes traditional machine learning models like Support Vector Regression (SVR), ensemble approaches such as SVR, LightGBM, and XGBoost, as well as advanced deep learning models like the Transformer and Convolutional Neural Networks (CNN), along with other ensemble and Reinforcement Learning (RL) models.

In the previous study, DOW30 was represented as 30 × 30 undirected and directed graph [2]. Baseline DNN is trained with time series data and some extracted features to predict the daily stock price increase. Graph based models were trained with combining the same data and features with the output of CNN which is feed with graph representation of the market. At the end, the previous study calculated the average annual return for daily buy/sell decisions based on the stock price prediction of DNNs. This way, it was shown that combining undirected or directed graph representation with time series increases the average annual return of daily buy/sell decisions.

In a recent study the S&P 500 dataset was represented as an undirected graph, treating each stock as a vertex. Spearman correlation coefficient was used as the weight of the edges between two stocks. The authors state that each vertex's degree represented how central the corresponding stock was. Therefore, they were able to analyze the most reliable stocks and sectors [10]. In our study, we applied Spearman correlation metric to generate graphs [10] and trained them like in the previous study [2]. The important takeaway of this study [10] is that it gave us an idea of how to generate new graph types, train them, and outperform the performance of the previous [2] model.

3. Method

In this study, we modified and enhanced the model proposed in [2]. Our main contribution and enhancement to that former study can be summarized in two folds: Firstly, we provide several graph representations for the same problem showing that graph data provide additional robust insights independent of the way the graph is created. Secondly, we show that an ensemble network trained with the outputs of earlier DNNs achieves better average annual return than heuristic strategies. We use the same dataset, but extended period of data to 2024, and experimental setup with the earlier study [2].

A. Dataset

In this study, we used DOW30 stocks, as in the previous study [2]. We gathered the daily stock prices of DOW30 companies from 2012 to 2024 using finance.yahoo.com. We trained several DNNs with graphs and their ensemble versions to identify the optimal daily buy–sell strategy that maximizes the annual return. We selected the daily closing prices of DOW30 as labels in this dataset. However, we calculated the percentage increase in stock prices between two consecutive days to make a stock price time series stationary. During the training of the models, the label values were normalized inside the range of [−1, 1].

B. Baseline Models

In the previous study [2], four models were implemented: MLP, MLP-multi, Graph, Graph-directed. For each stock, a model was trained to predict the daily price rate of change of that stock. Then buy/sell strategy was applied according to a heuristic strategy: The stock that was predicted to increase the most was bought. This process was repeated 30 times and average annual return was calculated. At the end, it was observed that MLP based models bring less average annual profit than the average market growth rate. On the other hand, custom developed graph based models resulted in more annual return than the average market growth rate [2]. This was expected since graph based models also learn from the graph representation and relation between the stocks in addition to the corresponding time series.

In Graph model, at a given time step, the stock market is represented as an undirected graph such that vertices of the graphs correspond to stocks and there is a weighted edge between each pair of vertices. The weight of an edge is calculated by Pearson correlation coefficient of the prices of two stocks for the last N days. N is the parameter that provides the window length for the correlation calculation. Therefore, the resolution of the history of stocks can be determined by changing N. A graph is represented with its adjacency matrix. It is preferred since CNN is specialized to extract features from 2-Dimensional data.

Graph-directed model also uses the adjacency matrix. However, the adjacency matrix of Graph-directed model is not symmetric. Therefore, an edge from stock A to stock B is different than an edge

Table 1
Algorithmic stock trading with graph models.

Art.	Data set	Period	Time resolution	Method	Performance criteria
[25]	CSI 300 CSI 500	2015–2019	Daily	Multi-GCGRU	Acc PR RC F1 MDD
[26]	DJI	2002–2020	Daily	GCN, RL	MDD SR Return
[27]	Taiwan Stock S&P 500 NASDAQ	–	Daily	FinGAT	MRR Precision
[28]	NASDAQ	2011–2021	Daily	GCNET	Accuracy MRR
[29]	S&P 500 CSI 300	2016–2021	Daily	THGNN	Acc PR RC F1 MDD CR
[30]	Bitcoin Litecoin Ethereum Dash Coin	2015–2021	Daily	GNN, LSTM	MAE RMSE
[28]	S&P 500 NASDAQ	2012–2016 2011–2021	Daily	GCNET	Accuracy
[31]	DowJones2005 EuroBonds ItBondComodities WorldMixBonds	2013–2018	Daily	GNNs	TR SR
[32]	WTI	1983–2022	Daily	LSTM, GCN	MSE RMSE R2 MAPE
[33]	Chinese market	–	–	GCN	Acc, F1, Pr
[34]	American & Taiwan market	–	1 Day 3 Days 7 Days	CNN-LSTM	Acc
[35]	China A-shares market	2018–2019	Daily	MAGNN	A return D return Sharpe Ratio
[36]	S&P 500	2005–2021	Daily	SCRG NIST-GNN NIST-GNN-SCRG	PnL Sharpe Ratio t-Statistics

from stock B to stock A. The weight of an edge from stock A to stock B represents the strength of the correlation between the past price of stock A and the current price of stock B. In other words, it shows how the price of stock B is affected by the price of stock A. This is accomplished with adding the following additional constraints to Pearson correlation formula which was given in the previous study:

$$G_t(i,j) = \max(C((t-k)_i, t_j)) \; 0 \le k \le M-1 \tag{1}$$

$$G_t(i,j) = G_t(j,i) \; \exists \; (i,j) \in D \tag{2}$$

Sliding window method is used to calculate the weight $G_t(i,j)$. For this, initially the correlation between stock i and j is calculated from the data with window size N. The position of the window is fixed for stock j but each time it is shifted one day backwards for stock i and new correlation is calculated. This process is repeated for M times and maximum value is chosen as the weight of the edge from stock A to stock B.

We selected several basic and greedy strategies to compare the performance of our novel models. First, the DOW30 index is the average return of 30 Dow stocks for each year. In Strategy1, we invested only in the stock that brought the highest annual return in the previous year. In Strategy2, we bought 3 stocks, which resulted in the highest annual returns in the previous year. While strategies 1 and 2 represent good baseline strategies, there were issues with the growth rates of the APPLE stock price in 2020 and CAT (Caterpillar Inc.) in 2017. These rates were significantly higher than the DOW Jones average or their previous growth rate. Therefore, these values were assumed to be anomalies. As a result, we implemented two additional strategies, Strategy3 and Strategy4, for this case. These strategies excluded the aforementioned stocks when calculating baseline greedy strategies solely for the years that exhibited anomalies. Therefore, in 2017, CAT stock market was removed for calculating strategies 3 and 4, and in 2020, APPLE was removed as well. Hence, for Strategies 3 and 4, the calculations for these two years were based on 29 stock prices. While Strategy3 mirrored Strategy1, and Strategy4 was identical to Strategy2, the sole difference lied in the exclusion of stock prices for the years 2017 and 2020. We have modified these strategies to ensure a more comparable evaluation. As a result, we decided to use the following models and strategies as baselines:

- MLP [2]

Table 2
Algorithmic stock trading with ensemble models.

Art.	Data set	Period	Time resolution	Method	Performance criteria
[37]	A-share market	2013–2019	Daily	HGTAN	Acc Pre Recall F1
[38]	DAX MASI HKEX CAC 40 NASDAQ FTSE 250	2018–2023	Daily	XGBoost LSTM LSTM-XGBoost	MAE MAPE RMSE R2
[39]	S&P 500	2012–2019	Daily	RL	MDD Return etc.
[40]	BTC ETH XRP	2018–2019	Hourly	LSTM BiLSTM CNN	Acc, AUC, F1
[41]	DAX DOW S&P500	2014–2019 2000–2019 2017–2021	Daily	CNN-LSTM GRU-CNN Average Bagging Stacking	MSE, MAE
[42]	Cryptocurrency	–	30 m	CNN, DRL	APV SR MDD
[43]	S&P500	–	Daily	Transformer Graph CNN Ranking Ensemble	Return rate SR MDD
[44]	JSE	–	–	ANN Bagging Ensemble Monte Carlo	MSE RMSE R2
[45]	SP100	2011–2021	Daily	Fuzzy Ensemble RL Ensemble	Return rate SR
[46]	EU ETS SCI BTC	2008–2020 2010–2020 2012–2020	Daily	ARMA CNN-LSTM	RMSE MAPE MAE Dstat
[47]	DOW30 Japanese Stocks British Stocks	2000–2021	Daily	RL	AR, CR SR, MD
[48]	Various dataset	–	Daily	BiLSTM DL, RL	RMSE, MAE, DA
[49]	CSI 300	2020–2022	Daily	SVR, Lightgbm XGBoost, Random Forest LSTM	Various metrics

- MLP-Multi [2]
- Graph Undirected [2]
- Graph Directed [2]
- Strategies 1, 2, 3 and 4

C. Similarity Metrics and Graph Models

In the previous study, Pearson correlation coefficient was used as the weights of the edges in graph representation of the stock market. However, other methods also can be used as weights as long as they represent a similarity or dissimilarity metric between two stocks. Therefore, in the first part of this study we implemented other graph representations in order to test whether a DNN could also bring similar annual profit independent of the graph representation.

Firstly, graph representation with Spearman correlation coefficient as its weights was implemented. Spearman correlation is a similarity metric like Pearson correlation. Its formula is:

$$r_s = \rho_{R(x),R(y)} = \frac{cov(R(x), R(y))}{\sigma_{R(x)}\sigma_{R(y)}} \tag{3}$$

$$C(t_i, t_j) = r_s \tag{4}$$

where ρ denotes the Pearson correlation coefficient applied to rank functions of two variables. Spearman correlation calculates monotonic relationship between two variables whereas Pearson correlation calculates the linear relationship. In linear relationship, the change in one variable is proportional to the change in the other variable. In monotonic relationship, the variables move together but not at a constant rate. Therefore, unlike Pearson correlation, Spearman correlation coefficient can calculate the correct weight when the correlation relation is nonlinear.

We also used the inverse of Euclidean distance as the weight of the edge between two stocks in order to show that a DNN can also learn the relationship between stocks from a graph representation based on dissimilarity metric. Euclidean distance between two points is:

$$d(X,Y) = \sqrt{\sum_{n=1}^{N}(X_n - Y_n)^2} \tag{5}$$

Then, the weight between stocks i and j becomes:

$$C(t_i, t_j) = \frac{40}{d(X_{t_i}, X_{t_j})} \tag{6}$$

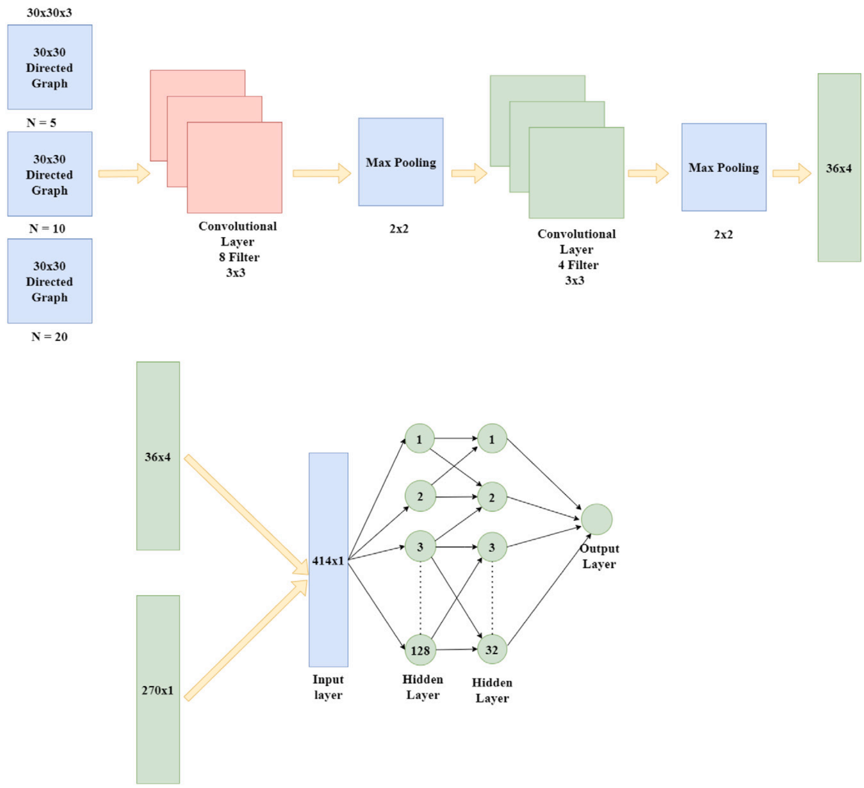

Fig. 1. CNN model trained by Graph-combined data.

where 40 is the normalization constant.
Spearman and Euclidean models were trained with undirected graph which was created with Spearman correlation coefficient and Euclidean distance respectively for a window size of N similar to the earlier study. Also, directed graphs were created with the same sliding window process of the previous study which were used to train Spearman-directed and Euclidean-directed models. Graph representations were used to train CNN the same way with the previous study except the hyper-parameter tuning stage.
The weights of the edges were constructed differently in each graph model. Therefore, it can be claimed that they can capture the different aspects of the relationship between stocks. If this hypothesis is true, a DNN trained by the combination of all graph representations (Graph-combined) can have better inference performance and bring more annual profit. In order to test it, graph representations with window size 5, 10, 20 were combined as single data. There were 6 graph representations in total: Graph, Graph-directed, Spearman, Spearman-directed, Euclidean, Euclidean-directed. Therefore, the input of Graph-combined model becomes a tensor with size $18 \times 30 \times 30$. Fig. 1 illustrates the CNN trained with combined graph data in details. The only difference in its structure from the previous study's was that the input is $18 \times 30 \times 30$ tensor instead of $3 \times 30 \times 30$ tensor.

D. Ensemble Models

Ensemble networks are widely used since they can make inference that other models cannot with combined knowledge of several models. There are 30 stocks in DOW30. Successfully determining the behavior of all stocks in each day is a difficult task. Moreover, instead of determining which stock prices will increase next day, determining the stock whose price will increase the highest might be more desirable even if it is more challenging. This can be accomplished with ensemble models. In our study, a total of 30 models were trained, each predicting the behavior of a stock price in the next day independent of the others. Rather than employing a different strategy for each stock, a single unified strategy can be developed. This approach allows for investment in the market with potentially greater success. The main motivation of using ensemble networks is to formulate such strategy.
In the previous study [2] a heuristic investment strategy was applied for the buy/sell decision: Each day, the money was invested to the stock which was predicted to increase the most. In this study, we trained a DNN to predict the best stock for investment. The DNN was further trained with the outputs of the 30 models which were each previously trained to predict the percent profit of a specific stock. In addition to the outputs of 30 models, other features were also fed to the ensemble model in order to provide a better understanding of the problem. These features are:

- R2 score between last predictions and target (real) values
- Correlation coefficient between last predictions and target values
- Ratio of correct sell/buy transactions between last predictions and target values.
- Target values of the last day

These features were calculated with a window size of 10 which is optimized through exhaustive search. Each feature was represented as a 30×1 vector, reflecting inputs from 30 distinct models. After combining all features with the predictions of the models, a 150×1 input vector was created to train the ensemble neural network. The ensemble neural network, which is shown in Fig. 2, consists of the following layers:

- Input layer consists of 150 neurons
- Hidden layer consists of 128 neurons with ReLU activation function
- Hidden layer consists of 256 neurons with ReLU activation function
- Output layer consists of 30 neurons with softmax activation function.

– Hidden layer consists of 256 neurons with ReLU activation function
– Output layer consists of 30 neurons with softmax activation function.

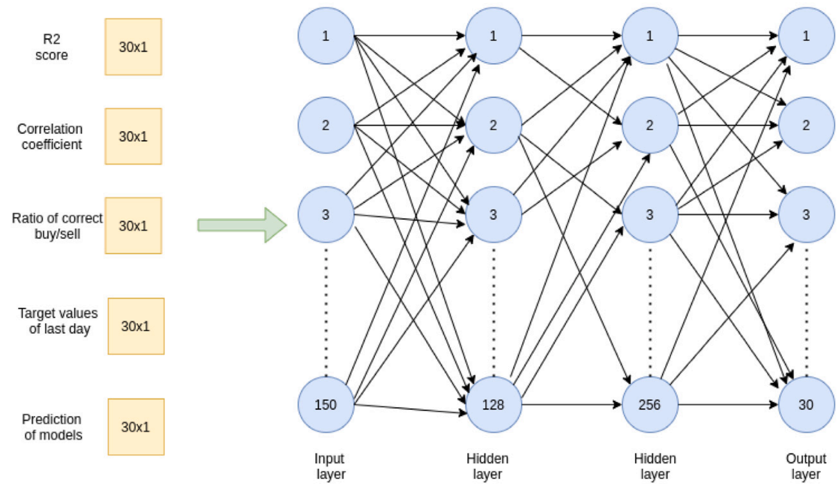

Fig. 2. Ensemble neural network trained to predict the stock with the most percentage increase.

The value of each output layer neuron provides the corresponding stock's likelihood of being the stock with the largest percentage increase in the next day. Therefore, the label of the ensemble model is a 30 × 1 one-hot vector. The main purpose of the ensemble network is to learn the combined behavior of stock prices and choose the best stock to invest for each day. Categorical cross-entropy was used to train the ensemble network. Once the ensemble network was trained, the stock with the highest probability was selected for trading in making buy/sell decisions.
We trained the ensemble DNN model to generate a good and reliable buy/sell daily trading strategy since neural networks can create robust mathematical models and have been used in real world problems. It is also possible to formulate strategies to choose the best stock fund for investment without using an ensemble model. One way of accomplishing that is using greedy algorithms. We developed several greedy strategies for choosing the best stock fund for buy/sell decisions in order to use them as baselines and evaluated the performance of the developed ensemble model. The most successful heuristic algorithm involved purchasing the stock forecasted to yield the highest profit. Therefore, we compared the annual return of that strategy with the annual return of the ensemble model. Ensemble models were trained only for graph based models since they performed significantly better than MLP based models. For each graph based application, an ensemble model was trained and tested. Each ensemble model received the outputs from 30 models as inputs and generated a prediction for the following day in the form of a 30 × 1 vector. This vector provided the probability of a favorable investment choice for each stock. Then, similar to the heuristic strategy, we purchased the stock that was projected to generate the highest profit.

4. Performance evaluation and discussions

In order to compare the results of this study with the results of the previous study [2], the same evaluation setup and metrics of the previous study were used. Each model was trained for two years to predict the daily percentage price change of a stock. In the following year, predictions of the models were used for buy/sell decisions and the annual return of the heuristic strategy or ensemble model was calculated. This process was repeated 30 times for the test year and their averages are presented for each year.

In the previous study [2], Graph and Graph-directed models were trained to predict the daily price change. Then the heuristic strategy of buying the stock with the most anticipated percentage increase was applied for buy/sell decisions. In that study, all models were trained until 2019 [2]. We retrained Graph and Graph-directed models from 2012 to 2023 in this study. As seen in Table 4, their average annual returns were 14.15% and 14.43%, respectively, which were both greater than the average of DOW30, MLP and MLP-Multi. In this study, even though the results were promising, we anticipated that the performance could be further enhanced through the application of additional heuristics or strategic modifications. To investigate that, we calculated the annual returns of several greedy strategies including Strategy1, Strategy2, Strategy3 and Strategy4. Strategy1 (and also Strategy3) was not only more greedy than Strategy2 (and Strategy4) but also it was more sensitive to noise due to its single choice approach, yet, in practice, it was able to bring more annual profit than Strategy2 as it can be seen from Table 3. Table 3 illustrates the annual returns of Graph, Graph-directed, Strategy1, Strategy2, Strategy3, Strategy4 and growth rate of DOW30. The average annual returns of Graph and Graph-directed models were not greater than the average annual return of Strategy1, 2, 3, 4 except DOW30 Average. Strategy1, with 24.76% annual profit, gave the best performance among heuristic strategies. Despite Strategy1's success, its worst-case scenario, which was −16.16%, indicates it is also a risky strategy. Strategy2 was much more reliable and less volatile. Strategy3 and Strategy4 were modified versions of Strategy1 and Strategy2 due to anomalies in the volatility of stock prices in 2020 and 2017. Upon eliminating the anomalies, the average annual returns of Strategy1 and Strategy2 dropped from 24.76% and 18.48%, to 18.73% and 15.0%, respectively. This adjustment allowed us to evaluate our models without the influence of abnormal increases. However, Graph and Graph-directed models still could not perform better than Strategy1, Strategy2, Strategy3 and Strategy4.

Realizing that Graph and Graph-directed models could not outperform the simple greedy algorithms used in Strategy1, Strategy2, Strategy3, and Strategy4 from a return performance perspective, we focused on enhancing their strategies through the use of ensemble models to see if we could surpass their average performance. In Ensemble-undirected and Ensemble-directed models, instead of the heuristic strategy, corresponding ensemble networks were used to improve the performance of Graph and Graph-directed models respectively. After the training process explained in the previous section, Ensemble-undirected and Ensemble-directed models resulted in annual returns of 18.33% and

Table 3
Comparison of the average annual return of several heuristic strategies and Pearson based graph models from 2012 to 2023.

Year	Graph	Graph directed	Strategy1	Strategy2	Strategy3	Strategy4	DOW30 average
2023	−1.37	−4.99	1.84	−5.90	1.84	−5.90	−1.47
2022	−5.09	−1.63	−6.61	17.7	−6.61	17.7	−2.79
2021	23.58	23.64	38.06	37.9	38.06	37.9	25.49
2020	11.58	9.68	78.24	37.17	39.94	16.14	5.48
2019	22.31	26.26	23.60	25.27	23.60	25.27	25.55
2018	0.97	2.58	10.77	−3.26	10.77	−3.26	0.29
2017	25.26	22.43	72.7	45.81	38.61	25.06	25.89
2016	15.04	15.67	−16.16	−1.65	−16.16	−1.6	16.52
2015	2.25	4.81	−2.40	3.14	−2.40	3.14	2.46
2014	19.49	19.83	−2.67	9.72	−2.67	9.72	15.79
2013	34.23	35.89	47.87	38.9	47.87	38.9	30.38
2012	21.50	18.98	51.87	16.95	51.87	16.95	14.56
Average	14.15	14.43	24.76	18.48	18.73	15.0	13.11

Table 4
Comparison of the average annual returns of baseline models and Pearson correlation based graph models with heuristic strategy from 2012 to 2023.

Year	MLP	MLP-Multi	Graph	Graph directed	DOW30 average
2023	−13.65	−1.48	−1.37	−4.99	−1.47
2022	−18.45	−8.05	−5.09	−1.63	−2.79
2021	25.42	29.74	23.58	23.64	25.49
2020	−1.33	1.41	11.58	9.68	5.48
2019	7.88	17.93	22.31	26.26	25.55
2018	−9.05	−3.92	0.97	2.58	0.29
2017	23.21	22.82	25.26	22.43	25.89
2016	16.49	16.11	15.04	15.67	16.52
2015	0.80	6.49	2.25	4.81	2.46
2014	20.93	14.53	19.49	19.83	15.79
2013	33.83	35.0	34.23	35.89	30.38
2012	19.99	17.70	21.50	18.98	14.56
Average	8.83	12.36	14.15	14.43	13.18

Table 5
Comparison of the average annual returns of Strategy1, Strategy3, Pearson correlation based models with heuristic strategy and ensemble networks from 2012 to 2023.

Year	Graph	Graph directed	Ensemble undirected	Ensemble directed	Strategy1	Strategy 3
2023	−1.37	−4.99	−8.574	−8.60	1.84	1.84
2022	−5.09	−1.63	−6.92	−9.51	−6.61	−6.61
2021	23.58	23.64	25.54	19.31	38.06	38.06
2020	11.58	9.68	8.540	15.81	78.24	39.94
2019	22.31	26.26	32.26	31.34	23.60	23.60
2018	0.97	2.58	2.03	−2.33	10.77	10.77
2017	25.26	22.43	25.25	29.58	72.7	38.61
2016	15.04	15.67	28.76	26.96	−16.16	−16.16
2015	2.25	4.81	9.79	13.82	−2.4	−2.4
2014	19.49	19.83	26.20	24.91	−2.67	−2.67
2013	34.23	35.89	46.64	37.33	47.87	47.87
2012	21.50	18.98	30.42	33.73	51.87	51.87
Average	14.15	14.43	18.33	17.69	24.76	18.73

17.69% respectively in the test period. Hence, annual return of the heuristic strategy was overwhelmingly boosted by the ensemble networks as presented in Table 5. Both ensemble models also outperformed Strategy4 and they had comparable performances with Strategy3 and even Strategy2, which included anomalies.

In addition to Pearson correlation based models (Graph and Graph-directed), Spearman, Spearman-directed, Euclidean, Euclidean-directed and Graph-combined models are also implemented. Their average annual returns with heuristic strategies and ensemble models are presented in Tables 6 and 7 respectively. As it can be seen from Table 6, the results of directed graphs are better than their corresponding undirected graphs. However, Pearson, Spearman and Euclidean graphs provided similar results regardless of their types. Hence, these results show that graph representation is powerful and that it improves the annual return regardless of the graph representation method. Similarly, the effectiveness of ensemble methods is evident, as shown in Table 7, where all graph types benefit from the use of ensemble models. These models not only yielded better results but also demonstrated to be robust across various graph types.

At any given time step, Graph-combined network was trained with $18 \times 30 \times 30$ input which was the combination of 6 graph representations. Using the heuristic strategy with Graph-combined network resulted in 16.69% average annual return. Meanwhile, the ensemble model trained with Graph-combined network outputs brought 20.09% average annual return which was the highest achieved average annual return. This indicates that integrating various graph representations enhanced the annual return. Graph-combined network achieved more profit than Strategy2, Strategy3, Strategy4 except for Strategy1. Hence, we showed that Graph-combined networks can be highly practical to implement algorithmic trading with less risk and more profit.

Tables 8, 9, 10 compares the performance of the graph based models with the heuristic strategy from 2012 to 2024 using the financial evaluation metrics, namely cumulative return (CR), win ratio (WR), sharpe ratio (SR) and maximum drawdown (MD). In Table 8, the CR and SR of Graph, Graph Directed models were strictly greater than MLP and MLP-Multi. In Table 9, although the WR and MD values of all models were within close proximity, the CR and SR values of the graph combined model were strictly greater than the other graph-based models. It can be noticed from Table 9 that Sharpe ratios of all models reflected the relative performance observed through the CR values, as expected. Sharpe ratio and cumulative returns of directed graphs were satisfactory compared to those of undirected graphs. Consequently, the combined graph model conducted transactions that generated higher profits and avoided riskier ones. Table 10 shows the performance of the ensemble model from 2012 to 2023 using the same financial evaluation metrics. The ensemble of directed graphs performed better than the ensemble of undirected graphs except the Pearson directed graph. The ensemble model, which integrated all the graphs, achieved the highest Cumulative Return (CR) and Sharpe Ratio (SR) compared to other ensemble and graph models, with these scores being significantly higher than the rest.

Fig. 3 displays the cumulative returns for MLP, MLP-Multi, Graph, and Graph-Directed models using the heuristic strategy from 2012 to 2023. Among these, the Graph and Graph-Directed models achieved the highest profit compared to MLP and MLP-Multi. As clearly illustrated in Fig. 4, the Graph-combined model secured the highest profit across all time periods. Fig. 5 presents the cumulative returns of the ensemble models from 2012 to 2023, where the ensemble approach of the Graph-combined model delivered the greatest cumulative and average profit among the models.

5. Conclusion and future work

There are several studies that use graph representation for financial forecasting. The main motivation of this study is to focus on the annual return of the investment strategy and try to develop a reliable and better performing graph based trading strategy. For this purpose, we investigated several graph representations of the stock market. For each stock, a DNN was trained with graph and time series data. From these models a final ensemble model was trained. The algorithmic trading strategy based on the decisions made by the graph ensemble model was applied according to the output of this model and 20.09% average annual profit was achieved.

Our results indicate that using various graph models outperforms the performance of financial forecasting by extracting additional features from graph information and feeding them into DNN networks. Moreover, adding ensemble models to graph models and training them boosts their performance by comparing the heuristic human based strategies. These models not only brought more profit, but they were

Table 6
Comparison of the average annual returns of graph based models with heuristic strategy from 2012 to 2023.

Year	Graph	Graph directed	Spearman	Spearman directed	Euclidean	Euclidean directed	Graph combined
2023	−1.37	−4.99	−3.21	−1.39	−0.18	−0.21	2.440
2022	−5.09	−1.63	1.89	−4.74	−2.91	−0.804	−0.36
2021	23.58	23.64	17.82	29.97	21.13	20.96	22.28
2020	11.58	9.68	17.81	11.17	8.64	10.35	15.49
2019	22.31	26.26	22.98	32.51	27.61	30.29	25.84
2018	0.97	2.58	−0.27	7.07	2.40	4.53	3.30
2017	25.26	22.43	28.47	24.18	26.53	26.08	26.60
2016	15.04	15.67	17.11	18.50	17.80	17.05	18.21
2015	2.25	4.81	0.87	4.52	2.02	3.58	10.52
2014	19.49	19.83	15.02	10.41	18.59	14.09	13.15
2013	34.23	35.89	35.29	33.71	28.43	34.05	42.48
2012	21.50	18.98	20.93	19.12	14.50	18.37	20.27
Avg	14.15	14.43	14.56	15.42	13.71	14.86	16.69

Table 7
Comparison of the average annual returns of graph based models with ensemble model from 2012 to 2023.

Year	Graph	Graph directed	Spearman	Spearman directed	Euclidean	Euclidean directed	Graph combined
2023	−8.57	−8.60	−12.42	−12.80	−8.19	−11.54	−5.64
2022	−6.92	−9.51	−8.70	−8.80	−7.93	−5.10	−7.96
2021	25.54	19.31	21.0	25.89	25.05	20.84	23.08
2020	8.54	15.81	14.13	23.40	15.25	20.29	18.16
2019	32.26	31.34	28.65	26.29	26.54	27.25	33.82
2018	2.03	−2.33	−3.99	−3.79	−4.36	−4.38	3.62
2017	25.25	29.58	28.49	25.92	28.48	26.03	26.08
2016	28.76	26.96	27.54	35.26	29.73	30.66	35.34
2015	9.79	13.82	14.65	14.39	8.42	11.24	13.54
2014	26.20	24.91	29.05	28.98	25.51	28.94	27.04
2013	46.64	37.33	44.10	47.37	38.67	43.94	49.50
2013	30.42	33.73	28.68	31.76	29.86	21.84	24.52
Avg	18.33	17.70	17.60	19.49	17.25	17.50	20.09

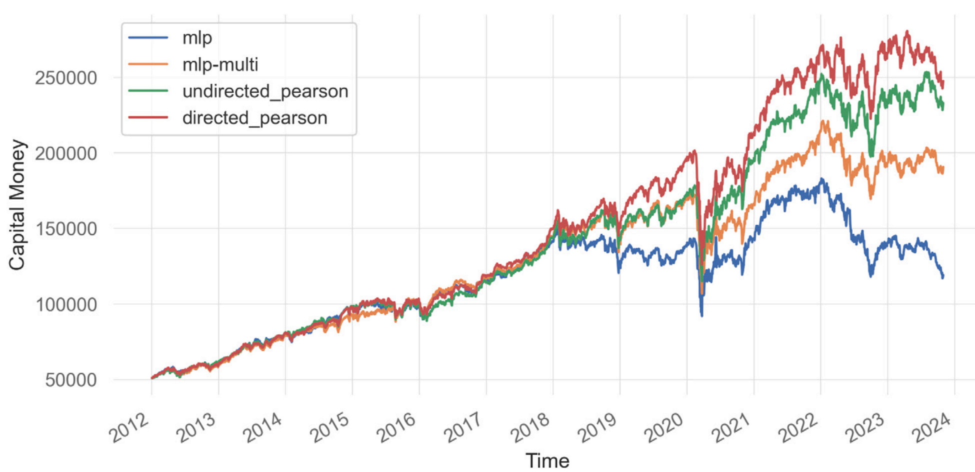

Fig. 3. Cumulative returns of graph based models from 2012 to 2023.

Table 8
Assessing the performance metrics MLP, MLP-Multi [2] and Pearson correlation graph based models from 2012 to 2023. The table includes four acronyms: CR (Cumulative Return), WR (Win Ratio), SR (Sharpe Ratio), and MD (Maximum Drawdown).

	MLP	MLP-Multi	Graph	Graph directed
CR(%)	138.40	281.42	366.06	395.34
WR	0.515	0.519	0.519	0.521
SR	0.353	0.518	0.586	0.594
MD	−0.581	−0.493	−0.482	−0.471

also more reliable and robust. Hence, using this ensemble structure with graph-based models is a novel approach to financial forecasting. In the future, these novel approaches will be used on various financial datasets and stock prices in order to determine the relationship between different stock prices and investment strategies. Therefore, its general structure is appropriate for various datasets. Another study that can be considered as future work is that this ensemble neural network structure can be used with different types of models that have been studied in the literature previously or will be studied next because our ensemble neural network model structure is highly suitable for any model type with sufficient effort, such as fine-tuning this structure or changing its network size for any other problems. In the future, combining various models with this graph-based model can potentially yield higher profits. Therefore, our graph models and ensemble model structure can be integrated into existing financial forecasting; they can outperform existing financial forecasting strategies or be studied

Table 9
Comparison of the evaluation scores of graph based models with ensemble networks from 2012 to 2023.

Year	Graph	Graph directed	Spearman	Spearman directed	Euclidean	Euclidean directed	Graph combined
CR(%)	366.06	395.34	389.83	451.03	371.07	374.77	508.83
WR	0.519	0.521	0.521	0.521	0.521	0.52	0.523
SR	0.587	0.595	0.595	0.638	0.586	0.599	0.669
MD	−0.483	−0.472	−0.458	−0.453	−0.479	−0.477	−0.497

Table 10
Comparison of the evaluation scores of graph based models with ensemble networks from 2012 to 2023.

Year	Graph	Graph directed	Spearman	Spearman directed	Euclidean	Euclidean directed	Graph combined
CR(%)	617.14	590.85	563.96	647.27	574.04	531.91	740.71
WR	0.521	0.520	0.518	0.522	0.520	0.519	0.522
SR	0.656	0.655	0.628	0.689	0.632	0.649	0.725
MD	−0.522	−0.511	−0.546	−0.525	−0.530	−0.507	−0.507

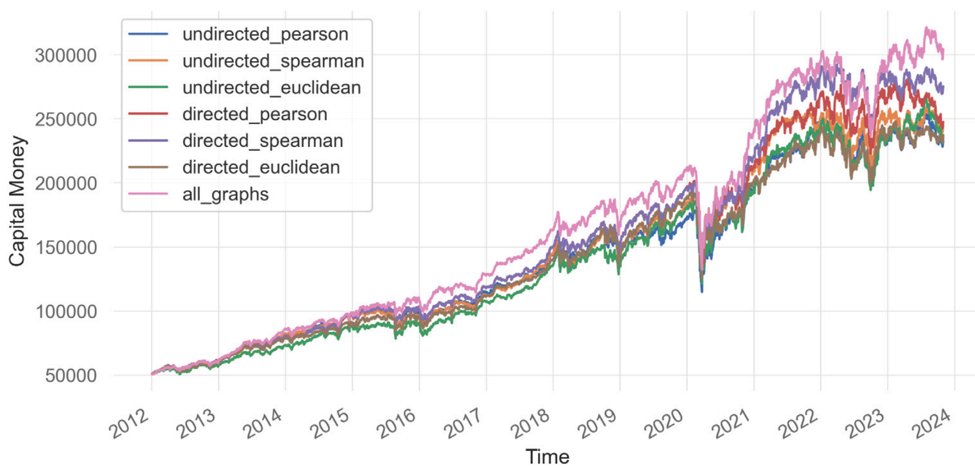

Fig. 4. Cumulative returns of graph based models from 2012 to 2023.

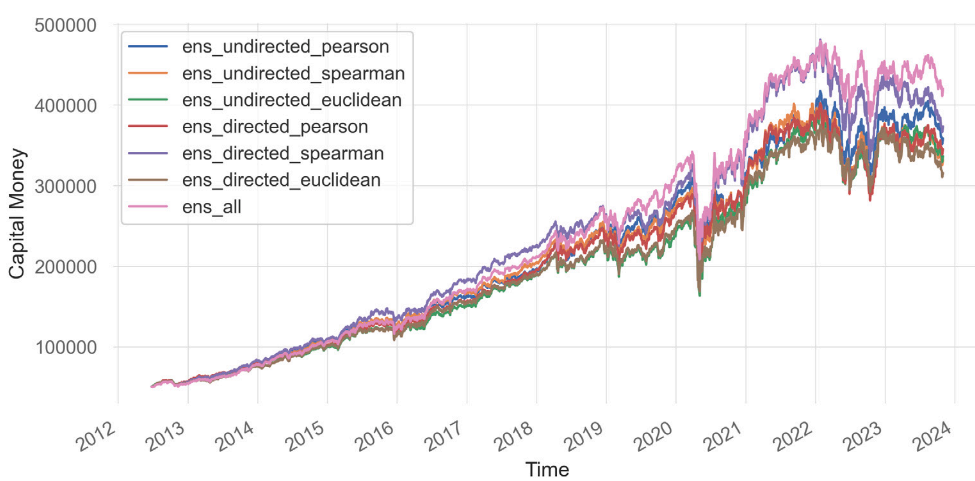

Fig. 5. Cumulative returns of ensemble models from 2012 to 2023.

in future works. In the future, we believe further enhancements are possible through alternative heuristics or strategy adjustments. Instead of using ensemble models with our graph-based models, algorithms such as genetic algorithms and reinforcement learning can be used to find investment strategies. These types of algorithms find different strategies where deep neural networks cannot, but they require more attention, effort, and training time compared to deep neural networks on financial problems due to their highly dynamic properties. These ideas will be examined and compared with our neural network ensemble models. This study indicates that using and training various graph models are innovative approaches that have the potential to achieve good performance. Furthermore, ensemble neural networks trained with time graphs present high performance, improvement, and robustness.

CRediT authorship contribution statement

Muhammed Yilmaz: Writing – original draft, Supervision, Methodology, Conceptualization. **Mustafa Mert Keskin:** Writing – original draft, Validation, Methodology, Data curation. **Ahmet Murat Ozbayoglu:** Writing – review & editing, Writing – original draft, Supervision, Methodology.

Declaration of competing interest

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.

Data availability

Data will be made available on request.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used chatGPT in order to rephrase some of the sentences and explanations for better clarity. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

References

[1] O.B. Sezer, M.U. Gudelek, A.M. Ozbayoglu, Financial time series forecasting with deep learning: A systematic literature review: 2005–2019, Appl. Soft Comput. 90 (2020) 106181.
[2] M.M. Keskin, M. Yilmaz, A.M. Ozbayoglu, A deep neural network model for stock investment recommendation by considering the stock market as a time graph, in: 2021 2nd International Informatics and Software Engineering Conference, IISEC, IEEE, 2021, pp. 1–6.
[3] R. Flanagan, L. Lacasa, Irreversibility of financial time series: a graph-theoretical approach, Phys. Lett. A 380 (20) (2016) 1689–1697.
[4] J. Long, Z. Chen, W. He, T. Wu, J. Ren, 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 (2020) 106205.
[5] Y. Long, Visibility graph network analysis of gold price time series, Phys. A 392 (16) (2013) 3374–3384.
[6] E.J. Ruiz, V. Hristidis, C. Castillo, A. Gionis, A. Jaimes, Correlating financial time series with micro-blogging activity, in: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, 2012, pp. 513–522.
[7] L. Wang, Z. Wang, S. Zhao, S. Tan, Stock market trend prediction using dynamical Bayesian factor graph, Expert Syst. Appl. 42 (15–16) (2015) 6267–6275.
[8] D. Xiao, J. Wang, Graph based and multifractal analysis of financial time series model by continuum percolation, Int. J. Nonlinear Sci. Numer. Simul. 15 (5) (2014) 265–277.
[9] Y. Xiu, G. Wang, W.K.V. Chan, Crash diagnosis and price rebound prediction in NYSE composite index based on visibility graph and time-evolving stock correlation network, Entropy 23 (12) (2021) 1612.
[10] J.R. Abrams, J. Celaya-Alcalá, D. Baldwin, R. Gonda, Z. Chen, Analysis of equity markets: A graph theory approach, Soc. Ind. Appl. Math. (2016) doi 10.
[11] S. Jeon, B. Hong, V. Chang, Pattern graph tracking-based stock price prediction using big data, Future Gener. Comput. Syst. 80 (2018) 171–187.
[12] C.K.-S. Leung, R.K. MacKinnon, Y. Wang, A machine learning approach for stock price prediction, in: Proceedings of the 18th International Database Engineering & Applications Symposium, 2014, pp. 274–277.
[13] H. Liu, B. Song, Stock price trend prediction model based on deep residual network and stock price graph, in: 2018 11th International Symposium on Computational Intelligence and Design, ISCID, Vol. 2, IEEE, 2018, pp. 328–331.
[14] J.M.-T. Wu, Z. Li, N. Herencsar, B. Vo, J.C.-W. Lin, A graph-based CNN-LSTM stock price prediction algorithm with leading indicators, Multimedia Syst. (2021) 1–20.
[15] Y. Wang, C. Zhang, S. Wang, S.Y. Philip, L. Bai, L. Cui, Deep co-investment network learning for financial assets, in: 2018 IEEE International Conference on Big Knowledge, ICBK, IEEE, 2018, pp. 41–48.
[16] W. Li, R. Bao, K. Harimoto, D. Chen, J. Xu, Q. Su, Modeling the stock relation with graph network for overnight stock movement prediction, in: No. CONF, 2020, pp. 4541–4547.
[17] R. Kim, C.H. So, M. Jeong, S. Lee, J. Kim, J. Kang, Hats: A hierarchical graph attention network for stock movement prediction, 2019, arXiv preprint arXiv:1908.07999.
[18] W. Chen, M. Jiang, W.-G. Zhang, Z. Chen, A novel graph convolutional feature based convolutional neural network for stock trend prediction, Inform. Sci. 556 (2021) 67–94.
[19] Y. Chen, Z. Wei, X. Huang, Incorporating corporation relationship via graph convolutional neural networks for stock price prediction, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018, pp. 1655–1658.
[20] D. Loe, S.-l. Chang, J. Chau, Stock market movement prediction using graph convolutional networks, UCSD Data Sci. Capstone Proj. 2021 (2020).
[21] D. Matsunaga, T. Suzumura, T. Takahashi, Exploring graph neural networks for stock market predictions with rolling window analysis, 2019, arXiv preprint arXiv:1909.10660.
[22] P. Patil, C.-S.M. Wu, K. Potika, M. Orang, Stock market prediction using ensemble of graph theory, machine learning and deep learning models, in: Proceedings of the 3rd International Conference on Software Engineering and Information Management, 2020, pp. 85–92.
[23] X. Ying, C. Xu, J. Gao, J. Wang, Z. Li, Time-aware graph relational attention network for stock recommendation, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2020, pp. 2281–2284.
[24] J. Ye, J. Zhao, K. Ye, C. Xu, Multi-view graph convolutional networks for relationship-driven stock prediction, 2020, arXiv preprint arXiv:2005.04955.
[25] J. Ye, J. Zhao, K. Ye, C. Xu, Multi-graph convolutional network for relationship-driven stock movement prediction, in: 2020 25th International Conference on Pattern Recognition, ICPR, IEEE, 2021, pp. 6702–6709.
[26] F. Soleymani, E. Paquet, Deep graph convolutional reinforcement learning for financial portfolio management–DeepPocket, Expert Syst. Appl. 182 (2021) 115127.
[27] Y.-L. Hsu, Y.-C. Tsai, C.-T. Li, Fingat: Financial graph attention networks for recommending top-k k profitable stocks, IEEE Trans. Knowl. Data Eng. 35 (1) (2021) 469–481.
[28] A. Jafari, S. Haratizadeh, GCNET: graph-based prediction of stock price movement using graph convolutional network, Eng. Appl. Artif. Intell. 116 (2022) 105452.
[29] S. Xiang, D. Cheng, C. Shang, Y. Zhang, Y. Liang, Temporal and heterogeneous graph neural network for financial time series prediction, in: Proceedings of the 31st ACM International Conference on Information & Knowledge Management, 2022, pp. 3584–3593.
[30] W. Yin, Z. Chen, X. Luo, B. Kirkulak-Uludag, Forecasting cryptocurrencies' price with the financial stress index: a graph neural network prediction strategy, Appl. Econ. Lett. 31 (7) (2024) 630–639.
[31] Ö. Ekmekcioğlu, M.Ç. Pınar, Graph neural networks for deep portfolio optimization, Neural Comput. Appl. 35 (28) (2023) 20663–20674.
[32] A. Lazcano, P.J. Herrera, M. Monge, A combined model based on recurrent neural networks and graph convolutional networks for financial time series forecasting, Mathematics 11 (1) (2023) 224.
[33] H. Xu, Y. Zhang, Y. Xu, Promoting financial market development-financial stock classification using graph convolutional neural networks, IEEE Access (2023).
[34] J.M.-T. Wu, Z. Li, N. Herencsar, B. Vo, J.C.-W. Lin, A graph-based CNN-LSTM stock price prediction algorithm with leading indicators, Multimedia Syst. 29 (3) (2023) 1751–1770.
[35] D. Cheng, F. Yang, S. Xiang, J. Liu, Financial time series forecasting with multi-modality graph neural network, Pattern Recognit. 121 (2022) 108218.
[36] C. Luo, T. Ma, M. Cucuringu, Spatial-temporal stock movement prediction and portfolio selection based on the semantic company relationship graph, 2024, Available at SSRN.
[37] Z. Sun, A. Harit, A.I. Cristea, J. Wang, P. Lio, Money: Ensemble learning for stock price movement prediction via a convolutional network with adversarial hypergraph model, AI Open 4 (2023) 165–174.
[38] H. Oukhouya, H. Kadiri, K. El Himdi, R. Guerbaz, Forecasting international stock market trends: Xgboost, LSTM, LSTM-XGBoost, and backtesting XGBoost models, Statist. Optim. Inf. Comput. 12 (1) (2024) 200–209.
[39] S. Carta, A. Corriga, A. Ferreira, A.S. Podda, D.R. Recupero, A multi-layer and multi-ensemble stock trader using deep learning and deep reinforcement learning, Appl. Intell. 51 (2021) 889–905.
[40] I.E. Livieris, E. Pintelas, S. Stavroyiannis, P. Pintelas, Ensemble deep learning models for forecasting cryptocurrency time-series, Algorithms 13 (5) (2020) 121.
[41] H. Song, H. Choi, Forecasting stock market indices using the recurrent neural network based hybrid models: Cnn-lstm, gru-cnn, and ensemble models, Appl. Sci. 13 (7) (2023) 4644.
[42] F. Gu, Z. Jiang, J. Su, Application of features and neural network to enhance the performance of deep reinforcement learning in portfolio management, in: 2021 IEEE 6th International Conference on Big Data Analytics, ICBDA, IEEE, 2021, pp. 92–97.

[43] J.-S. Kim, S.-H. Kim, K.-H. Lee, Diversified adaptive stock selection using continual graph learning and ensemble approach, IEEE Access (2023).
[44] R. Du Plooy, P.J. Venter, A comparison of artificial neural networks and bootstrap aggregating ensembles in a modern financial derivative pricing framework, J. Risk Financ. Manag. 14 (6) (2021) 254.
[45] Z. Hao, H. Zhang, Y. Zhang, Stock portfolio management by using fuzzy ensemble deep reinforcement learning algorithm, J. Risk Financ. Manag. 16 (3) (2023) 201.
[46] K. He, Q. Yang, L. Ji, J. Pan, Y. Zou, Financial time series forecasting with the deep learning ensemble model, Mathematics 11 (4) (2023) 1054.
[47] X. Yu, W. Wu, X. Liao, Y. Han, Dynamic stock-decision ensemble strategy based on deep reinforcement learning, Appl. Intell. 53 (2) (2023) 2452–2470.
[48] Y. Li, S. Jiang, Y. Wei, S. Wang, Take bitcoin into your portfolio: a novel ensemble portfolio optimization framework for broad commodity assets, Financ. Innov. 7 (1) (2021) 63.
[49] Z. Zhou, Z. Song, T. Ren, L. Yu, Two-stage portfolio optimization integrating optimal sharp ratio measure and ensemble learning, IEEE Access 11 (2022) 1654–1670.