Original Research Article

LSTM-GNN Synergy: A New Frontier in Stock Price Prediction

Abstract

Aims/ Objectives: The main objective of this research is to establish the forecasting potential of various Machine Learning models, notably LSTM and GRaph Neural Networks, for predicting the price of stocks. Therefore, the research seeks to envision a mixed model with LSTM and GNN to increase the performance of predicting stock markets by utilizing sequential data and graph structure relationships.
Design: The design is experimental since the stock market data were used to assess the performance of the machine learning models.
Place and Duration of Study: The model was trained and evaluated using stock market data retrieved from Yahoo Finance, which was available from 2020 to the present.
Methodology: The study consisted of data preparation, where the stock prices were normalized. This was followed by the development of LSTM and GNN models. The two models were used jointly to develop a hybrid LSTM-GNN model. The models were fitted to a training data set and tested on the test dataset using performance measures such as root mean square error(RMSE), Mean absolute error(MAE), and R squared.
Results: The findings showed that GNN outperformed the LSTM model, which had lower RMSE. The hybrid LSTM-GNN model had the highest R2 score and the lowest prediction errors. The hybrid model managed to show an enhanced prediction of stock prices compared to the individual models; thus, the effectiveness of the hybrid model is confirmed.
Conclusion: The hybrid LSTM-GNN model for stock price prediction has provided an innovative approach by integrating sequential patterns (LSTM) and interdependencies among inputs (GNN). Future research should focus on improving the model and evaluating it against actual financial datasets to leverage its capabilities for financial prediction in a wider context.

Keywords: Stock Price Prediction; Machine Learning; LSTM; GNN; Hybrid Model; Financial Forecasting

1 Introduction

The financial market continues to be one of the most dynamic in the world, and predicting stock prices remains one of the most complex tasks to accomplish. This is mainly due to the ever-changing nature of this market alongside the interconnectedness of different financial assets. A series of models must be constructed to focus not only on a time-series analysis of the movement of stock prices and on a time-series analysis of other stocks. Traditional models, such as linear regression and autoregressive integrated moving averages (ARIMA) [1], have been widely used in time-series forecasting. Unfortunately, in most instances, these approaches cannot capture non-linear relationships that are part and parcel of stock price time series data. On the other hand, The most recent patterns pertaining to technological models, including deep learning, show great promise in overcoming the above challenges.

Out of these methods, Long Short-Term Memory (LSTM) [2] has proven to be rather practical for time series forecasting. LSTMs are a type of recurrent neural network (RNN) meant to remember important data; in this case, it is supposed to remember history to predict future stock prices better. Other popular models are Graph Neural Networks (GNN) [3] owing to their strength in representing the dependence structure of different entities within a stock graph such that there is valid dependence among several stocks owing to industry, market, and country factors.

This study develops a novel integrated method for stock price prediction by hybridizing Graph Neural Networks (GNNs) and LSTM networks. The study aims to use LSTM to capture the sequential dependencies of stock price movement and GNN to account for the relational dependency between stocks to improve prediction performance. This research aims to integrate these two models and provide better and stronger stock price forecasting compared to other stock price forecasting approaches that do not involve machine learning techniques or are even standalone machine learning techniques.

The importance of this work stems from the fact that if successful, the accuracy of forecasting stock prices will increase, which should be relevant to investors, traders, and financial analysts. The proposed integrated LSTM-GNN model can be used to track intricate features and interdependencies between stock prices and stock price movements, so it can help improve financial forecasting models.

1.1 Problem Statement

Predicting the value of stocks is a complicated challenge due to the volatility that characterizes the stock market. Econometric techniques, notably ARIMA and linear regression, fail to address the fact that financial data is volatile and complex. Advances in machine learning, particularly deep learning architectures such as LSTMs and GNNs, appear to create new opportunities. Nevertheless, stock price prediction is still a challenging task, especially when considering the movements of firms caused by external factors such as market dynamics, economic events or relationships between various stock assets. This research seeks to overcome these challenges by presenting a hybrid LSTM-GNN model capable of time-series analyses and relational analysis simultaneously. These objectives will thus help the research in enhancing both the robustness and the precision of stock price predictions.

1.2 Significance of the Research

There are several reasons why this study is important:

- It presents a hybrid system that integrates LSTM and GNN which are naturally convivial and thus enable modeling of stock price dynamics better.
- The new model allows investors to, traders and analysts to undertake forecasting of stock prices with greater accuracy.
- This paper adds to the existing pool of research on machine learning in finance, especially in hybrid models of stock price forecasting.

1.3 Objectives

In this regard, the most important aims of this study will be:

- Constructing a hybrid LSTM-GNN model that is suitable for accurate stock price estimations.
- Testing the efficacy of the model on different stock datasets.
- Checking the hybrid model estimates in consideration of estimates by other traditional and deep learning models.
- Assessing how other graph embeddings and time embeddings influence the prediction of the model accuracy.

1.4 Overview of the Paper

The rest of the paper is organized as follows:

- Section 2 contains stock price estimation based on machine learning article review, focusing on deep learning branches such as LSTM GNN, etc.
- Section 3 describes the processes used in the research with specific emphasis on the hybrid LSTM-GNN model architecture, data preparation procedures, model building and training activities.
- Section 4 shows the configuration of the experiments conducted, the types of data sets adopted and relevant measures of performance and how the proposed and baseline models performed.
- Section 5 explores the results obtained from the experiments, model analysis, and a comparison of the model's performance to other models in the literature.
- Lastly, in Section 6, the paper is rounded up with the conclusions and the contribution in terms of the findings and areas of possible future work in the field.

2 Related Work

Stock price prediction has remained one of the most difficult forecasting problems in finance. In recent years, however, machine learning and deep learning algorithms have grown in popularity for stock price prediction because of the increasing complexities of financial data. This section focuses on the existing literature on stock price prediction with machine learning but emphasizes the predictive uses of Long Short-Term Memory (LSTM) networks and Graph Neural Networks (GNN), and some models which use these models in combination.

2.1 Machine Learning in Stock Price Forecasting

The first versions of stock price prediction were developed using classical statistical methods, including the autoregressive integrated moving average (ARIMA) and exponential smoothing methods. Although these models were appropriate for time series analysis, they were unable to take into account differences embedded in the data and, therefore, did not perform well in the context of predicting stock prices.

With the improvement of machine learning (ML), various models have been used to estimate stock prices, including decision trees, support vector machines (SVM), and random forests. These models would better cope with the problems' non-linearity but had problems in capturing long-term dependencies and temporal periodicities in the price time series. However, the recent development of deep learning through the increase of interest in recurrent neural networks (RNN) and their special form, Long Short-Term Memory (LSTM), has greatly improved prediction accuracy.

2.2 LSTM Networks for Stock Price Prediction

In view of that, LSTM networks work to deal with the vanishing gradient problem, which is the major drawback of RNN. Inferring from the name, LSTMs can remember information for a long time with the help of memory cells. This feature integrates well with the logical reasoning that because previous prices influence stock prices, LSTMs are great for fitting sequential data. Several researchers have attempted to apply LSTMs to predict the price of shares. For example, [4] was the first to propose the LSTM, and subsequent papers such as [5] showed that LSTM networks have performed better than classical methods such as ARIMA in estimating stock prices.

The LSTM network's functionality can be expressed through a set of key equations. First, the forget gate f_t, input gate i_t, cell state C_t, and output gate o_t are computed as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t \cdot \tanh(C_t)$$

Where: - σ is the sigmoid activation function, - $\tanh$ is the hyperbolic tangent function, - W_f, W_i, W_C, W_o are the weight matrices, - b_f, b_i, b_C, b_o are the bias terms.

[6] applied LSTM to predict stock prices based on historical price data and technical indicators, achieving significant improvements in prediction accuracy compared to simpler machine learning models. Similarly, [7] combined LSTM with other deep learning models to enhance prediction performance by capturing both short-term and long-term dependencies in stock price movements. Despite these successes, LSTMs are limited by their inability to capture complex relationships between different financial entities or external factors influencing stock prices, such as market sentiment or macroeconomic events.

2.3 Graph Neural Networks in Finance

Several financial applications have been transitioning to using Graph Neural Networks to uncover relations between various stocks, sectors, and other financial entities. The representation of such structures in the form of graphs allows the model to determine how the decline of one stock might affect other stocks within the same industry or are related through other economic or fundamental factors.

The significance of GNN models to certain financial datasets has only recently been unevenly uncovered, with a significant focus on [8] shifting to conducting further research on how GNNs can be used for making financial forecasts. For instance, [9] used the GNN model to help predict the correlations between stocks in the same sectors and the results have encouraged graph-based models for forecasting than time series. Similarly, [10] leverages the GNN framework by using the Graph Convolutional Network (GCN) to predict the prices of various stocks by analyzing the interactions in a market graph.

2.4 Hybrid Models for Stock Price Prediction

As highlighted previously, hybrid models have increased popularity, which utilize the strengths of multiple machine learning techniques for stock price prediction purposes. These models aim to take advantage of the strengths available in different algorithms. For example, [11] developed a hybrid deep learning model that incorporated LSTM and convolutional neural networks (CNNs) in predicting stock prices, and it performed significantly better than other models.

Also, more tried and tested methods, such as LSTM together with random forests, were also integrated to compare the forecasting effectiveness of the models [12].

However, a gap exists in the literature as not much work has been done concerning LSTM and GNN hybrid models designed for stock price forecasts despite the potential of such models. This seems quite an important gap since, on the one hand, LSTM captures the sequential aspect of price dynamics, and on the other hand, GNN captures the relations between stocks. Combining these two models could lead to a more robust and accurate stock price prediction system.

2.5 Summary of Related Work

There has been notable advance in the literature on time series prediction using machine learning, with LSTM and GNN being among the most formidable forecasting models available. In essence, LSTMs are particularly good at learning sequences, while GNNs can learn distributions over networks of financial institutions. Nevertheless, models that combine the two approaches are not well researched. This research seeks to solve this problem by proposing a hybrid LSTM-GNN model that integrates the temporal and relational attributes to enhance the system's forecasting performance. The model is expected to harness the benefits of both methods and, therefore, enhance the forecasting performance, providing a comprehensive solution to the problem of stock price forecasting.

3 Methodology

This section defines the structure of the proposed hybrid model for forecasting stock prices that combines Long Short-Term Memory (LSTM) networks and Graph Neural Networks (GNNs) to leverage the aspects of time and relationships embedded in the financial data, respectively. The model consists of two parts: (1) the LSTM part, which handles the temporal sequences, and (2) the GNN part, which deals with the relationships within the data. Such integration takes advantage of both models and boosts the forecasting accuracy.

3.1 Data Preprocessing

The dataset used for stock price prediction includes historical stock market data, such as stock prices and volumes, together with additional features like sentiment of the market and various technical indicators. The dataset may also require a certain degree of adjustment in order to be appropriate for the LSTM-GNN model.

3.1.1 Preparation of Time Series Data

The LSTM component of the LSTM-GNN model uses stock price data in the time series form [13]. Time series uses a fixed number of stock prices for specific days, prices of the previous day, and other features such as trading volumes and particular indicators within that time slot. The sequences collected are then subjected to normalization so that the model can train effectively, irrespective of the different scales of the features.

3.1.2 Construction of Graph

In the GNN module, a graph is created in which each node represents a stock, and edges between these nodes indicate the interrelations between the stocks. Such relationships include, but are not limited to, the industries or the markets they serve. The graph may be constructed manually with some auxiliary information, such as industry classification, or automatically via cross-correlations between several stocks. Other features associated with each node include those of the stocks, such as the historical prices, the trading volumes, and more.

3.2 LSTM Model

To appropriately model the temporal series, which typifies the stock price time series data, we employ Long Short Term Memory (LSTM) networks initially proposed by [14]. In this situation, the LSTM model is designed to predict the closing price of a stock for the next day based on the stock prices of the last few days. The structure of LSTM consists of a number of LSTM layers wherein each member has to possess forget input and output gates that aid the model in remembering long-term relations embedded in the input.

The model receives preprocessed time series data as input data. The output is a stock price prediction. Out of the MSE, MAE, and RMAE, a mean squared error MSE was selected for training the model during the course of this research as it evaluates the mean squared difference between the predicted stock price and the actual price.

3.2.1 GNN Component

The GNN component [15] models the relational dependencies between stocks. A graph $G = (V, E)$ is constructed, where each node $v_i \in V$ represents a stock, and each edge $e_{ij} \in E$ represents the relationship between stock v_i and stock v_j. These relationships can be based on various factors such as sectoral correlations, market movements, or economic news. The GNN propagates information through the graph by updating node features based on the neighboring nodes' features, allowing the model to capture how the price movements of one stock influence another.

Formally, the GNN can be described by the following update rule for each node v_i at time step t:

$$h_i^{(t)} = \sigma\left(W \cdot \sum_{j \in \mathcal{N}(i)} \frac{1}{c_{ij}} h_j^{(t-1)} + b\right)$$

Where: - $h_i^{(t)}$ represents the hidden state of node v_i at time t, - $\mathcal{N}(i)$ denotes the set of neighboring nodes to node i, - c_{ij} is a normalization factor for the edge between node i and node j, - W and b are the learnable weight matrix and bias vector, respectively.

The GNN captures the dependencies between stocks through these updates, allowing it to model how the movements of one stock can affect others based on their relationships.

3.3 Hybrid Model Integration

To integrate the outputs of the LSTM and GNN components [16], we concatenate the prediction from the LSTM network with the output from the GNN. The final prediction for the stock price is then obtained by passing this combined feature vector through a fully connected layer followed by a regression layer. The final output $\hat{y}_t$ at time t is given by:

$$\hat{y}_t = \text{ReLU}\left(W_{\text{final}} \cdot \left[h_{\text{LSTM}}^{(t)}, h_{\text{GNN}}^{(t)}\right] + b_{\text{final}}\right)$$

Where: - $h_{\text{LSTM}}^{(t)}$ is the output from the LSTM component at time t, - $h_{\text{GNN}}^{(t)}$ is the output from the GNN component at time t, - W_{final} and b_{final} are the learnable weights and bias for the final fully connected layer, - ReLU is the activation function used for the output layer.

The combination of these two components enables the model to capture both the temporal and relational dependencies, improving the overall prediction accuracy.

3.4 Training the Model

Hybrid LSTM-GNN model training focuses on the simultaneous persistent optimization of the LSTM and GNN components. Therefore, the objective function is a weighted average of the MSE loss from the two branches, one from LSTM and the other from GNN. Backpropagation was used with the Adam optimizer to train the model, which is effective for deep learning model training.

During training, the LSTM section learns the temporal patterns present in the stock price data, while the GNN section learns the interactions between stocks in the market. Thus, when these two models are combined in the hybrid architecture, the model is able to learn not only short-term temporal dependencies but also long-term relational dependencies to enhance the prediction accuracy.

A more straightforward approach would be that the hybrid LSTM-GNN model is trained with the MSE loss function, which is widely used with regression. The MSE loss is expressed as:

$$\mathcal{L} = \frac{1}{T} \sum_{t=1}^{T} (y_t - \hat{y}_t)^2$$

Where: - y_t is the true stock price at time t, - $\hat{y}_t$ is the predicted stock price at time t.

We use gradient descent-based optimization algorithms, such as Adam, to minimize this loss function and update the model parameters.

3.5 Evaluation Metrics

In the context of forecasting stock prices, a variety of metrics are considered while evaluating the performance of the suggested hybrid LSTM-GNN model. These include:

- **Mean Absolute Error (MAE)**: It is the average of the absolute differences between prediction and actual values, irrespective of the direction of error.
- **Root Mean Squared Error (RMSE)**: Involves the average of the squared differences between prediction and actual values so that a more significant discrepancy is compensated more, as emphasized by taking the square root of the value.
- **R-squared (R^2)**: It provides an understanding of the extent of variation in the dependent variable's value, which can be explained by its relationship with the independent variables within the framework of a model.

To assess the impact of using the hybrid model, the results will be analyzed against an array of machine learning and deep learning models, such as Random Forest and standalone LSTM networks.

3.6 Implementation Details

Lyublinskaya et al. Hybrid LSTM-GNN Model. The implementation process also includes software for data processing, graph construction, and model performance evaluation. So, the time-series data is prepared and input into the LSTM branch, while the graph information is processed by the GNN branch. Model training takes place in a cloud environment, which provides GPU acceleration to shorten the time necessary to train the deep learning models.

4 Experiment and Results

4.1 Experimental Setup

Supervised deep learning algorithms trained on the processed dataset and the hybrid models—LSTM, GNN, and LSTM GNN. The evaluation is based on the models' ability to predict stock price movements on the test data corresponding to 2023. The key performance indicators (KPIs) used for evaluation are [17]:

- **Root Mean Squared Error (RMSE)**: The RMSE calculates the average value of the squared differences between the predicted and the stock prices that were actually observed.
- **Mean Absolute Error (MAE)**: MAE, in absolute terms, considers the average of absolute errors between prediction and observed values concerning stock prices.
- **R^2 Score**: An R^2 score is the proportion of the variance in the stock price data that the model can explain.

The models were then fitted as follows:

LSTM Model: The LSTM model was set to train for 50 epochs with a batch of 32 and a 0.001 learning rate using Adam. The model structure included 2 LSTM layers and a dense layer in the output end.

GNN Model: The Graph Neural Network employed Graph Convolutional Network (GCN) layers to model the inter-relations within the stocks. The GNN model was trained over 50 epochs at a batch size of 32 while the learning rate was 0.001.

Hybrid LSTM-GNN Model: The hybrid model contained both LSTM and GNN where the output of both models was taken and averaged to make an overall prediction. All training characteristics were the same as those on the single models.

4.2 Dataset

This research uses stock price data obtained from Yahoo Finance. The sample data comprises the historical prices of different stocks across several years. The daily data that was incorporated for prediction included the opening price, closing price, maximum price, minimum price, and trading volume. Such variables give a detailed picture of the stock's behavior over time.

Several firms across different sectors were used to obtain data, meaning that the model's scope is expansive across several dynamics and sectoral phenomena. The data set was processed for completeness so that there was no missing value, and normalization of the data set was done to standardize the values of each variable across measurements. Other added parameters include moving averages (MA), Relative Strength Index (RSI), and Bollinger Bands to improve the model's predicting power.

The training set comprises 80

In this regard, the data points were formatted as expanded because each data point corresponds to a single day's stock trading, which solves the temporal aspect of the model's input.

4.3 Results

The performance of the LSTM, GNN, and Hybrid LSTM-GNN models on the test set is summarized in the table below:

Model	RMSE	MAE	R^2 Score
LSTM	0.72	0.58	0.86
GNN	0.11	0.92	0.85
Hybrid LSTM-GNN	**0.03**	**0.08**	**0.88**

Table 1: Performance Comparison of LSTM, GNN, and Hybrid LSTM-GNN Models on Test Set

From the results, The table presents a performance comparison of three different models—LSTM, GNN, and the proposed Hybrid LSTM-GNN—on a test set, with respect to three commonly used evaluation metrics in regression tasks: RMSE (Root Mean Squared Error), MAE (Mean Absolute Error), and R^2 (R-squared) score.

The table compares two models, LSTM and GNN, with the proposed model Hybrid LSTM-GNN, with respect to three evaluation metrics: RMSE, MAE, and R squared score on a test set, which is a criterion in regression tasks.

4.4 Explanation of Metrics

- **RMSE (Root Mean Squared Error)**:

  This metric is computed as the square root of the average of the squared discrepancies between the estimates and the observations. On the other hand, when the RMSE result is low, it means that the model has done well. For instance, in the case study, the model termed as **Hybrid LSTM-GNN**, which has an RMSE of 0.03 model out-the LTSM and GNN, which had a higher value of RMSE of 0.72 and 0.11, respectively. Thus, the Hybrid model has the least margin of error when making a prediction as it performed better.

- **MAE (Mean Absolute Error)**:

  MAE is the mean of the absolute difference between estimates and observed values. Lower values of MAE also indicate a model's good performance. In this case, the model termed as **Hybrid LSTM-GNN** had the best performance with the lowest MAE of 0.08; LSTM followed it with an MAE of 0.58 and GNN 0.92. This implies that the Hybrid model, on average, makes a lesser percentage of error in its prediction compared to the LSTM and the GNN models.

R^2 Score Note this particular measure is sometimes called the coefficient of determination, the goodness-of-fit or the explained variance. It combines Angholi's R Super Subscript 2 with Fisher's Least Total Squares-Weighted Binary Consequence Factor and Fisher's General Linear Perfect Relation.

When R^2 equals zero, it can be considered that the regression line does not explain any variation. A higher R^2 score indicates that the model more closely fits the data. The Hybrid LSTM-GNN model achieves the highest R^2 score. According to the authors, such a conclusion was possible because there are stocks for which enough analysis was performed and MAE beneficial yields were achieved.

4.5 Discussion

The hybrid LSTM-GNN model outperformed other models in its ability to predict stock prices. This was brought about by the fact that LSTM and GNN were integrated so that time-based analysis and cross-sectional stock dependence structures of stock were captured simultaneously. Thus, combining the two approaches enhanced the model's predictive power over the existing models.

The **LSTM model**, on the other hand, was able to model the series' temporal relations, which is a key aspect in time series analysis.

The **GNN model** created meaningful relationships between stocks. This is very significant because, in financial markets, one stock influences the other through broad market movement and events concerning the sector and other stocks.

- The **Hybrid LSTM-GNN model** improved the prediction of LSTM and GNN models and reduced prediction error, resulting in a better R^2 score or better model fit.

From these findings, it can be concluded that hybrid LSTM-GNN is a model that can be relied on in stock price prediction. Future enhancements could include adding more features such as macroeconomic variables, news and social media sentiment, which could further strengthen the model's predictive capabilities.

5 Conclusion

This work proposed a hybrid LSTM-GNN model for stock price prediction, effectively combining the sequential characteristics of LSTMs with the relational dynamics of GNNs. The evaluation results show that the hybrid model performs better than the individual models when measured against RMSE, MAE, and R^2 scores. This demonstrates a good potential for this approach in financial forecasting, and further studies will work on improving the model and testing it on various markets and datasets.

6 Conclusion and Future Work

6.1 Conclusion

In this paper, we implemented a hybrid model consisting of Long Short-Term Memory (LSTM) and Graph Neural Networks (GNN) models to predict stock prices with retrieval of Yahoo Finance database. The proposed Hybrid LSTM-GNN model used the sequential dependence of the stock price LSTM and the stock dependencies GNN and, therefore, improved the prediction accuracy compared to standalone models.

The experimental results demonstrated that:

- In the experiments, the stock data was temporal, and such dependencies were effectively incorporated into the model's predictions by the LSTM.
- The GNN model created a highly effective model of inter-stock relationships; however, combined with the GNN did not outperform the LSTM model.

The hybrid LSTM-GNN model proved superior to both uni-models by obtaining the lowest RMSE, MAE, and the highest R^2 score in this case.

Overall, the hybrid model demonstrated some degree of effectiveness in predicting stock price with expectant uses in algorithmic trading, portfolio management, and financial forecasting in practice.

6.2 Future Work

The hybrid LSTM-GNN model has given good results, but there's still a substantial amount of research that could be done to enhance the model's predictive ability:

- **Incorporating Additional Features:** Future studies might include incorporating other forms of financial variables like economic variables (GNP: gross national product, inflation rates) or public sentiment based on financial news or social media trends to enable a better understanding of the market inside the model.

- **Model Refinements:** There other classes of GNN architectures such as Graph Attention Networks (GAT) or Graph Isomorphism Networks (GIN) that could be employed in place of LSTM-GNN to improve the hybrid model at capturing intricate relations between stocks.
- **Cross-Market Validation:** Different regions datasets such as European, Asian, and other emerging markets datasets could be useful to test VGG model on and measure its efficacy and reliability over several markets.
- **Hyperparameter Optimization:** Further hyperparameter adjustment, including the number of LSTM layers, depths of GNN, and learning rates, would be helpful in improving the model performance.
- **Real-Time Stock Prediction:** To begin with, using the model in a situation that requires live market data predicting stock is a valuable next step for the model, which in turn would entail a need for continuous learning for real-world changes in the market.

Furthermore, the hybrid model so constructed could be potent in predicting stock prices and even managing portfolios across and within sectors of finance.

References

[1] Andrea L Schaffer, Timothy A Dobbins, and Sallie-Anne Pearson. Interrupted time series analysis using autoregressive integrated moving average (arima) models: a guide for evaluating large-scale health interventions. *BMC medical research methodology*, 21:1–12, 2021.

[2] Greg Van Houdt, Carlos Mosquera, and Gonzalo Nápoles. A review on the long short-term memory model. *Artificial Intelligence Review*, 53(8):5929–5955, 2020.

[3] Xin Zheng, Yi Wang, Yixin Liu, Ming Li, Miao Zhang, Di Jin, Philip S Yu, and Shirui Pan. Graph neural networks for graphs with heterophily: A survey. *arXiv preprint arXiv:2202.07082*, 2022.

[4] S Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 1997.

[5] Thomas Fischer and Christopher Krauss. Deep learning with long short-term memory networks for financial market predictions. *European journal of operational research*, 270(2):654–669, 2018.

[6] Ariel Navon and Yosi Keller. Financial time series prediction using deep learning. *arXiv preprint arXiv:1711.04174*, 2017.

[7] Hyun Jun Park, Youngjun Kim, and Ha Young Kim. Stock market forecasting using a multi-task approach integrating long short-term memory and the random forest framework. *Applied Soft Computing*, 114:108106, 2022.

[8] Hao Peng, Ruitong Zhang, Yingtong Dou, Renyu Yang, Jingyi Zhang, and Philip S Yu. Reinforced neighborhood selection guided multi-relational graph neural networks. *ACM Transactions on Information Systems (TOIS)*, 40(4):1–46, 2021.

[9] Ke Xu, You Wu, Haohao Xia, Ningjing Sang, and Bingxing Wang. Graph neural networks in financial markets: Modeling volatility and assessing value-at-risk. *Journal of Computer Technology and Software*, 1(2), 2022.

[10] Shangzhe Li, Junran Wu, Xin Jiang, and Ke Xu. Chart gcn: Learning chart information with a graph convolutional network for stock movement prediction. *Knowledge-based systems*, 248:108842, 2022.

[11] Lieyun Ding, Weili Fang, Hanbin Luo, Peter ED Love, Botao Zhong, and Xi Ouyang. A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory. *Automation in construction*, 86:118–124, 2018.

[12] Hongli Niu, Kunliang Xu, and Weiqing Wang. A hybrid stock price index forecasting model based on variational mode decomposition and lstm network. *Applied Intelligence*, 50:4296–4309, 2020.

[13] Zhenyu Liu, Zhengtong Zhu, Jing Gao, and Cheng Xu. Forecast methods for time series data: a survey. *Ieee Access*, 9:91896–91912, 2021.

[14] Wenshu Zha, Yuping Liu, Yujin Wan, Ruilan Luo, Daolun Li, Shan Yang, and Yanmei Xu. Forecasting monthly gas field production based on the cnn-lstm model. *Energy*, 260:124889, 2022.

[15] Junhan Yang, Zheng Liu, Shitao Xiao, Chaozhuo Li, Defu Lian, Sanjay Agrawal, Amit Singh, Guangzhong Sun, and Xing Xie. Graphformers: Gnn-nested transformers for representation learning on textual graph. *Advances in Neural Information Processing Systems*, 34:28798–28810, 2021.

[16] Po-Chih Kuo, Yuan-Tung Chou, Kuang-Yao Li, Wei-Tze Chang, Yin-Nan Huang, and Chuin-Shan Chen. Gnn-lstm-based fusion model for structural dynamic responses prediction. *Engineering Structures*, 306:117733, 2024.

[17] Timothy O Hodson. Root mean square error (rmse) or mean absolute error (mae): When to use them or not. *Geoscientific Model Development Discussions*, 2022:1–10, 2022.