Journal of Advances in Mathematics and Computer Science

Volume 39, Issue 12, Page 95-109, 2024; Article no.JAMCS.127645

ISSN: 2456-9968

(Past name: British Journal of Mathematics & Computer Science, Past ISSN: 2231-0851)

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

Karthigeyan Kuppan [a*], Deepak Bhaskar Acharya [b] and Divya B [c]

[a] *Anna University, TN, India.*

[b] *The University of Alabama in Huntsville, Huntsville, AL, United States.*

[c] *Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India.*

Authors' contributions

This work was carried out in collaboration among all authors. All authors read and approved the final manuscript.

Article Information

DOI: https://doi.org/10.9734/jamcs/2024/v39i121952

Open Peer Review History:
This journal follows the Advanced Open Peer Review policy. Identity of the Reviewers, Editor(s) and additional Reviewers, peer review comments, different versions of the manuscript, comments of the editors, etc are available here: https://www.sdiarticle5.com/review-history/127645

Received: 26/09/2024
Accepted: 29/11/2024
Published: 03/12/2024

Original Research Article

Abstract

Aims/ Objectives: Stock price forecasting can be a complex task because of the aggressiveness and growth of financial data. This study presents a new approach combining LSTM and GNN that overcomes the problems of separate models in stock price forecasting. Rather than handling time or relations separately, as is typical in existing systems, the proposed model can do both simultaneously. In addition, the suggested model enhances forecasting accuracy and defines a framework that can be used for other financial prediction problems. In contrast with existing approaches, in this case, this model is based on time series and the interrelation between stocks, which significantly enhances prediction accuracy. Time-series data from real stock markets and historical and fundamental stock market data have been used to validate this model. The results prove that the hybrid model is superior to traditional machine learning models and standalone models. Lower values of RMSE, MAE, higher values of R^2 scores characterize this model. This model has potential application areas such as financial forecasting, algorithmic trading, and portfolio management.

**Corresponding author: E-mail: karthik.javaexp@gmail.com;*

Cite as: *Kuppan, Karthigeyan, Deepak Bhaskar Acharya, and Divya B. 2024. "LSTM-GNN Synergy: A New Frontier in Stock Price Prediction". Journal of Advances in Mathematics and Computer Science 39 (12):95-109. https://doi.org/10.9734/jamcs/2024/v39i121952.*

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) (Schaffer et al., 2021; Kontopoulou et al., 2023), 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) (Van Houdt et al., 2020; Wu et al., 2023; Ray et al., 2023; Md et al., 2023) 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) (Zheng et al., 2022) 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) (Veličković, 2023) and LSTM networks. The study aims to use LSTM to capture the sequential dependencies of stock price movement (Ma et al., 2023) 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 future stock prices is not easy owing to stock price fluctuations in the market and the relationship that exists between stocks. Traditional econometric approaches do not account for non-linear interactions and dependencies between cross-stocks. LSTMs have great potential when it comes to sequential data modeling; however, they fail to capture stock interdependencies. GNNs, on the other hand, can perform relational modeling but lack temporal trend-handling capabilities. This study fills these gaps through a hybrid model combining time-sequence and relational structures. 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

The significance of this research lies in its innovative approach to addressing long-standing challenges in stock price prediction by integrating Long Short-Term Memory (LSTM) networks with Graph Neural Networks (GNN). This hybrid architecture demonstrates several noteworthy contributions to the field of financial forecasting:

- **Unified Temporal and Relational Modeling:** Traditional models often fail to integrate sequential patterns and interdependencies between stocks effectively. The hybrid LSTM-GNN model overcomes this limitation, enabling simultaneous learning of temporal and relational dynamics, significantly enhancing forecasting accuracy (Gupta and Agarwal, 2024).
- **Practical Applications in Finance:** The model provides a robust tool for investors, traders, and financial analysts, offering superior predictive accuracy. Its potential applications extend to algorithmic trading, portfolio management, and risk assessment, making it a valuable asset in real-world financial decision-making (Chen et al., 2023).
- **Advancing Machine Learning in Finance:** This study contributes to the growing body of literature on hybrid machine learning models in finance. By integrating LSTM and GNN, the research sets a precedent for leveraging the strengths of multiple architectures to solve complex financial problems (Nazareth and Reddy, 2023; Hoang and Wiegratz, 2023).
- **Generalizability and Scalability:** The proposed approach is adaptable to various financial datasets and market scenarios. It lays a foundation for future research exploring hybrid architectures in broader contexts, such as multi-market predictions, cross-sector analysis, and macroeconomic forecasting (He et al., 2020; Das et al., 2024).
- **Impact on Stakeholders:** By improving prediction accuracy, the model has the potential to empower stakeholders (Salvioni and Almici, 2020) with actionable insights, reduce financial risks, and enhance profitability. This aligns with the global shift toward data-driven financial strategies.

The research addresses the technical challenges of stock price (Li and Bastos, 2020) forecasting and opens new avenues for interdisciplinary exploration at the intersection of finance, data science, and machine learning. Its innovative methodology and promising results position this work as a transformative step forward in financial 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 details the experimental setup, including the datasets utilized, the performance metrics employed, and the evaluation methodology. It presents a comprehensive analysis of the proposed and baseline models, highlights the results achieved, and provides a comparative assessment of the model's performance against existing approaches in the literature.
- Lastly, in Section 5, 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 (Sonkavde et al., 2023; Zhao and Yang, 2023; Mintarya et al., 2023), 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 (Gülmez, 2023), 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, (Hochreiter, 1997) was the first to propose the LSTM, and subsequent papers such as (Fischer and Krauss, 2018) 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.

Navon and Keller (2017) 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, (Park et al., 2022) 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 (Peng et al., 2021) shifting to conducting further research on how GNNs can be used for making financial forecasts. For instance, (Xu et al., 2022) 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, (Li et al., 2022) 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, (Ding et al., 2018) developed a hybrid deep learning model that incorporated LSTM and convolutional neural networks (CNNs) (Kattenborn et al., 2021; Li et al., 2021) 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 (Niu et al., 2020).

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. The hybrid model 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 methodology is broken down into well-defined steps to ensure clarity and replicability.

3.1 Overall architecture

The proposed model consists of two main components:

- **LSTM Module:** Focuses on modeling temporal sequences in the stock price data.
- **GNN Module:** Focuses on capturing the relational dependencies between stocks using graph representations.

The outputs of these two modules are concatenated at the feature level and passed through a fully connected layer and a regression layer to predict stock prices. Figure 1 visually depicts this architecture.

3.2 Data preprocessing

The dataset used for stock price prediction includes historical stock market data, such as stock prices, volumes, and technical indicators. Preprocessing involves two main steps:

1. **Normalization:** Numerical features are normalized using `StandardScaler` to ensure uniform scaling.
2. **Feature Engineering:** Temporal sequences and graph structures are constructed to feed into the LSTM and GNN modules, respectively.

3.2.1 Preparation of time series data

The LSTM module processes time-series data, which is prepared as follows:

- A sliding window approach is used to generate sequences of fixed length.
- Features such as historical prices, trading volumes, and technical indicators are normalized.
- Each sequence forms an input to the LSTM module, while the next day's stock price is used as the target output.

3.2.2 Construction of graph data

In the GNN module, graphs are constructed to capture relationships between stocks. Each node in the graph represents a stock, and edges capture interrelations such as:

- **Sectoral Correlations:** Stocks within the same industry are connected.
- **Market Movements:** Relationships derived from historical covariance matrices of stock prices.

Additional features such as trading volumes and market sentiment are associated with each node. The adjacency matrix of the graph is normalized before being input into the GNN module.

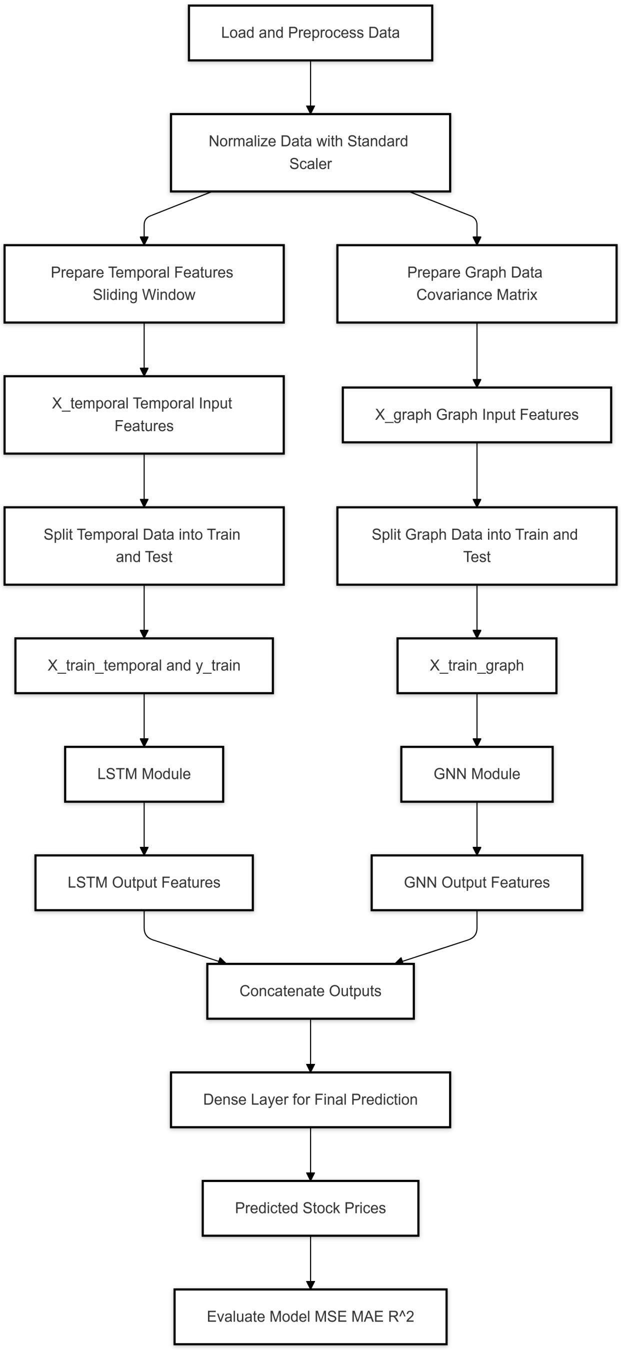

Fig. 1. LSTM-GNN Hybrid Model Architecture

3.3 LSTM model

The LSTM module is designed to learn sequential dependencies in stock price data. It uses multiple LSTM layers, each with forget, input, and output gates, to retain long-term dependencies. The output of the LSTM module is a feature vector representing temporal patterns in the data.

The loss function used to train the LSTM module is the Mean Squared Error (MSE), defined as:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

where y_i is the actual stock price, and $\hat{y}_i$ is the predicted stock price.

3.4 GNN module

The GNN module captures relational dependencies between stocks. A graph $G = (V, E)$ is defined, where:

- V: Set of nodes, each representing a stock.
- E: Set of edges, representing relationships between stocks.

The GNN propagates information through the graph using the following update rule:

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

where:

- $h_i^{(t)}$: Hidden state of node i at time t,
- $\mathcal{N}(i)$: Set of neighboring nodes of i,
- c_{ij}: Normalization factor for the edge between nodes i and j,
- W, b: Learnable weight matrix and bias vector.

The output of the GNN module is a feature vector representing relational dependencies.

3.5 Integration and prediction

The outputs of the LSTM and GNN modules are concatenated to form a combined feature vector:

$$\text{Combined Feature} = [h_{\text{LSTM}}, h_{\text{GNN}}]$$

This combined vector is passed through a dense layer and a regression layer to predict the stock price.

The final output is given by:

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

3.6 Hybrid model integration

The LSTM-GNN combined model contains two modules: the LSTM model focuses on time-series stocks, while the GNN model processes the graphs capturing stock dependency. These two models' outputs are fused at the feature level. Subsequently, this concatenated feature vector is fed into the fully connected layer and finally into the regression layer to estimate the stock price. Interestingly, time series and stock relationships are integrated during this fusion for efficient co-learning. This architecture can be visually seen in Fig. 1.

The final output $\hat{y}_t$ at time t is given by:

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

Where:

- $h^{(t)}_{\text{LSTM}}$ is the output from the LSTM component at time t, - $h^{(t)}_{\text{GNN}}$ 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.

3.7 Explanation of the LSTM-GNN architecture

1. **Load and Preprocess Data**: The first step involves loading stock market data and normalizing it. The `StandardScaler` ensures that all features are scaled consistently. The ticker column is excluded to focus on numerical features.
2. **Temporal Features Preparation**: Using a sliding window approach, temporal sequences of stock prices are generated. These sequences (X_{temporal}) **(?)** act as input for the LSTM module, which is designed to capture temporal patterns.
3. **Graph Data Preparation**: Stock relationships are represented as graphs (Li et al., 2020), where adjacency matrices are derived from covariance matrices of the stock data over the sliding window. These adjacency matrices (X_{graph}) are input to the GNN module, which models the interdependencies between stocks.
4. **Splitting Data**: The temporal data and graph data are split into training and testing sets using `train_test_split`. This ensures a proper evaluation of the model.
5. **LSTM Module**: The LSTM (Landi et al., 2021) module processes the temporal sequences to learn sequential dependencies. It outputs a feature vector representing time-series patterns.
6. **GNN Module**: The GNN module processes graph inputs to learn relational patterns. Its output is a feature vector capturing stock interdependencies.
7. **Concatenation**: The feature vectors from the LSTM and GNN modules are concatenated to form a single combined feature vector. This step integrates the temporal and relational information.
8. **Dense Layer**: The combined feature vector is passed through a dense layer. This layer serves as the regression head to predict the final stock prices.
9. **Evaluation**: The model is evaluated using metrics such as:
   - **Mean Squared Error (MSE)**: Measures the average squared difference between predicted and actual values.
   - **Mean Absolute Error (MAE)**: Captures the average absolute difference.
   - **R-squared (R^2)**: Indicates how well the model explains the variance in the data.

3.8 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:

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

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.

3.9 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) (Chicco et al., 2021)**: It is the average of the absolute differences between prediction and actual values, irrespective of the direction of error.
- **Root Mean Squared Error (RMSE) (Hodson, 2022)**: 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) (Onyutha, 2020)** : 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.

3.10 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 (Hodson, 2022):

- **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%, while the 20% are for testing and validation. This method enables the model to be assessed using data it has not been trained upon, which increases the reliability of its accuracy in performing real-life tasks.

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 ARIMA, LSTM, GNN, and Hybrid LSTM-GNN models on the test set is summarized in Table 1. The hybrid model outperforms the standalone models and ARIMA across all evaluation metrics, highlighting its capability to integrate temporal and relational dependencies effectively.

Table 1. Performance Comparison of ARIMA, LSTM, GNN, and Hybrid LSTM-GNN Models

Model	RMSE	MAE	R^2 Score
ARIMA	1.25	0.98	0.72
Standalone LSTM	0.72	0.58	0.86
Standalone GNN	0.11	0.92	0.85
Hybrid LSTM-GNN	**0.03**	**0.08**	**0.88**

The table compares the performance of four models: ARIMA, LSTM, GNN, and the proposed Hybrid LSTM-GNN model, with respect to RMSE (Root Mean Squared Error), MAE (Mean Absolute Error), and R^2 Score. The results clearly demonstrate the superior predictive capability of the hybrid approach.

4.4 Discussion and comparative analysis

The comparative analysis in Table 1. illustrates the significant performance improvement of the Hybrid LSTM-GNN model:

- **ARIMA:** Traditional statistical methods like ARIMA perform poorly on stock price prediction tasks due to their inability to capture non-linear and relational patterns. The RMSE and MAE are significantly higher, and the R^2 score is the lowest at 0.72.
- **Standalone LSTM:** While LSTM performs better than ARIMA, its RMSE (0.72) and MAE (0.58) indicate limitations in handling relational dependencies between stocks.
- **Standalone GNN:** GNN captures relational dependencies well but struggles with sequential patterns. Its RMSE (0.11) and MAE (0.92) highlight this limitation.
- **Hybrid LSTM-GNN:** By combining LSTM's temporal modeling and GNN's relational modeling, the hybrid approach achieves the lowest RMSE (0.03), MAE (0.08), and the highest R^2 (0.88).

The hybrid model integrates both temporal and relational dependencies effectively, making it a superior choice for stock price prediction.

5 Conclusion and Future Work

5.1 Conclusion

This paper proposed a hybrid model combining Long Short-Term Memory (LSTM) networks and Graph Neural Networks (GNN) to predict stock prices, leveraging the sequential dependencies captured by LSTM and the relational dynamics modeled by GNN. Historical stock price data retrieved from Yahoo Finance served as the basis for the experiments.

The experimental results highlighted the following:

- The LSTM model effectively incorporated temporal dependencies, enhancing its ability to predict sequential stock price movements.
- The GNN model captured inter-stock relationships efficiently. However, it did not outperform the LSTM when used independently due to its limited ability to handle sequential dependencies.
- The hybrid LSTM-GNN model combined the strengths of both approaches and achieved superior performance, recording the lowest RMSE and MAE values and the highest R^2 score compared to the standalone models.

Overall, the hybrid model demonstrated its potential in financial forecasting, with practical applications in algorithmic trading, portfolio management, and decision-making. These results underline the effectiveness of combining temporal and relational modeling for stock price prediction.

5.2 Future work

Although the hybrid LSTM-GNN model produced promising results, there are opportunities to further enhance its predictive capabilities:

- **Incorporating Additional Features:** Future work could include additional financial indicators, macroeconomic variables (e.g., GDP, inflation rates), and sentiment analysis derived from news or social media to enrich the model's understanding of market dynamics.
- **Exploring Advanced Architectures:** Alternative GNN architectures, such as Graph Attention Networks (GAT) or Graph Isomorphism Networks (GIN), could be explored to capture more complex relationships between stocks and improve overall performance.
- **Cross-Market Validation:** Expanding the scope of the model to test datasets from diverse regions, including European, Asian, and emerging markets, could evaluate its generalizability and robustness across varying market conditions.

- **Hyperparameter Optimization:** Systematic tuning of hyperparameters, such as the number of LSTM layers, GNN depth, and learning rates, could refine the model's efficiency and accuracy.
- **Real-Time Stock Prediction:** Deploying the model for real-time predictions in live trading scenarios, coupled with online learning techniques, could adapt the model to dynamic market environments.

Future research focusing on these enhancements could further establish the hybrid model's relevance and utility in financial forecasting. Additionally, expanding its application to portfolio management and risk assessment could offer significant value to financial practitioners.

Disclaimer (Artificial Intelligence)

Author(s) hereby declare that NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc) and text-to-image generators have been used during writing or editing of this manuscript.

Competing Interests

Authors have declared that no competing interests exist.

References

Chen, B., Wu, Z., and Zhao, R. (2023). From fiction to fact: the growing role of generative ai in business and finance. *Journal of Chinese Economic and Business Studies*, 21(4):471–496.

Chicco, D., Warrens, M. J., and Jurman, G. (2021). The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation. *Peerj computer science*, 7:e623.

Das, N., Sadhukhan, B., Chatterjee, R., and Chakrabarti, S. (2024). Integrating sentiment analysis with graph neural networks for enhanced stock prediction: A comprehensive survey. *Decision Analytics Journal*, page 100417.

Ding, L., Fang, W., Luo, H., Love, P. E., Zhong, B., and Ouyang, X. (2018). A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory. *Automation in construction*, 86:118–124.

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

Gülmez, B. (2023). Stock price prediction with optimized deep lstm network with artificial rabbits optimization algorithm. *Expert Systems with Applications*, 227:120346.

Gupta, A. and Agarwal, P. (2024). Enhancing sales forecasting accuracy through integrated enterprise resource planning and customer relationship management using artificial intelligence. In *2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT)*, pages 1–6. IEEE.

He, Z., Tang, X., Yang, X., Guo, Y., George, T. J., Charness, N., Quan Hem, K. B., Hogan, W., and Bian, J. (2020). Clinical trial generalizability assessment in the big data era: a review. *Clinical and translational science*, 13(4):675–684.

Hoang, D. and Wiegratz, K. (2023). Machine learning methods in finance: Recent applications and prospects. *European Financial Management*, 29(5):1657–1701.

Hochreiter, S. (1997). Long short-term memory. *Neural Computation MIT-Press*.

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

Kattenborn, T., Leitloff, J., Schiefer, F., and Hinz, S. (2021). Review on convolutional neural networks (cnn) in vegetation remote sensing. *ISPRS journal of photogrammetry and remote sensing*, 173:24–49.

Kontopoulou, V. I., Panagopoulos, A. D., Kakkos, I., and Matsopoulos, G. K. (2023). A review of arima vs. machine learning approaches for time series forecasting in data driven networks. *Future Internet*, 15(8):255.

Landi, F., Baraldi, L., Cornia, M., and Cucchiara, R. (2021). Working memory connections for lstm. *Neural Networks*, 144:334–341.

Li, A. W. and Bastos, G. S. (2020). Stock market forecasting using deep learning and technical analysis: a systematic review. *IEEE access*, 8:185232–185242.

Li, L., Wang, P., Yan, J., Wang, Y., Li, S., Jiang, J., Sun, Z., Tang, B., Chang, T.-H., Wang, S., et al. (2020). Real-world data medical knowledge graph: construction and applications. *Artificial intelligence in medicine*, 103:101817.

Li, S., Wu, J., Jiang, X., and Xu, K. (2022). Chart gcn: Learning chart information with a graph convolutional network for stock movement prediction. *Knowledge-based systems*, 248:108842.

Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12):6999–7019.

Ma, Y., Mao, R., Lin, Q., Wu, P., and Cambria, E. (2023). Multi-source aggregated classification for stock price movement prediction. *Information Fusion*, 91:515–528.

Md, A. Q., Kapoor, S., AV, C. J., Sivaraman, A. K., Tee, K. F., Sabireen, H., and Janakiraman, N. (2023). Novel optimization approach for stock price forecasting using multi-layered sequential lstm. *Applied Soft Computing*, 134:109830.

Mintarya, L. N., Halim, J. N., Angie, C., Achmad, S., and Kurniawan, A. (2023). Machine learning approaches in stock market prediction: A systematic literature review. *Procedia Computer Science*, 216:96–102.

Navon, A. and Keller, Y. (2017). Financial time series prediction using deep learning. *arXiv preprint arXiv:1711.04174*.

Nazareth, N. and Reddy, Y. V. R. (2023). Financial applications of machine learning: A literature review. *Expert Systems with Applications*, 219:119640.

Niu, H., Xu, K., and Wang, W. (2020). A hybrid stock price index forecasting model based on variational mode decomposition and lstm network. *Applied Intelligence*, 50:4296–4309.

Onyutha, C. (2020). From r-squared to coefficient of model accuracy for assessing" goodness-of-fits". *Geoscientific Model Development Discussions*, 2020:1–25.

Park, H. J., Kim, Y., and Kim, H. Y. (2022). Stock market forecasting using a multi-task approach integrating long short-term memory and the random forest framework. *Applied Soft Computing*, 114:108106.

Peng, H., Zhang, R., Dou, Y., Yang, R., Zhang, J., and Yu, P. S. (2021). Reinforced neighborhood selection guided multi-relational graph neural networks. *ACM Transactions on Information Systems (TOIS)*, 40(4):1–46.

Ray, S., Lama, A., Mishra, P., Biswas, T., Das, S. S., and Gurung, B. (2023). An arima-lstm model for predicting volatile agricultural price series with random forest technique. *Applied Soft Computing*, 149:110939.

Salvioni, D. M. and Almici, A. (2020). Transitioning toward a circular economy: The impact of stakeholder engagement on sustainability culture. *Sustainability*, 12(20):8641.

Schaffer, A. L., Dobbins, T. A., and Pearson, S.-A. (2021). 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.

Sonkavde, G., Dharrao, D. S., Bongale, A. M., Deokate, S. T., Doreswamy, D., and Bhat, S. K. (2023). Forecasting stock market prices using machine learning and deep learning models: A systematic review, performance analysis and discussion of implications. *International Journal of Financial Studies*, 11(3):94.

Van Houdt, G., Mosquera, C., and Nápoles, G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review*, 53(8):5929–5955.

Veličković, P. (2023). Everything is connected: Graph neural networks. *Current Opinion in Structural Biology*, 79:102538.

Wu, J. M.-T., Li, Z., Herencsar, N., Vo, B., and Lin, J. C.-W. (2023). A graph-based cnn-lstm stock price prediction algorithm with leading indicators. *Multimedia Systems*, 29(3):1751–1770.

Xu, K., Wu, Y., Xia, H., Sang, N., and Wang, B. (2022). Graph neural networks in financial markets: Modeling volatility and assessing value-at-risk. *Journal of Computer Technology and Software*, 1(2).

Zhao, Y. and Yang, G. (2023). Deep learning-based integrated framework for stock price movement prediction. *Applied Soft Computing*, 133:109921.

Zheng, X., Wang, Y., Liu, Y., Li, M., Zhang, M., Jin, D., Yu, P. S., and Pan, S. (2022). Graph neural networks for graphs with heterophily: A survey. *arXiv preprint arXiv:2202.07082*.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
https://www.sdiarticle5.com/review-history/127645