



## STOCK PRICE PREDICTION IN BIST30 USING GRAPH CONVOLUTIONAL NETWORK, LONG SHORT-TERM MEMORY AND HYBRID MODELS: A COMPARATIVE ANALYSIS

Sevim ÖLMEZ<sup>1\*</sup>, Asım Egemen YILMAZ<sup>2</sup>

<sup>1</sup>Ankara University, Graduate School of Natural and Applied Sciences, Department of Artificial Intelligence Technology, 06110, Ankara, Türkiye

<sup>2</sup>Ankara University, Faculty of Engineering, Department of Electrical-Electronics Engineering, 06830, Ankara, Türkiye

**Abstract:** This study investigates the effectiveness of specific neural network architectures for predicting stock prices using data from the BIST30 index. Recognizing the inadequacy of traditional models in handling the volatile and interconnected nature of financial markets, our study introduces a hybrid deep learning model. We evaluate the performance of a Graph Convolutional Network (GCN) compared to a hybrid model that integrates both GCN and Long Short-Term Memory (LSTM) layers. This hybrid approach uniquely captures the crucial relational dependencies among BIST30 stocks (via GCNs) alongside the sequential temporal patterns in their price movements (via LSTMs). Our methodology involves constructing a stock graph for GCN-based feature extraction, while the LSTM component processes individual stock time series. The outputs from these processing streams are integrated to produce final predictions, incorporating additional external factors like dollar and gold parity. Our findings indicate that the hybrid model, by leveraging the structural insights of GCNs and the temporal memory of LSTMs, exhibits superior performance in capturing the complex dynamics within the stock market data compared to the standalone GCN approach.

**Keywords:** Stock price prediction, Graph convolutional network, Long short-term memory, Hybrid model

\*Corresponding author: Ankara University, Graduate School of Natural and Applied Sciences, Department of Artificial Intelligence Technology, 06110, Ankara, Türkiye

E mail: solmez@ankara.edu.tr (S. ÖLMEZ)

Sevim ÖLMEZ  <https://orcid.org/0009-0000-8822-9413>

Asım Egemen YILMAZ  <https://orcid.org/0000-0002-4156-4238>

Received: November 06, 2025

Accepted: January 03, 2026

Published: January 15, 2026

**Cite as:** Ölmez, S. & Yılmaz, A. (2026). Stock price prediction in BIST30 using graph convolutional network, long short-term memory and hybrid models: A comparative analysis. *Black Sea Journal of Engineering and Science*, 9(1), 362-368.

### 1. Introduction

Accurate stock price prediction is a complex and challenging task due to the inherent volatility and nonlinearity of financial markets. Traditional methods often struggle to capture the intricate patterns and dependencies that drive stock price movements. In recent years, deep learning models have emerged as promising tools for time series forecasting, offering the ability to learn complex patterns from large datasets.

This study focuses on two prominent deep learning architectures: GCN and LSTM. GCNs excel at modeling relationships between entities, making them suitable for capturing interdependence among stocks. LSTMs, on the other hand, are designed to capture long-term dependencies in sequential data, making them effective for time series analysis.

We hypothesize that a hybrid model that combines GCN and LSTM can take advantage of the strengths of both architectures to achieve improved prediction accuracy. By integrating the relational modeling capabilities of GCNs with the temporal learning capacity of LSTMs, the

hybrid model aims to capture a more comprehensive understanding of the dynamics of the stock market.

In the article (Ran et al., 2024), the model first uses LSTM to process individual stock time series data and generate node features. Then, it calculates the relationships between stocks using Pearson correlation to form the graph structure (adjacency matrix). Finally, a GCN processes the graph structure and node features to extract relevant information for predicting stock price trends.

In essence, in the article (Sonani et al., 2025) authors provided a model that first processes the sequential stock data using LSTM, builds a graph representing inter-stock relationships using correlation and association analysis, processes the graph using GNN, and then combines the learned temporal and relational features to predict stock prices. The expanding window approach ensures the model stays updated with the latest market dynamics.

Based on the analysis of these two provided articles, both propose a hybrid model combining LSTM and GCN for



stock market prediction, with the aim of using both temporal and relational dependencies in stock data. Although they share this fundamental concept, they differ in specific architectural details, methodologies to build stock relationships, and evaluation approaches.

In the article (Ran et al., 2024), the authors explicitly state using all outputs of the LSTM's hidden state as graph node embeddings. The article (Sonani et al., 2025) discusses transforming LSTM outputs into feature vectors for integration. Our study combines the output of the LSTM and Graph Convolution layers, but the specific method to utilize the hidden states of the LSTM as input to the GCN is not as explicitly detailed as in the article (Ran et al., 2024). In our study, LSTM layers and Graph Convolution layers as separate components whose outputs are later concatenated.

In summary, while the fundamental concept of combining LSTM for temporal features and GCN for relational features is consistent across (Ran et al., 2024) and (Sonani et al., 2025) and our study. Our study provides a specific implementation instance that may vary in graph construction techniques, exact integration of LSTM and GCN outputs, detailed model architecture, the dataset used, and the evaluation process compared to the methodologies outlined in (Ran et al., 2024) and (Sonani et al., 2025).

### 1.2. Related Work

The financial market, characterized by its inherent dynamism, presents a formidable challenge in the realm of stock price prediction. This complexity arises primarily from the continuous evolution of the market and the intricate interconnectedness of various financial assets. Consequently, effective predictive modeling necessitates the construction of sophisticated frameworks that go beyond simple time series analysis of individual stock price movements and extend to capturing the interrelationships between various stocks. Traditional statistical models, such as linear regression and autoregressive integrated moving averages (ARIMA) (Schaffer et al., 2021) and (Kontopoulou et al., 2023), have historically been employed for time series forecasting. However, a significant limitation of these approaches lies in their inability to adequately model the nonlinear relationships that are intrinsic to stock price time series data. In contrast, recent advances in technological models, particularly within the domain of deep learning, have demonstrated considerable promise in overcoming these inherent challenges.

Among these advanced methodologies, LSTM networks (Van et al., 2020; Lin et al., 2023; Singh et al., 2023) have emerged as highly practical tools for time series forecasting in finance. As a specialized type of recurrent neural network (RNN), LSTMs are specifically designed to retain and leverage important historical data, enabling them to better predict future stock price movements by learning from past patterns. Complementing the strengths of LSTMs are Graph Neural Networks (GNNs) (Zhang et al., 2022), which have gained prominence due

to their capacity to effectively represent the dependence structure among different entities within a stock graph. This capability allows GNNs to model the valid dependencies that exist between various stocks due to shared industry affiliations, broader market trends, and even geographical factors.

The manuscript (Veličković, 2023) introduces a novel integrated methodology for stock price prediction by strategically hybridizing Graph Neural Networks (GNNs) and LSTM networks. The core objective of this research is to leverage the unique strengths of each model: employing LSTMs to capture the sequential dependencies inherent in stock price movements and utilizing GNNs to account for the crucial relational dependencies that exist between different stocks, ultimately aiming to enhance overall prediction performance. This research endeavors to create a synergistic integration of these two powerful model types to achieve more accurate and robust stock price forecasts compared to traditional approaches that either do not employ machine learning techniques or rely on standalone machine learning models.

Building upon the established success of Graph Neural Networks (GNNs) in handling graphical data, particularly after the advent of Graph Convolutional Networks (GCNs) (Berg et al., 2017), it is crucial to consider the complementary role of Long Short-Term Memory (LSTM) networks in financial time series prediction. LSTMs have become prominent due to their inherent ability to capture long-term dependencies and effectively manage the vanishing gradient problem, making them well suited for modeling the nonlinear and volatile characteristics of stock prices (Fjellström, 2022). Their successful application in various financial contexts underscores their adaptability in recognizing intricate patterns and their robustness in the face of market volatility (Yeung et al., 2020).

Notably, comparative studies have indicated that LSTM networks often outperform traditional time series models like ARIMA in stock price forecasting, especially when dealing with nonlinear time series data (Jarrah et al., 2023). The integration of LSTM networks with GNNs into hybrid models represents a powerful paradigm that takes advantage of the distinct strengths of both methodologies, enabling the simultaneous modeling of temporal sequential patterns and complex relational data. (Cheng et al., 2022) demonstrated significant improvements in predictive accuracy by effectively combining relational insights derived from GNNs with the temporal pattern recognition capabilities of other models. Similarly, (Shi et al., 2024) observed that such hybrid architectures achieve more accurate predictions precisely because they can simultaneously capture both the evolving temporal dynamics of individual stocks and the intricate inter-stock relationships that influence market behavior. However, the development and deployment of these sophisticated hybrid models are not without their challenges. Increased computational demands due to the complexity of processing both graph

and sequential data, along with potential risks of overfitting or data leakage if not meticulously implemented, are significant considerations (Tang et al., 2021). Therefore, careful implementation strategies, rigorous validation procedures, and a deep understanding of the underlying data are essential to mitigate these potential pitfalls and ensure the reliability and generalizability of hybrid GNN-LSTM models in financial forecasting (Mehtab et al., 2020).

## 2. Materials and Methods

This study proposes a spatio-temporal graph-based deep learning framework for stock price forecasting, combining Graph Convolutional Networks (GCNs) with Long Short-Term Memory (LSTM) networks. The methodology consists of four main components: data collection and preprocessing, graph construction, spatio-temporal model architecture, training and evaluation procedures.

The dataset comprises the daily closing prices of 30 stocks from the BIST30 index. Data is collected between January 1, 2020 and June 3, 2024 resulting in a total of 1048 trading days per stock. To ensure temporal consistency and prevent information leakage, the dataset is split chronologically into training, validation and test sets using proportions of 75%, 20%, and 5% respectively. Prior to model input, the data underwent Z-score normalization on a per-stock basis. Crucially, the standardization parameters (mean and standard deviation) were derived solely from the training set and applied consistently to the validation and test sets to maintain data integrity.

Our study compares a standalone GCN and a novel hybrid GCN-LSTM model. The hybrid approach leverages graph convolutions (using sector or correlation-based edges) for spatial feature extraction, followed by an LSTM layer (16 units, ReLU) for temporal analysis. The models predict a 1-step horizon using a 2-step input sequence. Training was optimized via RMSprop with a learning rate of 0.02 and Mean Squared Error loss, utilizing early stopping (patience = 10) to ensure robust convergence within 100 epochs.

This data undergoes preprocessing and transformation to ensure it is in a format suitable for both Graph Convolutional Networks (GCNs) and Long Short-Term Memory (LSTM) models. For our GCN model, we construct a graph where individual stocks are represented as nodes. The edges in this graph signify relationships between stocks, which are determined by factors such as shared industry sectors or broader market trends. The GCN then learns to extract relevant features from this defined graph structure.

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network (RNN) architecture specifically designed for handling sequential data. Their ability to retain information over extended sequences and capture dependencies across time makes them well suited for identifying patterns within time series data. The LSTM

model processes the time series data for each stock individually, learning to recognize temporal dependencies and patterns.

Our hybrid model integrates the outputs from both the GCN and the LSTM, effectively combining relational insights with temporal information to generate the final predictions.

Essentially, the architecture processes the input data in parallel through the LSTM and GCN pathways, extracting temporal and relational features, respectively. These extracted features are then fused together and fed into dense layers to produce the final prediction. This approach leverages the distinct strengths of both network types. The specific number of LSTM and Graph Convolutional layers, along with their dimensionalities and those of the subsequent dense layers, would be defined during the instantiation of the GCN-LSTM function based on the chosen model configuration.

The architecture of the hybrid GCN-LSTM model can be explained as follows:

The model is built using the Keras functional API, allowing flexible connections between layers. The GCN-LSTM function defines the structure.

**Input Layer:** The model starts with an input layer (`keras.Input`). This layer is designed to accept the preprocessed time series data for all stocks, likely shaped as (number of samples, sequence length, number of features) where the number of features would be the number of features for each stock at each timestep (e.g., normalized closing price). The code does not explicitly show the input shape definition within the GCN-LSTM function call in the provided snippets, but this is the standard input format for sequential data in Keras.

**LSTM Layers:** Following the input, one or more layers of LSTM are added. These layers process the time series data for each stock independently across the sequence length. The LSTM layers are responsible for learning temporal dependencies and patterns within each individual stock's price history. The output of the LSTM layer(s) will contain learned representations that capture the sequential nature of the data.

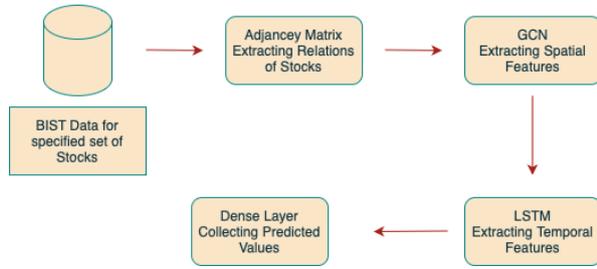
**Graph Convolutional Layer:** These layers operate on the graph structure defined earlier (based on the chosen graph option like correlation or sector). The Graph Convolutional layer takes the learned features from the LSTM (or potentially the initial node features if the LSTM output is shaped appropriately to serve as node features) and the adjacency matrix (representing the graph connections) as inputs. It aggregates information from a node's neighbors in the graph, allowing the model to learn relationships and dependencies between different stocks.

**Feature Concatenation/Combination:** The outputs from the LSTM and Graph Convolutional layers are then combined. `Concatenate` call, indicating that the learned temporal features from the LSTM and the relational features from the GCN are joined together. This concatenated vector represents a richer feature

representation that incorporates both the historical patterns of individual stocks and their relationships within the market graph.

**Dense Layers:** One or more layers. Dense (fully connected) layers are added after the concatenation. These layers take the combined feature vector and perform non-linear transformations to learn complex interactions between the temporal and relational features.

**Output Layer:** The final layer is typically one of several layers. Dense layer with a single unit (for predicting a single value like the next day's closing price) and a linear activation function (for regression).



**Figure 1.** Architecture of hybrid model.

A brief summary of our mathematical formulation of the proposed hybrid model starts with processing input data which has graph-structured spatial and sequential temporal dimensions.

In this formulation the core components are a custom Graph Convolutional layer followed by an LSTM layer. The Graph Convolutional Layer updates the node representations by aggregating information from their local neighborhoods and combining it with their own current representations. This process is applied independently at each time step.

Let  $X_v^{(t)} \in \mathbb{R}^{D_{in}}$  be the feature vector of node  $v$  at time step  $t$ .

The layer utilizes a trainable weight matrix  $W_g \in \mathbb{R}^{D_{in} \times D_{out}}$ . Let  $\mathcal{N}(v)$  denote the set of source nodes that have edges pointing to node  $v$ .

The operations within the Graph Convolution Layer are in equation 1.

$$X'_{v,(t)} = X_v^{(t)} W_g \quad (1)$$

where  $X'_{v,(t)} \in \mathbb{R}^{D_{out}}$ .

Next, for each node the features from its neighboring nodes  $u \in \mathcal{N}(v)$  are aggregated.

The raw features of the neighbors  $X_u^{(t)}$  are used for this aggregation in equation 2.

$$A_{v,(t)} = AGG_{u \in \mathcal{N}(v)}(X_u^{(t)}) \quad (2)$$

where  $AGG$  can be one of sum, mean or maximum and  $A_{v,(t)} \in \mathbb{R}^{D_{in}}$ .

The aggregated neighborhood information  $A_{v,(t)}$  is then linearly transformed using the same weight matrix  $W_g$  in equation 3.

$$M_{v,(t)} = A_{v,(t)} W_g \quad (3)$$

So,  $M_{v,(t)} \in \mathbb{R}^{D_{out}}$ .

Finally, the new representation of the node  $H_v^{(t)}$  is computed by combining its transformed self-representation  $X'_{v,(t)}$  with the transformed aggregated messages  $M_{v,(t)}$ , followed by an activation function  $\sigma$ . At the end the two representations are concatenated in equation 4 and equation 5.

$$H_v^t = \sigma(\text{concat}(X'_{v,(t)}, M_{v,(t)})) \quad (4)$$

$$\sigma(\text{concat}(X_v^{(t)} W_g, (AGG_{u \in \mathcal{N}(v)}(X_u^{(t)})) W_g)) \quad (5)$$

The resulting node representation  $H_v^{(t)}$  has a dimension of  $2 \times D_{out}$ . Then these output sequences are fed into an LSTM layer to make predictions.

The performance of our model is observed by summing the squares of the difference between the expected output value ( $x_i$ ) of the network and the obtained output value ( $y_i$ ). This method is called Mean Squared Error (MSE) and is defined by the formula in equation 6.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2 \quad (6)$$

The main goal of MSE is to minimize the sum of the differences between the expected value and the obtained output values.

### 3. Results

Our experimental results demonstrate that the hybrid GCN-LSTM model outperforms standalone GCN model in terms of prediction accuracy. The hybrid model effectively captures both the relational and temporal dynamics of the stock market, leading to more accurate predictions.

We further analyze the performance of the models by incorporating additional features such as the dollar and gold parity. The inclusion of these external factors leads to improved accuracy for certain stocks, particularly those with direct correlations to these indicators.

The final evaluation results are summarized in Tables 1 and 2, presenting stocks with low and high MSE values, respectively.

#### 3.1. Analysis of GCN Results

In our results for GCN model, stocks such as YKBNK (MSE: 16.78), AKBNK (MSE: 13.61), and PETKM (MSE: 22.57) show relatively low MSE values, indicating good predictive performance. Some stocks, like BIMAS (MSE: 572.21), THYAO (MSE: 848.28), and SAHOL (MSE: 711.8), exhibit significantly higher MSE values, indicating the model struggles with abrupt price changes. GCNs appear to excel in cases where stock price movements are relatively smooth, leveraging temporal dependencies effectively. In stocks with high volatility (e.g., THYAO, BIMAS), the model struggles to align with the actual prices.

#### 3.2. Analysis of Hybrid GCN-LSTM Results

Our hybrid GCN-LSTM model shows that stocks like ODAS (MSE = 0.94) and SASA (MSE = 0.14) have low MSE values, suggesting high prediction accuracy.

In contrast, stocks like FROTO (MSE = 675.68) and ARCLK (MSE = 243.0) have significantly higher MSE values, indicating poor prediction performance for these stocks.

For many stocks (e.g., AKBNK, KOZAA, TCELL), the predicted trend aligns reasonably well with the actual trend, even though there are some deviations in amplitude or specific points.

In some cases, such as FROTO and THYAO, the model struggles to follow the actual trend, leading to large errors.

Stocks with highly volatile or sharp price changes (e.g., ARCLK, THYAO) seem to be more challenging for the model to predict.

On the other hand, stocks with smoother trends (e.g., ODAS, SASA) are modeled more accurately.

In some cases, the predicted values consistently overestimate or underestimate the actual prices, suggesting potential bias in the model. AKSEN shows persistent underestimation. PETKM predictions closely follow actual values but with slight overestimation in some areas.

The performance appears to vary across different stocks, likely due to differences in their underlying data patterns, trading volumes or other external factors.

Stocks with consistent historical patterns (e.g., ODAS, SASA) are better suited for this GCN-LSTM approach.

For stocks with lower MSE, the predictions closely track the actual values, providing confidence in the model's ability for smoother trends.

Our study highlights the potential of hybrid deep learning models for stock price prediction. By combining the strengths of GCN and LSTM, we can achieve a more comprehensive understanding of the dynamics of the stock market and generate more accurate predictions. The inclusion of external factors such as the dollar and gold parity can further enhance prediction accuracy, particularly for stocks with direct correlations to these indicators.

### 3.3. Analysis of Hybrid Model with External Features

The inclusion of macroeconomic features such as USD/TRY and gold parity in the hybrid GCN-LSTM model resulted in targeted accuracy improvements, particularly for commodity-linked and export-oriented stocks like SASA (MSE: 0.08) and KOZAL (MSE: 0.42), where the models effectively leveraged these external signals to align predictions with actual market trends. However, persistent challenges remain for high-volatility stocks like THYAO (MSE: 965.51) and FROTO (MSE: 756.9); their high error rates likely stem from extreme price scales, sensitivity to industry-specific geopolitical events not captured by parities, and potential noise introduced by additional features that increased model complexity without providing sufficient explanatory power. While the hybrid architecture captures relational dependencies and the added features offer valuable context for sectors with direct correlations to global indicators, the systematic underprediction in stocks like AKSEN and

overestimation in PGSUS suggest that further refinements in feature engineering—such as incorporating sentiment analysis or sector-specific news—are essential to mitigate biases and generalize the benefits of macroeconomic indicators across the diverse BIST30 index.

**Table 1.** Stocks with strong predictive performance (Low MSE Values)

Stock	GCN	GCN+LSTM	Dollar+Gold
AKBNK	13.61	4.07	2.81
HEKTS	101.3	0.48	2.18
ISCTR	297.24	0.5	0.45
ODAS	81.67	0.94	0.14
TCELL	143.49	19.65	22.26
SASA	116.44	0.14	0.08
PETKM	22.57	5.22	1.21
KOZAL	69.83	5.38	0.42
YKBANK	16.78	2.61	3.08

**Table 2.** Stocks with high predictive error (High MSE values)

Stock	GCN	GCN+LSTM	Dollar+Gold
ARCLK	479.63	243.0	110.38
AKSEN	106.61	7.51	3.98
BIMAS	572.21	194.49	62.25
FROTO	449.08	675.68	756.9
PGSUS	35.76	293.66	344.32
SAHOL	711.8	46.01	56.88
THYAO	848.28	708.08	965.51

## 4. Discussion

Our findings illuminate both the inherent complexities of stock price prediction and the nuanced performance variations across different deep learning models. While the hybrid GCN-LSTM model consistently demonstrated superior overall predictive capabilities compared to standalone GCN and LSTM architectures, its effectiveness was not uniform across all stocks. This variability suggests that the optimal modeling approach may be highly dependent on the specific characteristics of the stock under analysis.

A significant challenge observed was the persistent difficulty in accurately predicting highly volatile stocks, exemplified by FROTO and THYAO, even with the inclusion of external factors like dollar and gold parity. This suggests that extreme volatility may introduce a level of noise or intricate complexities that current models struggle to fully capture. Future research could investigate advanced techniques specifically designed to mitigate the impact of volatility or explore additional features that might offer deeper insights into the behavior of such turbulent assets.

Conversely, the inclusion of dollar and gold parity as features yielded notable improvements in prediction accuracy for certain stocks, particularly those with direct

correlations to these indicators, such as KOZAL and SASA. This underscores the potential benefits of integrating relevant external macroeconomic factors into predictive models. However, it also highlights the critical importance of judicious feature selection, as the inclusion of irrelevant or noisy features can conversely detract from model performance. The observed stock specific performance variations further emphasize the necessity of tailoring prediction model selection and design to individual stock characteristics. Factors such as trading volume, industry sector, and unique historical patterns likely influence a model's suitability and predictive power. Moreover, some models exhibited discernible biases, consistently overestimating or underestimating prices for particular stocks, indicating specific areas for future model refinement and bias mitigation strategies. Modeling and predicting future stock prices remain pivotal yet inherently challenging tasks within financial research. Accurate forecasts are fundamental to robust risk analysis and informed investment decision-making in stock trading, making this a field of enduring interest. Recent studies, by demonstrating prediction accuracies surpassing 50 percent, increasingly challenge the Efficient Market Hypothesis (EMH), which traditionally posits that stock prices are unpredictable.

This evolving landscape prompts crucial questions regarding the maximal accuracy achievable by state-of-the-art models and the most effective strategies for integrating diverse predictive models with various data types. Although contemporary surveys reveal a proliferation of innovative methods for stock price prediction, there remains a notable gap in approaches that successfully combine and integrate these techniques into a cohesive composite system from a statistical perspective. Addressing these gaps has the potential to significantly improve predictive performance and contribute substantively to the field. Both LSTM networks and GCNs are powerful tools for analyzing historical stock market data, yet they focus on distinct characteristics of time series. LSTMs excel at capturing temporal dependencies within sequential data, effectively remembering information over long periods. Conversely, GCNs are adept at modeling spatial dependencies and intricate relationships among multiple time series, leveraging the inherent graph structure of financial assets. While LSTMs might be more efficient for purely sequential data without explicit inter-series connections, they face limitations in capturing complex dependencies across multiple sources of sequential information. Given these complementary strengths LSTM's ability to model the time-dependent behavior of a single stock and GCN's capacity to represent relationships between different stocks or economic indicators that we propose a method that synergistically combines LSTM and GCN for the prediction of the Turkish stock market. The complex interrelations among stock prices make graph-based models particularly useful for exploring these dependencies.

### Author Contributions

The percentages of the authors' contributions are presented below. All authors reviewed and approved the final version of the manuscript.

	S.Ö.	A.E.Y.
C	50	50
DCP	100	0
DAI	80	20
L	100	0
W	100	0
CR	0	100
SR	100	0

C=Concept, DCP= data collection and/or processing, DAI= data analysis and/or interpretation, L= literature search, W= writing, CR= critical review, SR= submission and revision.

### Conflict of Interest

The authors declared that there is no conflict of interest.

### Ethical Consideration

Ethics committee approval was not required for this study because of there was no study on animals or humans.

### Acknowledgements

This work is derived from the Ph.D. thesis of the first author, Sevim Ölmez, who gratefully acknowledges the financial support provided by TÜBİTAK under the 2211-National PhD Scholarship Program.

### References

- Berg, R., Kipf, T. N. & Welling, M. (2017). *Graph convolutional matrix completion.* arXiv. <https://doi.org/10.48550/arXiv.1706.02263>
- Cheng, D., Yang, F., Xiang, S. & Liu, J. (2022). Financial time series forecasting with multi-modality graph neural network. *Pattern Recognition*, 121, Makale 108218. <https://doi.org/10.1016/j.patcog.2021.108218>
- Fjellström, C. (2022). Long short-term memory neural network for financial time series. *2022 IEEE International Conference on Big Data (Big Data)* içinde (pp. 3496-3504). IEEE. <https://doi.org/10.1109/BigData55660.2022.10020443>
- Jarrah, M. & Derbali, M. (2023). Predicting Saudi stock market index by using multivariate time series based on deep learning. *Applied Sciences*, 13(14), Makale 8356. <https://doi.org/10.3390/app13148356>
- Kontopoulou, V. I., Panagopoulos, A. D., Kakkos, I. & Matsopoulos, G. K. (2023). A review of ARIMA vs. machine learning approaches for time series forecasting in data driven networks. *Future Internet*, 15(8), Makale 255. <https://doi.org/10.3390/fi15080255>
- Lin, S., Lin, W., Wu, W., Zhao, F., Mo, R. & Zhang, H. (2023). *SegRNN: Segment recurrent neural network for long-term time series forecasting.* arXiv. <https://doi.org/10.48550/arXiv.2308.11200>
- Mehtab, S., Sen, J. & Dutta, A. (2020). Stock price prediction using machine learning and LSTM-based deep learning models. *Symposium on Machine Learning and Metaheuristics Algorithms, and Applications* içinde (ss. 88-106). Springer.

- [https://doi.org/10.1007/978-981-16-0419-5\\_8](https://doi.org/10.1007/978-981-16-0419-5_8)
- Ran, X., Shan, Z., Fan, Y. & Gao, L. (2024). A model based on LSTM and graph convolutional network for stock trend prediction. *PeerJ Computer Science*, 10, Makale e2326. <https://doi.org/10.7717/peerj-cs.2326>
- Schaffer, A. L., Dobbins, T. A. & 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), 1-12. <https://doi.org/10.1186/s12874-021-01235-8>
- Shi, Y., Wang, Y., Qu, Y. & Chen, Z. (2024). Integrated GCN-LSTM stock prices movement prediction based on knowledge-incorporated graphs construction. *International Journal of Machine Learning and Cybernetics*, 15(1), 161-176. <https://doi.org/10.1007/s13042-023-01893-6>
- Singh, K. N., Sharma, K., Avinash, G., Kumar, R. R., Ray, M., Ramasubramanian, V. & Lal, S. B. (2023). LSTM based stacked autoencoder approach for time series forecasting. *Journal of the Indian Society of Agricultural Statistics*, 77(1), 71-78.
- Sonani, M. S., Badii, A. & Moin, A. (2025). *Stock price prediction using a hybrid LSTM-GNN model: Integrating time-series and graph-based analysis.* arXiv. <https://doi.org/10.48550/arXiv.2502.15813>
- Tang, Q., Shi, R., Fan, T., Ma, Y. & Huang, J. (2021). Prediction of financial time series based on LSTM using wavelet transform and singular spectrum analysis. *Mathematical Problems in Engineering*, 2021, 9942410. <https://doi.org/10.1155/2021/9942410>
- Van Houdt, G., Mosquera, C. & Nápoles, G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review*, 53(8), 5929-5955. <https://doi.org/10.1007/s10462-020-09838-1>
- Veličković, P. (2023). Everything is connected: Graph neural networks. *Current Opinion in Structural Biology*, 79, Makale 102538. <https://doi.org/10.1016/j.sbi.2023.102538>
- Yeung, J., Wei, Z., Chan, K. Y., Lau, H. & Yiu, K. F. (2020). Jump detection in financial time series using machine learning algorithms. *Soft Computing*, 24, 14811-14829. <https://doi.org/10.1007/s00500-020-04835-2>
- Zhang, W., Chen, Z., Miao, J. & Liu, X. (2022). Research on graph neural network in stock market. *Procedia Computer Science*, 214, 786-792. <https://doi.org/10.1016/j.procs.2022.11.242>