A Short Survey on Graph Neural Networks Based Stock Market Prediction Models

Lei Zhou
Department of Computer Science and Software Engineering
Auckland University of Technology
Auckland, New Zealand
lei.zhou@autuni.ac.nz

Yuqi Zhang
Department of Computer Science and Software Engineering
Auckland University of Technology
Auckland, New Zealand
yuqi.zhang@autuni.ac.nz

Jian Yu
Department of Computer Science and Software Engineering
Auckland University of Technology
Auckland, New Zealand
jian.yu@aut.ac.nz

Sira Yongchareon
Department of Computer Science and Software Engineering
Auckland University of Technology
Auckland, New Zealand
sira.yongchareon@aut.ac.nz

Samaneh Madanian
Department of Computer Science and Software Engineering
Auckland University of Technology
Auckland, New Zealand
sam.madanian@aut.ac.nz

Guiling Wang
School of Information Science and Technology
North China University of Technology
Beijing, China
wangguiling@ncut.edu.cn

***Abstract*—Stock market predictions are challenging due to the complexity and volatility of markets across the globe. Influential factors include, but are not limited to, economic conditions, political events, investor sentiment, and even natural calamities. In this survey, we review the current literature on stock market predictions using various approaches and propose a framework that facilitates the categorization and analysis of existing works. A novel taxonomy is also proposed within this framework for Graph Neural Network (GNN)-based stock market prediction methods. Potential research gaps are identified, and future research directions are discussed towards the end of this survey.**

***Index Terms*—Graph neural networks, stock prediction, deep learning.**

I. INTRODUCTION

The stock market's complexity and dynamic nature necessitate accurate, interpretable prediction models. Traditional forecasts using time series models such as ARIMA [1] and GARCH [2] face limitations due to the market's nonlinear evolution.

Deep learning, applicable across fields including Natural Language Processing(NLP) and financial forecasting, surpasses traditional methods by handling diverse data types and capturing non-linear stock relationships. Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory(LSTM) [3] and Gated Recurrent Unit(GRU) [4], excel in time series but struggle with inter-stock dynamics. Graph Convolutional Networks (GCN) [5] enable relational reasoning, improving performance by incorporating stock relationships.

To use Graph Neural Networks (GNN) [6] for stock prediction, one can represent stocks as nodes to facilitate node classification tasks. This leverages the stock relationships to enhance prediction models. Chen et al. [7] demonstrated GNN's potential in stock market representation and prediction.

Despite extensive surveys on machine learning and deep learning in stock prediction [8]–[15], a gap exists in dedicated GNN applications. This survey addresses graph-based deep learning advancements in stock prediction, proposing a GNN classification framework, a novel taxonomy, identifying research gaps, and suggesting further directions.

This paper is organized as follows: Section II introduces prediction methods and our framework. Section III reviews GNN applications and our taxonomy. Section IV discusses prior research, and Section V concludes.

II. CLASSIFICATION FRAMEWORK

In this section, we propose a novel classification framework that analyses existing approaches from three aspects, including *model architecture, dataset feature and graph construction*. Figure 1 illustrates the proposed taxonomy based on the classification framework.

A. Model Architecture.

We have identified three types of model architectures that can represent most existing literature: (1) RNN-GNN architecture, (2) iterative RNN-GNN architecture, and (3) parallel RNN-GNN architecture. Figure 2 shows the structure of the three architectures. The RNN-GNN architecture is the most commonly used for graph-based stock market prediction. Stock time series data are first fed into a Recurrent Neural

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-147-3

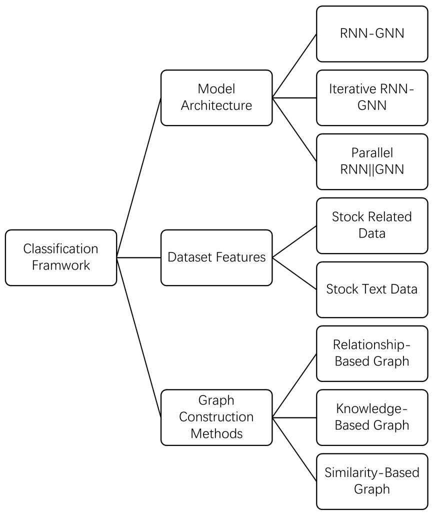

Fig. 1: Classification Framework.

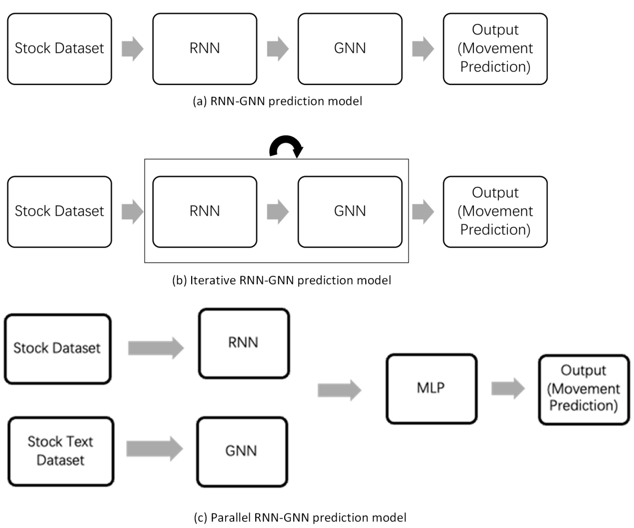

Fig. 2: Common architecture using GNN for stock prediction.

Network, such as LSTM [3], GRU [4], or Bi-directional [16], to extract features. Then, the hidden features are processed by a Graph Neural Network, such as GCN [5], Graph Attention Network(GAT) [17], or GraphSAGE [18], for node classification. The other two architectures are less common in the literature.

As shown in Figure 2 (b), the iterative RNN-GNN architecture achieves deeper integration and collaboration between the RNN and GNN, including an information exchange mechanism, resulting in better capture of the complex relationships between temporal and relational data.

The parallel RNN || GNN architecture, shown in Figure 2 (c), effectively captures both temporal and relational information in complex datasets. This integration facilitates a more comprehensive understanding of the data, leading to improved performance in tasks such as sequence modeling, time series forecasting, and relational reasoning.

Compared with the other two architectures, the RNN-GNN model is simpler and easier to implement, as it uses the output of the RNN directly as input to the GNN. However, it may underperform in some complex tasks. The choice of architecture depends on task-specific requirements and data characteristics.

B. Datasets Features

Stock price fluctuations depend on numeric and text data, including prices and volumes [19], as well as announcements and social media [20]. Using historical prices helps predict future trends, with numeric data being more standardized and accessible. Text data, requiring sentiment analysis for integration [21], contributes to forecasting but cannot solely predict prices. Combining both data types enhances prediction accuracy.

C. Graph Construction Method

In literature on GNN for stock market prediction, three primary methods for graph construction are identified: correlation-based, knowledge-based, and similarity-based graph constructions.

Relationship-based graph construction [7] constructs graphs from internal stock relationships, utilizing key market indicators such as prices and volumes to create nodes for each stock. Historical data analysis, through correlation coefficients or time series models, determines edges reflecting stock correlations.

Knowledge-based graph construction [22] [23] leverages external information, such as sector dependencies or economic impacts, to enhance graph structure. This approach integrates expert insights or industry reports, capturing relationships beyond dataset information.

Similarity-based graph construction [24] builds graphs by identifying similarities among stocks, useful for implicit relationship mapping. Nodes represent stocks, with edges based on similarity scores (e.g., cosine similarity or Euclidean distance), highlighting clusters of similar stocks.

III. Approaches

In this section, we review existing approaches that use GNNs for the stock market. We discuss each approach from three aspects: architecture, dataset features, and graph construction methods, as proposed in the previous section.

A. Model Architecture

1. RNN-GNN architecture

Most models adopt the RNN-GNN architecture, embedding time series data for graph-based prediction. Combining RNNs' ability to capture temporal sequences with GNNs' insight into stock interrelations, data is encoded using LSTM or GRU for pattern recognition and long-term dependencies.

Once the time series data is embedded, it is fed into the GNN component. The GNN utilizes graph convolutional layers

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-147-3

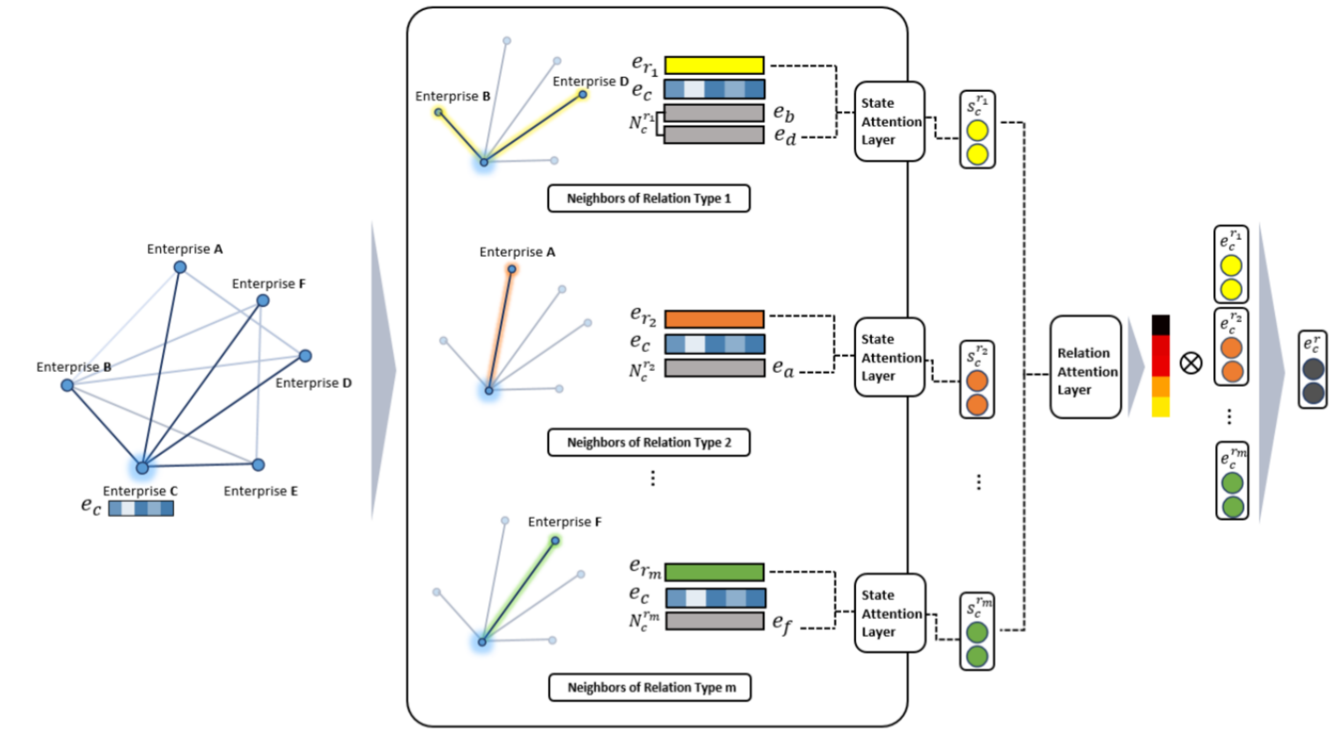

Fig. 3: Hierarchical Attention Network for Stock Prediction.

to propagate information among the interconnected stocks in a graph structure. By considering the correlations and dependencies between stocks, the GNN can capture collective behaviour and market dynamics.

In [7], Chen et al. propose a joint RNN-GNN model called the Incorporating Corporation Relationship-Graph Convolutional Neural Network (ICR-GCN), which employs the RNN-GNN architecture. The model is composed of two parts. The first part is an LSTM that encodes time series information to extract features for each company. These features then serve as the node attributes in a graph that represents the relationships between companies. Subsequently, a three-layer GCN is applied for node classification.

$$Y = \mathrm{softmax}\left(\widehat{A}ReLU\left(\widehat{A}ReLU\left(\widehat{A}X'W\right)W\right)W\right)$$

where $\widehat{A}$ is the adjacency matrix, X' represents the historical features, W is the learnable weight matrix. $ReLU(\cdot)$ and $softmax$ functions are used as the activation functions, and cross-entropy is used as the loss function.

Compared to similar studies, a significant advancement in [25] is its consideration of social media text's impact on stock prices. By integrating social media text with financial data and stock relationships, it introduces additional dimensions of signals for stock prediction.

In [26], Kim et al. propose a Hierarchical Attention Network for Stock prediction (HATS) that uses relational data for stock market prediction. It selectively aggregates information on different relation types and adds the information to the representations of each company. As shown in Figure 3. HATS is the RNN-GNN model; it has three layers including the Feature Extraction layer, the Relational Modeling layer and the Prediction Layer. In the feature extraction layer, one of LSTM and GRU is used to encode features.

The Relational Modeling layer is used to encode the graph structure. The HATS layout is shown in Figure 3. e_{r_m} is the relation type, e_n is the feature of node n, and $N_i^{r_m}$ is the set of neighboring nodes of i for relation type m.

The prediction layer focuses on individual stock and S&P500 Index movements, classifying stocks into three categories: up, down, and neutral, using a linear transformation for individual predictions. Mean pooling calculates the index's graph representation. HATS, similar to ICR-GCN, employs RNN-GNN architecture but differs by using GAT, whereas ICR-GCN utilizes GCN.

HATS selectively aggregates information from different types of relationships and adds the information to the representation of each company.

In Feng et al.'s study [27], the Relational Stock Ranking (RSR) framework employs the Temporal Graph Convolution (TGC) model for stock prediction, featuring a three-layered joint RNN-GNN model: a sequential embedding layer with LSTM for capturing stock sequences, a relational embedding layer using TGC for stock interconnections, and a ranking scores prediction layer for stock ranking. Different from other models, RSR-TGC uniquely captures temporal dynamics with

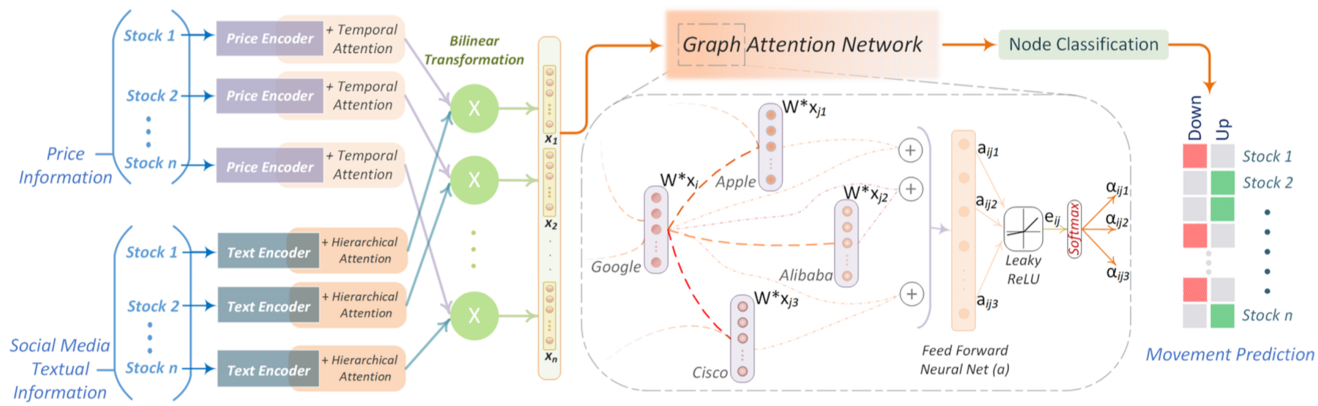

Fig. 4: MAN-SF Model.

TGC, distinguishing it from GCN and GAT models.

This approach contrasts with Wang et al. [28], which uses a hierarchical model to analyze temporal stock relationships and market dynamics across multiple timescales, considering financial, social media sentiment, and other factors.

In [25], Sawhney et al. propose a Multipronged Attention Network for Stock Forecasting (MAN-SF) by learning from historical prices, social media, and inter-stock relations. It is made up of a hierarchical attention network and a GAT. The Hierarchical Attention Network (HAN) is responsible for capturing relevant signals across diverse data, while the GAT is responsible for predicting stock movements.

MAN-SF model is a joint RNN-GNN architecture. As shown in Figure 4, first, GRU is used as a Price Encoder (PE) that takes the prices of a stock over a period of time and uses that to produce a price feature. The temporal attention mechanism is a way of aggregating information from different time steps into an overall representation. This is done by assigning learned weights to each time step, which allows the most important information to be aggregated together. For example, the formula of temporal attention mechanism $\zeta(\cdot)$ is shown as:

$$\zeta\left(\bar{h}_z\right) = \sum_i \beta_i h_i \tag{1}$$

$$\beta_i = \frac{\exp\left(h_i^T W \bar{h}_z\right)}{\sum_{i=1}^{T} \exp\left(h_i^T W \bar{h}_z\right)} \tag{2}$$

where $\bar{h}_z$ is the hidden states of GRU, β_i is the attention weight and W is the learnable parameter matrix.

Secondly, the Social Media Information Encoder (SMI) employs GRU to distill tweet data, using a hierarchical attention mechanism to encode this information into vectors. Thirdly, the Blending Multimodal Information layer merges features from PE and SMI, applying a bilinear transformation for learning price-tweet interactions, optimizing the mix of data inputs. Lastly, stock movement prediction is performed using a GAT.

Similar to the above models, MAN-SF also uses the RNN-GNN architecture pattern; the main difference is that three attention mechanisms are used to extract features, which include price data, news data and stock relations data by temporal attention, hierarchical attention and graph attention.

2. Iterative RNN-GNN architecture

In [24], Li et al. propose an LSTM Relational Graph Convolutional Network (LSTM-RGCN) model that predicts the overnight stock movement based on the correlation between stocks. This paper constructs a graph by converting each stock's news into a vector and calculating the relationship between each stock by using the cosine similarity. LSTM-RGCN is the first model that unitized connection among stocks to predict the movement of stocks that are not directly associated with news.

The LSTM-RGCN model, depicted in Figure 2(b), embodies an iterative RNN-GNN architecture. As illustrated in Figure 5, the process begins with LSTM encoding news data into vectors. Subsequently, the model merges the news vector with the node embedding to form the node vector. Finally, RGCN encodes the graph structure.:

$$N^{l+1} = \sigma\left(\sum_r D_r^{-\frac{1}{2}} A_r D_r^{-\frac{1}{2}} H^l W_r^l + W_h H^l\right) \tag{3}$$

where A_r is the adjacency matrix of relation r, $D_r^{-\frac{1}{2}} A_r D_r^{-\frac{1}{2}}$ is the normalized adjacency matrix. W_r^l is the learnable parameter matrix. W_h is the learnable parameter matrix for the node vector. The parameter matrices are shared across layers. H^l represents the hidden representations of all the nodes in the l-th layer. N^{l+1} is the aggregated neighbor information for the $(l+1)$-th layer.

Finally, the model predicts stock movement based on the node representation in the graph. $Sigmoid(\cdot)$ and $softmax$ functions are used as the activation functions. The cross-entropy is used as a loss function for this two-class classification task.

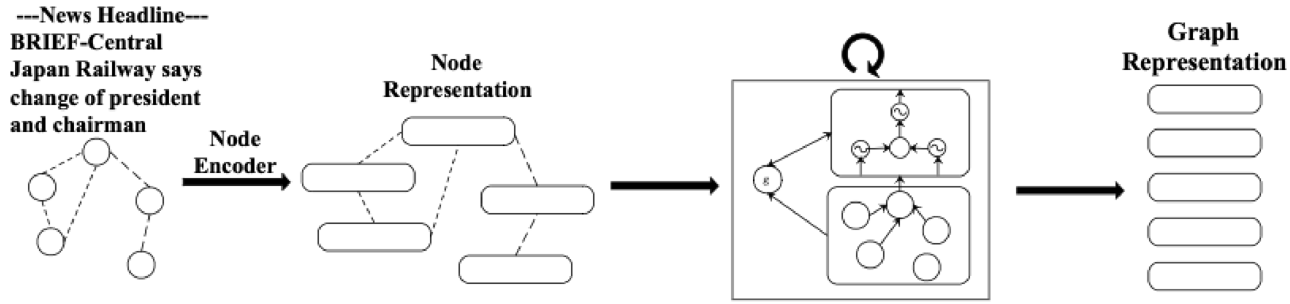

Fig. 5: LSTM-RGCN Model.

3. Parallel RNN || GNN architecture

In their paper [23], Zhao et al. proposed a method called Dual Attention Networks to learn Stock Movement Prediction (DANSMP). This method leverages a market knowledge graph to model the relationships between stocks and make predictions about stock momentum. The graph comprises various types of information, including the relationships between companies and their executives.

DANSMP integrates three layers: stock sequential embedding, stock relational embedding, and prediction.

Initially, it merges technical and sentiment features using GRU for feature extraction. The relational layer uses dual attention networks for spillover signal representation, focusing on company-executive relationships via inter- and intra-class networks. Inter-class networks compare company and executive features, while intra-class networks assess same-type entity interactions.

Finally, embeddings are combined in a neural network for stock movement prediction. DANSMP's innovation lies in its parallel RNN-GNN structure and dual attention mechanism, enhancing market relationship analysis.

B. Dataset Feature

1. Data Information

The ICR-GCN dataset, sourced from Tushare API [29], comprises CSI 300 historical prices for listed companies from 29/04/2017 to 31/12/2017. It features five numeric attributes per company: open, close, high, low, and volume, utilizing seven days of historical data for input.

HATS dataset uses S&P 500 historical price data of listed companies between 08/02/2013 and 17/06/2019 (in total, 1174 trading days). For each company, there are three numeric features including open price, close price, and volume. The authors use historical price change rate $\boldsymbol{R}_i^t = \frac{(P_i^t - P_i^{t-1})}{P_i^{t-1}}$ as model input, where P_i^t is the closing price at time t of a company i and P_i^{t-1} is the closing price at time $t-1$.

In the RSR-TGC model, data collection comprises three categories, starting with sequential price data from New York Stock Exchange (NYSE) and NASDAQ between February 1, 2013, and December 8, 2017. This dataset encompasses open, close, high, low, and volume metrics, using twenty-nine daily close prices for calculating (Moving Average) MA5, MA10, MA20, and MA30. These indicators, combined with the closing price, undergo normalization for model input.

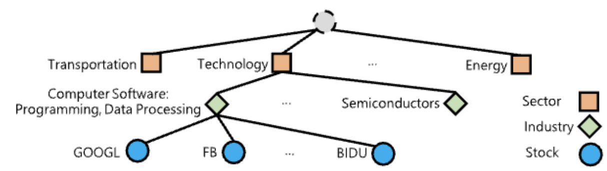

Fig. 6: Sector-industry relations and wiki company-based relations.

The MAN-SF dataset used in this study was the StockNet dataset [30] which contains data on high-trade-volume stocks in the S&P 500 index in the NYSE and NASDAQ markets. The dataset is split into three parts: training, validation, and testing. The training data was used to train the MAN-SF model.

The DANSMP includes total of 185 stocks from the China Securities Index 300(CSI300E) and 73 stocks from the CSI100E are used to create two datasets which are collected from the China Securities Index (CSI). The market dataset includes historical price information (opening price, close price, highest price, lowest price and volume).

2. Stock Text Information

The second type in RSR-TGC model is sector-industry relations and the third type is Wiki relations such as supplier-consumer relations and ownership relations.

The LSTM-RGCN model also includes Financial news and market data from Tokyo Stock Exchange (TSE) from 1/1/2013 to 29/08/2018 from Reuters. There are two numeric features including open price and close price that are used to calculate the overnight movement. The formula for calculating the overnight movement is shown below:

$$\text{Movement } = \left(p_o^t - p_c^{t-1}\right) / p_c^{t-1} \tag{4}$$

where p_o^t is the open price of the current trading day and p_c^{t-1} is the close price of the previous trading day. Global Vectors for Word Representation (GloVe) [31] and Bidirectional Encoder

Representations from Transformers (BERT) [32] are used as word embedding on financial news to generate model input features.

In DANSMP model also includes the news dataset from four financial mainstream sites [33] [34] [35] [36].

C. Graph Construction Method

1. Correlation-based Graph Construction Methods

The ICR-GCN utilizes a data-driven graph construction method based on financial investment facts, using a graph of companies (stocks). This graph construction approach is designed to capture relevant relationships between companies. In this weighted graph, each node corresponds to a company, while the edges connecting the nodes represent the relationships between these companies. Furthermore, the weight assigned to each edge reflects the shareholding ratio between the connected companies.

2. Knowledge-based Graph Construction Methods

The HATS model constructs a heterogeneous graph from Wikidata to analyze relations between entities such as companies and persons. It simplifies this graph into a company-focused homogeneous one using a meta-path, representing companies as nodes connected by various relationship types, such as 'Owned by'.

The RSR-TGC model constructs its graph using sector-industry information, as illustrated in Figure 6, and company relations sourced from Wikidata. It categorizes stocks by industry and establishes connections between stocks based on first and second-order relations derived from Wikidata [37].

MAN-SF leverages first and second-order relations from Wikidata to map the S&P 500 index stocks' relationships, focusing on direct company connections.

DANSMP employs a Bi-typed Hybrid-relational Market Knowledge Graph (MKG) from Tushare API [29] data, featuring company and executive entities with both explicit (directly stated relationships) and implicit (inferred from attributes) relations. This approach enables a detailed analysis of company and executive interconnections for stock movement predictions, enriching market analysis and investment decisions.

3. Similarity-based Graph Construction Methods

In the LSTM-RGCN model, a stock correlation graph is built using historical prices, distinguishing relationships as positive or negative correlations. Connections between stock nodes are made if their similarity's absolute value meets a set threshold, enabling the model to understand stock interdependencies from historical prices.

IV. Discussion

In this section, we compare prior research, highlighting Table I's summary of model differences, including type, architecture, dataset, and data types. Table I shows the prevalent use of the RNN-GNN model, with three studies employing GCN-based GNNs (ICR-GCN, TGC, RGCN) and three using GAT-based models (HATS, MAN-SF, DANSMP). Dataset analysis reveals MAN-SF and HATS share a dataset; others vary, with sources ranging from Chinese to US and Japanese stock markets. Most datasets encompass price and news data. Graph-wise, DANSMP uniquely utilizes a heterogeneous graph; others use homogeneous ones, mostly constructed from Wikidata, except RGCN's price-based graphs.

We identify two key enhancement areas for stock prediction models:

1. Integrating transformers with GNN-based temporal models could improve accuracy by capturing long-range dependencies.

2. Developing dynamic spatial-temporal graphs with market data for transformer networks might refine models by leveraging complex spatial-temporal relationships.

These approaches suggest promising directions for refining stock market predictions, offering potential benefits for investors and analysts through advanced techniques and comprehensive market data utilization.

V. Conclusion

Stock market prediction is complex due to various impacting factors. This paper reviews six articles applying GNN to stock prediction, analyzing model architectures, data types, and graph construction methods.

Three model architecture patterns were identified, combining RNN and GNN, capturing temporal and relational stock data information. The primary data types were historical price, financial news, and financial indicators.

Graphs were constructed based on stock correlations, similarity measures, or network propagation techniques, capturing stock relationships and dependencies. Most models adopted the RNN-GNN framework, improving performance by leveraging the graph's structural information.

The review underscores the efficacy of GNN in stock prediction and the value of graph-based modelling with traditional sequential models. This paper provides an overview of GNN-based stock prediction, laying the foundation for future research.

References

[1] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the arima model," in *2014 UKSim-AMSS 16th international conference on computer modelling and simulation*. IEEE, 2014, pp. 106–112.

[2] B. M. Awartani and V. Corradi, "Predicting the volatility of the s&p-500 stock index via garch models: the role of asymmetries," vol. 21, no. 1. Elsevier, 2005, pp. 167–183.

[3] S. Hochreiter and J. Schmidhuber, "Long short-term memory," vol. 9, no. 8. MIT Press, 1997, pp. 1735–1780.

[4] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[5] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in *International Conference on Learning Representations*, Feb. 2017. [Online]. Available: http://arxiv.org/abs/1609.02907

[6] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," vol. 20, no. 1. IEEE, 2008, pp. 61–80.

TABLE I: COMPARISON MODEL & METRICS BETWEEN ARTICLES

Framework	Model	Dataset	Graph relationships	AUX DATA
ICR-GCN	LSTM-GCN	CSI300	Stock-Stock	Financial investment fact from WIND
HATS	RNN-GAT	S&P500	Stock-Stock Stock-Owner	Wikidata
TGC	LSTM-TGC	NASDAQ NYSE	Stock-Stock	Wikidata
MAN-SF	GRU-GAT	S&P500	Stock-Stock Stock-Owner	Wikidata
RGCN	LSTM-RGCN	TPX500 TPX100	Stock-Stock	News
DANSMP	GRU-DAN	CSI100E CSI300E	Stock-Stock Stock-Owner Owner-Owner	News

[7] Y. Chen, Z. Wei, and X. Huang, "Incorporating corporation relationship via graph convolutional neural networks for stock price prediction," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018, pp. 1655–1658.

[8] I. K. Nti, A. F. Adekoya, and B. A. Weyori, "A systematic review of fundamental and technical analysis of stock market predictions," *Artificial Intelligence Review*, vol. 53, no. 4, pp. 3007–3057, 2020.

[9] W. Jiang, "Applications of deep learning in stock market prediction: recent progress," *Expert Systems with Applications*, vol. 184, p. 115537, 2021.

[10] I. K. Nti, A. F. Adekoya, B. A. Weyori, and O. Nyarko-Boateng, "Applications of artificial intelligence in engineering and manufacturing: a systematic review," *Journal of Intelligent Manufacturing*, vol. 33, no. 6, pp. 1581–1601, 2022.

[11] A. W. Li and G. S. Bastos, "Stock market forecasting using deep learning and technical analysis: a systematic review," *IEEE access*, vol. 8, pp. 185 232–185 242, 2020.

[12] M. M. Kumbure, C. Lohrmann, P. Luukka, and J. Porras, "Machine learning techniques and data for stock market forecasting: A literature review," *Expert Systems with Applications*, p. 116659, 2022.

[13] T. J. Strader, J. J. Rozycki, T. H. Root, and Y.-H. J. Huang, "Machine learning stock market prediction studies: review and research directions," *Journal of International Technology and Information Management*, vol. 28, no. 4, pp. 63–83, 2020.

[14] M. Nikou, G. Mansourfar, and J. Bagherzadeh, "Stock price prediction using deep learning algorithm and its comparison with machine learning algorithms," *Intelligent Systems in Accounting, Finance and Management*, vol. 26, no. 4, pp. 164–174, 2019.

[15] M. Obthong, N. Tantisantiwong, W. Jeamwatthanachai, and G. Wills, "A survey on machine learning for stock price prediction: Algorithms and techniques," 2020. [Online]. Available: https://eprints.soton.ac.uk/437785/

[16] E. P. Lafortune and Y. D. Willems, "Bi-directional path tracing," 1993. [Online]. Available: https://masters.donntu.ru/2013/fknt/kalamitra/library/Bidirectional%20Path%20Tracing%201990.pdf

[17] P. Casanova, A. R. P. Lio, and Y. Bengio, "Graph attention networks," *ICLR. Petar Velickovic Guillem Cucurull Arantxa Casanova Adriana Romero Pietro Liò and Yoshua Bengio*, 2018.

[18] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[19] M. Vijh, D. Chandola, V. A. Tikkiwal, and A. Kumar, "Stock closing price prediction using machine learning techniques," *Procedia computer science*, vol. 167, pp. 599–606, 2020.

[20] F. Li *et al.*, "Textual analysis of corporate disclosures: A survey of the literature," *Journal of accounting literature*, vol. 29, no. 1, pp. 143–165, 2010.

[21] J. Li and P. Meesad, "Combining sentiment analysis with socialization bias in social networks for stock market trend prediction," *International Journal of Computational Intelligence and Applications*, vol. 15, no. 01, p. 1650003, 2016.

[22] Y. Liu, Q. Zeng, J. Ordieres Meré, and H. Yang, "Anticipating stock market of the renowned companies: a knowledge graph approach," *Complexity*, vol. 2019, 2019.

[23] Y. Zhao *et al.*, "Stock movement prediction based on bi-typed and hybrid-relational market knowledge graph via dual attention networks," *arXiv preprint arXiv:2201.04965*, 2022.

[24] W. Li, R. Bao, K. Harimoto, D. Chen, J. Xu, and Q. Su, "Modeling the stock relation with graph network for overnight stock movement prediction." in *IJCAI*, vol. 20, 2020, pp. 4541–4547.

[25] R. Sawhney, S. Agarwal, A. Wadhwa, and R. Shah, "Deep attentive learning for stock movement prediction from social media text and company correlations," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 8415–8426.

[26] R. Kim, C. H. So, M. Jeong, S. Lee, J. Kim, and J. Kang, "Hats: A hierarchical graph attention network for stock movement prediction," *arXiv preprint arXiv:1908.07999*, 2019.

[27] F. Feng, X. He, X. Wang, C. Luo, Y. Liu, and T.-S. Chua, "Temporal relational ranking for stock prediction," vol. 37, no. 2. ACM New York, NY, USA, 2019, pp. 1–30.

[28] H. Wang, S. Li, T. Wang, and J. Zheng, "Hierarchical adaptive temporal-relational modeling for stock trend prediction." in *IJCAI*, 2021, pp. 3691–3698.

[29] "Tushare." [Online]. Available: http://tushare.org/

[30] Y. Xu and S. B. Cohen, "Stock movement prediction from tweets and historical prices," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 1970–1979.

[31] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[32] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[33] "Wind." [Online]. Available: http://www.wind.com.cn/

[34] "Sina." [Online]. Available: http://www.sina.com/

[35] "Hexun." [Online]. Available: http://www.hexun.com/

[36] "Sohu." [Online]. Available: http://www.sohu.com/

[37] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," vol. 57, no. 10. ACM New York, NY, USA, 2014, pp. 78–85.