ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230476

Stock Movement Prediction via Attention-Aware Multi-Order Relation Graph Neural Network

Hao Peng[a;*] and Jie Yang[b]

[a]School of Computing, Newcastle University, Newcastle, UK
hao.peng97@outlook.com
[b]Research Institute of Information Technology, Tsinghua University, Beijing, China
totoroyang@tsinghua.edu.cn

Abstract. Stock Movement Prediction (SMP) is a challenging task that aims at predicting the future stock price trend of companies in the stock. Recent advances mainly apply the Graph Convolutional Network (GCN) to learn connections among companies for SMP. However, these methods usually ignore the semantics of the specific relations (e.g., investment and share) between two entities (i.e., companies and persons) on the market knowledge graph. Meanwhile, considering the long-chain cross-shareholding structures among entities, it is difficult for GCN to obtain high-order neighbor information over long distances. To address these two problems, we present an Attention-aware Multi-order Relation GCN for SMP (AMRGCN-SMP). Specifically, an attention-aware multi-channel aggregation manner achieves the weighted fusion of nodes across multiple semantic channels. Moreover, the dynamic update of the adjacent tensor can fuse the multi-order relation representations and bring more abundant long-chain connections. The experiments on the CSI100E and CSI300E datasets demonstrate that the proposed method achieves state-of-the-art performances compared with the recent advances.

1 Introduction

Stock Movement Prediction (SMP) is an important task that can help investors to predict the stock price trend of companies in the financial market [1, 2]. With the recent upheavals in the stock market, the SMP task has spurred the interest of researchers to continuously present better predictive models [3]. In particular, the application of the shallow machine learning & deep learning approaches brings about a promising performance for the SMP task [4, 5].

The main efforts for SMP can be divided into two genres including the methods based on the technical analysis and the approaches based on the fundamental analysis [6]. In detail, the technical analysis usually utilizes the time-series historical information of the stock market (e.g., stock price and volume) to construct features and further learn the hidden trading pattern for SMP [7, 8, 9]. Meanwhile, the fundamental analysis usually utilizes the cues derived from the outside market price data (e.g., economic environment and some other qualitative & quantitative factors) to predict the price movement of the stock market. Recently, many SMP methods have been presented to apply the time-series data and the social media information (e.g., news and twitter texts) for SMP [1, 10, 11]. They simultaneously consider the technical analysis and fundamental analysis, which exploit historical stock price information and social media information to predict the stock price movement.

The early works for SMP mainly exploit shallow machine learning to learn implicit trading modes by constructing some hand-craft features, while these methods suffer from expensive time costs and weak generalization [12]. Recently, with the development of deep learning, many approaches exploit some neural networks such as Convolutional Neural Networks (CNN) [13], Recurrent Neural Networks (RNN) [14] and Transformers [15] for SMP, as they can automatically extract stock features. However, these methods usually only consider the time-series data whereas they ignore the connections among the public companies such as *cross-shareholding* and *industry specialization* [6]. The lack of correlations for companies may lead to limited performances for SMP. To alleviate this issue, Graph Convolutional Network (GCN) has been noticed and widely applied for the SMP task, which achieves some competitive performances [5, 16, 17, 18].

However, the recent GCN-based methods for SMP usually directly compute the weighted sum of neighbor node embeddings during aggregation. They neglect the relational semantics between two nodes. In general, because of the existence of the control relationship, the company may be affected by some roles (i.e., relations) connected to different implicit entities. For instance, the Credit Suisse failure made the Saudi *investment* entity suffer heavy losses. Thus, the specific relations between two entities are necessary when the neighbor node information is aggregated to the targeted company for SMP. In addition, considering the long-chain cross-shareholding structures among entities, it is difficult to obtain the high-order neighbor node information only by stacking more GCN layers, which may affect the performance of SMP. To address this problem, in this work, we introduce an attention-aware semantic aggregation based on the specific relations and multi-order dynamics of adjacent tensor to improve the performance of SMP.

To sum up, the main contributions of this work are as follows:

- To better encode the neighbor information, we apply an attention-aware multi-channel aggregated mechanism across relational semantic dimensions, which can achieve the effective fusion of node information from the different semantic channels.
- Considering the existence of the long-chain cross-shareholding structures among entities, we propose an Attention-aware Multi-order Relation GCN for SMP (AMRGCN-SMP), which can achieve dynamic update of adjacent tensor based on the multi-order relations and improve the performance of SMP.

* Corresponding Author. Email: hao.peng97@outlook.com.

- The experimental results on the CSI100E and CSI300E datasets demonstrate that the proposed AMRGCN-SMP model achieves state-of-the-art performances compared with the recent advances.

2 Related Works

Stock Movement Prediction. Stock movement prediction (SMP) has been always a hot topic in the fintech domain because of its great benefits for investors in the stock market. According to the Efficient Market Hypothesis (ETH) theory, it is not realistic to accurate the stock real-time prices due to the hysteresis quality of stock information [10]. For this reason, some works attempted to indirectly predict the stock price trend using technical analysis and fundamental analysis, which achieve some comparable performances, especially after the application of deep learning techniques. For instance, to reduce the stochasticity of the influences from the news outsides the stock markets, Xu et al. presented a generative model using a market information encoder and a variational movement decoder to respectively encode tweets texts and generate results of SMP [10]. Hu et al. proposed a hierarchical attention network to encode the news texts to aggregate the temporal cues to predict future stock price trends [1]. Feng et al. and Xu et al. respectively apply adversarial training and dual-channel generative adversarial network to enhance the generalization for SMP [19, 20]. Liu et al. utilized a Transformer and a Capsule network to encode the social medical information for the SMP task [21]. Zhang et al. designed a state frequency memory recurrent network based on the Fourier transform to learn the hidden trading pattern obtained from stock price sequence to predict the stock movement [11]. Although these methods achieved impressive performances for SMP, they ignore the explicit and implicit relations among entities and may limit the potential of the existing approaches.

GCNs for Stock Movement Prediction. Recently, compared with the above traditional approaches based on the time-series data, some advances have applied graph neural network to capture non-local relations [22, 23] among public companies on the market knowledge graph to better predict stock movement. For example, by considering the cross effects among companies, Ye et al. presented a Multi-GCGRU framework consisting of a GCN and a Gated Recurrent Unit (GRU) to predict stock movement [5]. Zhao et al. applied a dual-attention mechanism to fuse the time-series features and learn the connections among companies in the stock market [6]. To predict the overnight SMP between the previous close price and the open price, Li et al. proposed an LSTM-RGCN framework to model the complex connections among entities by using their correlation matrix [24]. Although the GCN-based methods can implement more abundant relation interactions among entities and enhance the performances of SMP, these approaches ignore the fact that most GCNs fail to fuse relation semantics between two entities and lack the ability to capture high-order information by stacking limited layers.

3 Methodology

As illustrated in Fig. 1, the proposed SMP method consists of a tensor incorporation module, a temporal learning module, an Attention-aware Multi-order Relation GCN (AMRGCN) module, and a binary classifier module. Noted that the inputs of the proposed model are the historical price representation $\boldsymbol{p}_i \in \mathbb{R}^5$ and the sentiment representation $\boldsymbol{q}_i \in \mathbb{R}^3$ for i-th company in the certain trading day. Also, the Market Knowledge Graph (MKG) consists of the targeted public companies and some corresponding executives (i.e., persons). These above-described data had been concluded and released in [6].

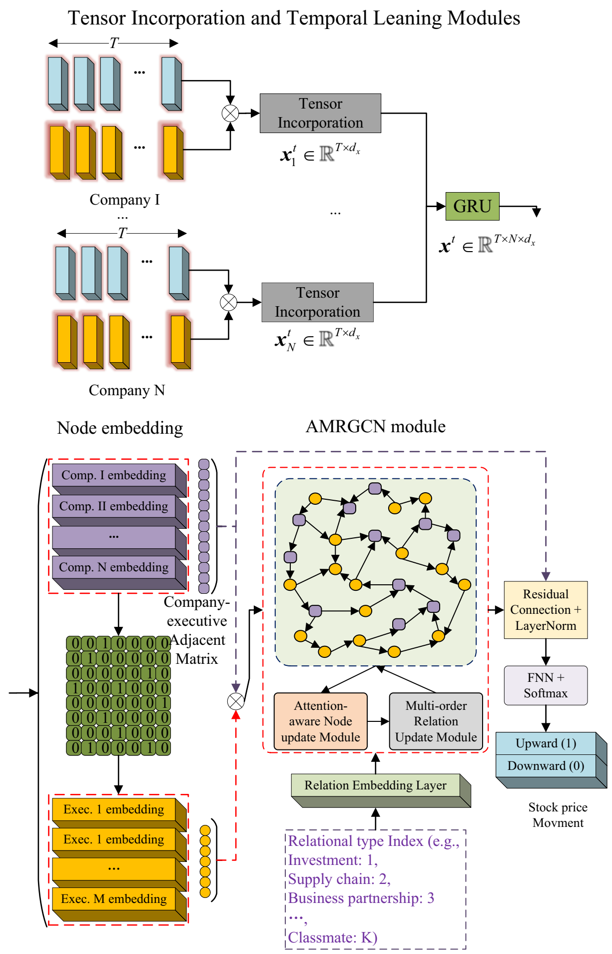

Figure 1. Illustration of the proposed SMP architecture.

3.1 Tensor Incorporation Module

To respectively extract stock price and sentiment features, we first introduce a 1-dimension convolutional operator $\mathrm{Conv1D}(\cdot)$ to obtain the high-level features from the price and sentiment vectors:

$$\begin{aligned} \boldsymbol{x}^p &= \mathrm{Conv1D}(\boldsymbol{p}) \in \mathbb{R}^{N\times 5}, \\ \boldsymbol{x}^q &= \mathrm{Conv1D}(\boldsymbol{q}) \in \mathbb{R}^{N\times 3}, \end{aligned} \tag{1}$$

where $\boldsymbol{p} = [\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_N] \in \mathbb{R}^{N\times 5}$, $\boldsymbol{q} = [\boldsymbol{q}_1, \boldsymbol{q}_2, \cdots, \boldsymbol{q}_N] \in \mathbb{R}^{N\times 3}$, N is the number of companies.

Further, along the representation channel of $\boldsymbol{x}_p$ and $\boldsymbol{x}_q$, we then apply an external attention network to achieve feature interaction of price and sentiment signals across companies:

$$\begin{aligned} \boldsymbol{x}^{p-a}_{:,j} &= \mathrm{ExternalAtten}(\boldsymbol{x}^p_{:,j}, S) \in \mathbb{R}^N, \\ \boldsymbol{x}^{q-a}_{:,j} &= \mathrm{ExternalAtten}(\boldsymbol{x}^q_{:,j}, S) \in \mathbb{R}^N, \end{aligned} \tag{2}$$

where $\mathrm{ExternalAtten}(\cdot)$ denotes the applied external attention operator proposed in [25], S is the hidden size, and $\boldsymbol{x}^{p-a} = [\boldsymbol{x}^{p-a}_{:,1}, \boldsymbol{x}^{p-a}_{:,2}, \cdots, \boldsymbol{x}^{p-a}_{:,5}]^{\mathrm{T}} \in \mathbb{R}^{N\times 5}$ and $\boldsymbol{x}^{q-a} = [\boldsymbol{x}^{q-a}_{:,1}, \boldsymbol{x}^{q-a}_{:,2}, \boldsymbol{x}^{q-a}_{:,3}]^{\mathrm{T}} \in \mathbb{R}^{N\times 3}$ mean the updated company feature tensors respectively split along the dimension of the price and the sentiment. Besides, to avoid the issue of network degradation and accelerate convergence, we also introduce residual connections which are:

$$\begin{aligned} \boldsymbol{x}^{p-a'} &= \mathrm{ReLU}(\boldsymbol{x}^{p-a} + \boldsymbol{x}^{p}) \in \mathbb{R}^{5}, \\ \boldsymbol{x}^{q-a'} &= \mathrm{ReLU}(\boldsymbol{x}^{q-a} + \boldsymbol{x}^{q}) \in \mathbb{R}^{3}, \end{aligned} \tag{3}$$

where $\mathrm{ReLU}(\cdot)$ is an activation function.

Finally, to achieve the fusion of the price and sentiment features, a nonlinear layer is employed to generate the multi-modal representation of the i-th company in a certain trading day:

$$\boldsymbol{x}_i = \mathrm{Tanh}(\boldsymbol{W}_f \cdot \hat{\boldsymbol{x}}_i + \boldsymbol{b}_f) \in \mathbb{R}^{d_x}, \tag{4}$$

where $\hat{\boldsymbol{x}}_i = \boldsymbol{x}_i^p \odot \boldsymbol{x}_i^q \odot \boldsymbol{x}_i^{p-a'} \odot \boldsymbol{x}_i^{q-a'} \in \mathbb{R}^{16}$ is the concatenated representation, $\odot$ denotes a concatenated operator, $\mathrm{Tanh}(\cdot)$ is an activation function, d_x is the dimension of the multi-modal representation, and $\boldsymbol{W}_f$ and $\boldsymbol{b}_f$ are respectively the projected matrix and the bias vector.

3.2 Temporal Learning Module

In general, SMP for i-th company in t-th trading day usually depends on the historical feature tensor $\boldsymbol{x}_i^t = [\boldsymbol{x}_i^{t-T}, \boldsymbol{x}_i^{t-T+1}, \cdots \boldsymbol{x}_i^{t-1}] \in \mathbb{R}^{T \times d_x}$ in the past T trading days. Thus, we can use a unidirectional GRU network to capture the stock temporal feature for i-th company in t-th trading day:

$$\boldsymbol{x}_i^t = \overrightarrow{\mathrm{GRU}}([\boldsymbol{x}_i^{t-T}, \boldsymbol{x}_i^{t-T+1}, \cdots \boldsymbol{x}_i^{t-1}]), \tag{5}$$

where $\boldsymbol{x}_i^t \in \mathbb{R}^{d_g}$ is the computed temporal representation for the i-th company in t-th trading day and d_g is the output dimension of GRU network.

It should be remarked that the graph simultaneously contains the company nodes and the exective nodes. To obtain the embedding of the executive nodes, we build a executive-company binary ajacent matrix $\boldsymbol{A}^{ec} \in \mathbb{R}^{M \times N}$ to aggregate the executive-related company embeddings as the executive representations, which are followed as:

$$\boldsymbol{h}_{e,j}^t = \tilde{\boldsymbol{x}}^t \cdot \boldsymbol{A}_{:,j}^{ec} \in \mathbb{R}^{d_g}, \tag{6}$$

where M is the number of the executives and $\tilde{\boldsymbol{x}}^t = [\boldsymbol{x}_1^t, \boldsymbol{x}_2^t, \cdots, \boldsymbol{x}_M^t]^{\mathrm{T}} \in \mathbb{R}^{d_g \times M}$. As a result, all node embeddings in the released market knowledge graph can be represented as $\boldsymbol{H} = [\boldsymbol{x}_1^t, \boldsymbol{x}_2^t, \cdots, \boldsymbol{x}_M^t, \boldsymbol{h}_{e,1}^t, \boldsymbol{h}_{e,2}^t, \cdots, \boldsymbol{h}_{e,N}^t] \in \mathbb{R}^{(M+N) \times d_g}$.

3.3 AMRGCN Module

In this subsection, the proposed AMRGCN is an extension of the traditional GCN, which can incorporate the relation semantic information to obtain better node representations. Specifically, AMRGCN first constructs an adjacent matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ $(n = M + N)$ which consists of the relation indexes. By passing through a relation embedding layer with the random initialization, the adjacent matrix can be transformed into an adjacent tensor $\boldsymbol{E} \in \mathbb{R}^{n \times n \times p}$, where $\boldsymbol{E}_{i,j,:} \in \mathbb{R}^p$ is the p-dementional relation representation between entities i and j. p is viewed as the number of channels in the adjacent tensor.

To fully mine the latent relation semantics and further generate better node representations for the SMP task, two submodules are implemented at each layer l of AMRGCN including an attention-aware node update submodule and a multi-order relation update submodule:

$$\boldsymbol{H}^l, \boldsymbol{E}^l = \mathrm{AMRGCN}(\boldsymbol{H}^{l-1}, \boldsymbol{E}^{l-1}). \tag{7}$$

Attention-aware Node Update Submodule

With the public companies in the market interpreted as nodes in the market knowledge graph, an Attention-aware Node Update (AANU) Submodule updates the node representation of stock by aggregating the information from its neighbors across multiple semantic channels in the adjacent tensor. Formally, the above operation can be written as:

$$\begin{aligned} \boldsymbol{H}^l &= \mathrm{AANU}(\boldsymbol{E}^{l-1}, \boldsymbol{H}^{l-1}) \\ &= \sigma(\mathrm{Atten}(\boldsymbol{H}_1^l, \boldsymbol{H}_2^l, \cdots, \boldsymbol{H}_p^l)), \end{aligned} \tag{8}$$

where $\sigma(\cdot)$ is a ReLU activation function and $\mathrm{Atten}(\cdot)$ is an attention operator.

Specifically, the aggregated operation in AANU can be divided into two parts including channel-inner aggregation and multi-channel aggregation. The channel-inner aggregation is a standard aggregation referred to the traditional GCN, which are:

$$\boldsymbol{H}_k^l = \boldsymbol{E}_{:,:,k}^{l-1} \cdot \boldsymbol{H}^{l-1} \cdot \boldsymbol{W} \in \mathbb{R}^{d_g}, \tag{9}$$

where $\boldsymbol{E}^{l-1}$ is the adjacent tensor obtained from initialization or the last AMRGCN layer, $\boldsymbol{H}^0$ denotes the output $\boldsymbol{H}$ in the temporal learning module, $\boldsymbol{W} \in \mathbb{R}^{d_g \times d_g}$ is a learnable filter. Furthermore, to achieve the multi-channel aggregation of the channel-inner aggregated node representations, we introduce an attention-aware mechanism to adaptively compute the weighted sum of node embeddings in the different channels. Mathematically, this operation for i-th company is expressed as:

$$\begin{aligned} \mathrm{Atten}(\boldsymbol{H}_1^{l,i}, \cdots, \boldsymbol{H}_p^{l,i}) &= \sum_{k=1}^{p} \alpha_k \cdot \boldsymbol{H}_k^{l,i}, \\ \alpha_k^i &= \frac{\exp(\boldsymbol{W}_a^T \cdot \boldsymbol{H}_k^{l,i})}{\sum_{s=1}^{p}(\exp(\boldsymbol{W}_a \cdot \boldsymbol{H}_s^{l,i}))} \end{aligned} \tag{10}$$

where $\boldsymbol{H}_k^{l,i} \in \mathbb{R}^{d_g}$ denotes the node embedding acquired from the k-th channel for i-th company and $\boldsymbol{W}_a \in \mathbb{R}^{d_g \times 1}$ is a trainable projected vector.

Multi-order Relation Update Submodule

The original relation embeddings are obtained from the relation embedding layer. However, these embeddings in the adjacent tensor should convey different signals for different companies and executives, which are not company-independent. Also, considering the long-chain cross-shareholding structure in the market knowledge graph, it is not easy to directly stack too many GCN layers for aggregating farther neighbors with higher-order relations. To address this problem, we present a novel Multi-order Relation Update (MORU) Submodule to dynamically update adjacent tensors according to the nodes and the multi-order relations. Formally, MORU operation can be defined as:

$$\boldsymbol{E}_{i,j,:}^l = \mathrm{MORU}(\mathrm{ITGA}(\boldsymbol{E}^{l-1}), \boldsymbol{h}_i^l, \boldsymbol{h}_j^l), \tag{11}$$

where $\mathrm{ITGA}(\cdot)$ is an iterative operator based on a gated mechanism, and $\boldsymbol{h}_i^l \in \boldsymbol{H}^l$ and $\boldsymbol{h}_j^l \in \boldsymbol{H}^l$ respectively denote the embeddings of node i and node j. Noted that $\mathrm{ITGA}(\cdot)$ is related to the layer number l which determines to keep the number of the multi-order adjacent tensors. For instance, in the l-th layer, we compute the high-order adjacent tensors along each semantic channel, which are followed as:

$$\begin{aligned}
&\boldsymbol{E}^{(l-1,0)} = \boldsymbol{E}^{l-1}, \\
&\boldsymbol{E}^{(l-1,c)}_{:,:,k} = \boldsymbol{E}^{(l-1,c-1,\mathrm{T})}_{:,:,k} \cdot \boldsymbol{E}^{(l-1,c-1)}_{:,:,k}, \\
&\beta = \mathrm{Sigmoid}(\boldsymbol{W}^{c-1}_{\beta} \cdot \boldsymbol{E}^{(l-1,c-1)}_{i,j,:}), \\
&\boldsymbol{E}^{(l,c)}_{i,j,:} = (1-\beta) * \boldsymbol{E}^{(l-1,c-1)}_{i,j,:} + \beta * \boldsymbol{E}^{(l-1,c)}_{i,j,:},
\end{aligned} \tag{12}$$

where $c = 0, 1, 2, \cdots, c_{max}$, $c_{max} < l$ means a maximum number of iterations, $\boldsymbol{W}^{c-1}_{\beta} \in \mathbb{R}^{p \times p}$ is a transformation matrix and $\mathrm{Sigmoid}(\cdot)$ is an activation function. Based on this, we collect the final relation embeddings in the computed multi-order adjacent tensor and their corresponded nodes to conduct concatenated operation $\boldsymbol{m}^{l}_{i,j} = \boldsymbol{h}^{l}_{i} \odot \boldsymbol{h}^{l}_{j} \odot \boldsymbol{E}^{(l-1,c_{max})}_{i,j,:} \in \mathbb{R}^{2d_g+p}$. As a result, we feed $\boldsymbol{m}^{l}_{i,j}$ into a linear layer to complete the update of the adjacent tensor which can be written as:

$$\boldsymbol{E}^{l}_{i,j,:} = \boldsymbol{W}_m \cdot \boldsymbol{m}^{l}_{i,j} + \boldsymbol{b}_m, \tag{13}$$

where $\boldsymbol{W}_m$ and $\boldsymbol{b}_m$ respectively denote the projected matrix and the bias vector.

Classifier Layer

By collecting the node presentations of companies obtained from the temporal learning module and each layer in MRGCN, we apply a linear layer and a residual layer to acquire the final embeddings of all the public companies in the market knowledge graph, which can be written as:

$$\overline{\boldsymbol{x}}^{t}_{i} = \mathrm{Normalization}(\boldsymbol{W}_t \cdot \boldsymbol{x}^{t,\odot}_{i} + \boldsymbol{x}^{t}_{i}), \tag{14}$$

where $\mathrm{Normalization}(\cdot)$ is a normalization operator, $\boldsymbol{x}^{t,\odot}_{i} = \boldsymbol{x}^{t}_{i} \odot \boldsymbol{h}^{1}_{i} \odot \boldsymbol{h}^{2}_{i} \odot \cdots \odot \boldsymbol{h}^{l}_{i} \in \mathbb{R}^{(l+1)d_g}$ is the concatenated node embedding for i-th company in the t-trading day, $\boldsymbol{W}_t \in \mathbb{R}^{(l+1)d_g \times d_g}$ is a transformation matrix. Therefore, the final predicted probability distribution of the stock price movement can be expressed as:

$$\hat{\boldsymbol{y}}^{t}_{i} = \mathrm{Softmax}(\boldsymbol{W}_y \cdot \overline{\boldsymbol{x}}^{t}_{i} + \boldsymbol{b}_y) \in \mathbb{R}^2, \tag{15}$$

where $\boldsymbol{W}_y$ and $\boldsymbol{b}_y$ are respectively the projected matrix and the bias vector and $\mathrm{Softmax}(\cdot)$ is an activation function. Based on the predicted probability, we construct a cross-entropy loss which is utilized to conduct the model optimization.

Hyper-parameters	CSI100E	CSI300E
Window size T	20	20
Hidden size S	8	8
Output dimension d_x	8	8
GRU hidden size d_g	78	44
Edge embedding size p	10	50
GCN layer number L	3	3
Dropout rate	0.5	0.5
Learing rate	0.00078	0.0008

Table 1. The hyper-parameters setting on datasets.

Models	CSI100E		CSI300E	
	ACC (%)	AUC (%)	ACC (%)	AUC (%)
LSTM [26]	51.14	51.33	51.78	52.24
GRU [27]	51.66	51.46	51.11	52.30
GCN [28]	51.58	52.18	51.68	51.81
GAT [29]	52.17	52.78	51.40	52.24
RGCN [30]	52.33	52.69	51.79	52.59
HGT [31]	53.01	52.51	51.70	52.19
STHAN-SR [32]	52.78	53.05	52.89	53.48
AD-GAT [2]	54.56	55.46	52.63	54.29
DANSMP [6]	57.75	60.78	55.79	59.36
AMRGCN-SMP (ours)	**62.03**	**65.66**	**57.88**	**62.30**

Table 2. The performance comparison of different models for the SMP task.

4 Experiments

4.1 Dataset and Evaluation Metrics

We conduct experiments on the CSI100E and CSI300E datasets, which are the standard datasets released in [6] for the stock movement prediction task. CSI100E contains 73 company nodes and 163 executive nodes. CSI300E covers 185 public companies and 275 executives. The nodes in these two datasets are linked by 10 kinds of relations with specific semantics such as *Investment*, *Supply chain*, and *Business partnership*. The period of training data is from 21/11/2017 to 05/08/2019 and the period of developing data ranges from 06/08/2019 to 22/10/2019. For the testing data, its covered interval is from 23/10/2019 to 31/12/2019. Referring to the evaluation in [6], we test the models using the official scorer in terms of the Accuracy (ACC) and Area Under Curve (AUC).

4.2 Hyper-parameter Setting

The hyper-parameters in the proposed model are manually tuned by strict grid searches on the developing dataset. As a result, the optimal hyper-parameter groups are listed in Tab. 1. In addition, all the experiments using Pytorch 1.11.0 are conducted on Nvidia GeForce 3090 GPU with Intel(R) Xeon(R) Platinum 8255C CPU.

4.3 Baselines

To comprehensively evaluate the presented SMP model, especially the designed AMRGCN module, we compare it with a range of baselines and recent state-of-the-art methods, which can be concluded into two genres including the sequence-based and GCN-based approaches.

Sequence-based models operates on the historical trading data based on Recurrent Neural Network (RNN). LSTM [26] and GRU [27] are the variants of standard RNN that can learn the time-series stock data of a company to generate the stock feature in the certain trading day for the SMP task.

GCN-based models apply graph convolutional network over the market knowledge graph to exploit potential among companies and executives. 1-4) GCN [28], GAT [29], RGCN [30], and HGT [31] respectively employ linear aggregation, weighted aggregation, heterogeneous mapping aggregation, and heterogeneous transformer aggregation to generate the high-level company node representations for the SMP task. 5) STHAN-SR [32] utilizes hypergraph structure and temporal Hawkes attention to rank stocks based on the historical prices and firm relations during predicting the future stock movement. 6) AD-GAT [2] presents an attribute-driven graph attention network to model interactive market information space to improve stock movement prediction. 7) DANSMP [6] releases the standard CSI100E and CSI300E datasets and uses a dual attention network to learn the momentum spillover signals on the constructed market knowledge graph for SMP.

4.4 Overall Performance

We report the experimental results on the CSI100E and CSI300E datasets in Tab. 2. Obviously, the proposed AMRGCN-SMP achieves state-of-the-art performance and outperforms all the listed baselines. We argue that the performance gain is attributed to three aspects:

1) The application of the relation semantics labels (i.e. edge labels between two entities in the market knowledge graph). AMRGCN outperforms all the compared GCN-based SMP models which only employ the fuzzy binary relations (i.e., 0/1 adjacent matrix) and ignore the specific relation semantics between two entities. This fact demonstrates that relation labels in the market knowledge graph are capable of supplying key information for the SMP task.

2) The design of the attention-ware multi-channel aggregation. Benefiting from the introduction of the relation labels, AMRGCN-SMP far outputs the RGCN with an absolute margin of 9.7% ↑ ACC and 12.97% ↑ AUC. We consider that RGCN utilizes different convolutional filters to respectively aggregate relation-specific neighbors and subsequently aggregates all relation-specific targeted nodes to generate the final targeted node embedding. Thus, the relation semantics in the node aggregation is not fully exploited. By contrast, AMRGCN learns effective relation semantics and provides access that can use an attention mechanism to adaptively aggregate targeted nodes obtained from the multiple semantics channels and thus better captures the information under relations between two entities.

3) The dynamic update of the adjacent tensor based on the multi-order relation. In Tab. 2, recent GCN-based methods are based on a static binary graph whose weights in the adjacent matrix are fixed and lack high-order relation information between entities. Actually, the weights in the adjacent matrix should not be entity-independent and be continuously updated with the node aggregation. Besides, in the modern stock market, there are many long-chain cross-shareholding controlled structures among different entities. Therefore, high-order relations among entities should be concerned in the GCN-based methods. In this work, the proposed AMRGCN achieves the dynamics of the adjacent tensor based on the multi-order relations, which achieve the obvious performance improvement compared with the recent best advance with a margin of 2.48% ↑ ACC and 4.88% ↑ AUC.

5 Analysis

5.1 The Overall Ablation Study

To demonstrate the effectiveness of all components in AMRGCN including the AANU and MORU, we execute an ablation study on the CSI100E dataset as Tab. 3 shows 1) Attention-aware Node Update Module (AANU): to study whether attention-aware multi-channel aggregation contributes to the performance for SMP, we degrade each relation semantic embedding $\boldsymbol{E}_{i,j:} \in \mathbb{R}^p$ $(p > 1)$ in adjacent tensor into a scalar value (i.e., $\boldsymbol{E} \in \mathbb{R}^{n\times n\times p} \to \boldsymbol{E} \in \mathbb{R}^{n\times n\times 1}$), which makes AANU submodule invalided (see Eq. (10)). As a result, the above operation hurts the performance of SMP respectively by 2.28% ACC and 2.76% AUC. This fact verifies that AANU based on the weighted aggregation for multi-channel targeted nodes can provide more evident information by the guidance of the relational semantics for SMP. 2)-Multi-order Relation Update Module (MORU): we remove the MORU submodule in the proposed model and cancel the dynamics of adjacent tensor after node aggregation. This setting means that the adjacent tensor is fixed and losses the information supplement acquired from the multi-order relations. Therefore, the final ACC and AUC performances of SMP respectively reduce by 7.76% and 9.49%, which again demonstrates the effectiveness of the idea about the dynamics of adjacent tensor based on the multi-order relation semantics. In the following subsection, we will continuously evaluate the inner structures of MORU with more detailed views (see subsection 5.3).

Models	CSI100E	
	ACC (%)	AUC (%)
-Best AMRGCN-SMP	**62.03**	**65.66**
w/o AANU submodule	59.75	62.90
w/o MORU submodule	54.27	56.17

Table 3. The ablation study of each submodule in AMRGCN on CSIE100E dataset.

5.2 Effect of Attention-aware Aggregation

To further confirm the effectiveness of the proposed attention-aware multi-channel aggregation, we add another experiment by adding different multi-channel aggregated mechanisms in AANU submodule, and the experimental results are reported in Tab. 4. It is clear that the presented attention-ware multi-channel aggregated mechanism achieves the best performance compared with the listed traditional aggregated manners. In detail, the attention-ware multi-channel aggregated mechanism can automatically allot weights for the multi-channel targeted nodes after channel-inner aggregation. By contrast, the traditional aggregated manners including Max, Sum, and Avg usually respectively suffer from information dropout, equivalent aggregation, and information attenuation. Thus, the attention-aware weighted aggregation is a natural idea to compute multi-channel aggregation and has been demonstrated effectively in the ablation experiments.

Models	CSI100E	
	ACC (%)	AUC (%)
-Max Aggregation	61.62	63.64
-Sum Aggregation	60.77	63.25
-Avg Aggregation	56.96	60.45
-Atten Aggregation*	**62.03**	**65.66**

Table 4. The performance comparison of the different aggregated manners in AMRGCN. * is the proposed attention-aware aggregated manner in AMRGCN.

5.3 Effect of iterative Multi-order Graph

By observing the Eq. (12), we respectively remove the gated mechanism (i.e. setting $\beta = 1$) and control the maximum iteration number c_{max} (if $c_{max} = 1$ and $L = 3$, the relation between layer number l and iteration number c is $c = 0$ with $l = 1$, $c = 1$ with $l = 2$, and $c = 1$ with $l = 3$). Noted that $c_{max} = 0$ means the lack of the multi-order relation information (i.e., Eq. (12) is removed and $\boldsymbol{E}^{l-1,l-1}_{i,j,:} = \boldsymbol{E}^{l-1}_{i,j,:}$). The experimental results are reported in Tab. 5. As shown in Tab. 5, removing the gated mechanism in MORU can lead to obvious decreases of ACC and AUC respectively by absolute margins of 2.22% and 1.23%. This observation demonstrates the introduction of the gated mechanism can effectively adjust the degree of fusion among multi-order adjacent tensors and provide more cues for the SMP task. Moreover, by controlling the maximum iteration number c_{max}, the higher-order relation information can be incorporated into the adjacent tensor during the iterative computation with the increase of c_{max}. Therefore, in Tab. 5, the performances of SMP including ACC and AUC become better with the increase of c_{max}. These results verify that the GCN-based models indeed need to consider the multi-order relation information to supply more cues for SPM, especially in the case of long-chain cross-shareholding structures among public companies.

Models	CSI100E	
	ACC (%)	AUC (%)
-MORU w/o GATE	59.81	64.43
-MORU (c_{max}= 0)	60.03	62.75
-MORU (c_{max}= 1)	61.18	64.59
-MORU (c_{max}= 2)*	**62.03**	**65.66**

Table 5. The effectiveness evaluation of inner structures in MORU. * denotes the best setting of c_{max} in the proposed AMRGCN module.

5.4 *Effect of Relation Embedding Size*

In this work, we introduce relation embeddings in the adjacent tensor to facilitate AMRGCN to achieve multi-channel aggregation and dynamic update of adjacent tensor based on multi-order relations. Thus, the dimension of the relation embeddings may be effective for the presented AMRGCN during prediction stock movement. Based on this, we study the performance of the models with different dimensions of relation representation in this subsection. The value of dimension is varied from 10 to 400 and the experimental results on CSI100E dataset are shown in Fig. 2. We conclude that the best ACC&AUC can be obtained when the dimension is 10. Actually, a larger dimension is not beneficial for the performance improvement of AMRGCN.

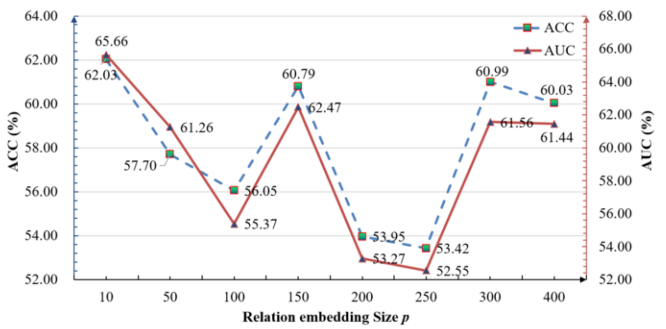

Figure 2. The ACC & AUC variation with relation embedding size on the test dataset of CSI100E.

5.5 *Impact of AMRGCN layers*

As the model stacks L AMRGCN layers, it is necessary to investigate the effect of the layer number L on the final performance of the SMP task. Therefore, we set the different number of layers which range from 1 to 5. The experimental results are illustrated in Fig. 3. As shown in Fig. 3, it can be observed that the overall performance trend is increased if the layer number is set as $L \in \{1, 2, 3\}$. When the layer number is 3, the proposed method can achieve the best performance for the SMP task. However, with the further increase of the layer numbers ranging from 3 to 5, it can be noted that the performances of AMRGCN drop dramatically. This phenomenon is explicable because the model is suffering from the issue of overfitting if the model stacks too many AMRGCN layers. Besides, the dynamic update of the adjacent tensor in AMRGCN can also increase the complexity of the model and bring more severe performance deterioration for the SMP task.

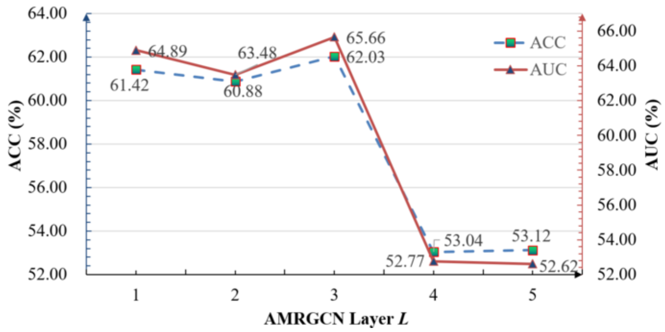

Figure 3. The ACC & AUC variation with GCN number layer on the test dataset of CSI100E.

6 Conclusion

In this paper, we propose a novel method named Attention-aware Multi-order Relation GCN (AMRGCN) for the stock movement prediction. Based on the introduced relational semantics, AMRGCN can achieve attention-aware multi-channel node aggregation and the dynamic update of adjacent tensor using multi-order relational information. The attention-aware multi-channel aggregation mechanism can adaptively fuse the targeted node embeddings derived from the multiple semantic channels and the dynamic update of adjacent tensor based on the multi-order relations can facilitate the model to capture multi-order neighbor information. Experiments show that our model achieves the start-of-the-art results respectively on the SCI100E and CSI300E datasets. In the future, we would like to explore some novel mechanisms to improve the problem of overfitting during applying the AMRGCN when it has stacked more layers.

References

[1] Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu, 'Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction', in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, eds., Yi Chang, Chengxiang Zhai, Yan Liu, and Yoelle Maarek, pp. 261–269. ACM, (2018).

[2] Rui Cheng and Qing Li, 'Modeling the momentum spillover effect for stock prediction via attribute-driven graph attention networks', in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 55–62. AAAI Press, (2021).

[3] Guifeng Wang, Longbing Cao, Hongke Zhao, Qi Liu, and Enhong Chen, 'Coupling macro-sector-micro financial indicators for learning stock representations with less uncertainty', in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 4418–4426. AAAI Press, (2021).

[4] Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua, 'Temporal relational ranking for stock prediction', *ACM Trans. Inf. Syst.*, **37**(2), 27:1–27:30, (2019).

[5] Jiexia Ye, Juanjuan Zhao, Kejiang Ye, and Chengzhong Xu, 'Multi-graph convolutional network for relationship-driven stock movement prediction', in *25th International Conference on Pattern Recognition, ICPR 2020, Virtual Event / Milan, Italy, January 10-15, 2021*, pp. 6702–6709. IEEE, (2020).

[6] Yu Zhao, Huaming Du, Ying Liu, Shaopeng Wei, Xingyan Chen, Fuzhen Zhuang, Qing Li, Ji Liu, and Gang Kou, 'Stock movement prediction based on bi-typed hybrid-relational market knowledge graph via dual attention networks', *CoRR*, **abs/2201.04965**, (2022).

[7] Jung-Bin Li, Sheng-Hsiu Wu, Mu-Yen Chen, and An-Pin Chen, 'Applying technical analysis of stock trends to trading strategy of dynamic portfolio insurance', in *Proceedings of the 2006 Joint Conference on Information Sciences, JCIS 2006, Kaohsiung, Taiwan, ROC, October 8-11, 2006*. Atlantis Press, (2006).

[8] Chi Chen, Li Zhao, Jiang Bian, Chunxiao Xing, and Tie-Yan Liu, 'Investment behaviors can tell what inside: Exploring stock intrinsic properties for stock trend prediction', in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, eds., Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis, pp. 2376–2384. ACM, (2019).

[9] David M. Q. Nelson, Adriano C. M. Pereira, and Renato A. de Oliveira, 'Stock market's price movement prediction with LSTM neural networks', in *2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA, May 14-19, 2017*, pp. 1419–1426. IEEE, (2017).

[10] Yumo Xu and Shay B. Cohen, 'Stock movement prediction from tweets and historical prices', in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, eds., Iryna Gurevych and Yusuke Miyao, pp. 1970–1979. Association for Computational Linguistics, (2018).

[11] Liheng Zhang, Charu C. Aggarwal, and Guo-Jun Qi, 'Stock price prediction via discovering multi-frequency trading patterns', in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pp. 2141–2149. ACM, (2017).

[12] Wen Long, Zhichen Lu, and Lingxiao Cui, 'Deep learning-based feature engineering for stock price movement prediction', *Knowl. Based Syst.*, **164**, 163–173, (2019).

[13] Wenjie Lu, Jiazheng Li, Jingyang Wang, and Lele Qin, 'A cnn-bilstm-am method for stock price prediction', *Neural Comput. Appl.*, **33**(10), 4741–4753, (2021).

[14] Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, and Rajiv Ratn 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, Online, November 16-20, 2020*, eds., Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, pp. 8415–8426. Association for Computational Linguistics, (2020).

[15] Qiuyue Zhang, Chao Qin, Yunfeng Zhang, Fangxun Bao, Caiming Zhang, and Peide Liu, 'Transformer-based attention network for stock movement prediction', *Expert Syst. Appl.*, **202**, 117239, (2022).

[16] Shangzhe Li, Junran Wu, Xin Jiang, and Ke Xu, 'Chart GCN: learning chart information with a graph convolutional network for stock movement prediction', *Knowl. Based Syst.*, **248**, 108842, (2022).

[17] Heyuan Wang, Shun Li, Tengjiao Wang, and Jiayi Zheng, 'Hierarchical adaptive temporal-relational modeling for stock trend prediction', in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, ed., Zhi-Hua Zhou, pp. 3691–3698. ijcai.org, (2021).

[18] Yu Ma, Rui Mao, Qika Lin, Peng Wu, and Erik Cambria, 'Multi-source aggregated classification for stock price movement prediction', *Inf. Fusion*, **91**, 515–528, (2023).

[19] Fuli Feng, Huimin Chen, Xiangnan He, Ji Ding, Maosong Sun, and Tat-Seng Chua, 'Enhancing stock movement prediction with adversarial training', in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, ed., Sarit Kraus, pp. 5843–5849. ijcai.org, (2019).

[20] Hongfeng Xu, Donglin Cao, and Shaozi Li, 'A self-regulated generative adversarial network for stock price movement prediction based on the historical price and tweets', *Knowl. Based Syst.*, **247**, 108712, (2022).

[21] Jintao Liu, Hongfei Lin, Xikai Liu, Bo Xu, Yuqi Ren, Yufeng Diao, and Liang Yang, 'Transformer-based capsule network for stock movement prediction', in *Proceedings of the First Workshop on Financial Technology and Natural Language Processing*, pp. 66–73, Macao, China, (August 2019).

[22] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio, 'Graph attention networks', *CoRR*, **abs/1710.10903**, (2017).

[23] Yilun Jin, Guojie Song, and Chuan Shi, 'Gralsp: Graph neural networks with local structural patterns', in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 4361–4368. AAAI Press, (2020).

[24] Wei Li, Ruihan Bao, Keiko Harimoto, Deli Chen, Jingjing Xu, and Qi Su, 'Modeling the stock relation with graph network for overnight stock movement prediction', in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, ed., Christian Bessiere, pp. 4541–4547. ijcai.org, (2020).

[25] Meng-Hao Guo, Zheng-Ning Liu, Tai-Jiang Mu, and Shi-Min Hu, 'Beyond self-attention: External attention using two linear layers for visual tasks', *CoRR*, **abs/2105.02358**, (2021).

[26] Sepp Hochreiter and Jürgen Schmidhuber, 'Long short-term memory', *Neural Comput.*, **9**(8), 1735–1780, (1997).

[27] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, 'Learning phrase representations using RNN encoder-decoder for statistical machine translation', in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, eds., Alessandro Moschitti, Bo Pang, and Walter Daelemans, pp. 1724–1734. ACL, (2014).

[28] Thomas N. Kipf and Max Welling, 'Semi-supervised classification with graph convolutional networks', in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, (2017).

[29] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio, 'Graph attention networks', in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, (2018).

[30] Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling, 'Modeling relational data with graph convolutional networks', in *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings*, eds., Aldo Gangemi, Roberto Navigli, Maria-Esther Vidal, Pascal Hitzler, Raphaël Troncy, Laura Hollink, Anna Tordai, and Mehwish Alam, volume 10843 of *Lecture Notes in Computer Science*, pp. 593–607. Springer, (2018).

[31] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun, 'Heterogeneous graph transformer', in *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, eds., Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen, pp. 2704–2710. ACM / IW3C2, (2020).

[32] Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, Tyler Derr, and Rajiv Ratn Shah, 'Stock selection via spatiotemporal hypergraph attention network: A learning to rank approach', in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 497–504. AAAI Press, (2021).