Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Highlights

Multimodal multiscale dynamic graph convolution networks for stock price prediction

Pattern Recognition xxx (xxxx) xxx

Ruirui Liu, Haoxian Liu, Huichou Huang, Bo Song, Qingyao Wu*

- We propose a novel Multiscale Multimodal Dynamic Graph Convolution Network.
- The MMFDB effectively extract and align rich multimodal feature representations.
- The STGCL are powerful in learning complex relations simultaneously.
- Extensive experiments justify the superior performance of our proposed model.

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Multimodal multiscale dynamic graph convolution networks for stock price prediction

Ruirui Liu [a,b,1], Haoxian Liu [c,d,1], Huichou Huang [e,1], Bo Song [c,d], Qingyao Wu [c,*]

[a] Department of Economics and Finance, Brunel University London, London, United Kingdom
[b] Data Analytics for Finance and Macro Centre, King's College London, London, United Kingdom
[c] School of Software Engineering, South China University of Technology, Guangzhou, China
[d] Key Laboratory of Big Data and Intelligent Robot, Ministry of Education, Guangzhou, China
[e] Global Research Unit, College of Business, City University of Hong Kong, Hong Kong, China

ARTICLE INFO

Keywords:
Stock movement prediction
Multimodal feature fusing
Multiscale architecture
Graph convolutional network

ABSTRACT

Predicting directional future stock price movements is very challenging due to the complex, stochastic, and evolving nature of the financial markets. Existing literature either neglects other timely and granular alternative data, such as media text data, or fails to extract and distill predictive multimodal features from the data. Moreover, the time-varying cross-sectional relations beyond sequential dependencies of stock prices are informative for forecasting price fluctuations, for which the modelling flexibility, however, is not adequate in most of the previous studies. In this paper, we propose a novel **M**ultiscal**e** Multimo**d**al **D**ynamic **G**raph **C**onvolution **N**etwork (Melody-GCN) to address these issues in stock price prediction. It contains three core modules: (1) multimodal fusing-diffusing blocks that effectively integrate and align the numerical and textual features; (2) a multiscale architecture that extracts and refines temporal features via a fine-to-coarse descending path and a coarse-to-fine ascending path progressively; and (3) dynamic spatio-temporal graph convolutional layers that learn the complex and evolving stock relations not only in between industries and individual companies but also across time horizons. Extensive experimental results and trading simulations on two real-world datasets demonstrate the superior performance of our proposed approach beyond other state-of-the-art models.

1. Introduction

Predicting directional future stock price movements is one of the key investment activities of practitioners in the financial industry. It is of paramount importance for funds' daily portfolio and risk management. Comprehensive studies in the applications of machine learning, deep learning, reinforcement learning, *etc.*, to relevant fields have achieved promising outcomes and thereby attract growing research interests [1]. However, this task still remains very challenging due to the complex, stochastic, and evolving nature of the financial markets. Some existing literature overlooks the importance of readily available alternative data, such as media text data, which is known to offer timely and granular predictive information for the task. Others using this type of data fail to extract and distill rich and useful predictive multimodal features out of the data for the forecasting exercises. Moreover, even though the time-varying cross-sectional and sequential relations beyond stock prices contain valuable supplementary information for future price fluctuations, such as the industry linkages that implicitly reflect supply and demand chains in the entire company network, they are not explicitly considered in most previous works. On the other hand, graph convolutional networks (GCN) have been proven to be good at processing sparse and non-grid data in various fields. Previous works based on GCN [2,3] always require expert prior knowledge to calculate the adjacency graphs in advance and set them fixed during training, which was inflexible.

In this paper, we propose a **M**ultiscal**e** Multimo**d**al **D**ynamic **G**raph **C**onvolution **N**etwork (Melody-GCN) to tackle the aforementioned challenges in the task of stock price prediction. The pipeline is shown in Fig. 2. Intuitively speaking, the task aims to learn both local (short-term) fine-grained features and global (long-term) coarse-grained patterns from the price time series as well as the corresponding media text data. However, the main issue here is the low signal-to-noise ratio in both stock price time series and media text data in the information

* Correspondence to: South China University of Technology, 382 Zhonghuan Road East, Panyu District, Guangzhou Higher Education Mega Centre, Guangzhou, 510006, China.
E-mail addresses: ruirui.liu@brunel.ac.uk (R. Liu), 1241020710@qq.com (H. Liu), huichou.huang@exeter.oxon.org (H. Huang), bradleysong@163.com (B. Song), qyw@scut.edu.cn (Q. Wu).
[1] Contribute equally to this work.

https://doi.org/10.1016/j.patcog.2023.110211
Received 6 April 2023; Received in revised form 20 November 2023; Accepted 15 December 2023

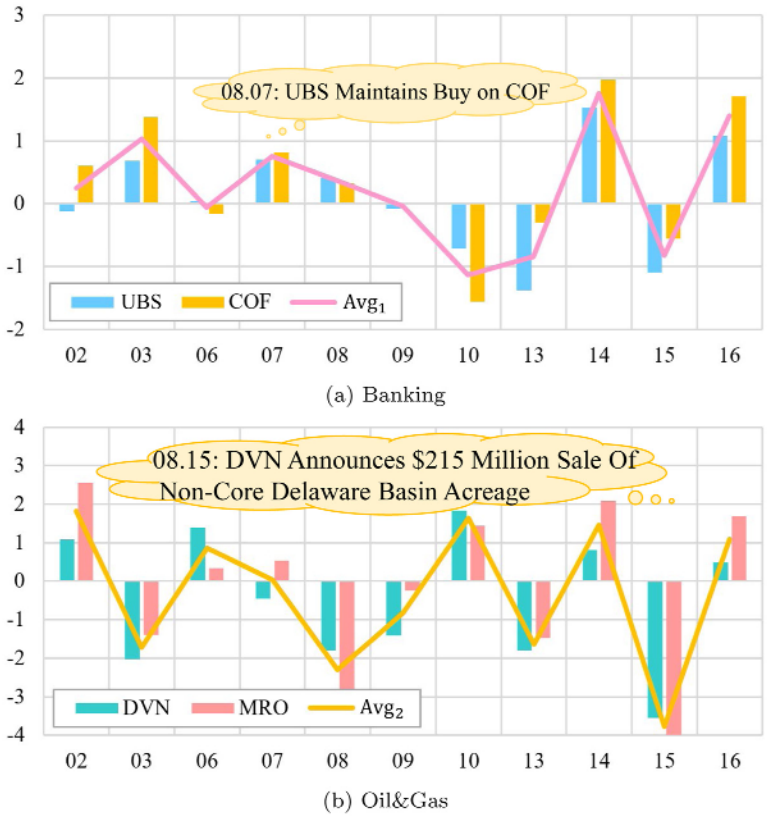


Fig. 1. The return ratio of four firms from 2018.08.02 to 2018.08.16. In sub-figure (a), UBS and COF belong to the banking industry. In sub-figure (b), DVN and MRO are oil and gas production firms. It shows the existence of the cross-sectional and time-evolving "momentum spillover" effects of stock price movements (possibly through the risk-premium channel), and how the stock prices potentially react to media news.

fusion process, for which we propose two ways to tackle relevant problems.

The first is to use a multiscale approach to extract local (short-term) features at the finest scale and gradually downsample and distill the sequences along the temporal axis in order to obtain smoother sequences that carry more information from their neighbourhoods, building up the descending path. Then they are upsampled and reconstructed back towards the finer scales, forming the ascending path, so that the hidden feature representations are adequately refined to produce rich and accurate feature representations. Furthermore, to fully leverage the supplementary information in media text data in the multimodal feature blending process, we develop Multimodal Fusing-Diffusing Blocks (MMFDB) that produce converged features using a fusing module and two separate diffusing modules with residual connections for exploiting complementary features from the mixtures in order to update the original numerical and textual features. Unlike the conventional manners widely used in previous studies that concatenate multimodal features from two independent streams, our proposed MMFDB repeatedly integrates and decouples two heterogeneous features until the relevant features are distilled.

Second, the existence of inter-industry and inter-company "momentum spillovers" effects is economically meaningful, *i.e.*, a firm-specific shock may be gradually propagated through the supply and demand chains and company-level networks, and ultimately contaminate the entire economic system affecting all other industries. The shock transmission gives rise to the observed lead–lag effects. For example, as shown in Fig. 1, On August 7th, when the positive news of "UBS maintains buy on COF" in the banking industry was announced, both stocks rose. On August 15th, after the news of "DVN announces sale of Delaware Basin Acreage" in the oil and gas industry was announced, not only the stocks of two companies in the energy industry (oil and gas), but also the two stocks in the banking industry fell.

Given this evidence, both spatial and temporal dependencies are crucial for predicting stock prices. Dealing with stock price and media text data together will inevitably encounter asynchronous and imbalanced issues between two-panel time series, and the extracted features may be attributable to future stock price movements in different horizons. To handle these problems, we design a set of dynamic Spatio-Temporal Graph Convolutional Layers (STGCL) to simultaneously learn the complex and evolving dynamics of the inter-industry, inter-company, and inter-day relations. Instead of using the predefined linkages, our proposed STGCL addresses the learnability and optimality of parameters via the adjacency matrices during the training process without resorting to prior expert knowledge.

In summary, our contributions are summarized as follows:

- We propose a novel **M**ultiscale **M**ulti**m**odal **Dy**namic **G**raph **C**onvolution **N**etwork (Melody-GCN) for stock price prediction.
- The proposed multiscale framework captures the local fine-grained and global coarse-grained features, the innovative Multimodal Fusing-Diffusing Block (MMFDB) further effectively extracts and aligns rich multimodal feature representations for the task, and the adaptive Spatio-Temporal Graph Convolution Layer (STGCL) are powerful in learning complex and evolving dynamics of the inter-industry, inter-company, and inter-day relations simultaneously in a flexible manner.
- Extensive empirical experiments on two real-world datasets using various evaluation metrics justify the superior performance of our proposed model beyond other methods.

To better interpret our contributions, we highlight them in comparison with the related SOTA models in the following Related Work section.

2. Related work

Traditional methods for predicting stock prices focus on time-series analysis using statistical methods or conventional machine learning algorithms that fail to account for nonlinearity and interactions among predictors. Recent years have witnessed the popularity of deep learning methods that have produced promising outcomes in the applications to various fields, among which stock price prediction attracts notable attention. Previous works build various models based on classical networks, like RNN [4], CNN [5], Transformer [6], and GCN [3]. In the following, we will further divide them into individual and correlated prediction methods and give a more detailed introduction. Our work falls in the field of correlated prediction, and the contributions of our proposed model are highlighted in comparison with the related SOTA models.

2.1. Individual stock price prediction

Individual stock price prediction only uses the historical information of single stocks, ignoring the correlations across stocks. In other words, it simply explores sequential dependencies for the forecasting exercises. Some of the studies focus on the price sequences [7,8], while others involve other auxiliary information corresponding to stocks [9].

Specifically, among the first kind of methods, Zhang et al. [10] propose a State Frequency Memory (SFM) Recurrent Network, which decomposes the hidden state of data into multiple frequency components, each component simulating a specific frequency of trading pattern behind stock price fluctuations. Then, the future stock price is predicted with a non-linear mapping function that combines these components using Inverse Fourier Transform (IFT). Modelling multi-frequency trading patterns can provide more accurate predictions for various time ranges: short-term forecasts typically depend on high-frequency trading patterns, while long-term forecasts should focus more on low-frequency trading patterns. Ding et al. [7] use Transformer as a basic network to predict the fluctuation of stock prices and make improvements to the three problems existing in the original transformer model. Firstly, the ability of the Transformer to perceive local relationships is enhanced by adding a multiscale Gaussian prior kernel. Secondly, orthogonal regularization constraints are used to avoid redundancy in the multi-headed self-attention system. At the same time, a multi-level segmentation of the time series, "minute level - day level - week level," is implemented to learn the hierarchical characteristics of the financial data. Experiments show that the proposed method has the advantage of mining extremely long-term dependencies from stock time series. Liu et al. [8] propose a bidirectional multiscale network that extracts multiscale data representations from data obtained based on wavelet transform and downsampling operation, respectively, to improve the accuracy of prediction. Zhang et al. [11] point out that the current prediction model is trained on a long-term dataset, which performs best on average but cannot adapt to different markets in different periods. To solve this problem, they propose a novel online model optimization algorithm. A lightweight model library is built. Each lightweight model in the library corresponds to different market distributions. By designing an appropriate reward function, the algorithm can accurately estimate the profits of each model through inverse reinforcement learning. Then in real-time trading, the model can be automatically selected, so that the trading strategy can automatically adapt to the changes in the trading market.

In addition to the stock price sequence itself, other related information, such as media texts, company annual reports, and earnings calls, may also contain valuable signals for stock forecasting. Therefore, many efforts have also introduced these data to provide auxiliary information and strive to make more accurate predictions. Among these methods, Ding et al. [12] propose an event-driven approach for stock market forecasting. First, events are extracted from news text and represented as dense vectors using a new neural tensor network. Secondly, a deep convolutional neural network is used to simulate the short-term and long-term effects of events on stock price fluctuations. The experimental results show that compared to the most advanced baseline methods, the model can achieve nearly 6% improvement in real-world datasets. Besides, trading simulation results show that the system is more profitable than previous methods. To address this challenge, the work mimics the learning process of humans facing chaotic online news. Specifically, the work designs a hybrid attention network with a self-paced learning mechanism to predict stock trends based on sequences of recent relevant news. The stock prediction and investment simulation experiments using this model on real-world stock datasets have achieved good results. Liu et al. [9] propose a hierarchical complementary attention network to capture valuable complementary information from the news content in addition to using the news title. This model uses a two-level attention mechanism to quantify the importance of words and sentences in a given news item. In addition, a new measurement method was designed to calculate attention weight to avoid capturing redundant information in news headlines and content. Li et al. [13] point out that there are two challenges in processing the multimodal stock and related media text data, i.e., how to model the interactions between two modalities and how to align the two heterogeneous modalities. They propose a tensor-based event-driven LSTM approach to address this issue. Unlike them, we design a sophisticated and efficient Multimodal Fusing-Diffusing Block (MMFDB). First, these two heterogeneous data are aligned using a fusion module, and then two independent diffusing modules are used to explore their interactions and supplement each other with cross-modal information.

2.2. Correlated stock price prediction

Correlated stock price prediction involves not only temporal dependencies but also cross-sectional or spatial correlations. Related works are mainly based on various variants of graph convolution networks (GCN). Cao et al. [14] propose an end-to-end Spectral Temporary Graph Neural Network, StemGNN, which processes data characteristics in the spectral domain to predict stock price fluctuations. In StemGNN, Graph Fourier Transform is used to model spatial dependencies between stocks, while Discrete Fourier Transform is used to model temporal correlations. Combining the features learned by these two modules, a more discriminative data pattern can be obtained, allowing for effective prediction through subsequent convolutional and learning modules. In addition, StemGNN can automatically learn cross-sequence correlations from data without using predefined prior knowledge.

Li et al. [15] propose an LSTM-based graph convolutional network that uses correlation matrices between data to simulate the interactions between stocks. Zhao et al. [3] propose a graph-based approach that combines information from multiple perspectives, such as long-term trends, short-term shocks, and unexpected events, into a heterogeneous graph to learn the relationship between multiple stocks. Wang et al. [2] propose a new hierarchical adaptive time relational network to characterize and predict the movement of stock trends. On the one hand, short-term and long-term data characteristics are gradually grasped from stock trading sequences through a multiscale structure with stack dilated causal convolution and gated paths. On the other hand, a dual attention mechanism is proposed, which combines the Hawkes process and specific target queries to detect important time points based on individual stock characteristics. In addition, a multi-graphical interaction module has been designed, integrating prior knowledge and data-driven adaptive learning methods to capture inter-dependencies between stocks.

Ye et al. [16] propose Multi-GCGRU, which combines a graph convolution network and a gated recurrent unit to predict stock movements. Specifically, they built a fixed graph based on financial domain knowledge, corresponding with a dynamic graph with learnable parameters, to learn the cross-sectional spatial correlations. At each time step,

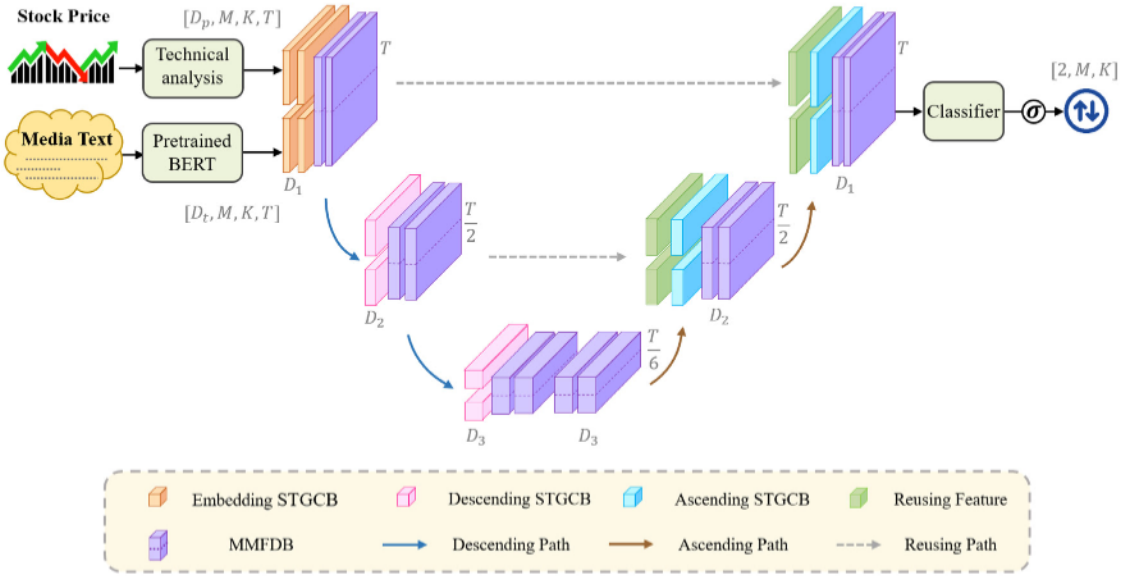


Fig. 2. Pipeline of our proposed **M**ultiscale **M**ultimodal **D**ynamic **G**raph **C**onvolution **N**etwork (Melody-GCN), that principally consists of a descending path and a symmetrical reusing and an ascending path to extract and further refine the local (short-term) features and the global (long-term) features along the temporal axis for producing rich and accurate feature representations. The stock price data and the media text data are preprocessed using technical analysis and pre-trained BERT before further fed into the Embedding Spatio-Temporal Graph Convolution Block (STGCB), and the multimodal numerical and textual features are merged using the Multimodal Fusing-Diffusing Blocks (MMFDB).

the joint multi-graph convolution is used to extract spatial features, and then the relationship between time frames is extracted using GRU. Their model architecture is relatively simple in comparison to our proposed model in the sense that (i) we design a dedicated mechanism for deep mining of multi-granularity information, and (ii) we utilize auxiliary information from the relevant media-text multimodality.

Yin et al. [17] propose a Graph Attention Long Short-Term Memory (GALSTM) for learning the correlations between stocks and automatically predicting their future prices. First, a multi-Hawkes process is used to initialize the correlation graph between stocks. This process provides a good training starting point. Then, an attention-based LSTM is built to learn the weighting matrix of the graph for extracting relationships between time frames and making frame-to-frame forecasts. To avoid (i) the potential accumulated errors of the LSTM-based methods and (ii) the instability and gradient descent issues of frame-to-frame forecasting in capturing long-term feature dependencies, we propose the use of the multi-layer fully dynamic graph convolution for learning the cross-sectional relations in our Melody-GCN, where the layers are sequentially stacked and each layer of graph convolution learns an adjacency matrix adaptively and distinctively from each other across layers. Therefore, overall, more accurate discriminative features can be extracted. Furthermore, our model utilizes STGCL to extract both temporal and spatial features simultaneously, which is computationally simpler and more efficient.

Cheng et al. [18] propose a multimodal graph neural network that uses stock prices and media news to predict financial time series. This method introduces a two-stage attention mechanism, which makes the model interpretable. Users can analyse the importance of inner-modality and inter-modality data. Different from their work, we build a fully dynamic and adaptive graph convolution network in a multiscale architecture, enabling it to better extract multimodal data characteristics.

3. Methodology

We consider the stock price prediction task with C corporations. Each corporation has two kinds of input data, *i.e.*, the stock price correlated numerical data $\boldsymbol{S} = \{\mathbf{s}_1^c, \mathbf{s}_2^c, \dots, \mathbf{s}_T^c\}_{c=1}^{C} \in \mathbb{R}^{D_p \times C \times T}$ and the media text data $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{D_t \times C \times T}$. Here $\mathbf{s}_t^c \in \mathbb{R}^{D_p}$ indicates each company contains D_p kinds of numerical data every day, including open price, high price, low price, close price, and other indicators, and T represents the length of the look-back window. We define the label as the future directional movements of the close price at the $T + \xi$ day by the indicator function $\mathbf{y} = \left\{ \mathbb{I}\left(p_{T+\xi}^c > p_T^c\right) \right\}_{c=1}^{C} \in \mathbb{R}^C$. Given $\boldsymbol{S}$ and $\boldsymbol{\mathcal{T}}$ as input, we can construct a deep neural network $\mathcal{F}$ with parameters $\boldsymbol{\theta}$: $\mathbf{y} = \mathcal{F}_{\boldsymbol{\theta}}\left(\boldsymbol{S}, \boldsymbol{\mathcal{T}}\right)$ to predict the movement. In the following subsections, we will first introduce the overall architecture of our proposed Melody-GCN, then we will show the details of the dynamic STGCB, and the MMFDB.

3.1. The architecture of the Melody-GCN

Instinctively, not only stock time series but also media text sequences possess the properties of short-term local fluctuations and long-term global trends. This motivates us to propose the UNet-like multi-scale architecture, *i.e.*, Melody-GCN that extracts both features along the temporal axis through bi-directional fine-to-coarse (descending) and coarse-to-fine (ascending) paths. In particular, in the latter path, the mixed and refined hidden features are upsampled progressively via the accompanying ascending and reusing path. This procedure enriches the feature representations for the classification head.

Descending Path. As shown in Fig. 2, the original price and text data are first fed into the technical analysis and pre-trained BERT modules respectively to acquire the preliminary features $\boldsymbol{S}$ and $\boldsymbol{\mathcal{T}}$. Then two corresponding STGCB embed them into hidden spaces. The fine-to-coarse descending path with MMFDB and descending STGCB perform feature extraction and abstraction from local to global levels with emphases on the short-term fluctuations and long-term trends respectively. Specifically, the descending blocks reduce the feature maps along the temporal axis with the STGCB (as shown in Fig. 4) but without residual connections, as the feature sizes of the outputs are not identical to those of the corresponding inputs, and the MMFDB aligns and integrates multimodal numerical and textual features.[2]

[2] In the U-Net framework, the descending path aims to perform feature extraction and abstraction from local to global levels. In this way, the temporal feature information is extracted and merged based on the compressed intervals of time in the downsampling so that, in each time step, these features represent global-level information of the time span.

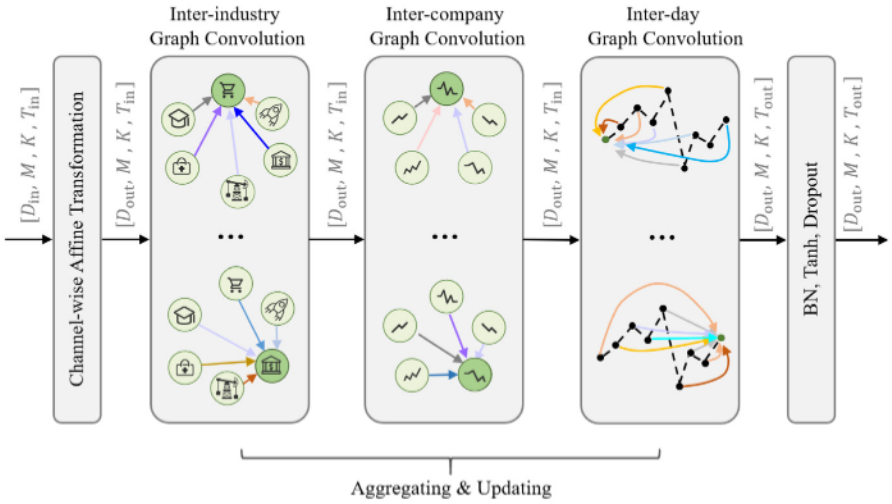


Fig. 3. Illustration of the dynamic STGCL. To aggregate and update the hidden features, the STGCL first project inputs into a high-dimensional space using a channel-wise affine transformation, followed by three sequential graph convolution layers with learnable and dynamic adjacency matrices that adaptively learn the spatio-temporal corrections across different industries, companies, and temporal points.

Reusing and Ascending Path. To bridge interactions among features at different levels and better integrate them with each other, features in the coarse-to-fine ascending path are first upsampled by the ascending STGCB. Meanwhile, the hidden features encoded by the descending blocks at the same scale levels are passed to the ascending blocks via the reusing path. Finally, the enriched and refined features are concatenated by the MMFDB.[3]

Classification Head. The finest features at the first scale output in the reusing and ascending path are denoted by $\mathbf{H}_p \in \mathbb{R}^{D\times M\times K\times T}$ and $\mathbf{H}_t \in \mathbb{R}^{D\times M\times K\times T}$, to which the channel-wise concatenation is applied. Next, the STGCL projects the combined features into the up-or-down bi-directional probabilities of stock price movements in the future day $\tilde{\mathbf{y}} \in \mathbb{R}^{2\times M\times K}$ that can be written as:

$$\tilde{\mathbf{y}} = \sigma\left(\text{STGCL}(\mathbf{H}_p \copyright \mathbf{H}_t)\right), \tag{1}$$

where $\sigma(\cdot)$ means the Sigmoid activation function.

Finally, the parameters are optimized by minimizing the cross-entropy loss throughout N training samples as follows:

$$\mathcal{L} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{k=1}^{K}[y_{n,m,k}\log\tilde{y}_{n,m,k} + \left(1-y_{n,m,k}\right)\log\left(1-\tilde{y}_{n,m,k}\right)]. \tag{2}$$

3.2. Dynamic Spatio-Temporal Graph Convolution Block

To learn the inter-industry, inter-company, and inter-day stock correlations simultaneously, we design the STGCB, which consists of two dynamic STGCL. In particular, we divide C stocks into different clusters according to the industry categories, resulting M industries and K companies ($K = C/M$). Therefore, the sizes of numerical and textual data become $\mathcal{S} \in \mathbb{R}^{D_p\times M\times K\times T}$ and $\mathcal{T} \in \mathbb{R}^{D_t\times M\times K\times T}$ respectively. As shown in Fig. 3, the STGCL first feeds the input data into a hidden space with the channel-wise affine transformation, and then three graph convolution layers in turn, which aggregate and update the inter-industry, inter-company, and inter-day features respectively. Take the inter-industry graph convolution process as an example, it builds the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ by formulating the features of each industry as a vertex in $\mathcal{V}$ and the correlations between each other as an edge in $\mathcal{E}$. Consequently, the learning of the relationships among industries is tied to a learnable dynamic adjacency matrix $\mathbf{A}_{\text{ind}}$. Finally, let $\mathbf{H}^{l-1} \in \mathbb{R}^{D_{\text{in}}\times M\times K\times T_{\text{in}}}$ be the input features for the current STGCL, the outputs can be computed as:

$$\begin{aligned}\mathbf{H}^{l} &= \text{STGCL}(\mathbf{H}^{l-1}) \\ &= \rho\left(\mathbf{W}^{\top}\mathbf{H}^{l-1}\mathbf{A}_{\text{ind}}\mathbf{A}_{\text{com}}\mathbf{A}_{\text{day}} + \mathbf{b}\right)\end{aligned} \tag{3}$$

where $\mathbf{W} \in \mathbb{R}^{D_{\text{in}}\times D_{\text{out}}}$ and $\mathbf{b} \in \mathbb{R}^{D_{\text{out}}}$ represent the parameters of the affine transformation shared by all M, K, and T_{in}. $\mathbf{W}^{\top}$ is the transpose of $\mathbf{W}$. The"$\mathbf{A}$"s are adaptive adjacency matrices containing inter-industry features denoted by 'ind', inter-company features denoted by 'com', and inter-day features denoted by 'day' with the sizes of $M\times M$, $K\times K$, and $T_{\text{in}}\times T_{\text{out}}$, respectively. $\rho(\cdot)$ is the optional operations, including the batch normalization, the Tanh activation function, and the dropout.

Moreover, to extract the hidden features more effectively and aggressively, we sequentially stack two STGCL together and employ residual connections across the modules if the sizes of inputs and outputs are identical, which forms the STGCB, as shown in Fig. 4.[4] Hence the STGCB could be calculated as

$$\mathbf{H}^{l} = \text{STGCB}(\mathbf{H}^{l-1}) = \text{STGCL}_2\left(\text{STGCL}_1\left(\mathbf{H}^{l-1}\right)\right) + \lambda\mathbf{H}^{l-1} \tag{4}$$

where λ is a binary scalar with the value of 0 or 1.

3.3. Multimodal fusing-diffusing block

A common strategy in most of the existing literature is to concatenate multimodal data without considering their interactions, which are, however, highly important to promote the effectiveness of the features learning in respect modes. To account for the multimodal interactions and alignments, we design novel Multimodal Fusing-Diffusing Blocks (MMFDB) as shown in Fig. 5. MMFDB first concatenate the price features $\mathbf{H}_p^{l-1}$ and text features $\mathbf{H}_t^{l-1}$, and feed them into a fusing module $\mathcal{F}_{\text{fus}}$ made up of the STGCB to obtain converged multimodal features

[3] The upsampled features in the ascending path along the temporal dimension correspond to the downsampled features in the descending path via the reusing path. The reliance on the reusing path determines the order of the descending and ascending paths. So, these paths are not designed to be swapped.

[4] We address the over-smoothing problem in two ways: (1) We limit the number of graph convolutions for each type of relation graph, i.e. industry, company, and day. In our model, there are six layers of STGCB in total. (2) The reusing path method can alleviate the over-smoothing problem by integrating the features in the ascending and descending paths. Moreover, the features in the descending path are only processed by a few graph convolution layers and do not suffer from over-smoothing.

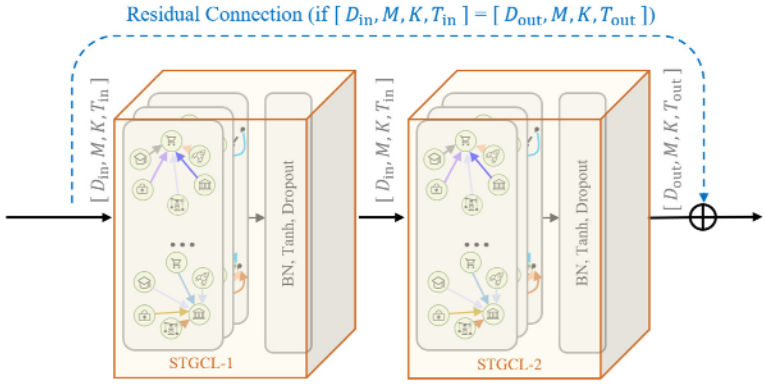


Fig. 4. The construction of basic Spatio-Temporal Graph Convolution Block (STGCB). STGCB consists of two sequentially stacked STGCL with residual connections applied to the input and output when their feature sizes are identical. It helps us to more effectively and aggressively extract the hidden features.

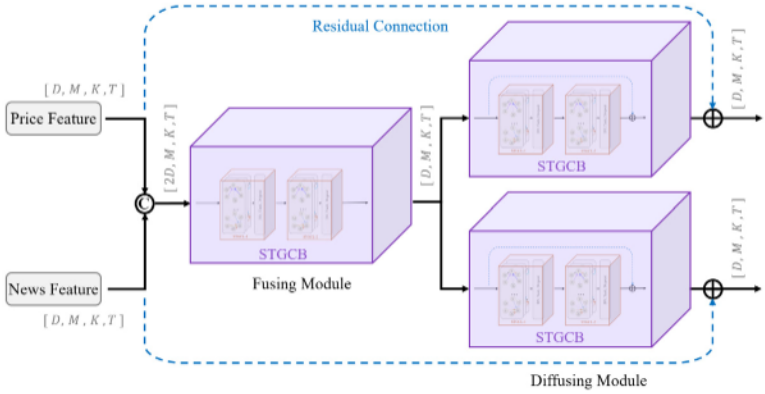


Fig. 5. The component of basic Multimodal Fusing-Diffusing Block (MMFDB). MMFDB built on three STGCBs is used to align and integrate the numerical and textual features. Firstly a fusing module integrates the concatenated price and text features, and then two separate residual diffusing modules extract complementary information from the mixed one to augment their original single-modal features.

$\mathbf{H}_{\text{fus}}$. Then two diffusing modules $\mathcal{T}_{\text{dif-p}}$ and $\mathcal{T}_{\text{dif-t}}$ consist of the STGCB extract supplementary features from the mixtures to update the original single-modal data with residual connections. Finally, MMFDB output the enriched and refined price features $\mathbf{H}_p^l$ and text features $\mathbf{H}_t^l$ as follows:

$$\begin{cases} \mathbf{H}_{\text{fus}} = \mathcal{T}_{\text{fus}}\left(\mathbf{H}_p^{l-1} \copyright \mathbf{H}_t^{l-1}\right), \\ \mathbf{H}_p^l = \mathcal{T}_{\text{dif-p}}\left(\mathbf{H}_{\text{fus}}\right) + \mathbf{H}_p^{l-1}, \\ \mathbf{H}_t^l = \mathcal{T}_{\text{dif-t}}\left(\mathbf{H}_{\text{fus}}\right) + \mathbf{H}_t^{l-1}, \end{cases} \tag{5}$$

where ©denotes the channel-wise concatenation, and the sizes of $\mathbf{H}_p^{l-1}$, $\mathbf{H}_t^{l-1}$, $\mathbf{H}_{\text{fus}}$, $\mathbf{H}_p^l$, and $\mathbf{H}_t^l$ are all $D \times M \times K \times T$.

4. Experiment

To evaluate the performance of our proposed Melody-GCN, we conduct extensive experiments on two real-world datasets of multimodal data, including a dataset we collected **US40**, and a commonly used dataset **ACL18** [19]. Both contain stock prices and media texts. In the following subsections, we will briefly introduce the experiment setup, followed by discussions on the experimental results and ablation studies. Due to the page limit, more details could be found in the supplementary material.

4.1. Experimental setup

Datasets. As shown in Table 1, the **US40** dataset consists of price sequences of 40 stocks across four industries (10 stocks for each industry) with a sample period from 2010.05.01 to 2020.04.30 (2518 trading days in total). The corresponding media texts were collected from the headlines of daily news (34102 feeds in total). We use 30 days of historical data to forecast the next day's stock price movements. The commonly used dataset **ACL18** [19] has a much shorter sample period than our collected one, but is larger in the number of cross-section and media text feeds. It contains only 504 trading days in time series, ranging from 2014.01.01 to 2015.12.31. As a result, the authors use only 4 days of historical data to formulate the forecasts of the next day's stock price movements, and we follow this setup. 70 stocks across seven industries (again, 10 stocks for each industry) are used for the task[5] To construct a balanced panel, we drop stocks with less than 504 days of data and industries with less than 10 stocks. The selection criterion is the daily trading volumes, which are indicative of tradability and execution liquidity. The media texts were collected from the tweets, amounting to 87045 feeds.

Pre-processing Procedures. There are four prices for each trading day in both datasets, *i.e.*, open, high, low, and close prices. We convert the non-stationary prices into stationary returns for modelling, and also construct technical indicators out of the close prices as the price feature inputs, *i.e.*, Moving Average (MA), Standard Deviation (STD), Lower Band (LB), Upper Band (UB), EMA (Exponential Moving Average), Moving Average Convergence Divergence (MACD), and Short-Term Reversal (STR), the calculation details are demonstrated in Table 2. On

[5] The original dataset comprises 88 stocks across nine industries. The industry-based clusters are one-to-one mappings between stocks and industry categories. This is based on the de facto synergy effect concentrating in the stocks within the same industry that the interconnected stocks exhibit stronger co-movements inner-industry than inter-industry (chain effect).

Table 1
Descriptive statistics of the US40 and the ACL18 datasets.

Dataset	US40	ACL18
Country	U.S.	U.S.
Trading Days	2518	504
From	2010.05.01	2014.01.01
To	2020.04.30	2015.12.31
Train	2010.05.01–2018.04.30	2014.01.01–2015.07.31
Validate	2018.05.01–2019.04.30	2015.08.01–2015.09.30
Test	2019.05.01–2020.04.30	2015.10.01–2015.12.31
Industries (K)	4	7
Companies (M)	40	70
Texts	34 102	87 045
Formation Days (T)	30	4

Table 2
Feature calculations of technical analysis, where w denotes the window size, and p_i indicates the close price of day t.

Feature	Calculation
$\mathrm{MA}(w)_t$	$\frac{1}{w}\sum_{i=t-w+1}^{t} p_i$
$\mathrm{STD}(w)_t$	$\sqrt{\frac{1}{w-1}\sum_{i=t-w+1}^{t}\left(p_i-\mathrm{MA}(w)_t\right)^2}$
$\mathrm{LB}(w)_t$	$\mathrm{MA}(w)_t-\mathrm{STD}(w)_t^2$
$\mathrm{UB}(w)_t$	$\mathrm{MA}(w)_t+\mathrm{STD}(w)_t^2$
$\mathrm{EMA}(w)_t$	$\frac{2}{w+1}\cdot p_t+\frac{w-1}{w+1}\cdot\mathrm{EMA}(w)_{t-1}$
MACD_t	$\mathrm{EMA}(12)_t-\mathrm{EMA}(26)_t$
STR_t	$\ln(p_t-p_{t-1})$

the other hand, for media text data, we first collect a large number of raw financial news headlines, filter punctuation marks, and unimportant function words, and then filter out relevant news based on the keywords in the stock name dictionary. Next, news headline sentences are fed into the pre-trained BERT model, and the hidden features of the last layer are used as the encoding of news. After that, relevant news features are sorted according to the corresponding stocks and trading days. If there is no corresponding news for a certain industry, company, or trading day, we pad the sequence with zero. Finally, we get the preliminary textual inputs $\mathcal{T}$ with a size of $D_t \times M \times K \times T$.

Metrics & Baselines. We employ five key performance metrics for classification tasks, *i.e.*, Accuracy, Precision, Recall, F1-score, and Matthews Correlation Coefficient (MCC). In Particular, MCC is calculated as $\mathrm{MCC}=\frac{\mathrm{TP}\times\mathrm{TN}-\mathrm{FP}\times\mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}$. It indicates the correlation between the output classification and the ground truth categories, ranging from −1 to 1. In other words, the better the samples are classified, the closer the MCC is to 1, and vice versa, and if it returns 0, the model is no better than randomly guessing. We compare our proposed model with four categories of methods: (1) **Random Guessing**, *i.e.*, guess randomly according to the statistical frequencies of the ground truth label in the training set. (2) Conventional machine learning methods (ML), including **SVM**, Gaussian Naive Bayes (**Gaussian NB**), **Random Forest**, and **Ababoost**. (3) Deep learning-based models with only price data (*i.e.*, Single-Modal, SM), including **LSTM** [20], **EIIE** [5], **MGCGRU** [16], **GAT** [21], **RAT** [6] and **FactorVae** [22]. (4) Deep learning-based models with multimodal data (MM), including **EDLSTM** [13], **SARL** [23], and **DeepTrader** [24]. Since our experiments differ from those of the original publications, to run a fair horse race, we modify their published codes for this task with identical setups.

Implementation Details. When training our Melody-GCN on the US40 dataset, we apply a scale level of 3 to it with the hidden feature dimension D of {64, 96, 128}, and the temporal length of {30, 15, 5} from the finest level to the coarsest level, respectively. While on the ACL18 dataset, the scale level is set as 2, D are {96, 128}, and temporal length are {4, 2}. Besides, the dimension of the initial price data feature D_p and textual data feature D_t are 11 and 768, the dropout rate is 0.1, and the mini-batch size is 16. We train our method on an Nvidia GeForce RTX 3080 GPU for 500 epochs, using Adam optimizer with a learning rate of $1e-4$. The results are averaged over five runs of random seeds for all deep learning-based models.

4.2. Baseline comparisons

Classification Results.

Table 3 and Fig. 6 show performance comparisons among all competing methods. Our model generates superior or competitive results across all evaluation metrics on the two datasets.

As the random guessing ignores the semantic information of data, the percentages of the corresponding TN, FP, FN, and, TP in the confusion matrix are expected to be close to one quarter, and its MCC is approximately 0, as shown in Fig. 7. The performance of the traditional machine learning methods is no better than random guessing. By contrast, the deep learning-based methods significantly outperform "RG" and "ML" models, even a simple LSTM outperforms the best of them by a large margin. When multimodal data are leveraged, a similar model could obtain performance gains. For example, the ACC of EDLSTM gets an increase of 5.99% on the US40 dataset and 0.90% on the ACL18 dataset, compared with LSTM. As for GCN-based models, DeepTrader also yields much better results than MGCGRU, due to its complex and subtle model design. Meanwhile, our proposed approach gives the best prediction on both datasets in terms of ACC, and other comprehensive metrics, including F1-score and MCC.

In addition, we also conducted an in-depth analysis of the reasons why the other two evaluation metrics, Precision and Recall on the ACL18 dataset of Melody-GCN and the Recall on the US40 dataset did not achieve the best. First of all, in terms of the Recall, Melody-GCN was exceeded by EDLSTM and MGCGRU in the US40 dataset and ACL18 dataset, respectively. The reason is that they both tend to judge uncertain samples that are difficult to classify as positive, which leads to a higher FP rate and a lower FN rate, as shown in Fig. 7, which ultimately leads to the fact that although their Recall exceeds Melody-GCN, their Precision is far lower than the latter. In the same way, on the ACL18 dataset, DeepTrader tends to judge uncertain samples that are difficult to classify as negative, which results in a lower FP rate and a higher FN rate, as shown in Fig. 7(b). Therefore, although its Precision exceeds MM-GCN, its Recall lags by a large margin. The above analysis shows that there is a conflicting relationship between Precision and Recall to a certain extent, and improving Precision is likely to be at the cost of reducing Recall. From this perspective, the truly comprehensive metrics that can more comprehensively represent the performance of the model are the other three ones, and the Melody-GCN we proposed achieves the best in other metrics, further confirming its effectiveness and superiority.

Trading Simulation. To study the profitability of four deep-learning-based multimodal methods, we conduct the "only long" trading strategy day by day (*i.e.*, buy those which are predicted to have the top-k largest rising probability with equal weights then sell them on the next day). We evaluate the performance using Annualized Return (AR), Annualized Volatility (AV), Sharp Ratio (SR), Maximum Drawdown (MDD), and Calmar Ratio (CR), please refer to the supplementary material for calculation details. Table 4 shows that our model produces the best SR, MDD, and CR. Even though our model does not obtain the best AR, AR should be evaluated against VR, i.e., the risk-adjusted return SR. It is clear that although our model is not a multi-tasking design with multiple risk objectives, it still offers the best comprehensive portfolio choice, which, to some extent, alleviates the safety (risk) concerns. For more aspects related to the model safety (risk), please see the section of Limitation and Future Work.

In-depth Analysis of Model Performance: The above comparative results validate the advantages of the proposed method from an experimental perspective. In order to further explore what potential features the model has learned, especially whether the dynamic STGCL can effectively capture inter-industry, inter-company, and inter-day feature

Table 3
Comparisons of average scores with baselines on two datasets. All of the baselines except "Random Guessing" can be divided into three categories, that "ML" means machine learning, while "SM" and "MM" stands for the single-modal and multimodal respectively. Note that "↑" means the larger the better, the best results are highlighted in **bold**. The number after ± represents the variance of the five runs result, showing the robustness of the model.

		US40					ACL18				
		ACC ↑	Precision ↑	Recall ↑	F1 ↑	MCC ↑	ACC ↑	Precision ↑	Recall ↑	F1 ↑	MCC ↑
	Random Guessing	0.5057	0.5109	0.5191	0.5150	0.0111	0.5130	0.5045	0.5296	0.5153	0.0249
ML	SVM	0.5089	0.5095	0.7642	0.6114	0.0143	0.5045	0.4917	0.5774	0.5190	0.0003
	Gaussian NB	0.5011	0.5048	0.6844	0.5810	−0.0019	0.5147	0.5018	0.5917	0.5330	0.0251
	Random Forest	0.4901	0.4958	0.5158	0.5056	−0.0203	0.5119	0.5027	0.5335	0.5134	0.0271
	Adaboost	0.5031	0.5069	0.6267	0.5604	0.0036	0.5109	0.5028	0.5500	0.5207	0.0217
SM	LSTM	$0.7423_{\pm 0.0149}$	$0.7400_{\pm 0.0157}$	$0.7598_{\pm 0.0149}$	$0.7485_{\pm 0.0162}$	$0.4846_{\pm 0.0114}$	$0.7088_{\pm 0.0177}$	$0.7245_{\pm 0.0159}$	$0.6619_{\pm 0.0143}$	$0.6865_{\pm 0.0141}$	$0.4214_{\pm 0.0096}$
	FactorVae	$0.7635_{\pm 0.0324}$	$0.7843_{\pm 0.0326}$	$0.7531_{\pm 0.0298}$	$0.7684_{\pm 0.0301}$	$0.5274_{\pm 0.0204}$	$0.7203_{\pm 0.0297}$	$0.6928_{\pm 0.0263}$	$0.7453_{\pm 0.0284}$	$0.7181_{\pm 0.0242}$	$0.4426_{\pm 0.0176}$
	EIIE	$0.7735_{\pm 0.0212}$	$0.7511_{\pm 0.0184}$	$0.8421_{\pm 0.0192}$	$0.7907_{\pm 0.0201}$	$0.5544_{\pm 0.0137}$	$0.7185_{\pm 0.0175}$	$0.7524_{\pm 0.0162}$	$0.6486_{\pm 0.0184}$	$0.6905_{\pm 0.0155}$	$0.4430_{\pm 0.0124}$
	MGCGRU	$0.7905_{\pm 0.0247}$	$0.7810_{\pm 0.0265}$	$0.8191_{\pm 0.0224}$	$0.7975_{\pm 0.0243}$	$0.5831_{\pm 0.0159}$	$0.7150_{\pm 0.0217}$	$0.6726_{\pm 0.0223}$	$\mathbf{0.7726}_{\pm 0.0208}$	$0.7191_{\pm 0.0194}$	$0.4370_{\pm 0.0112}$
	GAT	$0.7963_{\pm 0.0261}$	$0.7771_{\pm 0.0243}$	$0.8162_{\pm 0.0274}$	$0.7962_{\pm 0.0255}$	$0.5936_{\pm 0.0202}$	$0.7297_{\pm 0.0196}$	$0.7359_{\pm 0.0206}$	$0.7112_{\pm 0.0184}$	$0.7233_{\pm 0.0175}$	$0.4595_{\pm 0.0146}$
	RAT	$0.8023_{\pm 0.0297}$	$0.8066_{\pm 0.0286}$	$0.8069_{\pm 0.0245}$	$0.8054_{\pm 0.0243}$	$0.6059_{\pm 0.0215}$	$0.7311_{\pm 0.0218}$	$0.7619_{\pm 0.0234}$	$0.6795_{\pm 0.0266}$	$0.7183_{\pm 0.0212}$	$0.4656_{\pm 0.0196}$
MM	EDLSTM	$0.7868_{\pm 0.0231}$	$0.7083_{\pm 0.0248}$	$\mathbf{0.9833}_{\pm 0.0067}$	$0.8234_{\pm 0.0231}$	$0.6229_{\pm 0.0194}$	$0.7152_{\pm 0.0207}$	$0.7303_{\pm 0.0193}$	$0.6681_{\pm 0.0187}$	$0.6978_{\pm 0.0173}$	$0.4311_{\pm 0.0152}$
	SARL	$0.7968_{\pm 0.0304}$	$0.7422_{\pm 0.0295}$	$0.9163_{\pm 0.0286}$	$0.8201_{\pm 0.0273}$	$0.6101_{\pm 0.0184}$	$0.7233_{\pm 0.0281}$	$0.7206_{\pm 0.0273}$	$0.715_{\pm 0.0285}$	$0.7182_{\pm 0.0243}$	$0.4464_{\pm 0.0127}$
	DeepTrader	$0.8395_{\pm 0.0276}$	$0.8377_{\pm 0.0263}$	$0.8465_{\pm 0.0265}$	$0.8421_{\pm 0.0244}$	$0.6791_{\pm 0.0186}$	$0.7416_{\pm 0.0237}$	$\mathbf{0.8001}_{\pm 0.0221}$	$0.6332_{\pm 0.0196}$	$0.7069_{\pm 0.0204}$	$0.4920_{\pm 0.0153}$
	Melody-GCN (ours)	$\mathbf{0.8661}_{\pm 0.0277}$	$\mathbf{0.8644}_{\pm 0.0286}$	$0.8768_{\pm 0.0282}$	$\mathbf{0.8692}_{\pm 0.0247}$	$\mathbf{0.7334}_{\pm 0.0218}$	$\mathbf{0.7585}_{\pm 0.0245}$	$0.7605_{\pm 0.0244}$	$0.7435_{\pm 0.0208}$	$\mathbf{0.7519}_{\pm 0.0214}$	$\mathbf{0.5169}_{\pm 0.0182}$

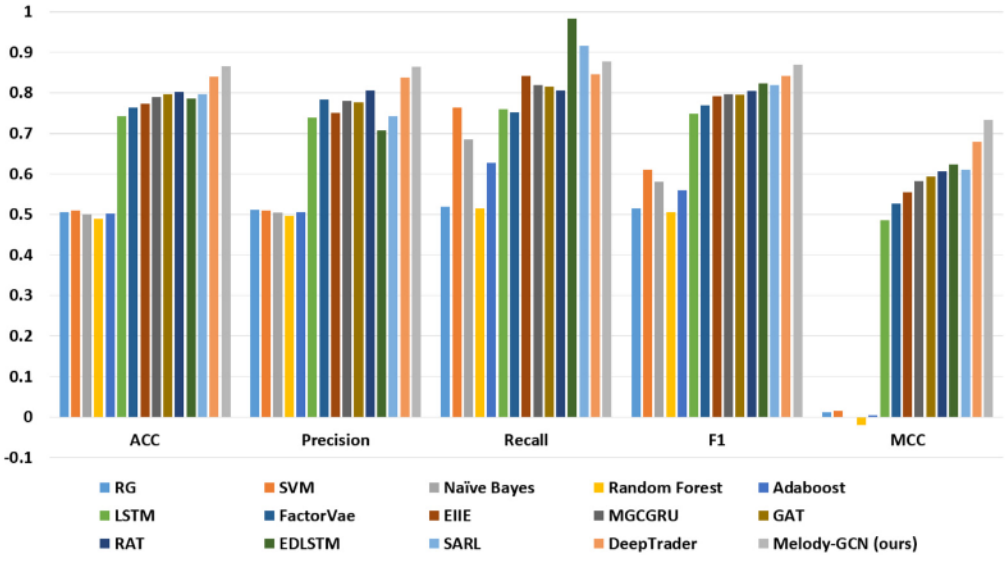


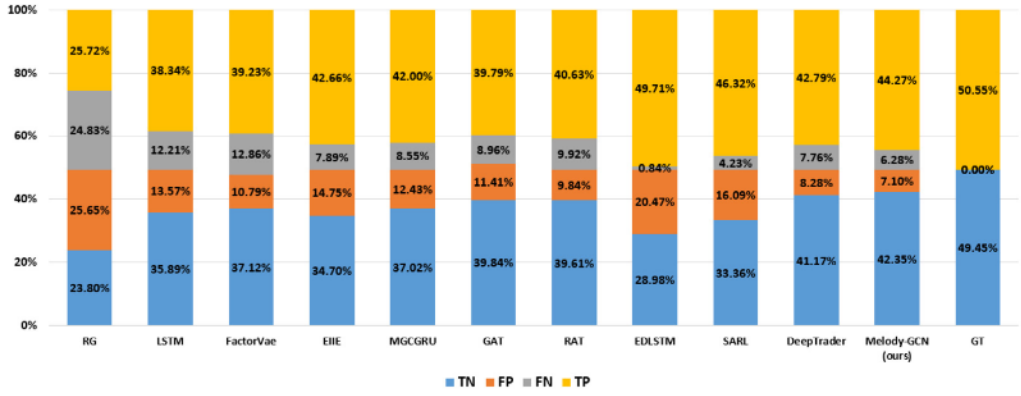
Fig. 6. The column chart of all results on the US40 dataset.

Table 4
The profitability results under the "only long" strategy (buy stocks with the top-k largest rising probability each day and sell them on the next day. The k is set 4 on the US40 dataset while 7 on the other, both 1/10 of the total stock number).

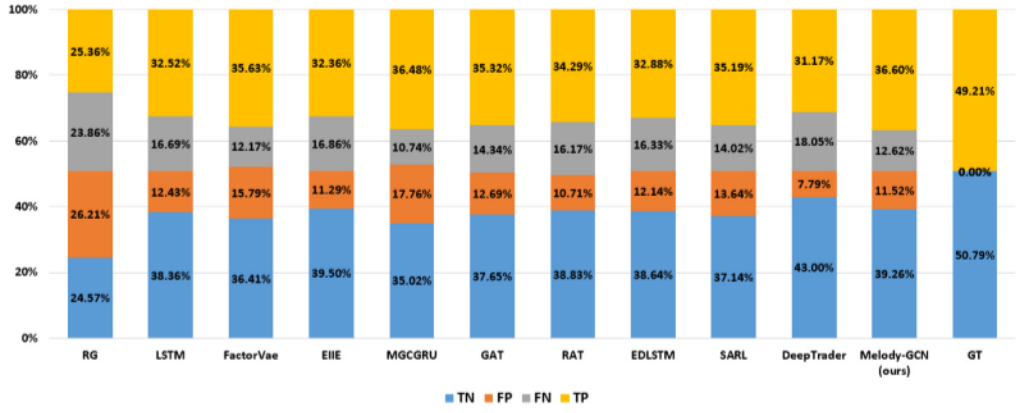
	US40					ACL18				
	AR (%) ↑	AV (%) ↓	SR ↑	MDD ↓	CR ↑	AR (%) ↑	AV (%) ↓	SR ↑	MDD ↓	CR ↑
LSTM	0.9153	11.6056	0.0789	0.7671	1.3155	1.8408	6.9014	0.2667	0.1899	5.3628
FactorVae	5.4376	13.6374	0.4735	0.5132	2.0545	4.7475	7.5143	0.6348	0.1638	6.3948
EIIE	6.0097	12.4112	0.4842	0.4412	2.4027	4.6345	7.6713	0.6041	0.1407	7.4367
MGCGRU	8.0036	15.6229	0.5123	0.3042	3.5504	7.8315	9.1410	0.8567	0.0741	14.5521
GAT	12.7645	17.5422	0.7324	0.0982	11.4831	8.7462	8.7691	0.9913	0.0824	13.1973
RAT	14.3637	19.3249	0.7433	0.0931	12.2839	9.1367	8.8407	1.0335	0.0736	14.8283
EDLSTM	12.4073	20.5435	0.6040	0.1001	11.2295	8.9632	8.9088	1.0061	0.0724	15.0501
SARL	14.4431	20.4063	0.7078	0.0884	12.9460	9.3338	9.6657	0.9656	0.1011	10.8144
DeepTrader	**15.1839**	19.5199	0.7779	0.0974	11.8258	**9.8628**	8.9040	1.1077	0.0856	12.8344
Melody-GCN (ours)	14.5136	**10.7399**	**1.3513**	**0.0825**	**13.8804**	9.0388	**6.8851**	**1.3128**	**0.0716**	**15.2288**

dependencies, here we visualize the learned adjacency matrices $\mathbf{A}_{\text{ind}}$, $\mathbf{A}_{\text{com}}$, and $\mathbf{A}_{\text{day}}$ of the STGCL at the end of the Melody-GCN.

Robustness Checks. We train and test our proposed model together with other baselines five times in the experiment. The results are averaged over five runs of random seeds for all deep learning-based models. One way to examine model robustness is to check the variance of the results in different runs. In Table 3, the number after ± represents the variance of the results from five runs, indicating the robustness of our proposed model. As we can see, compared with other models, the variance of the performance metrics of our model is about the same as other deep learning-based models. Specifically, for our model, the variances of all metrics divided by their means have an average of 2.79% across two datasets. This is evident that the performance of our model does not fluctuate significantly and exhibits notable robustness. Another way to ensure model robustness is to test whether or not the model survives changing or even unprecedented market conditions. As

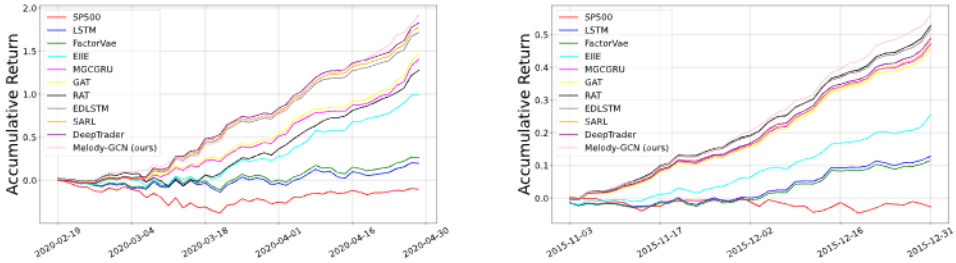


(a) US40



(b) ACL18

Fig. 7. The mosaic diagram of the classification results (in percentage) on the two datasets. Note that "RG" means "Random Guessing", and "GT" denotes Ground Truth.



(a) US40 (b) ACL18

Fig. 8. The accumulated returns (scaled to the same risk level in terms of volatility) of all models during the S&P500 Index crash and wobbling periods in the test data.

Table 5
The comparison of SR, MDD, and CR across all models during the S&P500 crash and wobbling periods in test data.

	US40			ACL18		
	SR ↑	MDD ↓	CR ↑	SR ↑	MDD ↓	CR ↑
LSTM	0.0738	0.7569	0.0076	0.5961	0.2319	0.1340
FactorVae	0.3905	0.6239	0.0588	0.5075	0.2828	0.0877
EIIE	0.8772	0.2453	1.1603	1.0268	0.0673	1.2626
MGCGRU	1.0655	0.0797	5.4729	1.36988	0.0056	35.2795
GAT	1.0873	**0.0686**	6.7674	1.3798	0.0062	30.9140
RAT	0.9158	0.1525	2.3597	1.3784	0.0053	42.6603
EDLSTM	1.1670	0.1281	5.0212	1.3714	0.0047	47.319
SARL	1.1781	0.1152	5.8195	1.3708	0.0050	39.7484
DeepTrader	1.1893	0.1271	5.4776	1.3808	0.0044	46.5825
Melody-GCN (ours)	**1.1931**	0.0968	**7.1859**	**1.3816**	**0.0039**	**60.2015**

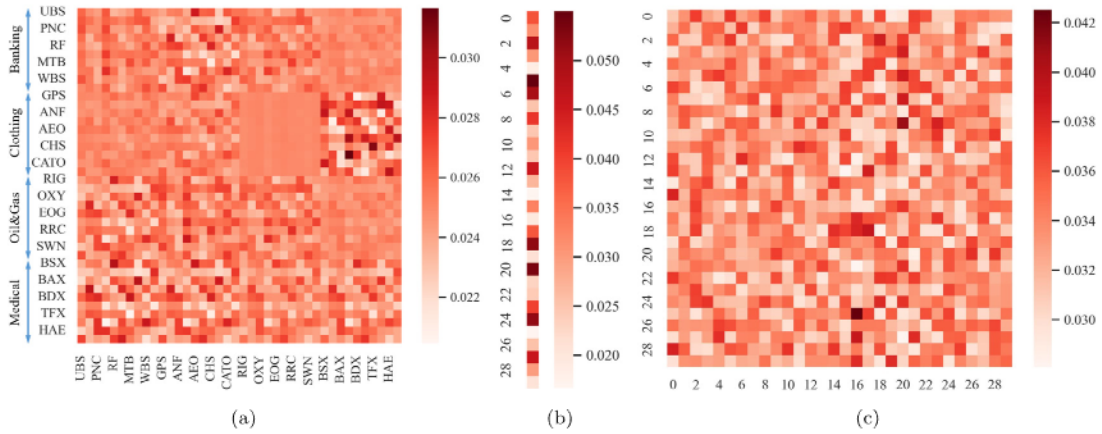


Fig. 9. Heatmap of the learned adjacency matrix from the STGCL at the end of the Melody-GCN. Sub-figure (a) shows the spatial corrections learned by the inter-industry and inter-company graph convolution layers, and Sub-figure (b) and (c) represent temporal dependencies learned by inter-day graph convolution layers.

both datasets used in this paper belong to the U.S. market, the market portfolio S&P500 Index is an ideal indicator of the market state widely used as a benchmark. For the US40 dataset, S&P500 suffered a huge loss from 2020/02/20 in the test data, and for the ACL18 dataset, S&P500 started to wobble from 2015/11/03 in the test data. These types of market behaviour are often regarded as the dramatic switches of market states. In particular, the market crash due to the outbreak of COVID-19 was an extreme event with huge impacts on the global economic and financial system that is unseen to the models given the training data. Thus, it is worth examining our model performance against the baseline models and S&P500 Index during these periods. As shown in Fig. 8, our model achieves better performance in terms of accumulated portfolio returns (scaled to the same risk level in terms of volatility) in both crash and wobbling periods of the S&P 500 Index. Moreover, we conduct a performance comparison on multiple risk metrics, i.e., SR, MDD, and CR, during both crash and wobbling periods in Table 5. Our model still achieves the best performance in terms of two risk-adjusted metrics, i.e., SR and CR, across two datasets. Although it does not achieve the best outcome in terms of MDD in the US40 sample, the MDD should be compared against the annualized return (AR), i.e., the CR metric, which means that the MDD is best covered by the AR generated by our model compared to all baselines. For more aspects related to the model robustness, please see the section of Limitation and Future Work.

The leftmost sub-figure (a) of Fig. 9 represents the spatial dependencies learned by the classifier at the end of Melody-GCN. Note that the US40 dataset has 4 industries, each containing 10 companies. That is to say, the inter-industry and inter-company spatial correlations correspond to two matrices with a size of 4×4 and 10×10, respectively, and we integrate the two into a unified 40×40 matrix. It can be seen that the 10 companies in the banking industry are more affected by their own industry, clothing industry, and oil and gas industry, while the impact of the medical industry is less. At the same time, the clothing industry is more affected by the medical industry, while the oil and gas industry is the least affected by the medical industry. It turns out that the inter-industry graph convolution and inter-company graph convolution within the classifier of Melody-GCN have indeed learned spatial semantic information. The middle sub-figure (b) represents the temporal correlations learned by the inter-day graph convolution in the classifier. Since the predictor maps the hidden features of the past 30 days to the fluctuation probability of the next 1 day, the size of the matrix is 30×1. It can be seen that the features at different time points in the historical sequence have different degrees of impact on future price movement forecasts. In addition, in order to further explore the interaction between all time points of the observed sequence, we show the adjacency matrix accumulated from the previous inter-day graph convolution next to the classifier. As shown in the far right sub-figure (c), it can be seen that the model could assign different attention weights to different time points.

4.3. Ablation study

To validate the design components of our proposed approach, we conduct several ablation studies and give an in-depth analysis of five critical elements as follows. The results in Table 6 and Fig. 10 are obtained by training 200 epochs on the ACL18 dataset.

Number of Scales. We conduct experiments with one, two, three, and four temporal scale levels, denoted as "L1", "L2", "L3", and "L4" respectively. It shows the increase of the scale level brings about better performance to some degree, but when the scale levels increase to four, the ACC drops back, due to the excessive redundancy of parameters.

Modal Setting. We make stock movement predictions from only media text, only historical stock prices, and combined data of both modalities. It shows the "Text" variant results in the worst results, due to the sparsity of the media texts (according to Section 4.1, $34102/2518/4/10 \approx 0.3$ which means each company has only one related text in every three days on average). By contrast, the "Stock" setting causes a drop of only 0.025 in ACC, which means the textual data play an essential but less important role than the numerical price data.

Fusing Manner. By simply concatenating the media text and the price data together, and putting them holistically into the network ("Concat"), the ACC sharply decreases compared with fusing them through the specially designed MMFDB. Because the dense numerical price data and the sparse media texts are entirely heterogeneous, an ordinary model without any particular pre-processing module cannot capture the crucial and enhanced complementary features from the mixed ones.

Residual Connection. We discard all internal residual connections ("w/o Res") and find the ACC decreases than the default setting ("w/ Res"). It demonstrates that residual connections are of great importance to re-extract the complementary features and make learning more accessible.

Basic Operator. To validate the usefulness of our inter-industry, inter-company, and inter-day STGCL, we replace it with the depth-wise separable 2D convolution layer for spatial features accompanied with a linear layer for temporal features, denoted as "SepConv", and control the number of parameters within a similar scale, so as to ensure a fair comparison.

Table 6 shows that its ACC drops to 0.7938, even worse than that of the single-modal "Stock" in the second group, which shows the effectiveness of our STGCL.

In summary, all components of our Melody-GCN contribute to the overall superior results, and the multimodal setting and the MMFDB module play the most crucial roles compared with other elements.

Table 6
Results from the five groups of ablation studies on the US40 dataset. Note that "∗" indicates the default choices.

	① Number of scale level				②Modal setting			③Fusing manner		④Residual connection		⑤Basic operator	
	L1	L2	L3*	L4	Text	Stock	Both*	Concat	MMFDB*	w/o Res	w/ Res*	SepConv	STGCB*
ACC ↑	0.7559	0.7769	**0.8339**	0.7256	0.5337	0.8089	**0.8339**	0.5377	**0.8339**	0.7506	**0.8339**	0.7938	**0.8339**
Precision ↑	0.7676	0.7724	**0.8336**	0.7308	0.5441	0.8123	**0.8336**	0.5412	**0.8336**	0.7523	**0.8336**	0.7557	**0.8336**
Recall ↑	0.7437	0.7977	**0.8432**	0.7375	0.4746	0.8107	**0.8432**	0.5616	**0.8432**	0.7567	**0.8432**	**0.8749**	0.8432
F1 ↑	0.7537	0.7830	**0.8369**	0.7313	0.5049	0.8097	**0.8369**	0.5488	**0.8369**	0.7536	**0.8369**	0.8110	**0.8369**
MCC ↑	0.5139	0.5557	**0.6689**	0.4526	0.0682	0.6193	**0.6689**	0.0694	**0.6689**	0.5010	**0.6689**	0.5945	**0.6689**

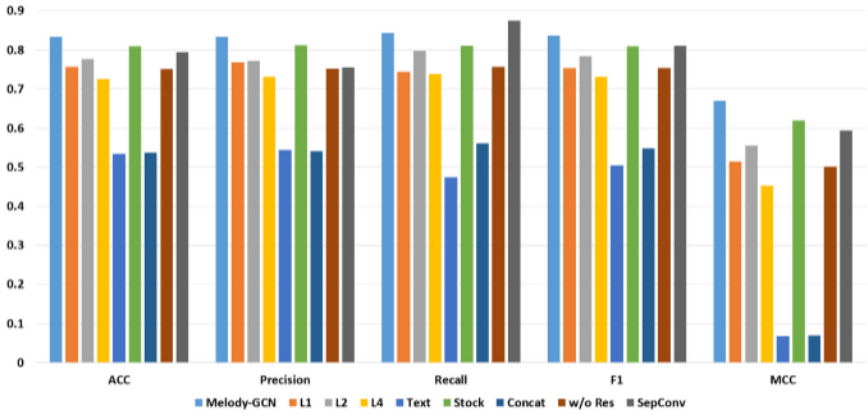


Fig. 10. Visualization of the experimental results from various ablative variants by training 200 epochs on the US40 dataset.

Table 7
Model complexity and time-consuming. The unit of the number model parameters is millions (M) and the unit of time consumption per epoch is seconds (S).

	Number of model parameters (M)	The time consumption per epoch (S)
LSTM	0.3136	8.2
FactorVae	1.1754	34.5
EIIE	2.6162	12.4
MGCGRU	1.7476	23.2
GAT	0.0481	6.7
RAT	0.0181	21.3
EDLSTM	0.0074	18.9
SARL	0.0073	7.5
DeepTrader	0.0117	19.5
Melody-GCN (ours)	2.5657	29.1

4.4. Complexity analysis

We compare our proposed model with the baselines in terms of complexity and time consumption. The baseline models are parameterized according to implementation details in respective papers. As shown in Table 7, in general, our proposed model has a larger number of model parameters and higher time consumption per epoch than the baselines, as our proposed model has more parameters and a deeper network to extract the features from the stock price-related numerical and the media text data. However, it performs much better in the task of stock price prediction. Therefore, our proposed model has made a good trade-off between model performance and complexity.

4.5. Limitations and future work

Our proposed model progressively extracts and aligns the numerical and textual features via a fine-to-coarse descending path and a coarse-to-fine ascending path. Moreover, we also leverage the interrelations among stocks to learn the complex and evolving relations not only across industries and individual companies but also across time horizons. However, there are still some limitations of our model as follows.

Interpretability: Even though it explicitly considers the inter-industry and inner-industry relations in modelling the dynamic sequential patterns and stock relations across both numeric and text modalities, some relevant techniques can be incorporated in order to improve the interpretability of the model. In particular, more and more recent studies focus on enhancing the interpretability of the stock price predictability, e.g., Shi et al. [25] carefully designs a deep neural network (DNN) architecture with hierarchical factorization; Deng et al. [26] resort to the external knowledge from the financial domain using event embeddings; Yang et al. [27] employ multiple sub-additive DNNs with each of them capturing only one particular predictive mechanism; Yun et al. [28] propose a feature subset selection procedure to assess the feature-importance interpretability; Deng et al. [29] rely on causal inference from the financial news graph; among many others. Moreover, the white-box model that achieves full mathematical interpretability via sparse rate reduction proposed in [30] provides an ideal framework.

Robustness: Besides the aspects discussed in the section of Robustness Checks, Generative Adversarial Networks (GAN) can be a powerful tool given the limited data availability in financial markets. A trade-off between adversarial robustness and predictive accuracy can be made [31]. Simulations and training via GAN can produce more robust sequential predictions that are not confined to be deterministic [32]

in various finance applications, e.g., volatility modelling [33], exotic options pricing [34], *etc*.. Similarly, learning feature representations in financial markets that are characterized by low signal-to-noise ratio via probabilistic models with inherent randomness, such as Variational Autoencoder (VAE) and diffusion models, can also improve model robustness [35,36], e.g., for stock return prediction task, Duan et al. [22] propose VAE-based dynamic factor model (which our model outperforms) and Koa et al. [37] incorporate a diffusion denoising process into VAE, both achieve SOTA performance.

Safety (Risk): For stock price or trend prediction as a classification task, Accuracy, Precision, Recall, F1-score, and MCC are often used as comprehensive evaluation metrics of model performance. However, these metrics ignore the risks of the trading programme associated with the predictions, i.e., the portfolio risks. From this perspective, a multi-tasking framework is recommended in order to take the risk-adjusted returns and downside risks into account, e.g., the recent work of Yang et al. [38] simultaneously considers stock return and financial risk in a multi-task forecasting setup. Although our proposed model also achieves the best results in the risk-adjusted return metrics (i.e., SR and CR), future work can be built analytically on a multi-objective model that aims to strike a balance among multiple loss functions by learning the gradient trade-offs [39], e.g., in predictive accuracy and risk-adjusted reward metrics, with a weighted or ordered multi-objective setup [40].

Given the limitations and inspiring recent literature highlighted above, in our future work, we aim to design an interpretable framework for robust stock price prediction with multi-tasking safety concerns on balanced portfolio risks.

5. Conclusion

We have presented a novel multimodal multiscale spatio-temporal GCN-based framework for the stock price prediction task. Our method leverages the media texts along with the stock price data for more valuable predictive information, and further integrates and aligns the multimodal feature by the multimodal fusing-diffusing blocks. Besides, the multiscale architecture fully distills and reconstructs the hidden features, and the adaptive spatio-temporal graph convolutional layers extract the cross-sectional relations as well as the sequential dependencies dynamically. Our proposed approach outperforms other state-of-the-art models on two real-world multimodal datasets, and the influences of its components are analysed in-depth and proved to contribute to the overall superior performance. However, there are still some limitations of our proposed model that we aim to tackle in the research agenda, e.g., introducing an extra robustness module, building on an interpretable multi-tasking framework with identified predictive mechanisms and multiple weighted or ordered safety (risk) objectives.

CRediT authorship contribution statement

Ruirui Liu: Methodology, Project administration, Supervision, Writing – original draft. **Haoxian Liu:** Investigation, Methodology, Validation, Visualization, Writing – original draft, Writing – review & editing. **Huichou Huang:** Investigation, Methodology, Writing – original draft, Writing – review & editing. **Bo Song:** Data curation, Investigation, Methodology, Writing – original draft. **Qingyao Wu:** Methodology, Project administration, Writing – original draft.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Qingyao Wu reports financial support was provided by National Natural Science Foundation of China (NSFC). Qingyao Wu reports financial support was provided by Guangdong Basic and Applied Basic Research Foundation. Qingyao Wu reports financial support was provided by the Qatar Centre for Global Banking and Finance of King' Business School, King's College London.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by National Natural Science Foundation of China (NSFC) 62272172, Guangdong Basic and Applied Basic Research Foundation (2023A1515012920). Ruiriu Liu also thanks the financial grant on big-data analytics and machine learning from the Qatar Centre for Global Banking and Finance of King' Business School, King's College London.

References

[1] J. Yoo, Y. Soun, Y.-c. Park, U. Kang, Accurate multivariate stock movement prediction via data-axis transformer with multi-level contexts, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 2037–2045.

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

[3] L. Zhao, W. Li, R. Bao, K. Harimoto, Y. Wu, X. Sun, Long-term, short-term and sudden event: Trading volume movement prediction with graph-based multi-view modeling, in: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Montreal, Canada, August 21-26, 2021, 2021.

[4] R. Sawhney, S. Agarwal, A. Wadhwa, 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.

[5] Z. Jiang, D. Xu, J. Liang, A deep reinforcement learning framework for the financial portfolio management problem, 2017, arXiv preprint arXiv:1706.10059.

[6] K. Xu, Y. Zhang, D. Ye, P. Zhao, M. Tan, Relation-aware transformer for portfolio policy learning, in: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, IJCAI, 2021, pp. 4647–4653.

[7] Q. Ding, S. Wu, H. Sun, J. Guo, J. Guo, Hierarchical multi-scale Gaussian transformer for stock movement prediction, in: IJCAI, 2020, pp. 4640–4646.

[8] G. Liu, Y. Mao, Q. Sun, H. Huang, W. Gao, X. Li, J. Shen, R. Li, X. Wang, Multi-scale two-way deep neural network for stock trend prediction, in: IJCAI, 2020, pp. 4555–4561.

[9] Q. Liu, X. Cheng, S. Su, S. Zhu, Hierarchical complementary attention network for predicting stock price movements with news, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018, pp. 1603–1606.

[10] L. Zhang, C. Aggarwal, G.-J. 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, 2017, pp. 2141–2149.

[11] W. Zhang, N. Zhang, J. Yan, G. Li, X. Yang, Auto tuning of price prediction models for high-frequency trading via reinforcement learning, Pattern Recognit. 125 (2022) 108543, http://dx.doi.org/10.1016/j.patcog.2022.108543, URL https://www.sciencedirect.com/science/article/pii/S0031320322000243.

[12] X. Ding, Y. Zhang, T. Liu, J. Duan, Deep learning for event-driven stock prediction, in: Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.

[13] Q. Li, J. Tan, J. Wang, H. Chen, A multimodal event-driven lstm model for stock prediction using online news, IEEE Trans. Knowl. Data Eng. 33 (10) (2020) 3323–3337.

[14] D. Cao, Y. Wang, J. Duan, C. Zhang, X. Zhu, C. Huang, Y. Tong, B. Xu, J. Bai, J. Tong, et al., Spectral temporal graph neural network for multivariate time-series forecasting, in: Advances in Neural Information Processing Systems, vol.33, 2020, pp. 17766–17778.

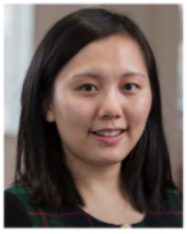
[15] W. Li, R. Bao, K. Harimoto, D. Chen, J. Xu, Q. Su, Modeling the stock relation with graph network for overnight stock movement prediction, in: IJCAI, vol. 20, 2020, pp. 4541–4547.

[16] J. Ye, J. Zhao, K. Ye, C. Xu, Multi-graph convolutional network for relationship-driven stock movement prediction, in: 2020 25th International Conference on Pattern Recognition, ICPR, IEEE, 2021, pp. 6702–6709.

[17] T. Yin, C. Liu, F. Ding, Z. Feng, B. Yuan, N. Zhang, Graph-based stock correlation and prediction for high-frequency trading systems, Pattern Recognit. 122 (2022) 108209, http://dx.doi.org/10.1016/j.patcog.2021.108209, URL https://www.sciencedirect.com/science/article/pii/S0031320321003903.

[18] D. Cheng, F. Yang, S. Xiang, J. Liu, Financial time series forecasting with multi-modality graph neural network, Pattern Recognit. 121 (2022) 108218, http://dx.doi.org/10.1016/j.patcog.2021.108218, URL https://www.sciencedirect.com/science/article/pii/S003132032100399X.

[19] Y. Xu, 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.
[20] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780, http://dx.doi.org/10.1162/neco.1997.9.8.1735.
[21] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, in: International Conference on Learning Representations, 2018.
[22] Y. Duan, L. Wang, Q. Zhang, J. Li, Factorvae: A probabilistic dynamic factor model based on variational autoencoder for predicting cross-sectional stock returns, in: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, no. 4, 2022, pp. 4468–4476.
[23] Y. Ye, H. Pei, B. Wang, P.-Y. Chen, Y. Zhu, J. Xiao, B. Li, Reinforcement-learning based portfolio management with augmented asset movement prediction states, in: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, no. 01, 2020, pp. 1112–1119.
[24] Z. Wang, B. Huang, S. Tu, K. Zhang, L. Xu, DeepTrader: A deep reinforcement learning approach for risk-return balanced portfolio management with market conditions embedding, in: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, no. 1, 2021, pp. 643–650.
[25] L. Shi, Z. Teng, L. Wang, Y. Zhang, A. Binder, DeepClue: visual interpretation of text-based deep stock prediction, IEEE Trans. Knowl. Data Eng. 31 (6) (2018) 1094–1108.
[26] S. Deng, N. Zhang, W. Zhang, J. Chen, J.Z. Pan, H. Chen, Knowledge-driven stock trend prediction and explanation via temporal convolutional network, in: Companion Proceedings of the 2019 World Wide Web Conference, 2019, pp. 678–685.
[27] Z. Yang, A. Zhang, A. Sudjianto, GAMI-Net: An explainable neural network based on generalized additive models with structured interactions, Pattern Recognit. 120 (2021) 108192.
[28] K.K. Yun, S.W. Yoon, D. Won, Interpretable stock price forecasting model using genetic algorithm-machine learning regressions and best feature subset selection, Expert Syst. Appl. 213 (2023) 118803.
[29] Y. Deng, Y. Liang, S.-M. Yiu, Towards interpretable stock trend prediction through causal inference, Expert Syst. Appl. 238 (2024) 121654.
[30] Y. Yu, S. Buchanan, D. Pai, T. Chu, Z. Wu, S. Tong, B.D. Haeffele, Y. Ma, White-box transformers via sparse rate reduction, 2023, arXiv preprint arXiv:2306.01129.
[31] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, M. Jordan, Theoretically principled trade-off between robustness and accuracy, in: International Conference on Machine Learning, PMLR, 2019, pp. 7472–7482.
[32] J. Yoon, D. Jarrett, M. Van der Schaar, Time-series generative adversarial networks, in: Advances in Neural Information Processing Systems, vol.32, 2019.
[33] M. Wiese, R. Knobloch, R. Korn, P. Kretschmer, Quant GANs: Deep generation of financial time series, Quant. Finance 20 (9) (2020) 1419–1440.
[34] H. Jang, J. Lee, Generative Bayesian neural network model for risk-neutral pricing of American index options, Quant. Finance 19 (4) (2019) 587–603.
[35] A. Camuto, M. Willetts, S. Roberts, C. Holmes, T. Rainforth, Towards a theoretical understanding of the robustness of variational autoencoders, in: A. Banerjee, K. Fukumizu (Eds.), Proceedings of the 24th International Conference on Artificial Intelligence and Statistics, in: Proceedings of Machine Learning Research, vol.130, PMLR, 2021, pp. 3565–3573.
[36] C. Xiao, Z. Chen, K. Jin, J. Wang, W. Nie, M. Liu, A. Anandkumar, B. Li, D. Song, Densepure: Understanding diffusion models for adversarial robustness, in: The Eleventh International Conference on Learning Representations, 2022.
[37] K.J. Koa, Y. Ma, R. Ng, T.-S. Chua, Diffusion variational autoencoder for tackling stochasticity in multi-step regression stock price prediction, in: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, 2023, pp. 1087–1096.
[38] L. Yang, J. Li, R. Dong, Y. Zhang, B. Smyth, NumHTML: Numeric-oriented hierarchical transformer model for multi-task financial forecasting, in: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, no. 10, 2022, pp. 11604–11612.
[39] H. Chai, J. Cui, Y. Wang, M. Zhang, B. Fang, Q. Liao, Improving gradient trade-offs between tasks in multi-task text classification, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2023, pp. 2565–2579.
[40] E.M. Hahn, M. Perez, S. Schewe, F. Somenzi, A. Trivedi, D. Wojtczak, Multi-objective omega-regular reinforcement learning, Form. Asp. Comput. (2023).



Ruirui Liu is currently an Assistant Professor at Brunel University London, and a research fellow and visiting lecturer at King's College London, United Kingdom. She received her B.A., M.Sc., and Ph.D. degrees from China University of Mining and Technology, University of Glasgow, and King's College London in 2014, 2015, and 2021, respectively. Her research interests include computational finance, natural language processing, machine learning, and statistical learning.

Haoxian Liu is currently a master student at the School of Software Engineering, South China University of Technology, China. He received his bachelor's degree from the School of Software Engineering, South China University of Technology in 2021. His research interests include quantitative finance and data mining.

Huichou Huang is currently a research affiliate with the Global Research Unit at City University of Hong Kong, and was a visiting scholar at Duke University, Humboldt University of Berlin, and Washington University in St. Louis. He received his B.Sc., M.Sc., and Ph.D. degrees from Sun Yat-Sen University, University of Oxford, Bayes Business School of City UoL, and Adam Smith Business School of UoG in 2007, 2008, 2010, and 2015, respectively. His research interests include quantitative finance, computational statistics, and machine learning.

Bo Song received the B.E. degree from South China University of Technology, GuangZhou, China, in 2020. He is currently a master's degree candidate in South China University of Technology. His research interests include machine learning and quantitative finance.

Qingyao Wu received the B.S. degree in software engineering from the South China University of Technology, China, in 2007, and the Ph.D. degree in computer science from the Harbin Institute of Technology, China, in 2013. He is currently a Professor with the School of Software Engineering, South China University of Technology. His current research interests include computer vision and data mining.