Modeling the Momentum Spillover Effect for Stock Prediction via Attribute-Driven Graph Attention Networks

Rui Cheng, Qing Li*
Fintech Innovation Center
School of Economic Information Engineering
Southwestern University of Finance and Economics

Abstract

In finance, the momentum spillovers of listed firms is well acknowledged. Only few studies predicted the trend of one firm in terms of its relevant firms. A common strategy of the pilot work is to adopt graph convolution networks (GCNs) with some predefined firm relations. However, momentum spillovers are propagated via a variety of firm relations, of which the bridging importance varies with time. Restricting to several predefined relations inevitably makes noise and thus misleads stock predictions. In addition, traditional GCNs transfer and aggregate the peer influences without considering the states of both connected firms once a connection is built. Such non-attribute sensibility makes traditional GCNs inappropriate to deal with the attribute-sensitive momentum spillovers of listed firms wherein the abnormal price drop of one firm may not spill over if the trade volume of this decreasing price is small or the prices of the linked firms are undervalued. In this study, we propose an attribute-driven graph attention network (AD-GAT) to address both problems in modeling momentum spillovers. This is achieved by element-wisely multiplying the nonlinear transformation of the attributes of the connected firms with the attributes of the source firm to consider its attribute-sensitive momentum spillovers, and applying the unmasked attention mechanism to infer the general dynamic firm relation from observed market signals fused by a novel tensor-based feature extractor. Experiments on the three-year data of the S&P 500 demonstrate the superiority of the proposed framework over state-of-the-art algorithms, including GCN, eLSTM, and TGC.

1 Introduction

In stock markets, there are momentum spillovers among the relevant firms, wherein the past returns of one firm can predict the returns of firms that are linked to it (Ali and Hirshleifer 2020). Unfortunately, most studies on stock predictions in machine learning ignored the interferences from other firms with an assumption that the historical information of a stock determines its future (Li et al. 2018). With the advancement of graph neural networks (GNNs) (Scarselli et al. 2009), few researchers have been exploring the momentum spillover effect on stock prediction with graph convolutional networks (GCNs) (Chen, Wei, and Huang 2018; Feng et al. 2019; Li et al. 2020b). In these studies, each firm is treated as a node in the graph, and the edge between two nodes is determined by a predefined firm relation. To capture the momentum spillovers of listed firms, each node is represented by its relational embedding that is the aggregation of the attributes of its neighbors in the graph. However, traditional GNNs transfer and aggregate the peer influences without considering the states of both connected firms once a connection is built. In real stock markets, the influence propagation of linked firms is attribute-sensitive. For example, the abnormal price drop of one firm may not spill over if the trade volume of this decreasing price is small, or the prices of the linked firms are undervalued.

In addition, these studies rely on a set of predefined relations, such as shareholders (Chen, Wei, and Huang 2018), industry (Feng et al. 2019), or price comovement (Li et al. 2020b). However, the momentum spillover effect is led by a variety of inter-firm linkages of which the bridging importance varies with time. It is hard to assume one certain type of known firm relation is superior to others when they are applied for studying the momentum spillover effect. An alternative approach is to combine various predefined relations to form a dominant relation to study the momentum spillover effect. Even though many financial studies have been devoted to discovering the relations of listed firms, there are still a number of important relations to be explored (Ali and Hirshleifer 2020). Apparently, studying the momentum spillovers of listed firms with explicit relations is limited by the undiscovered relations and the efficient way to leverage the predefined relations. To solve both problems, we model the attribute-sensitive momentum spillover effect and estimate the latent relation of listed firms via a novel attribute-driven graph attention network (AD-GAT). Three unique contributions are proposed as follows:

- An attribute-mattered aggregator is proposed to capture the attribute-sensitive momentum spillovers of listed firms. This is achieved by element-wisely multiplying the nonlinear transformation of the connected firms' attributes with the attributes of the source firm.
- To discover the dominant relation for the momentum spillovers of listed firms, an unmasked attention mechanism is applied to infer the general dynamic firm relation from observed market signals.

*Corresponding author: liqing@swufe.edu.cn

- A novel tensor-based modeling is presented to capture multimodal market signals with feature interactions to provide solid ground truth for implicit inferring firm states and relations.

Experiments performed on three-year data of the stocks listed in the S&P 500 index demonstrate AD-GAT's superiority over state-of-the-art algorithms, including eLSTM (Li et al. 2020a) and TGC (Feng et al. 2019). Relative to these algorithms, the proposed approach achieves performance enhancements of at least 6.4% and 10.7% in terms of directional accuracy (DA) and AUC, respectively.

2 Related Work

The stock market is a dynamic system in which stocks are highly influenced by a variety of time-series market signals. Several RNN variants, including LSTM (Hochreiter and Schmidhuber 1997) and GRU (Cho et al. 2014), have been applied to generate the sequential embeddings for stock predictions, which capture the time dependencies in historical market signals. A common strategy that mingles different types of market factors in previous studies is to concatenate the market signals of different sources into a compound vector, which inevitably ignores their intrinsic association (Kolda and Bader 2009; Li et al. 2016). Some researchers have taken a further step by fusing different market factors along with their interactions. For instance, Ming et al. (2014) constructed a unified matrix to characterize the "co-movements" between stock prices and news articles. Li et al. (2016) modeled market information with tensors and proposed a support tensor machine to predict stock trends. Zhang et al. (2018) further constructed two auxiliary matrices, i.e., the stock quantitative feature matrix and the stock correlation matrix, to assist the tensor decomposition process. The way these works handling the interactions of different market factors is independent of stocks and downstream tasks (e.g., stock movement prediction). However, such assumptions are too ideal to be accepted by real stock markets. Here, we argue that the interactions of different market factors vary with stocks and should be modeled with a consideration of downstream tasks. In this study, we capture the interactions of market signals with tensors, which are learned toward downstream tasks and personalized for different stocks.

Essentially, the stock fluctuations of one firm are caused by its own market signals along with the interferences from its related firms. The momentum spillover effect is often ignored by previous studies on stock predictions due to the lack of an efficient way to incorporate the spillover relevance. Only few works on stock predictions have explored this effect (Chen, Wei, and Huang 2018; Feng et al. 2019; Li et al. 2020b). These studies relied on traditional graph convolutional networks (GCNs) in which each firm is represented as a node and each edge is built in terms of some predefined firm relations. For instance, Chen, Wei, and Huang (2018) constructed a graph in terms of firm investment and proposed a joint stock prediction model based on GCN to consider the influences of related stocks. Feng et al. (2019) applied two types of firm relation, i.e., industry category or Wikipedia linkage, to build a graph for stock predictions. Li et al. (2020b) studied the momentum spillover effect with GCNs built in terms of the co-movements of historical prices. These studies treated firm links in a static way by building a graph with the fixed predefined firm relations. However, firms are dynamically linked via a variety of relations of which the importance varies with time. One of the pilot studies on the dynamical linkage of listed firms is the work of Feng et al. (2019) which adjusts the predefined relations in the learning process. However, restricting firm relatedness to a particular type of predefined relation inevitably generates noise and thus misleads the predictive results.

In addition, the momentum spillovers of a firm attribute rely on the states of other attributes of both connected firms. For example, the abnormal price drop of one firm may not spill over if the trade volume of this decreasing price is limited, or the prices of the linked firms are undervalued. GNNs measure the momentum spillovers of one firm by the amount of its attribute transmitting to its related firms in the graph. Each node assigns each of its attributes a weight, which is obtained in the learning process. All the received spillovers of one firm are considered as its relational embedding to represent the node or to update its state. Several aggregators have been explored to gather the information from the neighbors to the target node in previous studies. However, traditional GNNs transfer and aggregate the peer influences without considering the states of the attributes of both connected firms once a connection is built. Most GNNs, including GCN (Scarselli et al. 2009) and GAT (Vaswani et al. 2017), generate relational embeddings via linear aggregators. Specifically, the attributes of the relevant firm nodes are linearly transformed via a weight matrix that measures the importance of each attribute, and are transported to the target node. In such a way, the weight of each attribute is fixed and shared across all firm nodes. Therefore, it can not dynamically adjust the attribute proportion to be transported in terms of the attribute states of the connected firm nodes as explained in the previous example about the influence of the abnormal price drop. Even though non-linear aggregators are sensitive to the attribute states, they typically adjust the amount of an attribute to be transferred in terms of the entire states of this attribute among all of the connected nodes. For example, in the max-pooling aggregator, only neighboring nodes hold the max value in certain attributes can generate spillovers and affect the target firm (Gao, Wang, and Ji 2018). In the LSTM-based aggregators, only the neighbors' attributes are considered and sequentially fed into LSTMs in a predefined order when modeling their spillovers (Hamilton, Ying, and Leskovec 2017). Apparently, the momentum spillovers of listed firms are attribute mattered and tradition GNNs are unable to handle this situation. In this study, an attribute-mattered aggregator is proposed to adjust the spillovers of one attribute in terms of other attribute states of two connected firm nodes.

3 Model Architecture

Figure 1 is an overview of the proposed framework for stock prediction. The tensor fusion (TF) module is first applied to merge technical indicators and textual media features into

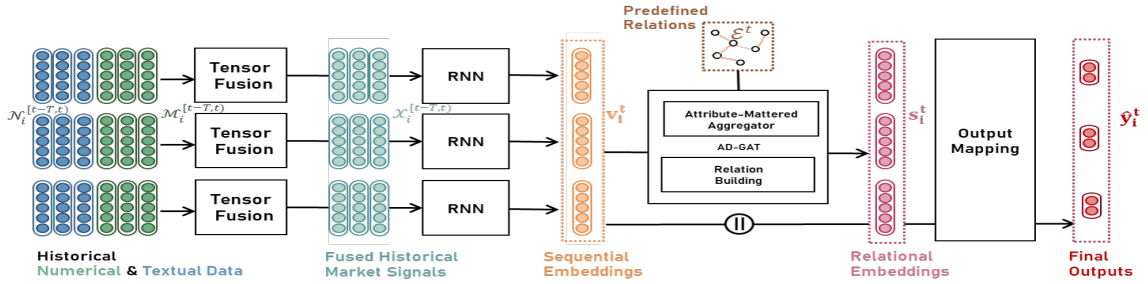

Figure 1: The proposed framework

one fused vector preserving their interactions. Second, the time-series fused vectors are fed into the RNN module to generate the sequential embedding of a listed firm which preserves the time dependencies of its fused market information. After that, the sequential embeddings of all listed firms, along with the predefined relations, are further processed by the proposed AD-GAT, which consists of the attribute-mattered (AM) aggregator and the relation building (RB) module, to generate the relational embeddings for all listed firms that represent their received spillovers. At last, the concatenations of sequential embeddings and relational embeddings are fed into the Output Mapping (OM) module to predict stock movements.

In the following, we use x (lower-case letter) to denote index, X (upper-case letter) to denote a scalar, $\mathbf{x}$ (bold lower-case letter) to denote a vector, $\mathcal{X}$ (script letter) to denote a matrix, and θ to denote parameters that are learned during the training process.

3.1 The Sequential Embeddings of Listed Firms

The stock market is a dynamic system, in which firms are highly influenced by various types of time-series market signals. A variety of RNNs, including LSTM and GRU, has been applied to capture the sequential dependencies of market signals for stock predictions (Li et al. 2016; Chen, Wei, and Huang 2018; Feng et al. 2019; Li et al. 2020a). In previous studies, technical indicators and textual media features have been concatenated into a super vector as the input of RNNs, which inevitably ignores the intrinsic associations between them (Kolda and Bader 2009; Li et al. 2016). In this study, we propose the TF module to capture the interactions of different market signals, and combine it with the RNN module to generate richer sequential embeddings for further processing.

Tensor Fusion Module The TF module aims to extract high-level features from numerical and textual data while preserving their interactions for stock prediction. In this study, for a given firm i, $\mathbf{m}_i^t \in \mathbb{R}^L$ is the L-dimensional feature vector representing the technical indicators on the day t, and $\mathbf{n}_i^t \in \mathbb{R}^{L'}$ is the L'-dimensional feature vector representing the relevant media articles released on the day t.

The challenge lies in how to mingle $\mathbf{m}_i$ and $\mathbf{n}_i$ into one vector $\mathbf{x}_i$ while preserving their interactions for stock prediction. Here, superscript t is omitted for simplicity. One of alternatives to capture feature interactions is to reassemble $\mathbf{m}_i$ and $\mathbf{n}_i$ into tensors, and apply global tensor decomposition algorithms (Kolda and Bader 2009; Li et al. 2016). However, it is independent of downstream tasks and stocks.

To learn the task- and firm-specific interactions between $\mathbf{m}_i$ and $\mathbf{n}_i$, the proposed framework applies a K-dimensional bilinear tensor product term, $\mathbf{m}_i\mathcal{T}_i^{[1:K]}\mathbf{n}_i$, to capture the intrinsic associations between $\mathbf{m}_i$ and $\mathbf{n}_i$. $\mathcal{T}_i^{[1:K]} \in \mathbb{R}^{L\times L'\times K}$ is a third-order tensor of which the parameters are tuned towards downstream tasks. The k-th entry in the bilinear tensor product term is computed by one slice of the tensor. That is, $\mathbf{m}_i\mathcal{T}^k\mathbf{n}_i = \sum_{l\in L}\sum_{l'\in L'} T_{l,l'}^k \mathbf{m}_{i,l}\mathbf{n}_{i,l'}$.

To preserve their independent effects, $\mathbf{m}_i$ and $\mathbf{n}_i$ are concatenated and linearly transformed by a weight matrix, $\mathcal{W}_i \in \mathbb{R}^{(L+L')\times K}$. The fused daily market signals of firm i, $\mathbf{x}_i \in \mathbb{R}^K$, is represented by the sum of these two terms,

$$\mathbf{x}_i = tanh(\mathbf{m}_i\mathcal{T}_i^{[1:K]}\mathbf{n}_i + \mathcal{W}_i[\mathbf{m}_i||\mathbf{n}_i] + \mathbf{b}_i), \quad (1)$$

where $||$ denotes the concatenation, $\mathbf{b} \in \mathbb{R}^K$ is the bias vector, and $tanh$ is the activation function. The set of the learned parameters is $\theta = [\mathcal{T}_i^{[1:K]}, \mathcal{W}_i, \mathbf{b}_i, \forall i \in N]$, where N represents the number of firms.

RNN Module For a given firm i, to generate its sequential embedding,$\mathbf{v}_i^t$, on the day t, its fused market signal representations in the past T days, $\mathcal{X}_i^{[t-T:t)}$, are fed into the RNN module,

$$\mathbf{v}_i^t = RNN_i(\mathcal{X}_i^{[t-T:t)}), \quad (2)$$

where $\mathbf{v}_i^t \in \mathbb{R}^F$, F is the hidden size of RNN, and $\mathcal{X}_i^{[t-T:t)} = [\mathbf{x}_i^{t-T}, \ldots, \mathbf{x}_i^{t-1}]$ denotes the fused historical representations of stock i in the past T days.

In this study, GRU (Cho et al. 2014) is selected as our RNN module, since it is easier to train in practice and achieves similar performance with other variants of RNNs (Xu and Cohen 2018). More details can be referred to the work of Cho et al. (2014).

3.2 The Relational Embeddings of Listed Firms

In real markets, the movement of one firm is affected by its related firms which is well known as the momentum spillover effect in finance. There are two challenges in predicting stock movements with momentum spillovers. The

first is to deal with attribute-mattered momentum spillovers with a proper network design and the second is to find a dominant relation to bridge the relevant firms for momentum spillovers.

Attribute-Mattered Aggregator To model the momentum spillovers of listed firms, the stock market is treated as a graph, in which $\mathcal{V} = [\mathbf{v}_1^t, \ldots, \mathbf{v}_N^t] \in \mathbb{R}^{N \times F}$ is the representation of N firms, and $\mathcal{E} \in \mathbb{R}^{N \times N}$ reflects the predefined relation of listed firms, whose indexes are listed firms and entires are their connection strengths.

Previous works relied on GCNs to generate relational embeddings of listed firms which represents the received spillovers (Chen, Wei, and Huang 2018; Feng et al. 2019). Here, we use $\mathcal{S} = [\mathbf{s}_1^t, \ldots, \mathbf{s}_N^t] \in \mathbb{R}^{N \times F'}$ to represent the F'-dimensional firm relational embeddings. For a given firm i, its relational embedding at time t, $\mathbf{s}_i^t$, is calculated as the weighted sum of its neighbors' attributes via the linear aggregator of GCNs,

$$\mathbf{s}_i^t = \sigma\Big(\sum_{j, j \neq i}^{N} \underbrace{E_{i,j} \mathcal{W}_s \mathbf{v}_j^t}_{\text{spillovers from } j \text{ to } i}\Big), \tag{3}$$

where $\mathcal{W}_s \in \mathbb{R}^{F' \times F}$ is a weight matrix shared by all firms, which linearly transforms the neighbors' attributes into higher-level features, $E_{i,j} \in \mathcal{E}$ is the normalized relation between firm i and j, and σ is sigmoid function. However, Eq. 3 fails to consider the interferences of connected firms' attributes on momentum spillovers, which are vital for stock predictions. Figure 2 shows an example of such attribute interference. Suppose firm i and j have three attributes, i.e., "price", "volume", and "Price-to-Earning ratio (P/E)", which stands for the trading price, the quantity of traded shares, and whether the stock price is overvalued or not, respectively. In traditional GCNs, the abnormal "price" drop of firm j spills to i in terms of the stock graph presented at the left top of Figure 2. However, in the real market, the abnormal "price" drop of firm j should not affect the price of firm i since the price drop of firm j is only accompanied by a small amount of trade volume, and firm i has a low "P/E" indicating its undervalued trading price, as shown in the attribute table at the left bottom of Figure 2. To model the attribute-mattered momentum spillovers of listed firms, the AM aggregator introduces into the linear aggregator (Eq. 3) a gate mechanism that is a non-linear transformation ($c(\cdot)$) of the related firm attributes, and element-wisely multiply ($\otimes$) the gate with the attributes of the source firm to discriminate the spillover of one attribute in terms of other attributes. Therefore, the relational embedding of firm i at time t is redefined as

$$\mathbf{s}_i^t = \sigma\Big(\sum_{j, j \neq i}^{N} \underbrace{E_{i,j} \mathcal{W}_s \mathbf{v}_j^t \otimes c(\mathbf{v}_i^t, \mathbf{v}_j^t)}_{\text{spillovers from } j \text{ to } i}\Big). \tag{4}$$

Here, a single layer feed-forward neural network with $tanh$ activation function is applied to obtain the information gate, $c(\cdot)$, in terms of the current states of firm i and j which is defined in Eq. 2,

$$c(\mathbf{v}_i^t, \mathbf{v}_j^t) = tanh(\mathcal{W}_c[\mathbf{v}_i^t || \mathbf{v}_j^t] + \mathbf{b}_c), \tag{5}$$

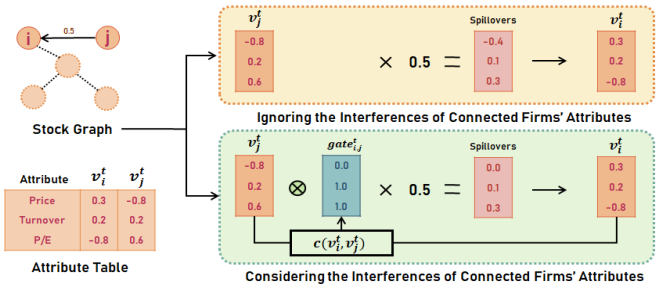

Figure 2: Attribute-mattered momentum spillovers

where $\mathcal{W}_c \in \mathbb{R}^{F' \times 2F}$ is the weight matrix, and $\mathbf{b}_c \in \mathbb{R}^{F'}$ is the bias vector. By adjusting the spillovers of one attribute in terms of the states of other attributes of connected firms using both $\mathcal{W}_s$ and $c(\mathbf{v}_i^t, \mathbf{v}_j^t)$, the proposed AM aggregator also partially releases the assumption that they are invariant when propagating between different firm pairs in different timestamps in Eq. 3, which only involves $\mathcal{W}_s$.

Relation Building Module The momentum spillovers of listed firms are essentially led by a fusion of various firm relations that change over time (Ali and Hirshleifer 2020). However, previous studies only leverage one or two types of firm relations (Chen, Wei, and Huang 2018; Feng et al. 2019).

Rather than fusing a dominant firm relation with a variety of predefined firm relations, we infer the latent firm relation, $R_{i,j}^t$, between firm i and j from the observed market signals at time t, that is,

$$R_{i,j}^t = r(\mathbf{v}_i^t, \mathbf{v}_j^t). \tag{6}$$

Different from TGC (Feng et al. 2019), we perform an unmasked attention mechanism that allows each firm to be attended by every other firm, and ignores all predefined relations that inevitably generate noise in the long run. Here, a single-layer feed-forward neural network with LeakyReLU activations (Maas, Hannun, and Ng 2013) is adopted as the shared attention mechanism to infer the latent firm relation,

$$R_{i,j}^t = LeakyReLU(\mathbf{a}_\mathbf{r}^\top \mathcal{W}_r[\mathbf{v}_i^t || \mathbf{v}_j^t]), \tag{7}$$

where $\mathbf{v}_i^t$ and $\mathbf{v}_j^t$ are concatenated and mapped into a F' dimensional vector using $\mathcal{W}_r \in \mathbb{R}^{F' \times 2F}$, and it is further transformed into a scalar that represents their connection strength via a vector $\mathbf{a}_\mathbf{r} \in \mathbb{R}^{F'}$.

To make $R_{i,j}^t$ comparable over all firms, $R_{i,j}^t$ is further normalized with a softmax function over all choices of j,

$$\tilde{E}_{i,j}^t = softmax_j(R_{i,j}^t) = \frac{exp(R_{i,j}^t)}{\sum_{k \in N, k \neq i} exp(R_{i,k}^t)}, \tag{8}$$

where $\tilde{E}_{i,j}^t$ is the normalized connection strength from j to i at time t.

Therefore, the relational embedding of firm i can be further redefined as

$$\mathbf{s}_i^t = \sigma\Big(\sum_{j, j \neq i}^{N} \tilde{E}_{i,j}^t \mathcal{W}_s \mathbf{v}_j^t \otimes c(\mathbf{v}_i^t, \mathbf{v}_j^t)\Big). \tag{9}$$

Here, we replace the predefined static relation $E_{i,j}$ in Eq. 4 with a dynamic relation $\tilde{E}^t_{i,j}$.

To stabilize the learning process, we adopt the multi-heads setting as suggested by Vaswani et al. (2017). Specifically, we execute Eq. 9 with M independent attention mechanisms, and concatenate the results as

$$\mathbf{s}^t_i = \Big\|_{m=1}^{M} \sigma\Big(\sum_{j,j\neq i}^{N} \tilde{E}^{t,[m]}_{i,j} \mathcal{W}^{[m]}_s \mathbf{v}^t_j \otimes c^{[m]}(\mathbf{v}^t_i, \mathbf{v}^t_j)\Big), \quad (10)$$

where $||$ represents concatenation, $\tilde{E}^{t,[m]}_{i,j}$, $\mathcal{W}^{[m]}_s$, and $c^{[m]}(\cdot)$ are the estimated dynamic firm relation, the input linear transformation's weight matrix, and the information gate in the m-th attention head, respectively. Note that, by this definition, the final relational embedding of firm i, $\mathbf{s}^t_i$, consists of MF' features instead of F'. Here, the learned parameter set is $\theta = [\mathcal{W}^{[m]}_s, \mathcal{W}^{[m]}_c, \mathcal{W}^{[m]}_r, \mathbf{b}^{[m]}_c, \mathbf{a}^{[\mathbf{m}]}_{\mathbf{r}}, \forall m \in M]$.

3.3 Output Mapping Module

Finally, a single layer feed-forward neural network with the softmax function is applied to generate the probability of future stock trends, denoted as

$$\hat{\mathbf{y}}^t_i = O_i(\mathbf{v}^t_i || \mathbf{s}^t_i) = softmax(\mathcal{W}'_i[\mathbf{v}^t_i || \mathbf{s}^t_i] + \mathbf{b}'_i), \quad (11)$$

where $\mathcal{W}'_i \in \mathbb{R}^{(MF'+F)\times C}$ is the weight matrix, C is the number of classes, and $\mathbf{b}'_i \in \mathbb{R}^C$ is the bias vector. The set of learned parameter is $\theta = [\mathcal{W}'_i, \mathbf{b}'_i, \forall i \in N]$. The cross-entropy loss between $\hat{\mathbf{y}}^t_i$ and $\mathbf{y}^t_i$ is back-propagated to learn the parameters of the proposed framework.

3.4 Market Signal Representation

Financial studies have attributed stock movements to three types of market information, i.e., the numerical technical indicators, the textual media features, and the relational data about firm relevance. In this study, we adopted all three types as described below.

Technical indicators Transactional data are the main manifestation of firms' intrinsic value and investor expectations. Five technical indicator attributes are selected and each attribute has been shown to have some degree of predictive value (Li et al. 2020a). These attributes are the following: highest/lowest price, opening/closing price, and trade volume. Note that, to make stock prices and trade volumes comparable over all firms, the stock price at day t, P^t, is transferred to the return ratio, R^t, which is denoted as $R^t = (P^t - P^{t-1})/P^{t-1}$, and the trade volume is transferred to the turnover ratio by normalizing it with the total stock share of the corresponding firm.

Textual media Modern behavioral finance believes that investors are irrational, tending to be influenced by the opinions expressed in the media. The media sentiment is proved to be a leading signal for stock volatilities (Li et al. 2020a; Sedinkina, Breitkopf, and Schütze 2019; Rekabsaz et al. 2017; Wang et al. 2013). Most previous studies relied on the L&M financial sentiment dictionary published by Loughran and McDonald (2011) to represent textual media for stock predictions (Sedinkina, Breitkopf, and Schütze 2019; Wang et al. 2013). In this study, six sentiment features defined by the last version (2018) L&M dictionary are extracted to represent the media factor $\mathbf{m}^t_i$. These are positive, negative, uncertainty, litigious, constrain, strong, moderate, and weak.

Firm Relations One of the advantages of the proposed framework is the ability to infer the latent relation of the listed firms with observed market signals. However, to make comparisons with the baselines, we collected five popular firm relationships for this study. These are industry category, supply chain, competition, customer, and strategic alliance. Here, each type of firm relation is represented by an adjacency matrix, of which the column and row are indexed with firms. If there is a link between two firms, the corresponding element value in the matrix is 1, otherwise is 0.

4 Experimental Evaluation

To the best of our knowledge, the proposed framework is the first one that fuses numerical, textual, and relational data together for stock predictions. Previous works focus on partial of market information, which makes their experimental data unsuitable for our study. Here, we constructed a dataset incorporating all three types of market data and shared it along with our source code for peer researches[1].

To gauge the effectiveness of the proposed framework for predicting stock movements, we carried out a series of experiments on the actual market data of S&P 500 firms from February 8, 2011 to November 18, 2013. During this period, there are 700 transaction days in total. The daily transaction data were taken from Wharton Research Data Services[2] (WRDS). The textual media was generously provided by Duan et al. (2018), which are financial news articles published by Reuters and Bloomberg during the same period. 198 stocks without missing transaction data and having at least 100 related news articles during the selected period are kept. Five types of firm relations, described in Section 3.4, are collected from S&P Capital IQ[3].

4.1 Evaluation setting

Stock prediction is normally treated as a binary classification problem. If the closing price is higher than the opening price, the sample is labeled with "upward" ($y^t_i = 1$), otherwise labeled with "downward" ($y^t_i = -1$). In predicting whether we need to purchase stocks at the beginning of market, we made strict control that only news articles that are already released are considered. There are 51.2% "upward" samples and 48.8% "downward" samples. We divided 700 transaction days into three periods. Specifically, the first 560 days are used for training, the following 70 days for validation, and the last 70 days for testing.

The directional accuracy (DA) and the area under the precision-recall curve (AUC) score are adopted as evaluation metrics in our experiments, which are widely adopted in previous studies (Duan et al. 2018; Li et al. 2020a). The

[1] https://github.com/RuichengFIC/ADGAT

[2] https://wrds-www.wharton.upenn.edu

[3] https://www.capitaliq.com

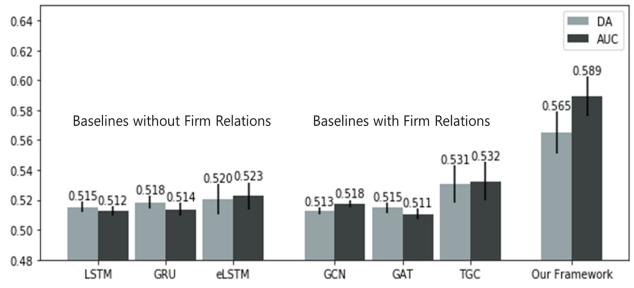

Figure 3: Comparing with baseline methods

evaluation is conducted in a rolling window fashion, as suggested by Li et al. (2020a). That is, we applied the market signals during the past T transaction days to predict stock movements on the t^{th} day.

Note that, to strengthen the robustness of our evaluation, for each methods compared in our experiments, we trained it 30 times with different initializations. We ranked these 30 runs and selected top-5 in terms of their performances in the validation period. The average performance of the selected runs in the testing period is reported to eliminate the fluctuations caused by random initializations. Each training procedure costs 2 hours in average with one NVIDIA Tesla V100 GPU card.

4.2 Hyper Parameters and Training Settings

Here, the grid search is employed to select the optimal hyper-parameters regarding DA. For the proposed framework, the window size T is searched within $\{5, 10, \ldots, 50\}$ the hidden size of tensor product (K) is searched within $\{5, 10, 15, 20\}$, and the number of attention heads (M) is searched within $\{2, 4, 8, 16, 32\}$. Unless otherwise specified, the dimensions of all other parameters in this paper are searched within $\{30, 60, 120, 150, 240, 300, 360\}$, including the hidden size of the GRU (F) and the attention layer (F'). All parameters are initialized using the Glorot initialization (Glorot and Bengio 2010) and are trained using the Adam optimizer (Kingma and Ba 2015) with an initial learning rate of 0.0005. In this study, the optimal window size T is 30. the dimension of tensor product K is set to 5, the hidden size of GRU F is set to 360, The attention layer consists of $M = 6$ attention heads and its hidden size F' is set to 60.

4.3 Comparison

To evaluate the overall performance of the proposed framework, we compared it with several baselines including both non-graph-based and graph-based methods. In particular,

Non-graph-based Methods

- LSTM: Long Short-Term Memory is one of the most powerful deep learning models for time series forecasting. The LSTM with 2 layers was implemented in our evaluation.
- GRU: Compared to LSTM, GRU holds a simpler design, which makes it easier to train and helps obtain consistent results with deeper model architecture. The GRU with 2 layers was implemented in our evaluation.
- eLSTM: Li et al. (2020a) proposed the eLSTM to effectively utilize the historical market signals when making predictions. The original settings of eLSTM were adopted.

Graph-based Methods

- GCN: The neighbors' attributes in GCN are linearly aggregated to central nodes with normalized firm relations. The GCN with 2 convolution layers was implemented.
- GAT: It specifies different weights to different nodes in a neighborhood. The GAT with 2 masked self-attention layers was implemented.
- TGC: Feng et al.(Feng et al. 2019) proposed the Temporal Graph Convolution (TGC) model for stock predictions, which dynamically adjusts the predefined firm relations before feeding them into GCN. The original settings were adopted.

All baselines considers technical indicators and textual media features. Specifically, the technical indicators and textual media features of the past T days are concatenated sequentially and fed into LSTM, GRU, eLSTM, and TGC. Since both GCN and GAT are unable to handle sequential data, for a given firm, its technical indicators and textual media features of the past T day are concatenated into a super compound vector to represent firm nodes. All graph-based baselines considered predefined firm relations, which are represented by an adjacency matrix. For a given element in the matrix, it is set to 1 if its corresponding firm pair has any of the five predefined firm relations.

Figure 3 shows the results of all baseline models and the proposed method. LSTM, GRU, and eLSTM, which only consider the sequential dependencies among market signals, achieve similar performances with GAT and GCN, which only consider the structural information of the stock market. TGC achieves the second-best performance in terms of DA and AUC. Compared to other methods, it considers both the sequential dependencies among market signals and the structural information of the stock market. The proposed approach achieves the best performances, with enhancements of at least 6.4% and 10.7% in terms of the DA and AUC, respectively. The p-values of the t-tests are all less than the critical confidence value (0.05), indicating that the superior performance of the proposed approach was statistically significant. Comparing with TGC, our approach infers the ultimate firm relations responsible for propagating spillovers from observed market signals without any help from the predefined firm relations and achieves a promising result. This indeed paves a way for the problem of capturing the influential power from peers even without any predefined peer relation at hand.

4.4 Effectiveness of the Proposed Approach

Effectiveness of AD-GAT In finance, the momentum spillovers of listed firms is well acknowledged. A common strategy of the pilot work is to adopt graph convolution networks (GCNs) with some predefined firm relations. In this

Firm Relation	AM		Linear	
	DA	AUC	DA	AUC
Industry	0.5322	0.5351	0.5210	0.5298
Supplier	0.5288	0.5388	0.5172	0.5222
Customer	0.5302	0.5473	0.5203	0.5306
Competitor	0.5325	0.5505	0.5219	0.5425
Alliance	0.5346	0.5472	0.5232	0.5389
Combination of All	0.5501	0.5550	0.5355	0.5433
Inferred Relation	0.5647	0.5894	0.5436	0.5672

Table 1: Results of different relations and aggregators

study, we infer the latent firm relation from observed market signals to avoid the bias generated by the predefined relations. To judge whether incorporating with the predefined relations can further boost the proposed implicit relation inference, the predefined firm relation is added to $R^t_{i,j}$ in Eq. 8 before normalization. We carried out a series of experiments with all of these five predefined relations. Table 1 shows the results of different aggregators (columns) and firm relations (rows). It can be observed that inferring the implicit relation is superior to any predefined firm relations, and achieves the best performance with enhancements of at least 2.7% and 6.2% in terms of the DA and AUC, respectively. This finding further proves that the momentum spillovers of listed firms follows a variety of firm links, and utilizing the predefined firm relations as the hidden spillover channel inevitably causes bias and misleads stock predictions.

In addition, traditional GNNs based on linear aggregators transfer and aggregate the peer influences without considering the interferences of connected firms' attributes once a connection is built. Such non-attribute sensibility makes traditional GCNs inappropriate to deal with the attribute-sensitive momentum spillovers of listed firms. To judge the effectiveness of the AM aggregator, which models the attribute-mattered spillover effect in stock markets, we also compared it with the linear aggregator (Eq. 3). In Table 1, the proposed AM aggregator outperforms the linear aggregator under all firm relations, with average enhancements of 2.4% and 2.3% in terms of DA and AUC, respectively.

Effectiveness of Tensor Fusion The tensor fusion module is proposed to facilitate the RNN module to generate high-level features that are used to represent firm states and infer their relation. Different from previous methods in which the interactions of market signals are simply ignored or considered identical of all firms, the tensor fusion module learns the firm-specific interactions of market signals directly from observed price fluctuations.

To judge whether it is necessary to consider firm-specific interactions of market signals, we explored the proposed approaches with two variants:

- AD-GAT$_C$: Numerical and textual data, $x^t_i = [m^t_i||n^t_i]$, are concatenated to represent the daily market signals without considering their interactions.
- AD-GAT$_S$: Instead of using firm-specific tensor fusion modules, a shared tensor fusion module is used to model the interactions of market signals. This is achieved by removing the subscripts i of all parameters in Eq. 1, which assumes that all of these parameters are identical for all listed firms.

	DA	AUC
AD-GAT$_C$	0.5450	0.5604
AD-GAT$_S$	0.5542	0.5679
Our method	0.5647	0.5894

Table 2: Effectiveness of tensor fusion module

In Table 2, AD-GAT$_S$ shows a better performance than AD-GAT$_C$, indicating that capturing the interactions of different market signals is critical to stock predictions. Comparing with AD-GAT$_S$, the proposed method further improves the predictive performance with enhancements of 1.9% and 3.8% in terms of DA and AUC, respectively. This finding indicates that feature interactions are firm-specific and should be treated in term of each firm.

5 Conclusion

In stock markets, the past returns of one firm affect the returns of firms that are linked to it. Only few studies in machine learning predicted the trend of one firm in terms of its relevant firms. A common strategy in previous studies is to adopt GCNs with predefined firm relations that have been explored in financial studies. We argue that it is inappropriate to rely on predefined firm relations since the spillover channel is dynamically changing over time. In this study, we infer the implicit spillover channel via observed market signals. In addition, to model market signals precisely, we propose a tensor-based fusion module to capture the interactions of different signals which is typically ignored in previous studies. Experiments on the listed firms of the S&P 500 shows that the estimated firm relation is more efficient than the well-acknowledged firm relations to capture momentum spillovers, and modeling market information space with feature interactions can further improve stock predictions.

More importantly, in real stock markets, the influence propagation of linked firms is attribute-sensitive, wherein the spillover of one attribute is affected by the other attributes of two connected listed firms. However, traditional GCNs transfer and aggregate the peer influence without considering the states of both connected firms once a connection is built. In this study, we propose the attribute-driven graph attention network that holds a novel aggregator to capture the attribute-mattered momentum spillovers. Experiments on the three-year data of the S&P 500 demonstrate the superiority of the proposed framework over state-of-the-art algorithms, including GCN, eLSTM, and TGC with enhancements of at least 6.4% and 10.7% in terms of the DA and AUC, respectively. Indeed, the proposed attribute-driven graph attention network can be generalized to other problems with implicit relations or attribute-mattered information propagation, such as the estimation of option implied volatilities and bulk futures. However, its power is yet to be explored in the near future.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (NSFC) (71671141, 71873108, and 62072379), the Department of Science and Technology of Sichuan Province (2019YJ0250 and 2020ZYD018), and the Key Laboratory of Financial Intelligence and Financial Engineering of Sichuan Province.

References

Ali, U.; and Hirshleifer, D. 2020. Shared analyst coverage: Unifying momentum spillover effects. *Journal of Financial Economics* 136(3): 649–675.

Chen, Y.; Wei, Z.; and Huang, X. 2018. Incorporating corporation relationship via graph convolutional neural networks for stock price prediction. In *Proceedings of the International Conference on Information and Knowledge Management*, 1655–1658.

Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using RNN Encoder–Decoder for statistical machine translation. In *Empirical Methods in Natural Language Processing*, 1724–1734.

Duan, J.; Zhang, Y.; Ding, X.; Chang, C.; and Liu, T. 2018. Learning target-specific representations of financial news documents for cumulative abnormal return prediction. In *International Conference on Computational Linguistics*, 2823–2833.

Feng, F.; He, X.; Wang, X.; Luo, C.; Liu, Y.; and Chua, T.-S. 2019. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems* 37(2): 1–30.

Gao, H.; Wang, Z.; and Ji, S. 2018. Large-scale learnable graph convolutional networks. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1416–1424.

Glorot, X.; and Bengio, Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, volume 9 of *JMLR Proceedings*, 249–256.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Advances in neural information processing systems*, 1024–1034.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8): 1735–1780.

Kingma, D. P.; and Ba, J. 2015. Adam: a method for stochastic optimization. In *International Conference on Learning Representations*.

Kolda, T. G.; and Bader, B. W. 2009. Tensor decompositions and applications. *SIAM review* 51(3): 455–500.

Li, Q.; Chen, Y.; Jiang, L. L.; Li, P.; and Chen, H. 2016. A tensor-based information framework for predicting the stock market. *ACM Transactions on Information Systems* 34(2): 1–30.

Li, Q.; Chen, Y.; Wang, J.; Chen, Y.; and Chen, H. 2018. Web media and stock markets: A survey and future directions from a big data perspective. *IEEE Transactions on Knowledge and Data Engineering* 30(2): 381–399.

Li, Q.; Tan, J.; Wang, J.; and Chen, H. 2020a. A multimodal event-driven lstm model for stock prediction using online news. *IEEE Transactions on Knowledge and Data Engineering* doi:10.1109/TKDE.2020.2968894.

Li, W.; Bao, R.; Harimoto, K.; Chen, D.; Xu, J.; and Su, Q. 2020b. Modeling the Stock Relation with Graph Network for Overnight Stock Movement Prediction. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 4541–4547.

Loughran, T.; and McDonald, B. 2011. When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. *The Journal of Finance* 66(1): 35–65.

Maas, A. L.; Hannun, A. Y.; and Ng, A. Y. 2013. Rectifier nonlinearities improve neural network acoustic models. In *International Conference on Machine Learning*, volume 30, 3.

Ming, F.; Wong, F.; Liu, Z.; and Chiang, M. 2014. Stock market prediction from WSJ: text mining via sparse matrix factorization. In *2014 IEEE International Conference on Data Mining*, 430–439.

Rekabsaz, N.; Lupu, M.; Baklanov, A.; Dür, A.; Andersson, L.; and Hanbury, A. 2017. Volatility Prediction using Financial Disclosures Sentiments with Word Embedding-based IR Models. In *Proceedings of the Association for Computational Linguistics*, 1712–1721.

Scarselli, F.; Gori, M.; Tsoi, A. C.; Hagenbuchner, M.; and Monfardini, G. 2009. The graph neural network model. *IEEE Trans. Neural Networks* 20(1): 61–80.

Sedinkina, M.; Breitkopf, N.; and Schütze, H. 2019. Automatic Domain Adaptation Outperforms Manual Domain Adaptation for Predicting Financial Outcomes. In *Proceedings of the Association for Computational Linguistics*, 346–359.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Wang, C.-J.; Tsai, M.-F.; Liu, T.; and Chang, C.-T. 2013. Financial sentiment analysis for risk prediction. In *Proceedings of the International Joint Conference on Natural Language Processing*, 802–808.

Xu, Y.; and Cohen, S. B. 2018. Stock movement prediction from tweets and historical prices. In *Proceedings of the Association for Computational Linguistics*, 1970–1979.

Zhang, X.; Zhang, Y.; Wang, S.; Yao, Y.; Fang, B.; and Philip, S. Y. 2018. Improving stock market prediction via heterogeneous information fusion. *Knowledge-Based Systems* 143: 236–247.