AI Open 2 (2021) 168–174

Contents lists available at ScienceDirect

AI Open

journal homepage: www.keaipublishing.com/en/journals/ai-open

Heterogeneous graph knowledge enhanced stock market prediction

Kai Xiong, Xiao Ding *, Li Du, Ting Liu, Bing Qin

Research Center for Social Computing and Information Retrieval, Harbin Institute of Technology, 150000, Harbin, China

ARTICLE INFO

Keywords:
Stock market prediction
Heterogeneous graph
Representation learning
Heterogeneous graph neural networks

ABSTRACT

We focus on the task of stock market prediction based on financial text which contains information that could influence the movement of stock market. Previous works mainly utilize a single semantic unit of financial text, such as words, events, sentences, to predict the tendency of stock market. However, the interaction of different-grained information within financial text can be useful for context knowledge supplement and predictive information selection, and then improve the performance of stock market prediction. To facilitate this, we propose constructing a heterogeneous graph with different-grained information nodes from financial text for the task. A novel heterogeneous neural network is presented to aggregate multi-grained information. Experimental results demonstrate that our proposed approach reaches higher performance than baselines.

1. Introduction

Stock market prediction has been a long-standing challenging task in natural language processing community. Previous financial studies demonstrate that stock prices are the reflection of all known information, which called the efficient market hypothesis (Fama, 1965). Moreover, Ball & Brown (1968) indicate that *financial text* including news and financial reports can be a major factor that drives stock market fluctuations. As a result, there is an increasing interest in predicting future stock market movements using natural language processing techniques based on financial text.

Previous studies on financial-text-based stock market prediction can be mainly categorized into three types: lexical-feature-based, event representation learning based and document representation learning based methods. Essentially, these methods are distinguished from each other by the granularity of the textual information they utilize. Feature-based methods only use simple lexical features such as bags-of-words, noun phrases and named entities (Kogan et al., 2009; Schumaker and Chen, 2009) extracted from financial text to predict the movement of stock market. To better predict the movement of stock market, Ding et al. (2015, 2019) propose to utilize the event level information extracted from titles or abstracts. Moreover, to capture contextual information within document, many document representation learning based methods (Augenstein et al., 2016; Chang et al., 2016; Tang et al., 2015; Yang et al., 2016) are proposed to learn a distributed representation of the whole document or abstract, and build prediction model based on the representation vector, so that information from overall document can be utilized for making predictions.

However, the word level, sentence level and contextual information entailed in the financial text are all indispensable information for predicting the movement of stock market. More importantly, these information may form into complex interrelationship patterns. The ignorance of such patterns in previous methods will not only lead to difficulties in identifying predictive information, but also bring noise into the prediction process.

Du et al. (2019) point out that contextual information is necessary for understanding the event. Fig. 1 provides two historically happened cases. In both cases, the event *"sues"* has led to the stock price movement of the plaintiff company and defendant company. However, in these two cases, the direction of stock market movement caused by the same event *"sues"* is opposite, as the same event corresponds to two different contexts. As Fig. 1 (a) shows, it is because of that "Apple infringed Qualcomm's patents", so the event "Qualcomm *sues* Apple" had a positive influence on the future stock price of Qualcomm, and a negative influence on that of Apple, respectively. While as shown in Fig. 1 (b), "someone impersonate Laoganma's employees and cheated Tencent", the "sues" event is a mistaken incident caused by Tencent, this would lead to positive and negative predictions for Tencent and Laoganma, respectively. Hence, without the help of contextual information, it is almost impractical to accurately predict the movement of stock market merely based on the lexical or event information.

On the other hand, as shown in Table 1, the main idea of this news is

* Corresponding author.
E-mail addresses: kxiong@ir.hit.edu.cn (K. Xiong), xding@ir.hit.edu.cn (X. Ding), ldu@ir.hit.edu.cn (L. Du), tliu@ir.hit.edu.cn (T. Liu), qinb@ir.hit.edu.cn (B. Qin).

https://doi.org/10.1016/j.aiopen.2021.09.001
Received 10 May 2021; Received in revised form 27 July 2021; Accepted 12 September 2021
Available online 14 September 2021

"the opinion of *H & M* on Xinjiang cotton" has angered Chinese people, which further leads to a decrease in stock price of *H & M*. However, in the news text, there is also noise such as "three days rise on stock prices", which may misguide the prediction model. Hence, it would be necessary to model the relationship patterns between events and sentences, so that we will know that "the opinion of *H & M* on Xinjiang cotton" is the predictive information and exclude "three days rise on stock prices" to lead to the right prediction.

To address these issues, we propose to predict the stock market movement by exploiting the relationship patterns between the words, events and contextual information within financial text. In specific, we build a heterogeneous graph to model the connections between word nodes, event nodes and sentence nodes. Hence, the fine-grained, medium-grained and coarse-grained information within the financial text can be exhaustively included in the heterogeneous graph. In addition, inspired by previous relational representation learning methods (Fan et al., 2019; Linmei et al., 2019; Wang et al., 2019; Wang et al., 2020; Yun et al., 2019), in this work, we propose a **H**eterogeneous **G**raph based Sequential **M**ulti-**G**rained **I**nformation aggregation **F**ramework (**HGM-GIF**), to model the relationship patterns and interactions between the different-grained information within financial text for stock market prediction. By sequentially integrating the word-to-event, word-to-sentence, event-to-event, sentence-to-event and event-to-sentence information, HGM-GIF can effectively model the complex relationship pattern between multi-grained information, so that fine-grained information can refine the information of coarse-grained information, and coarse-grained information can provide contextual information to enrich fine-grained information.

We highlight our contributions as follows:

(1) To our knowledge, this is the first work to construct a heterogeneous graph for stock market prediction, which contains multi-grained information;
(2) Our proposed framework can make all nodes in the heterogeneous graph interact and integrate with each other for information selection and exchange;
(3) Our approach can outperform all baselines on stock market prediction dataset. Ablation study demonstrates the effectiveness of our model.

2. Problem formulation

Given one or several news documents of a corporation, our goal is to predict the future stock reaction of this corporation based on corresponding financial text. Rather than predicting the specific price reaction range, we formulate the prediction as fluctuation polarity. The prediction will be "*Positive*" for the rise in stock price or "*Negative*" for the decline in stock price.

Table 1
Example of sentences in Financial Text and stock prices of corporations.

Sentence	After three consecutive days rise on stock prices, H & M was pushed to the forefront of public opinion due to the Xinjiang cotton incident.
Stock Price Reaction	**H&M**: Negative

To effectively exploit the predictive information within the financial text and exclude the noise, we propose to predict the stock price of a corporation k based on a heterogeneous graph HG_k built upon corresponding financial texts. In specific, let $C_k = \{c_k^1, \ldots, c_k^n\}$ denotes the financial text collection of corporation k, where c_k^i denotes the i-th document of k. Then from C_k, We can extract a series of event triples $E_k = \{et_k^1, \ldots, et_k^m\}$, where $et_k^i = (subj, v, obj)$ is the i-th event of corporation k, $subj$, v and obj are the subject, predicate and object of the event E_k^i, respectively. Meanwhile, we split each document into a set of sentences $S_k = \{s_k^1, \ldots, s_k^r\}$. To obtain the word nodes, the vocabulary in C_k is defined as $W_k = \{w_k^1, \ldots, w_k^z\}$. After that, all of the words, events and sentences are taken as nodes of the heterogeneous graph HG_k. We derive the connections among the multi-grained nodes using heuristic rules, which are described in detail in the experimental section. Formally, the heterogeneous graph is denoted as $HG_k = (N_k, D_k)$, where $N_k = \{W_k, E_k, S_k\}$ is the node set, and D_k is the edge set. The edges in the heterogeneous graph are undirected.

3. Method

As shown in Fig. 2, our proposed HGM-GIF framework consists of four components: a multi-grained heterogeneous graph constructor for heterogeneous graph construction, a multi-grained node encoder for encoding nodes into embeddings, a sequential heterogeneous information aggregator for modeling and aggregating information about multi-grained information relationship patterns, and a stock market movement predictor for the final stock price movement polarity prediction.

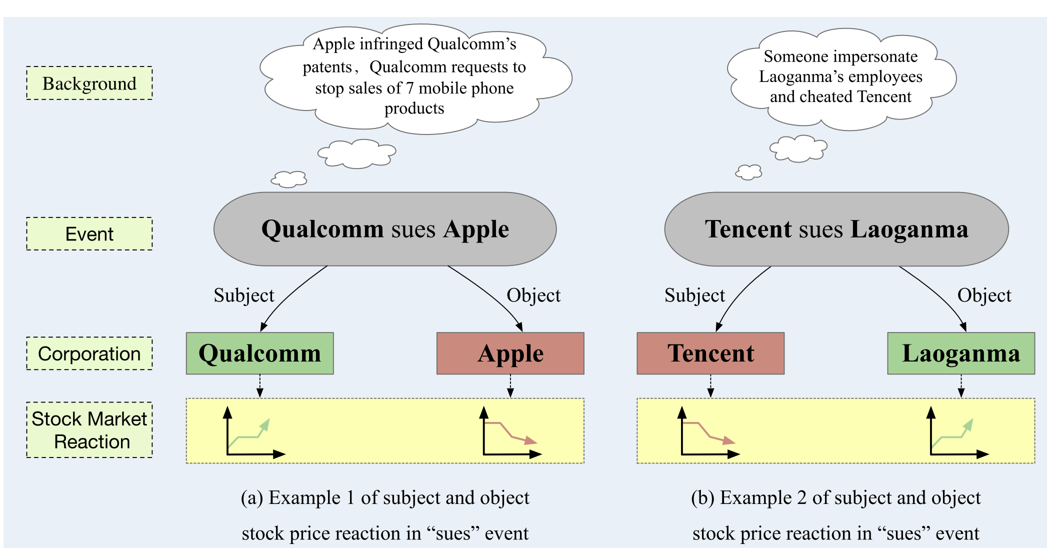

Fig. 1. Different stock prices reaction between two "sues" events.

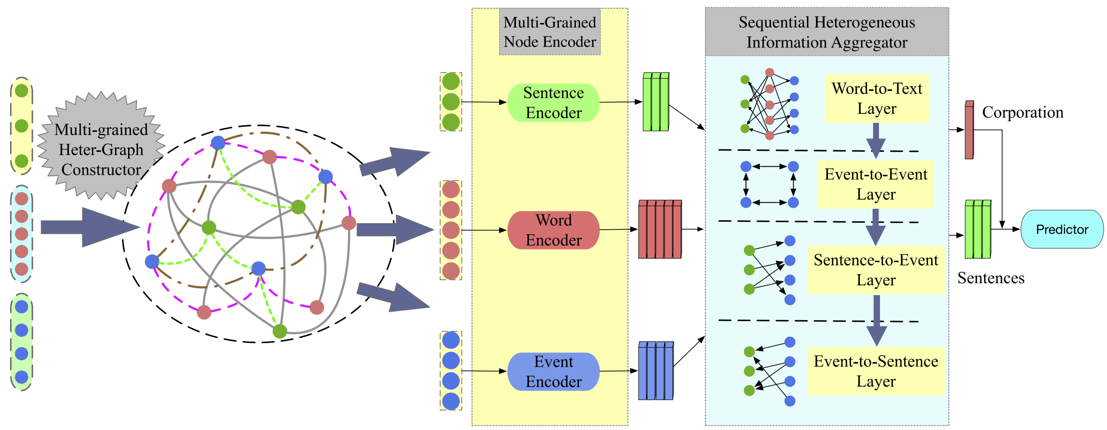

Fig. 2. The Architecture of our Proposed Framework. Green, red and blue solid circles denote sentence, word and event triple nodes, respectively.

3.1. Multi-grained heterogeneous graph constructor

For modeling the connections among different-grained information within financial text, given documents C_k of a corporation k, we first construct a heterogeneous graph with multi-grained information. To obtain word nodes (fine-grained information), we use a stopwords list to filter out stopwords in the vocabulary W_k. To obtain event triple nodes (medium-grained information), we extract a series of event triples from C_k using an exiting openIE tool (Schmitz et al., 2012). To obtain sentence nodes (coarse-grained information), we execute sentence splitting on C_k. Thereafter, we use heuristic rules to build connections among words, event triples and sentences. The rules are listed as follows:

(1) A word is connected with the sentence it belongs to.
(2) A word is connected with the event triple it belongs to.
(3) An event triple is connected with the sentence it belongs to.
(4) Two event triples are connected if they share the same named entity.
(5) Two event triples are connected if they are extracted successively.
(6) Two event triples are connected if they locate in adjacent sentences.

3.2. Multi-grained node encoder

In order to get the representations of different-grained nodes based on the constructed heterogeneous graph HG_k, we employ multi-grained encoders to encode the nodes within HG_k into embeddings.

Word Encoder We use the 300-dimension GloVe word embedding (Pennington et al., 2014) to encode our word nodes in the heterogeneous graph:

$$H_w = GloVe(W_k)W_{word}, \tag{1}$$

where $H_w \in \mathbb{R}^{z\times d}$ denotes the embeddings of word nodes, z is the number of word nodes and d is the dimension of node features. W_k is the word list of the *financial text*. $W_{word} \in \mathbb{R}^{300\times d}$ is a trainable parameter, 300 is the dimension of pretrained word embedding.

Event Triple Encoder For each event triple extracted from the sentences, we first use a word-wise attention to get the representations of every element. We take the embedding of subject as an example, and the embedding of predicate and object can be obtained in the same manner:

$$A^i_{subj} = Softmax(X^i_{subj}U_{subj}), \tag{2}$$

$$H^i_{subj} = A^i_{subj}X^i_{subj}, \tag{3}$$

where $X^i_{subj} \in \mathbb{R}^{l\times d}$ denotes the embeddings of words in subject of i-th event triple, where l is the number of words in subject and d is the dimension of features. $A^i_{subj} \in \mathbb{R}^l$ is the weight matrix and $U_{subj} \in \mathbb{R}^d$ is a trainable parameter. $H^i_{subj} \in \mathbb{R}^d$ is the features of subject. And then, we use the same function to get the embeddings of $H^i_v \in \mathbb{R}^d$ and $H^i_{obj} \in \mathbb{R}^d$ for predicate and object in i-th event triple, respectively.

Then we use a element-wise attention mechanism to get the representations of event triples:

$$X^i_{et} = Concat[H^i_{subj}; H^i_v; H^i_{obj}], \tag{4}$$

$$A^i_{et} = Softmax(X^i_{et}U_{et}), \tag{5}$$

$$H^i_{et} = A^i_{et}X^i_{et}, \tag{6}$$

where $X^i_{et} \in \mathbb{R}^{3\times d}$ indicates the concatenation of subject, predicate and object. $A^i_{et} \in \mathbb{R}^3$ indicates the attention matrix and $U_{et} \in \mathbb{R}^d$ is a trainable matrix. $H^i_{et} \in \mathbb{R}^d$ indicates the representation of i-th event triple.

Sentence Encoder In order to get the representation of sentences, we use Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) for capturing n-gram and semantic information, respectively. Then we concatenate two components as the representation of sentences.

$$X^i_{cnn} = CNN(X^i_s), \tag{7}$$

$$X^i_{lstm} = LSTM(X^i_s), \tag{8}$$

$$H^i_s = W_{con}Concat[X^i_{cnn}, X^i_{lstm}], \tag{9}$$

where $X^i_s \in \mathbb{R}^{b\times d}$ is the initial representation of i-th sentence, b is the length of a sentence. $X^i_{cnn} \in \mathbb{R}^o$ is the feature induced by CNN, o is the dimension of the feature, and $X^i_{lstm} \in \mathbb{R}^{2f}$ is the feature induced by Bi-LSTM, f is the hidden size of Bi-LSTM. $H^i_s \in \mathbb{R}^d$ is the representation of i-th sentence, d is the feature size, $W_{con} \in \mathbb{R}^{(2f+c)\times d}$ is a trainable parameter.

3.3. Sequential heterogeneous information aggregator

For modeling relationship patterns among different-grained information based on the initial representations of words, event triples and sentences, inspired by Wang et al. (2020), we propose a heterogeneous graph neural network to make different-grained nodes interact and integrate with each other. This framework is composed of four sequential components which progressively integrate different-grained information: *Word-to-Text Layer* for word information integration, *Event-to-Event Layer* for event relationship understanding, *Sentence-to-Event Layer* for event information supplement and *Event-to-Sentence Layer* for event information integration.

In each layer, we use a multi-head attention mechanism (Vaswani et al., 2017) to integrate information in the graph. In specific, each attention head is defined as:

$$w_{ij}^k = LeakyReLU(W_a^k[W_q^k h_i; W_p^k h_j]), \tag{10}$$

$$\alpha_{ij}^k = \frac{exp(w_{ij}^k)}{\sum_{l \in N_i} exp(w_{il}^k)}, \tag{11}$$

$$u_i^k = \sigma\left(\sum_{j \in N_i} \alpha_{ij}^k W_g^k h_j\right), \tag{12}$$

where $W_a^k, W_q^k, W_p^k, W_g^k$ are trainable parameters and k indicates k-th attention head. h_i is the node to be updated, and α_{ij}^k is the weight between h_i and h_j. N_i is neighbor nodes set of node i and u_i is the aggregated information from neighbors of node i.

To pool the information aggregate by each head, we concatenate the output of each head as output of multi-head attention. The multi-head attention can be defined as:

$$u_i = \|_{k=1}^K \sigma\left(\sum_{j \in N_i} \alpha_{ij}^k W_u^k h_i\right) \tag{13}$$

Finally, we add a residual connection to avoid gradient vanishing:

$$h_i^{'} = h_i + u_i. \tag{14}$$

For brevity, in the following sections, we denote the above mentioned process as:

$$h_i^{'} = InfoLayer(h_i, N_{type(i)}^t), \tag{15}$$

where $t \in \{w, et, s\}$ denotes neighbor node type (word, event triple or sentence) and $type(i) \in \{w, et, s\}$ denotes the function mapping the index of the node to node type, this information aggregation method is used in the following sections.

Word-to-Text[1] **Layer** In this layer, we use words to interact with text (event triples and sentences) for linguistic information. Information of event triples and sentences will be updated with information of words using the defined information aggregation method:

$$H_{et}^{'} = InfoLayer(H_{et}, N_{et}^w), \tag{16}$$

$$H_s^{'} = InforLayer(H_s, N_s^w), \tag{17}$$

where N_{et}^w denotes the word neighbor nodes set of event triple nodes, N_s^w denotes the word neighbor nodes of sentence nodes.

Event-to-Event Layer We use event triples interact with each other for event relationship understanding. Information of event triples will be updated by its neighbor event triples:

$$H_{et}^{''} = InfoLayer(H_{et}^{'}, N_{et}^{et}), \tag{18}$$

where N_{et}^{et} denotes the event triple neighbor nodes of event triple nodes.

Sentence-to-Event Layer Information of event triples will be updated by sentences in this layer for event contextual information supplement:

$$H_{et}^{'''} = InfoLayer(H_{et}^{''}, N_{et}^s), \tag{19}$$

where N_{et}^s denotes the sentence neighbor nodes of event triple node.

Event-to-Sentence Layer Event triples will be used to update information of sentences for new and key information selection:

$$H_s^{''} = InfoLayer(H_s^{'}, N_s^{et}), \tag{20}$$

where N_s^{et} denotes the event triple neighbor nodes of sentence nodes. After the information aggregation among different-grained nodes, we assume that the information within sentence nodes is supplemented and refined appropriately, and hence the stock market prediction can be conducted based on sentence nodes, which aggregate the information of words and event triples and key information is selected.

3.4. Stock market movement predictor

In order to select key information and get the stock price fluctuation of a specific corporation. Firstly, we use the corporation to softly select relevant information from sentences:

$$\alpha_i = H_c H_s^{''(i)}, \tag{21}$$

$$A_i = \frac{\alpha_i}{\sum_{j \in S_k} \alpha_j}, \tag{22}$$

$$U = AH_s, \tag{23}$$

where $H_c \in \mathbb{R}^d$ is the corporation representation and $H_s^{''} \in \mathbb{R}^{r \times d}$ is the embeddings of sentences. $A \in \mathbb{R}^r$ is the weight matrix and $U \in \mathbb{R}^d$ is the weight sum of sentences representations.

Thereafter, based on U, we use a linear function to predict future stock market fluctuation of this corporation:

$$X_{pred} = Concate[H_c, U], \tag{24}$$

$$Probabilities = Softmax(W_{pred} X_{pred}), \tag{25}$$

$$Prediction = ArgMax(Probabilities), \tag{26}$$

where $X_{pred} \in \mathbb{R}^{2 \times d}$ is the concatenation of corporation and sentence representation and $W_{pred} \in \mathbb{R}^d$ is a trainable parameter. *Probabilities* $\in \mathbb{R}^2$ denotes the final distribution of *Positive* and *Negative*. *Predictions* is the final prediction on stock market fluctuation, 0 and 1 denotes *Negative* and *Positive*, respectively.

4. Experiments

We conduct experiments on a stock market prediction dataset to evaluate the performance of our approach together with baselines.

4.1. Settings

Dataset We use the dataset released by Duan et al. (2018) in our experiments. The dataset includes more than 100k financial articles from Reuters[2], which also includes stock prices of different corporations. The time interval of the articles starts from October 2006 to December 2015. Only articles that mentioned at least one firm are selected by us,

Table 2
The statistics of our dataset.

	Train	Dev	Test
Positive	9,376	477	973
Negative	9,394	486	981
total	18,770	963	1,954
docs	27,657	1,419	2,989

[1] Text denotes Event Triples and Sentences.

[2] http://www.reuters.com/.

and the dataset is balanced on positive and negative classes. The statistics of this dataset are listed in Table 2.

Evaluation Metrics Follow the metrics used in prior works (Chang et al., 2016; Duan et al., 2018), we adopt the area under the precision-recall curve (AUC) as our evaluation metrics.

4.2. Training details

To construct the heterogeneous graph, we use an open-source openIE tool (Schmitz et al., 2012) for information extraction. Following the settings of Wang et al. (2020), we initialize the word nodes with 300-dimension GloVe embeddings (Pennington et al., 2014), and the dimension of every node feature is $d = 128$. The heads of multi-head attention is set as 8.

For the training process, the learning rate is set as 5*e*-4. Moreover, we apply an early stopping mechanism, training will be stopped if the performance on the development set doesn't improve for 10 epochs.

4.3. Baselines

Sentiment (Mayew and Venkatachalam, 2012) is a lexicon-based sentiment analysis method. We use the sentiment lexicons and method released by Loughran and McDonald (2011).

Event Tensor (Ding et al., 2015) is a neural tensor network based event representation learning method. They extract event triples from titles or abstracts. The event triples from the abstract will be averaged for prediction.

Event Tensor-CS[3] (Ding et al., 2019) is an external commonsense knowledge enhanced event representation learning method. Similar events are trained to be close to each other in vector space by predicting their sentiment polarities and contextual events.

TGT-CTX-LSTM (Chang et al., 2016) uses dependency parse tree to learn a target-related representation of abstract for stock market prediction.

Conditional Encoding (Augenstein et al., 2016) learns a target-related representation of abstract by using target as initial states for stance detection. We reimplement this method for stock market prediction.

TGT-HN (Tang et al., 2015) is a hierarchical document representation learning method. They use a LSTM to encode the sentence and another LSTM to encode the document for sentiment classification. We reimplement this method for stock market prediction.

TGT-HAN (Yang et al., 2016) assigns different weights for words and sentences in a document with an external feed-forward neural network for document classification. We reimplement this method for stock market prediction.

GCN (Yao et al., 2019) uses graph convolution to update and aggregate information with neighbor nodes. GCN is performed on the same heterogeneous graph and we use this method as our information interaction layer for stock market prediction. We use TD-IDF values and normalize them as the adjacency matrix among different nodes.

GAT (Veličković et al., 2017) assigns different weights for neighbor nodes, and gets a weighted sum of neighbor nodes for information aggregation. GAT is performed on the same heterogeneous graph and we use this method as our information interaction layer for stock market prediction.

5. Results and analysis

5.1. Overall results

We implement baselines and our approach on the stock market prediction dataset described in Sec 4.1. The overall results are shown in Table 3, from which we can make the following observations:

Table 3
Overall results on stock market prediction test set.

Method	AUC
Sentiment (Mayew and Venkatachalam, 2012)	0.533
Event Tensor + Title (Ding et al., 2019)	0.544
Event Tensor + Abstract (Ding et al., 2019)	0.549
Event Tensor-CS + Title (Ding et al., 2019)	0.564
Event Tensor-CS + Abstract (Ding et al., 2019)	0.570
Conditional Encoding (Augenstein et al., 2016)	0.603
TGT-CTX-LSTM (Chang et al., 2016)	0.632
TGT-HN (Tang et al., 2015)	0.615
TGT-HAN (Yang et al., 2016)	0.633
GCN (Yao et al., 2019)	0.621
GAT (Veličković et al., 2017)	0.615
HGM-GIF (Ours)	**0.638**

(1) Semantic knowledge acquisition methods (Event Tensor, Conditional Encoding, TGT-CTX-LSTM, TGT-HN, TGTHAN, HGM-GIF) achieve better results than feature-based method. The reason is that learning semantics within financial text is more effective than learning linguistic features for stock market prediction.
(2) Comparison between Event Tensor –CS, Conditional Encoding, TGT-CTX-LSTM, TGT-HN, TGT-HAN, HGM-GIF and Event Tensor shows that external knowledge or sentences within documents can supplement contextual information and offer more useful information for prediction. This confirms that predictions need rich contextual information. With the *sequential heterogeneous graph layer*, we can capture rich contextual information within financial text.
(3) Comparison between sentence-level methods (Conditional Encoding, TGT-CTX-LSTM) and document-level methods (TGT-HN, TGT-HAN, HGM-GIF), Conditional Encoding is worse than all document-level methods but TGT-CTX-LSTM is better than TGT-HN. We suggest that the dependency parse tree makes model better understand the information within the abstract and target-related representation can capture more predictive information. This indicates that how to understand information properly is also an important issue.
(4) Homogeneous graph based methods (GCN, GAT) have inferior performance than the HGM-GIF framework. Although GCN and GAT perform a good graph information aggregation operation, they treat the graph as a homogeneous graph and ignore different effects of different-grained nodes. This indicates that heterogeneous graph neural networks is reasonable for modeling information aggregation in the multi-grained information graph.

5.2. Ablation study

We conduct ablation studies to investigate the specific influence of different types of heterogeneous nodes, selection of nodes for prediction, the number of information aggregation iterations and different information aggregation layers.

Node Types We remove event triple and sentence nodes within the heterogeneous graph to investigate the influence of event triple nodes and sentence nodes, respectively. The experimental results are shown in Table 4. After removing event triples or sentences, the performance of our approach has a 0.007 and 0.066 decline, respectively. Which

Table 4
Influence of different node types, w/o denotes without.

Model	AUC
HGM-GIF	**0.638**
w/o Event Triples	0.631
w/o Sentences	0.572

[3] CS denotes Commonsense.

indicates that both two kinds of nodes include important information for prediction, and sentence nodes are far more important.

Prediction Nodes We investigate the influence of prediction nodes by using event triples only and the concatenation of event triples and sentences for prediction, respectively. Results are shown in Table 5. From which we observe that using event triples for prediction decline the performance. This indicates that sentences contain more information useful for prediction. However, concatenating event triples with sentences for prediction also declines the performance, we suggest that the representation of the corporation which comes from word nodes attends more weight to event triples, so more weight to event triples leads to performance decline.

Information Aggregation Iterations We investigate the influence of information aggregation times by changing the times of information aggregation. Results are shown in Table 6. From which we can observe that iterating only once achieves the best performance, while aggregating more than once leads to a decrease of performance, as the stack of multiple graph neural networks may lead to over-smooth (Zhao and Akoglu, 2019).

Information Aggregation Layer We investigate the influence of different information aggregation among different types of nodes. We remove information interaction between words and text, information interaction between events and events, information between sentences and events, respectively. Results are shown in Table 7. From which we can observe that without word-to-text interaction, the performance has 0.023 declines, we suggest that words can perform a fine-grained information selection of sentences and events; and performance has 0.022 declines if we remove event-to-event interaction, the reason might be that interaction within events can make model understand event relationship better. Furthermore, without sentence-to-event and event-to-sentence interactions has 0.01 decline in performance, we suggest that without sentence-to-event and event-to-sentence interactions might lose the key information selection process which can lead to a decline in performance.

Table 5
Experimental results of using different nodes for prediction.

Prediction Nodes	AUC
Event Triples	0.579
Concat (Sentences, Event Triples)	0.625
Sentences	**0.638**

Table 6
Influence of the number of information aggregation iterations.

Information Aggregation Iterations	AUC
1	**0.638**
2	0.535
3	0.506

Table 7
Influence of different information aggregation layer, w/o denotes without.

Information Aggregation Layer	AUC
HGM-GIF	**0.638**
w/o Word-to-Text Layer	0.615
w/o Event-to-Event Layer	0.616
w/o Sentence-to-Event & Event-to-Sentence Layer	0.628

6. Related work

6.1. Stock market prediction

To efficiently utilize the financial text information for stock prediction, previous works proposed to exploit the new and key information within financial text which is the major factor causing stock price fluctuations. Linguistic features have been studied for this task. Shallow lexical and syntactic features are exploited to make stock market prediction (Schumaker and Chen, 2009; Wang and Hua, 2014). Features such as n-grams, noun-phrases and named entities reach success in some concepts. However, these approaches ignore the rich semantic information within financial text. When there are two corporations to be predicted (eg. Fig. 1) or need the understanding of the text for prediction, these models are hard to make a right prediction. Semantic information is exploited by semantic frames (Xie et al., 2013) and event representation learning (Ding et al., 2014, 2015, 2016, 2019) for stock market prediction. These methods successfully capture semantic information within events, but lack massive contextual information within documents.

Compared to previous methods, our approach proposed to improve the AUC score to 0.638 with sentences information aggregated by multi-grained information.

6.2. Heterogeneou graph neural networks

To efficiently utilize and integrate graph structure information coming from multiple sources (eg. actor-director-movie graph), there is an improving interest in studying heterogeneous graph neural networks in recent years. These works propose different information aggregation methods (eg. heterogeneous attention neural network) to adapt different node types in the heterogeneous graph (Fu et al., 2020; Hu et al., 2020; Li et al., 2021; Wang et al., 2019; Yun et al., 2019; Zhang et al., 2019). Based on the theoretical researches, some researchers exploit constructing different heterogeneous graphs (eg. user-item-query graph, topic-text-entity graph) for application, such as text classification (Linmei et al., 2019), intent recommendation (Fan et al., 2019), cyberbullying detection (Chen and Li, 2020), event detection (Peng et al., 2019) and extractive document summarization (Jia et al., 2020).

Inspired by these works, we propose the HGM-GIF framework to aggregate different-grained information by using a *sequential heterogeneous graph layer*. Different-grained nodes can interact with each other through different layers for information supplement and selection.

7. Conclusion

To efficiently utilize the global information of documents, we construct a heterogeneous graph with multi-grained information for stock market prediction. Moreover, we propose the HGM-GIF framework to model the relationship among different kinds of nodes, and then aggregate information from the heterogeneous graph. Experimental results show that our approach achieves a higher AUC score on this task, as compared to event representation learning and document representation learning methods. Empirical studies show that heterogeneous graph is good at modeling documents and capturing massive information and contextual information within documents. Moreover, different-grained information which interact with each other for contextual information supplement and key information selection are all important to prediction.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank the reviewers for their constructive comments, and gratefully acknowledge the support of the Technological Innovation “2030 Megaproject” - New Generation Artificial Intelligence of China (2018AAA0101901), and the National Natural Science Foundation of

China (61976073).

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.aiopen.2021.09.001.

References

Augenstein, I., Rocktäschel, T., Vlachos, A., Bontcheva, K., 2016. Stance Detection with Bidirectional Conditional Encoding arXiv preprint arXiv:1606.05464.

Ball, R., Brown, P., 1968. An empirical evaluation of accounting income numbers. J. Account. Res. 159–178.

Chang, C.Y., Zhang, Y., Teng, Z., Bozanic, Z., Ke, B., 2016. Measuring the information content of financial news. Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 3216–3225.

Chen, H.-Y., Li, C.-T., 2020. Henin: Learning Heterogeneous Neural Interaction Networks for Explainable Cyberbullying Detection on Social Media arXiv preprint arXiv:2010.04576.

Ding, X., Liao, K., Liu, T., Li, Z., Duan, J., 2019. Event representation learning enhanced with external commonsense knowledge. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. EMNLP-IJCNLP), pp. 4896–4905.

Ding, X., Zhang, Y., Liu, T., Duan, J., 2014. Using structured events to predict stock price movement: an empirical investigation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1415–1425.

Ding, X., Zhang, Y., Liu, T., Duan, J., 2015. Deep learning for event-driven stock prediction. In: Twenty-fourth International Joint Conference on Artificial Intelligence.

Ding, X., Zhang, Y., Liu, T., Duan, J., 2016. Knowledge-driven event embedding for stock prediction. In: Proceedings of Coling 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 2133–2142.

Du, L., Ding, X., Liu, T., Li, Z., 2019. Modeling event background for if-then commonsense reasoning using context-aware variational autoencoder. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. EMNLP-IJCNLP), pp. 2682–2691.

Duan, J., Zhang, Y., Ding, X., Chang, C.Y., Liu, T., 2018. Learning target-specific representations of financial news documents for cumulative abnormal return prediction. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 2823–2833.

Fama, E.F., 1965. The behavior of stock-market prices. J. Bus. 38, 34–105.

Fan, S., Zhu, J., Han, X., Shi, C., Hu, L., Ma, B., Li, Y., 2019. Metapath-guided heterogeneous graph neural network for intent recommendation. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2478–2486.

Fu, X., Zhang, J., Meng, Z., King, I., 2020. Magnn: metapath aggregated graph neural network for heterogeneous graph embedding. In: Proceedings of the Web Conference 2020, pp. 2331–2341.

Hu, Z., Dong, Y., Wang, K., Sun, Y., 2020. Heterogeneous graph transformer. In: Proceedings of the Web Conference 2020, pp. 2704–2710.

Jia, R., Cao, Y., Tang, H., Fang, F., Cao, C., Wang, S., 2020. Neural extractive summarization with hierarchical attentive heterogeneous graph network. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. EMNLP), pp. 3622–3631.

Kogan, S., Levin, D., Routledge, B.R., Sagi, J.S., Smith, N.A., 2009. Predicting risk from financial reports with regression. In: Proceedings of Human Language Technologies: the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 272–280.

Li, J., Peng, H., Cao, Y., Dou, Y., Zhang, H., Yu, P., He, L., 2021. Higher-order attribute-enhancing heterogeneous graph neural networks. In: IEEE Transactions on Knowledge and Data Engineering.

Linmei, H., Yang, T., Shi, C., Ji, H., Li, X., 2019. Heterogeneous graph attention networks for semi-supervised short text classification. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. EMNLP-IJCNLP), pp. 4823–4832.

Loughran, T., McDonald, B., 2011. When is a liability not a liability? textual analysis, dictionaries, and 10-ks. J. Finance 66, 35–65.

Mayew, W.J., Venkatachalam, M., 2012. The power of voice: managerial affective states and future firm performance. J. Finance 67, 1–43.

Peng, H., Li, J., Gong, Q., Song, Y., Ning, Y., Lai, K., Yu, P.S., 2019. Fine-grained Event Categorization with Heterogeneous Graph Convolutional Networks arXiv preprint arXiv:1906.04580.

Pennington, J., Socher, R., Manning, C.D., 2014. Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. EMNLP), pp. 1532–1543.

Schmitz, M., Soderland, S., Bart, R., Etzioni, O., et al., 2012. Open language learning for information extraction. In: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 523–534.

Schumaker, R.P., Chen, H., 2009. Textual analysis of stock market prediction using breaking financial news: the azfin text system. ACM Trans. Inf. Syst. 27, 1–19.

Tang, D., Qin, B., Liu, T., 2015. Document modeling with gated recurrent neural network for sentiment classification. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1422–1432.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention Is All You Need arXiv preprint arXiv:1706.03762.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y., 2017. Graph Attention Networks arXiv preprint arXiv:1710.10903.

Wang, D., Liu, P., Zheng, Y., Qiu, X., Huang, X., 2020. Heterogeneous Graph Neural Networks for Extractive Document Summarization arXiv preprint arXiv:2004.12393.

Wang, W.Y., Hua, Z., 2014. A semiparametric Gaussian copula regression model for predicting financial risks from earnings calls. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics 1, 1155–1165. Long Papers).

Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., Yu, P.S., 2019. Heterogeneous graph attention network. In: The World Wide Web Conference, pp. 2022–2032.

Xie, B., Passonneau, R., Wu, L., Creamer, G.G., 2013. Semantic frames to predict stock price movement. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pp. 873–883.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E., 2016. Hierarchical attention networks for document classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1480–1489.

Yao, L., Mao, C., Luo, Y., 2019. Graph convolutional networks for text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 7370–7377.

Yun, S., Jeong, M., Kim, R., Kang, J., Kim, H.J., 2019. Graph transformer networks. Adv. Neural Inf. Process. Syst. 32, 11983–11993.

Zhang, C., Song, D., Huang, C., Swami, A., Chawla, N.V., 2019. Heterogeneous graph neural network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 793–803.

Zhao, L., Akoglu, L., 2019. Pairnorm: Tackling Oversmoothing in Gnns arXiv preprint arXiv:1909.12223.