Modeling the Stock Relation with Graph Network for Overnight Stock Movement Prediction

Wei Li\textsuperscript{1}, Ruihan Bao\textsuperscript{2,*}, Keiko Harimoto\textsuperscript{2}, Deli Chen\textsuperscript{1}, Jingjing Xu\textsuperscript{1} and Qi Su\textsuperscript{1}
\textsuperscript{1}MOE Key Lab of Computational Linguistics, School of EECS, Peking University
\textsuperscript{2}Mizuho Securities Co.,Ltd
liweitj47@pku.edu.cn, \{ruihan.bao, keiko.harimoto\}@mizuho-sc.com, \{chendeli, jingjingxu, suokia\}@pku.edu.cn

Abstract

Stock movement prediction is a hot topic in the Fintech area. Previous works usually predict the price movement in a daily basis, although the market impact of news can be absorbed much shorter, and the exact time is hard to estimate. In this work, we propose a more practical objective to predict the overnight stock movement between the previous close price and the open price. As no trading operation occurs after market close, the market impact of overnight news will be reflected by the overnight movement. One big obstacle for such task is the lacking of data, in this work we collect and publish the overnight stock price movement dataset of Reuters Financial News. Another challenge is that the stocks in the market are not independent, which is omitted by previous works. To make use of the connection among stocks, we propose a LSTM Relational Graph Convolutional Network (LSTM-RGCN) model, which models the connection among stocks with their correlation matrix. Extensive experiment results show that our model outperforms the baseline models. Further analysis shows that the introduction of the graph enables our model to predict the movement of stocks that are not directly associated with news as well as the whole market, which is not available in most previous methods.\textsuperscript{1}

1 Introduction

Stock movement prediction is one of the most attractive topics in the Fintech area [Bollen et al., 2011]. Many researches are devoted to predicting the movement trend of stocks based on news or historic market information. Researchers try to predict the stock price based on historic market data [Feng et al., 2019], the stock related news [Hu et al., 2018] or the combination of both [Xu and Cohen, 2018]. These researches all focus on predicting on the level of a trading day. However, it is a widely accepted fact that the stock movement is highly stochastic and can be influenced by complicated factors [Malkiel, 1999]. Experts in the financial area agree that the time for the market to absorb the impact of news is uncertain, which ranges from a few minutes to hours, but usually less than a day. Therefore, using the news signal to predict the stock movement for the next day is not very reliable.

In this paper, we explore the prediction of the stock movement in a more practicable way. We propose to predict the overnight stock movement based on the overnight financial news. Overnight movement means the movement between the closing price of the previous day and the opening price of the next day. Only the news happened after the market being closed is considered. This way, the reaction of the market on the news can be more precisely reflected because there is no trading operation during the closing hours of the market.

When predicting the stock movement of a company, previous works only consider the news and market data of a single company. This omits the connection among related stocks. It is a common knowledge for market participants that the stock price of a company is often related to others that have business connection. For example, the stock of Toyota is related to the stock of Honda, because they are both in the automobile industry. Therefore, in this work, we propose to consider the information of related stocks when predicting the stock movement instead of treating them as isolated ones.

To represent the connection between two companies, we propose to adopt the correlation matrix among companies, which market participants often refer to. This correlation matrix is calculated based on the correlation of historic market data, which introduces very valuable information. Inspired by the success of graph neural networks, we propose a Long Short Term Memory Relational Graph Convolution Networks model (LSTM-RGCN) to represent the correlation among stocks. In the graph, each stock is a node, and the stock nodes are connected by the correlation between the two stocks filtered by a threshold.

To test the effectiveness of our proposed model, we collect and publish an overnight stock movement prediction dataset of Reuters Financial News, which is widely used in the financial industry. In the dataset, there are the financial news and market data from Reuters during 01-01-2013 to 09-28-2018 for Tokyo Stock Exchange (TSE). The experiment results show that our model outperforms various strong baseline models. Moreover, the introduction of the graph structure in-
deed helps predict the stock movement. Further analysis sug-
suggest that our model can infer the price movement of stocks
which are not associated with any news and the whole market
because of the graph representation.
We conclude our contributions as follows:
• We propose a more practical objective that aims to pre-
dict the overnight movement. One big obstacle in stock
prediction is the lack of data, in this work we publish
the corresponding dataset from a professional content
provider Reuters Financial News.
• We propose to consider the connection among stocks
when predicting the stock movement and propose a
LSTM-RGCN model to represent the connection.
• Extensive experiment results show that our model out-
performs all the baseline models. Further analysis sug-
jects that the introduction of the graph makes our model
able to infer the price movement of related stocks that
do not have news as well as the whole market.

2 Task Formulation and Dataset
In this section, we describe the task formulation and the
dataset. This task aims to predict the overnight stock move-
ment as positive or negative given the overnight news. By
overnight movement, we mean the movement between the
opening price of the current trading day \( p_o^t \) and the closing
price of the previous trading day \( p_c^{t-1} \):

\[
Movement = (p_o^t - p_c^{t-1}) / p_c^{t-1}
\]

Because stock price is volatile in normal cases, we consider
the movement as positive or negative only when it exceeds 0.5
times of hourly standard deviation of the stock movement.
By overnight news we mean the news that take place after
the trading market being closed. We choose overnight news
because the effect of normal news tends to be absorbed by the
market within an hour or even few minutes during the trading
hours of the day. On the contrary, the effect of the overnight
news would be reflected on the overnight movement.
The dataset consists of the headline of the news and the tar-
get stock overnight movement. The news are associated with
the stocks based on the “RIC” labels provided by Reuters.
We choose the data during 2013-01-01 to 2018-09-28. Some
statistics of the dataset is shown in Table 1.

<table>
<thead>
<tr>
<th># News</th>
<th>Avg Len</th>
<th>Max Len</th>
<th>Min Len</th>
<th>Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>363,929</td>
<td>72.8</td>
<td>262</td>
<td>4</td>
<td>376,414</td>
</tr>
</tbody>
</table>

Table 1: Statistic of our collected data. Movement means the num-
ber of price movements exceeding 0.5 hourly standard deviation.

3.1 Stock Correlation Graph
To model the correlation among stocks, we build a stock cor-
relation graph. In the graph, each node represents a stock.
Each node is attached with some news text data. The nodes
are connected in reference to a correlation matrix, which is
calculated based on the historic market price. The correlation
matrix will be published with the dataset. The historic mar-
ket price considers the market movement information. There-
fore, this correlation matrix provides very valuable information
about the inter-stock relation. The correlation values can be
either positive (including 0) or negative. Therefore, we de-
fine two kinds of relationships between nodes depending on
the polarity of the value, positively correlated (correlation \( \geq \)
threshold) or negatively correlated (correlation \(<\) threshold).
To reduce the noise of the correlation matrix, we connect the
two nodes only when the absolute value of the correlation
score is above a threshold in the matrix.

3.2 Node News Encoder
LSTM has been successfully applied in encoding the context
information of text data. Therefore, we propose to encode the
news headline of a node with LSTM:

\[
h_w = LSTM(h_{w-1}^t, x_w^t)
\]

where \( x_w \) is the word embedding, \( d^t \) is the \( t \)-th word token in
the news, \( h_w^t \) is the hidden state of word \( d^t \).

Since different words are not equally important in the
news, we propose to represent the sentence with attention
mechanism. We choose the stock embedding as the query
and do attention on the hidden vectors of the news words:

\[
s^t = softmax(W_s([x_s; h_w^t]))
\]

\[
h_n = sum(s^t \times h_w^t)
\]

where \( s^t \) is the stock embedding of the node. \( W_s \) is a learn-
able parameter matrix, \([\cdot;\cdot]\) means concatenation of vectors.

To represent the node feature, we combine the news text
vector \( h_n \) and the company embedding together:

\[
v = W_v([h_n; x_s])
\]

where \( W_v \) is a learnable parameter matrix.

3.3 Graph Encoder
In this section, we describe our proposed LSTM RGCN based
graph encoder.

GCN [Kipf and Welling, 2017] is able to model the graph
structure, which is the correlation among stocks in this case.
In our correlation matrix, there are two kinds of relationships
representing positive and negative correlation relations. The
original GCN is designed for the case where there is only
one kind of relation. Therefore, we propose to adopt Relational
Graph Convolutional Networks (RGCN) [Schlichtkrull et al.,
2018] to encode the graph structure:

\[
N^{t+1} = \sigma(\sum_r D_r \frac{1}{2} A_r D_r \frac{1}{2} H_r W_r + W_n H)
\]

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

Li et al. [2018] claim that GCN is vulnerable to the over-smoothing problem, which means that the value of different nodes would be very close after multiple layers of propagation. To alleviate this over-smoothing problem, we propose to add LSTM mechanism between RGCN layers so that the gate mechanism can dynamically select which part of the information should be transmitted to upper layers. Furthermore, we argue that the movement of one stock is related to the movement trend of the whole market. To model the movement trend of the whole market, we propose to add a global node to the graph, which can interact with each stock node. The LSTM process is calculated as follows:

\[
d_i^l, f_i^l, o_i^l = f_0^l, f_\theta^l, f_\omega^l(h_i^{l-1}; x_i; g_i^{l-1}; N_i^l) \\
u = \tanh(W_u[h_i^{l-1}; x_i; g_i^{l-1}; N_i^l] + b_u) \\
c_i^l = f_i^l \odot c_i^{l-1} + f_i^{l-1} \odot u \\
h_i^l = o_i^l \odot \tanh(c_i^l)
\]  

where $h_i$ is the aggregated vector calculated with the RGCN, $f_\theta$ is a one-layer feed forward network with sigmoid activation function and parameters $\theta$. $i, f, o$ indicate input, forget and output gates respectively. Different from the original design of LSTM, we also take the node embedding and $v_u$ and global node vector $g$ into consideration. $v_u$ serves as the role similar to residual connection, while $g$ can provide the information of the whole market.

### 3.4 Global Node

To calculate the hidden state of the global node, we first aggregate the hidden information of all the nodes with attentive pooling:

\[
\alpha_i = u(W_a h_i) \\
\text{score}_i = \frac{\exp(\alpha_i)}{\sum_j \exp(\alpha_j)} \\
\bar{h} = \sum_j \text{score}_j h_j
\]

where $W_a$ and $u$ are learnable parameters.

Then, we use LSTM mechanism to filter the aggregated global information based on the hidden state of the global node in the previous layer and the updated node representations of the current layer:

\[
f_{g}^l, f_{\theta}^l, o_{g}^l = f_{g}, f_{\theta}, f_{\omega}(g^{l-1}; h_i^{l-1}) \\
f_{0}^l, \ldots, f_{m}^l, g^l = \text{softmax}(f_{0}^l, \ldots, f_{m}^l, g^l) \\
c_{g}^l = f_{g}^l \odot c_{g}^{l-1} + \sum_{i} f_{i}^l \odot c_{i}^{l-1} \\
g^l = o^l \odot \tanh(c_{g}^l)
\]

where $f_g, f_i, o_g$ are the forget gate, input gate and the output gate of the global node, respectively.

### 3.5 Objective

After we get the hidden state of each node in the graph, we can predict the movement label:

\[
P = \text{softmax}(W h) \\
\text{loss} = -\sum q\log(P)
\]

where $W$ is a learnable matrix, $q$ is the gold label.

The task is modeled as a two-class classification problem. We use the standard cross entropy as the objective function.

### 4 Experiment

In this section, we describe the experiment setting, results and give detailed analysis.

#### 4.1 Data

We choose the stocks within the TPX500 and TPX100 index. Because the news data contains noisy news that do not influence

Tokyo Stock Price Index, commonly known as TOPIX or TPX, is an important stock market index for the Tokyo Stock Exchange. TPX500 and TPX100 are the indexes for the top 500 and 100 stocks in TPX.
In the experiment, we first filter the news with the “RIC” label provided in the data by Reuters, which are the stock codes that the news may influence. Then we filter the news with some financial keywords described in the paper of Chen et al. [2019b]. Because much of the news is not related to the market price, we choose the keywords in the category of earnings, affairs, business, ratings and corporate. News that do not contain these keywords are filtered out. In the task, we only predict the movement when the both news is available and the price movement exceeds 0.5 times of hourly standard deviation. As the results, there are 10,367 positive movements and 3,647 negative ones in TPX100. We choose the data in the period of 01-01-2018 ~ 04-30-2018 as the development set and the data in the period of 05-01-2018 ~ 09-30-2018 as the test set. Some details of the data are described in Table 2.

### 4.4 Results

In Table 3 we show the experiment results. From the results we can see that our proposed model outperforms all other baseline models. The random guess generally results in an accuracy of around 50. Simple models can produce similar results compared with deep learning based baseline models. We assume that this is because the expression form of the financial news is relatively simple, which makes the deep learning based text classifiers do not have big advantage over the simple models.

Both the simple models and deep learning based baseline models do not perform as well as our proposed one. We argue that this is because the news in the market is still not enough to infer the movement of a stock. Even filtered with the topic keywords, there is still much noisy news that do not influence the price of the stock. Therefore, by introducing the information of relevant companies, our model can figure out the trend of the stock from the neighboring nodes and further validate the effect of the attached news.

### 4.5 The Effect of Graph

In Figure 2, we show the experiment result with and without the graph structure. From the figure we can see that the accuracy on both TPX500 and TPX100 increase by a big margin when adding the graph. We assume that this is because that the information from the related companies can supplement the news information of the current stock. Without the neighboring news, the model would suffer from information deficiency problem. Furthermore, by introducing the information of the related companies, our model can cross validate the effect of the news on the stock price.

---

**Table 2: Statistics of the dataset used in the experiment.**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Node #</th>
<th>Valid Movement #</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPX500</td>
<td>498</td>
<td>16,190 1,055 1,171</td>
</tr>
<tr>
<td>TPX100</td>
<td>95</td>
<td>7,527 461 526</td>
</tr>
</tbody>
</table>

Valid Movement # here means the number of movement that exceeds 0.5 hourly standard deviation and is attached with at least one piece of news.

---

2. https://nlp.stanford.edu/software/GloVe-1.2.zip

---

- **Random Forest** [Pagolu et al., 2016]: this model takes word embedding of the news headline as the input feature and applies Random Forest classifier\(^3\) to predict the movement label. The word embedding is learned with GloVe\(^4\) on the bloomberg news data.
- **Naive Bayes**: this model also takes the word embedding of the news headline as the input features, but applies Naive Bayes classifier\(^5\) to predict the movement label.
- **Linear Regression**: this model also takes the word embedding of the news headline as the input features, but applies linear logistic regression\(^6\) to predict the movement label.
- **Hierarchical Attention Networks (HAN)** [Yang et al., 2016]: a state of the art text classification model using hierarchical bidirectional LSTM structure with attentive pooling to encode the word and sentence. In our task, each headline is treated as a sentence in the HAN model.
- **S-LSTM** [Zhang et al., 2018]: a state of the art text representation model using LSTM to encode text. A global node is inserted to interact with each word.

- **Transformer** [Vaswani et al., 2017]: a self-attention based model uses attention to encode context information of each word. A special “CLS” token is inserted in the front of the text, the hidden vector of which represents the whole text.

We use two kinds of word embeddings (GloVe and BERT) as the input features for the Random Forest, Naive Bayes and Linear regression three classifiers. For GloVe, we use the sum of the word embeddings. For BERT, we use the sentence vector. We do not use the sentence vector of BERT in our model because the vocabulary in the financial news headlines is very different from the vocabulary of the pre-trained BERT.

---

**4.3 Setting**

In the experiment, we set the layer number of S-LSTM and the proposed LSTM RGCN to be 3. The layer number of Transformer (baseline model) is 6. The headline length is truncated to 50. The maximum sentence number in hierarchical attention networks is truncated to 10. We set the threshold of correlation edge to 0.6, that is, only when the weight of the edge exceeds 0.6, there is an edge built between the two nodes. The embedding size of GloVe [Pennington et al., 2014] is 50. We use BERT (base) model to get the sentence vector, whose dimension is 768. We use Adam optimizer to train the model parameters. The learning rate is initially set to 0.001 and decayed by half after each iteration. The hidden size is 300.
Table 3: Experiment results (accuracy) on TPX500 and TPX100. Naive Bayes, Linear regression and random forest are traditional classification models using word embeddings as the features. “G” means using the sum of the GloVe word vectors, “B” means using the BERT sentence vectors. HAN, Transformer and S-LSTM are deep learning based models. Results show that our proposed model outperforms all the baseline models.

<table>
<thead>
<tr>
<th>Model</th>
<th>TPX500</th>
<th>TPX100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>50.34</td>
<td>50.55</td>
</tr>
<tr>
<td>Naive Bayes (G)</td>
<td>54.44</td>
<td>50.85</td>
</tr>
<tr>
<td>Naive Bayes (B)</td>
<td>44.66</td>
<td>41.63</td>
</tr>
<tr>
<td>Linear Regression (G)</td>
<td>54.86</td>
<td>49.91</td>
</tr>
<tr>
<td>Linear Regression (B)</td>
<td>52.35</td>
<td>52.09</td>
</tr>
<tr>
<td>Random Forest (G) [Pagolu et al., 2016]</td>
<td>49.66</td>
<td>54.06</td>
</tr>
<tr>
<td>Random Forest (B)</td>
<td>51.75</td>
<td>50.19</td>
</tr>
<tr>
<td>HAN [Yang et al., 2016]</td>
<td>54.35</td>
<td>54.63</td>
</tr>
<tr>
<td>Transformer [Vaswani et al., 2017]</td>
<td>55.38</td>
<td>53.50</td>
</tr>
<tr>
<td>S-LSTM [Zhang et al., 2018]</td>
<td>52.17</td>
<td>53.69</td>
</tr>
<tr>
<td>Proposal</td>
<td>56.14</td>
<td>58.71</td>
</tr>
</tbody>
</table>

Table 4: Associative stock movement inference result. The price movement in this experiment do not have directly attached news. Other models cannot infer the movement of these stocks, because there is no available information.

<table>
<thead>
<tr>
<th>Model</th>
<th>TPX500</th>
<th>TPX100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>50.34</td>
<td>50.55</td>
</tr>
<tr>
<td>Proposal</td>
<td>52.72</td>
<td>57.53</td>
</tr>
</tbody>
</table>

Table 5: Experiment results on data eliminating news of each topic. For instance, “-ratings” means we do not use the keywords from the topic of “ratings”.

<table>
<thead>
<tr>
<th>Topic</th>
<th>TPX500</th>
<th>TPX100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full data</td>
<td>56.14</td>
<td>58.71</td>
</tr>
<tr>
<td>-ratings</td>
<td>54.40</td>
<td>57.70</td>
</tr>
<tr>
<td>-affairs</td>
<td>53.50</td>
<td>55.95</td>
</tr>
<tr>
<td>-corporate</td>
<td>53.05</td>
<td>57.05</td>
</tr>
<tr>
<td>-business</td>
<td>51.21</td>
<td>56.91</td>
</tr>
<tr>
<td>-earnings</td>
<td>55.21</td>
<td>56.90</td>
</tr>
</tbody>
</table>

4.7 Whole Index Inference

To test whether our model can capture the price movement of the whole market, we design an experiment that predicts the price movement of the whole TPX index. The training process remains the same, while during evaluation, we predict the index price movement $P_{index}$ based on the global graph-level representation $g$ (depicted in section 3.4).

$$P_{index} = \text{softmax}(Wg)$$

The prediction process is the same as ordinary stocks. The parameter W is shared with the ordinary prediction in Eqn. 17. We use the news of TPX500 stocks. The prediction accuracy is 55.74, which is rather satisfactory.

Actually, the prediction of the index price movement is also an attractive objective. However, it is hard to infer the index price because there is no directly attached news and the data is quite limited compared with ordinary stock. In this paper, we provide a view that predicts the market level price movement based on the global node in the graph. The global node is calculated with attentive pooling on the stock nodes, which gives the model the ability to dynamically select the important information from the stock nodes.

4.8 Effect of Different News Topics

In Table 5 we show the results of data eliminating news of one specific topic. In the experiment, we iteratively eliminate
the filtering keywords from one topic out of the five topics we use, which are “ratings, affairs, corporate, business and earnings”. From the results we can see that the topic of “affairs” generally has the biggest influence on the stock especially for the TPX100. We assume that this is because the content of news in the “affairs” topic is generally negative and the pattern is easier to catch. As long as the model can detect a negative affair happening to a company, its stock is likely to be predicted to decline. When consulting to the domain experts, we expect that news of “earnings” topic would have the biggest influence on the price. However, the results show that news of “earnings” only has medium influence on the price. We assume that this is because the pattern behind the “earnings” topic is more complicated, which involves the digit numbers, the comparison between the published earnings and the earning expectation from the market and so on. By now, this kind of problem is still very difficult to solve especially with very limited data.

5 Related Work

5.1 Stock Prediction

Traditional researches use human engineered features to predict stock price movement. Schumaker and Chen [2009] propose to use features like noun phrases and entities to predict the stock price. Oliveira et al. [2013] propose to use several sentiment indicators to predict stock market variables. Ding et al. [2015] further propose to use the result of information retrieval systems to predict the event-based stock price. Qin et al. [2017] and Feng et al. [2019] propose to predict the stock price trend based solely on the market data sequence. However, the stock market is highly stochastic, it is almost impossible to predict the stock movement with only the historic market price. Hu et al. [2018] propose to predict stock price movement based on sequential news. Xu and Cohen [2018] propose to use sequential tweets and market data to predict the stock movement. Apart from stock movement prediction, Chen et al. [2019a] propose a hierarchical framework to predict the Forex movement by grouping and summarizing a large amount of finance text.

5.2 Graph Neural Networks on Text Representation

Graph neural networks are designed for graph structured data, which have been widely applied in various NLP tasks. Yao et al. [2019] propose to organize documents and words into a unified heterogeneous graph and apply GCN to learn the document node representation, which are later used to predict the label of the documents. Peng et al. [2018] propose to organize the text into local text windows that can be easily modeled by convolution operations. These works organize the text with word co-occurrence information and focus on building the graph based on the local intra-sentence information. Yasunaga et al. [2017] and Sahu et al. [2019] propose to capture the local and non-local dependencies the graph constructed out of the inter- and intra-sentence dependencies. Yin et al. [2019] and Li et al. [2019] propose to model the inter-sentence dependency with the shared entities between sentences.

6 Conclusion

In this paper, we propose the objective of overnight stock price movement prediction, which aims to predict the overnight stock price movement based on the overnight news that take place during the stock market closing time. Because the stocks are not independent, we propose to consider the information of related companies by introducing the stock correlation matrix, which is built based on the historic market information. To make better use of the graph structure, we propose the LSTM-RGCN model that can handle both the positive and negative correlation. Furthermore, the LSTM module makes the proposed model less vulnerable to the over-smoothing problem, which is faced by many graph based neural networks. Experiment results show that our model outperforms the strong baselines. Moreover, our model can infer the price movement of stocks that are not attached with any news as well as the whole market.

Acknowledgements

This work is supported by a Research Grant from Mizuho Securities Co., Ltd. Mizuho Securities also provide very valuable domain experts suggestions. The experiment data is provided by Mizuho Securities and Reuters.

References


