

Fundamental Analysis based Neural Network for Stock Movement Prediction

Yangjia Zheng, Xia Li*, Junteng Ma, Yuan Chen

School of Information Science and Technology,
Guangdong University of Foreign Studies, Guangzhou, China
{yjzheng, xiali, junteng.ma, yuanchen}@gdufs.edu.cn

Abstract

Stock movements are influenced not only by historical prices, but also by information outside the market such as social media and news about the stock or related stock. In practice, news or prices of a stock in one day are normally impacted by different days with different weights, and they can influence each other. In terms of this issue, in this paper, we propose a fundamental analysis based neural network for stock movement prediction. First, we propose three new technical indicators based on raw prices according to the finance theory as the basic encode of the prices of each day. Then, we introduce a coattention mechanism to capture the sufficient context information between text and prices across every day within a time window. Based on the mutual promotion and influence of text and price at different times, we obtain more sufficient stock representation. We perform extensive experiments on the real-world StockNet dataset and the experimental results demonstrate the effectiveness of our method.

1 Introduction

Stock Movement Prediction aims to predict the future price trend of a stock based on its historical price or related information. Stock price prediction can help investors, ordinary users and companies to predict the stock trend in the future, which has good application value.

The high randomness and volatility of the market make the task of Stock Movement Prediction a big challenge (Adam et al., 2016). However, with the development of neural network technology, stock movement prediction has achieved good results in recent years (Nelson et al., 2017; Hu et al., 2018; Xu and Cohen, 2018; Feng et al., 2019a; Sawhney et al., 2020; Tang et al., 2021; Zhao et al., 2022). Based on fundamental and technical analysis, existing methods can be roughly grouped into two categories, namely methods based on price factors only and methods based on price and other factors (e.g., news of the stock.). Nelson et al. (2017) used the LSTM(Hochreiter and Schmidhuber, 1997) network to predict future stock price trends based on historical price and technical analysis indicators. Feng et al. (2019a) used the adversarial training as perturbations to simulate the randomness of price variables, and trained the model to work well with small but intentional perturbations. They also extracted 11 related price features to effectively help the model to predict future changes.

According to the Efficient Market Hypothesis (EMH) (Fama, 1970), price signals themselves cannot capture the impact of market accidents and unexpected accidents, while social media texts such as tweets could have a huge impact on the stock market. Based on this idea, different models have been proposed to model relevant news texts to improve the overall performance of stock movement prediction. Hu et al. (2018) proposed to use the hierarchical attention mechanism to predict the trend of stocks based on the sequence of recent related news. Xu and Cohen (2018) integrated signals from social media which reflected the opinions of general users and used Variational Autoencoder(VAE) to capture the randomness of prices and the importance of different time steps by adding temporal attention. Sawhney et al. (2020) introduced a novel architecture for efficient mixing of chaotic temporal signals from financial data, social

*Corresponding author: xiali@gdufs.edu.cn

©2022 China National Conference on Computational Linguistics

Published under Creative Commons Attribution 4.0 International License

media, and inter stock relationships in a hierarchical temporal manner through Graph Attention Neural Network.

Although previous studies have achieved good results, whether it is a purely technical approach based on historical prices or a fundamental approach based on multiple factors such as prices and news, they can be improved in terms of the full integration of the two important factors of texts and prices. We found that previous works usually encode news and prices separately according to time series, and then fuse them through simple concatenation operation, similar to the work of Sawhney et al. (2020). In fact, in practice, prices on a given day can be influenced by different news at different times (e.g., previous day or after two days). Similarly, some news about a stock on a given day may be influenced by stock prices at different times. As is shown in Figure 1, if we can capture the context information of each price and text by different days, we can get more sufficient information for predicting the stock price accurately.

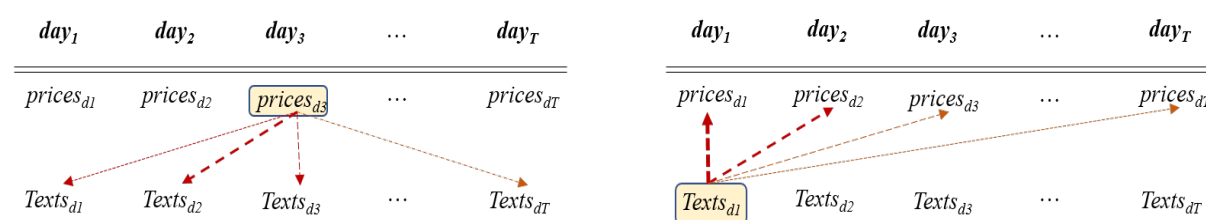


Figure 1: Contexts of prices and texts across the history captured by co-attention. Left shows each price representation of one day captures context of all news about the stock from day_1 to day_T by different attention weights. Right shows each texts representation of one day captures context of all prices about the stock from day_1 to day_T by different attention weights.

To this end, in this paper, we propose a fundamental analysis based neural network for stock movement prediction. More specifically, we first use Bi-GRU to encode the original texts of each day. Then, we use text-level attention to get a text representation of each day. As for the prices of each day, we use the existing 11 indicator features and 3 indicators we proposed in this paper as price representation of each day. Then we use the coattention mechanism (Xiong et al., 2016) to capture more information between texts and prices across every day within a time window. Finally, we incorporate a Bi-GRU to encode the fully integrated texts and prices representation according to the time window, so that it can obtain various prices and text-related information of the stock, and obtain the final effective representation of the stock.

The contributions of this work are as follows:

- We propose a fundamental analysis based neural network for stock movement prediction. The model introduces the coattention mechanism into text and price features of a stock to learn the effective context information of them. The method can obtain sufficient stock representation based on the mutual promotion and influence of texts and prices at different times.
- We also introduce three technical indicators based on raw prices in the financial field as their input features to better reflect the fluctuation information of the market. We perform multiple experiments on the StockNet dataset and the results demonstrate the effectiveness of our model.

2 Related Work

In this section, we will review the related work about stock movement prediction from technical analysis based approach and fundamental analysis based approach.

2.1 Technical Analysis based Approach

Technical analysis based approach is to predict the trend of a stock based on its historical price features such as close price and movement percent of price, which follows the assumption that future price changes are the result of historical behavior. Most recent stock movement prediction methods are based on deep learning. Among them, recurrent neural networks such as LSTM and GRU have become a

key part for capturing the temporal patterns of stock prices. This is because they can further capture long-term dependencies in time series. [Nelson et al. \(2017\)](#) used LSTM networks to study future trends, predicting stock prices based on historical stock prices and technical analysis indicators. These indicators are mathematical calculations designed to determine or predict the characteristics of a stock based on its historical data. A total of 175 technical indicators are generated each period, and they are designed to represent or predict a very different set of characteristics of a stock, like the future price, volume to be traded and the strength of current movement trends. [Feng et al. \(2019a\)](#) proposed to use adversarial training and add perturbations to simulate the randomness of price variables, and trains the model to work well with small but intentional perturbations. In addition, they extracted 11 related price features that effectively help the model predict future changes. [Feng et al. \(2019b\)](#) proposed the Temporal Graph Convolution (TGC) model combining historical prices for predicting movement of stock, which dynamically adjusts the predefined firm relations before feeding them into Graph Convolution Network (GCN) ([Kipf and Welling, 2017](#)). As LSTM struggles to capture extremely long-term dependencies, such as the dependencies across several months on financial time series. Transformer-based employs multi-head self-attention mechanism to globally learn the relationships between different locations, thus enhancing the ability to learn long-term dependencies. [Ding et al. \(2020\)](#) proposed various enhancements for Transformer-based models, such as enhancing locality of Transformer with Multi-Scale Gaussian Prior, avoiding learning redundant heads in the multihead self-attention mechanism with Orthogonal Regularization and enabling Transformer to learn intra-day features and intra-week features independently with Trading Gap Splitter. However, in reality, it is often difficult to find clear pattern of change from the market historical data. Furthermore, it fails to reveal the rules governing market volatility beyond stock price data.

2.2 Fundamental Analysis based Approach

Efficient Market Hypothesis tells that textual information can be used to extract implicit information for helping predict the future trend of stock prices, such as financial news and social media. Fundamental analysis based approach is able to capture information that is not available in traditional price-based stock prediction. A hybrid attention network ([Hu et al., 2018](#)) is proposed to predict stock trends by imitating the human learning process. In order to follow three basic principles: sequential content dependency, diverse influence, and effective and efficient learning, the model builds news-level attention and temporal attention mechanisms to focus on key information in news, and applies self-paced learning mechanisms to automatically select suitable training samples for different training stage improves the final performance of the framework. Different from the traditional text embedding methods, [Ni et al. \(2021\)](#) proposed Tweet Node algorithm for describing potential connection in Twitter data through constructing the tweet node network. They take into account the internal semantic features and external structural features of twitter data, so that the generated Tweet vectors can contain more effective information. Financial news that does not explicitly mention stocks may also be relevant, such as industry news, and is a key part of real-world decision-making. To extract implicit information from the chaotic daily news pool, [Tang et al. \(2021\)](#) proposed News Distilling Network (NDN) which takes advantage of neural representation learning and collaborative filtering to capture the relationship between stocks and news. [Xie et al. \(2022\)](#) conducted adversarial attacks on the original tweets to generate some new semantically similar texts, which are merged with the original texts to confuse the previously proposed models, proving that text-only stock prediction models are also vulnerable to adversarial attacks. This also reflects that the model obtained only by text training is less robust, so it is still necessary to incorporate knowledge such as relevant historical price features and the relationship between stocks to better improve the performance of the model.

Therefore, some studies fuse price and text data to build models, and even add the relationship between stocks to improve the performance of the model. A novel deep generation model that combines tweets and price signals is proposed by ([Xu and Cohen, 2018](#)). They introduced temporal attention to model the importance of different time steps and used Variational Autoencoder(VAE) to capture randomness of price. Recent studies have attempted to simulate stock momentum spillover through Graph Neural

Networks(GNN). Sawhney et al. (2020) introduced an architecture for efficient mixing of chaotic temporal signals from financial data, social media, and inter stock relationships in a hierarchical temporal manner. Cheng and Li (2021) proposed a momentum spillover effect model for stock prediction through attribute-driven Graph Attention Networks (GAT) (Veličković et al., 2017), and the implicit relations between stocks can be inferred to some extent. Zhao et al. (2022) constructed a market knowledge graph which contains dual-type entities and mixed relations. By introducing explicit and implicit relationships between executive entities and stocks, dual attention network is proposed to learn stock momentum over-flow features.

Since stock prices have temporal characteristics, that is, the price of a day will be affected by the price and news text of previous days, in this paper, we propose to use coattention mechanism to obtain the context information of stock prices and news text under different timestamp, so as to improve the final representation of the stock and the prediction performance.

3 Our Method

3.1 Task Definition

Similar to the previous work Xu and Cohen (2018), we define the stock movement prediction task as a binary classification problem. Given a stock s , we define the price movement of the stock from day T to $T + 1$ as:

$$Y_{T+1} = \begin{cases} -1, & p_{T+1}^c < p_T^c \\ 1, & p_{T+1}^c \geq p_T^c \end{cases} \quad (1)$$

where p_T^c represents adjusted closing price on day T , -1 represents stock price goes down and 1 represents the stock price goes up. The goal of the task is to predict the price movement Y_{T+1} of a stock s according to its historical prices collections P and news text collections L in a time sliding window of T days, where $P = \{P_1, P_2, \dots, P_i, \dots, P_T\}$, $L = \{L_1, L_2, \dots, L_j, \dots, L_T\}$, where P_i is the price features of the stock s on day i and L_j is the news text collection of the stock s on day j .

3.2 Overall Architecture

The whole architecture of our method is shown in Figure 2. As is shown in Figure 2, we first encode raw text for each stock across every day over a fixed time window. As for the price, the existing price features and the three new proposed indicators are concatenated together as the price representation. Then richer information will be captured by our introduced coattention mechanism. In order to obtain the integrated information of various prices and texts within the time window, we adopt a Bi-GRU for final encoding.

In the following sections, we will describe text and price features encoding in Section 3.3 and 3.4. And we will introduce temporal fusion to handle prices and text in Section 3.5 and introduce global fusion by sequential modeling in Section 3.6. Finally, model training will be introduced in Section 3.7.

3.3 Text Encoding

As each text contains rich semantic information, we use a Bi-GRU to encode the text and get the representation of each text in one day. Besides, different texts within the same day about the same stock may also be different (e.g., one text contains important information about the stock while other texts don't have valuable information about the stock.). For addressing that, we use a soft-attention operation to get the weighted representation of the texts of one day.

Following the work of Xu and Cohen (2018), we incorporate the position information of stock symbols in texts to handle the circumstance that multiple stocks are discussed in one single text. Given stock s contains K number of related texts on day m , which is denoted as $L_m = \{l_{m1}, l_{m2}, \dots, l_{mi}, \dots, l_{mK}\}$, where l_{mi} denotes the i -th text of stock s on day m . For each text $l_{mi} = \{w_1, w_2, \dots, w_n\}$, suppose that the location where the stock symbol appears first is denoted as z , we use two GRUs to encode the words sequence from w_1 to w_z to get the hidden representations \vec{h}_f and words sequence from w_z to w_n to get the hidden representations \vec{h}_b , respectively. We use the average of the last hidden states of the two GRUs \vec{h}_z and \vec{h}_z as the hidden representation of the text $h_{l_{mi}}$:

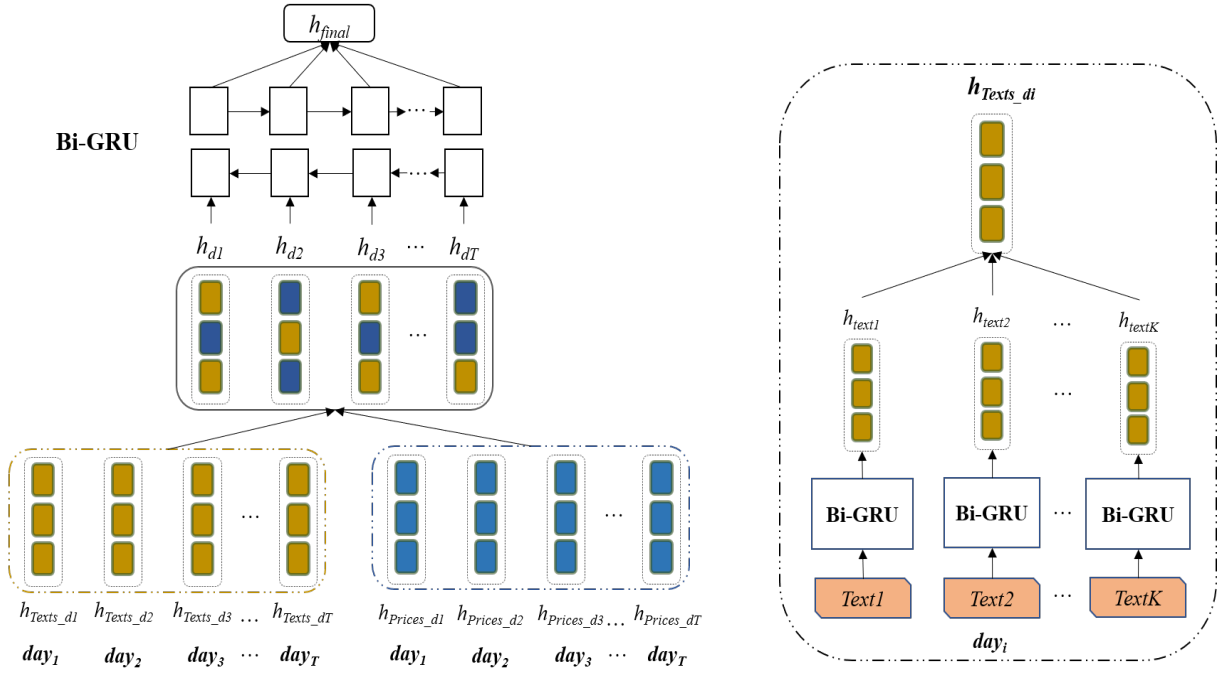


Figure 2: Overview architecture of our method.

$$\vec{h}_f = \overrightarrow{GRU}(e_f, \vec{h}_{f-1}) \quad (2)$$

$$\overleftarrow{h}_b = \overleftarrow{GRU}(e_b, \overleftarrow{h}_{b+1}) \quad (3)$$

$$h_{l_{mi}} = (\vec{h}_z + \overleftarrow{h}_z) / 2 \quad (4)$$

Where e_f, e_b is the word embedding using pre-trained Global Vectors for Word Representation (GloVe) (Pennington et al., 2014) for words of the text, $f \in [1, \dots, z], b \in [z, \dots, n]$. After that, we can get all the text representations $M_i = [h_{l_{m1}}, h_{l_{m2}}, \dots, h_{l_{mK}}]$. Since the text quality is different, we use a text-level attention mechanism to identify texts that could have a more substantial impact on the market every day, and finally obtain a final representation of all texts. The calculation formula is as follows:

$$u_K = \tanh(M_i W_m + b_m) \quad (5)$$

$$\alpha_K = \text{softmax}(u_K W_u) \quad (6)$$

$$h_{Texts_dm} = \sum_K \alpha_K h_{l_{mK}} \quad (7)$$

where α_K is the attention weight, W_m and W_u are the parameters to be learned, b_m is the bias terms. h_{Texts_dm} is the representation of the news text of stock s on m -th day (day_m). According to the time sliding window defined previously, the text data in the window is finally recorded as $H_t = [h_{Texts_d1}, h_{Texts_d2}, \dots, h_{Texts_dT}]$.

3.4 Price Features

As mentioned in Section 2.2, the models that predict stock trends only based on text are often fragile, while price features have been shown to effectively reflect market volatility. In this paper, we introduce three new relevant price features to be used in our method. The three new technical indicators are from financial domain and are used to describe fluctuation of stock, namely Average True Range (ATR) (Bruni, 2017), Bias Ratio (BIAS) and Momentum (MTM) (Lin et al., 2017). The detailed calculation of the three indicators is shown in Table 1. We describe the tree indicators as follows:

- **ATR:** ATR is a volatility indicator that was developed by Wilder (1978) and is used to measure the volatility or the degree of price movement of security. It was originally designed for commodity trading, which is frequently subject to gaps and limit moves. As a result, ATR takes into account gaps, limit moves, and small high-low ranges in determining the true range of a commodity, and it also applies to the stock market.
- **BIAS:** BIAS is the deviation between the closing price and moving average. When the stock price moves drastically to deviate from the trend, the possibilities for a pullback or rebound increase; When the stock price movement does not deviate from the trend, it is likely that the trend will continue.
- **MTM:** MTM is an indicator that shows the difference between today’s closing price and the closing price n days ago. Momentum generally refers to the continued trend of prices. Momentum shows a trend, staying positive for a sustained uptrend or negative for a sustained downtrend. An upward crossing of zero can be used as a signal of buying, and a downward crossing of zero can be used as a signal of selling. How high the indicator is (or how low when negative) indicates how strong the trend is.

| Features | Calculation |
|----------|--|
| ATR | $EMA(max(high_t, close_{t-1}) - min(low_t, close_{t-1}), n)$ |
| BIAS | $\frac{close_t}{\sum_{i=0}^4 close_{t-i}/5} - 1$ |
| MTM | $close_t - close_{t-1}$ |

Table 1: The three price features.

Following previous work, We adopt 11 temporal price features based on the raw price (Feng et al., 2019a), denoted as $F_1 = \{p_1, p_2, \dots, p_{11}\}$ and our proposed three new price features, denoted as $F_2 = \{p_{atr}, p_{bias}, p_{mtm}\}$, as our final price features. The two are concatenated together to get the final price features of m -th day, recorded as $h_{Prices_dm} = [F_1, F_2]$. According to the time sliding window defined above, the price features in the window are finally recorded as $H_p = [h_{Prices_d1}, h_{Prices_d2}, \dots, h_{Prices_dT}]$.

3.5 Temporal Fusion by Coattention Neural Network

After Section 3.3 and Section 3.4, the coding features of price and text were obtained as H_p and H_t respectively. To effectively blend text and price, we use the coattention mechanism (Xiong et al., 2016) to learn the fusion between text and price to obtain richer implicit information. First, we use a nonlinear projection layer to convert the dimension of the price feature into the same dimension as the text with the following formula:

$$H'_p = \tanh(H_p W_p + b_p) \quad (8)$$

Applying the coattention mechanism to focus on both text and price, and learn about fusion. We first compute an affinity matrix that contains the corresponding affinity scores of all prices hidden states and texts hidden state pairs. Then the affinity matrix is normalized by Softmax, attention weights are generated for text features by row, and attention weights of price features are generated by columns. The calculation formula is as follows:

$$L = H_t (H'_p)^T \quad (9)$$

$$A_t = softmax(L) \quad (10)$$

$$A_p = softmax(L^T) \quad (11)$$

Next, we calculate the attention context of price features based on the attention weight of text features. The calculation formula is as follows:

$$C_t = A_t H_p' \quad (12)$$

Meanwhile, we compute the attention context of the text features as $A_p H_t$ based on the attention weights of the price features. Following Xiong et al. (2016), we also calculate $A_p C_t$ which maps text feature encoding into the space of price feature encoding. The calculation formula is as follows:

$$h_d = A_p [H_t, C_t] \quad (13)$$

Where h_d is interdependent representation of the text and the price. The $[\]$ denotes for concatenation operation.

3.6 Global Fusion by Sequential Encoding

We input h_d obtained from Section 3.5 into the bidirectional GRU to obtain the hidden states for each time t . To capture past and future information as its context, we connect the hidden states from the two directions to construct a two-way encoding vector h_i with the following formulas:

$$\vec{h}_i = \overrightarrow{GRU}(h_d) \quad (14)$$

$$\overleftarrow{h}_i = \overleftarrow{GRU}(h_d) \quad (15)$$

$$h_i = [\vec{h}_i, \overleftarrow{h}_i] \quad (16)$$

In addition to its own information, h_i also contains information about its adjacent contexts. In this way, we encoded its time series. Since news releases on different dates contributed unequally to stock trends, we employed soft attention mechanism which is calculated as follows:

$$o_i = \tanh(h_i W_h + b_h) \quad (17)$$

$$\beta_i = \text{softmax}(o_i W_o) \quad (18)$$

$$h_{final} = \sum_i \beta_i h_i \quad (19)$$

where β_i is the attention weight, W_h and W_o are the parameters to be learned, b_h is the bias terms. Finally, h_{final} is input into a classic three-layer preceptron (MLP) to predict the future trend of stocks.

3.7 Model Training

We use cross entropy for model training, which is calculated by equation (20), where N is the total number of stocks, y_i^t and \hat{y}_i^t represent the ground truth and predict stock trend of stock i at t day, respectively.

$$l = - \sum_{i=1}^N \sum_t y_i^t \ln(\hat{y}_i^t) \quad (20)$$

4 Experiments

4.1 Dataset

We use the SotckNet¹ dataset (Xu and Cohen, 2018) to train and validate the model. The dataset contains historical data on the high trading volumes of 88 stocks in the NASDAQ and NYSE stock markets. We annotate the samples based on the movement percent of the adjusted closing price of stock, and label the samples as up and down when movement percent $\geq 0.55\%$ or $\leq -0.5\%$, respectively. We split the dataset temporarily with 70/20/10, leaving us with date ranges from 2014-1-1 to 2015-8-1 for training, 2015-8-1 to 2015-10-1 for validation and 2015-10-1 to 2016-1-1 for testing. Similarly, we adjusted trading days by removing samples with missing prices or texts and further aligned data for all trading day windows to ensure that data is available for all trading days in all windows.

¹<https://github.com/yumoxu/stocknet-dataset>

4.2 Experiment Settings

We use a 5-day trading day sliding window to build the samples. Following the setting of [Xu and Cohen \(2018\)](#), we set the maximum number of texts in a day to 30, and each text has a maximum of 40 words. Glove word embedding was also used to initialize words into 50-dimensional vectors. We train the model using the Adam optimizer, with an initial learning rate set to $5e-5$. The bidirectional GRU hidden dimensions for encoding tweets and sequential modeling were set to 100 and 64, respectively. Each model is trained for 40 epochs with a batch size of 32. We report the best average test performance of the model on the validation set at 5 different runs.

Following previous studies ([Xu and Cohen, 2018](#); [Sawhney et al., 2020](#)), we use Accuracy (Acc), F1 score, and Matthews Correlation Coefficient (MCC) as evaluation metrics for this classification task.

4.3 Compared Models

To demonstrate the effectiveness of our proposed model, we compare the results with the following comparative models.

- **RAND.** A simple predictor to make random guess about the rise and fall.
- **ARIMA.** Autoregressive Integrated Moving Average, an advanced technical analysis method using only price signals. ([Brown, 2004](#)).
- **Adversarial LSTM.** [Feng et al. \(2019a\)](#) proposed a deep model using an adversarial attention LSTM mechanism, which exploits adversarial training to simulate randomness during model training. They propose the use of adversarial training to improve the generalization of neural network prediction models, since the input feature for stock prediction is usually based on stock price, which is essentially a random variable that naturally changes over time. They added perturbations to their stock data and trained the model to work well with small but intentional perturbations.
- **RandForest.** [Pagolu et al. \(2016\)](#) implemented a sentiment analysis model based on Twitter data. The authors used Word2vec to analyse the polarity of sentiments behind the tweets and directly assessed tweets related to stock and tried to predict the price of the stock for the next day.
- **TSLDA.** A new topic model, Topic Sentiment Latent Dirichlet Allocation (TSLDA), which can obtain new feature that captures topics and sentiments on the documents simultaneously and use them for prediction of the stock movement ([Nguyen and Shirai, 2015](#)).
- **HAN.** A hybrid attention network that predicts stock trends by imitating the human learning process. Follows three basic principles: sequential content dependency, diverse influence, and effective and efficient learning. The model includes news-level attention and temporal attention mechanisms to focus on key information in news ([Hu et al., 2018](#)).
- **StockNet.** A Variational Autoencoder (VAE) to encode stock inputs to capture randomness and use temporal attention to model the importance of different time steps ([Xu and Cohen, 2018](#)). We compare with the best variants of StockNet.
- **MAN-SF.** Multipronged Attention Network (MAN-SF) jointly learns from historical prices, tweets and inter stock relations. MAN-SF through hierarchical attention captures relevant signals across diverse data to train a Graph Attention Network (GAT) for stock prediction. And the study considers one pre-built graph from Wikidata ([Sawhney et al., 2020](#)).

4.4 Experimental Results

We conduct several experiments to evaluate the performance of our method. In this section, we analyze the benchmark performance and the results of our model on the StockNet dataset. The experimental results of the different models are shown in Table 2.

First, we compare the first three baseline models presented in this paper. All three baseline methods use only historical price information, although Adversarial LSTM with more representative features and

| Model | Acc | F1 | MCC |
|---------------------------------------|-------------|-------------|--------------|
| RAND | 50.9 | 50.2 | -0.002 |
| ARIMA (Brown, 2004) | 51.4 | 51.3 | -0.021 |
| Adversarial LSTM (Feng et al., 2019a) | 57.2 | 57.0 | 0.148 |
| RandForest (Pagolu et al., 2016) | 53.1 | 52.7 | 0.013 |
| TSLDA (Nguyen and Shirai, 2015) | 54.1 | 53.9 | 0.065 |
| HAN (Hu et al., 2018) | 57.6 | 57.2 | 0.052 |
| StockNet (Xu and Cohen, 2018) | 58.2 | 57.5 | 0.081 |
| MAN-SF (Sawhney et al., 2020) | 60.8 | 60.5 | 0.195 |
| ours | 62.6 | 61.1 | 0.228 |

Table 2: The results of different models.

training with adversarial learning achieved better performance. Our model clearly exceeds these three methods in each evaluation indicator.

Second, our model is compared to models that only use textual information, such as RandForest, TSLDA, and HAN. Our model also significantly outperforms these three methods, outperforming the best-performing HAN by 5, 3.9, and 0.176 in Acc, F1, and MCC, respectively. So far, we can find that the performance of the model using only price or text is not satisfactory enough.

Finally, compared to StockNet, which also uses texts and prices, our model is 4.4, 3.6 and 0.147 higher on Acc, F1 values and MCC, respectively. Compared to another MAN-SF using the same data, our model contains no additional knowledge of stock relations. But the result still demonstrates that our model is 1.8, 0.6, and 0.033 higher than the MAN-SF on Acc, F1 values, and MCC, respectively. Overall experimental results demonstrate the effectiveness of the proposed model.

4.5 Ablation Study

In order to better demonstrate the different effects of components of our method, we conduct ablation studies to investigate the different contribution of coattention mechanism and the three proposed financial indicators. The results are shown in Table 3. We mainly design two variants: ours w/o coattention and ours w/o ATR-BIAS-MTM.

For w/o coattention, we change the method of learning effective implicit information between price and text from the coattention mechanism to the direct concatenation of the two. This model drops 1.7, 0.7 and 0.014 compared to the full model on Acc, F1 value and MCC, respectively, proving that the coattention mechanism can effectively improve the performance of the model and obtain richer information between price and text.

For w/o ATR-BIAS-MTM, We remove the three features proposed earlier in this paper and only use the 11 features proposed in previous studies (Feng et al., 2019a). The experimental results of the model decreased by 0.3, 0.3 and 0.007 on Acc, F1 values and MCC, respectively, which also prove that these three features help the performance of the model by reflecting the volatility of the market. Here we take ATR as an example to analyze, it can simply be understood as the expectations and enthusiasm of traders. Large or increasing volatility indicates that traders may be prepared to continue buying or selling stocks during the day. A reduction in volatility indicates that traders are not showing much interest in the stock market.

| Model | Acc | F1 | MCC |
|------------------|-------------|-------------|--------------|
| ours | 62.6 | 61.1 | 0.228 |
| w/o coattention | 60.9 | 60.4 | 0.214 |
| w/o ATR-BIAS-MTM | 62.3 | 60.8 | 0.221 |

Table 3: The ablation study of our method.

4.6 Case Study

As mentioned before, we use the coattention mechanism in the model to capture richer information, which in turn help to learn more precise attention weights of intra-day tweets (Tweet-level attention) and inter-day of time slide window (Temporal attention). In order to investigate how the coattention mechanism guides the learning of attention weights, we conducted a case study on a sample of \$FB(FaceBook) between Nov 5th and Nov 9th, 2015, which is finally used to predict the rise or fall of Nov 10th, 2015.

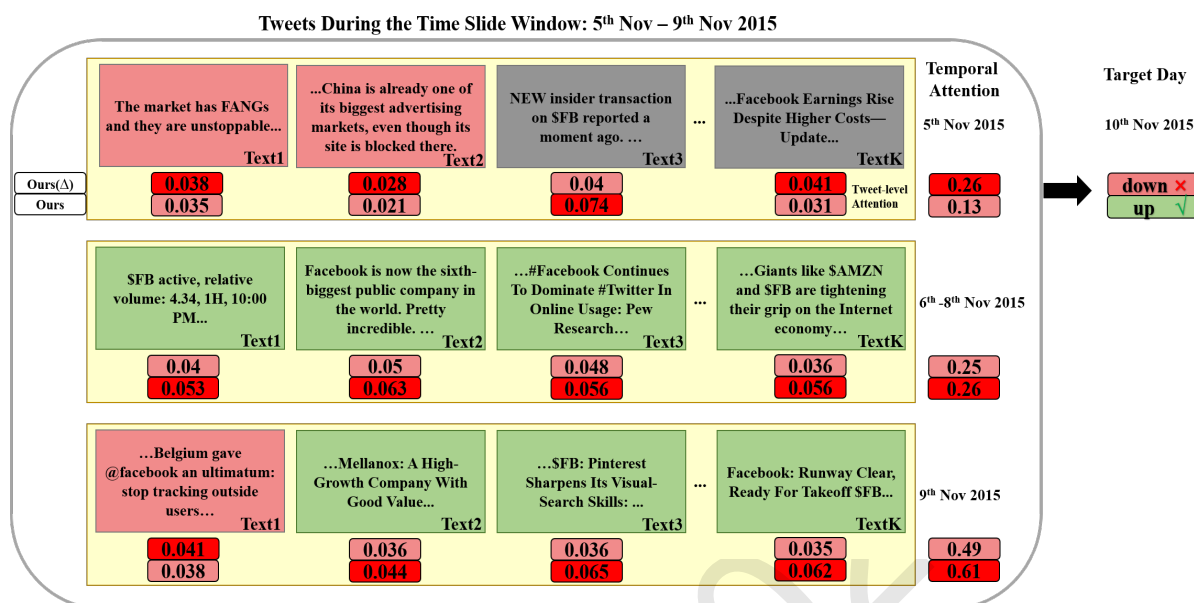


Figure 3: Text-level and Temporal attention weights learned by Ours and Ours(Δ) (as mentioned ours w/o coattention) on a sample of \$FB (FaceBook). Numbers represent weight values and darker colors indicate greater weights. Text on green, red and grey backgrounds represent signals with positive, negative and neutral respectively.

As shown in Figure 3, a row represents a day. For example, the first row represents texts of 5th. And we use the trading day alignment, because the 7th and 8th are weekends, so the text data for the three days from the 6th to the 8th were merged together. Each rectangle inside each row represents the content of a text. All texts within a day are denoted as [Text1, Text2, ..., TextK]. And We present the attention weights learned by our model (Ours) and without coattention mechanism (denoted as Ours(Δ)).

First, we can see that the closer to the target day, the more weight Ours gives to that day. This is also in line with the laws of the real world, and the newer news can have a greater impact. Specifically, Ours pays more attention to the positive signals from the 6th to the 9th. On the 5th, it pays too much attention to a neutral Text3 whose impact is uncertain. However, because of giving it a lower weight on the day, it can help its correct prediction for the rise. On the contrary, Ours(Δ) has a greater weight than Ours on the 5th. At the same time, the tweet texts with negative signals in the 5th and 9th are more concerned by Ours(Δ), and finally make a wrong prediction.

Next we analyze the texts for each day in more detail. For a more intuitive understanding, we artificially add different background colors to each rectangle to represent different tendencies of the text, such as green, red and grey backgrounds representing signals with positive, negative and neutral respectively. On the 5th day, we can see that Ours(Δ) has higher attention than Ours on the two negative texts Text1 and Text2. During the period from the 6th to the 9th, Ours gives a higher weight value to the texts with positive signals than Ours(Δ), such as the Text2 from the 6th to the 8th and the TextK of the 9th, which all reflect the good development prospects of FaceBook. In particular, Ours has a smaller weight than Ours(Δ) on the Text1 with negative influence in 9th. Although this negative news appears on the day closest to the target prediction, because the model combined with coattention can fuse the information of the entire window, and analyzes that Facebook stock is still showing an upward trend in general.

The observation shown in Figure 3 indicates that the coattention mechanism can guide the model to pay more attention to texts with tendencies and can effectively model the temporal. With more accurate attention weights, Ours can capture more effective representation, thus it can achieve better performance than Ours(Δ).

5 Conclusion

To effectively fuse texts and prices to predict future stock movements, in this paper, we propose a fundamental analysis based neural network for stock movement prediction. Our model introduces the coattention mechanism to capture richer implicit information between text and price as a better representation of a stock. We also introduce three new technical indicators in the financial field as price features. We perform the extensive experiments on the StockNet dataset and the experimental results show the effectiveness of our proposed method. In the future, we plan to use more data other than stock prices, such as financial reports, relationships between stock, to better capture market dynamics. In addition, extracting features that can better reflect trend changes is still a direction worth exploring.

Acknowledgements

This work is supported by National Natural Science Foundation of China (No. 61976062).

References

- Klaus Adam, Albert Marcet, and Juan Pablo Nicolini. 2016. Stock market volatility and learning. *The Journal of finance*, 71(1):33–82.
- Robert Goodell Brown. 2004. *Smoothing, forecasting and prediction of discrete time series*. Courier Corporation.
- Renato Bruni. 2017. Stock market index data and indicators for day trading as a binary classification problem. *Data in brief*, 10:569–575.
- Rui Cheng and Qing Li. 2021. Modeling the momentum spillover effect for stock prediction via attribute-driven graph attention networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 55–62.
- Qiangang Ding, Sifan Wu, Hao Sun, Jiadong Guo, and Jian Guo. 2020. Hierarchical multi-scale gaussian transformer for stock movement prediction. In *IJCAI*, pages 4640–4646.
- Eugene F Fama. 1970. Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, 25(2):383–417.
- Fuli Feng, Huimin Chen, Xiangnan He, Ji Ding, Maosong Sun, and Tat-Seng Chua. 2019a. Enhancing stock movement prediction with adversarial training. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5843–5849. International Joint Conferences on Artificial Intelligence Organization, 7.
- Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua. 2019b. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems (TOIS)*, 37(2):1–30.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. 2018. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 261–269.
- Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.
- Tao Lin, Tian Guo, and Karl Aberer. 2017. Hybrid neural networks for learning the trend in time series. In *Proceedings of the twenty-sixth international joint conference on artificial intelligence*, pages 2273–2279.

- David MQ Nelson, Adriano CM Pereira, and Renato A De Oliveira. 2017. Stock market's price movement prediction with lstm neural networks. In *2017 International joint conference on neural networks (IJCNN)*, pages 1419–1426. IEEE.
- Thien Hai Nguyen and Kiyooki Shirai. 2015. Topic modeling based sentiment analysis on social media for stock market prediction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1354–1364.
- Huihui Ni, Shuting Wang, and Peng Cheng. 2021. A hybrid approach for stock trend prediction based on tweets embedding and historical prices. *World Wide Web*, 24(3):849–868.
- Venkata Sasank Pagolu, Kamal Nayan Reddy, Ganapati Panda, and Babita Majhi. 2016. Sentiment analysis of twitter data for predicting stock market movements. In *2016 international conference on signal processing, communication, power and embedded system (SCOPEs)*, pages 1345–1350. IEEE.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, and Rajiv Shah. 2020. Deep attentive learning for stock movement prediction from social media text and company correlations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8415–8426.
- Tsun-Hsien Tang, Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2021. Retrieving implicit information for stock movement prediction. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2010–2014.
- Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph attention networks. *6th International Conference on Learning Representations*.
- J Welles Wilder. 1978. *New concepts in technical trading systems*. Trend Research.
- Yong Xie, Dakuo Wang, Pin-Yu Chen, Xiong Jinjun, Sijia Liu, and Oluwasanmi Koyejo. 2022. A word is worth a thousand dollars: Adversarial attack on tweets fools stock prediction. *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.
- Caiming Xiong, Victor Zhong, and Richard Socher. 2016. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*.
- Yumo Xu and Shay B Cohen. 2018. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1970–1979.
- Yu Zhao, Huaming Du, Ying Liu, Shaopeng Wei, Xingyan Chen, Huali Feng, Qinghong Shuai, Qing Li, Fuzhen Zhuang, and Gang Kou. 2022. Stock movement prediction based on bi-typed and hybrid-relational market knowledge graph via dual attention networks. *arXiv preprint arXiv:2201.04965*.