EAN: Event Attention Network for Stock Price Trend Prediction based on Sentimental Embedding

Yaowei Wang
City University of Hong Kong
Department of Computer Science
Hong Kong, China
yaowwang@cityu.edu.hk

Qing Li
The Hong Kong Polytechnic University
Department of Computing
Hong Kong, China
csqli@comp.polyu.edu.hk

Zhexue Huang, Junjie Li
Shenzhen University
College of Computer Science & Software Engineering
Shen Zhen, China
{zx.huang,jj.li}@szu.edu.cn

ABSTRACT

It is only natural that events related to a listed company may cause its stock price to move (either up or down), and the trend of the price movement will be very much determined by the public opinions towards such events. With the help of the Internet and advanced natural language processing techniques, it becomes possible to predict the stock trend by analyzing great amount of online textual resources like news from websites and posts on social media. In this paper, we propose an event attention network (EAN) to exploit sentimental event-embedding for stock price trend prediction. Specially, this model combines the merits from both event-driven prediction and sentiment-driven prediction models, in addition to exploiting sentimental event-embedding. Furthermore, we employ attention mechanism to figure out which event contributes the most to the result or, in another word, which event is the main cause of the price fluctuation. In our model, a convolution neural network (CNN) layer is used to extract salient features from transformed event representations, and the latter are originated from a bi-directional long short-term memory (BiLSTM) layer. We conduct extensive experiments on a manually collected real-world dataset. Experimental results show that our model performs significantly better in terms of short-term stock trend prediction.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**; • **Information systems** → **World Wide Web**; *Web applications*.

KEYWORDS

stock trend prediction; financial text mining; sentimental event embedding; attention-based deep learning

WebSci '19, June 30–July 3, 2019, Boston, MA, USA

ACM ISBN 978-1-4503-6202-3/19/06. . . $15.00
https://doi.org/10.1145/3292522.3326014

ACM Reference Format:
Yaowei Wang, Qing Li, and Zhexue Huang, Junjie Li. 2019. EAN: Event Attention Network for Stock Price Trend Prediction based on Sentimental Embedding. In *11th ACM Conference on Web Science, June 30–July 3, 2019, Boston, MA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3292522.3326014

1 INTRODUCTION

The stock trend prediction aims to predict the future trends of a stock in order to help investors to make good investment decisions. It has become a challenging problem because of highly volatile and non-stationary nature of the market [1]. Nevertheless, the pursuit of maximizing profits has attracted both stock investors and market researchers to make continuous attempts on stock movement prediction.

So far, existing prediction approaches can be divided into technical analysis based vs. peripheral analysis based. The technical analysis based approach is to make predictions on time-series historic market data. It is conducted by analyzing the patterns of historical and current market behavior. Three principles are usually employed: the market behaviors involve all relevant information; the prices move along with the trend; and the history will repeat. Engle [8] was the first to explore this approach. He adapted autoregressive conditional heteroscedasticity, and Taylor [21] applied stochastic volatility to construct time-series data modeling. However, the accuracy and confidence of such these approaches have their limitations, because they omit the interests-driven nature of the market, and neglect the influence on the stock trend from recent social events, new policies on the industry, public opinions and emotions. All of those have the potential to produce large amounts of abnormal transactions and unusual market behaviors, which lead to severe price fluctuations.

On the other hand, peripheral analysis based approaches focus on the impact of social events and real life news on the financial market. With the help of the Internet and the advanced natural language processing, researchers can access abundant instant online resources as references and be capable of processing the information efficiently in order to make more accurate in-time predictions. Two major sources, *i.e.*, news from financial websites and opinions from social media, are widely studied nowadays. Financial news receive more attention because most of the financial articles are written by professional analysts, so that explicit and in-depth views can be delivered from financial news, including interpretations on specific events that could have impacts on some companies' income or the confidence of general public, and new policies that may influence the future macroeconomy indicators or industry development. Many research works have been conducted on Wall Street Journal [2], Bloomberg [6], Yahoo Finance [19], *etc.* For example, Li *et.al* [16] applied a sentence-level summarization model to full-length financial news and built a prediction model on Hong Kong Hang Sang Index. Another popular text mining source is the social media. Though more noisy data (*e.g.*, ungrammatical twitter), the contents generated by public users are representative of the opinions and emotions from the majority of individual investors. Indeed, the public opinions extracted from social media are widely correlate to the stock movement. Gilbert *et.al* [9] showed the degree of public anxiety is a considerable factor of price movement by using Granger Causality Test. Bollen [4] explored seven emotional dimensions and revealed that the prediction results can be enhanced if the scores of Calm and Happiness are evaluated together with historical transaction data. Many researchers [4][25][3][22] made some breakthroughs using twitter texts for market prediction.

From our preliminary empirical studies, we have found that online textual contents from different sources function well in different ways. News contents are more suited than those from social media for event extraction and clustering, because news can tell a relatively complete story about a specific event and we can use the information provided in the news to index the event. However, due to the nature of professionalism (*i.e.*, neutral statements), it is difficult to label the public opinions of such news, especially (expert) financial news. The contents from social media and public forums indicate strong sentiment preference towards a single event, yet the textual descriptions from the social media usually are lexically incomplete, sometimes inconsistent, which makes them unsuitable for event extraction. To utilize the advantages of both sources, we propose a model to detect the events from multi-sourced textual contents and predict the trend of stocks based on related events and public sentiment. In particular, our approach is to conduct event-based sentiment analysis to reveal the causality relationship between the public opinions on social events and unusual market reactions.

The contributions of our work include the following:

(1) We propose a novel sentiment-driven event representation for predicting stock price movement, which is the first piece of work which introducing a sentimental event model into stock trend prediction.
(2) We develop an attention-based hybrid framework to better integrate extracted temporal event information with feature embedding, and exploit the short-term influence of sentimental events.
(3) We adopt Convolutional Neural Networks (CNNs) to handle the prediction task, and the experimental results on real-world datasets indicate the superiority of our model in comparison with several baselines.

The rest of this paper is organized as follows: Section 2 reviews the prior work on technical, event and sentiment analysis for stock prediction. The architecture of the proposed event attention network (EAN) is presented in Section 3. Section 4 provides the details of the datasets we use, and shows the performance of the proposed framework, compared with various baselines on real-world datasets. Finally, in Section 5 we discuss some characteristics of our framework and present the direction of our future research.

2 RELATED WORK

Stock trend prediction is a crucial reference for maximizing profits and avoiding risks. Continuous efforts with implementing various techniques, from machine learning methods to deep learning models, have been applied in stock trend prediction. Meanwhile, peripheral analysis based on online textual resources is also a hot research direction in recent years.

2.1 Technical analysis based prediction

Many machine learning techniques including support vector machine (SVM) [12][11][20] have been exploited to analyze time-series numerical data for stock trend prediction. Tay and Cao [20] modeled multiple features based on a five-day time window, and their experiments show that the SVM can outperform a back-propagate neural network. The limitation of this approach is that it cannot correctly react after significant events happen (for example, the tariff conflicts between China and the US), since it only takes the past numerical data into account. Another typical trend prediction model is the Autoregressive (AR) model [14], which is a time-series-based model. In particular, the AR model uses linear and stationary time-series data to predict the trend of next time step by looking at the current and previous statuses. However, due to the non-linear and non-stationary nature of the stock market, the AR model suffers from the same dilemma as SVM, so it performs terrible after some unprecedented market shocks occur. With the development of deep learning techniques, more attempts have been made on financial prediction. For instance, some recurrent models, especially, Long Short-Term Memory (LSTM) model [18] to better model the time dependency of historic stock data, have been deployed for financial prediction. Nevertheless, the reliability and accuracy of those technical analysis methods suffer from neglecting some outside-market factors, such as events and opinions.

2.2 Event analysis for stock price prediction

To seek the correlation between real life events and stock price turbulence, Yoshihara *et.al* [24] examined temporal effects of past significant events. For example, the prices of many stocks plummeted after the bankruptcy of Lehman Brothers. In their paper, RNN-RBM is adopted to automatically identify useful features from a large amount of texts. Jin *et.al* [13] detected events from Google search trend and

Twitter burst features, and mined the opinion factor from Bloomberg news. The Delta Naïve Bayes model is used to make predictions. Ding [7] used structured events with different time intervals to make stock movement predictions on S&P 500 Index and several individual stocks. Nascimento *et.al* [17] adopted structured-event embedding technique to extract events from news stream, and utilized the event embeddings as input to forecast the stock trend. However, their methods may neglect most of the small enterprises because of few related events to report.

2.3 Sentiment analysis for stock price prediction

Besides event-driven prediction, sentiment analysis prediction is also widely studied because public emotions and opinions about specific events can affect the stock movement more profoundly than events themselves. Li *et.al* [15] implemented a stock price prediction framework analyzing the sentiments from financial news. However, it is difficult to retrieve emotion labels from news articles because of the proficiency nature of the financial news, instead, strong emotion and opinion expressions can be identified from social media. Vu *et.al* [22] explored the correlation between public emotions and trend of stock movement on Twitter, and utilized sentiment features in financial prediction. Most recently, Hu *et.al* [10] designed an attention-based news-oriented stock trend perdition model based on the sequence of recent related news. Their experiment was conducted on Chinese stock market, and the results show that the confidence of prediction can be elevated considerably after the sentiment of related news is counted. While there have been many efforts in exploiting sentiment from single data source (*e.g.*, news) for stock predication, few of them considered the link between sentiment and events.

3 THE FRAMEWORK OF THE EVENT ATTENTION NETWORK

In this section, we start with characterizing the underlying perspective of events for the task of stock trend prediction, followed by a detailed description of our proposed framework.

3.1 Event Characteristics

To better model the influence of the events on relevant stocks, we address the challenges in the stock trend prediction using events, which can be summarized into following characteristics:

- Imbalanced distribution of events: during the same period of time, there may exist very few news for many small-volumed companies, but more frequent news reports for some big enterprises or companies under some critical and abnormal conditions, which can cause the event dictionary to be either too sparse or too dense for different stocks.
- Inconsistent effect of events: it is important for event processing to distinguish the effect of the same news on different stocks, especially those stocks across industries. For example, the news that "petroleum prices decline due to new energy source emerges" may have a positive effect on auto industry but negative effect on petrochemical industry.
- Distinct importance of events: when multiple events emerge together during a given time window, it is critical for the model to determine which event is the most significant and plays a vital role in stock trend prediction task.
- Different life-time of events: sequential and temporal properties can help reveal the inner relationships between or among the events. It is also vital to determine the life-time of influence, dependency, and causality relationships. However, these are difficult to capture when extracting the events.

Taking into account these characteristics, an ideal learning framework for stock trend prediction should have the ability to distinguish those events holding more intensive and durable influence. In the next subsection, we present our EAN approach as an attempt to address these issues holistically.

3.2 Model Description

We regard the problem of stock trend prediction as a classification problem. For a given date t, a given stock s and the target date (*e.g.*, $t+5$ days), we can calculate its rise percent by:

$$R(s,t) = \frac{Close\_Price_{t+5} - Close\_Price_t}{Close\_Price_t}, \quad (1)$$

where $Close\_Price_t$ and $Close\_Price_{t+5}$ represents the closing price of the *t-th* day and *t-th*+5 day, respectively. Similar to many previous studies, we use the price change to represent the stock movement in order to obtain a binary label (1 for price going up, and -1 for price going down):

$$L(s,t) = \begin{cases} 1 & R(s,t) > 0; \\ -1 & R(s,t) \leq 0. \end{cases} \quad (2)$$

The stock trend prediction task can be formulated as follows: given the length of a time sequence N, the stock s and date t, the goal is to use the news and social media corpora sequence from time $t-N+1$ to t, denoted as $[C_{t-N+1}, C_{t-N+2}, ..., C_t]$, to predict the class of $R(s,t)$ (*e.g.*, 5 days). Note that each corpus C_i contains a set of documents with the size of D, *i.e.*, $C_i = [n_{i,1}, n_{i,2} ..., n_{i,D}]$ denoting D related pieces of news on date i.

The architecture of our proposed Event Attention Network is shown in Fig. 1. It is composed of two parts: the Input Attention Mechanism, and Temporal Hidden Represent. The first part is to encode the event and sentiment information, and generate the contextualized event representation with the attention mechanism. At the beginning, an event embedding layer transforms the news corpus of *i-th* day into the event and sentiment vector with the size of L, *i.e.*,

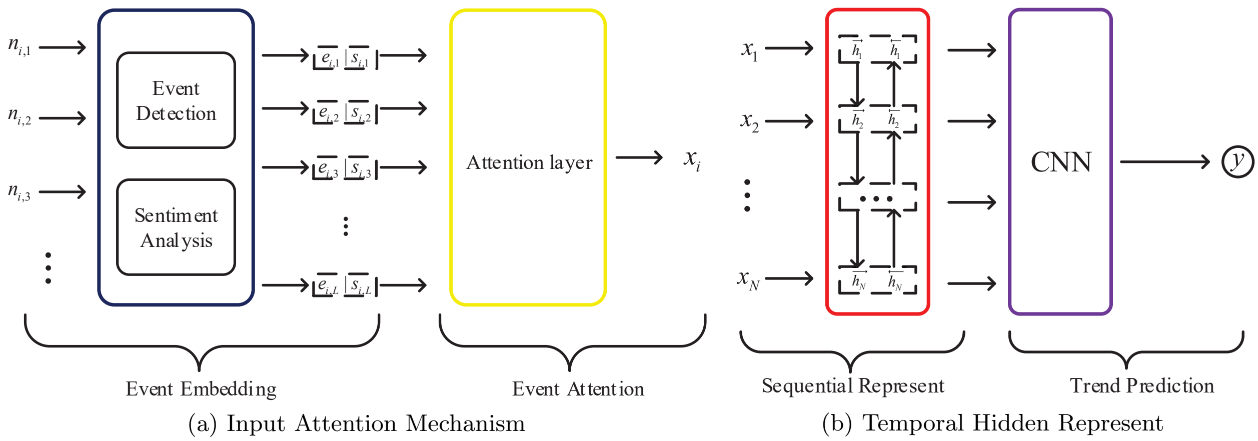


Figure 1: The overall framework of the Event Attention Network.

$[(e_{i,1}; s_{i,1}), (e_{i,2}; s_{i,2}), ..., (e_{i,L}; s_{i,L})]$ which denotes L related sentimental events on date i. An event attention layer is next assigned with an attention value for each event vector on date i, and the weighted mean of such event vectors are calculated as a feature vector x_i for this date. The second part is to predict the stock trend using these event vectors within N days, $[x_1, x_2, ..., x_N]$. These feature vectors are encoded by a BiLSTM, which results in the hidden sequential representation $[(\overrightarrow{h_1}; \overleftarrow{h_1}), (\overrightarrow{h_2}; \overleftarrow{h_2}), ..., (\overrightarrow{h_N}; \overleftarrow{h_N})]$. Finally, we adopt the CNN to produce the stock trend prediction result y, based on the sequential representation of the sentimental event information.

3.2.1 Event Embedding Layer. To tackle the problems of sparsity and uniqueness in the representation space of events, we devise a method for normalized sentimental event embedding representation. In particular, we calculate the sentimental indicators by the sentiment classifier (*e.g.*, Linear Regression). Based on the event tuple vectors and two dimensional emotion values, we then obtain $(e; s)$ called sentimental event representation which is the more effective for stock trend prediction.

Event Detection: Following the previous work [23], we reconstruct the related textual content with features representation by extracting the information like date, location, entities and actions. To handle the information properly, an unsupervised event dictionary (D_e^{Time}, $D_e^{Location}$, D_e^{Name}, D_e^{Action}) is built to retrieve the reconstruction representation (X_d^{Time}, $X_d^{Location}$, X_d^{Name}, X_d^{Action}) from daily related news. The new event dictionary is gifted with significant advantages over complexity and robustness, which makes it more efficient to process large amount of data and noisy data. The objective function with constraints of low rank and local invariance is as follows:

$$\min_{\tilde{E}_i, D_e^m} \sum_{d=1}^{D} \sum_{m=1}^{M} \|X_d^m - D_e^m \tilde{E}_i\|_F^2 + \gamma_1 \|\tilde{E}_i\|_F^2 + \gamma_2 tr(\tilde{E}_i \tilde{E}_i^T), \tag{3}$$

where M is the number of event components, $\|M\|_F$ is the *Frobenius* norm of matrix M, $\tilde{E}_i$ is the reconstruction of a set of daily events, and the terms γ_1, γ_2 represent the regularization parameters.

Secondly, a batch alignment model is designed to conduct events clustering and obtain the event dictionary. We adopt incremental event detection to automatically filter the content and reconstruct the representation. By introducing a specific optimization algorithm, we can obtain high-quality low-noise event representation from retrieved documents.

Sentiment Analysis: We construct the sentiment model with hierarchical neural networks. Furthermore, we adopt the research findings of Bollen *et.al.* [4], in which user moods are measured in terms of six dimensions (*Calm*, *Alert*, *Sure*, *Vital*, *Kind*, and *Happy*); the results indicate that the accuracy of stock prediction can be significantly improved by specific dimensions, namely, *Calm* and *Happiness*. When the *Calm* value and *Happiness* value are calculated by feeding relevant documents into the model, these two sentiment values are concatenated together and treated as two dimensional emotion vector of each document.

Sentimental Event Encoding: It is no strange that different stocks may share the same event, which makes the representation space for events very sparse. Thus, a sentimental event encoder is designed to encode the entire event set for tackling the sparsity and uniqueness problems. Specifically, the sentimental event encoding is obtained by aggregating sentiment values of the related documents, as detailed below. For each event $\tilde{e}_{i,l}$, the event-based *Calm* value of related news can be calculated on a daily basis according to the following:

$$\tilde{s}_{i,l}^C = \ln \frac{\zeta + N_{i,l}^p}{\zeta + N_{i,l}^n}, \tag{4}$$

where $N_{i,l}^p$ and $N_{i,l}^n$ represent the numbers of positive and negative documents related the event l on the i-th day, respectively; ζ is a tiny number for smoothing. Similarly, the event-based *Happiness* value has the indicator $\tilde{s}_{i,l}^H$. To enable fair ensemble for $\tilde{e}_{i,l}$ and $(\tilde{s}_{i,l}^C; \tilde{s}_{i,l}^H)$, we use the z-score to normalize the data within a sliding window of length N. The z-scores for $\tilde{e}_{i,l}$ is:

$$e_{i,l} = \frac{\tilde{e}_{i,l} - \mu(\sum_{i,l} \tilde{e}_{i,l})}{\sigma(\sum_{i,l} \tilde{e}_{i,l})}, \tag{5}$$

where μ and σ are the means and standard deviations of all $N \times L$ temporal events related to a stock, respectively. Similarly, we also normalize the sentiment values using z-score. For simplicity, we use the notation $s_{i,l}$ to represent the two dimensions of emotion values $(s^C_{i,l}; s^H_{i,l})$.

Note that the sentimental event representation $(e_{i,l}; s_{i,l})$ is more significant and robust than the original event tuple represent $\tilde{e}_{i,l}$. Supposing that two different stocks are influenced by the same event, they still may have distinct event representations because of different sets of events and sentiment values. Simultaneously, they also extend the representation space of events, which avoids the learning to stop early in the training process of the stock trend prediction model.

3.2.2 Event Attention Layer. Since not all events contribute equally to the movement of a stock price, we devise an attention mechanism to aggregate the events by an assigned attention value, *i.e.*, higher weights are assigned to the more significant events automatically. Specifically,

$$\begin{aligned} u_{i,l} &= \delta(W_{e;s}(e_{i,l}; s_{i,l}) + b_{e;s}) \\ \alpha_{i,l} &= \frac{e^{u_{i,l}}}{\sum_l e^{u_{i,l}}} \\ x_i &= \sum_{l=1}^{L} \alpha_{i,l}(e_{i,l}; s_{i,l}), \end{aligned} \tag{6}$$

where $u_{i,l}$ is the latent representation of encoded sentimental event vectors $(e_{i,l}; s_{i,l})$ through one-layer network with parameters $W_{e;s}$ and $b_{e;s}$ plus a sigmoid function δ. By combining them through a *softmax* layer, we can get a normalized attention weight $\alpha_{i,l}$ to distinguish the importance of different events. Finally, we calculate the overall event feature vector x_i by a weighted sum of each sentimental event vector, and use this vector x_i to represent all event expressions on the i-th date.

Consequently, we get a temporal sequence of feature vector $X = \{x_1, x_2, ..., x_N\}$ with N being the size of time sliding window. The attention layer can gradually learn to assign higher attention value to the reliable and important event based on its sentimental event information.

3.2.3 Bi-directional LSTM layer. To combine the temporal information of events in the convolution-based architecture, we employ a BiLSTM to accumulate the context information for each input event sequence. Belonging to RNN, LSTM is used as an event encoder to obtain a latent distributional representation for two main reasons: (1) The content lengths of events are usually different, and RNN is known to be good at dealing with the variable length problem; (2) LSTM has the ability to handle sequential input and semantics of events.

A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The compact form of the equations for the LSTM unit can be given as:

$$\begin{aligned} f_t &= \delta(W_f x_t + U_f h_{t-1} + b_f) \\ i_t &= \delta(W_i x_t + U_i h_{t-1} + b_i) \\ o_t &= \delta(W_o x_t + U_o h_{t-1} + b_o) \\ c_t &= f_t \circ c_{t-1} + i_t \circ \delta(W_c x_t + U_c h_{t-1} + b_c) \\ h_t &= o_t \circ \delta(c_t), \end{aligned} \tag{7}$$

where t denotes a given date, f_t denotes the forget gate controlling how much past information should be kept, i_t is the input gate deciding how much new information should be added, and o_t means the output gate expressing how much information should be exhibited. Particularly, c_t represents the state of current time, controlling how mush past state should be used for updating the new state. Finally, the hidden represent h_t is linearly computed by the current state and output.

Therefore, we can get the latent representation for the t-th day through LSTM. In order to capture the information from the past and future of an event, we construct a bi-directional LSTM encoding h_i as follows:

$$\begin{aligned} \overrightarrow{h_i} &= \overrightarrow{LSTM}(x_i), i \in [1, N] \\ \overleftarrow{h_i} &= \overleftarrow{LSTM}(x_i), i \in [1, N] \\ h_i &= [\overrightarrow{h_i}; \overleftarrow{h_i}]. \end{aligned} \tag{8}$$

The result h_i incorporates the information of both its surrounding events and itself. Thus we transfer the sentimental event to temporal sequence representation.

3.2.4 Convolutional Feature Extractor. We adopt a CNN to extract stock-driven feature representation. We know that big stock price fluctuation is usually determined by key events. The CNN is therefore preferred as the top-level prediction module for its capability of extracting the local features. Note that, the same event might cause different effects on different stocks. Thus, we must take stock company information into consideration to construct event relevance. Specifically, we first calculate the relevance v_i between the collection of events on *i-th* day and the target stock prices:

$$v_i = \begin{cases} \frac{e^{r_i}}{\sum_i e^{r_i}} & |p_i - p_A| > \epsilon \\ 0 & |p_i - p_A| \leq \epsilon, \end{cases} \tag{9}$$

where p_i and p_A are the target stock price of i-th day and average value of N days, respectively, ϵ is a pre-specified constant to avoid flat stock, and $r_i = w_r(h_i; p_i) + b_r$ is a linear transformation between events and target stock price. Then, for a given stock, we use the relevance v to help CNN locate the key event:

$$h_i^{conv} = h_i * v_i, i \in [1, N]. \tag{10}$$

We feed the weighted representation using relevance to the convolutional layer to generate the feature map, as follows:

$$m_i = ReLU(\mathbf{w}_{conv}^T \mathbf{h}_{i:i+s-1}^{conv} + b_{conv}), \tag{11}$$

where $\mathbf{h}_{i:i+s-1}$ is the concatenated vector of $h_i, ..., h_{i+s-1}$ and s is the kernel size, $\mathbf{w}_{conv}$ and $\mathbf{b}_{conv}$ are learnable weights of the convolutional kernel. To capture the most informative features, we apply max pooling and obtain the latent feature z by employing k kernels:

$$z = [max(m_1), max(m_2), ..., max(m_k)]^T. \quad (12)$$

Finally, we pass z to a *softmax* output layer for the final stock trend prediction:

$$p(y|z) = Softmax(W_f z + b_f), \quad (13)$$

where W_f and b_f are learnable parameters.

4 PERFORMANCE EVALUATION

In this section, we firstly describe the collecting process of real-world data. Then, we present the experimental setup, and conduct comprehensive experiments to verify the effectiveness of our proposed Event Attention Network (EAN).

4.1 Datasets

To make the experimental results and conclusions convincing and close to the real market, two groups of datasets from Shenzhen (SZ) and Hong Kong (HK) markets are collected and used to evaluate our framework. These datasets are from the following three data sources:

(1) Stock Price Data: We collect stock price data from Yahoo Finance by employing different strategies for the two markets. For HK market, we select 10 famous companies because their names frequently appeared in the news. Most of these stocks tend to be quite stable when compared with other stocks. To test the effectiveness of our model for different stock market, we randomly select 10 balanced stocks from the SZ market. The balanced stocks have the following conditions: fluctuating market performance, ample amount of related news, and diverse public opinions. Interestingly, these companies are listed as small or medium-sized enterprises. In general, these potential stocks are more difficult to predict[1].

(2) Financial News Data: We obtain relevant news documents from news publishing websites, namely, Hong Kong news from the Finet, and Shenzhen stock news from the Tencent News and Sina News. These financial news documents fall into the period from April 2015 to April 2018. In total, there are 202,156 documents from HZ stocks and 145,714 from HK stocks. News titles and contents are extracted from HTML. The timestamps of the news are also extracted. As our focus is on the 20 stocks from SZ and HK, queries against these 20 stocks can be easily issued and processed based on stock id, name, symbol of company, and so on.

(3) Sentiment Corpus Data: For the sentimental comments on financial news, we obtain the data from an influential social medial platform called the East Money Forum, which is one of the biggest and specialized stock forums. Different from public forums like Facebook or Twitter, each stock has its own sub-forum to ensure that most posts there are published by the investors who hold or sell that particular stock.

[1] It is more reasonable to assume that people would be more interested in investing such potential stocks, because the famous stocks are usually very expensive and less fluctuations, which lend to less investment opportunities.

We automatically align the 20 stocks of daily trading data with news titles and contents, 2/3 of which are used for training, 1/6 for validating, and 1/6 for testing. The statistics of the datasets are summarized in Table 1. The training, validating and testing datasets are split temporally, with the data from 01/04/2015 to 31/03/2017 for training, the data from 01/04/2017 to 31/09/2017 for validating, and the data from 01/10/2017 to 01/04/2018 for testing.

Table 1: Statistics of the datasets.

		Training	Validation	Testing
#documents	SZ	134,770	33,692	33,694
	HK	97,142	24,285	24,287
#positive labels	SZ	8,652	1,344	1,421
	HK	11,345	2,684	2,533
#words	SZ	83,321,869	27,749,044	27,764,352
	HK	65,732,769	23,744,578	23,766,825
#avg events per day	SZ	13	8	11
	HK	34	31	28
time interval	SZ & HK	01/04/2015-31/03/2017	01/04/2017-31/09/2017	01/10/2015-01/04/2018

4.2 Experimental Setup

In the following experiments, the evaluation metrics are the Accuracy and MacroAveraged F1, the latter is more appropriate for datasets with unbalanced classes. We compare our results with the performance of the following methods on the short-term (5 days) prediction:

- RAND: a naive predictor ignoring all the news, simply adopting random guess for up (+1) or down (-1) at the chance of 50%. We also use this model to check the balance of our datasets.
- ARIMA [14]: Autoregressive Integrated Moving Average, a traditional analysis method using only price information.
- SVM [20] a classical model with extensive feature engineering. To construct the input, a corpus vector for one day is used by averaging all news vectors of a certain date, and the corpus vectors in N days (with the same window size) are concatenated.
- MLP [5]: the multi-layer perceptron (MLP) classifier, which has five layers sized as 500, 400, 300, 200 and 100, respectively. The input of MLP is the same as that in SVM.
- HAN [10]: a state-of-the-art deep neural network with hierarchical attention which predicts stock trend by using a sequence of recent related news through a self-paced learning mechanism.

In order to make a detailed analysis of all the main components of EAN, we also construct the following three variations:

- EAN-SVM: It uses the SVM as a replacement of the CNN in EAN.
- EAN-MLP: The model changes the final decision layer of EAN to MLP.
- EAN-E: It only uses news event information, the purpose of which is to test the effectiveness of the sentimental event embedding.

We tokenize each news and remove the stop words. For out-of-vocabulary words, we randomly sample their embeddings from the uniform distribution. Then, we tune our model by the validation set to find the best fitted hyper parameters. Table 2 summarizes the final hyper-parameters combinations which strive to balance the speed and prediction accuracy.

Table 2: Setting of hyper-parameters.

Hyper-parameters	EAN-E		EAN	
	SZ	HK	SZ	HK
ζ	0.0001		0.01	
ϵ	0.01	0.3	0.05	0.1
L	3		15	
dim_h^{lstm}	300		300	
dim_h^{cnn}	50		50	
batch size	1024	512	512	256
dropout rates (p_{lstm}, p_{cnn})	(0.3,0.6)		(0.5,0.5)	
kernel size k	40		40	
learning rate	0.001	0.1	0.001	0.01

4.3 Main Results

As shown in Table 3, EAN consistently achieves the best performance on all the datasets, which verifies the efficacy of whole EAN framework. Moreover, EAN can perform well for different kinds of news contents, such as news with relatively formal sentences, and reviews with ungrammatical sentences. The reason is due to its CNN-based predictor which enables EAN to extract more accurate latent features from ungrammatical sentences.

Table 3: Overall experimental results

	Models	SZ		HK	
		ACC	Macro-F1	ACC	Macro-F1
Baselines	RandomGus	51.32	53.27	43.24	45.48
	ARIMA	53.27	52.36	51.93	49.38
	SVM	55.86	58.31	57.23	51.86
	MLP	56.11	56.85	58.35	52.95
	HAN	65.77	63.20	64.29	62.96
EAN variants	EAN-SVM	59.33	57.54	58.33	55.40
	EAN-MLP	61.32	59.35	64.36	63.57
	EAN-E	62.37	60.58	63.52	61.40
	EAN	**66.80**	**70.32**	**65.55**	**67.38**

On the other hand, the performance of the baseline methods varies for the two different markets. For RandomGus, we can see that its performance on the stocks of SZ is closer to the norm (50%). For the "famous" companies in HK, however, their stocks are mostly stable. Thus, RandomGus cannot perform as effectively as it does for SZ stocks. Note that the Macro-F1 measure is more significant and indicative for such unbalanced data.

We can also make some further observations and conclusions based on the overall experimental results, as follows:

1. LSTM-based models relying on sequential information can perform well for financial news by capturing more useful context features.
2. For ungrammatical text and sentimental reviews, CNN-based approaches have some advantages because CNN aims to extract the most informative features, and is less sensitive to informal texts without strong sequential patterns.
3. The news-based methods (EAN-E and EAN) perform better than those price-based methods.
4. The sentimental information is useful in improving the news-based models (ref. EAN vs. EAN-E).
5. Different from HAN [10], the sentimental event embedding is employed by our EAN to represent the news features, which enables EAN to outperform HAN for the two markets under all measures.

4.4 Effects of window size

A natural and interesting question related to events is how long the life cycles of these events are in terms of influence. In other word, what is the effect of the time window for such events? We investigate two alternatives here: event embedding only (*i.e.*, EAN-E) and event with sentimental information (*i.e.*, EAN). Table 4 shows the experimental results of these two alternatives.

As can be seen, both EAN-E and EAN perform the best when the window size N of SZ market is about $10 \sim 11$. In contrast, the effect of the historical news only lasts, on average, 4 days for HK market. The major reason as we observing is that news from big companies are more frequent than small unknown enterprises. Thus, the "famous" companies have a shorter event life cycle.

The lag window also varies for the individual stocks. We can see that events get updated faster for the more popular domains. For example, the 0700.HK is a famous technological company, whose effective window size is only one day. On the contrary, the financial enterprise 1398.HK needs to feed more historical events in order to achieve good performance. Also note that the lag window is mainly the property of specific stocks, *i.e.*, no matter EAN-E (without sentiment information) or EAN (with sentiments), the average lag window remains similar to each other.

4.5 Performance of Sentimental Event

To investigate the impact of sentimental event features, we look at Table 3 again by comparing EAN and EAN-E. In general, the sentimental information improved the accuracy by 4.43% and 2.03% in SZ market and HK market, respectively.

Table 4: Performance and the best N for different stocks

		EAN-E			EAN		
	Stocks	N	ACC	Macro-F1	N	ACC	Macro-F1
SZ	000573	11	57.33	58.64	10	61.30	64.11
	000886	10	52.40	55.41	8	67.75	68.34
	100017	12	60.50	58.47	12	61.30	60.17
	159938	11	64.55	60.26	11	66.54	60.50
	163819	13	69.82	67.88	12	71.64	70.45
	168301	9	62.32	60.58	9	62.32	60.58
	300031	15	60.29	57.63	13	64.68	65.25
	300488	10	55.43	58.56	12	60.45	64.12
	200056	11	60.17	60.54	8	63.27	64.91
	300622	10	73.44	70.66	10	75.11	74.33
	Avg	**11.5**	**61.63**	**60.86**	**10.5**	**65.44**	**65.28**
HK	0700	1	76.40	74.82	1	80.65	81.51
	0939	1	76.25	72.54	1	78.55	79.71
	0941	3	58.73	60.38	3	59.65	61.33
	0005	4	63.50	63.22	3	65.94	65.88
	1299	5	52.73	55.56	8	61.74	64.17
	2318	3	72.60	67.39	3	73.28	70.66
	1398	9	66.50	64.82	9	66.50	64.85
	0883	4	65.20	61.69	5	66.45	61.55
	2388	2	66.40	63.97	4	68.20	70.53
	3333	11	57.95	58.35	9	63.21	64.88
	Avg	**4.3**	**65.63**	**64.27**	**4.6**	**68.42**	**68.51**

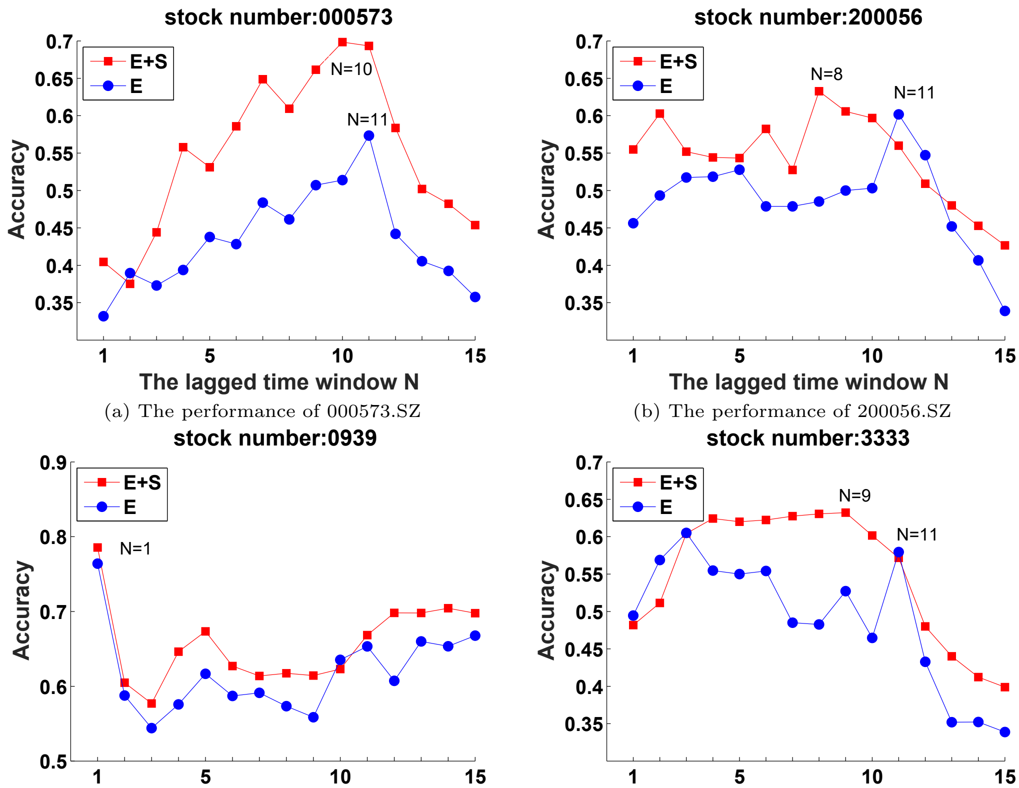
Furthermore, EAN has some distinguished improvements on the Macro-F1 measure: it has 70.32% and 67.34% Macro-F1 value for SZ and HK, respectively. The reason is that many stock prices often maintain a stable state for a certain period of time, *i.e.*, they do not flip up and down everyday. Thus, a crucial task is to predict more accurately the turning point after a stable state. Lacking of the sentimental information, the attention values of events in EAN-E make a little difference. Such an event-only model tends to predict the same label with a stable historical state, resulting in unbalanced labels hence leading to a lower Macro-F1 value. But the sentimental event is more sensitive to new events. Our model assigns a higher attention value for a bursting event with the fluctuation of public sentiments, and this applies to each individual stock. As shown in Table 4, EAN has higher Macro-F1 values than EAN-E does. We can also find that for stocks on which the event-only model (*i.e.*, EAN-E) performs poorly, adding sentiment information can achieve obvious improvement. For example, the 000886 in SZ and 1299 in HK get improved by 15.35% and 9.01% accuracy, respectively, *i.e.*, from 52.40% (by EAN-E) to 67.75% (by EAN), and from 52.73% (by EAN-E) to 61.74% (by EAN).

Fig 2 shows some intuitive examples, from which we can see that sentimental events help achieve better results, especially when N is near the optimal setting. More notably, in Fig 2 (b), the blue curve lags behind the red curve for about 3 days, which shows that the added sentiment information is more helpful in capturing new burst events. In other word, the events with sentimental information can be captured faster. The approach is also more robust. From Fig 2 (d), we can see that the blue curve improves sharply when N increases towards 9. For a large N, the performance of EAN-E becomes more sensitive, probably due to that more events may involve more noisy and fake information.

4.6 Case Study

Table 5 provides some examples for case study. To show the relationship between events and stocks more clearly, we change the event tuples to natural sentences, and symbolize the sentimental values by using the notations P and N in the table, representing positive and negative, respectively. For each sentiment, we highlight the label with a particular color (red for Happiness and blue for Calm), and the most informative event feature captured by EAN-E is the same as that by EAN, very similar to news title in HAN. To test whether our framework can select significant events and filter out those less informative ones, we also show some detailed results of these (normalized) event attention values. The final prediction uses the binary label pos and neg to indicate whether the price goes up or down.

By capturing news features correctly, all the three methods (HAN, EAN-E, EAN) can predict stock price trend accurately for the first stock 0700.HK which is a famous company with many news articles everyday. It is a relatively simple task to extract key information from the news. For the second stock, its first and second most informative events did



(a) The performance of 000573.SZ (b) The performance of 200056.SZ (c) The performance of 0939.HK (d) The performance of 3333.SZ

Figure 2: The influence of different size of lagged window for events and sentimental events.

Table 5: Example predictions. ✓indicates correct prediction.

Stock	Sentimental Events	Attention	HAN	EAN-E	EAN
0700.HK	Naspers sells out Tencent shares (N,N)	0.54	*neg*✓	*neg*✓	*neg*✓
	Tencent buys out New Classics Media (N,P)	0.14			
	Tencent video reached 62.59 million members (P,P)	0.13			
	Others $(-,-)$	0.19			
000573.SZ	One big transaction occurred (P,N)	0.71	*neg*	*pos*✓	*pos*✓
	New policy for free trade (P,N)	0.23			
	DongGuan Winnerway Industrial Zone was fined (N,N)	0.03			
	Others $(-,-)$	0.03			
0005.HK	HSBC publish the Annual Report (P,P)	0.36	*neg*	*neg*	*pos*✓
	HKMA support the HK$ (P,P)	0.36			
	HSBC applies blockchain to trade finance (N,P)	0.17			
	Others $(-,-)$	0.11			

not mention the stock and company. Both EAN-E and EAN can still make correct prediction even when the features are hidden. For the third stock which usually does not publish any unusual information except annual report, EAN can still make correct prediction while the other two cannot. As reported in [4], the important event sentiments to stocks are the positive Happiness and positive Calm. The annual report is very beneficial to its stock holders who either feel happy or at least calm. Our sentimental event embedding, armed with the event-level feature representation with sentimental information, is capable of handling such cases.

Another useful facility of our framework is the temporal event attention mechanism. In general, events happened recently have higher impact on the current stock price movement than those happened earlier. However, there may exist some events which affect a specific stock for a long period of time. For example, the third event in stock 0700.HK is the latest related event, but the attention value of the first event dominates all events, which likely leads to the final prediction correctly. This example also demonstrates a phenomenon that if a company loses some important share holders, the stock price of this company will fall in next several days. Another interesting observation is on financial enterprises. The result indicates that economic policies announced by the government tend to have higher attention values, hence have a great influence on the financial companies, especially on banks.

5 CONCLUSIONS

In this paper, we have proposed a sentimental event embedding approach called EAN for short-term stock price trend prediction, which can generate event representation integrated with not only temporal effect but also sentiment attributes. The attention mechanism is adopted to assign a weighted value to each event according to its significance on stock price movement. A Bi-directional LSTM layer is deployed to figure out the sequential dependency of a series of events. A CNN layer is deployed to extract both event features and sentiment features from the encoded information, and those features are fed to the convolutional layer to make the predictions of stock movement. The experimental results show the superiority of our model, as well as the efficacy of its different modules, thereby verifying the rationality of the EAN architecture.

For our future work, we will investigate on how to address the sparsity of stocks-related news for more accurate stock price trend prediction. And we plan to extract events from multiple data sources. In our current experiment, many significant events are found from the social media and entertainment news rather than finical news. Moreover, the posts from the social media may contain a lot of fake and noisy information that may confuse the general public. These issues will be considered by our subsequent research.

ACKNOWLEDGMENTS

The research presented in this article has been supported by an ITF grant from the Innovation and Technology Commission (Project No. GHP/036/17SZ) and Shenzhen-Hong Kong Technology Cooperation Fund (Project No. SGLH2016 12091011009 26).

REFERENCES

[1] Klaus Adam, Albert Marcet, and Juan Pablo Nicolini. 2016. Stock market volatility and learning. *The Journal of Finance* 71, 1 (2016), 33–82.

[2] Werner Antweiler and Murray Z Frank. 2004. Is all that talk just noise? The information content of internet stock message boards. *The Journal of finance* 59, 3 (2004), 1259–1294.

[3] Johan Bollen and Huina Mao. 2011. Twitter mood as a stock market predictor. *Computer* 44, 10 (2011), 91–94.

[4] Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. *Journal of computational science* 2, 1 (2011), 1–8.

[5] Robert Goodell Brown. 2004. *Smoothing, forecasting and prediction of discrete time series.* Courier Corporation.

[6] Arjun Chatrath, Hong Miao, Sanjay Ramchander, and Sriram Villupuram. 2014. Currency jumps, cojumps and the role of macro news. *Journal of International Money and Finance* 40 (2014), 42–62.

[7] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 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)*. 1415–1425.

[8] Robert F Engle. 1982. Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica: Journal of the Econometric Society* (1982), 987–1007.

[9] Eric Gilbert and Karrie Karahalios. 2010. Widespread Worry and the Stock Market.. In *ICWSM*. 59–65.

[10] Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. [n. d.]. 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*. ACM, 261–269.

[11] Wei Huang, Yoshiteru Nakamori, and Shou-Yang Wang. 2005. Forecasting stock market movement direction with support vector machine. *Computers & Operations Research* 32, 10 (2005), 2513–2522.

[12] Kyoung jae Kim. 2003. Financial time series forecasting using support vector machines. *Neurocomputing* 55, 1-2 (2003), 307–319.

[13] Fang Jin, Wei Wang, Prithwish Chakraborty, Nathan Self, Feng Chen, and Naren Ramakrishnan. 2017. Tracking multiple social media for stock market event prediction. In *Industrial Conference on Data Mining*. Springer, 16–30.

[14] Lili Li, Shan Leng, Jun Yang, and Mei Yu. 2016. Stock Market Autoregressive Dynamics: A Multinational Comparative Study with Quantile Regression. *Mathematical Problems in Engineering* 2016 (2016).

[15] Xiaodong Li, Haoran Xie, Li Chen, Jianping Wang, and Xiaotie Deng. 2014. News impact on stock price return via sentiment analysis. *Knowledge-Based Systems* 69 (2014), 14–23.

[16] Xiaodong Li, Haoran Xie, Yangqiu Song, Shanfeng Zhu, Qing Li, and Fu Lee Wang. 2015. Does summarization help stock prediction? A news impact analysis. *IEEE intelligent systems* 30, 3 (2015), 26–34.

[17] Janderson B Nascimento and Marco Cristo. 2015. The Impact of Structured Event Embeddings on Scalable Stock Forecasting Models. In *Proceedings of the 21st Brazilian Symposium on Multimedia and the Web*. ACM, 121–124.

[18] Akhter Mohiuddin Rather, Arun Agarwal, and VN Sastry. 2015. Recurrent neural network and a hybrid model for prediction of stock returns. *Expert Systems with Applications* 42, 6 (2015), 3234–3241.

[19] Robert P Schumaker, Yulei Zhang, Chun-Neng Huang, and Hsinchun Chen. 2012. Evaluating sentiment in financial news articles. *Decision Support Systems* 53, 3 (2012), 458–464.

[20] Francis EH Tay and Lijuan Cao. 2001. Application of support vector machines in financial time series forecasting. *omega* 29, 4 (2001), 309–317.

[21] Stephen J Taylor. 2008. *Modelling financial time series.* world scientific.

[22] Tien-Thanh Vu, Shu Chang, Quang Thuy Ha, and Nigel Collier. 2012. An experiment in integrating sentiment features for tech stock prediction in twitter. (2012).

[23] Hang Yang, Yubo Chen, Kang Liu, Yang Xiao, and Jun Zhao. 2018. DCFEE: A Document-level Chinese Financial Event Extraction System based on Automatically Labeled Training Data. In *Proceedings of ACL 2018, System Demonstrations*. 50–55.

[24] Akira Yoshihara, Kazuhiro Seki, and Kuniaki Uehara. 2015. Leveraging temporal properties of news events for stock market prediction. *Artificial Intelligence Research* 5, 1 (2015), 103.

[25] Yang Yu, Wenjing Duan, and Qing Cao. 2013. The impact of social and conventional media on firm equity value: A sentiment analysis approach. *Decision Support Systems* 55, 4 (2013), 919–926.