Soft Computing (2021) 25:4433–4450
https://doi.org/10.1007/s00500-020-05454-x

METHODOLOGIES AND APPLICATION

Sentiment analysis of stock markets using a novel dimensional valence–arousal approach

Jheng-Long Wu[1] · Min-Tzu Huang[1] · Chi-Sheng Yang[1] · Kai-Hsuan Liu[1]

Published online: 3 January 2021

Abstract

Sentiment analysis has been used in many studies to predict stock market trends. Current sentimental analysis approaches focus only on the upward and downward movement of the price, which is not sufficient for more precise prediction of stock sentiments. Previous studies have focused on the trend (valence) regarding stocks because it represents the upward and downward trend of a stock. However, investors should be attention on trading (arousal) because the stock price increases or decreases even quickly or slowly; they may not to do trade. Therefore, this paper applies the concept of dimensional valence–arousal to define sentiments which intensities of trading and trends of the stock market for data annotation. The paper proposes a deep learning prediction model to predict the intensities of stock dimensional valence–arousal (SDVA), called the hierarchical title–keyword-based attentional hybrid network (HAHTKN). The paper modifies the hierarchical attention network (HAN) to our proposed HAHTKN model, comprising (1) a title encoder, (2) a keyword encoder, (3) a title–keyword encoder, (4) word-level encoder, (5) a sentence-level encoder and (6) stock DVA prediction layer. The experimental results show that the proposed HAHTKN prediction model for the SDVA task outperformed other baseline machine learning models and HAN-based models.

Keywords Deep learning · Stock sentiment analysis · Dimensional valence–arousal · Bidirectional gated recurrent unit · Attention mechanism

1 Introduction

Big data analysis is used to systematically extract critical information for crucial decision making in fields such as the medical industry (Carnicero and Rojas 2019), manufacturing (Shang and You 2019) and business. Big data analysis has also been applied to the stock market. Typically, statistical analysis and machine learning (ML) methods are used to extract trading rules from trading information such as stock prices and trading volumes. Based on these rules, investment advice is provided to investors (Patel et al. 2015).

Natural language processing (NLP) for text-based stock market information extraction has received considerable attention in decade because it has achieved satisfactory results (Ding et al. 2016; Feuerriegel and Gordon 2018). Typically, investors refer to the processed textual information to make investment decisions. The stock market is dynamic and changes constantly. In this digital age, information about stocks is available online across numerous websites and social media platforms. Therefore, many researchers have focused on text-based data to accurately predict stock market behavior (Wu et al. 2019a).

Financial reports and online stock forums provide crucial updates about the stock market. Several studies have been conducted on using online information to predict stock trends (Zhang et al. 2018b; Yadav et al. 2019). Agarwal et al. (2019) studied stock trends from 1992 to 2017 and concluded that sentiment analysis of views on stocks from online sources can help investors make investment decisions. Several studies have focused on identifying the trends of message such as stock news using sentiment analysis for stock market analysis and forecasting, predicting stock prices (Kim et al. 2018; Shah et al. 2018; Vanstone et al. 2019) and determining trading

Communicated by V. Loia.

✉ Jheng-Long Wu
jlwu@gm.scu.edu.tw

[1] School of Big Data Management, Soochow University, Taipei City, Taiwan

strategy design (Stefan and Helmut 2014). However, these studies have only focused on the price trend of the stock. An increase in the price indicates positive sentiment and vice versa, and positive sentiment indicates the potential of the price of the stock to increase further (Wu et al. 2014). For example, if a stock exhibits a positive trend and limited fluctuations, then investors do not sell the stock because they will expect the stock price to increase in the future and thus hold the stock. In this case, the sentiment is at the point of high valence with low arousal. Thus, such a state indicates uptrend and delayed trading. The valence considers the intensity of positive and negative sentiment, and arousal considers the intensity of calm and excitement (Warriner et al. 2013). The dimensional method is used to express the emotional state as a multidimensional continuous value, which results in accurate intensity. Therefore, the stock dimensional valence–arousal (SDVA) task for stock market is essential analysis. For an effective prediction model for SDVA task, accurate sentiment prediction of text is necessary. Deep learning (DL) algorithms extract complex data at a high level of abstraction, resulting in accurate text analysis. The hierarchical attention network (HAN) model can be used to extract textual information through a hierarchical coding process. The model retains contextual relevance markers, resulting in high classification performance (Gao et al. 2018).

Although stock market trends have often been analyzed, dimensional sentiment analysis has seldom been applied to stock market analysis. Therefore, we performed dimensional sentiment analysis to the Taiwan stock market to define stock sentiment status and artificial intelligence (AI) techniques to solve the SDVA task of the stock market. A modified HAN model exhibited superior prediction performance concerning SDVA for stock market and provided accurate information for investment decisions. Therefore, the stock dimensional valence–arousal prediction will be solved by our proposed hierarchical title–keyword-based attention hybrid network (HAHTKN). There are some hypotheses including our proposed HAHTKN can be improved prediction performance compared with baseline machine learning models; title–keyword-based attention mechanism can be reduced the prediction error compared with the common attention mechanism. Besides, this paper has some limitations such as the dataset in this paper is created by this research which is the first time to build stock valence–arousal indicators with real value. So that there is no benchmark dataset to evaluate our proposed model. Another is that the attention mechanism in this paper needs to use a major feature to attention another feature so that not only content in stock news can provide information but also the title or keyword is very important information compared to content.

The contributions of the proposed method are as follows:

1. Using a dimensional valence–arousal approach for stock sentiment analysis of traditional Chinese text regarding the Taiwan stock market. The goal is to define SDVA for data annotation and the SDVA indictors can assist investors by providing them with effective investment advice.
2. The HAHTKN model was used to fit the sentiment analysis of the SDVA task on stock message. The model included the six sub-models such as title encoder, keyword encoder, title–keyword encoder, word-level encoder, sentence-level encoder and the SDVA prediction layer. Therefore, the estimating processes of the proposed model are similar to human decision making. More complete extraction of text features is superior to the extraction of other baseline models.
3. An annotation data set was derived for the dimensional valence–arousal task for stock news.

The remainder of this paper is organized as follows. Section 2 reviews the literature including stock prediction, sentiment analysis and deep learning; Sect. 3 describes the proposed HAHTKN model for trend and trading prediction of stock; Sect. 4 gives the experimental results, including model performance comparisons and statistical test; conclusions and directions for future work are presented in Sect. 5.

2 Related work

In this section, we review the literature on the development and application trends in stock prediction, sentiment analysis and deep learning.

2.1 Stock prediction

Stock markets play a crucial role in the economy because investors can share the profits of the company or share the losses of the company by buying and selling the stocks. To maximize the profit in stock market for investors, many researchers have conducted several methods to predict trends of stock market. Depending on the type of data assessed, stock market prediction is basically divided into technical analysis and fundamental analysis, which are typically based on time-series stock transaction information and text-based information (Islam et al. 2018; Nti et al. 2020). Between these methods, the technical analysis approach has always been a crucial part of stock market prediction because of its measurability and credibility. The technical indicators have been used in several researches to predict future stock price or trend. A Xuanwu system was proposed by Zhang et al. (2018a) for predicting upward, downward and flat stock trends. The Xuanwu system uses the random forest model and was developed through technical analysis of a 7-year period of data from the Shenzhen Growth Enterprise Market in China to improve the accuracy. A particle swarm optimization (PSO) algorithm was used with a neural network (NN) model to predict stock price direction and create initial weights of the NN. This approach reduced the time required for computation by the PSO algorithm (Chiang et al. 2016). These researches have built an automatic system to predict the stock market which is using automatically generating training samples and meaningful features as technical indicators. However, technical analysis still is an important approach in the stock prediction task.

With the development of the Internet, NLP and ML techniques, extensive text-based information regarding the stock market can be processed automatically, which enhances the power of fundamental analysis. Many studies have verified the stock market forecasting capabilities of unstructured stock messages. For example, experimental results in (Ingle and Deshmukh 2017) indicated that their purposed HMM model achieves improved accuracy with extracting term frequency-inverse document frequency (TF-IDF) features from online news sources. Also, many studies have been used some features to predict stock market performance such as news articles related to the financial market (Shi et al. 2018), assessing political situations (Khan et al. 2019) and social media (Saumya et al. 2016). Ding et al. (2016) proposed a model that employed a knowledge graph to extract event embedding, and the results demonstrated that this method predicted stock market volatility accurately.

However, several researchers have used both structured data, such as technical indicators, and unstructured data, such as news articles, to forecast stock market performance (Chen et al. 2016a). Gálvez et al. (2017) applied the same classification system with multiple combinations of technical indicators and stock message board information and demonstrated that text-based information can improve the performance of the classification model. Wang et al. (2017) used Stanford Parser to build a grammar tree that helps authors to identify the correspondence between core words and their sentiment values. But these methods may not robust enough and ignore the structure of the whole contents. Shi et al. (2018) proposed a deep neural network model to hierarchically deconstruct news headlines, trying to let computers learn the way people read.

In summary, these studies suggest that text-based data can be used in stock market prediction and emphasize the reliability of fundamental analysis to some extent. Several researchers have used both structured data, such as technical indicators, and unstructured data, such as news articles, to forecast stock market performance. Thus, unstructured data can be used to evaluate market reaction, and this improves the predictability of the stock market. However, sentiment analysis plays a crucial role in the use of unstructured data.

2.2 Sentiment analysis

The emotional response of the public toward the market as a whole can be understood using text-based stock messages. This helps investors predict stock trends. Therefore, analyzing the mood of the public in stock market prediction has been a topic of considerable concern in the research community. With the emergence of social media and sentiment-tracking techniques, many researchers extract sentiments from websites such as Yahoo! Finance (Ranco et al. 2016) and social networking sites such StockTwits (Batra and Daudpota 2018). In social media platforms, stock trend prediction based on expressions of sentiment made on Twitter has attracted considerable attention from many researchers.

Many researchers classify the sentiment extracted from the public emotions into categories, such as positive, neutral and negative, and believe that most positive sentiment means that the stock trend is bullish and vice versa. For instance, researchers (Pagolu et al. 2016) build a sentiment analyzer especially for identifying stock market sentiment which classified the sentiment into 3 classes. The research (Chen et al. 2016b) assigns the respective labels microblogs into 7 emotions (Happiness, Good, Sadness, Surprise, Fear, Disgust and Anger) and selected the emotions which appeared most times in each microblog. And their results showed that only “Happiness” and “Disgust” have significant causative relationships with stock prices movement. Different from the two studies categorized public sentiments into positive and negative with five sub-classes (Joy, Anger, Disgust, Fear and Sadness), but also considered the Bullishness which is greater if the number of positive microblogs increases, and the Agreement which is greater if more people share the same sentiment. Li et al. (2017) transformed the sentiment into discrete labels after applying sentiment dictionaries SentiWordNet 3.0 and developed a concept graph which considered the relationship between entities in Tweets. Previous studies applied categorical sentiments to analyze the text messages toward the stock market which is easy and intuitive to understand, but this method ignores that sentiment is not an absolute category; it may be neutral to positive/negative and other emotions between categories.

While there are some works measured sentiments in statistical-based method, Oliveira et al. (2017) investigated microblogging data to forecast stock market variables such as stock returns. Wang et al. (2016) proposed a novel sentimental feature extraction technique, using TF-IDF to find the most important words which named basic senti-word and assigning sentimental scores based on the number of occurrences of each word and the rise or fall ratio of related historical stocks. They took another perspective to define the sentiment of a word without predefine the sentiment of each word but using historical stock market fluctuations to push back and forth the sentiment of the word. Most importantly, they considered the importance of each word in the contents. This concept is similar to the attention mechanism we proposed. In our model, the model will focus on the correspondingly important part of content using the vector features of keyword and title.

However, most sentiment analyses concerning the stock market have mainly concentrated on stock trends, but this makes predictive ability limited because stock trend prediction does not consider trading immediacy. In sentiment analysis, dimensional sentiment such as VA feature space provides a deeper interpretation of sentiment. Unlike categorical sentiment analysis, dimensional sentiment analysis aims to transform emotional states into continuous numerical values on multiple dimensions (Russell 1980). By using this approach, researchers can detect fine-grained differences between the same categorical sentiment in texts. For example, the authors (Hasan et al. 2019) were proposed a classification model to classify the Tweets in multi-dimension emotions of valence–arousal automatically in real time. The authors (Salehan and Kim 2020) studied the relationship between valence–arousal sentiments and influence of numerous Tweets. Their result shows that negative sentiment with high arousal notably expands the information spread in social media. Conversely, the tweets with low-arousal negative sentiment relatively retweet less frequencies than high arousal. In addition to the mentioned research areas, the VA model has been used in stock prediction. Dong et al. (2015) examined the relationship between social moods extracted from Sina Weibo and the Shanghai Composite Index, with the results indicating that negative sentiment with low arousal tends to induce risky decision making and behaviors. Ge et al. (2020) explored that messages on social media affect the stock market trend. According to their experimental result, the high arousal emotions in social media bring greater volatility to the stock market after the market crash. In addition, the VA concept has also been widely used in research related to social network analysis (Max et al. 2020) and consumer behavior (Jaeger et al. 2019). However, they just used the VA mode for sentiment word instead of the sentiment of stock market. For example, the word such happiness belongs to high valence with high arousal. Wu et al. (2019b) proposed a deep learning model to predict VA of sentiment on stock news for stock market. They have used an attention mechanism to estimate relationship between summary and keyword.

According to previous studies, sentiment analysis has become to solve multiple-dimensional sentiment tasks in which the valence–arousal is growing up in the past few years. In the stock market analysis field, there is very few research about the multiple-dimensional sentiment analysis using the valence–arousal model, but this concept and definition are very useful to help investors capture what's situation of stock which they need to know. Therefore, this study also establishes the relationships on two-dimensional sentiment including trend direction and trading actions according to the text of summary, title and keyword. The different stock messages influence the investor's emotions, which are represented in two dimensions, have remained unanswered because applying dimensional sentiment analysis in stock prediction has remained rare. Thus, to better comprehend the influence of stock messages, applying deep learning techniques to solve the VA sentiment prediction is the focus of this research.

2.3 Machine learning

Machine learning has been widely adopted and continues to gradually increase. In the medical field, Rohini et al. (2020) used the multiple linear regression, logistic regression and support vector machine to predict and classify the features of Alzheimer's disease or normal cognitive decline in older adults. The results show that the use of simple machine learning models in disease prediction can increase accuracy. In the industrial field, Beninger et al. (2020) aimed to detect mind wandering and predict the response time on driving pattern based in a fully immersive driving. They compared the effectiveness of support vector machines, random forests and multi-layer perceptrons on the data. These results indicate that the effect of the random forest is better. Besides, Gaurav et al. (2020) researched the classification of email spam on the Internet and compared the three methods for Naive Bayes, decision trees and random forests. The research confirmed that random forests have high precision in this case. However, in the field of stock prediction, many ML models have been confirmed to improve classification accuracy. For example, random forest model can be used to forecast medium- and long-term trends of stock prices (Basak et al. 2019). Support vector regression (SVR) has been used to predict stock prices for firms with large and small capitalization, and the results have demonstrated that SVR improves the prediction performance during periods of low volatility (Henrique et al. 2018). However, the above researches only use past data to make predictions and did not use other stock-related text messages as information into consideration.

To achieve better forecasting performance, text-based data have been conducted to capture the more important information through a lot of processes by NLP techniques. And the deep learning techniques also use to enhance prediction performance by adding more hidden layers. Maqsood et al. (2019) proposed a deep learning method to forecast stock prices. They performed sentiment analysis on tweets related to stocks. The result showed that these text data improved the performance of stock price prediction. The another (Abdi et al. 2019) have been used a deep learning model called RNSA to classify the sentiment of large-scale reviews. As shown in the results, RNSA model is composed by RNN and LSTM which has improved the classification accuracy of the sentiment analysis on reviews. Song et al. (2019) proposed a CNN-LSTM framework to do abstractive text summarization task, and the results show that their model outperforms than other approaches in both semantics and syntactic structure. To analyze the semantics and emotions in the text of stock news, the reader feelings play a crucial role in the prediction results. Therefore, the authors (Liang et al. 2018) studied reader emotions and proposed a model which is a labeled topic model and to solve sparsity problem in short texts. However, studies have focused on feature engineering to obtain effective text features and improved prediction performance of many ML approaches (Ikonomakis et al. 2005). However, in previous ML techniques, the bag-of-words model was mainly used, which entails the text being represented as a collection of words, with the order of the words ignored. Semantic misunderstanding during emotion analysis can easily result. These researches indicated that DL approaches with text-based data using more layers with different NN models such as CNN and RNN have a better understanding of the text which greatly improves the prediction performance.

To better understand unstructured data such as texts, deep learning models are used to perform semantic extraction tasks. By directly extracting features of the texts, model prediction can be directly completed without any feature engineering process. Yang et al. (2016) proposed the HAN model and used it to solve document classification tasks, which included six large-scale text classifications. The HAN model is akin to the process of human reading; first, a human/the HAN focuses on the words of the sentence and then focuses on the sentences of the document. At present, the HAN model is applied in lyric analysis for music classification. The results showed that the HAN model is superior to other non-neural models and simple neural models (Tsaptsinos 2017). A hierarchical structure and attention mechanism has been used to model user dynamics preferences. The results of the experiments proved that this method outperformed traditional methods (Ying et al. 2018). Another study used the HAN model for document classification to extract abstract features (Abreu et al. 2019).

In summary, machine learning and deep learning model have obtained enough prediction performance in many sentiment analysis tasks, but this paper is a two-dimensional prediction problem and it is a difference to single predicting target. According to past researches, there is no related work to solve the trend and trading prediction from the stock message, like valence and arousal in sentiment analysis. But the feature from stock message has provided more important information to enhance prediction performance. Therefore, we use a powerful HAN model to capture sentiment states (trend and trading) of the stock message such news, because HAN has high performance for long length document and stock news have more precise information compared to other sources such as social media.

3 Methodology

The paper aimed to predict SDVA of text related to the stock market, and the flow of proposed model is shown in Fig. 1. First, stock-related online messages such as stock news were collected. Then, the intensity of SDVA was annotated by experts. Second, data pre-processing was performed followed by word segmentation, and third, the proposed HAHTKN model was used to train the model from the training set. The best model was selected according to performance on the validation set. Last, the test set was used to evaluate the performance of the proposed model.

3.1 Stock message collection and annotation

We collected relevant stock market news articles from the Internet. The stock articles were used as information to

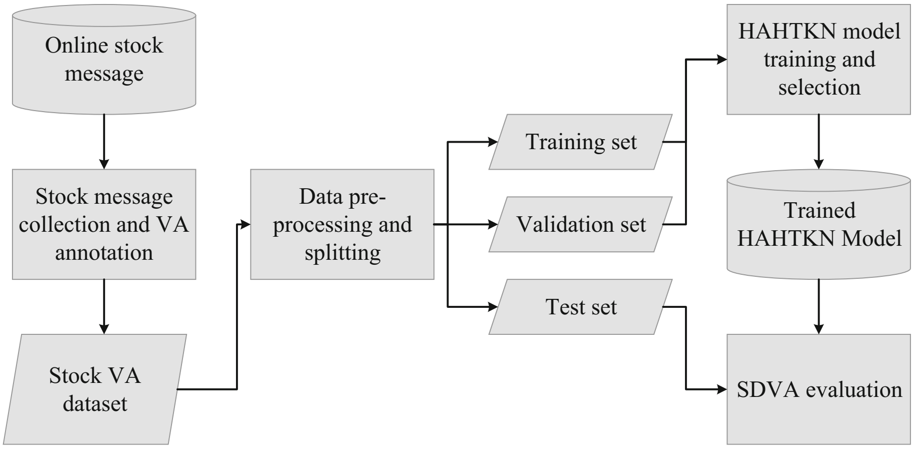
Online stock message
Stock message collection and VA annotation
Stock VA dataset
Data pre-processing and splitting
Training set
Validation set
Test set
HAHTKN model training and selection
Trained HAHTKN Model
SDVA evaluation

Fig. 1 Flowchart of SDVA task using HAHTKN model

form the text feature of each stock news item. Three annotators annotated each stock news item independently. The annotators label each news item with a valence value (trend intensity) according to the stock trend (upward or downward). In labeling arousal, trading intensity is a major concept in which the investor holds a stock or trades it immediately. The valence and arousal values range from 1 to 10. For example, a rating of 1 for valence is a negative sentiment (downtrend), and a rating of 10 for valence is a positive sentiment (uptrend). Regarding arousal, a rating of 1 for arousal indicates low trading intensity (no trading activity), and a rating of 10 indicates high intensity (trade immediately). The average values of valence and arousal for each stock news item from three annotators were used.

3.2 Text pre-processing and splitting

Because the proposed HAHTKN is a hierarchical architecture, we segmented sentences using punctuation such as periods, commas, colons and semicolons. Then, we used the CKIP tagger[1] to segment words of each sentence related to the stock news item. Regarding the text of stock news items, we used the title, summary and keywords as the text-based content. To obtain the weights of the best HAHTKN model, we proposed training set for model training, validation set for model selection and test set for evaluation.

Figure 2 shows the text pre-processing flow. First, the text of title and summary needs to sentences splitting by punctuation, and then these sentences will be segmented by CKIP tagger. Another, the keywords originally existed in the form of words, so do not run the segment process. Second, the words of the training set include title, keyword and summary use to build a vocabulary list which is using to convert word to index. Finally, all words of keyword, title and summary in all sets are converted into index according to the vocabulary list.

3.3 The HAHTKN model

We proposed the HAHTKN model to predict SDVA with respect to trading intensity and trends. In the proposed HAHTKN, multiple DL sub-models were combined as a nonlinear regression model for SDVA prediction. Figure 3 depicts the proposed architecture including title encoder, keyword encoder, title–keyword encoder, word-level encoder, sentence-level encoder and stock DVA prediction layer.

Algorithm 1 summarizes the whole working procedure of our proposed HAHTKN model. For each batch in each epoch, we not only embed title but also use a bidirectional GRU (line 3) and an attention mechanism to obtain v^{T} firstly. Second, we embed keywords and use an attention mechanism in keyword encoder (line 4), and we encode v^{T} and v^{K} into v^{TK} (line 5). Third, we encode words on each sentence using a CNN, a bidirectional GRU and an attention mechanism in word encoder (line 6) to get sentence vector s_i. Fourth, in sentence encoder (line 7), we used a bidirectional GRU and the Luong attention mechanism to get the text vector v. Next, we used a linear model to calculate $\hat{y}$ (line 8), and then we selected the mean square error (MSE) optimizer to minimize the square error between the prediction value $\hat{y}$ and the actual target value y (line 9). Finally, we used Adam optimizer to optimize all training parameters (line 10).

[1] Chinese Knowledge and Information Processing (CKIP) (https://ckip.iis.sinica.edu.tw/).

```
Algorithm 1: HAHTKN model training procedure
Input: title: $[w_1^T,…,w_l^T,…,w_L^T]$, Keyword: $[w_1^K,…,w_m^K,…,w_Q^K]$, Summary: $[w_{i1},…,w_{iu},…,w_{iN}]$
Output: stock valence and arousal prediction $\hat{y}$
For each epoch:
    For each batch:
        $v^T$ ← Title_Encoder($[w_1^T,…, w_l^T,…, w_L^T]$)
        $v^K$ ← Keyword_Encoder($[w_1^K,…, w_m^K,…, w_Q^K]$)
        $v^{TK}$ ← Title_Keyword_Encoder($v^T$ , $v^K$)
        $[s_1,…, s_i,…, s_M]$ ← Word_Encoder($[w_{i1},…, w_{iu},…, w_{iN}]$)
        $v$ ← Sentence_Encoder($[s_1,…, s_i,…, s_M]$, $v^{TK}$)
        $\hat{y}$ ← Stock_DVA_Prediction_Layer($v$)
        loss ← Loss_Function($y$, $\hat{y}$)
        Backward and update model's weights
    End for
End for
```

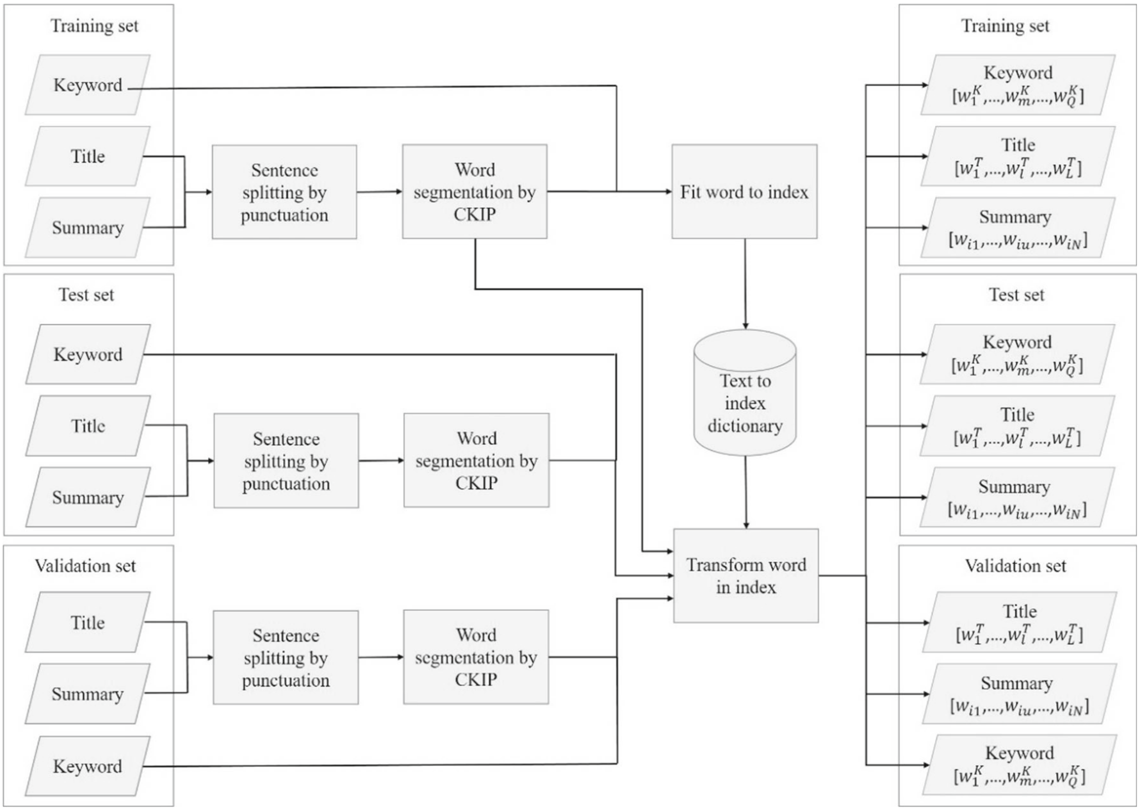
Training set
Keyword
Title
Summary
Sentence splitting by punctuation
Word segmentation by CKIP
Fit word to index
Test set
Keyword
Title
Summary
Sentence splitting by punctuation
Word segmentation by CKIP
Text to index dictionary
Transform word in index
Validation set
Title
Summary
Keyword
Sentence splitting by punctuation
Word segmentation by CKIP
Training set
Keyword $[w_1^K,...,w_m^K,...,w_Q^K]$
Title $[w_1^T,...,w_l^T,...,w_L^T]$
Summary $[w_{i1},...,w_{iu},...,w_{iN}]$
Test set
Keyword $[w_1^K,...,w_m^K,...,w_Q^K]$
Title $[w_1^T,...,w_l^T,...,w_L^T]$
Summary $[w_{i1},...,w_{iu},...,w_{iN}]$
Validation set
Title $[w_1^T,...,w_l^T,...,w_L^T]$
Summary $[w_{i1},...,w_{iu},...,w_{iN}]$
Keyword $[w_1^K,...,w_m^K,...,w_Q^K]$

Fig. 2 The flowchart of text pre-processing

The proposed HAHTKN model comprised six sub-models; the detailed processing is as follows.

3.3.1 Title encoder

Given a title with segmented words $w_l^{\mathrm{T}}, l \in [1, L]$, we obtained each word embedding x_l^{T} from the embedding matrix W_{T}. In this case, we used a bidirectional gated recurrent unit (GRU) to obtain hidden states of word

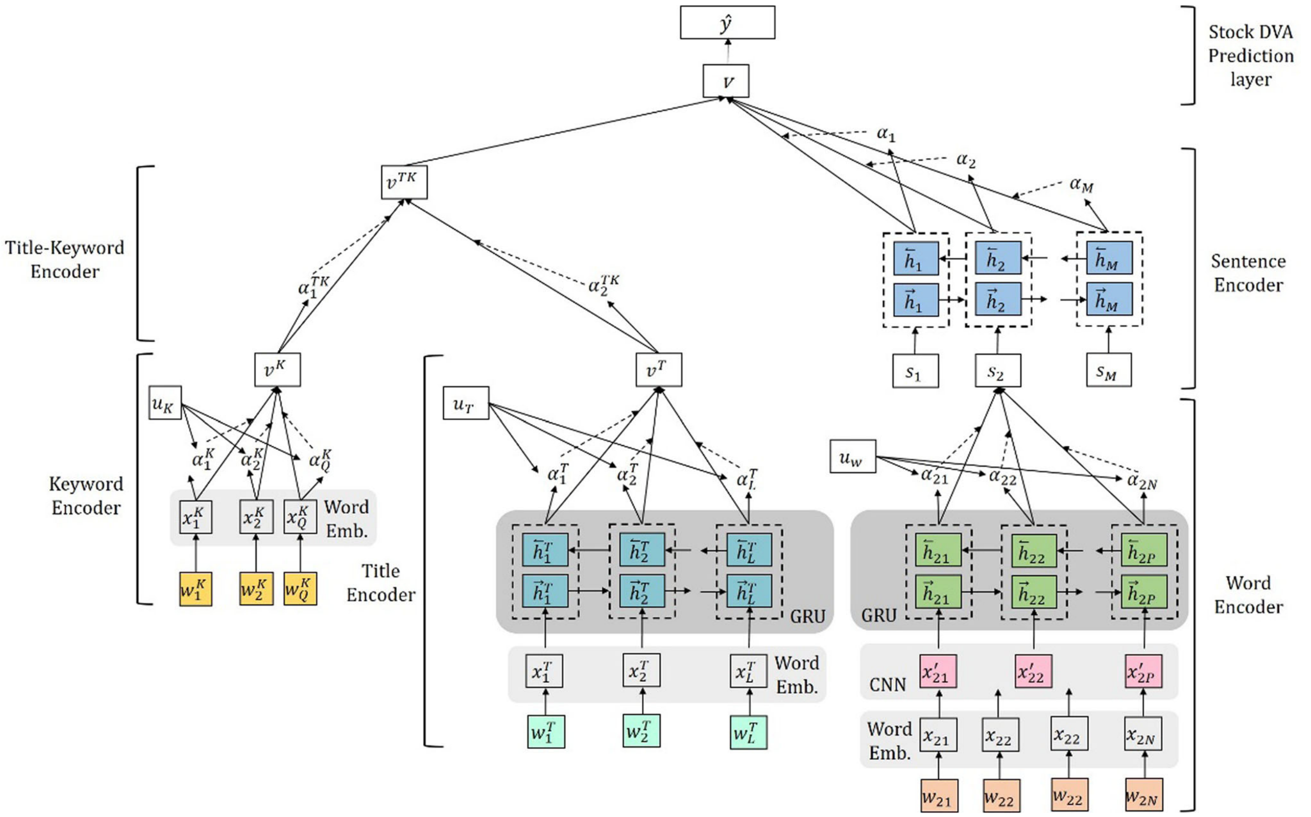
$\hat{y}$
v
Stock DVA Prediction layer
v^{TK}
Title-Keyword Encoder
α_1^{TK}
α_2^{TK}
α_1
α_2
α_M
Sentence Encoder
$\overleftarrow{h}_1$ $\overleftarrow{h}_2$ $\overleftarrow{h}_M$
$\overrightarrow{h}_1$ $\overrightarrow{h}_2$ $\overrightarrow{h}_M$
s_1 s_2 s_M
v^K
v^T
u_K
u_T
u_w
Keyword Encoder
α_1^K α_2^K α_Q^K
x_1^K x_2^K x_Q^K
Word Emb.
w_1^K w_2^K w_Q^K
Title Encoder
α_1^T α_2^T α_L^T
$\overleftarrow{h}_1^T$ $\overleftarrow{h}_2^T$ $\overleftarrow{h}_L^T$
$\overrightarrow{h}_1^T$ $\overrightarrow{h}_2^T$ $\overrightarrow{h}_L^T$
GRU
x_1^T x_2^T x_L^T
Word Emb.
w_1^T w_2^T w_L^T
α_{21} α_{22} α_{2N}
$\overleftarrow{h}_{21}$ $\overleftarrow{h}_{22}$ $\overleftarrow{h}_{2P}$
$\overrightarrow{h}_{21}$ $\overrightarrow{h}_{22}$ $\overrightarrow{h}_{2P}$
GRU
CNN
x'_{21} x'_{22} x'_{2P}
Word Emb.
x_{21} x_{22} x_{22} x_{2N}
w_{21} w_{22} w_{22} w_{2N}
Word Encoder

Fig. 3 Architecture of the proposed model

embeddings from both directions, that is,$h_l^{\mathrm{T}} = \left[\overrightarrow{h}_l^{\mathrm{T}}, \overleftarrow{h}_l^{\mathrm{T}}\right]$. A bidirectional GRU contains the forward $\overrightarrow{GRU}$, which reads the title from x_1^{T} to x_L^{T}, and a backward $\overleftarrow{\mathrm{GRU}}$, which reads the title from x_L^{T} to x_1^{T}:

$$x_l^{\mathrm{T}} = W_{\mathrm{T}}\varphi\left(w_l^{\mathrm{T}}\right), \quad l \in [1, L] \tag{1}$$

$$\overrightarrow{h}_l^{\mathrm{T}} = \overrightarrow{GRU}\left(x_l^{\mathrm{T}}\right), \quad l \in [1, L] \tag{2}$$

$$\overleftarrow{h}_l^{\mathrm{T}} = \overleftarrow{GRU}\left(x_l^{\mathrm{T}}\right), \quad l \in [L, 1] \tag{3}$$

where φ is a one-hot operation, l is the number of words of each title, and the W_{T} denotes the trainable parameters in the HAHTKN. Standard GRU models were used for $\overrightarrow{GRU}$ and $\overleftarrow{\mathrm{GRU}}$. However, not all words of the title contribute equally to the meaning of the title. Therefore, we used a simple attention mechanism with a title-level context vector u_l^{T} to estimate the importance value of each word and combined those informative words into a title vector v^{T} that summarizes the information of words in the title of the stock article. A title-level representation v^{T} is expressed as follows:

$$u_l^{\mathrm{T}} = \tanh\left(W_{T'} h_l^{\mathrm{T}} + b_{\mathrm{T}}\right) \tag{4}$$

$$\alpha_l^{\mathrm{T}} = \frac{\exp\left(\left(u_l^{\mathrm{T}}\right)^{\mathrm{T}} u_{\mathrm{T}}\right)}{\sum_v \exp\left(\left(u_v^{\mathrm{T}}\right)^{\mathrm{T}} u_{\mathrm{T}}\right)} \tag{5}$$

$$v^{\mathrm{T}} = \sum_l \alpha_l^{\mathrm{T}} u_l^{\mathrm{T}} \tag{6}$$

where $W_{T'}$, b_{T}, and u_{T} are the trainable parameters in the HAHTKN. The model uses u_l^{T} to measure the importance of a word and acquire a normalized importance weight α_l^{T}, and l and v both represent the sequence of a word in each title. The model computes the title vector v^{T} as a weighted sum of the title-level representations based on the weights.

3.3.2 Keyword encoder

Given a stock message with the keyword $w_m^K, m \in [1, Q]$, Q denotes the maximum number of keywords in a stock news item. We obtained each keyword embedding from the embedding matrix W_K. To measure the varying importance of a keyword, we used the simple attention mechanism to evaluate the importance of keywords in each stock news item and aggregated the representation of those informative words into a keyword vector as follows:

$$x_m^K = W_K \varphi\left(w_m^K\right), m \in [1, Q] \tag{7}$$

$$u_m^K = \tanh\left(W_{K'} x_m^K + b_K\right) \tag{8}$$

$$\alpha_m^K = \frac{\exp\left(\left(u_m^K\right)^\top u_K\right)}{\sum_r \exp\left(\left(u_r^K\right)^\top u_K\right)} \tag{9}$$

$$v^K = \sum_m \alpha_m^K x_m^K \tag{10}$$

where W_K, $W_{K'}$, b_K, and u_K are trainable parameters in the HAHTKN. We used u_m^K to measure the importance of keywords and acquired a normalized importance weight α_m^K, and m and r both represent the sequence of a keyword in each stock news item. The keyword representation is obtained by summing the weighted keyword-level vector.

3.3.3 Title–keyword attention

Given title vector v^T and keyword vector v^K, v^T and v^K are concatenated using a simple attention mechanism to obtain a title–keyword context vector u^{TK}. Then, the title–keyword context vector u^{TK} is used to estimate the importance value of different titles and keywords. These informative words are combined into a title–keyword-based word vector v^{TK} that summarizes all the information of titles and keywords in a stock message as follows:

$$u^{TK} = \tanh\left(w^{TK}\left[v^\mathrm{T}, v^K\right]\right) \tag{11}$$

$$\alpha_o^{TK} = \frac{\exp\left(\left(u_o^{TK}\right)^\top u_{TK}\right)}{\sum_p \exp\left(\left(u_p^{TK}\right)^\top u_{TK}\right)} \tag{12}$$

$$v^{TK} = \sum_o \alpha_o^K u_o^{TK} \tag{13}$$

where w^{TK} and u_{TK} are trainable parameters in the HAHTKN.

3.3.4 Word-level encoder

Given the ith sentence with words $w_{iu}, u \in [1, N]$, N is the maximum number of words in a sentence. First, we obtained each word embedding from the embedding matrix W_E. Then, we apply the standard convolution neural networks (CNN) as filters to produce a feature map from word embeddings. Moreover, this paper designs multiple filters with max-pooling operation to form a CNN feature extractor and features of each filter will be concatenated to hidden features x_{ij}. After CNN extraction, we also used a bidirectional GRU to obtain hidden states of word embeddings from both directions, that is,$h_{ij} = \left[\vec{h}_{ij}, \overleftarrow{h}_{ij}\right]$. A bidirectional GRU contains the forward $\overrightarrow{GRU}$, which reads the sentence s_i from x_{i1} to x_{iP}, and a backward $\overleftarrow{GRU}$, which reads from x_{iP} to x_{i1}:

$$x_{iu} = W_\mathrm{E}\varphi(w_{iu}), \quad u \in [1, N] \tag{14}$$

$$\left[x'_{i1}, x'_{ij}, x'_{iP}\right] = CNN([x_{i1}, x_{iu}, x_{iN},]), \tag{15}$$

$$\vec{h}_{ij} = \overrightarrow{GRU}\left(x'_{ij}\right), \quad j \in [1, P] \tag{16}$$

$$\overleftarrow{h}_{ij} = \overleftarrow{GRU}\left(x'_{ij}\right), \quad j \in [P, 1] \tag{17}$$

where W_E denotes the trainable parameters in the HAHTKN. To weigh words that are relevant, we used a simple attention mechanism with a word-level context vector u_W to estimate the importance value of each word and combined those informative words into a sentence vector s_i that summarizes all the information of words in a sentence. A sentence-level representation s_i is expressed as follows:

$$u_{ij} = \tanh\left(W_\mathrm{S} h_{ij} + b_\mathrm{S}\right) \tag{18}$$

$$\alpha_{ij} = \frac{\exp\left(u_{ij}^\top u_\mathrm{W}\right)}{\sum_q \exp\left(u_{iq}^\top u_\mathrm{W}\right)} \tag{19}$$

$$s_i = \sum_j \alpha_{ij} h_{ij} \tag{20}$$

where W_S, b_S, and u_W are trainable parameters in the HAHTKN. The model uses u_W to measure the importance of a word and acquire a normalized importance weight α_{ij}, and j and q both represent the sequence of a word in each sentence. The weighted word-level hidden states are summed and represented by the sentence vector s_i.

3.3.5 Sentence-level encoder

Given the sentence vectors s_i, sentence representations by forward and backward bidirectional GRUs to encode the sentences are as follows:

$$\vec{h}_i = \overrightarrow{GRU}(s_i), \quad i \in [1, M] \tag{21}$$

$$\overleftarrow{h}_i = \overleftarrow{GRU}(s_i), \quad i \in [M, 1] \tag{22}$$

here $\vec{h}_i$ and $\overleftarrow{h}_i$ are concatenated into a sentence hidden state, that is, $h_i = \left[\vec{h}_i, \overleftarrow{h}_i\right]$. Then, the Luong attention mechanism is used to estimate the importance of each sentence hidden state h_i based on title–keyword-based vector v^{TK} and to aggregate the representation of those informative sentences into a text vector v:

$$u_i = \begin{cases} h_i^{\mathrm{T}} v^{TK} & \text{dot} \\ h_i^{\mathrm{T}} W_{\mathrm{b}} v^{TK} & \text{general} \\ v_{\mathrm{b}}^{\mathrm{T}} \tanh(W_{\mathrm{b}}[h_i, v^{TK}]) & \text{concat} \end{cases} \tag{23}$$

$$\alpha_i = \frac{\exp(u_i)}{\sum_j \exp(u_j)} \tag{24}$$

$$v = \sum_i \alpha_i h_i \tag{25}$$

where W_{b} and $v_{\mathrm{b}}^{\mathrm{T}}$ denote the trainable parameters in the HAHTKN. The model uses u_i to measure the importance of a sentence and acquire a normalized importance weight α_i. Then, the weighted sentence-level hidden states are summed. In addition, the *concat* operation is evaluated by concatenating sentence and keywords vectors. The *general* operation directly uses the keywords vector to estimate the attention weight, and in the *dot* operation, no model weight is attached to estimate the attention weight.

3.3.6 Stock DVA prediction layer

The text vector v is a high-level representation of the message, and it can be used to predict stock valence and arousal $\hat{y} \in \mathbb{R}^2$:

$$\hat{y} = W_{VA} v + b_{VA} \tag{26}$$

where W_{VA} and b_{VA} are trainable parameters in the HAHTKN.

3.4 Model training and optimization

The mean square error (MSE) is used to calculate the loss between predicting SDVA and target SDVA. The training loss function is defined as follows:

$$L(\theta) = \frac{1}{L} \sum_l (\hat{y}_l - y_l)^2 \tag{27}$$

where y_l is the target SDVA values of the lth stock message. The parameters of θ are jointly trained, converged and optimized using the Adam optimizer.

4 Experiments

We collected 3588 stock market messages from January to April 2019 from the Anue website (https://www.cnyes.com/). Our data contained the title, summary and keywords of a stock message. Valence, as trend intensity, and arousal, as trading intensity, were independently annotated by three experts. The trend and trading values ranged between 1 and 9. The deviation values were removed, and the average values of the valence and arousal of three annotated values were obtained to get the final SDVA values. In addition, this data set was collected for the SDVA sentiment analysis. It is the first study in the relevant stock sentiment analyses.

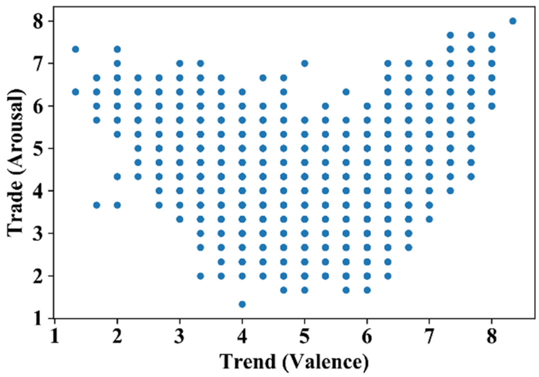

Fig. 4 Valence and arousal score distributions

Figure 4 illustrates the valence and arousal distributions of the collected data. Both low valence and high valence typically indicate high arousal, which reveals that a stock price is expected to increase or decrease sharply in the future. Thus, an investor should trade immediately. Furthermore, middle arousal and low arousal typically have middle valence because the message does not reveal a drastic increase or decrease in the stock price. An investor is expected to hold their stocks.

Table 1 presents the statistics of the collected data set. A total of 3588 documents along with the annotated SDVA was divided into data for training (73.2% of the data), validation (13.9%) and test (12.9%). The average value of valence was approximately 5.73, and the average value of arousal was approximately 4.06. Thus, the average value of arousal in our data set had a downward bias. The annotators may have been more conservative about transactions. Most of the stock news items did not motivate immediate trading.

4.1 Evaluation metrics

We compared the SDVA values labeled by each annotator against their corresponding means across the three annotators to calculate the error rates using the following metrics:

- Mean absolute error (MAE):

$$\mathrm{MAE} = \frac{1}{L} \sum_{l=1}^{L} |A_l - \bar{A_l}| \tag{28}$$

- Root-mean-square error (RMSE):

Table 1 Data set statistics

Item	Dataset		
	Training	Validation	Test
Number of document	2628	498	462
Average number of sentence	5.28	5.20	5.20
Average number of word/per sentence	9.88	9.89	10.10
Average number of word/per title	15.98	16.57	16.06
Average number of keyword	5.21	5.32	5.45
Average valence	5.79	5.63	5.75
Average arousal	3.82	4.42	3.93

Table 2 Annotation evaluation on annotators

	MAE		RMSE	
	Valence	Arousal	Valence	Arousal
Annotator A_1	0.457	0.957	0.633	1.211
Annotator A_2	0.610	0.943	0.838	1.198
Annotator A_3	0.505	1.221	0.718	1.479
(Mean, SD)	(0.524, 0.078)	(1.040, 0.157)	(0.730, 0.103)	(1.296, 0.159)

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{l=1}^{L}(A_l - \bar{A}_l)^2}{L}} \tag{29}$$

where A_l denotes the valence or arousal value of the lth message rated by an annotator, $\bar{A}_l$ denotes the mean valence or arousal of the lth message calculated over the three annotators, and L is the total number of messages in our collected data set.

For the model evaluation, the two metrics of MSE and medium absolute error (MDAE) were used to evaluate each model's performance with respect to stock valence and arousal. The MDAE formula is defined as follows:

$$\mathrm{MDAE} = \mathrm{median}(|\hat{y}_1 - y_1|, \ldots, |\hat{y}_L - y_L|) \tag{30}$$

4.2 Human annotation results of valence and arousal with respect to stock news

Table 2 depicts the error rates of different annotators in rating the valence and arousal of news in the data set. In all metrics, the mean error rates of arousal ratings were higher than those of valence ratings. In the valence dimension, the MAE was approximately 0.457–0.610, and the RMSE was approximately 0.633–0.838. In the arousal dimension, the MAE was approximately 0.943–1.221, and the RMSE was approximately 1.198–1.479, reflecting the maximum gap of all error rates. The annotators' attitudes toward valence were more consistent than arousal. We concluded that the applied two dimensions of SDVA in the stock market were consistent with other VA models used in sentiment analysis.

4.3 Prediction models for SDVA task

In the experiments, the HAN model was used as a reference because the model has been proven to be effective in understanding the meaning of text. We proposed eight neural network models which base on HAN and two machine learning models to compare model performance; these are detailed as follows:

1. Baseline model:
   - **RF:** a Random Forest Regression model, which uses the TF-IDF approach to transform all words of the document into a vector.
   - **SVR**: a Support Vector Regression model with a linear kernel, which uses the TF-IDF approach to transform all words of document into a vector.
   - **HAN**: the baseline model (Yang et al., 2016).
2. Advanced model:
   - **HAHN**: a CNN is added to the HAN model. This CNN operates on the first layer of the NN.
   - **HAKN**: the proposed model, which uses Luong attention mechanism in the sentence layer with keyword-based attention.
   - **HAHKN**: a CNN is added to the HAKN model, which operates on the first layer of the NN.

Table 3 Prediction performances with respect to difference estimation

Models	Attention estimation	Validation loss	Test loss	Trend		Trading	
				MSE	MDAE	MSE	MDAE
HAKN	Concat	0.902	0.890	0.842	0.527	**0.937**	0.668
	Dot	**0.860**	**0.864**	**0.715**	**0.481**	1.013	**0.656**
	General	0.861	0.881	0.738	0.508	1.024	0.717
HAHKN	Concat	**0.876**	**0.831**	**0.772**	0.458	**0.889**	**0.613**
	Dot	0.974	0.950	0.800	**0.455**	1.100	0.661
	General	1.035	1.005	0.934	0.513	1.076	0.679
HATN	Concat	**0.840**	**0.802**	**0.660**	**0.426**	**0.943**	0.696
	Dot	0.941	0.987	0.909	0.516	1.065	**0.679**
	General	0.903	0.853	0.746	0.447	0.959	0.715
HAHTN	Concat	**0.864**	**0.832**	**0.756**	**0.452**	**0.907**	**0.654**
	Dot	0.936	0.914	0.858	0.502	0.970	0.664
	General	0.962	1.078	1.131	0.755	1.024	0.667
HATKN	Concat	0.858	**0.841**	**0.748**	0.484	0.934	0.634
	Dot	**0.830**	0.852	0.835	**0.445**	**0.869**	**0.628**
	General	1.056	0.975	0.972	0.460	0.977	0.691
HAHTKN	Concat	**0.830**	**0.772**	**0.718**	0.452	**0.826**	0.592
	Dot	0.961	0.852	0.799	0.458	0.904	0.667
	General	0.904	0.899	0.902	**0.426**	0.895	**0.584**

Bold value indicate best result between three attention estimations for each model

- **HATN**: the proposed model, which uses Luong attention mechanism in the sentence layer with title-based attention.
- **HAHTN**: a CNN is added to the HATN model, which operates on the first layer of the NN.
- **HATKN**: the proposed model, which uses Luong attention mechanism in the sentence layer with title-based attention and keyword-based attention.
- **HAHTKN**: a CNN is added to the HATKN model, which operates on the first layer of the NN.

4.4 Hyper-parameters setup

Our proposed deep learning model has several hyper-parameters: hidden size for all sub-model is 100, 300 and 500; learning rates are 0.001 and 0.01; filter sizes of a multiple channel CNN are [2,3,4]; and another cell as LSTM cell is used to evaluation. The maximum number of word for title encoder and word-level encoder is 26 and 44, respectively. The number of sentence for sentence encoder is 12, and all words for keyword encoder have used. We set several parameters for RF regression including criterion that is MSE, numbers of estimators are [25,50,100], and numbers of min sample splitting are [2,10]. For SVR model, kernels are [linear, poly, rbf, sigmoid]; cost and gamma are 1; degrees are [3,5]. Noted that gamma is set in [poly, rbf, sigmoid] kernel, and degree is only set in poly kernel.

4.5 Performance comparisons regarding different estimations of Luong attention

In this experiment, we used Luong attention to focus on the sentences in the stock market news to predict SDVA scores, which included calculation methods such as *concat*, *dot* and *general*. Table 3 presents a comparison of the effectiveness of each model in each prediction method. The minimum validation among different prediction methods is presented for comparison among the models. Only the HAKN achieved the best results in dot. Its test loss was 0.864. Other models were better in the *concat* operation. HAHTKN was the best model, obtaining 0.772 test losses. Furthermore, in trend, 0.718 MSE and 0.452 MDAE on test data were obtained using the model. In trading, 0.826 MSE and 0.592 MDAE on test data were obtained. The results show that *concat* can maintain more information. The original information can provide a better estimate.

4.6 Performance comparisons for different hidden sizes

We tested the hidden sizes of 100, 300 and 500. When the hidden size was equal to 500, HAN, HAKN, HAHTN and HATKN had a better effect with test losses of 0.834, 0.864, 0.832 and 0.841, respectively. HAHN, HAHKN, HATN and HAHTKN had a better effect when the hidden size was

Table 4 Prediction performances with respect to different hidden sizes

Models	Hidden size	Validation loss	Test loss	Trend		Trading	
				MSE	MDAE	MSE	MDAE
HAN	100	0.872	1.004	0.835	0.452	1.172	0.693
	300	0.880	0.898	0.809	0.524	**0.986**	0.677
	500	**0.848**	**0.834**	**0.681**	**0.445**	0.987	**0.676**
HAHN	100	**0.902**	0.923	0.902	**0.441**	0.944	0.673
	300	0.924	**0.822**	**0.777**	0.458	**0.866**	**0.627**
	500	0.962	0.884	0.818	0.467	0.949	0.666
HAKN	100	**0.857**	0.878	0.788	**0.471**	**0.967**	0.661
	300	0.861	0.881	0.738	0.508	1.024	0.717
	500	0.860	**0.864**	**0.715**	0.481	1.013	**0.656**
HAHKN	100	1.112	1.015	0.853	**0.390**	1.176	0.735
	300	**0.876**	**0.831**	**0.772**	0.458	**0.889**	**0.613**
	500	0.897	0.900	0.852	0.427	0.947	0.645
HATN	100	0.866	0.944	0.836	0.488	1.051	0.737
	300	**0.840**	**0.802**	**0.660**	**0.426**	0.943	0.696
	500	0.862	0.833	0.731	0.446	**0.934**	**0.660**
HAHTN	100	0.918	0.916	0.855	0.475	0.977	0.670
	300	0.912	0.878	0.832	0.514	0.924	0.661
	500	**0.864**	**0.832**	**0.756**	**0.452**	**0.907**	**0.654**
HATKN	100	0.984	0.941	0.888	**0.431**	0.994	0.665
	300	0.953	0.861	0.778	0.525	0.944	0.668
	500	**0.858**	**0.841**	**0.748**	0.484	**0.934**	**0.634**
HAHTKN	100	1.042	0.956	0.838	**0.441**	1.074	0.697
	300	**0.830**	**0.772**	**0.718**	0.452	**0.826**	**0.592**
	500	0.960	0.837	0.769	0.474	0.904	0.653

Bold value indicate best result between three hidden sizes for each model

Table 5 Prediction performances with respect to different RNN cells

Models	RNN cell	Validation loss	Test loss	Trend		Trading	
				MSE	MDAE	MSE	MDAE
HAN	GRU	0.848	**0.834**	**0.681**	0.445	**0.987**	**0.676**
	LSTM	**0.826**	0.887	0.765	**0.437**	1.009	0.729
HAHN	GRU	**0.924**	**0.822**	**0.777**	**0.458**	**0.866**	**0.627**
	LSTM	0.962	0.884	0.818	0.467	0.949	0.666
HAKN	GRU	**0.860**	**0.864**	**0.715**	**0.481**	1.013	0.656
	LSTM	1.000	0.940	0.857	0.513	**1.023**	**0.714**
HAHKN	GRU	**0.876**	**0.831**	0.772	0.458	**0.889**	**0.613**
	LSTM	0.894	0.882	**0.748**	**0.439**	1.015	0.711
HATN	GRU	**0.840**	**0.802**	**0.660**	**0.426**	0.943	0.696
	LSTM	0.862	0.833	0.731	0.446	**0.934**	**0.660**
HAHTN	GRU	0.912	0.878	0.832	0.514	0.924	0.661
	LSTM	**0.864**	**0.832**	**0.756**	**0.452**	**0.907**	**0.654**
HATKN	GRU	0.953	0.861	0.778	0.525	0.944	0.668
	LSTM	**0.858**	**0.841**	**0.748**	**0.484**	**0.934**	**0.634**
HAHTKN	GRU	**0.830**	**0.772**	**0.718**	**0.452**	**0.826**	**0.592**
	LSTM	0.960	0.837	0.769	0.474	0.904	0.653

Bold value indicate best result between two RNN cells for each model

Table 6 Prediction performances with respect to difference learning rates

Models	Learning rating	Validation loss	Test loss	Trend		Trading	
				MSE	MDAE	MSE	MDAE
HAN	0.001	**0.848**	**0.834**	**0.681**	**0.445**	0.987	**0.676**
	0.01	0.880	0.898	0.809	0.524	**0.986**	0.677
HAHN	0.001	0.924	**0.822**	**0.777**	0.458	**0.866**	**0.627**
	0.01	**0.902**	0.923	0.902	**0.441**	0.944	0.673
HAKN	0.001	**0.860**	**0.864**	**0.715**	**0.481**	**1.013**	**0.656**
	0.01	0.861	0.881	0.738	0.508	1.024	0.717
HAHKN	0.001	**0.876**	**0.831**	**0.772**	**0.458**	**0.889**	**0.613**
	0.01	1.002	0.979	0.944	0.500	1.013	0.686
HATN	0.001	**0.840**	**0.802**	**0.660**	**0.426**	**0.943**	**0.696**
	0.01	0.964	0.930	0.900	0.565	0.960	0.721
HAHTN	0.001	**0.864**	**0.832**	**0.756**	**0.452**	**0.907**	**0.654**
	0.01	1.060	0.986	0.903	0.582	1.068	0.756
HATKN	0.001	**0.858**	**0.841**	**0.748**	**0.484**	**0.934**	**0.634**
	0.01	0.953	0.861	0.778	0.525	0.944	0.668
HAHTKN	0.001	**0.830**	**0.772**	**0.718**	**0.452**	**0.826**	**0.592**
	0.01	1.100	1.003	0.965	0.601	1.040	0.659

Bold value indicate best result between two learning rates for each model

Table 7 Comparing the prediction performances of joining CNN filter

CNN	Model	Test loss	Trend		Trading	
			MSE	MDAE	MSE	MDAE
Without	HAN	0.834	0.681	0.445	0.987	0.676
	HAKN	0.864	0.715	0.481	1.013	0.656
	HATN	**0.802**	**0.660**	**0.426**	0.943	0.696
	HATKN	0.841	0.748	0.484	**0.934**	**0.634**
With	HAHN	0.822	0.777	0.458	0.866	0.627
	HAHKN	0.831	0.772	0.458	0.889	0.613
	HAHTN	0.832	0.756	**0.452**	0.907	0.654
	HAHTKN	**0.772**	**0.718**	**0.452**	**0.826**	**0.592**

Bold value indicate best result between with and without CNN for each model

300. Test losses were 0.822, 0.831, 0.802 and 0.772, respectively. All results are shown in Table 4.

4.7 Performance comparisons for different recurrent neural network (RNN) cells

Table 5 presents results of the performances of different models on various cells. Two cells are used in the all experimental models, namely GRU and long short-term memory (LSTM). HAHTN and HATKN achieved the smallest test loss in the LSTM which were 0.832 and 0.841, respectively. However, other models except HAHTN and HATKN are exhibited superior results in the GRU. These two models which are better in LSTM cell that were exhibited 0.033 reduced average test loss compared with themselves using GRU cell. By contrast, the models that were superior in GRU exhibited a reduced average test loss of 0.056 compared with themselves using LSTM cell. Thus, the effect of GRU was better than that of LSTM.

4.8 Performance comparisons for different learning rates

Table 6 presents the results of different learning rates. Overall, when the 0.001 learning rate is used to model training, 100% and 87.5% of models had superior MSE and MDAE, respectively, in trend intensity (valence). In trading intensity (arousal), 87.5% and 100% of models had better MSE and MDAE, respectively. However, some models using the 0.01 learning rate have outperformed 0.001 learning rate such as the HAN has better MSE of trading intensity, HAHN has better MDAE of trend intensity.

4.9 Performance comparisons for CNN

The overall SDVA prediction performances of the models are presented in Table 7. Without the CNN, the HATN exhibited the best performance. The test loss was 0.802. Regarding trend, MSE of 0.660 and MDAE of 0.426 on test data were obtained. In trading, 0.943 MSE and 0.696 MDAE on test data were obtained. If adding the CNN, the HAHTKN exhibited the best performance, with test loss of 0.772, MSE of 0.718 and MDAE of 0.452 in trend and

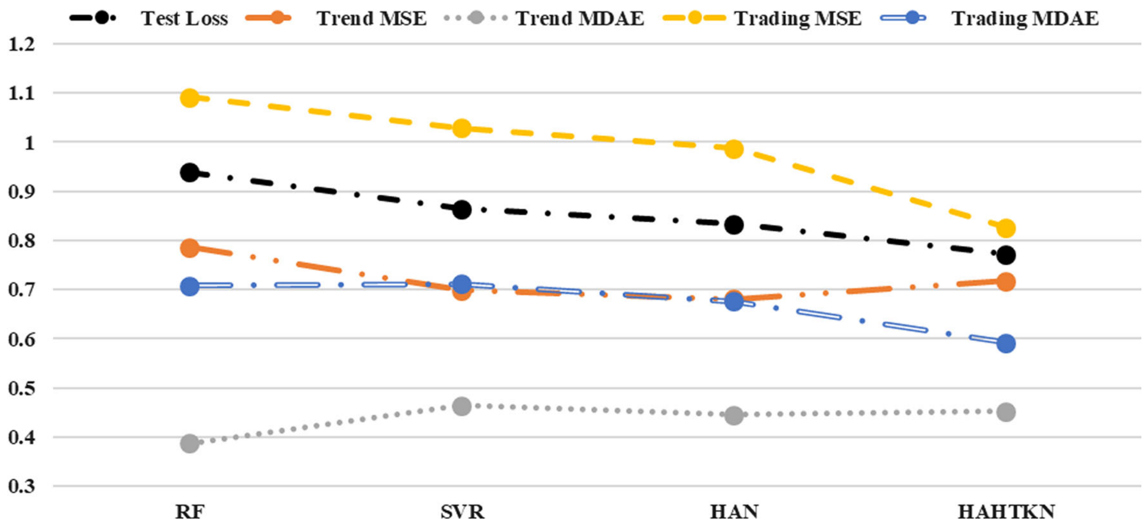

Fig. 5 Overall prediction performances of four models on test set

MSE of 0.826 and MDAE of 0.592 in trading. In addition, it can be seen that only HAHTN model exhibited worse prediction results after adding CNN extraction features. The cause of this situation is likely to be exaggerated title or the inconsistency of the title and contents. However, misleading by biased title is not a mistake only made by machine but also humans. Therefore, if we use title and keywords information, we can measure the weights in title and keywords to achieve better predictions. The result of this experiment concurred with this observation. The HAHTKN achieved this performance. With the information of title and keywords, the neural networks can pay more attention to important sentences in the content of summary which means the weight of these sentences is higher than others. Moreover, these models use the CNN filter, three experiments using CNN have best performance compared to those without CNN filter.

4.10 Overall performance comparison and statistical significance tests

The overall SDVA prediction performances of these models are presented in Fig. 5. We compared the three baseline models with our proposed model HAHTKN. The two machine learning models are RF and SVR. According to the experimental result, the best hyper-parameter setting for RF regression is that the number of estimators is 100 and min samples splitting is 2. Best hyper-parameters setting for SVR that kernel is linear. In trend prediction result, RF had MSE of 0.786 and MDAE of 0.387 and SVR had MSE of 0.699 and MDAE of 0.464. In the trading prediction result, RF had MSE of 1.092 and MDAE of 0.708, and SVR had MSE of 1.029 and MDAE of 0.712. HAHTKN model's performance was better than two machine learning models. Its test loss (overall MSE) was 0.834. However, HAHTKN was the best model in our experiment.

We also experiment with two statistical tests to prove that the losses of HAHTKN on the test set are significant improvements among the other three baseline models. A lot of researches have proved that their models can effectively predict trends of the stock market, but they lack discussion on the stock market trading. Therefore, we chose to use trading intensity for statistical tests. There are two statistical significance tests, the *F* test and *t* test to carry out the loss of the proposed HAHTKN and the other three models. The *F* test is to calculate whether there is a significant difference in the variance of the two populations, and the *t* test is to verify whether the difference between the mean of two populations is significant. We set *F* test and *t* test to be a one-tailed test and two-tailed test, respectively. The alpha of both tests was 0.05. The statistical results are shown in Table 8. In trading intensity, the difference between HAHTKN and RF is significantly different, because the *p* value is under 0.05 in *F* test. All *p* values in the *t* test are under 0.05. It represents that the loss of HAHTKN is significantly smaller than others.

4.11 Discussion

To illustrate this attention weight on these predication models, we will present a practical stock news as an example to show the differences. An example of a validation data set is presented in Table 9. According to the table, Sinosteel made a good profit last year, and the cash yield was not bad. Investors may believe that the current stock market trend is increasing. However, because the content does not mention the company's current and future

Table 8 Results of significant test on compared models in trading intensity

Compared models	*F* test (α = 0.05, one-tailed test)		*t* test (α = 0.05, two-tailed test)	
	F-score	*p* value	*t*-score	*p* value
HAHTKN versus RF	0.827	0.021*	− 3.361	0.001*
HAHTKN versus SVR	0.875	0.075	− 2.725	0.007*
HAHTKN versus HAN	0.866	0.061	− 1.977	0.048*

*Represents that the result of *p* value is under α = 0.05

Table 9 A SDVA example of stock news

Item	Content
Title	Sinosteel plans to distribute NT $ 1 in total and will issue NT $15.735 billion in cash. Yield 4%
Keywords	Sinosteel, Allotment, Closing price, Pure benefit
Summary	Sentence 1: Sinosteel (2002-TW) net profit after tax increased by more than 40% last year, Sentence 2: Which is expected to reach a new high of nearly 8 years. Sentence 3: It is announced today (25) that ordinary shares will be allotted a cash dividend of 1 yuan per share, Sentence 4: Which will total approximately NT $15.735 billion the cash dividend. Sentence 5: If calculated at today's closing price of NT $ 24.85, Sentence 6: The cash yield is about 4%.
Trend value	7.0
Trading value	5.0

Table 10 Prediction performances of each model in this example

Target	HAKN	HAHKN	HATN	HAHTN	HATKN	HAHTKN
Trend	6.671 (0.329)	6.011 (0.989)	6.632 (0.368)	6.427 (0.573)	6.752 (**0.248**)	6.682 (0.318)
Trading	5.262 (− 0.262)	3.937 (1.063)	5.466 (− 0.466)	4.171 (0.829)	4.806 (0.194)	4.913 (**0.087**)

Bold value indicate best result between models for each target

() is the error between target and prediction

goals and growth, the trading intensity is medium. The average value of three annotations for its trend and trading is 7.0 and 5.0.

In Table 10, we present a comparison of the errors in trend and trading among six models. The overall error of the HAHTKN was only 0.405 (0.318 + 0.087), which was the smallest of all models. According to result in Table 11, the HAHTKN focused on Sentence 1, which resulted in better prediction of trading intensity. Sentence 1 clearly revealed the profit growth of the company, which was closer to the investor's perspective in trading. By contrast, the HAHTKN did not exhibit the best trend intensity because the HATKN examines each sentence and the weight of Sentence 3 was higher. Sentence 3 explained the amount that will be allocated per share, which is also crucial information for investors to use to judge the stock market trend.

Table 11 Attention weights of each model for each sentence

Sentence no.	HAKN	HAHKN	HATN	HAHTN	HATKN	HAHTKN
1	0.166	0.190	0.173	0.000	0.192	1.000
2	0.166	0.189	0.173	0.000	0.100	0.000
3	0.166	0.185	0.169	0.000	0.214	0.000
4	0.166	0.185	0.168	0.000	0.166	0.000
5	0.166	0.129	0.169	0.000	0.150	0.000
6	0.166	0.122	0.148	1.000	0.178	0.000

In the HATN and HAHKN, which focused on each sentence with different weights. The HAHTN focused on the most relevant title sentence, "the cash yield is about 4%" because it added the CNN to extract the feature vector. The HAHTKN was focused on "Sinosteel (2002-TW) net profit after tax increased by more than 40% last year." The sentence that the HAHTKN followed was more specific than that of the HAHTN; thus, an impression was conveyed that the overall stock market was better than last year. This conclusion was similar to that of humans reading the articles. Specific and obvious growth draws attention and promotes purchase desire. Therefore, the HAHTKN exhibited the best performance in this experiment because it closely reflected human thinking.

In the comparison of Luong attention's calculation method, 66.7% of the models exhibited good effects on *concat*, and the other 33.3% of the models exhibited better effects on the *dot* operation. In the *concat* method, the sentence vector and the estimated vector of keyword and title were calculated together to obtain complete information during prediction. For hidden sizes of 300 and 500, the results were superior for a hidden size of 300 than 500. All models had better results when learning rate was 0.001. In the RNN cells, 75% of the models exhibited better results in GRU, and 25% had better results in LSTM.

5 Conclusion

We proposed a model for predicting SDVA sentiment in the stock market. According to all experimental results, the prediction performance of our proposed HAHTKN model has outperformed other HAN-based and machine learning-based baseline models. We used the title, keywords and summary of stock market–related messages to estimate all vectors and then calculated the sentence vector in the sentence encoder using Luong attention mechanism. The logic of the sentence encoder was similar to humans. In addition to identifying positive and negative stock market trends, the motivation of buying and selling is more important; the proposed HAHTKN model is expected to generate profit for investors because the thinking is similar to that of human beings when judging the stock market. In this experiment, it was proved that CNN feature extraction can effectively improve the effectiveness of the model when applied to our data set. Our proposed model has used word embedding to transform texts to vectors, but it cannot recognize words with multiple meanings. Our future works will focus on other pre-trained word embedding models such as the Elmo and BERT to improve performance in word embedding of keyword, title and word encoder and will include additional sentiment corpora to allow the system to capture more sentiment information such as sentiment word with valence and arousal.

Acknowledgments This research was partially supported by the Ministry of Science and Technology, Taiwan (Grant No. MOST-107-2218-E-031-002-MY2 and MOST-109-2221-E-031-003).

Compliance with ethical standards

Conflict of interest We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

References

Abdi A, Shamsuddin SM, Hasan S, Piran J (2019) Deep learning-based sentiment classification of evaluative text based on multi-feature fusion. Inf Process Manag 56(4):1245–1259

Abreu J, Fred L, Macêdo D, Zanchettin C (2019) Hierarchical attentional hybrid neural networks for document classification. Proc ICANN 2019:96–402

Agarwal S, Kumar S, Goel U (2019) Stock market response to information diffusion through internet sources: a literature review. Int J Inf Manag 45:118–131

Basak S, Kar S, Saha S, Khaidem L, Dey SR (2019) Predicting the direction of stock market prices using tree-based classifiers. N Am J Econ Finance 47:52–567

Batra R, Daudpota SM (2018) Integrating StockTwits with sentiment analysis for better prediction of stock price movement. Proc iCoMET 2018:1–5

Beninger J, Hamilton-Wright A, Walker H, Trick LM (2020) Machine learning techniques to identify mind-wandering and predict hazard response time in fully immersive driving simulation. Soft Comput. https://doi.org/10.1007/s00500-020-05217-8

Carnicero J, Rojas D (2019) Chapter 8—Healthcare decision-making support based on the application of big data to electronic medical records: a knowledge management cycle. In: Kobeissy F, Alawieh A, Zaraket FA, Wang K (eds) Leveraging biomedical and healthcare data. Academic Press, New York, pp 121–131

Chen YJ, Chen YM, Lu CL (2016a) Enhancement of stock market forecasting using an improved fundamental analysis-based approach. Soft Comput 21(13):3735–3757

Chen W, Cai Y, Lai K, Xie H (2016b) A topic-based sentiment analysis model to predict stock market price movement using Weibo mood. Web Intell 14(4):287–300

Chiang WC, Enke D, Wu T, Wang R (2016) An adaptive stock index trading decision support system. Expert Syst Appl 59:195–207

Ding X, Zhang Y, Liu T, Duan J (2016) Knowledge-driven event embedding for stock prediction. Proc COLING 2016:2133–2142

Dong YH, Chen H, Qian WN, Zhou AY (2015) Micro-blog social moods and Chinese stock market: the influence of emotional valence and arousal on Shanghai composite index volume. Int J Embed Syst 7(2):148–155

Feuerriegel S, Gordon J (2018) Long-term stock index forecasting based on text mining of regulatory disclosures. Decis Support Syst 112:88–97

Gálvez RH, Gravano A (2017) Assessing the usefulness of online message board mining in automatic stock prediction systems. J Comput Sci 19:43–56

Gao S, Young MT, Qiu JX, Yoon HJ, Christian JB, Fearn PA, Tourassi GD, Ramanthan A (2018) Hierarchical attention networks for information extraction from cancer pathology reports. J Am Med Inf Assoc 25(3):321–330

Gaurav D, Tiwari SM, Goyal A, Gandhi N (2020) Machine intelligence-based algorithms for spam filtering on document labeling. Soft Comput 24:9625–9638

Ge Y, Qiu J, Liu Z, Gu W, Xu L (2020) Beyond negative and positive: exploring the effects of emotions in social media during the stock market crash. Inf Process Manag. https://doi.org/10.1016/j.ipm.2020.102218

Hasan M, Rundensteiner E, Agu E (2019) Automatic emotion detection in text streams by analyzing Twitter data. Int J Data Sci Anal 7(1):35–51

Henrique BM, Sobreiro VA, Kimura H (2018) Stock price prediction using support vector regression on daily and up to the minute prices. J Finance Data Sci 4(3):183–201

Ikonomakis EK, Kotsiantis S, Tampakas V (2005) Text classification using machine learning techniques. WSEAS Trans Comput 4(8):966–974

Ingle V, Deshmukh S (2017) Predictive mining for stock market based on live news TF-IDF features. Int J Auton Comput 2(4):341–365

Islam MR, Al-Shaikhli IFT, Abdulkadir A (2018) A scientific review of soft-computing techniques and methods for stock market prediction. Int J Eng Technol 7(2.5):27–31

Jaeger SR, Roigard CM, Jin D, Vidal L, Ares G (2019) Valence, arousal and sentiment meanings of 33 facial emoji: insights for the use of emoji in consumer research. Food Res Int 119:895–907

Khan W, Malik U, Ghazanfar MA, Azam MA, Alyoubi KH, Alfakeeh AS (2019) Predicting stock market trends using machine learning algorithms via public sentiment and political situation analysis. Soft Comput 24:11019–11043

Kim M, Park EL, Cho S (2018) Stock price prediction through sentiment analysis of corporate disclosures using distributed representation. Intell Data Anal 22(6):1395–1413

Li B, Chan KCC, Ou C, Ruifeng S (2017) Discovering public sentiment in social media for predicting stock movement of publicly listed companies. Inf Syst 69:81–92

Liang W, Xie H, Rao Y, Lau RYK, Wang FL (2018) Universal affective model for readers' emotion classification over short texts. Expert Syst Appl 114:322–333

Maqsood H, Mehmood I, Maqsood M et al (2019) A local and global event sentiment based efficient stock exchange forecasting using deep learning. Int J Inf Manag 50:432–451

Max P, Simon S, David G et al (2020) The individual dynamics of affective expression on social media. EPJ Data Sci. https://doi.org/10.1140/epjds/s13688-019-0219-3

Nti IK, Adekoya AF, Weyori BA (2020) A systematic review of fundamental and technical analysis of stock market predictions. Artif Intell Rev 53:3007–3057

Oliveira N, Cortez P, Areal N (2017) The impact of microblogging data for stock market prediction: using Twitter to predict returns, volatility, trading volume and survey sentiment indices. Expert Syst Appl 73:125–144

Pagolu VS, Challa RKN, Panda G, Majhi B (2016) Sentiment analysis of Twitter data for predicting stock market movements. In: Proceedings of the 2016 international conference on signal processing, communication, power and embedded system (SCOPES), pp 1345–1350

Patel J, Shah S, Thakkar P, Kotecha K (2015) Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. Expert Syst Appl 42:259–268

Ranco G, Bordino I, Bormetti G, Caldarelli G, Lillo F, Treccani MT (2016) Coupling news sentiment with web browsing data improves prediction of intra-day price dynamics. PLoS ONE 11:e0146576

Rohini M, Surendran D (2020) Toward Alzheimer's disease classification through machine learning. Soft Comput. https://doi.org/10.1007/s00500-020-05292-x

Russell JA (1980) A circumplex model of affect. J Pers Soc Psychol 36(6):1161–1178

Salehan M, Kim D (2020) An investigation of predictors of information diffusion in social media: evidence from sentiment mining of Twitter messages. HICSS. https://doi.org/10.24251/HICSS.2020.098

Saumya S, Singh JP, Kumar P (2016) Predicting stock movements using social network. Proc I3E 2016:567–572

Shah D, Isah H, Zulkernine F (2018) Predicting the effects of news sentiments on the stock market. In: Proceedings of 2018 IEEE international conference on big data (IEEE big data), pp 10–13

Shang C, You F (2019) Data analytics and machine learning for smart process manufacturing: recent advances and perspectives in the big data era. Engineering 5:1010–1016

Shi L, Teng Z, Wang L, Zhang Y, Binder A (2018) DeepClue: visual interpretation of text-based deep stock prediction. IEEE Trans Knowl Data Eng 31(6):1094–1108

Song S, Huang H, Ruan T (2019) Abstractive text summarization using LSTM-CNN based deep learning. Multimed Tools Appl 78(1):857–875

Stefan F, Helmut P (2014) News-based trading strategies. Decis Support Syst 90:65–74

Tsaptsinos A (2017) Lyrics-based music genre classification using a hierarchical attention network. Proc ISMIR 2017:694–701

Vanstone BJ, Gepp A, Harris G (2019) Do news and sentiment play a role in stock price prediction? Appl Intell 49(11):3815–3820

Wang Y (2017) Stock market forecasting with financial micro-blog based on sentiment and time series analysis. J Shanghai Jiaotong Univ (Sci) 22(2):173–179

Wang F, Zhang Y, Rao Q, Li K, Zhang H (2016) Exploring mutual information-based sentimental analysis with kernel-based extreme learning machine for stock prediction. Soft Comput 21(12):3193–3205

Warriner AB, Kuperman V, Brysbaert M (2013) Norms of valence, arousal, and dominance for 13,915 English lemmas. Behav Res Methods 45(4):1191–1207

Wu J, Yu L, Chang P (2014) An intelligent stock trading system using comprehensive features. Appl Soft Comput 23:39–50

Wu GG, Hou TC, Lin J (2019a) Can economic news predict Taiwan stock market returns. Asia Pac Manag Rev 24(19):54–59

Wu J, Yang C, Liu K, Huang M (2019b) A deep learning model for dimensional valence–arousal intensity prediction in stock market. Proc IEEE iCAST 2019:1–6

Yadav R, Kumar AV, Kumar A (2019) News-based supervised sentiment analysis for prediction of futures buying behavior. IIMB Manag Rev 31(2):157–166

Yang Z, Yang D, Dyer C, He X, Smola A, Hovy E (2016) Hierarchical attention networks for document classification. In: Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies (NAACL HLT), pp 1480–1489

Ying H, Zhuang F, Zhang F et al (2018) Sequential recommender system based on hierarchical attention network. In: Proceedings of the IJCAI-ECAI-18, pp 3926–3932

Zhang J, Cui S, Xu Y, Li Q, Li Y (2018a) A novel data-driven stock price trend prediction system. Expert Syst Appl 97:60–69

Zhang X, Zhang Y, Wang S, Yao T, Fang B, Yu PS (2018b) Improving stock market prediction via heterogeneous information fusion. Knowl Based Syst 143:236–247