Expert Systems with Applications 158 (2020) 113540

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Semi-supervised learning with generative model for sentiment classification of stock messages

Jiangjiao Duan [a], Banghui Luo [b], Jianping Zeng [b,c,*]

[a] Business School, University of Shanghai for Science and Technology, Shanghai, 200093, China
[b] School of Computer Science, Fudan University, Shanghai, 200433, China
[c] Engineering Research Center of Cyber Security Auditing and Monitoring, Ministry of Education, Shanghai, 200433, China

ARTICLE INFO

Article history:
Received 6 February 2019
Revised 7 February 2020
Accepted 7 May 2020
Available online 11 May 2020

Keywords:
Sentiment analysis
Generative model
Semi-supervised learning
Stock message board

ABSTRACT

Classification of investors' sentiments in stock message boards has attracted a great deal of attention. Since the messages are usually short, we propose a semi-supervised learning method to make full use of the features in both train and test messages. The generative emotion model takes message, emotion and words into consideration simultaneously. Based on the facts that words are of different ability in discriminating sentiments, they are categorized into three classes in the model with different emotion strength. Training the generative model can transform the messages into emotion vectors which finally feeds to a sentiment classifier. The experiment results show that the proposed model and learning method are efficient for modeling sentiment in short text, and by properly selecting the amount of train data and the percent of test samples, we can achieve higher classification accuracy than traditional ones. The results indicate that the generative model is effective for short message sentiment classification, and provides a significant approach for the implementation of semi-supervised learning which is a typical expert and intelligent information processing method.

1. Introduction

Financial expert systems play an important role in making decision on stock investment. The experts should deal with all kinds of related information, such as stock transaction data, behavior in social media, sentiments in message boards, and so on. While, machine learning is a potential approach to process the various information, and extract the features hidden in different sources. Hence, it is always significant to do thorough insight into how to improve the performance of machine learning on the stock market related data.

The paper pays attention to the sentiment in message boards. As we know, more and more online stock message boards become popular, people are willing to join the discussions in social media. It is commonly approved that investors' sentiment in social media has impact on stock markets (Bartov, Faurel, & Mohanram, 2017; Renault, 2017). The positive sentiment might motivate investors to buy stocks, while negative sentiment likely triggers the intention of selling shares. Hence, the sentiment in messages is an important indicator for the prediction of investor behaviors.

The quantization approach for sentiment has drawn much attention from researchers (Hasan & Fong, 2018; Rajput & Dubey, 2016; Volkova, Wilson, & Yarowsky, 2013). Several methods take explicit variables as features, such as the number of messages, the number of active users, and so on, by performing simple calculation on the message board (Tumarkin, 2002). Although the values are easy to attain, they are weak in describing the investors' fine-grained sentiment since their real opinions are ignored in these variables. Therefore, many researches capture sentiment directly from the messages posted by users (Nguyen, Shirai, & Velcin, 2015; Oliveira, Cortez, & Areal, 2017). New approaches to predict stock price based on message board activities and sentiment has been proposed (Ruiz, Hristidis, Castillo, & Gionis, 2012; Zhao, He, Yuan, & Huang, 2016).

Because the sentiment expression in social media varies a lot, automatically detecting sentiment of stock messages becomes a challenge task. Simple approaches take pre-defined keywords which are of certain sentiments to calculate the degree of sentiment in messages (Oliveira, Cortez, & Areal, 2016). While, sophisticated methods consider the sentiment recognition as a classification problem, and then employ machine learning to infer the sentiment (Rajput & Dubey, 2016). Hence, around the related work of classification, there are many further researches, such as selection of sentiment classifier (Rajput & Dubey, 2016), feature

* Correspondence author.
E-mail addresses: jjduan@usst.edu.cn (J. Duan), bluo13@fudan.edu.cn (B. Luo), zjp@fudan.edu.cn (J. Zeng).

https://doi.org/10.1016/j.eswa.2020.113540
0957-4174/

selection, feature transformation (Ming, Wong, Liu, & Chiang, 2014), and so on. Various stock social medias, such as Saudishares forum (Hasan & Fong, 2018), StockTwits (Renault, 2017), and so on, provide enough message data sets to evaluate the classification performance. According to the evaluation results, the features used in the representation of text messages have great impact on the performance.

In this paper, we propose a new implementation method for semi-supervised machine learning based on the generative model which is designed according to the mechanism of message generation. The model named as **GEM-CW** (Generative Emotion Model with Categorized Words) can performed sentiment feature extraction from stock messages. User emotions are considered as an important factor that can influence the usage of words in the message. Therefore, the probability distribution of emotion in messages can be estimated by fitting the model to a given message text set. Based our previous research on stock message board (Luo, Zeng, & Duan, 2016), the new model takes seven kinds of emotions as the representative dimension of message. Furthermore, according to the intuitive ability in discriminating sentiments, all of the words in message board are classified into three categories, that is, origin words which are of explicit sentiment, synonym words which are similar words of origin words, and relevant words which are selected from the message corpus according to n-gram algorithm. So the categorized words are used to refine the model so that the probability distribution of words for a given emotion can be estimated more accurately.

The main contributions of the paper are as follows.

Firstly, the semi-supervised learning method for stock text messages is proposed. In stock message sentiment classification, the sentence is usually short, which lead to great sparsity in arithmetic representation. The paper employs the semi-supervised learning method, and takes the train set and test set together to avoid the affection of short messages. Thus, the inferred features can be more comprehensive than those that are derived by traditional learning methods which just consider train set. While, current methods for this question adopt text extension methods which introduce synonym words to make the message longer. Another approach is to transform the one-hot representation to distributional representation (Wang, Xu, & Xu, 2016). The performance of the approaches heavily depends on the corpus that are used to determine the extension and density representation.

Secondly, the idea of generative model is employed to realize the semi-supervised learning method. The generative model GEM-CW quantitatively describe the probability distribution about messages, word and sentiment. GEM-CW divides word space into three parts in which the words are with different emotion strength. In this way, we can distinguish the probability distribution of different types of words, as a result, the model can accurately estimate the fine-grained sentiment in messages. However, for other generative topic models, the topic usually has no clear semantic and finally limits the performance in feature extraction.

Finally, semi-supervised learning with generative model can be of valuable reference in expert and intelligent systems. In the systems, we might construct machine learning framework for large amount of short text. However, the sparsity of the word can degrade the performance of traditional machine learning methods. While, semi-supervised learning with generative model can well organize the train set and test set, and thus the extracted features can be strongly related to the data set. The application methods in the experiments show that the idea of generative model indeed provides a significant approach for the implementation of semi-supervised learning in intelligent systems.

The rest of this paper is organized as follows. In Section 2, related work is surveyed. The proposed model and parameter inference are described in details in the third section. In Section 4, the sentiment classification framework based on GEM-CW is presented. The data set and the experiments are described and the results are analyzed in Section 5. The final section concludes the work and points out the future work.

2. Related work

More and more investors express their opinion about stock market in social media, such as stock message board, twitter, so the overall sentiment in the messages has been considered as a new indicator of the status of stock market. The requirement of stock market prediction has promoted the research on sentiment analysis of stock social media. The predictability of financial movements using online information from twitter is surveyed based on the burgeoning literature (Nardo, Petracco-Giudici, & Naltsidis, 2016). Some research suggests that twitter sentiment is relevant for the forecasting of returns of S&P 500 index, and then confirms the usefulness of microblogging data for financial expert systems (Oliveira et al., 2017). Using a broad sample from 2009 to 2012, it is found that the aggregate opinion from individual tweets can successfully predict a firm's forthcoming quarterly earnings and announcement returns (Bartov et al., 2017).

Sentiment analysis approaches for stock message boards are generally categorized into two types, that is, heuristic method and classification method. For heuristic rule-based methods, keyword-based patterns which are associated with a sentiment are defined in a dictionary firstly, and then the sentiment in a message is calculated by matching the patterns in the message and the dictionary. A lexicon of words used by online investors from the social media platform Stock Twits is constructed, the results show that it is helpful for forecasting intraday stock index returns by aggregating individual message sentiment at half-hour intervals (Renault, 2017). In (Hasan & Fong, 2018), two corpora and one lexicon are manually built, and then sentiment is analyzed on Saudi shares forum, one of the biggest Saudi investment forums. They found that the rule-based heuristic approach can achieve the best performance. Sentiment lexicon is thus an important kind of knowledge for the improvement of sentiment calculation, and an automatic acquisition approach for the stock market sentiment lexicon is proposed using microblogging data (Oliveira et al., 2016). Although the method is simple to implement, the accuracy of the sentiment calculation is greatly depended on the prior patterns defined in the dictionary.

Sentiment analysis based on classification methods has become predominant in research and applications. Many researches focus on the selection of classifier since the performance of different classifiers varies a lot. A comparative study of naïve Bayes and SVM on the opinions of the reviewers of the stock market is presented in (Rajput & Dubey, 2016). Similar to common classification tasks, combination of different classifiers is also employed in text sentiment analysis. Yang applied SVM and CRF learners to classifying sentiments at the sentence level and then investigated several strategies to determine the overall sentiment of the document (Yang, Lin, & Chen, 2007).

Features in stock messages and their representation method are critical in training various classifiers, so they have attracted much attention recently. An efficient and general way to retrieve more useful prior knowledge as the features is a matter of great importance. While, several classical feature selection approaches, such as IG (Information Gain), MI (Mutual Information), and so on, are widely utilized in the selection of words from messages (Oliveira et al., 2016). However, the effectiveness of the selected features is limited since the messages in social media are usually short and the opinion expression is very flexible.

Feature transformation method which represents a message with some latent information rather than words, attracts much attention. Utilizing the latent space model proposed by Ming et al. (2014), the movements of both stock prices and social media content are found correlate (Sun, Lachanski, & Fabozzi, 2016). One of the typical transformation methods is to convert the messages into a new semantic space, that is, the topics about the messages. Latent Dirichlet Allocation (LDA) (Blei, Andrew, & Jordan, 2003) which is a widely used topic model can provide a method for the conversion. It has been widely applied in its extended forms in sentiment analysis and many other tasks (Moghaddam & Ester, 2011; Mukherjee & Liu, 2012). Iwata, Watanabe, Yamada, and Ueda (2009) utilize a continuous Dirichlet Process Mixture model to leverage topic for stock analysis. The joint sentiment/topic model is proposed to detect sentiment and topic simultaneously for movie review dataset (Lin & He, 2009), and the sentiment on the topics are automatically extracted to predict the move of stock market (Nguyen et al., 2015). Recently, a topic-dependent attention model based on Gated Recurrent Unit (GRU) is proposed for sentiment classification (Pergola, Gui, & He, 2019).

As a new feature extraction method, deep learning can extract many implicit features, and hence plays an important role in text analysis. An automatic lexicon creation approach based on neural network is proposed, and the idea is to incorporate the sentiment information into the neural network to learn the embeddings (Li & Shah, 2017). Hence it can capture the sentiment information as well as the syntactic contexts of words (Li & Shah, 2017). Recurrent neural networks with character-level language model are employed to model the changes in the stock price of the company mentioned in the news article and in the corresponding stock exchange index (S&P 500) (Pinheiro & Dras, 2017). A combined method for sentiment analysis based on the lexicon, feature selection and word embedding is developed, and it can achieve progressive performance (Zhao et al., 2016).

3. GEM-CW model

In this section, we present the GEM-CW (Generative Emotion Model with Categorized Words) model. In the model, we borrow the idea of generative topic model, establish a text as a probability distribution over the seven-dimensional emotion space, and represent the emotion as a probability distribution over words. Specially, the words are categorized into three kinds which provide different weights on sentiment. These words include origin words obtained from WordNet Affect, extended words which is semantically similar to WordNet words, and relevant words extracted from message corpus. Specifically, the prior emotional words are applied to simulate the iteration process on Gibbs sampling which estimates the parameters of GEM-CW.

3.1. Definition

For a given message text set, we have a vocabulary containing all of the distinctive words in corpus, and it is denoted as $V = \{w_1, w_2,\ldots,w_L\}$, and L is the total number of words. Let $C = \{d_1, d_2,\ldots, d_M\}$ be set of documents in the corpus. M is the number of the documents. Each document is denoted by a sequence of N_i words, that is, $d_i = \{w_1, w_2,\ldots,w_{Ni}\}$. The goal of the classification is to assign each document with a label "Buy", "Sell" and "Hold".

Six kinds of Ekman emotion serve an important role in the GEM-CW model. However, there are posts that state no sentiments in stock board message. Hence we add "no_emotion" to the emotion space for GEM-CW model, and the final dimension of the emotion space is 7.

Definition 1 (*(Emotion Space)*). *The emotion space is a seven-dimensional space, with addition and scalar multiplication operations defined on it. The dimensions are named anger, disgust, joy, fear, sadness, surprise, and no_emotion respectively. The emotion space is formulated as,*

$$E = \{e_1, e_2, \ldots, e_7\} \tag{1}$$

After the introduction of emotion space, the relation between message text and words can be divided into two relations according to the idea of topic modeling. The relations are the emotions in a given message, and the words for each emotion. The former is represented by a distribution $p(e_i|d_j)$, i = 1, 2, 3, 4, 5, 6, 7, and j = 1, 2, .., M, that means each emotion e appears with a certain probability in the message d.

Similarly, each emotion is characterized by a multinomial distribution over all the words in V, that is, $p(w_i|e_j)$, i = 1, 2, …, L, and j = 1, 2, 3, 4, 5, 6, 7.

Definition 2 (*(Categorized words)*). *The words in the messages can be grouped into three categories according to their sentiment weights. The categorized words are represented as,*

$$CW = \{Y_1, Y_2, Y_3\} \tag{2}$$

For those words in Y_1, they are of explicit emotions. We can simply employ the word set organized by Ekman model. The number of words for each kind of emotion is 399, 252, 197, 144, 71 and 53 for joy, anger, sadness, fear, surprise, disgust, respectively. Although these words are of explicit emotion, the number of words is small. Hence, we extend the words according to word similarity, and put those words with high similarity to Y_2. Finally, those words that are not in Y_1 and Y_2 but in the message set are put into Y_3. It is clear that the words in Y_3 might be of less sentiment or probably domain depended.

3.2. Utilization of the prior knowledge

Users might select words to express their attitude towards the stock market situations according to their emotions. For example, when stock price goes down rapidly, the investors tend to write messages with angry type words. However, the word emotion should be determined by the context and background, which are considered as the prior knowledge. Correspondingly, this knowledge may be obtained from public domain, knowledge-based systems and training message. However, it is impossible to gain enough prior knowledge or training data for every ad hoc domain. Therefore, we use three ways to get the prior knowledge.

3.2.1. OriginWords

The *OriginWords* means those words from public domain. We employ the Ekman emotion words proposed by the notable psychologists (Ekman, Sorenson, & Friesen, 1969) as the *OriginWords*. It is supposed that there are six basic emotions, that is, anger, disgust, joy, fear, happiness, sadness, and surprise in human, and they are independent of different culture. So the Ekman emotion words can act as a good public domain words, and it is suitable to incorporate the Ekman emotions into the generative model. We obtain the words belonging to Ekman emotion from WordNet Affect, and then save these words to a set of words named *OriginWords*, that is Y_1 in CW.

3.2.2. SynonymWords

Then, we compute the emotion's probability of synonyms of *OriginWords*. Synonyms are obtained from online synonyms dictionary. We assume that each word has a multinomial distribution over emotion categories E. We define a probability variable

$prob_k(w)$, which represents the probability of word w belonging to e_k. A word's emotions probability is the same with its synonyms. Specifically, if a word is a synonym of two or more emotion categories, we assume that the probability is the average. For example, the word "*blessed*" is synonymous of emotion category "*joy*" and "*surprise*". As a result, $prob_3(blessed) = 1/2$ and $prob_6(blessed) = 1/2$. We save the synonyms to a word set named *SynonymWords*, that isY_2 in CW.

3.2.3. RelevantWords

Lastly, we assume that words, co-occurring in a message text, are more likely to express the same emotions. So, for a word w without emotion labels, we find the co-occurrence words in the training set. The co-occurrence is based on the popular n-gram algorithm. Here, we use n = 3, that means those words fall into the window sized 3 are selected. In this case, $prob_k(w) = \sum_{wc \in cowords(w)} probk(wc)$, where $cowords(w)$ is the set of the co-occurrence words of w.

Finally, we save the words to *TrainedWords* and represented it as Y_3 in CW.

3.3. Generative process

GEM-CW is a kind of generative model that can incorporate message, emotion, words and word category together. The corresponding probabilistic graph representation is shown in Fig. 1. The model describes a message set containing M messages (documents), K ($K = 7$) emotions, and each message has N words.

We briefly explain the notations we used in the model. For convenience, we define that the notation $Mult()$ is an abbreviation of multinomial distribution, and $Dir()$ is an abbreviation of Dirichlet distribution which is a conjugate prior of multinomial distribution. We denote that $Mult(\varphi(s)\ ij) = P(w_{ij}|e_{ij}, s, \beta)$. s is the label of word category. If $s = 1$, the category is Y_1. If $s = 2$, the category is Y_2. If $s = 3$, the category is Y_3 containing remaining words in the whole message set.

Intuitively, we can interpret that multinomial parameter φ_{ik} as the word w_i distribution in emotion categories e_k, and the 7 dimensions multinomial parameter θ_{kj} as the emotion e_k distribution in document d_j.

Similar to the classical LDA model (Blei et al., 2003), we assume that an author would "write" a message d_j to express his opinions by following the steps a)-(d,

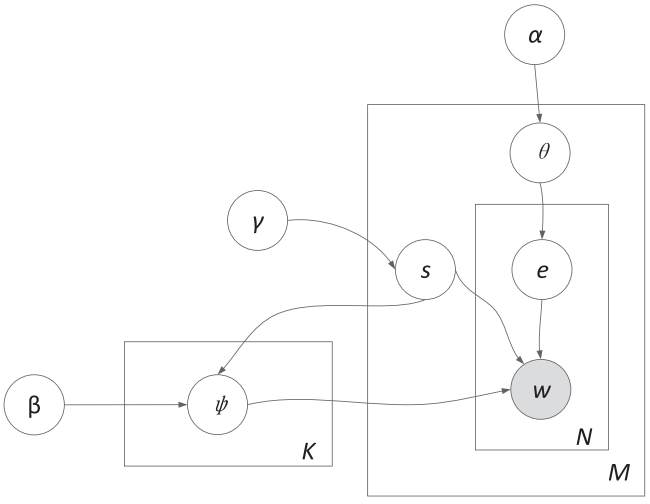

Fig. 1. Framework of GEM-CW.

a) Choose N from $Possion(\zeta)$.
b) The author would first decide the parameter θ, which is sampled from $Dir(\alpha)$.
c) Choose the distribution of word label sampled from $Dir(\gamma)$ for the message.
d) Then for each N word in document d_j, repeatedly execute the following steps.
   i. Choose an emotion e_{ij} from multinomial distribution $Mult(\theta)$.
   ii. Choose a word label s from $Mult(Dir(\gamma))$.
   iii. Choose a word w_{ij} from words set Y_s by $P(w_{ij}|e_{ij}, s, \beta)$, a multinomial probability conditioned on the emotion e_{ij}, word label s and Dirichlet prior parameter β.

Based on the generative process, we can see that a message text is actually a multinomial distribution in the emotion space defined in Section 3.1. Since the authors in stock message boards usually express some kinds of emotions, the distribution is reasonable. In this way, unlike the topic variable in LDA topic models, the GEM-CW model can avoid the unclear semantic expression of the latent variable. Furthermore, a small latent space with 7 dimensions can be kept and thus avoid the complexity in parameter inference.

On the other hand, a message is represented as the multinomial distribution φ over the categorized words. In fact, the parameter φ is composed of $\varphi^{(s=1)}$,$\varphi^{(s=2)}$and $\varphi^{(s=3)}$ which are dependent on Y_1, Y_2 and Y_3, respectively. That is, φ is a matrix of K^*V size, as follow.

$$\varphi = \left[\varphi^{(s=1)}, \varphi^{(s=2)}, \varphi^{(s=3)}\right] \tag{3}$$

Compared with the probabilistic graph model of LDA, we can see that the new random variables are added to determine the generation of words. From the point of view s, we can see that GEM-CW model is more general than LDA. GEM-CW becomes LDA with 7 topics if the words are not distinguished. By categorizing words with emotion, we can infer fine-grained emotion distribution more accurately. Therefore, this is one of the innovations in the research.

3.4. Parameter inference

In this section, we describe the parameter inference algorithm for GEM-CW model. Given an observed word w, the task of the inference is to compute the posterior distribution over emotion categories E, the mixing proportions φ, and emotions distribution θ. We use Gibbs sampling to estimate the latent variables θ and φ. After the sampler has burned-in, the parameters of φ and θ can then be estimated.

In each iteration of Gibbs, the important task is to estimate the distribution $p(e|w)$ over emotion for a given w. Since the words in V are categorized in the GEM-CW model, we can determine $p(e|w)$ for each category.

(1) For words in category Y_1,

Since each word has clear emotion label, so $p(e_i|w) = 1.0$ if the corresponding emotion of w is e_i, and otherwise $p(e_i|w) = 0$, where $i = 1, 2, 3, 4, 5, 6, 7$.

(2) For words in category Y_2,

We set $p(e_i|w) = prob_i(w)$ which is described in Section 3.2.2, and it represents the probability of word w belonging to e_j.

(3) For words in category Y_3,

It is much difficult to infer the distribution since the label of word are unknown and should be estimated. The details are described as follows.

We define summations by $N^{(s)}{}_{wk}$, that is the count of word from words set s assigned to emotion k. $N^{(s)}{}_{k}$ is the number of words assigned to emotion k. Moreover, N_{kj} represents the number of times a word in document j has been assigned to emotion k. N_j is the total number of the words in document j. And then, given

the current state of all but one word w_{ij} from words set s, we have the conditional probability of w_{ij} at position i in document j, the sampling formula can be inferred as bellow.

$$p(e_{ij} = k|e^{\neg ij}, \alpha, \beta) = \frac{1}{Z}\left(N_{kj}^{\neg ij} + \alpha\right)\left(\frac{N_{wk}^{(s)\neg ij} + \beta}{N_{k}^{(s)\neg ij} + V\beta}\right) \quad (4)$$

where Z is a normalization constant and the superscript ij indicates that the corresponding datum has been excluded in the count summations. In other words, we assume that except for w_{ij}, the emotions assigned to rest of whole words in corpus are correct. So, in this way, p($e_i|w$) can be calculated based on the sampling results of e.

At this stage of the determination on p($e|w$) for each category, p($w|e$) should be calculated next. Based on Bayesian theorem, we have

$$p(w|e) = p(e|w)p(w)/p(e) \quad (5)$$

where, p(w) is the priori which can be calculated based the message text, and p(e) can be consider as constant for each emotion. So p($w|e$)$\propto$p($e|w$)p(w).

Hence, given a sample we can then get the estimation of φ for the words in category Y_3, as follows.

$$\varphi_{wk}^{(s)} = \frac{N_{wk}^{(s)} + \beta}{N_{k}^{(s)} + V\beta} \quad (6)$$

From eq. (5), p($e|w$) comes from different categories Y_1, Y_2 and Y_3, and p($e|w$) for Y_1, Y_2 is calculated based on prior knowledge. Thus, in this way, we avoid estimating p($e|w$) for all of the words, and can effectively improve the parameter accuracy.

In a similar way, given a sample, the parameter θ can be estimated as follow.

$$\theta_{kj} = \frac{N_{kj} + \alpha}{N_j + 7\alpha} \quad (7)$$

To sum up the above analysis, we present the procedure to sample e in an iteration of Gibbs sampling which is a commonly used inference method for parameters in probabilistic graph model. For each word w_{ij} in document d_j,

(1) If w_{ij} is the origin emotional words in Y_1, the exact emotion label is sampled to it.
(2) Else if the word is in Y_2, then sample e_{ij} according to its probability $prob_i(w)$ which is described in Section 3.2.2.
(3) Otherwise, sample e_{ij} from the sampling formula (4).

4. Semi-supervised learning with GEM-CW for sentiment classification

In this section, we present the semi-supervised learning method with the application of GEM-CW to classify sentiment for stock message board. Typically, a classification task generally includes two procedures, that is, training and testing. Supervised learning is widely used in classification, and another approach is SSL (Semi-supervised learning) which gains much attention recently. As far as GEM-CW is concerned, it is very suitable for SSL dataset which includes samples with class label and samples without class label. The goal is to correctly classify the samples without class label.

The general process is shown in Fig. 2. The main steps are explained as follows.

(1) A set of train messages is necessary, and at the same time, test messages are selected from stock message board. The two kinds of message are combined and supplied to GEM-CW model. That is, the corpus in Section 3.1 is C={d_1, d_2,..., d_M}, where M is the total number of the messages.

(2) GEM-CW is learnt from the corpus C. In this process, prior knowledge about categorized word set Y_1 and Y_2 is required. After finishing GEM-CW model parameter inference, parameter φ and θ are attained in eq.(6) and eq.(7), respectively. And θ is the multinomial distribution p($e|d$) for each message over emotion space. In this way, the message text set can then be converted to new mathematical representations, and each message d can be represented as a vector dv,

$$dv = (p(e_1|d),\ p(e_2|d),\ p(e_3|d),\ p(e_4|d),\ p(e_5|d),\ p(e_6|d),\ p(e_7|d)) \quad (8)$$

Note that, in this procedure, train set and test set are put together to infer sentiment features, and we do not need the labels in the train set. By this approach, two vector sets *dvs_train* and *dvs_test* can be achieved, and they can be denoted as follows.

dvs_train={dv_i, l_i}, i = 1...tn, where tn means the number of train samples, and l_i is the class label of the i^{th} train message in C. *dvs_test*={dv_i}, i = 1...sn, where sn means the number of test samples.

(3) A classifier is trained by using *dvs_train* which is corresponding to the train messages. Various kinds of classifiers, such as Decision Tree, SVM, KNN, Bayes net, and so on, can be employed in this scene. Maybe parameters related to the classifiers should be optimized in the process. For example, the kernel function in SVM, and the number of neighbors in KNN are parameters which have impact on the performance of classification.

(4) After the classifier is attained, it can be used to predict the class label for those samples without class label in C. This is achieved by the calculation *on dvs_test* using the classification function provided by specific classifier.

Based on the classification procedures, we can see that GEM-CW actually serves as a feature extraction method. Test messages and train messages are treated equally and converted into vector representations. The classifier can predict the test messages which have been processed by GEM-CW. Therefore, the classification in the case is actually a transductive learning which is a special case of semi supervised learning (Bianchini, Belahcen, & Scarselli, 2016). Several typical feature extraction methods, such as SVD (Singular Value Decomposition), PCA (Principal Component Analysis), and so on, have been widely used in classification systems. However, being different from these methods, GEM-CW model can convert message text into a vector in emotion space, and hence has much explicit meaning related to sentiments.

5. Experiment

In the section, we design experiments to evaluate the GEM-CW and semi-supervised learning approach for classifying sentiments in stock message board. Especially, the mechanism of categorized words and emotion space are evaluated. Furthermore, the classification performance is compared with several previous methods, such as LDA, sentiment-LDA (Li, Huang, & Zhu, 2010) and ESM (Explicit Sentiment Model) (Luo et al., 2016).

5.1. Dataset and experiment method

We collect stock message comments data for experiments from TheLion forum (http://thelion.com). The forum is a booming financial community site for active traders and investors. It offers a highly-integrated community for the due diligence of all stocks. People can post messages on the board to express their opinions on stocks. Most importantly, the forum offers an option for users to select the opinion label of their posts. There are seven kinds of

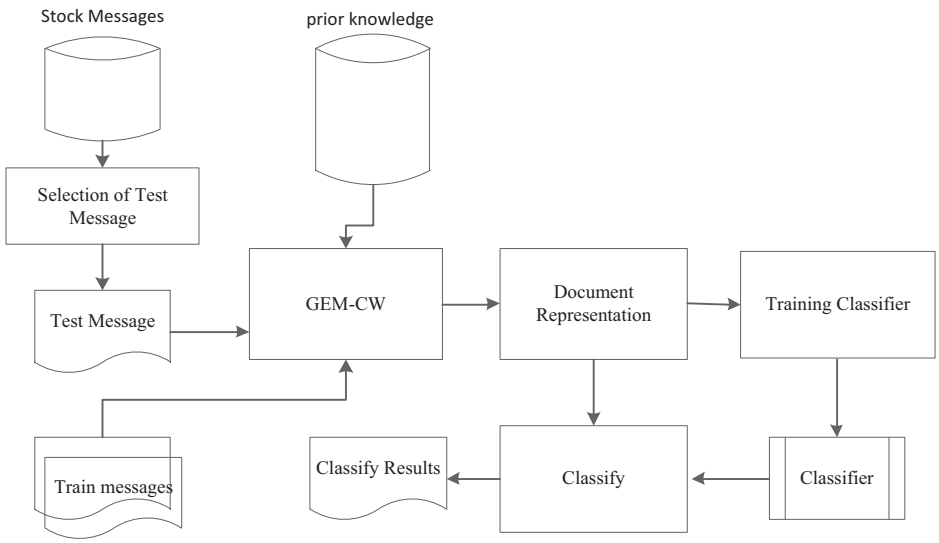

Fig. 2. Semi-supervised learning with GEM-CW for Sentiment classification.

tags {Strong Buy, Buy, Hold, Scalp, Short, Sell, Strong Sell} to express user's positive and negative attitudes. For example, the snapshot in Fig. 3 is five posts about JD from TheLion.

The categories of "Strong Buy", "Scalp" and "Buy" are grouped into "Buy", and categories of "Short", "Strong Sell" and "Sell" into Sell. Finally, we acquire a total of 7,000 comments marked with user's sentiments, 3000 for Buy, 3000 for Sell, and 1000 for Hold.

The prior knowledge about Y_1 and Y_2 are critical in the proposed method, hence we present the number of words in the two set for each Ekman emotion, as shown in Table 1, respectively.

Two kinds of experiments are done on the dataset. The first one is to verify the effectiveness the configuration of the semi-supervised learning for classification based on GEM-CW. The experiments are designed to verify the impact of three factors, that is, the categorized words, the classifier and the ratio of test message and train messages. The second type of experiments focuses on the comparison of the proposed method to the current approaches, such as traditional LDA, Sentiment-LDA, and ESM (Explicit sentiment model).

The performance in the experiments is measured by classification accuracy which is defined as N_c/N_a. N_c is the number of test messages that can be correctly classified, and N_a is the number of test messages.

5.2. Experiments on the effectiveness of semi-supervised learning

In the experiment, we would like to evaluate the factors that might have impacts on the performance in semi-supervised classification based on GEM-CW. The factors are categorized words, classifier, and the ratio of test message and train messages.

Table 1
Prior knowledge about Y_1 and Y_2.

Number	Joy	Anger	Disgust
Y_1	512	276	58
Y_2	1401	622	1321
	Fear	**Sadness**	**Surprise**
Y_1	211	298	97
Y_2	1606	576	875

5.2.1. The effectiveness of the categorized words

One of the novelty of proposed method lies in that words are categorized according to their emotion strength. Hence, we do experiments with different combinations of categorized words. And in the framework, we select decision tree as the classifier, and the ratio of test message and train messages is set to 0.5:9.5. The classification accuracy is shown in Table 2.

The accuracy becomes better when combing Y_1, Y_2, and $Y_1 + Y_2$ with Y_3, respectively. And the performance of combinations of $\boldsymbol{Y_1 + Y_2 + Y_3}$ is the best. The results show that the categorized words are necessary and effectiveness in classification. By categorizing the words, the importance of different words can be distinguished, and thus the quality of the inferred features can be improved.

5.2.2. The impact of classifier selection

In the framework of sentiment classification for stock message, classifier is utilized to distinguish different sentiment in emotion space. There are many choices for the classifier. Hence, we do experiments with different classifiers. We select traditional classifiers of SVM, Bayes and Decision tree since they are widely used. In

Msg #	Message Preview	Sentiment	Rec	Author	Date (ET)
22	**BABARAN:** Adding here Re: Grabbed a few calls for BABA fun. eom		-	ChineseChicken	09/15/2014 09:47
21	**BABARAN:** Grabbed a few calls for BABA fun.	Buy	-	ChineseChicken	09/11/2014 10:33
20	Even though Alibaba is the leader in Chinese e-commerce, JD is clearly a form...	Strong Buy	-	Sage	08/27/2014 13:54
19	love this one eom	Strong Buy	-	Sage	08/27/2014 12:06
18	**NUFF4ME:):** Out all common up nearly $5 and rest of calls up 225% booooooom!!! eom	Sell	-	EPR53167	08/27/2014 10:13

Fig. 3. Snapshot of comments for JD.com – JD.

Table 2
Classifying Results Using Different Categorized Words.

Categorized words	Y_3	$Y_1 + Y_3$	$Y_2 + Y_3$	$Y_1 + Y_2 + Y_3$
Accuracy	72.5%	73.2%	74.5%	76.3%

the experiment, we use all of the categorized words, and the ratio of test message and train messages is also set to 0.5:9.5. The performance is shown in Table 3.

We can see that the performance of Decision Tree is the best, and Bayes is the worst. However, the accuracy of the classifiers is not distinguished too much. The results show that classifier has impact on the classification results.

6. The impact of the ratio of test messages and train messages

In the framework, we select test messages, combine them with the train messages, and finally supply as a whole message set to GEM-CW to extract emotion features. The ratio of test messages and train messages (*sn*/*tn*) might be an important factor in the framework. In the experiment, we use categorized words, select decision tree as the classifier, while the ratio of test messages and train messages varies. Meanwhile, the total number of train samples is set to 500 and 3000, respectively. The classification accuracy is shown in Fig. 4. *sn*/*tn* is set to 0.3:9.7, 0.5:9.5, 0.7:9.3, 1:9, 2:8, 3:7, 4:6 and 5:5.

As we can see, the accuracy generally increases as the number of test messages increases when the train set is small. With more test messages, the larger corpus can be used to train GEM-CW. Thus, the emotion feature can be captured better. However, under the circumstance of large train set, such relation between the accuracy and the ratio seems not obvious. The reason may lie in that large corpus is enough for capturing the features by the generative model.

6.1. Top emotional words on each emotion

We can get φ, the probability distribution over words for each emotion. Table 4 shows the selected example of top emotional words on each *E*. The emotion probability distribution is intuitively proper. E.g. the word "*strong sell*" is in top of emotion "fear", modal particle word "*wow*" is in top of "*surprise*". It provides a strong evidence to verify the validity of *GEM-CW* model.

6.2. Comparison with other approaches

In the experiment, we would like to compare the proposed method in the paper with the current methods. Since GEM-CW is can be considered as a kind of feature selection methods, we conduct a series of comparative experiments on the same dataset with several commonly used feature selection methods, such as IG (Information Gain), MI (Mutul Information), LDA and ESM. On the other hand, sentiment-LDA which is a similar work is also compared. In the experiments, we select decision tree as the classifier, and the ratio of test messages and train messages is set to 1:9.

A. Comparison with other feature selections

In the experiment, we finally select 80 features based on IG and MI with consideration of accuracy and time consumed. By LDA, we set the number of topics to 7 since there are seven kinds of emotions in the messages. However, all of the words in the messages are considered to have equal weight on the emotion analysis, the accuracy is lower than that of *GEM-CW*. *ESM* is another classification model recently proposed for sentiments in stock messages (Luo et al., 2016). The model is also based on 7-dimension emotion space. However, the weight is determined based on heuristic rule. The performance of *ESM* is better than that of tradition methods, while lower than that of GEM-CW. Table 5 shows the comparative result.

Table 3
Classifying Results Using Different Classifiers.

Classifier	Decision Tree	SVM	Nave Bayes
Accuracy	76.3%	75.4%	72.7%

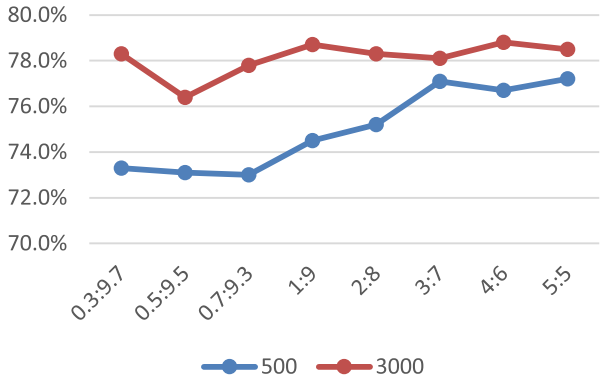

Fig. 4. The impact of train sample size and the ratio of *sn*/*tn*.

Table 4
Examples of Top Emotional Words.

Emotion	Words
Anger	small, pain, stop
Disgust	disclaimer, undervalue, termination
Fear	low, strong sell, lose, upset
Joy	love, approval, strong buy, huge
Sadness	break, gap, cheap, oversold
Surprise	lol, high, buy, !!!, wow, big

The result shows that our model is the best one. In gereral, it verifies that the proposed approach is efficient. The pivotal reason is that our model points out emotions, by explicitly analyzing sentiment on a meaningful space. On a clear semantic analysis, we can know better about what the model exactly does, which makes parameter estimation easier. Meanwhile, many traditional VSMs, based on IG and MI, have a high sensitivity on dimension parameters, while there is always not an inherent way to determine the optimal one. In the proposed model, because of the fixed emotion categories, it is not a concern anymore.

B. Comparison with sentiment-LDA

Sentiment-LDA model (Li et al., 2010) is a generative model that incorporates sentiment in LDA and utilizes three kinds of prior knowledge, that is, HowNet, WordNet Affect and MPQA. The model is similar to our proposed GEM-CW. For a fair comparison, we use the same dataset, which contains 4000 positive and 4000 negative reviews for Amazon products, as in Sentiment-LDA. We use WordNet Affect, because it labels words with the six basic emotions. We classify the trained documents into positive or negative.

Table 6 shows comparison experimental results. Except for the accuracy of GEM-CW model, all results are from the reference (Li et al., 2010). The performance of our model is the best, whatever the sentiment prior Sentiment-LDA used. This is because our model takes more reliable emotional prior knowledge automatically into sampling process and uses more elegant emotion analysis.

C. Comparison with CNN

Deep learning models based on CNN, RNN, BERT, etc, are also widely applied to classify sentiment. However, the deep learning models require large amount of train data, and in the task of stock

Table 5
Classifying Results Using Different Feature Selection Methods.

Feature Selection	Information Gain	Mutual Information	LDA(k = 7)	ESM	GEM-CW
Accuracy	72.1%	68.5%	73.8%	76.7%	78.7%

Table 6
Comparison Result with Sentiment-LDA.

Sentiment Prior	SentimentLDA	Supervised Classification	GEM-CW
HowNet	62.1%	Best Accuracy: 70.7%	
WordNet Affect	60.0%		74.6%
MPQA	65.3%		

market message sentiment classification, the train data with sentiment labels is not enough. In the experiment, we apply CNN with filter sizes as "3,4,5", the number of filters per filter size as 128, and the dimensionality of character embedding as 128. We achieve 72.6% accuracy, which is higher than that of traditional feature selection methods, but lower than that of GEM-CW, and LDA. In fact, the main goal of employing semi-supervised learning in the paper is to address the problem of the limited train data.

7. Conclusions and future work

As social media analysis draws much attention, sentiment classification of stock messages has become a hot research topic. Semi-supervised learning method is introduced to overcome the problems raised by short messages. To achieve this goal, the generative model GEM-CW is proposed to infer sentiment features from train set and test set. Finally, the features can be taken as the input to various classifiers. The novel design lies in that words are categorized according to their emotion weights so that it can provide fine-grained representation for sentiment in messages. Experiment results show that the performance of the proposed method is superior than that of tradition learning method, especially when the train set is large enough.

In addition, machine learning is popular in intelligent expert systems where processing of short messages is inevitable. Hence, the proposed method in the paper can be of valuable reference in expert systems. The generative models serve as a significant framework for the implementation of semi-supervised learning on text data. The theoretic aspect to support the inference of the parameters in the probability distribution is also useful for similar generative model design. Finally, the expert systems can borrow the idea to incorporate several kinds of interested information, as represented in the variable dependence in the GEM-CW model.

Although the semi-supervised learning model is significant and performs well, there are still several directions which are worth of further investigation.

1) For a given test set, the semi-supervised learning method might split the set and classify one by one depending on the size of train set. Thus, the efficiency of the classification is affected. In future research, how to make full use of the previously trained models in semi-supervised learning might be an important work.
2) One of the pivotal influence factors in *GEM-CW* is prior emotion words. The number of original emotion words in WordNet Affect is small and how to extend the emotional words on unique ad hoc domain makes difference. Synonym and relevance extending is feasible way, but not the best.
3) The other problem is that sarcastic sentences with or without sentiment words are hard to deal with. And many sentences without sentiment words can also imply sentiments. Hence, the adaptive prior knowledge extension methods to improve the performance can also be the future work.

CRediT authorship contribution statement

Jiangjiao Duan: Funding acquisition, Methodology, Investigation, Validation, Conceptualization. **Banghui Luo:** Formal analysis, Methodology, Visualization. **Jianping Zeng:** Supervision, Project administration, Writing - review & editing.

Acknowledgements

The paper is supported by the Ministry of Education Humanities and Social Science Foundation of China (Grant No. 13YJAZH019).

References

Bartov, E., Faurel, L., & Mohanram, P. S. (2017). Can Twitter help predict firm-level earnings and stock returns?. *The Accounting Review, 93*(3), 25–57.
Bianchini, M., Belahcen, A., & Scarselli, F. (2016). *A comparative study of inductive and transductive learning with feedforward neural networks[C]// Conference of the Italian Association for Artificial Intelligence*. Springer International Publishing.
Blei, D. M., Andrew, Y. N., & Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research, 3*, 993–1022.
Ekman, P., Sorenson, E. R., & Friesen, W. V. (1969). Pan-cultural elements in facial displays of emotion. *Science, 164*(4), 86–88.
Hasan A. A., & Fong A. C. (2018). Sentiment analysis based fuzzy decision platform for the Saudi stock market. 2018 IEEE International Conference on Electro/Information Technology (EIT), 23–29.
Iwata T., Watanabe S., Yamada T., & Ueda N. (2009).Topic tracking model for analyzing consumer purchase behavior. In Proceedings of IJCAI,1427–1432.
Li Q., & Shah S. (2017). Learning stock market sentiment lexicon and sentiment-oriented word vector from StockTwits. In Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), 301–310.
Li F., Huang M., & Zhu X. (2010). Sentiment analysis with global topics and local dependency. In Proceedings of AAAI.
Lin C., & He Y. (2009). Joint sentiment/topic model for sentiment analysis. Proceedings of the 18th ACM conference on information and knowledge management, 375–384.
Luo, B. H., Zeng, J. P., & Duan, J. J. (2016). Emotion space model for classifying opinions in stock message board. *Expert Systems with Applications, 44*, 138–146.
Ming F., Wong F., Liu Z., & Chiang M. (2014). Stock market prediction from WSJ: text mining via sparse matrix factorization. IEEE International Conference on Data Mining.
Moghaddam S., & Ester M. (2011). ILDA: Interdependent LDA model for learning latent aspects and their ratings from online product reviews. In Proceedings of the Annual ACM SIGIR International conference on Research and Development in Information Retrieval (SIGIR-2011).
Mukherjee A., & Liu B. (2012). Aspect extraction through semi-supervised modeling. In Proceedings of ACL, 339–348.
Nardo, M., Petracco-Giudici, M., & Naltsidis, M. (2016). Walking down Wall Street with a tablet: A survey of stock market predictions using the Web. *Journal of Economic Surveys, 30*(2), 356–369.
Nguyen, T. H., Shirai, K., & Velcin, J. (2015). Sentiment analysis on social media for stock movement prediction. *Expert Systems with Applications, 42*(24), 9603–9611.
Oliveira, N., Cortez, P., & Areal, N. (2016). Stock market sentiment lexicon acquisition using microblogging data and statistical measures. *Decision Support Systems, 85*, 62–73.
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 Systems with Applications, 73*(1), 125–144.
Pinheiro L., & Dras M. (2017). Stock market prediction with deep learning: A character-based neural language model for event-based trading. In Proceedings of Australasian Language Technology Association Workshop, 6–15.
Rajput V. S., & Dubey S. M. (2016). Stock market sentiment analysis based on machine learning. In Proceedings of 2nd International Conference on Next Generation Computing Technologies (NGCT).
Pergola, Gabriele, Gui, Lin, & He, Yulan (2019). TDAM: a Topic-Dependent Attention Model for Sentiment Analysis. *Information Processing & Management, 56*(6) 102084.

Renault, T. (2017). Intraday online investor sentiment and return patterns in the U. S. stock market. *Journal of Banking & Finance, 84*, 25–40.

Ruiz E. J., Hristidis V., Castillo C., & Gionis A. (2012). Correlating financial time series with micro-blogging activity. In Proceedings of Web searchand data mining, 513-522(WSDM-2012).

Sun, A., Lachanski, M., & Fabozzi, F. J. (2016). Trade the tweet: Social media text mining and sparse matrix factorization for stock market prediction. *International Review of Financial Analysis, 48*, 272–281.

Tumarkin, R. (2002). Internet message board activity and market efficiency: A case study of the Internet service sector using ragingbull.com. *Financial Markets, Institutions and Instruments, 11*(4), 313–335.

Volkova S., Wilson T. & Yarowsky D. (2013). Exploring demographic language variations to improve multilingual sentiment analysis in social media. In Proceedings of EMNLP, 1815–1827.

Wang, P., Xu, B., Xu, Xu. J. M., et al. (2016). Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. *Neurocomputing, 2016*(174), 806–814.

Yang C., Lin K. H., & Chen H.H. (2007). Emotion classification using web blog corpora. In WI '07: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, 275–278

Zhao B., He Y., Yuan C., & Huang Y. (2016). Stock market prediction exploiting microblog sentiment analysis. International Joint Conference on Neural Networks (IJCNN).