Expert Systems With Applications 118 (2019) 411–424

Contents lists available at ScienceDirect

Expert Systems With Applications

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

A new graphic kernel method of stock price trend prediction based on financial news semantic and structural similarity

Wen Long[a,b,c], Linqiu Song[a,b,c], Yingjie Tian[a,b,c,*]

[a] School of Economics & Management, University of Chinese Academy of Sciences, No. 80 of Zhongguancun East Street, Haidian District, Beijing 100190, PR China

[b] Research Center on Fictitious Economy & Data Science, Chinese Academy of Sciences, No. 80 of Zhongguancun East Street, Haidian District, Beijing 100190, PR China

[c] Key Laboratory of Big Data Mining & Knowledge Management, Chinese Academy of Sciences, No. 80 of Zhongguancun East Street, Haidian District, Beijing 100190, PR China

ARTICLE INFO

Article history:
Received 8 April 2018
Revised 31 August 2018
Accepted 3 October 2018
Available online 4 October 2018

Keywords:
Stock price movement prediction
Financial news
Information structure
S&S kernel

ABSTRACT

Lots of researches try to predict the stock price movement using financial news based on machine learning represented by SVM (Support Vector Machine). But almost all of them focus on the news contents while very few consider the information hiding in the relationship between different news. In this paper, we proposed a new kernel based on SVM concerning not only the contents themselves but also the information structures among them. As both the news contents and the information structures are imported into our kernel, this kernel is named as semantic and structural kernel, referred to S&S kernel. Medical industry financial news is used to illustrate the efficiency of our kernel. By comparing the predicting accuracy of S&S kernel with other kernels, such as linear kernel, we find our method outperforms the others by at least 5% on accuracy, which is a quite meaningful promotion. The result also confirms the information structure contained in daily financial news can offer extra information helping to predict the trend of stock price.

1. Introduction

Support vector machine (short as SVM) is now widely used in solving classification problems. The data in raw representation are transformed into feature vector representations using kernel functions, which enable users to operate in a high-dimensional, implicit feature space without ever computing the coordinates of the data in the original space. As the convenience of implementation, the application areas of kernel methods are diverse, including 3D reconstruction, bioinformatics, cheminformatics, information extraction, handwriting recognition and so on. Meanwhile, in order to adapt different data structure, derivatives of kernels are developed, for example, string kernels allow SVM and any other algorithms to work with strings without having to translate these to fixed-length, real-valued feature vectors, and graph kernels allow SVM to work directly with graphs.

One of the popular application scenarios is predicting stock price movement through financial news, which is a hot topic in recent years. Many factors affect the stock price trend, such as market conditions, inflation, trading strategies, return on net assets and so on. Lots of studies indicate that financial news have some influence on stock price. Dyck and Zingales (2003) found that the earnings announcement published by the news media would lead to greater volatility in the stock price. Especially when less analysts pay attention to the announcement and the media is very reliable, the price fluctuations caused by the announcement are particularly evident. Shiller (2005) argued that the media played a role in fueling the rise or fall of the stock market. The cognitive bias can create an “overreaction" to good news and bad news. Therefore, how to extract the information affecting the stock price trend from media coverage to better predict the market trends becomes very meaningful.

When studying this problem, we find there is few researches considering that the relationship between financial news may influence the stock price prediction. Currently, most researchers combine SVM with different news materials and different kernels, and usually consider each piece of news as a single input, regardless of the correlation between each two of them. In order to better grasp this feature, we construct a new kernel based on SVM to embed the correlativity between these inputs in the similarity calculation. We believe extra information can help with the prediction of stock price movement trends, and this new kernel can extend to other application scenarios like weather prediction, bank credit

* Corresponding author.
E-mail addresses: longwen@ucas.ac.cn (W. Long), tyj@ucas.ac.cn (Y. Tian).

https://doi.org/10.1016/j.eswa.2018.10.008
0957-4174/

rating prediction and so on. Therefore, the proposition of a new kernel containing both semantic and structural information among inputs is the main contribution of this paper, and its practical application in forecasting the stock market trend with financial news confirms its effectiveness.

2. Literature review

Since SVM model was invented, it has made great progress and been widely applied in various fields. With the development of SVM model, a lot of extensions are created. Multiclass SVM aims to assign more than 2 labels to instances by using support vector machines. Duan and Keerthi (2005) mentioned to reduce the single multiclass problem into multiple binary classification problems, which is also the most common way. Crammer and Singer (2001) propose to separate a multiclass classification problem into a single optimization problem, rather than decomposing it into multiple binary classification problems. Another important extension of SVM should be U-SVM. This model was first proposed by Vapnik (1998) and introduced into SVM to introduce unlabeled samples in the sample points of the two classifications. Weston, Collobert, and Sinz (2006) conduct experiments on Universum and SVM. The accuracy of the experimental results was greatly improved. Liu, Hsaio, and Lee (2017) combine the U-SVM model with semi-supervised learning and used this method to improve the accuracy of text classification. There is also a certain increase in the accuracy of the classification.

Though the variations of SVM have provided possibility to explore different applications, the innovation of kernels offers greater flexibilities. Lodhi, Saunders, Shawe-Taylor, Cristianini, and Watkins (2002) propose a new kernel for categorizing text documents without having to translate them to fixed-length, real-valued feature vectors. Pfoh, Schneider, and Eckert (2013) use a string kernel to make better use of the sequential information inherent in a system call trace to detect malwares. Vishwanathan, Schraudolph, Kondor, and Borgwardt (2010) discuss the concept called graph kernels, a kernel function that computed an inner product on graphs. Random walk kernel is one of the most intuitional examples. Zeng, Yi, and Liu (2010) apply spectral graph theory and random walk kernel on image denoising. Li, Su, and Wang (2012) capture consumer social influence similarities into a graph random walk kernel and build SVR models to predict consumer opinions.

We also concern the graphical representation of text. At present, the common text representation methods include Boolean model, vector space model, probability model and so on. These models solve the problem of processing text in the computer, but they do not consider the contextual structure of the text and the context order of the feature word. And the loss of this structure information will affect the full expression of the text content. Therefore, in recent years there are many scholars engaged in text representation of the study.

Lu (1990) propose a conceptual representation method applying to the field of information retrieval. Russell and Norvig (1995) put forward the semantic web representation of knowledge in his paper. Mani and Bloedorn (1997) present an algorithm using graph model to extract multi-document abstract. The idea is to add the semantic information of the text to the model. And then Choudhary and Bhattacharyya (2002) propose a method to construct the document feature vectors according to the UNL diagram. These methods have solved the problem of inadequately expressing text structure and relation to a certain extent, but these models are too complicated, and methods measuring the similarity of the models are still deficient.

Schenker (2003) propose the application structure of the text representation for the first time in his doctoral thesis. Schenker's diagram structure model is based on the HTML file. But the edge of the weight information is not considered. Jin and Srihari (2007) propose a model based on graphs to capture word order, frequency, word co-occurrence, and word meaning. This model is then applied to discover unrelated words in the text library. Wang and Liu (2010) propose a Chinese text categorization method based on graph model. They first use a weighting method to select the relevant feature building graph, then improve the text representation model and design a learning algorithm to classify Chinese text through the graphs. Bronselaer and Pasi (2013) propose a new graph-based model, by which a decomposition of textual documents is obtained where tokens are automatically parsed and attached to either a vertex or an edge. Kamaruddin et al. (2015) describe a text mining system that is able to detect sentence deviations from a collection of financial documents. The sentences are represented as graphs through a dissimilarity function. Yazdani, Murad, Sharef, Singh, and Latiff (2016) explore several types of feature spaces for sentiment classification of the news article and propose a method combining unigram and bigram. Experiments show that feature selection and feature weighting methods have a substantial role in sentiment classification. Das, Mehta, and Subramaniam (2017) propose AnnoFin, a new method, to help classify financial texts into different categories and a high accuracy of 73.56% is achieved even when the training data is just 30% of the total data set. Zhu and Iglesias (2018) exploit different semantic similarity methods based on various semantic resources, and the experimental results have shown this method is more effective than text similarity methods when contextual information is rare.

As for the application area of SVM and kernel methods, this paper focuses on financial issues. In the financial study of predicting stock price movement through financial text, most researchers tend to apply some machine learning methods on financial news, stock comments of some social platform as well as corporate financial announcements. Different materials are concerned when using text mining methods. Yu, Duan, and Cao (2013) use a dataset contains daily media features of 824 public traded firms across 6 industries. Ming, Wong, Liu, and Chiang (2015) uses daily articles from The Wall Street Journal to predict the closing stock prices. Sun, Lachanski, and Fabozzi (2016) investigate the textual information from user-generated microblogs to predict the stock market.

Meanwhile, text mining approaches are drawn into stock price prediction problem. Schumaker and Chen (2006) create the Arizona Financial Text System (AZFinText) by a synthesis of linguistic, financial and statistical techniques to predict the possibility of discrete stock price prediction. Tang, Yang, and Zhou (2009) combine news mining and time series analysis to forecast inter-day stock prices. Li and Wu (2010) studies online forums hotspot detection and forecast using unsupervised text mining approach combing K-means clustering with SVM. Huang and Zuo (2015) grab data from Sina Blog and process them into an emotional tendency time series. By adding Blog sentiment into their SVM predicting model, the predicting accuracy becomes higher. Tsai and Wang (2016) use a finance-specific sentiment lexicon to examine the relations between financial sentiment words and financial risk. The experimental results show that using only financial sentiment words results in performance comparable to using the whole texts. Long, Tang, and Tian (2016) apply U-SVM model to measure investor sentiment, and stock price fluctuations are determined by investors' sentiment measurement. By dividing stocks comments into three categories which are positive, negative and neutral, the neutral comments are included into this model as unlabeled samples. Seng and Yang (2017) design an algorithm calculating the sentiment orientation and score of data with added information. The results are then integrated for calculating stock market volatility. Jiang, Wang, Lan, and Wu (2017) propose an effective neural

network architecture GABi-LSTM to address fine-grained financial target-dependent sentiment analysis from financial microblogs and news, and it achieves the state-of-the-art performance. Kelly and Ahmad (2018) present a method analyzing the content of news using multiple dictionaries that accounts for the specific use of terminology in a given domain.

Through the existing researches, we can see almost all the scholars focus on refining their models to improve the predicting accuracy based on news contents only. Very few of them concern both contents and the relationship between the financial news when they study the impact of financial news on stock markets. In this paper, the information structure among texts is extracted and represented by graphs. In order to process content and structure simultaneously, we propose a new kernel, which is embedded into SVM model as the precomputed kernel and the model is used to forecast. In our experiment, as the kernel concerning both semantic and structural information, we name it as S&S kernel. Besides, the kernel we proposed is not limited to process news or other documents data. Any data with graph or network structure could apply this kernel to obtain more information.

The rest of this paper is arranged as follows: The construction of our kernel is proposed in Section 3. Each step of deriving our kernel based on SVM is explained in detail in this section. An empirical test is illustrated in Section 4, where we use financial news of a stock to show the effectiveness of S&S kernel compared with the other kernels. In Section 5, we add another 3 stocks to test the robustness of our model. The last section of this paper is the conclusion.

3. Method

SVM (Support Vector Machines) is first proposed by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963. Later Boser, Guyon, and Vapnik (1992) suggested a way to create nonlinear classifiers by applying the kernel trick to maximum-margin hyperplanes. Simply speaking, SVM is a tool used for classification and regression. Given a set of training examples, each assigned to one of the two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier.

Given a training dataset of n points of the form $(x_1,y_1),(x_2,y_2),\cdots\cdots,(x_n,y_n)$, where the y_i are either 1 or -1, each indicating the class to which the point x_i belongs. Each x_i is a p-dimensional real vector. The "maximum-margin hyperplane" is defined to divide the group of points so that the distance between the hyperplane and the nearest point x_i from either group is maximized. A linear hyperplane is

$$g(x) = w^T x + b, \tag{1}$$

which means when $w^T x + b > 0$, the output should be 1 and when $w^T x + b < 0$, the output should be -1. The goal is to maximize the distance between the points of each side to the hyperplane and keep all points out of the margin. With some mathematical processing, the problem is

$$\min \tfrac{1}{2} w^T w \text{ s.t. } y_i\left(w^T x_i + b\right) \geq 1, \forall x_i. \tag{2}$$

In order to take the outliers into consideration, a slack variable is added into the model and the model becomes to maximize the soft margin:

$$\begin{aligned} &\min \frac{1}{2} w^T w + C \sum_{i=1}^{N} \xi_i \\ &\text{s.t. } y_i\left(w^T x_i + b\right) + \xi_i \geq 1, \forall 1 \leq i \leq N \\ &\xi_i \geq 0, \forall 1 \leq i \leq N \end{aligned} \tag{3}$$

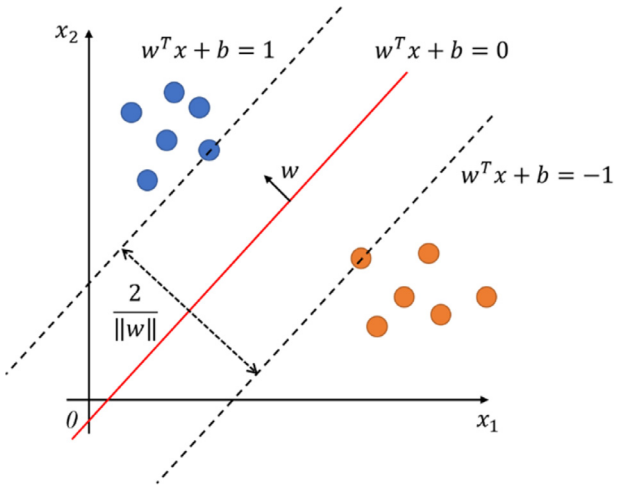


Fig. 1. A 2-dimensional example of using SVM in classification.

The original SVM performs linear classification, however, there are some data are linearly non-separable. Bernhard E. Boser, Isabelle M. Guyon and Vladimir N. Vapnik suggested a way to create nonlinear classifiers by applying the kernel trick to maximum-margin hyperplanes in 1992. With the kernel trick, the inputs are implicitly mapped into a high-dimensional feature space, which can be divided by a hyperplane. The model is as follows:

$$\begin{aligned} &\min \frac{1}{2} w^T w + C \sum_{i=1}^{N} \xi_i \\ &\text{s.t. } y_i\left(w^T \phi(x_i) + b\right) + \xi_i \geq 1, \forall 1 \leq i \leq N \\ &\xi_i \geq 0, \forall 1 \leq i \leq N \end{aligned} \tag{4}$$

In this optimization problem, the Lagrangian duality is often used to convert the original problem into a dual problem, and the solution to the original problem is obtained by solving the dual problem:

$$\begin{aligned} \max W(\alpha) &= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \phi(x_i) \phi\left(x_j\right) \\ &= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j k\left(x_i, x_j\right) \\ &\text{s.t. } \sum_{i=1}^{n} \alpha_i y_i = 0 \\ &0 \leq \alpha_i \leq C \end{aligned} \tag{5}$$

In order to better understand how SVM used in classification, a 2-dimensional example is illustrated in Fig. 1.

The two categories are classified by the colors. The linear hyperplane here corresponds to the red line in the middle. The maximum-margin hyperplane of SVM is trained with samples from two classes.

The commonly used kernels include Polynomial kernel, Gaussian radial basis function (short as RBF) and Hyperbolic tangent kernel. These kernels have been applied to solve many classification problems. The input, however, is required to be in a vector form. For example, when using SVM to classify the sentiment of financial news, only the contents of the news are concerned. In order to consider the relationship between each two news, we need to add the structural information into the algorithm (see Fig. 2). So based on SVM, we propose S&S kernel, which concerns both semantics and structures. What should be noted is the kernel we propose here is not specific for stock price prediction via financial news. It can be applied to any other text data as long as there is a relationship between each pair of them.

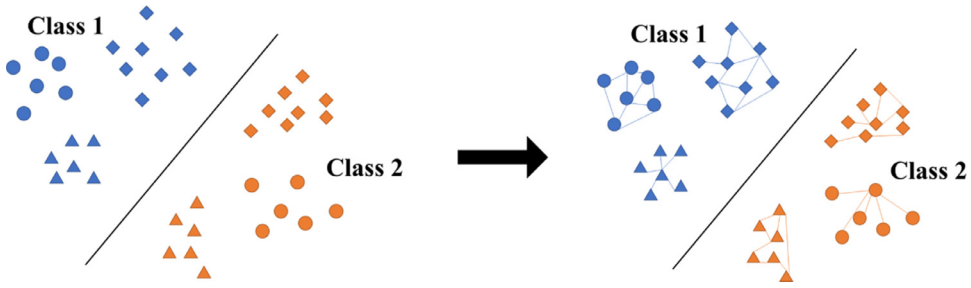


Fig. 2. Standard SVM classification and the new proposed SVM classification.

3.1. The construction of input structure

The construction of S&S kernel is a process of building multiple kernels into one. Basically, the process can be separated into two parts: semantic part and structural part. Visually, we can regard the structural part as graphs as each piece of news or each key word are taken as vertices and the edges are calculated based on the vertices. The features are extracted from news contents and are used as vertices, then the similarity among them is calculated. This part is defined as the semantic kernel part. Our main innovation is the structural kernel part. To define the structural kernel, we need construct the edges between the vertices first.

In order to represent the relationship comprehensively, we develop two types of graph structures: text points graph and key words graph. One of the differences between these two graphs is the definition of the vertices: text points graph uses each piece of text as vertices while key words graph uses key words. The key words here can be obtained manually or automatically through algorithms like bag-of-words or any other text to vectors methods.

3.1.1. Text points graph

In text points graph, we regard each piece of text as vertices. As for the edge, the key words co-occurrence of each pair of texts are used as the weight. In detail, for each pair of texts (texts s and text t), we pick the key words appear in both texts and sum up the minimum frequency of each key words as the weights between the pair of texts:

$$Freq_{s,t} = \sum_i \min\left(n_s^i, n_t^i\right) \tag{6}$$

For example, in stock price prediction via financial news, we take each financial news as vertices. The key words are manually defined by expert knowledge. The frequency of each key word i in each financial news is denoted as n^i and the weight between the pair of *news s* and *news t* is the sum of the minimum co-occurrence which is $\min(n_s^i, n_t^i)$. The text points graphs represent the connection among all the texts and the dimension of the graphs should be equal to the number of texts.

3.1.2. Key words graph

We use the key words information to supplement the text points graph, especially when there is only one piece of text under certain conditions, which would lead to the construction of text point graph becoming impossible. Extracted key words here are used as vertices. The key words can be obtained from expert knowledge or can be generated from certain algorithms. For each pair of key words Key_p and Key_q, we use the sum of the minimum co-occurrence of the words in each pair of text as the weight between the pair of key words:

$$Freq_{p,q} = \sum_i \min\left(n_p^i, n_q^i\right) \tag{7}$$

For example, in stock price prediction via financial news, the frequencies of key word p and key word q in each piece of financial news are counted first, denoted as n_p^i and n_q^i respectively. The minimum co-occurrence of these two words in each financial news is calculated, $\min(n_p^i, n_q^i)$. Then the frequency of each two words is the sum of the minimum co-occurrence through all the financial news. The key words graph is constructed and the dimension of these graphs should be equal to the number of the key words.

3.2. The definition of S&S kernel

Based on the graph construction, we can define kernels on each part of the graph. According to the segmentation between semantics and structures, the measurement is also divided into two parts: the similarity among the contents features and the similarity among the structures.

For the similarity between each two pieces of texts, which is equivalent to the similarity among the feature vectors of each two texts, we calculate the 2-norm distance between them and go through all pairs of the texts. Then $e^{-\text{average}(\Sigma distance)}$ is used to define the similarity between these two graphs. What should be noted here is that there may be two pieces of texts having nothing in common, meaning the distance between them cannot be calculated. Considering such situation, we define the similarity to be zero. So, for *text s* and *text t*, the contents similarity between them is:

$$k(s,t) = \begin{cases} \exp\left(-average\left(\sum_q \sum_p \|data_p - data_q\|\right)\right) \\ \qquad if\ s \neq t\ and \sum_q \sum_p \|data_p - data_q\| \neq 0 \\ 0, \qquad if\ s \neq t\ and \sum_q \sum_p \|data_p - data_q\| = 0 \\ 1, \qquad otherwise \end{cases} \tag{8}$$

where $data_p$ denotes the contents features of *text s* and $data_q$ denotes the contents features of *text t*.

When calculating the similarity among the structures, we also need to calculate the distance in two parts based on our construction separately. For text points graphs, we need to deal with the unequal dimension problem as the dimensions of these graphs equal to the numbers of texts each graph contains. Here we apply PCA to take the principal components of each piece of data first. Then the lower dimension of each pair of graphs is applied as the number of principal component we take. For example, G_s^1 and G_t^2 $(s \leq t)$ stand for the two graphs respectively, where s and t are the number of vertices in each graph correspondingly. For G_s^1 and G_t^2, we take their first s principal components as their feature vectors and calculate the 2-norm distance traversing all pairs of vectors. The average of all vector distances is taken as the distance between G_s^1 and G_t^2. Then $e^{-\text{average}(\Sigma distance)}$ is used to define the

edge similarity between these two graphs into the algorithm.

$$y(s,t)=\begin{cases}\exp\left(-average(\sum_{j=i+1}^{m}\sum_{i=1}^{n}\left\|data_{s,i}-data_{t,j}\right\|\right), \\ \qquad if\ s\neq t\ and\ \sum_{j=i+1}^{m}\sum_{i=1}^{n}\left\|data_{s,i}-data_{t,j}\right\|\neq 0 \\ 0, \qquad if\ s\neq t\ and\ \sum_{j=i+1}^{m}\sum_{i=1}^{n}\left\|data_{s,i}-data_{t,j}\right\|=0 \\ 1, \qquad otherwise\end{cases} \tag{9}$$

For key words graphs, the calculation is much simpler as the dimension of each graph is all the same, which is the number of key words. We traverse the 2-norm distance between the corresponding key words of two graphs and use the average of all the distances as the distance of the two days' graphs. Again, we use $e^{-\text{average}(\Sigma distance)}$ to define the edge similarity between these two graphs.

$$h(s,t)=\begin{cases}\exp\left(-average(\sum_{i=1}^{p}\left\|Key_i^{G_s}-Key_i^{G_t}\right\|\right), \\ \qquad if\ s\neq t\ and\ \sum_{i=1}^{p}\left\|Key_i^{G_s}-Key_i^{G_t}\right\|\neq 0 \\ 0, \qquad if\ s\neq t\ and\ \sum_{i=1}^{p}\left\|Key_i^{G_s}-Key_i^{G_t}\right\|=0 \\ 1, \qquad Otherwise\end{cases} \tag{10}$$

To denote the structural similarity, we use a hyper parameter α to combine $y(s,t)$ and $h(s,t)$ together.

$$E(s,t)=\ \alpha y(s,t)+(1-\alpha)h(s,t) \tag{11}$$

The parameter α is determined by Grid search.

To combine the contents and structures together, we use the same idea as we denote the edge similarity. Another hyper parameter λ is applied to combine matrix $E(s,t)$ and $k(s,t)$ together.

$$G(s,t)=\lambda E(s,t)+(1-\lambda)k(s,t) \tag{12}$$

Also, parameter λ is determined by Grid search. The kernel we proposed is based on the similarity G(s, t) we define above.

4. Experimental test based on financial news and stock price movement

As we have demonstrated in the literature review, financial news is tested to have certain effects on stock price and can be used to predict the stock price movement (Antweiler & Frank, 2004; Dyck & Zingales, 2003; etc.). Also, SVM has been widely used to predict the movement by classifying moving up or down. Based on the relationship between financial news and stock price movement, we now apply SVM with proposed S&S kernel to predict the stock price movement.

4.1. Data source and statistical features

The news text materials we use are all from the financial channel of ifeng.com, which is a well-known professional financial portal in China. Considering the popularity and the development of the industry these years, we choose medical and health industry as our object. The news is obtained by web crawler. We collect the financial news of stocks from September of 2012 to March of 2017. In order to demonstrate our model in detail, we take a random sample of stock SZ002424 to exhibit the results (see Fig. 3). More samples' results will be found in Section 5 and the appendix.

We aim to predict the trend of stock price based on financial news, so the return of a stock r_t within a trading day is processed into two signs:

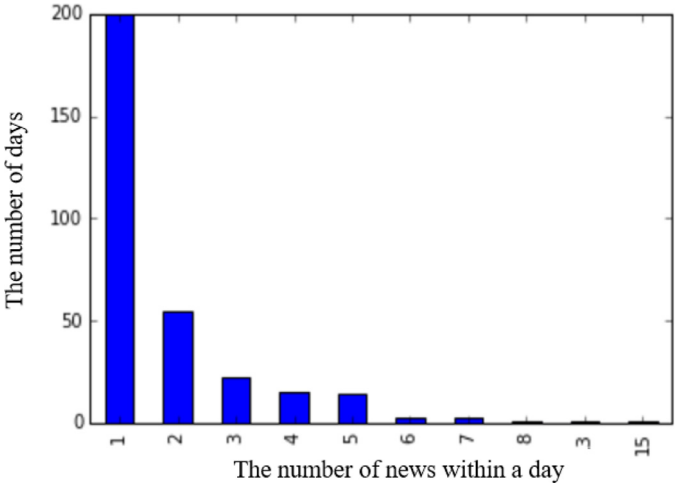


Fig. 3. The distribution of the news number for SZ002424.

$$tag_t=\begin{cases}1, & if\ r_t>0\\ 0, & if\ r_t\leq 0\end{cases} \tag{13}$$

4.2. Text preprocessing

As for text preprocessing, we choose bag-of-words model to extract the features out of financial news. The bag-of-words model is commonly used to process the raw text: the occurrence of each word is used as a feature for training a classifier. In this model, a text (such as a sentence or a document) is represented as the bag of its words, disregarding grammar and even word order but keeping multiplicity. This method has shown great efficiency and lead to relatively reasonable prediction accuracy. Though there are other text processing methods which may outperform bag-of-words, the point of our paper is the S&S kernel. Thus, bag-of-words model is applied to preprocess the original financial news here and the outputs of the model is used as features of each news.

4.3. Graph construction

Based on our primary thought, financial news within a day can be related to each other. The information behind the relationship has long been ignored. But the structural information may contribute a lot to our stock price prediction.

Before we construct the graph, we need to obtain the key words first so that we can take the frequency of them to build the graphs. We first list the key words for a specific industry. For example, in our experiment, the key words we listed for bio-medicine industry include *biology, medicine, health, nursing, intermediate* and so on. Besides, the abbreviations of the listed companies are also included in these key words. The reason is that the companies of the same sub-industry may share similar businesses or produce similar products, leading to a result that they are always mentioned together in the same news. Whereas a company usually run multiple businesses, so different news may have different emphasis and different companies may be mentioned in different news. In this paper, we focus on the medical industry and list 323 Chinese key words including both industrial terminology and stock abbreviations for the empirical test. Then the weight on each edge can be calculated and the graph can be built.

For text points graphs, we regard each news as a vertices. The key words graphs, which use each key word as a vertices, are then applied the to supplement the text points graphs, especially when there is only one piece of news within a day (This would cause

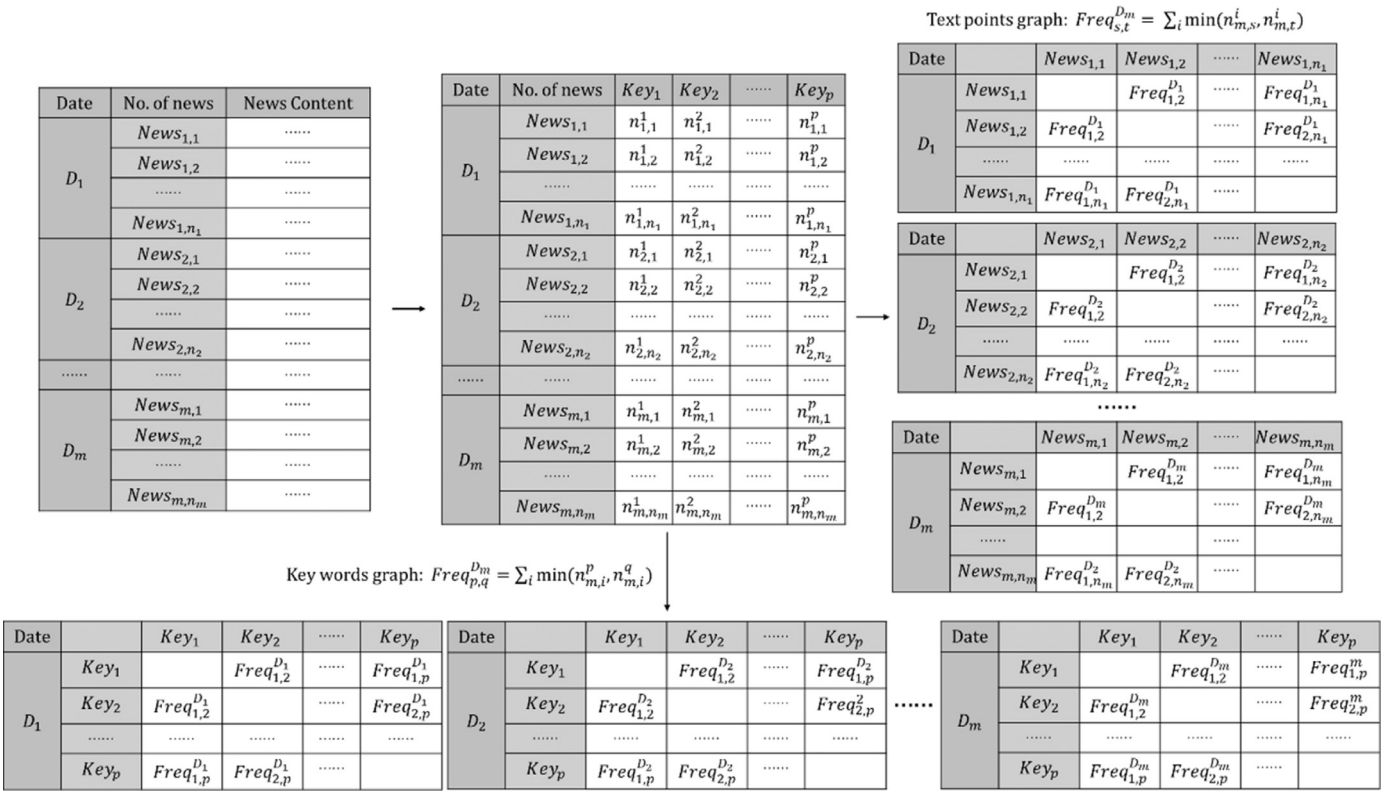


Fig. 4. Construction of news text points graph and key words graph.

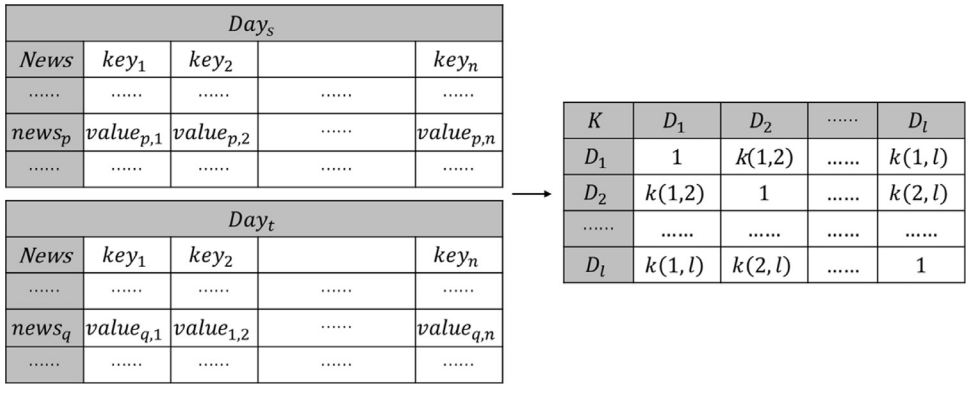


Fig. 5. Calculating the content similarity between the news features.

the construction of news graph becoming impossible). The graph construction process is shown in Fig. 4. In this paper, we divide the news construction by dates, meaning each graph corresponds to the financial news within a day.

4.4. Similarity measurement

After preprocessing the financial news as well as constructing daily graphs, we need to measure the similarity between each two graphs. The measurement is divided into two parts: the similarity between the news features that we gain from bag-of-words model, and the similarity between the daily graphs. The calculating process is shown in Figs. 5–7. We traverse all pairs of news in two days and use the average distance as the distance between the two days. So the distance matrix of news features can be acquired. Function k(s,t), y(s,t) and h(s,t) corresponds to function (5), (6) and (7) respectively.

As matrix Y and matrix H are used to denote the structure of the news, we use a hyper parameter α to combine them together.

$$E = \alpha Y + (1-\alpha)H \tag{14}$$

The parameter α is determined by Grid search. Also, hyper parameter λ is imported to combine matrix E and matrix K.

$$G = \lambda E + (1-\lambda)K \tag{15}$$

Also, parameter λ is determined by Grid search.

4.5. About the hyperparameters

As we construct our kernel using two hyperparameters: α, which is used to determine our daily news graph, and λ, which is used to determine the weight between the news content and the news graph within a day, grid search method is applied here to test different parameters. The step gap we use here is 0.1.

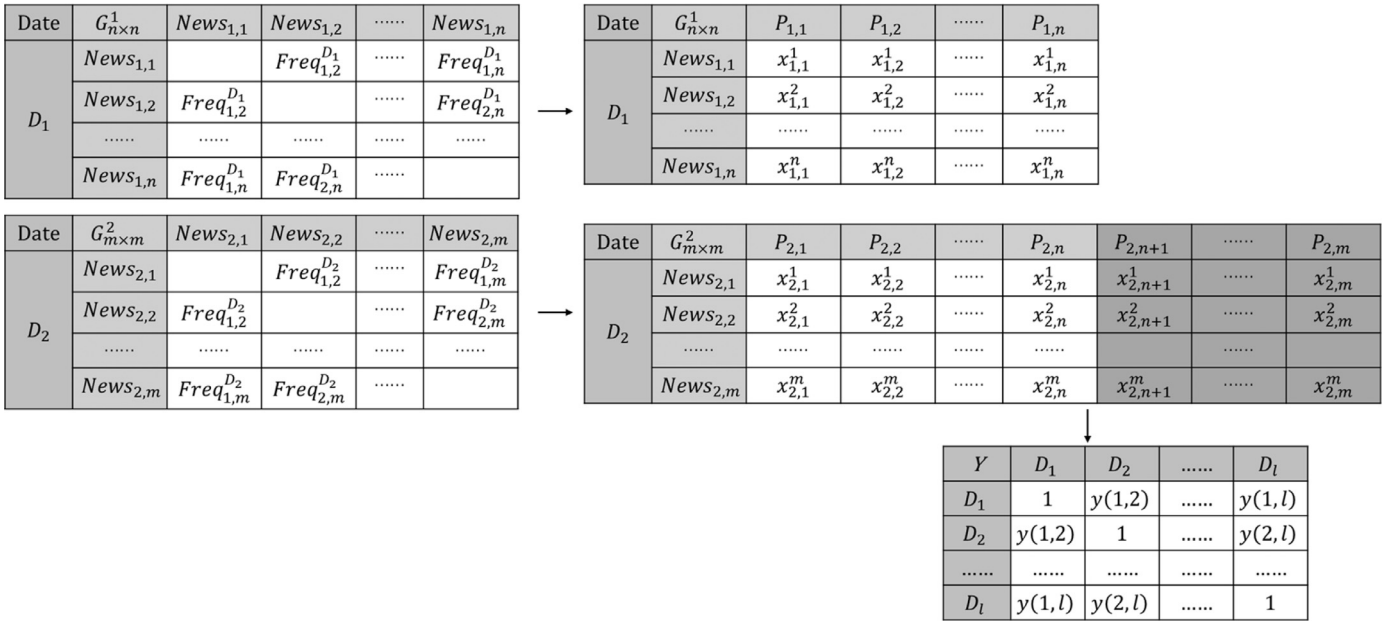


Fig. 6. Calculating the edge similarity between text points graphs.

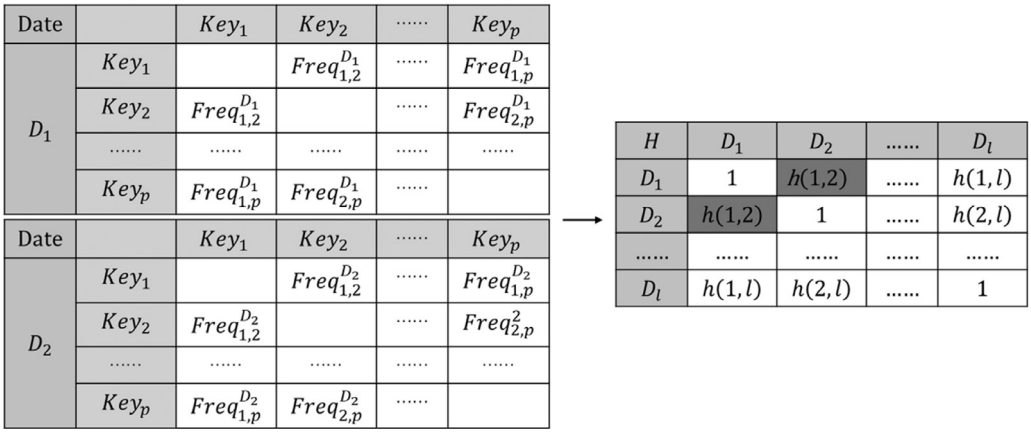


Fig. 7. Calculating the edge similarity between key words graphs.

5. Results and discussions

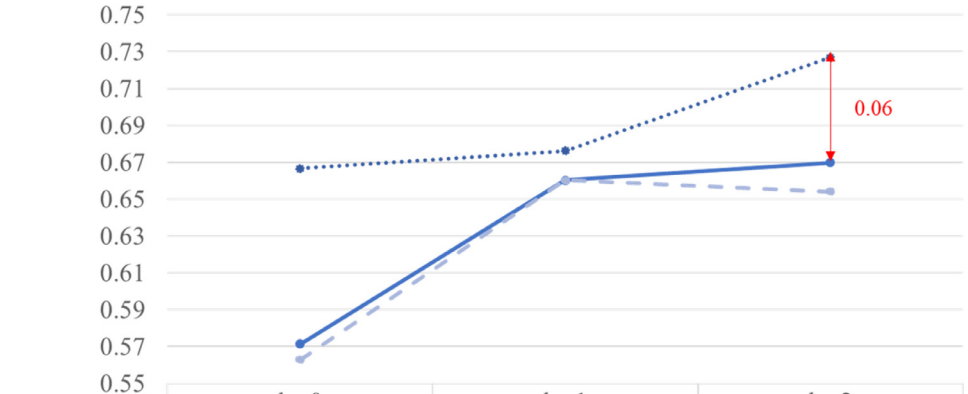
In this part, we analyze the prediction result of stock SZ002424 using SVM with S&S kernel. The prediction result of other socks can be found in Appendix.

First of all, we use the heatmap to test the influence of our hyper parameters α and λ. In Fig. 8, the horizontal axis represents α and the vertical axis represents λ. When we apply our method to predict the price movement with the news in the same day, which is shown in Fig. 8(a), we can see the color becomes darker with λ becomes greater. This means the greater the weight of the daily news edge is, the higher the predicting accuracy is. However, there is no clear gradual change along with the horizontal axis, meaning hyper parameter α may have less effect than λ. So, the weight assignment within the daily news graph construction may, somehow, provides similar information.

We also predict the price movement with 1 day to 3 days lag after the news, which is shown in Fig. 8(b)(c)(d). The same result can be found as the color becomes darker from top to bottom. Besides, the highest predicting accuracy is quite high and a 73% accuracy is achieved when we use our kernel to predict the stock price movement with 2 days lag. All these results strongly support our point of view that the relationship between the structures of daily news from day to day can provide us more information to predict the stock price movement, which can further help investors make right decisions.

But can our method provide higher accuracy compared with traditional SVM? Or the news contents have provided enough information to help predict the stock price movement? Here, we apply SVM with linear kernel, sigmoid kernel and radial basis function kernel (rbf kernel) to predict the stock price movement compared with our kernel. Considering the number of our sample, we apply a five-fold cross validation to gain the average predicting accuracy. The result is shown in Fig. 9. The accuracy of SMV with sigmoid kernel is the same as rbf kernel. We only show the result of rbf kernel here for simplicity. From the accuracy of price trend prediction, our kernel outperforms SVM on both rbf kernel and linear kernel. Fig. 9 is the predicting accuracy of the three different kernels of SVM with 0–2 days lag. We can see the kernel we propose always has higher accuracy than the other two kernels. The price movement predicting accuracy using news contents only is 57.14% with no day lag, while our kernel's accuracy is 66.66%, which is 9.52% higher than rbf kernel. Besides, our kernel can reach nearly 73% accuracy when predicting the price movement with 2 days lag.

Fig. 8. The heatmap of predicting accuracy using different hyper parameters
Note: The horizontal axis represents α and the vertical axis represents λ. Parameter α is the weight of text point graph and parameter λ is used to determine the weight between the news content and the news graph within a day. The color bar on the right is accuracy of prediction.



	lag0	lag1	lag2
our kernel	0.67	0.68	0.73
rbf	0.57	0.66	0.67
linear	0.56	0.66	0.65

Fig. 9. The predicting accuracy of three kernels with different lags.

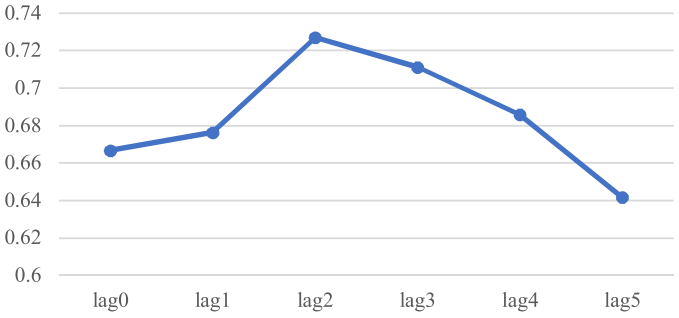


Fig. 10. The predicting accuracy along with time lags.

Table 1
The descriptive statistics of the testing stocks.

Stocks	Number of news	Days of news releasing	Max number of news per day
SZ002424	581	315	15
SZ300049	780	390	15
SZ300142	870	420	18
SZ000661	855	405	16

Based on this phenomenon, we propose a hypothesis that the financial news may have an effect on the stock price with time lag. In fact, sometimes a breaking favorable news can drive the stock price up for consecutive days. In order to test this effect, we use tag_{t+1} to tag_{t+5} separately to train our model. The period we choose here is 5 days as the trading days within a week is 5 days.

As there are 5 trading days within a week, we test our kernel with 1 day to 5 days lag and the result is shown in Fig. 10. There is a clear inverse U shape where the highest predicting accuracy is reached on 2 days lag while the accuracy of 5 days lag is the lowest. The experimental result can verify our hypothesis which means the financial market needs some time to digest the information from the news and this effect mainly lasts for two to three days after the news reported.

As the accuracy reaches the highest level when predicting the stock price 2 days after, we analyze the effect of our two hyper parameters again. We use the average predicting accuracy of all values of α when λ is given to represent the accuracy of the given λ. The same is done to each value of α. The result is shown in Fig. 11. For the average accuracy of different λ, we can see a clear upward trend, meaning the higher weight given to the graph we constructed, the higher accuracy can be acquired. The result here is as same as what we gain from the heat map. While for parameterα, there is a sudden high spot when $\alpha = 0.2$ and with α becoming greater (except $\alpha = 0.1$), the average accuracy moves downwards. Therefore, matrix H should be assigned a higher weight meaning the relationship between the key words structures contains more effective information.

6. Stability test

In order to further strengthen our point, we add another 3 stocks to test the universality of our kernel. These three stocks are stochastically chosen from our news data set, which are SZ300049, SZ300142 and SZ000661. A descriptive statistic of the 3 stocks as well as SZ002424 is shown in Table 1. The heatmaps of prediction accuracy of the three stocks are shown in the appendix.

We first use the daily news contents to predict the stock price on the same day, meaning we use day_t's news to predict the trend of day_t's stock price. Linear kernel, rbf kernel, sigmoid kernel and poly kernel are used as benchmark here. As we can see from Table 2, for these four stocks, our kernel outperforms the other four kernels. Our kernel can reach the highest accuracy for SZ300142, even using the other kernels have reached a 64.94% accuracy, which is quite high and also means the news really influence this stock's price trend. A clear result from this table is that the kernel we propose can always outperform the other kernels by 5%.

Table 2
The comparison of predicting accuracy different kernels.

	SZ 002424	SZ 300049	SZ 300142	SZ 000661
SVM-linear kernel	56.30%	53.85%	51.84%	49.00%
SVM-rbf kernel	57.14%	55.13%	64.94%	52.51%
SVM-sigmoid kernel	57.14%	55.13%	64.94%	52.51%
SVM-poly kernel	57.14%	55.13%	64.94%	52.51%
SVM- S&S kernel	**64.76%**	**58.73%**	**69.30%**	**59.01%**

Table 3
Predicting accuracy using five-fold cross validation.

	SZ 002424	SZ 300049	SZ 300142	SZ 000661
Lag 0 day	64.76%	58.73%	69.30%	59.01%
Lag 1 day	66.66%	61.28%	73.79%	65.67%
Lag 2 day	67.62%	73.34%	80.44%	71.61%
Lag 3 day	72.70%	73.33%	79.51%	71.11%

Table 4
Predicting accuracy using the earliest 80% to train and the rest to test.

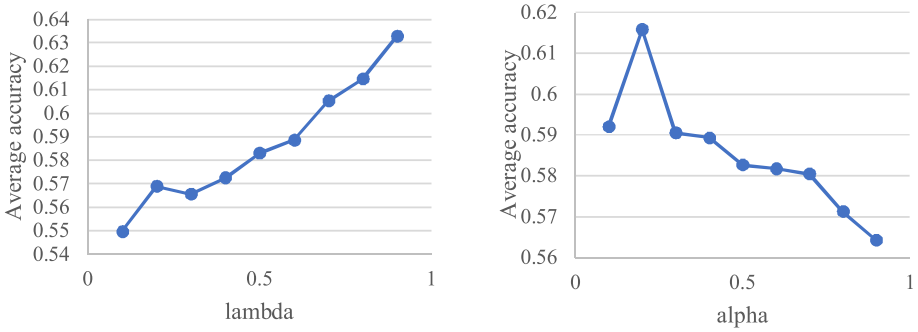
	SZ 002424	SZ 300049	SZ 300142	SZ 000661
Lag 0 day	66.67%	71.79%	83.33%	61.73%
Lag 1 day	71.43%	62.82%	83.33%	65.43%
Lag 2 day	71.43%	74.36%	90.48%	74.07%
Lag 3 day	79.37%	74.36%	89.29%	67.90%

We also need to test whether our kernel performs best when predicting the stock price with 3 days' lag as the result we gain from SZ 002424. We calculate the predicting accuracy of the 3 stocks price trend with 1 day, 2 days and 3 days lag, which means we use day_t's news to predict the trend of day_{t+1}, day_{t+2} *and* day_{t+3}'s stock price respectively. The results are summarized in Table 3. Different from SZ 002424, the accuracy of the three stocks all reaches highest with 2 days lag. Therefore, financial market needs some time to digest the information from the news and this effect is always reflected in the stock price with 2 to 3 days lag, which is consistent with the actual situation.

The time effect leads us thinking about that history information may contain some useful information for predicting future movement. So we use the earliest 80% of our news to train the model and use the rest 20% to test. The result is shown in Table 4. Compared with Table 3, we can see that the predicting accuracy improves except SZ 000661 with 1-day lag and 3-days lag. For SZ300142, the accuracy is quite astonishing and we think a deterministic promotion with some contingency factors can lead to the result.

7. Conclusion

In this paper, we propose a new kernel based on SVM, S&S kernel, to extract the correlation between inputs, and apply this new method on the problem of stock price trends prediction through financial news. This problem has drawn a lot of attention in these years along with the popularity of machine learning. But almost all the scholars focus on digging the information contained in the financial news contents. From our experiment, both the semantic and structural relevance between two days' news, which corresponding with two inputs, can help with the prediction of the stock price trend. Our empirical test shows that S&S kernel outperform the other four commonly used kernels by at least 5% on



(a) Average accuracy of different λ (b) Average accuracy of different α

Fig. 11. Average predicting accuracy with 2 days lag.

predicting accuracy. The improvement of predicting accuracy usually means that there is a higher probability of profitability if the stock is traded according to the signal sent by the model, which is very meaningful for investors. When we prolong the lag days of prediction, a clear inverse U shape can be found. It is helpful to show that financial news has an effect on stock price and this effect usually last for 2–3 days.

From the construction of our kernel, we can see that a higher weight is given to the graph we constructed instead of news content. This is a very novel finding as most researches have paid their attention to the news content only and tried to reach a higher accuracy through adjusting their model. The daily news graph we imported plays a more important role when predicting stock price, because the graph has offered some extra information on news structure. What else should be noticed here is that we only apply bag-of-words method and SVM here. Other models can be applied to process text and predict the stock price trend with importing the structural information in the future, and hopefully a better result can be acquired. Besides, historical information is concerned in the last part of our experiment. The experimental result helps verifying the information in the past, both contents and structures, can help with price prediction.

Although a creative kernel of SVM is proposed inspired by the problem of financial prediction in this paper, and has been verified effeteness in stock price trend prediction through financial news according to our experiment, S&S kernel can be widely extended to other areas such as the transmission of infectious disease and shopping recommendation for consumers, even the prediction of athletic contests can be suitable as the scenarios exist a graph structure. Any data with graph or network structure can apply this kernel to obtain more information. We hope this idea can be applied to more fields in the future.

Author contributions

Wen Long: Writing - review & editing; Software; Validation; Visualization.
Linqiu Song: Conceptualization; Roles/Writing - original draft.
Yingjie Tian: Investigation; Methodology; Funding acquisition.

Acknowledgements

This research was partly supported by National Natural Science Foundation of China (Nos. 71771204, 71731009, and 61472390).

Appendix. Heatmaps for stocks 300049.SZ, 300142.SZ and 000661.SZ

The heatmaps of predicting accuracy using different hyper parameters for stocks 300049.SZ, 300142.SZ and 000661.SZ are demonstrated in figure Figs. A.1–A.3.

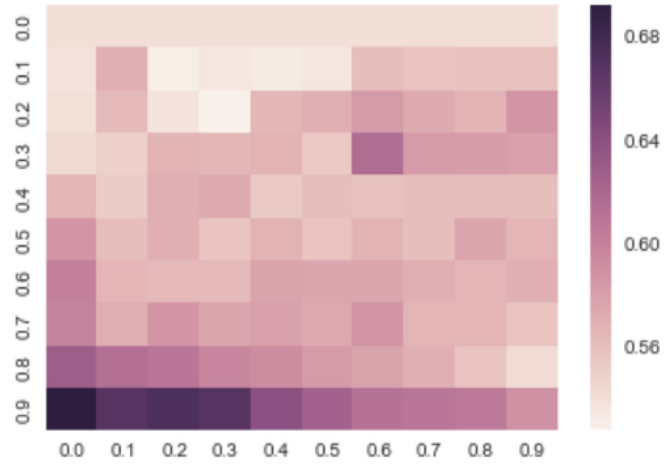
(a) Predicting the stock price of Day_0

(b) Predicting the stock price of Day_1

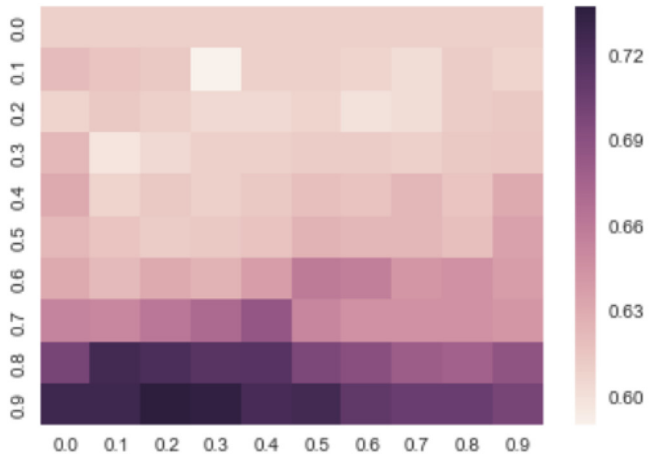
(c) Predicting the stock price of Day_2

(d) Predicting the stock price of Day_3

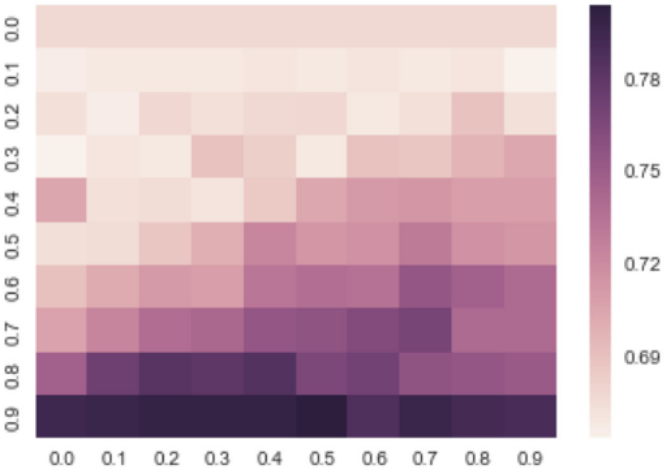
Fig. A.1. The heatmap of predicting accuracy using different hyper parameters for 300049.SZ.



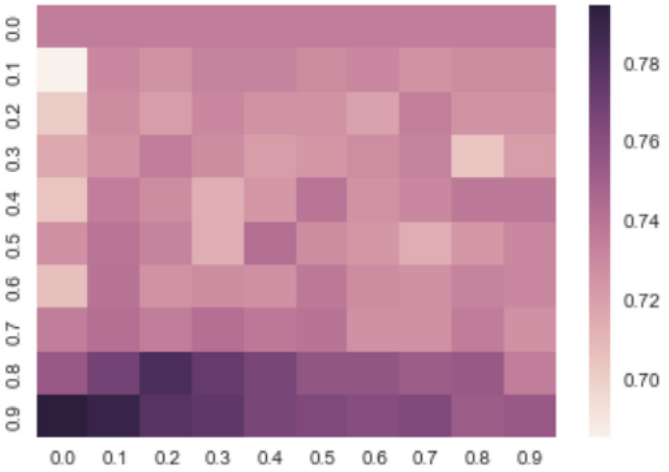
(a) Predicting the stock price of Day_0



(b) Predicting the stock price of Day_1



(c) Predicting the stock price of Day_2



(d) Predicting the stock price of Day_3

Fig. A.2. The heatmap of predicting accuracy using different hyper parameters for 300142.SZ.

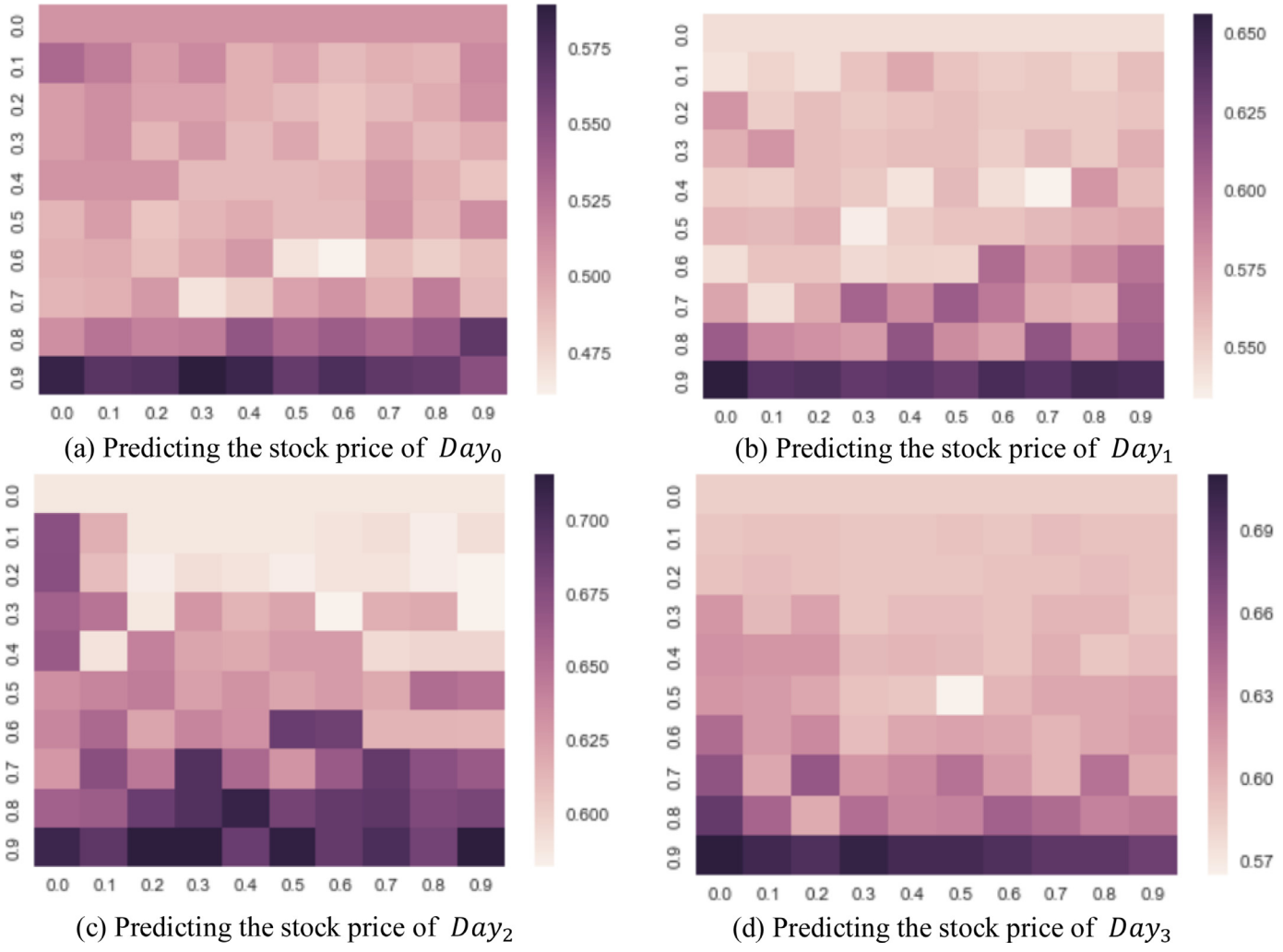


Fig. A.3. The heatmap of predicting accuracy using different hyper parameters for 000661.SZ.

References

Antweiler, W., & Frank, M. Z. (2004). Is all that talk just noise? The information content of internet stock message boards. *Journal of Finance, 59*(3), 1259–1294.

Bronselaer, A., & Pasi, G. (2013). In *An approach to graph-based analysis of textual documents: 32* (pp. 634–641). Atlantis Press.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *The workshop on computational learning theory: 5* (pp. 144–152).

Choudhary, B., & Bhattacharyya, P. (2002). Text clustering using universal networking language representation.

Crammer, K., & Singer, Y. (2001). On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research, 2*, 265–292.

Das, A. S., Mehta, S., & Subramaniam, L. V. (2017). Annofin–a hybrid algorithm to annotate financial text. *Expert Systems with Applications, 88*, 270–275.

Duan, K. B., & Keerthi, S. S. (2005). Which is the best multiclass SVM method? An empirical study. In *International workshop on multiple classifier systems: 3541* (pp. 278–285). Berlin, Heidelberg: Springer.

Dyck, A., & Zingales, L. (2003). In D. Ehrlich, I. Guttman, P. Schonback, & J. Mills (Eds.), *The media and asset prices*. Harvard Business School.

Huang, R. P., & Zuo, W. M. (2015). Predicting the stock market based on microblog mood. *Journal of Industrial Engineering & Engineering Management, 01* 47-52+215.

Jiang, M., Wang, J., Lan, M., & Wu, Y. (2017). An effective gated and attention-based neural network model for fine-grained financial target-dependent sentiment analysis. In *International Conference on Knowledge Science* (pp. 42–54).

Jin, W., & Srihari, R. K. (2007). Graph-based text representation and knowledge discovery. *ACM Symposium on Applied Computing*, 807–811 DBLP.

Kamaruddin, S. S., Bakar, A. A., Hamdan, A. R., Nor, F. M., Nazri, M. Z. A., Othman, Z. A., et al. (2015). A text mining system for deviation detection in financial documents. *Intelligent Data Analysis, 19*(s1), S19–S44.

Kelly, S., & Ahmad, K. (2018). Estimating the impact of domain-specific news sentiment on financial assets. *Knowledge-Based Systems, 155*, 116–126.

Li, N., & Wu, D. D. (2010). Using text mining and sentiment analysis for online forums hotspot detection and forecast. *Decision Support Systems, 48*(2), 354–368.

Li, X., Su, X., & Wang, M. Y. (2012). Social network-based recommendation: A graph random walk kernel approach. In *Proceedings of the ACM/IEEE joint conference on digital libraries.*. doi:10.1145/2232817.2232915.

Liu, C. L., Hsaio, W. H., Lee, C. H., Chang, T. H., & Kuo, T. H. (2017). Semi-supervised text classification with universum learning. *IEEE Transactions on Cybernetics, 46*(2), 462–473.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., & Watkins, C. (2002). Text classification using string kernels. *Journal of Machine Learning Research, 2*(3), 419–444.

Long, W., Tang, Y. R., & Tian, Y. J. (2016). Investor sentiment identification based on the universum SVM. *Neural Computing & Applications*, 1–10.

Lu, X. (1990). Document retrieval: A structural approach. *Information Processing & Management, 26*(2), 209–218.

Mani, I., & Bloedorn, E. (1997). Multi-document summarization by graph search and matching. In *Proc. of AAAI-97* (pp. 622–628).

Ming, F., Wong, F., Liu, Z., & Chiang, M. (2015). Stock market prediction from WSJ: Text mining via sparse matrix factorization. In *IEEE international conference on data mining* (pp. 430–439). IEEE.

Pfoh, J., Schneider, C., & Eckert, C. (2013). Leveraging string kernels for malware detection. In *Network and system security* (pp. 206–219). Berlin Heidelberg: Springer.

Russell, S. J., & Norvig, P. (1995). *Artificial intelligence: A modern approach*. Prentice Hall.

Schenker, A. (2003). *Graph-techniques for web content mining*. Florida: University of South Florida.

Schumaker, R. P., & Chen, H. (2006). Textual analysis of stock market prediction using financial news articles. In *Connecting the Americas. Americas conference on information systems, Amcis 2006* (p. 185). DBLP.

Seng, J. L., & Yang, H. F. (2017). The association between stock price volatility and financial news - a sentiment analysis approach. *Kybernetes, 46*(8), 1341–1365.

Shiller, R. J. (2005). *Irrational Exuberance: Irrational exuberance*. Princeton University Press.

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.

Tang, X., Yang, C., & Zhou, J. (2009). Stock price forecasting by combining news mining and time series analysis. In *International joint conferences on web intelligence and intelligent agent technologies, 2009. WI-IAT: Vol.1* (pp. 279–282). IEEE.

Tsai, M. F., & Wang, C. J. (2016). On the risk prediction and analysis of soft information in finance reports. *European Journal of Operational Research, 257*(1), 243–250.

Vapnik, V. N. (1998). Statistical learning theory. *Encyclopedia of the Sciences of Learning, 41*(4) 3185-3185.

Vishwanathan, S. V. N., Schraudolph, N. N., Kondor, R., & Borgwardt, K. M. (2010). Graph kernels. *Journal of Machine Learning Research, 11*(2), 1201–1242.

Wang, Z., & Liu, Z. (2010). Graph-based Chinese text categorization. In *International conference on electrical and control engineering* (pp. 1092–1095). IEEE.

Weston, J., Collobert, R., Sinz, F., & Vapnik, V. (2006). Inference with the universum. In *International Conference on Machine Learning: Vol. 2006* (pp. 1009–1016). ACM.

Yu, Y., Duan, W., & Cao, Q. (2013). The impact of social and conventional media on firm equity value: A sentiment analysis approach. *Decision Support Systems, 55*(4), 919–926.

Yazdani, S. F., Murad, M. A. A., Sharef, N. M., Singh, Y. P., & Latiff, A. R. A. (2016). Sentiment classification of financial news using statistical features. *International Journal of Pattern Recognition & Artificial Intelligence, 31*(03), 165–176.

Zeng, X. P., Yi, L., & Liu, G. J. (2010). Image denoising based on spectral graph theory and random walk kernel. *Journal on Communications, 31*(7), 116–121.

Zhu, G., & Iglesias, C. A. (2018). Exploiting semantic similarity for named entity disambiguation in knowledge graphs. *Expert Systems with Applications, 101*, 8–24.