IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON ADVANCED DATA MINING METHODS FOR SOCIAL COMPUTING

Received October 3, 2019, accepted October 17, 2019, date of publication October 23, 2019,
date of current version November 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2949074

Stock Volatility Prediction by Hybrid Neural Network

YUJIE WANG[1,2], HUI LIU[1,2], QIANG GUO[1,2], SHENXIANG XIE[3], AND XIAOFENG ZHANG[4]

[1]School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China
[2]Digital Media Technology Key Laboratory of Shandong Province, Jinan 250014, China
[3]School of Public Finance and Taxation, Shandong University of Finance and Economics, Jinan 250014, China
[4]School of Computer Science and Technology, Shandong Technology and Business University, Yantai 264005, China

Corresponding author: Hui Liu (liuh_lh@sdufe.edu.cn)

This work was supported in part by the National Natural Science Foundation under Grant 61572286, Grant U1609218, and Grant 61873145, in part by the Major Program of the National Social Science Fund under Grant 17ZDA097, in part by the Natural Science Foundation of Shandong Province under Grant ZR2017JL029 and Grant 2016ZRB01143, in part by the Taishan Younger Scholar Special Fund, and in part by the Fostering Project of Dominant Discipline through a Talent Team of Shandong Province Higher Education.

ABSTRACT Stock price volatility forecasting is a hot topic in time series prediction research, which plays an important role in reducing investment risk. However, the trend of stock price not only depends on its historical trend, but also on its related social factors. This paper proposes a hybrid time-series predictive neural network (HTPNN) that combines the effection of news. The features of news headlines are expressed as distributed word vectors which are dimensionally reduced to optimize the efficiency of the model by sparse automatic encoders. Then, according to the timeliness of stocks, the daily K-line data is combined with the news. HTPNN captures the potential law of stock price fluctuation by learning the fusion feature of news and time series, which not only retains the effective information of news and stock data, but also eliminates the redundant information of the text. Compared with the state-of-the-art methods, our method combines more abundant stock characteristics and has more advantages in running speed. Besides, the accuracy is averagely improved by nearly 5%.

INDEX TERMS Stock prediction, news, natural language processing, hybrid neural network.

I. INTRODUCTION

Many social activities, such as weather indices, energy data or economic activities, whose temporal characteristics are an important basis for making prediction. Among them, stock price forecasting is a challenging issue in time series analysis [36]. The high returns attract many investors and traders. In general, investors hardly master the price changes accurately in the stock market. Therefore, it is of great significance to construct a model with high predictive precision, so that we can have a good grasp of the volatility law of the stock market effectively and help the investors avoiding risks and improving profits.

Fama [1] theorized the efficient market hypothesis that for an economic market with perfect laws and regulations, the stock price reflects all available information, which means the relationship between historical stock price and current stock price can be used to forecast the future stock trend. But this method only considers stock price and ignores another important source of price volatility – news, which is time-stamped and situation-relevant [35]. Preis *et al.* [2] used the Google trends to determine the search volume, and found that some events had a large number of search volume about related events before the news occurred. They concluded that search volume not only can reflect the current state, but also predict the future trend. To validate Ref [2], we collect the search volume of Microsoft (MSFT) and the change of stock price (chg) movement in December 2018 on the Google website to verify the effectiveness of news. The trend of search volume and MSFT's chg in the same period are shown in FIGURE 1.

From FIGURE 1, we observe a phenomenon that when the trend heat increases, the corresponding stock's chg becomes higher, basically more than 2%. Accordingly, we can determine that the changes in stock price trends can be reflected in the news heat, and vice versa. Thus a fact can be affirmed that news and stock price reactions have an intimate connection. In this paper, we try to extract the useful text features

The associate editor coordinating the review of this manuscript and approving it for publication was Haishuai Wang.

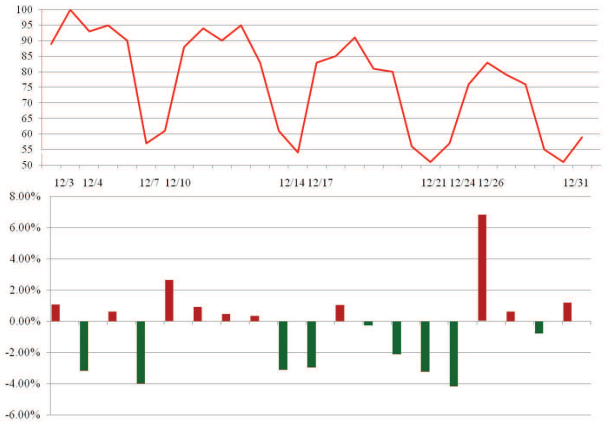

FIGURE 1. MSFT's search volume and chg over the same period.

from news, and combine the data of daily K-line for better forecasting ability of HTPNN. The main work includes: 1) News is mapped to the vector space by using a distributed model. Then sparse auto-encoder (SAE) is used to reduce the dimension of word vector matrix to obtain concise and useful text information; 2) A hybrid neural network model named HTPNN is constructed to predict the stock volatility by jointly using the deep convolution layers for capturing text features and Long Short-Term Memory (LSTM) layer for learning the law of stock prices fluctuate; 3) To retrieve the most relevant news features to explain or influence the stock volatility, deep convolutional layers are adopted; 4) In order to improve the forecasting precision and reduce the single feature prediction error, our approach feeds stock price into LSTM layer, extracts more advanced feature, and combines news and price features to generate the deeper fusion feature, which can effectively predict the stock trend. Experiments demonstrate that the classification accuracy of HTPNN is higher than typical methods.

The remainder of this paper is organized as follows: In Section II, we introduce the related works about the stock prediction; Section III gives the pre-processing and pre-sentation method of news texts; Section IV puts forward the design of our approach for forecasting the stock price fluctuation; Section V compares and analyzes the experimental results on sample sets; the summary of our work and the prospect of future work are presented in Section VI.

II. RELATED WORKS

So far, the studies of stock prediction are primarily divided into two categories: conventional time series prediction methods and soft computing methods [3], [4].

In earlier literature, a series of algorithms based on Auto Regressive and Moving Average model (ARMA) have been used in stock prediction [5]. However, ARMA is suitable for stationary time series analysis. When the time series has a trend of change, the common model is the Autoregressive Integrated Moving Average model (ARIMA), which is used for the analysis of homogeneous non-stationary time series. Ariyo *et al.* [6] used ARIMA model for the purpose of predicting the price in the New York Stock Exchange (NYSE) and Nigeria Stock Exchange (NSE). The experiment demonstrated that ARIMA had a strong ability of forecasting stock prices in short-term. However, in long-term time series prediction, the Generalized Autoregressive conditional heteroskedasticity model (GARCH) performs better than ARIMA. It focuses on analyzing the variation of conditional variance of random perturbation term in time series and describes the sequence more accurately [31]. Ariyo *et.al.* [8] compared different GARCH models and concluded that the asymmetric model was superior to other GARCH models. If asymmetric properties are neglected, the GARCH model with normal distribution is preferable to the models with more sophisticated error distributions. Refs [9] and [11] also illustrated the superiority of asymmetric GARCH models.

The soft computing methods overcome the limitation of linear models and focus on extracting the non-linear relation in stock series. It determined that the predicted results could be acquired without prior knowledge of the statistical distribution of the input data [10]. The soft computing techniques primarily include the neural network and the neuro-fuzzy techniques which are based on the deep learning strategy [4]. The deep learning method is a kind of non-linear model, which makes the best use of the training data and learns the internal correlations between training data in the original feature space after multiple iterations. Hence, for complex non-linear systems such as the stock market, the deep learning method can make more accurate prediction which is crucial for stock price forecasting [13]. Dixon *et al.* [14] used deep neural networks (DNNs) to predict the price changes and futures of 43 commodities within 5 minutes. The back propagation (BP) algorithm was used to achieve an accuracy of 42%. In Ref [15], Fehrer and Feuerriegel constructed a German stock return model based on news headlines. They used a recursive self-encoder with an additional softmax layer to predict the stock returns associated with the news headlines for the next day, and achieved 56% precision. Similarly, the volatility of Standard & Poor's (S&P) 500 forecasting model based on LSTM was proposed by Ref [16], which focuses on the effect of the input set on prediction results. The input set includes volatility and 25 domestic trends in the main areas of the country. And the accuracy exceeds 60% compared to other baseline models.

The models used in conventional methods can not capture the non-linear mode easily [4]. In recent literature, Zhang et al. proposed a State Frequency Memory (SFM) recurrent network to capture the multi-frequency trading patterns from historical prices [7]. The future stock prices were predicted as a nonlinear mapping in an Inverse Fourier Transform (IFT) fashion. In addition, Ref [38] noted that the prediction of price trends not only depends on historical stock data. Li et al. designed a Technical Trading Indicator Optimization (TTIO) framework [10]. The effective representations of stock properties were learned by a skip-gram architecture. Based on the learned stock representations, TTIO

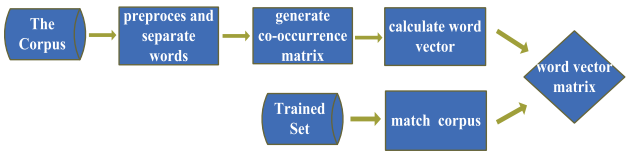

FIGURE 2. The flow chart of extracting news events and training word vectors.

learned a re-scaling network to optimize the performance of the indicator. The stock market experiment showed that the optimized indicator can produce stronger investment signals than the original ones. Similarly, Chen *et al.* [12] used stock representation to model the market state, generate a dynamic correlation between stock and market, and aggregate with dynamic stock indicators to achieve more accurate stock forecasting.

III. THE EMBEDDING OF NEWS

This section introduces the extraction of the key information in news and its description in HTPNN. Section A will introduce the news pre-processing in the experiments. Section B and Section C describe how to generate news word vectors.

Radinsky *et al.* [17] attested that news headlines were more helpful than contents in prediction. Therefore, we only extract events from news headlines. First, the pre-processed corpus is inputted into the word vectors model. Then the word vectors model computes the co-occurrence matrix, and calculates word vectors according to co-occurrence probability. Finally, a word vectors matrix about the training sets with ranking in the time order is generated. The process of extracting and training news events is shown in FIGURE 2.

A. PRE-PROCESSING OF NEWS HEADLINES

In order to convert text information from human language to machine-readable descriptions for subsequent processing, there should be a standardized pre-processing for text [39]. The steps of text normalization are discussed as following: 1) Fixed non-trading days can be observed in the stock market, but the date of stock news appearing is uncertain. According to experience, the important non-trading news still appears on the next trading day. So we remove the non-trading news to guarantee the news date matches the stock time [18]. For individual stocks that have no news on trading day, by default, the stock fluctuation is influenced by the last news events. 2) The word segmentation module NLTK in python3 is adopted to spell check for the news text, remove special symbols and divide news headlines into word sets. 3) News includes some most frequent words without meaning which provide less information value for the news [30]. So we use the stop-words list to take out stop words to eliminate the noise of news. After all news in the training set is split into a similar structure, the word co-occurrence matrix and the corresponding word vectors are calculated. In this way, we obtain the continuously distributed representation of each news.

B. TRAINING CO-OCCURRENCE MATRIX AND WORD VECTORS

Generally speaking, the representation methods for words include two types: one-hot representation and distributed representation. One-hot representation is the most intuitive and common technique. Each word in the thesaurus is coded to the same vector dimension. The word in corresponding positions is encoded 1, the rest dimensions are 0. However, this method only expresses the position of words, ignoring the semantic relationship between words. The distributed representation method proposed by Ref [19] stated that each word was represented by its context, which retaining more sentence information. Two kinds of typical distributed representation models are 1) Global matrix decomposition methods, such as latent semantic analysis (LSA) [33]; 2) The local context window methods, such as word2vec model [34].

LSA based on SVD matrix decomposition obtains two sets of vector for documents and words via document information and co-occurrence words of window size respectively. But it performs poor on word analogy. The word2vec model learns the occurrences of other words around one word to obtain the low-dimensional word vectors. However, this method trains each context window separately without taking advantage of the statistical information in the co-occurrence matrix, so the processing ability of polysyllabic words is weak. In order to overcome the shortcomings of the models mentioned above, we choose the weighted least squares model GLOVE proposed by [23], which is based on the distribution representation of the matrix. GLOVE model combines the advantages of one-hot and distributed representation. Rows of word vectors matrix represent the corresponding words. The similarity of words is expressed by the vector space distance directly which describes the context of word distribution. The operation of training non-zero elements in the word co-occurrence matrix effectively utilizes the global and context information of the corpus (sliding window) to generate meaningful substructure vector space which is suitable for different sizes of corpus. The procedure of calculating word vectors is as follows:

The co-occurrence matrix X is obtained after traversing the entire corpus. Based on the approximate relationship between word vectors and co-occurrence matrices, the general formula of word vectors is calculated.

$$log(X_{ik}) = v_i^T v_k + b_i + b_k, \tag{1}$$

where X_{ik} is the number of times the word k appears in the context of word i. v represents the word vector. b_i and b_k are the bias terms corresponding to word vectors v_i and v_k.

The loss function is defined as:

$$J = \sum_{i,j=1}^{N} f\left(X_{ij}\right)\left(v_i^T v_j + b_i + b_j - logX_{i,j}\right)^2, \tag{2}$$

where N denotes the size of the corpus. As a matter of experience, words with very high frequency may be noise. In order to reduce the impact of such words on results, weight function

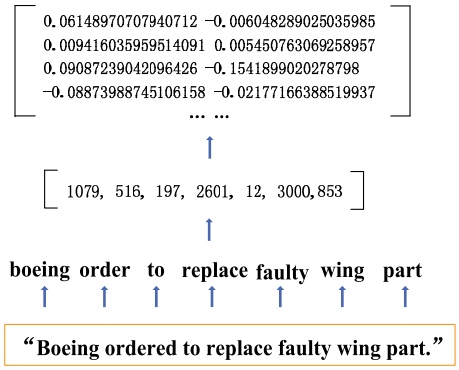

FIGURE 3. In the process of generating word vectors, the sentence is segmented to the word sets whose index is obtained by matching the keywords matrix of the corpus. The index matrix is inputted for training to get the word vectors matrix of news titles.

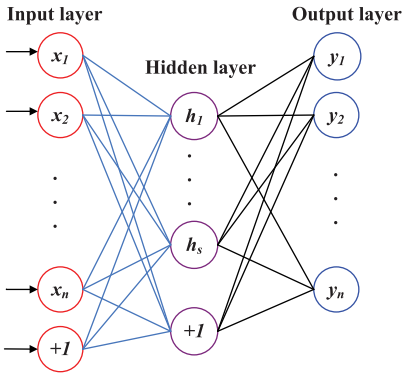

FIGURE 4. The structure of SAE. *h* is the expected feature.

$f(x)$ is added into the loss function. The higher the word frequency, the greater the weight.

In order to enrich the semantics of word vectors as much as possible, the news headlines of selected stocks from 2014 to 2016 are collected as corpus. After pre-processing, according to the occurrence frequency of words, 7,500 keywords with the most number of occurrences are retained, and the rest words in the corpus are signed ''unknown''. In this way, the keywords co-occurrence matrix has remained. Then, the training news texts are inputted into the GLOVE model for the final semantic word vectors matrix which is matched with the keywords matrix. For example, the training process of news headline ''Boeing ordered to replace faulty wing part'' is shown as FIGURE 3. From the overall training set, the maximum of each news headline to be represented by 25 words. The word of each headline is mapped into the word vector with 50 dimensions. The size of the sliding window in GLOVE is assigned as 10.

C. REPRESENTATION OF TEXTUAL INFORMATION IN HTPNN

In this paper, the dimension of news word vectors matrix is larger than that of price, which will enhance the importance of news and affect the final prediction accuracy. As is well known, when the text is with high dimension and a large amount of data, the matrix after segmentation is generally sparse. So deep SAE was adopted to reduce the dimension of text, by extracting the structure hidden in data. In Ref [20], the text feature was extracted by using PCA, a shallow SAE and a deep SAE. The result indicated that the deep SAE has higher recognition accuracy, better generalization and more stability. As a consequence, deep SAE is used to obtain the feature sets which are brief but still retain the main information. The structure of SAE is shown in FIGURE 4.

Deep SAE includes encoder and decoder with size of 25*25, the number of neurons in hidden layer is 256 according to experience. Then the text feature $h(x_i)$ can be given as:

$$h(x_i) = \sigma\left(\omega(x_i) + b_x\right), \tag{3}$$

where x is the word vectors matrix. σ denotes the Sigmoid function in (3). Weight matrix indicates $\omega \in R^{m*n}$. n is the scale of input layer, and m denotes the scale of hidden layer. b_x is the bias term.

In practice, sparse representation is more effective than others. In order to satisfy the sparsity and make the learned features constrained, the sparse penalty term was added to the cost function of SAE. The common term is KL divergence [21]. For avoiding over-fitting, we used the L2 norm which is a common regularized constraint item to improve the comprehensiveness of feature vectors [22]. Therefore, the cost functions L_{sparse} can be defined as:

$$L_{sparse} = \sum L\left(x, y\right) + \lambda \sum_{i,j} \omega_{i,j}^2 + \beta \sum_{j=1}^{m} KL\left(\rho || \hat{\rho}_j\right), \tag{4}$$

where $\sum L\left(x, y\right)$ denotes the average reconstruction error, $\omega_{i,j}^2$ is the weight-decay. To enhance sparsity and minimize the cost function, the average activation for each hidden unit $\hat{\rho}_j$ is expected to be close to the sparsity penalty factor ρ. λ is the weight attenuation parameter. β is the weight coefficient of sparsity penalty term. The error of SAE is relatively small, so we hold that the features obtained by encoder can represent the meaning of the text.

IV. THE STRUCTURE OF HTPNN MODEL

Having obtained the word vectors matrix in Section III.B, this section will retrieve the basic stock data and split it with the word vectors matrix to serve as the input of HTPNN, which is composed of deep convolution layer, LSTM layer, feature fusion layer,and full connection layer. FIGURE 5 describes the specific structure of HTPNN.

A. INPUT SET AND LABEL SET

We expect more complex features to improve the forecasting ability of our model. Thus in addition to the news vector matrix in Section III, we also use the web crawler to obtain the basic information about the stock price in Yahoo Finance, including the stocks codes, the daily opening price, closing price, the highest price, the lowest price, the transaction amount and volumes. In actual stock trading, some individual

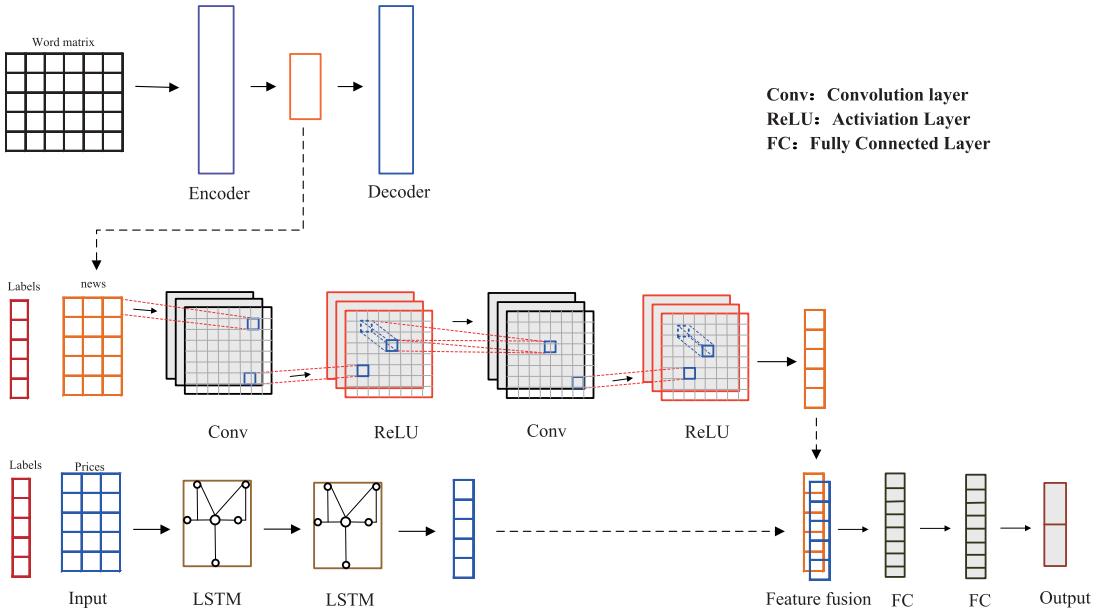

FIGURE 5. A graphical structure of HTPNN consists of SAE, deep convolution layer, LSTM layer, feature fusion layer and full connection layer.

stocks change their trend due to the dividends. The rehabilitation can repair the stock price, so that investors are able to judge whether the current price is at a relative historical high or low. Therefore, all the selected prices are split-adjusted data which represent the information of stock. The sample set is composed of 12 DOW component stocks including the Dow Jones Industrial index (DJIA) which represents the contrast of the benchmark. We regard the chg of each stock as the label set by calculating the relative rate of return R_t.

$$R_t = lnc_t - lns_t, \tag{5}$$

where lnc_t and lns_t are the logarithms of component stocks and DJIA daily closing price, respectively. We divided the data sets into three kinds according to time windows with different lengths: long-term (30 days), mid-term (7 days), short-term (1 day). Meanwhile, the EMV index was added to the stock price set on the basis of k-line data for expressing the stock trend chg fully. EMV index is an agile index that reflects stock price fluctuations and the relationship between stock price and trading volume [24]. When EMV index and the neural network are used to predict future stock volatility and investment, the profit results are higher than that of single index trading strategy [25]. The function of EMV can be written as (6):

$$\text{EMV} = \sum_{t=1}^{N} \left\{ \left(\frac{c^t_{\max} + c^t_{\min}}{2} - \frac{c^{t-1}_{\max} + c^{t-1}_{\min}}{2} \right) * \frac{c_{max} + c_{min}}{V} \right\}, \tag{6}$$

where $c_{(t)max}$ and $c_{(t)min}$ are the max/min value of the stock price on day t. c_{max} and c_{min} are the max/min value of the stock price in N days. V represents the stock trading volume. So the final stock information vector consists of [open, close, highest, lowest, volume, amount, EMV]. The output of the model will be labeled price rise (category 1) and other situations – price unchanged or falling (category 0).

B. THE CNN LAYER

The sequence of word vectors matrix is concise and arranged in chronological order. One-dimensional convolution can reduce the amount of computation and extract news features by sliding one-dimensional convolution kernels. The news text with length n (filled with vector 0 when the length is insufficient) can be expressed as:

$$x_{1:n} = x_{1\cdots k} \oplus x_{2\cdots k} \oplus \cdots \oplus x_{(i+j)\cdots k}, \tag{7}$$

where $\oplus$ denotes the connect operators, $x_{i:k}$ is the news $U_i = \left\{x_{i\cdots k}, x_{(i+1)\cdots k}, \cdots, x_{(i+j)\cdots k}\right\}$ with the number of words for j. The word vectors dimension of the i-th word in a sentence is k.

Based on experience in this paper, 2 convolution layers gave better experimental results. Each layer is convolved with an activation layer ReLU. The convolution operation contains a filter $\delta \in R^{hk}$ which is applied to the window of h words to extract the news feature c_i:

$$c_i = f\left(\delta * x_{i:i+h-1} + b_i\right), \tag{8}$$

where the convolution operation $*$ is based on sliding window feature extraction, $b_i \in R$ is the bias term. f indicates the non-linear function of the hyperbolic tangent. The filter δ is applied to every possible words window of the sentence $U_i = \left\{x_{i\cdots k}, x_{(i+1)\cdots k}, \cdots, x_{(i+j)\cdots k}\right\}$ to generate the feature map. The news events with length n are inputted into the convolution layer to gain the news features sequence $Q = (c_1, c_2, \cdots, c_n)$. Due to the dimension reduction of SAE, we can obtain low-dimensional eigenvectors without pooling.

After the convolution operation, the resulting feature vector (Q_i, Q_m, Q_l) consists of long-term, mid-term and short-term sequences. The news vectors will combine the features of stock price generated by LSTM layer.

C. THE LSTM LAYER

In the related work for stock data forecasting, the most commonly used model is Recurrent Neural Network (RNN). However, the training process of RNN requires more parameters. In addition, the serious problem that appears in the state calculation is gradient vanish. Therefore, as an improvement of RNN, LSTM has been widely used in time series data prediction. It adds the gate unit based on the RNN memory unit, so that each input can control the retention of previous and current information. Compared with RNN, LSTM learns the long-term dependence of sequences and makes full use of the temporal features. When stock fluctuation features are extracted, we should not only consider the spatial relationship between news and stocks, but also pay attention to the change of stock sequence in the time dimension. So LSTM is suitable for time-dependent learning of stock price.

The steps of training sequence can be described as follows: 1) The input at each moment is divided into three parts, including the output, the hidden state of the unit at the previous moment, and the sequence at the current moment. The forgetting rate of the unit state at the previous moment is calculated by sigmoid function of the forgetting gate, ranging from (0,1). 2) The input gate handles the input of the current sequence position, which consists of two parts: the first part goes through the sigmoid function, and the second part utilizes the Tanh activation function to update the hidden cell state. 3) Before passing through the output gate, the hidden state at the current moment is updated by multiplying the results forget gate in 1) and the input gate in 2) respectively. 4) At the final step, the output is calculated by the output state and sequence of the last moment via Sigmoid function, which will combine with the updated hidden state to get the final result.

The stock price sequence was fed into LSTM to get the feature $S_t = \{s_1, s_2, \cdots, s_n\}$($n$ is the sequence length), and

$$s_n = f\left(c_{n-1}, h_{n-1}, x_n\right), \tag{9}$$

where f is the way of LSTM to accomplish, c_{n-1}, h_{n-1}, x_n represent the output state at the previous moment, the unit hidden state, and the current input sequence respectively.

D. THE FULL CONNECTION LAYER

Based on the above operations, we get the news feature Q and stock feature S, which are complementary and explanatory to each other. After mapping them into the same dimension space, we input them to the fusion layer to get the fusion feature Z which will be the input of the full connection layers. The fusion operation is composing them together [29] as follows.

$$Z = \phi\left(\omega_Q \cdot Q + \omega_S \cdot S\right), \tag{10}$$

where ϕ is the ReLU, "+" indicates the element-wise addition. In order to simplify the exposition, the actual meanings of ω and b are distinguished by subscripts. ω^Q and ω^S are the weights of Q and S with a slight abuse of notation. The output of sequence Z in the fusion layer will pass through the full connection layer to obtain the output y_t. (11) explains the operation function in full connection layer.

$$y_t = \sigma\left(w_t Z_t + b\right), \tag{11}$$

where σ is the Sigmoid function, w_t is the weight vector and b is the bias item. So we get the output after the softmax layer:

$$y_{cls}\left(y_t \in \{0, 1\}\right), \tag{12}$$

where y_{cls} is the output of softmax layer which represents the probability of network output category. For the purpose of strengthening the network generalization ability, each layer joins the dropout layer and random loses part of neurons function to avoid the fixed combination. Finally, the category with the highest probability represents the forecasting result. In order to accelerate the speed of weight updating during training, the loss function is defined as the cross entropy. The model is trained for 60 epochs with Adam until the loss converges.

$$loss = -\sum_{i=1}^{n} y_t logs_t + (1 - y_t) log(1 - y_t), \tag{13}$$

The results of each training are expressed as a confusion matrix which is utilized to calculate the precision rate and the correlation coefficient. In machine learning algorithms, confusion matrix is the index to evaluate the results of model. Each column of the matrix represents the sample situation predicted by model. Each row is the real situation of the sample.

V. EXPERIMENTAL RESULTS AND ANALYSIS

This section primarily describes the selection process of experimental data and evaluation indices, analyzes the influence of different parameters on experiment and compares the performance with the baseline methods.

A. THE INTRODUCTION OF EXPERIMENTAL SET

As one of the three major indices, the DJIA index has the characteristics of large scale, good reputation and industrial representation, which is of great significance to reflect the development trend of stock market. The web crawler component was used to get the individual stock on Yahoo Finance during 2017-1-01 to 2019-3-01. However, due to Yahoo Finance's information restrictions, there may exist too little news about some stocks to extract enough effective information. Therefore, we selected the news which more than 200 items per year and corresponding stock information to download. For stocks with more than one news per day, we chose the news with the largest number of comments per day as representative. The contents of crawling consist of stock code, daily open price, close price, the highest price,

TABLE 1. Data distribution of three lengths in the sample set.

Data set	Short	Mid	Long
The number of Train	3640	2400	820
The number of Validation	1420	1000	600
The number of Test	1400	1200	710

TABLE 2. The general form of confusion matrix.

	Positive	Negative
True	TP	TN
False	FP	FN

lowest price, the trade volume, news headlines, and date. Having in mind that the news titles had more predictive value than news content [27], we only consider the headlines of stocks' news. The conjunction of stock information and word vectors matrix after dimension reduction becomes the training input of HTPNN. The experiment is conducted on three periods – short-term, mid-term and long-term. Data set is pre-processed for 5,460 trading days aligned with three periods respectively. The pre-treatment includes deleting the duplicate data and normalizing data. Table 1 describes the composition of data set, in which training set accounts for 2/3, verification and test set account for 1/6 respectively.

B. EXPERIMENTAL EVALUATION INDICES

Based on the relationship between the forecasting value and the true value, the quantitative performance of the model can be expressed by a confusion matrix, shown in Table 2. The samples were divided into four parts, namely:

True Positive (TP): Both the predicted value and the true value are 1.

False Positive (FP): The predicted value is 1, the true value is 0.

True Negative (TN): Both the predicted value and the true value are 0.

False Negative (FN): The predicted value is 0, the true value is 1.

Accuracy is the most intuitive indicator of the predicted result.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}}, \tag{14}$$

Nevertheless, Accuracy is sensitive for the data distribution. When the distribution of particular situation in prediction results is particularly large, the classifiers with predicting large logarithmic categories will lead to higher accuracy, which is not objective enough to evaluate a model. Accordingly, we cited the Matthews Correlation Coefficient (MCC) to avoid bias due to the data skew [26]. As a common indicator to evaluate the performance, MCC is a single summary value which contains all cells of the confusion matrix for predicting and observing the results.

$$\text{MCC} = \frac{\text{TP}\cdot\text{TN} - \text{FP}\cdot\text{FN}}{\sqrt{(\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}}, \tag{15}$$

TABLE 3. Parameter settings of HTPNN.

Parameters	Value
Convolution layer	2
LSTM layer	2
Size of input matrix	$25 * 100$
Convolution kernels	3
Neurons of convolution kernel	64, 32
Learning rate	0.005
Iteration number	1000

TABLE 4. Influence of news on experimental results.

Input	Accuracy	MCC
News+EMV+price	69.51%	0.337
News+price	65.97%	0.316
Only price	42.02%	0.023

TABLE 5. Influence of different amounts of short-term data.

Evaluation indices	Title	Content	Title+Content
Accuracy	69.51%	60.33%	64.07%
MCC	0.337	0.025	0.302

C. PARAMETER ANALYSIS

This subsection will discuss the importance of feature selection by using the evaluation indices in Section V.B. Considering the efficiency and speed of the model, the better network parameters are also analyzed.

1) IMPACT OF NEWS AND THE EMV INDEX ON FORECASTING RESULTS

In order to display the importance of news headlines and EMV index, the news vectors matrix and the price matrix were inputted into the model respectively. Parameter settings of the model are shown in Table 3. Table 4 lists the comparison results.

The results in Table 4 validate the necessity of news and EMV. The financial market price may not reflect all the changes of stock volatility. As social media, news can effectively aggregate effective information from various places and react to stock price changes. Combined with EMV indicator reflecting stock price fluctuations, it not only describes stock price changes more accurately, but also enriches the variety of learned features. Therefore, the news word vectors and EMV index improve the prediction accuracy effectively.

2) THE EFFECT OF TEXT DATA VOLUME ON PREDICTION RESULTS

Although news titles can provide the center information, the contents also provide some background knowledge or details. In this subsection, we design a comparative experiment to analyze the effectiveness of the news headlines and contents. Parameter settings are the same as Table 3. According to the results in Table 5, the news headline has the best performance in stock forecasting.

TABLE 6. Influence of different number of convolutional layers.

Layer	Evaluation indices	Short	Mid	Long	speed
1-layer	Accuracy	64.71%	60.59%	62.71%	1s
	MCC	0.425	0.198	0.371	
2-layer	Accuracy	69.51%	63.68%	64.19%	43s
	MCC	0.524	0.358	0.417	
3-layer	Accuracy	60.31%	58.73%	58.83%	83s
	MCC	0.195	0.162	0.163	
4-layer	Accuracy	58.07%	52.11%	48.83%	120s
	MCC	0.072	0.066	0.045	

The concise headline can overview the meaning of the whole news. In general, the number of words in news content is large, which means that the semantics is too complex to contain more noise. Table 5 describes the fact that the quality of news is more important than the quantity, which means the most relevant information (such as news headline) is better than the information with large amount but less relevant.

3) INFLUENCE OF THE DIFFERENT NUMBER OF CONVOLUTIONAL LAYERS ON PREDICTION RESULTS

The number of convolution layer has a certain impact on the experimental results. In this subsection, we mainly discuss the network structure of HTPNN. The different number of convolution layer was selected for the comparison, with the activation function ReLU. The results of HTPNN with 1 LSTM layer and 1 to 4 convolution layers are analyzed in Table 6.

It can be seen from Table 6 that the prediction result of 2-layer convolution is the best, which indicates that the multi-convolution layer can explain more complex classification relations. But when the model reaches 3-convolution layer, the accuracy and the speed decrease. The reason for decrease may be related to the size of data sets. The deeper the model layer, the greater the extension of amplitude, which will affect the model performance ultimately. Another possible reason is that the gradient disappear or gradient explosion. In the neural network, if the weight initialization value is too large, the gradient in back propagation may attenuation or increase in the form of index, which results in the decrease of prediction accuracy. In addition, the short-term results are superior to the mid-term and long-term. Perhaps the reason is the delay between price reaction and news. In some cases, events can lead to immediate changes in stock prices. For example, on May 30, 2019, US President Trump announced to impose tariffs of 25% on imports from Mexico, which caused panic in the stock market. The three major US stock indices all fell by more than 1%, among which the DJIA index fell by 354.84 points, about 1.41% [32].

4) INFLUENCE OF DIFFERENT LSTM LAYERS ON PREDICTION RESULTS

In our hybrid neural network, LSTM and convolution layer are the main factors that affect the prediction results. So we also discuss the effect of LSTM layers on the experiment when convolution layers in HTPNN are 2. The results are shown in FIGURE 6.

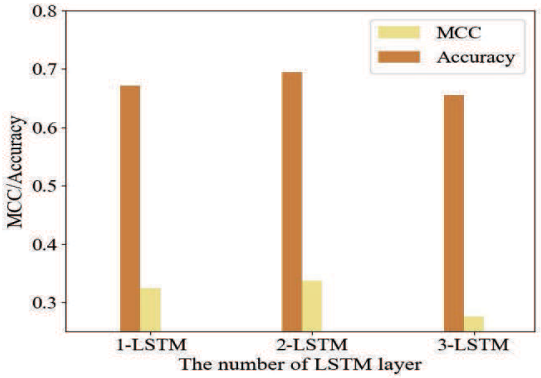

FIGURE 6. The effect of LSTM layer number on the results.

TABLE 7. The results on different models.

Methods	Accuracy	MCC
BOW-SVM	56.42%	0.073
WB-NN	60.25%	0.198
E-NN	60.84%	0.201
BOW-NN	60.61%	0.200
BOW-CNN	64.39%	0.446
GM-CNN	65.97%	0.451
HTPNN	69.51%	0.524

FIGURE 6 indicates that, with the increase of layers, the dimension of the hidden layer unit decreases, which leads to the decrease of the effective dimension of the model and the degradation of the network. In addition, too many layers of LSTM will lead to long training time of the model, and may also have the problem of fitting which is not suitable for a time series prediction need. Therefore, based on the quantitative results in the comparative experiment, the HTPNN of 2-layer CNN and 2-layer LSTM achieves superior performance.

D. COMPARATIVE EXPERIMENT

In order to give a comprehensive assessment to HTPNN, the experimental results are compared with the baseline models, as shown in Table 7.

1. BOW-SVM [28]: Luss proposed to construct a predictive model with SVM. The training set includes news documents and labeled classes. And the news features are determined by the Bag-of-Words model (BOW), the classification categories are calculated by linear functions.

2. GM-NN: The same word vectors and the standard feedforward neural network (NN) are built to compare with HTPNN.

3. E-NN [26]: The structured event tuples $E = (O_1; P; O_2)$ represented news documents. The relationship between events and stock price changes was studied by NN.

4. BOW-NN [29]: Methods of processing text also influence the prediction results. We select the BOW model to extract features of the news text and compare it with the

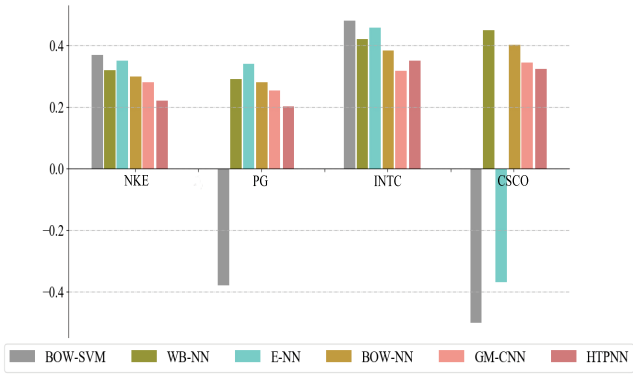

FIGURE 7. The average error of individual stocks. By comparison, the average error calculated by HTPNN is the lowest, ranging from 0.2 to 0.4.

GLOVE model. The word frequency represents the word vector. When the number of words appearing in the text is taken as the initial word frequency matrix, we use the TF-IDF model to calculate the inverse document frequency of words and delete the words with too low weight. The word frequency matrix of news text is fed into NN for predicting.

5. BOW-CNN: Based on the BOW-NN method, we test the ability of CNN to extract text features with the same BOW model. Furthermore, compared with GM-CNN, the importance of text representation is also discussed.

6. GM-CNN: In order to verify the ability of LSTM for extracting the sequence features, we design a convolutional neural network for comparison. The remainder parameter settings are the same as HTPNN.

Except for the comparison on the experimental set, four representative component stocks with good liquidity (NKE, PG, INTC, CSCO) are selected to calculate the average error of predicted value and true value in short-term. FIGURE 7 illustrates that the average error of HTPNN is the smallest.

Based on the above analysis, our model achieves better results on both the experimental set and individual stocks. From Table 7 and FIGURE 7, the performance of the CNN-based model outperforms the SVM-based or the NN-based prediction model. BOW-SVM model can not capture the meaning of the text since the words in BOW model are all independent without combining the semantic connection with the news [37]. SVM cannot joint the time characteristics of stock for its own characteristics, and it also shows weakness in multi-classification problems [40]. While NN only accepts a layer of all input to extract the characteristics, the forecasting result is not accurate as HTPNN. One-dimensional convolution layers learn the relationship between news events and extract more representative of the feature. And LSTM learns the hidden time dependencies in the stock price sequence. Hence, the ability of HTPNN to extract implicit features is stronger. On the one hand, the news title represents the main meaning. On the other hand, low-dimensional vectors can effectively solve the problem of feature sparsity. Furthermore, GLOVE may carry more expressiveness than the BOW model. If the news text is not filtered, some of the words may become noises and interfere with the original semantic feelings of news. In the case of using the same word vectors, HTPNN is superior to the single neural network such as the CNN-based model. It not only combines the features of events related to stocks, but also extracts the deeper feature of stock fluctuations by LSTM, which provides a comprehensive feature basis for the stock fluctuation prediction. And the feature fusion layer plays the role of reducing the feature dimensions and speeding up the running time of HTPNN.

VI. CONCLUSION

The main contribution of this paper is to provide the integration of advanced text features and basic stock information. When predicting the stock volatility, most works only consider the influence on volatility trends so that the extraction of stock features is not comprehensive enough. In this paper, we built a novel fluctuation forecasting model HTPNN with enhanced feature extracting ability via studying the implicit relationship between news and stock daily price. For this, HTPNN achieved superior performance in terms of balancing the prediction effect and running time. Meanwhile, the representation and processing methods of news were discussed. The transformation of news headlines into low-dimensional vector matrices with rich semantics is the key of the method. Our work demonstrated that the useable features embed in news promote the prediction accuracy significantly above the baseline methods. The test on individual stocks confirmed the practical applicability of HTPNN.

It should be noted that our segmentation length for news and index sequences are fixed. However, in actual trading, the effect of different events on stock fluctuation may be different. In future works, we will analyze the impact of event intensity and how to divide the sequence window more scientific for better prediction accuracy, ultimately achieve the goal of profitability by applying the investment strategy.

REFERENCES

[1] E. F. Fama, "The behavior of stock-market prices," *J. Bus.*, vol. 38, no. 1, pp. 34–105, 1965.

[2] T. Preis, H. S. Moat, and H. E. Stanley, "Quantifying trading behavior in financial markets using Google trends," *Sci. Rep.*, vol. 3, Apr. 2013, Art. no. 1684.

[3] G. S. Atsalakis and K. P. Valavanis, "Surveying stock market forecasting techniques—Part I: Conventional methods," *J. Comput. Optim. Econ. Finance*, vol. 2, no. 1, pp. 45–92, 2010.

[4] G. S. Atsalakis and K. P. Valavanis, "Surveying stock market forecasting techniques—Part II: Soft computing methods," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 5932–5941, 2009.

[5] T. Kimoto, K. Asakawa, M. Yoda, and M. Takeoka, "Stock market prediction system with modular neural networks," in *Proc. IJCNN Int. Joint Conf. Neural Netw.*, 1990, pp. 1–6.

[6] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the ARIMA model," in *Proc. UKSim-AMSS 16th Int. Conf. Comput. Modelling Simulation*, Mar. 2014, pp. 106–112.

[7] L. Zhang, C. Aggarwal, and G.-J. Qi, "Stock price prediction via discovering multi-frequency trading patterns," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 2141–2149.
[8] A. A. Adebiyi, A. O. Adewumi, and C. K. Ayo, "Comparison of ARIMA and artificial neural networks models for stock price prediction," *J. Appl. Math.*, vol. 2014, Mar. 2014, Art. no. 614342.
[9] B. M. A. Awartani and V. Corradi, "Predicting the volatility of the S&P-500 stock index via GARCH models: The role of asymmetries," *Int. J. Forecasting*, vol. 21, no. 1, pp. 167–183, 2005.
[10] Z. Li, D. Yang, L. Zhao, J. Bian, T. Qin, and T.-Y. Liu, "Individualized indicator for all: Stock-wise technical indicator optimization with stock embedding," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 894–902.
[11] J. Marcucci, "Forecasting stock market volatility with regime-switching GARCH models," *Stud. Nonlinear Dyn. Econ.*, vol. 9, no. 4, pp. 1–42, 2005.
[12] C. Chen, L. Zhao, J. Bian, C. Xing, and T.-Y. Liu, "Investment behaviors can tell what inside: Exploring stock intrinsic properties for stock trend prediction," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 2376–2384.
[13] H. Wang, Z. Cui, Y. Chen, M. Avidan, A. B. Abdallah, and A. Kronzer, "Predicting hospital readmission via cost-sensitive deep learning," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 15, no. 6, pp. 1968–1978, Dec. 2018.
[14] M. Dixon, D. Klabjan, and J. H. Bang, "Classification-based financial markets prediction using deep neural networks," *Algorithmic Finance*, vol. 6, nos. 3–4, pp. 67–77, 2017.
[15] S. Feuerriegel and R. Fehrer, "Improving decision analytics with deep learning: The case of financial disclosures," 2015, *arXiv:1508.01993*. [Online]. Available: https://arxiv.org/abs/1508.01993
[16] R. Xiong, E. P. Nichols, and Y. Shen, "Deep learning stock volatility with Google domestic trends," 2008, *arXiv:1512.04916*. [Online]. Available: https://arxiv.org/abs/1512.04916
[17] K. Radinsky, S. Davidovich, and S. Markovitch, "Learning causality for news events prediction," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 909–918.
[18] A. Groß-Klußmann and N. Hautsch, "When machines read the news: Using automated text analytics to quantify high frequency news-implied market reactions," *J. Empirical Finance*, vol. 18, no. 2, pp. 321–340, 2011.
[19] Z. S. Harris, "Distributional structure," *Word*, vol. 10, nos. 2–3, pp. 146–162, 1954.
[20] H. Liu and T. Taniguchi, "Feature extraction and pattern recognition for human motion by a deep sparse autoencoder," in *Proc. IEEE Int. Conf. Comput. Inf. Technol.*, Sep. 2014, pp. 173–181.
[21] H. Lee, C. Ekanadham, and A. Y. Ng, "Sparse deep belief net model for visual area V2," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 873–880.
[22] M. Längkvist and A. Loutfi, "Learning feature representations with a cost-relevant sparse autoencoder," *Int. J. Neural Syst.*, vol. 25, no. 1, 2015, Art. no. 1450034.
[23] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
[24] R. W. Arms, *Trading Without Fear: Eliminating Emotional Decisions With Arms Trading Strategies*, vol. 58. Hoboken, NJ, USA: Wiley, 1996.
[25] T. Chavarnakul and D. Enke, "Stock trading using neural networks and the ease of movement technical indicator," in *Proc. IIE Annu. Conf.* Peachtree Corners, GA, USA: Institute of Industrial and Systems Engineers, 2006, p. 1.
[26] X. Ding, Y. Zhang, T. Liu, and J. Duan, "Deep learning for event-driven stock prediction," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 2327–2333.
[27] X. Ding, Y. Zhang, T. Liu, and J. Duan, "Using structured events to predict stock price movement: An empirical investigation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1415–1425.
[28] R. Luss and A. D'Aspremont, "Predicting abnormal returns from news using text classification," *Quant. Finance*, vol. 15, no. 6, pp. 999–1012, 2015.
[29] T. Geva and J. Zahavi, "Empirical evaluation of an automated intra-day stock recommendation system incorporating both market data and textual news," *Decision Support Syst.*, vol. 57, pp. 212–223, Jan. 2015.
[30] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
[31] R. C. Garcia, J. Contreras, M. V. Akkeren, and J. B. C. Garcia, "A GARCH forecasting model to predict day-ahead electricity prices," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 867–874, May 2005.
[32] B. John. (2019). Dow Futures Plummet After Trump Announces New Mexico Tariffs. The Hill. [Online]. Available: https://thehill.com/policy/finance/446272-dow-plummets-after-trump-announces-new-mexico-tariffs
[33] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *J. Amer. Soc. Inf. Sci.*, vol. 41, no. 6, pp. 391–407, 1990.
[34] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*. [Online]. Available: https://arxiv.org/abs/1301.3781
[35] C. Xie, W. Chen, X. Huang, Y. Hu, S. Barlowe, and J. Yang, "VAET: A visual analytics approach for E-transactions time-series," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 2, pp. 1743–1751, Dec. 2014.
[36] H. S. Wang, J. Wu, P. Zhang, and Y. Chen, "Learning shapelet patterns from network-based time series," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 3864–3876, Jul. 2019.
[37] H. S. Wang, Q. Zhang, J. Wu, S. Pan, and Y. Chen, "Time series feature learning with labeled and unlabeled data," *Pattern Recognit.*, vol. 89, pp. 55–66, May 2019.
[38] C. Li, D. Song, and D. Tao, "Multi-task recurrent neural networks and higher-order Markov random fields for stock price movement prediction: Multi-task RNN and higer-order MRFs for stock price classification," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 1141–1151.
[39] J. Lu, W. Chen, Y. Ma, J. Ke, Z. Li, F. Zhang, and R. Maciejewski, "Recent progress and trends in predictive visual analytics," *Frontiers Comput. Sci.*, vol. 11, no. 2, pp. 192–207, 2017.
[40] Y. Ma, W. Chen, X. Ma, J. Xu, X. Huang, R. Maciejewski, and A. K. H. Tung, "EasySVM: A visual analysis approach for open-box support vector machines," *Comput. Vis. Media*, vol. 3, no. 2, pp. 161–175, 2017.

YUJIE WANG is currently pursuing the master's degree with the School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan, China. She was with the School of Information and Electrical Engineering, Ludong University, Yantai, China, from 2013 to 2017. She has been a Fellow Member with the Digital Media Technology Key Laboratory of Shandong Province, since 2017. Her current research interests include machine learning, time series analyzing, and data mining.

HUI LIU received the B.S., M.S., and Ph.D. degrees in computer science from Shandong University, Jinan, China, in 2001, 2004, and 2008, respectively.

She is currently a Professor with the Shandong Provincial Key Laboratory of Digital Media Technology, School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan. Her research interests include computer aided geometric design, multisource data mining, and artificial intelligence techniques.

QIANG GUO received the B.S. degree from the Shandong University of Technology, Zibo, China, in 2002, and the M.S. and Ph.D. degrees from Shanghai University, Shanghai, China, in 2005 and 2010, respectively. From 2012 to 2015, he was a Postdoctoral Fellow with Shandong University, Jinan, China. He is currently an Associate Professor with the School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan. His research interests include tensor low rank modeling of data, sparse representation, and object detection.

SHENXIANG XIE received the B.S. degree from the Anhui University of Finance and Economics, Bengbu, China, in 1999, and the Ph.D. degree from Nankai University, Tianjin, China, in 2010. He is currently a Professor with the School of Public Finance and Taxation, Shandong University of Finance and Economics, Jinan, China. His research interests include international economics and public finance.

XIAOFENG ZHANG received the B.S. and M.S. degrees from the Lanzhou University of Technology, Lanzhou, China, in 2000 and 2005, respectively, and the Ph.D. degree in computer science from Shandong University, Jinan, China, in 2014. He is currently an Associate Professor with the School of Computer Science and Technology, Shandong Technology and Business University, Yantai, China. His research interests include deep learning and pattern recognition.