Anticipating Stock Market of the Renowned Companies: A Knowledge Graph Approach

Yang Liu,1 Qingguo Zeng,2 Joaquín Ordieres Meré,1 and Huanrui Yang3

1Department of Industrial Engineering, Business Administration and Statistics, Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid, Madrid 28006, Spain
2School of Mathematical Sciences, South China Normal University, Guangzhou 510631, China
3Electrical and Computer Engineering Department, Duke University, Durham 27708, USA

Correspondence should be addressed to Yang Liu; yang.liu00@alumnos.upm.es

Received 26 March 2019; Accepted 18 July 2019; Published 7 August 2019

1. Introduction

Research on stock market of company prediction has become popular with the introduction of Fama’s efficient market hypothesis (EMH) [1], but in recent years, an increasing number of people have found that stock market of the renowned company changes is random, complex, and unstable given that they are affected by many factors. Stock market of company changes is affected by cultural aspects that fundamentally influence investor sentiment; traditional culture often determines people’s investment psychology, which in turn, affects the inflow and outflow of stock market capital. Company-related issues, such as the effects of company operations on stock prices and market factors [2]. Using these factors as bases for accurately predicting stock price movement gives investors hope that the maximum profits will be achieved with the least stock investment. Correspondingly, analyzing stock market movements is both challenging and attractive to researchers and investors. Moreover, as the development of knowledge graph in natural language processing (NLP), researchers in the financial field began paying attention to text mining in financial news. As maintained by EMH and artificial intelligence technique, the best results are achieved when news information is used to forecast stock market movement, which can also control financial risk in business activities.

In May 2012, Google formally announced its knowledge graph project, which is aimed at improving the search engines effectiveness and search quality as well as user experience with the engine [3]. The development of knowledge graph promoted the extensive use of artificial intelligence technology in smart search, smart question-and-answer tasks, and intelligent finance. In finance, knowledge graph is designed to find relationships amongst entities, such as the management...
of companies, news events, and user preferences. These entities can be used by investors to achieve efficient financial data-based decision-making and obtain business insights into predicting stock market [4]. For these reasons, the current research focused on how to use knowledge graph to improve the accuracy of stock price forecasts. Knowledge graph is a database that implements semantic searches by preserving relationships amongst multiple entities. Based on event tuples in knowledge graph [5], we can infer the definition of an event tuple as \((A, P, O)\), where \(A\) represents an agent, \(P\) denotes a predicate, and \(O\) represents an object [6]. In view of the objects converted by each event tuple are known, when they are converted to vectors, which loses more semantic information. Event tuples link relevant elements together and can be used as an efficient method of improving predictive accuracy. As a result, we propose a learning model of event tuples, which retains their semantic features at the maximum.

Knowledge graph embedding is a kind of representation learning in knowledge graph. Currently, several algorithms [3] explore the mapping relationship between entities and relationships in the translation distance model. The TransE model [7] is a computationally efficient predictive model that satisfactorily represents a one-to-one type of relationship. It has been used as a basis in the development of different translation distance models. The TransH model [8] maps the head and tail of a vector onto a hyperplane, after which these two components complete the translation process on the hyperplane. The TransR model [9], which is based on the TransE model, consists of entities and relationships that are located in different dimensions of space. For each relationship, the model defines a matrix that is used to convert entity vectors into space. The model is also dynamically determined by an entity relationship. The TransD model [10] is an improvement of the TransR model, which the former uses as a basis in considering a transformation matrix. In comparing these models, solving the head and tail of a news tuple for mapping in a single space necessitates the simple mapping of one-to-one entity extraction features. This functionality is found in the TransE model; this study is used to establish a feature combination model for news event metagroups.

This work selects six globally renowned companies to forecast their stock price movement, namely, Apple, Microsoft, Samsung, Boeing, Google, and Walmart as application scenarios. According to different data sources between Thomson Reuters and Cable News Network, we reveal the result of the case studies in some algorithms, which illustrates that the performance of combined feature outperforms that using only stock data, using a bag-of-words algorithm, and using convolutional neural network. Although there are already some examples of the powerful application of deep learning in NLP [11], such as speech recognition, text classification, and machine translation, most of the previous research in predicting stock price movement [12] is based on the semantic information news, which ignores the semantic features of structured events. The application of deep learning and knowledge graph on the renowned companies' stocks is rarely available. Thus, our work provides a viable application framework for financial markets, which can also be extended to other aspects of the financial field.

For stock market prediction, we formulated a knowledge graph embedding-driven approach that involves four main phases (Figure 1). The first is data retrieval, which begins with searching keywords such as "Apple" or "Google" over the Thomson Reuters or Cable News Network (CNN), as shown in Figure 2. The yellow part is the headline of financial news and the orange part is the release time of the financial news. A web crawler is then used to obtain financial news from the sites and match them with corresponding stock data, combining all this information into one corpus. The second phase involves preprocessing, which encompasses corpus analysis, text normalization, word tokenization, label tagging, and word-to-vector transformation. The features derived from word embeddings and stock data layers are then selected for the computation of eigenvalues, after which a feature vector is generated using machine learning and knowledge graph embedding. The third phase is model creation, in which stock market prediction labels (increase or decrease) are assigned to financial news to training a classification model. Finally, the finance decision relies on the predictive performance of this framework. Specifically, a small improvement directly affects decision-making, which increases the renowned company's profits. The fourth phase involves model evaluation, wherein the results and the extraction of conclusions of each machine learning model are analyzed.

The remaining part of the paper is organized as follows. Section 2 is a review of existing studies on the use of machine learning in stock market prediction. Section 3 introduces the methodology adopted in this study, including the data description and feature selection approach. Section 4 presents the classification results, and Section 5 discusses the results and implications. Section 6 concludes with the summary.

2. Literature Review

In Table 1, the application of machine learning techniques to stock market prediction is an emerging research field. The advantage of machine learning models is their ability to facilitate the processing of large amounts of data. A procedure common to these models is the concatenation of features from different sources into a feature vector. For most machine learning models used, research is focused on predicting stock trends (increase or decrease).

Various feature extractions have already been proposed to predict the stock price of the renowned companies. In the early years, most of the company's stock prediction depends on the empirical analysis of the econometric model, namely, the feature extracted is the raw data of the company's stock. However, the previous work ignored the potential impact of unstructured data on company stocks. Since the bag-of-words is widely utilized in the task of document classification, the frequency of word appearance can be used as a feature of the classification. Hence, the bags of words model only calculate the frequency of words and do not consider the word order or word sparsity in the context, which directly impacts the prediction result. Furthermore, EMH had found that the emotional impulses of the renowned company investors often observe abnormal fluctuations in company stocks.
**Figure 1:** Research workflow for machine learning and knowledge graph.

**Figure 2:** Example of collected financial news in CNN.
Tetlock [26] adopted the popular news from the Wall Street Journal to find that news sentiment has predictive power on company stocks. Chen et al. [27] found information comments have a significant influence on the stock return through sentiment analysis. Furthermore, the investors' sentiment trends after the positive news, which results in a buying trend, and higher stock market prices while after negative news stocks are sold result in a decrease of the price. However, sentiment analysis can only be employed in specific texts. If sentiment is implicit rather than direct emotional words, then the ability of using sentiment analysis for predicting the stock price of the renowned company is relatively limited. Mittermayer and Knolmayer [16] illustrated that news-CATS achieves a performance that is superior to that of other ATC prototypes used to forecast stock price trends. Li et al. [21] proposed a media-aware quantitative trading strategy by using sentiment information on web media. The simulation trading return was up to 166.11%. Nguyen et al. [22] proposed a feature topic sentiment to improve the performance of stock market prediction. Their method achieved an accuracy that is 9.83% better than that of the historical price method and 3.03% better than that of the human sentiment method. We utilize the characteristics of syntax analysis as being proposed in [6, 12]; namely, a structured tuple is extracted from an unstructured text based on the semantic structure of each piece of news. Knowledge graph can enrich the structured representation of the news event and effectively retain feature vectors for the news event. The main feature extraction in the previous studies [28, 29] is sentiment analysis, which neglected the event characteristics in the text. Furthermore, the existing literature [23, 29] had proved the positive effect of technical indicators on stock market prediction. In summary, our research highlights syntax analysis in financial news, which also incorporates with other features extraction (stock data, technical indicators, and bag-of-words). Because of the variety of considered features, this research will deliver an improved prediction of the stock market value for renowned companies in at least 3.6%.

Previous research [30] was applied with the traditional machine learning algorithms, especially, since the powerful application ability of deep learning based on machine learning. Deep learning is utilized in several studies for predicting stock price movement. Kraus and Feuerriegel [31] forecasted the stock returns based on financial disclosures,
Table 2: Quotes and renowned company names.

<table>
<thead>
<tr>
<th>Financial news dataset</th>
<th>Stocks</th>
<th>Company names</th>
<th>Time interval</th>
<th>Number of news headlines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thomson Reuters</td>
<td>AAAL</td>
<td>Apple Inc.</td>
<td>3.10.2011 - 31.7.2017</td>
<td>6423</td>
</tr>
<tr>
<td></td>
<td>MSFT</td>
<td>Microsoft Corporation</td>
<td>30.9.2011 - 29.11.2017</td>
<td>6623</td>
</tr>
<tr>
<td></td>
<td>SSNLF</td>
<td>Samsung Electronics Co., Ltd.</td>
<td>30.9.2011 - 29.11.2017</td>
<td>6900</td>
</tr>
<tr>
<td>CNN</td>
<td>BA</td>
<td>The Boeing Company</td>
<td>8.9.1998 - 23.2.2017</td>
<td>2999</td>
</tr>
<tr>
<td></td>
<td>GOOG</td>
<td>Google Inc.</td>
<td>25.5.2006 - 24.2.2017</td>
<td>5278</td>
</tr>
<tr>
<td></td>
<td>WMT</td>
<td>Walmart Inc.</td>
<td>7.5.1997 - 24.2.2017</td>
<td>4913</td>
</tr>
</tbody>
</table>

Table 3: Samples extracted from the dataset.

<table>
<thead>
<tr>
<th>Date</th>
<th>News headline</th>
<th>News headline event tuple</th>
</tr>
</thead>
<tbody>
<tr>
<td>3/18/2012</td>
<td>Chinese writers' group sues Apple: state media</td>
<td>Chinese writers' group, sues, Apple</td>
</tr>
<tr>
<td>4/13/2012</td>
<td>German court upholds ban on Apple push email</td>
<td>German court, upholds ban on, Apple push email</td>
</tr>
<tr>
<td>4/4/2012</td>
<td>Apple, two publishers hold out against settlement-report</td>
<td>Apple- two publishers, hold out against, settlement-report</td>
</tr>
<tr>
<td>3/26/2012</td>
<td>Apple CEO Cook meets Chinese officials</td>
<td>Apple CEO Cook, meets, Chinese officials</td>
</tr>
<tr>
<td>4/25/2012</td>
<td>US STOCKS-Apple results put Nasdaq on track for best gain of year</td>
<td>Apple, results, Nasdaq on track for best gain</td>
</tr>
</tbody>
</table>

and their result demonstrated that a higher directional of deep learning surpasses traditional machine learning. Ding et al. [5] illustrated that deep learning also can forecast stock market of the renowned company. Sim et al. [32] proposed that technical indicators transform into images of the time series graph, which examines the applicability of deep learning in the stock market. Overall, we adopt the multiple models for prediction stock price of the renowned company, which proves the reliability of our proposed model as comparing different algorithms. This work applies deep learning incorporated with knowledge graph embedding for feature extraction, which examines the applicability of combined features methods in the renowned company stock price movement.

3. Materials and Methods

We developed a knowledge graph-based approach that consists of three steps, namely, data description, data preprocessing, and feature selection.

3.1. Dataset Description. Table 2 shows the custom financial news corpus built with headlines from two datasets. The first dataset contains news articles published by Thomson Reuters, including those regarding Apple Inc. (AAPL), Microsoft Corporation (MSFT), and Samsung Electronics Co., Ltd. (SSNLF). The second dataset comprises news reports published in CNN, including reports on the Boeing Company (BA), Google Inc. (GOOG), and Walmart Inc. (WMT). It also consists of financial news headlines published at specific time intervals, with each news report accompanied by a title and a release date. Titles are used for event embedding and feature extraction, and release dates are used as a reference in ensuring alignment between corresponding financial news and trading data from a time series. As shown in previous work [5, 31], using a headline to build corpora can help reduce noise in text mining as headlines concisely represent the content of a text. We used only the news headlines from Reuters and CNN for the prediction of stock price movement.

Daily stock data from index report in each company are collected by Yahoo Finance in the same period during stock data and financial news headlines. Daily trading data, which are common predictors of stock price [23, 33], and technical indicator features were used in our model. There are opening price, closing price, high price, low price, and volume and three technical indicators.

Table 3 shows some examples of filtered financial news headlines. To illustrate, 6423 headlines regarding Apple Inc. were extracted and then reduced to 2799 headlines after filtering via Reverb [34]. Let us take the article “What is next for Apple’s board?”, published on 6 October 2011, as a specific example. The title of the article cannot be transformed into an event tuple using Reverb. Because the sentence is in the interrogative form, there is no event tuple that constitutes a rule. After the matching of time-wise data and stock data, 941 headlines were left. Daily news and stock data were aligned to create input-output pairs, except the days when no news was released. In 9 January 2012, for instance, three news articles were reported, but we chose only one headline for alignment with stock market data. News events possibly happen several times within one day, but they do not happen every day, unlike stock trading, which happens daily, except on nontrading days falling on weekends or holidays.

Table 4 shows that the matches found are pairs between event tuples and stock data. From this complete dataset, we used 80% of the samples in training data and the remaining 20% for testing data. This selection method is the same as the previous literature [5, 24].
Table 4: Training and testing datasets.

<table>
<thead>
<tr>
<th>Data source</th>
<th>Company</th>
<th>Training</th>
<th>Testing</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple</td>
<td>780</td>
<td>161</td>
<td></td>
<td>941</td>
</tr>
<tr>
<td>Reuters</td>
<td>832</td>
<td>208</td>
<td></td>
<td>1040</td>
</tr>
<tr>
<td>Samsung</td>
<td>855</td>
<td>214</td>
<td></td>
<td>1069</td>
</tr>
<tr>
<td>CNN</td>
<td>1010</td>
<td>252</td>
<td></td>
<td>1262</td>
</tr>
<tr>
<td>Google</td>
<td>798</td>
<td>200</td>
<td></td>
<td>998</td>
</tr>
<tr>
<td>Walmart</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Number of event characteristic labels for each company.

<table>
<thead>
<tr>
<th>Label</th>
<th>Companies</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>502</td>
</tr>
<tr>
<td>1</td>
<td>537</td>
</tr>
<tr>
<td>2</td>
<td>661</td>
</tr>
<tr>
<td>3</td>
<td>692</td>
</tr>
<tr>
<td>4</td>
<td>407</td>
</tr>
<tr>
<td>Total</td>
<td>2799</td>
</tr>
</tbody>
</table>

Table 6: Features of the prediction model.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Feature expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Stock price yesterday + yesterday’s trend (in Section 3.3.1)</td>
</tr>
<tr>
<td>2</td>
<td>Same + bag-of-words (in Section 3.3.2)</td>
</tr>
<tr>
<td>3</td>
<td>Same + some features inferred through a convolutional neural network (in Section 3.3.3)</td>
</tr>
<tr>
<td>4</td>
<td>Same + some features inferred through feature combination (in Section 3.3.4)</td>
</tr>
</tbody>
</table>

3.2. Data Preprocessing. The following three steps are for preprocessing, which prepares for feature extraction and model creation.

(1) Tagging the label for each news headline. Five possible labels are characterized by categorical values as follows: 0 for an extremely negative label, 1 for a negative label, 2 to signify a neutral label, 3 for a positive label, and 4 to denote an extremely positive label. According to the time characteristics of each news headline, an event is manually tagged with the label for each news headline. Table 5 shows the labeling applied for each company: Label 0 means that a company’s competitor has happened in this event; label 1 means that the company lost something; label 2 means that it did not cause any impact on the company; label 3 means that this event enabled the company to obtain something; and label 4 means that the company increased its profits or created more value.

(2) Word vector transformation. We used the word2vec [35] algorithm to train word embedding and set the number of dimensions to 300. The word embedding was also trained using the Google News dataset, which contains 100 billion words that are characterized by a continuous bag-of-words structure.

(3) Technical indicators calculation. Three additional technical indicators that are calculated on the basis of daily trading data were used as follows:

(a) Stochastic oscillator (%K). This indicator is a momentum analysis method created by George C. Lane. When the price trend rises, the closing price tends to approach the highest price of the day. When the price trend declines, the closing price tends to approach the lowest price of the day [36].

(b) Larry Williams %R indicator. This indicator is an oscillation indicator that measures the ratio of the highest price to the daily closing price. It indicates the proportion of stock price fluctuations in a certain period, thereby providing a signal out of the reversal of a stock market trend [37].

(c) Relative strength index (RSI). Buying and selling intentions in the market are analyzed by comparing the closing prices in a given period. Stocks that have had more or stronger positive changes have a higher RSI than do those that have had more or stronger negative changes. Strength indicators fall between 0 and 100; investors sell if this value is ≥ 80 and buy if it is ≤ 20 [23, 36].

3.3. Variable/Feature Selection. To assess the effectiveness of applying the prediction model on the basis of financial news, we designed four sets of features for predicting stock price movement (Table 6). Features 3 and 4 are used for event characteristics. Each feature is explained in the succeeding subsections. The target output consists of a binary variable, for which a value of 1 indicates that the closing price at day $t + 1$ will be higher than that at day $t$, and a value of 0 indicates that the closing price at day $t + 1$ will be lower than that at day $t$. 
3.3.1. Stock Data Only. We considered historical price as the input for predicting stock price movement and used it as a baseline for comparison with other sets of features. The features used to train the machine learning model that uses only stock data are \( \text{price}_{t-1} \) and \( \text{indicator}_{t-1} \). The output is the indicator of price movement (increase or decrease) at each transaction date examined.

3.3.2. Bag of Words. The input feature set is based on a bag of words for the news and price trend for stock. Previous studies [18, 38, 39] have widely used and confirmed the feasibility of the bag-of-words algorithm, but this method disregards elements such as grammar and word order in the text. In the present study, we first prepared each headline for data preprocessing and then transformed the preprocessed headline into a feature vector using the term frequency-inverse document frequency (TF-IDF) algorithm [40], which assigns high weight to eigenvectors. The studies [41, 42] have strongly proven the effectiveness of the TF-IDF algorithm in feature extraction from news headlines. It estimates the frequency of a term in one document over the maximum in a collection of documents and assesses the importance of a word in one set of documents. Such importance increases proportionally with the number of word appearances in a document. The features used to train the bag-of-words machine learning model are \( \text{price}_{t-1} \), \( \text{indicator}_{t-1} \), and \( \text{tf\_idf}_{t-1} \), which are the price movements (increase or decrease) at each transaction date examined.

3.3.3. Convolutional Neural Network. Given the sequence of words in a headline, the word2vec model [35] can be used to embed each of these words in a real valued-vector \( x \). In this work, we set the dimension of each word vector at 30 (i.e., \( x \in \mathbb{R}^{30} \)). We concatenated the word vectors of all the words in a headline sequentially to form a matrix \( X = [x^{(1)}, x^{(2)}, \ldots, x^{(N)}] \) [35] as the input to a convolutional neural network model. For a headline with \( N \) words, the resultant input matrix has dimensions of \( 30 \times N \), and the dimension of news representation also is 30.

In Figure 3, this convolutional neural network model is made up of four consecutive layers: the first layer is the input, the second layer is convolutional, the third layer is a max-pooling, and the last layer is a fully connected layer. The convolutional and max-pooling layers were designed using the text-attentional convolutional neural network [43], which effectively carries out sentiment classification. In the convolutional layer, input matrix \( X \) convolves with a kernel \( w \in \mathbb{R}^{n \times k} \), where \( n \) is the size of a word vector (30 in our work), and its dimension is 50. \( k \) denotes the size of a sliding window (\( k=3 \) in this study). The computation can be formulated as follows:

\[
Z_i = \sigma \left( wX_{i:i+k-1} + b \right),
\]

where \( X_{i:i+k-1} = [x_i, x_{i+1}, \ldots, x_{i+k-1}] \) is the portion of input \( X \) falling within the sliding window, \( b \) denotes the optional offset, and \( \sigma(\cdot) \) is the sigmoid function.

In the next step, we used the pooling layer to reduce the convolutional neural networks parameter space and consequently minimize information loss in the pooling processing and capture the most important features. The eigenvector of the filter in the pooling layer is cut into 3 chunks and a maximum value is taken in each chunk; we obtained 3 eigenvalues. Convolutional output vector \( Z \) was split into \( p \) windows, and only the maximum feature as 3 in each window was kept for passage onto the final fully connected layer. The fully connected layer is linear regression, and the output layer is the feature classification between 0 and 1.

Our convolutional neural network model is intended to extract a feature vector from the fully connected layer to describe the emotional characteristics of input headlines. Following the intuition that a good feature vector should lead to the accurate classification of the event characteristics of headlines, we attached a softmax layer after the fully
connected layer when the convolutional neural network model was trained. The entire model was trained to classify the five emotional labels that describe event characteristics, and the resultant model will be able to provide a meaningful emotional feature vector for each inputted headline. Meanwhile, we define the loss function of the convolutional network using

\[ E_s = \|h + r - t\|^2_2, \]

where \(h, r,\) and \(t\) represent head entity, relationship, and tail entity in event tuple, respectively [47].

And text representation \(E_d\) is consistent with Section 3.3.3, which denotes:

\[ E_d = f_s(h_i, t_d) + f_s(h_d, t_s) + f_s(h_d, t_d), \]

where \(f_s(h_d, t_d) = \|h + r - t\|^2_2\) represents the head entity and the tail entity is in text representation. \(f_s(h_d, t_s) = \|h + r - t\|^2_2\)

and \(f_s(h_i, t_d) = \|h + r - t\|^2_2\) represent one of the head entity \(h\) and the tail entity \(r\) in text representation, another one is in structure representation.

Moreover, we combined two types of representation learning (convolutional neural network and feature combination) to map news titles; these vectors will be into feature vectors. The relationship vector \(R\) identical to the result of the feature extraction layer of the convolutional neural network in the structural model. Besides, we combined this loss with the classification loss of the convolutional neural network. For instance, the sentence is "Samsung sues Apple for infringement", so "Samsung + Apple ≈ sues". The loss function \(E_s\) of this structural model is defined as follows:

\[ E_s = \|h + r - t\|^2_2, \]

where \(h, r,\) and \(t\) represent a word, an entity, and a relation, respectively. And \(r\) are the mapping matrices for entities in the structural model and represents the weights of the convolutional neural network.

In Figure 4, knowledge graph contains rich semantics in entity description texts, but it is not fully utilized in feature extraction. Most of the existing text representation models simply train the text into word vectors through word2vec, which obtains the text representation by means of averaging, etc. Hence, these ways often lose more semantic information.

Therefore proposed to extract feature vectors from news texts using a convolutional neural network model combined with the TransE model, which also fully integrates the two parts of the feature information.

In Figure 5, this architecture used in the feature combination model encompasses two parts, namely, a convolutional neural network (Section 3.3.3) and the TransE model. The feature combination sets are the average of each word vector in an entity and it was obtained using the word2vec model. The two entity vectors are mapped into the same relational space, and these entities using a trained low-rank weight matrix [47].

As assumed in the TransE model, relationship vector \(R\) should satisfy "\(E + R = T\)". The model can be represented as a parameter set \(\theta = (L_r, R, \theta)\), where \(X, E,\) and \(R\) represent a word, an entity, and a relation, respectively. And \(r\) are the mapping matrices for entities in the structural model and represents the weights of the convolutional neural network. For instance, the sentence is "Samsung sues Apple for infringement", so "Samsung + Apple ≈ sues". The loss function \(E_s\) of this structural model is defined as follows:

\[ E_s = \|h + r - t\|^2_2, \]

where \(h, r,\) and \(t\) represent head entity, relationship, and tail entity in event tuple, respectively [47].
where the head entity and tail entity are randomly replaced by an entity or relationship in another tuple. In particular, if the replaced tuple is already in \( T \), it would not be added to the negative sample. Since both \( h \) and \( t \) represent two types of entities, there are structure-based representations and text-based representations in interval-based loss functions. The stochastic gradient descent (SGD) is employed to minimize the above loss function.

In this structure, we choose the optimal parameters as follows: the learning rate of SGD \( \lambda = 0.001 \), the dimension of the representation vector of entities and relationships \( k = 100 \), the word vector dimension of the entity description text \( n = 100 \), and batch size during training is 1440. Convolution layer window size is \( W \in \{1, 2, 3\} \). This experiment is performed in 1000 iterations of training and the optimal parameters are based on the testing set.

Hence, the features used for training the machine learning model are \( price_{t-1}, t_{\text{indicator}}_{t-1}, \) and \( f_{\text{combine}}_{t-1} \), which are the price movement (increase or decrease) at corresponding transaction date.

### 4. Experiments and Results

#### 4.1. Experimental Setting

This section compares the performance of different models in feature extraction from financial news expended with full sets of features. The evaluation was intended to obtain evidence that the proposed feature combination model is superior to other feature selection models in predicting the stock price movement. As shown in Table 7, we chose linear models, for instance, logistical regression and naive Bayes. Others are nonlinear models, for instance, the ensemble learning (random forest, adaboost, and gradient boosting) for the comparison. We constantly adjusted the parameters in the grid search and selected the optimal parameter value; their parameter value has proven to work well on machine learning method [29, 48].

We used a computer consisting of an Intel Core i5 processor with four cores running at 2.9 GHz and 8 GB RAM under the MacOS platform. We used the Scikit-learn library in Python in the experiments involving traditional machine learning algorithms and TensorFlow 1.4 in the
Table 7: List of machine learning methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Classification</th>
<th>Parameter and their value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional machine learning</td>
<td>Decision tree (DT)</td>
<td>Criterion= gini, splitter= best, max depth=None, min samples split=2, min samples leaf=1, min weight fraction leaf=0.0,</td>
</tr>
<tr>
<td></td>
<td>Logistical regression (LR)</td>
<td>Random state=1</td>
</tr>
<tr>
<td></td>
<td>Naive Bayes (NB)</td>
<td>No kernel estimator</td>
</tr>
<tr>
<td></td>
<td>Stochastic gradient descent (SGD)</td>
<td>Loss=hinge, penalty=12</td>
</tr>
<tr>
<td></td>
<td>Support vector machine (SVM)</td>
<td>Polynomial kernel function with exponent = {1, 2}, RBF kernel function with gamma = 0.01</td>
</tr>
<tr>
<td>Ensemble learning</td>
<td>Random forest (RF)</td>
<td>Randomly sampled as candidates at each split = log2 + 1, max-depth=2, random state=0</td>
</tr>
<tr>
<td></td>
<td>AdaBoost (AB)</td>
<td>Polynomial kernel function with exponent = {1, 2}, RBF kernel function with gamma = 0.01, n-estimators=100</td>
</tr>
<tr>
<td></td>
<td>Gradient boosting (GB)</td>
<td>N-estimators =100, learning rate =1.0, max-depth=1, random state=0</td>
</tr>
</tbody>
</table>

Table 8: Abbreviations of the results for each algorithm considered.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Stock data (SD)</th>
<th>Bag of words (BOW)</th>
<th>Convolutional neural network</th>
<th>Feature combination model (FC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT</td>
<td>DT_1</td>
<td>DT_2</td>
<td>DT_3</td>
<td>DT_4</td>
</tr>
<tr>
<td>LR</td>
<td>LR_1</td>
<td>LR_2</td>
<td>LR_3</td>
<td>LR_4</td>
</tr>
<tr>
<td>NB</td>
<td>NB_1</td>
<td>NB_2</td>
<td>NB_3</td>
<td>NB_4</td>
</tr>
<tr>
<td>SGD</td>
<td>SGD_1</td>
<td>SGD_2</td>
<td>SGD_3</td>
<td>SGD_4</td>
</tr>
<tr>
<td>SVM</td>
<td>SVM_1</td>
<td>SVM_2</td>
<td>SVM_3</td>
<td>SVM_4</td>
</tr>
<tr>
<td>RF</td>
<td>RF_1</td>
<td>RF_2</td>
<td>RF_3</td>
<td>RF_4</td>
</tr>
<tr>
<td>AB</td>
<td>AB_1</td>
<td>AB_2</td>
<td>AB_3</td>
<td>AB_4</td>
</tr>
<tr>
<td>GB</td>
<td>GB_1</td>
<td>GB_2</td>
<td>GB_3</td>
<td>GB_4</td>
</tr>
</tbody>
</table>

Experiments involving deep learning and the TransE model. The abbreviations used for comparing classification performance are presented in Table 8. During testing, 2-fold cross-validation is applied to evaluate the stability of the models. And we compared performance in predicting stock price movement for the next day with a test dataset and evaluated the performance of the models in terms of accuracy and F1-score [49].

4.2. Results

4.2.1. Thomson Reuters Case Study. Thomson Reuters is a multinational mass media and information firm. Three renowned companies on the website were selected for the analysis. The first is Apple Inc., which is an American multinational technology company that designs, develops, and sells consumer electronics, computer software, and online services. The second company selected is Microsoft, which is an American multinational technology company that develops, manufactures, supports, and sells computer software, personal computers, and other services. The last company chosen is Samsung, which is a South Korean multinational electronics company. We considered data from these three typical technology companies for our problem. Measurements of average accuracy are shown in Table 9, and the results for each renowned company with respect to prediction based on different features are illustrated in Figure 6. We calculated the average of the four feature construction approaches for each model for comparison amongst the three companies. The proposed feature combination model generated the best results, achieving average accuracy levels of 61.63%, 59.18%, and 58.48% for Apple, Microsoft, and Samsung, respectively. These figures are consistent with the rates reported in previous research. However, many studies analyzed only one company or used only one algorithm. The current research probed into three companies on the basis of information from a common data source to build a prediction model with different feature selection functionalities and different algorithms. We used stock data, a bag-of-words algorithm, a convolutional neural network, and feature combination, together with eight algorithms. As indicated in Table 9, the proposed prediction model achieved 73.68% data extraction with the use of LR_4 for Apple and 67.78% data extraction with the use of SVM_4 for Microsoft. In particular, the LR algorithm used for Apple achieved accuracy and F1-score of 0.7326 and 0.7360, respectively, which highlights its powerful capability in a two-class classification.
To assess the effectiveness of our research, we compared the feature combination model that uses event tuples with the stock data approach and the bag-of-words algorithm. The results showed that the average accuracy of the event tuple-based model is better than that of the stock data approach and the bag-of-words algorithm by 10.01% and 10.87%, respectively. We also compared feature extraction using the stock data approach, the bag-of-words algorithm, and machine learning. The use of deep learning improved prediction accuracy by 5.25% and 6.11% over the levels achieved with stock data and bag of words, respectively. Therefore, we can conclude that using the proposed feature combination model and deep learning in feature extraction helps improve the accuracy with which stock price movement is predicted. These results also fully prove the effectiveness of embedding layer in feature extraction accuracy improvement.

4.2.2. CNN Case Study. CNN is an American basic cable and satellite television news channel owned by the Turner Broadcasting System. Three renowned companies on which these website reports were selected for the analysis. The first is Boeing, which is an American multinational corporation that designs, manufactures, and sells airplanes, rotorcraft, rockets, and satellites worldwide. The second company is Google, which is an American multinational technology
The results on average accuracy are shown in Table 10 and Figure 7. These findings are very similar to those derived in the previous case study, and their comparison confirmed that the proposed feature combination model can outperform other feature selection models in stock price movement prediction. With regard to feature selection underlain by the bag-of-words algorithm, the CNN case study generated more robust results than did the Reuters case study. The average accuracy of the bag-of-words algorithm was lower than that of the stock data approach in the Reuters case study. In the CNN case study, the average accuracy levels of the proposed feature combination model were 57.94%, 58.79%, and 57.67% for Boeing, Google, and Walmart, respectively. In the case study on Reuters, an average accuracy that exceeds 60% was not achieved for any of the companies, illustrating that differences in data source directly affect stock price movement.

We chose a combination of deep learning and knowledge graph to build our feature selection model because this combination exhibited superior prediction performance in the comparison experiments involving other feature selection models.
5. Discussion

5.1. Research Findings. We attempted to compare our prediction results with those made in previous studies wherein prediction was also based on the financial news of the renowned companies. However, the findings are incomparable because different studies use different datasets or algorithms, and these methods are difficult to investigate using financial news headlines. In the current research, we refrained from evaluating performance on one company dataset, this decision that enabled our knowledge graph method to exceed baseline predictions based only on the price by up to 3.6%. This also proves that feature extraction in deep learning is more effective than traditional machines. The combination of deep learning and knowledge graph fully integrates the semantic information in financial news, which effectively predicts the stock price movement of the renowned company.

This work demonstrated the application of deep learning and knowledge graph in finance. To the best of our knowledge, knowledge graph has been rarely applied in stock prediction. Because minimal financial training sets in knowledge graph are available, financial knowledge extraction is the main task for the organizational construction of a knowledge graph. Such extraction is critical for the understanding and processing of deep semantics in the event tuple, which also directly influences the feature extraction of financial news.

5.2. Implications for Business Activities. The utility of a feature combination model based on knowledge graph is not limited to financial analysis. Currently, knowledge graph data are available for medical diagnosis, speech recognition, precise marketing, and financial risk control [4]. Our model could also be applied in these areas.

Antifraud activities are an important part of finance. Applying our model on the basis of knowledge graph on customers helps organize all relevant knowledge fragments through deep semantic analysis and reasoning, which can be verified with a customer’s bank information. Moreover, customers typically use keywords to search for products, and knowledge graph can provide relevant information to a customer. If a complete knowledge system of users is described and collected, a system would better understand and analyze user behaviors.

5.3. Limitations and Future Work. A deep learning model uses supervised learning and needs a dataset with sufficient labels, but our datasets do not work well with deep learning as a small number of financial news articles match stock data. At present, knowledge graph embedding inevitably produces a loss in news semantics, either because of a learning principle or the tokenization of knowledge representation for vectorization. Thus, continuous representation in a knowledge graph is still a huge challenge. We attempted to apply regression using the above-mentioned algorithms, but the results indicated poor performance. More specifically, the results for Apple validated our expectations.

The application of large-scale knowledge graph is still relatively limited, and knowledge graph for smart search, smart question-and-answer tasks, social media, and other areas are in their initial stages, with considerable room for improvement. The following advantages of knowledge graph should be considered: (a) the effective organization and expression of semistructured data, (b) knowledge reasoning, and (c) the expansion of cognitive ability for incorporating deep learning. Traditional knowledge elements (entities, relationship, attributes), extraction technologies, and methods have achieved good results in limited areas, but because of numerous constraints and poor scalability, knowledge graph is not fully functional for financial forecasting because of numerous constraints and poor scalability.

6. Conclusions

Stock movement prediction of the renowned company is a daunting task because stock prices are affected by many factors. This research presented a novel method for integrating knowledge graph embedding with stock market prediction. The contributions of this study can be summarized as follows. First, we developed a novel feature combination model that constructs a feature mapping vector for each tuple–news pair by simultaneously considering the diversity of entities and relationships. Second, the feature combination model was successfully applied to different types of companies and datasets and exhibited good performance in classification tasks.

Stock market prediction of the renowned company grounded in knowledge graph is an interesting topic for business activities. Given that the study of knowledge graph for feature set is still in its infancy, we expect it to be applied in a wide range of academic research. More companies will also earn profits and create more opportunities through the use of knowledge graph in the feature set.

Data Availability

The data used in this study can be accessed via https://github.com/linechany/knowledge-graph.

Conflicts of Interest

The author declares that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the China Scholarship Council (CSC) throughout the grant number: 201508390019. The
authors also want to thank the research program RFCS from the EU, which partially supported this research through the research project AutoSurveillance, with project ID 847202.

References


Submit your manuscripts at
www.hindawi.com