Universiteit Leiden

Master Computer Science

Predicting Stock Price Movement with Multi-source Information

Name:	Keyan Jin
Student ID:	s2596261
Date:	[26/08/2021]
Specialisation:	Data Science
1st supervisor:	Suzan Verberne
2nd supervisor:	Jian Wang

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

Stock price movement prediction aims to predict the future trend of stocks so as to help investors to make correct investment decisions. Since the stock market is an unstable and highly dynamic market, which will be affected by many factors. Hence, an efficient fusion of multi-source information from different sources on stock price trend prediction is a promising direction. However, previous research mainly treats each stock as an independent research object to make predictions, As an valuable signal, the relationship between companies was not considered.

In this thesis, we first extract structured news headlines from raw texts, and compared the performance of different text representation methods for obtaining the semantic information of financial news headlines. Then, we propose a framework to integrate financial news, historical stock data, company relationship knowledge graph on real-world stock market to compare the influencing indicator of various source data. Finally, we did some post analysis to visualize the performance of different text representation methods.

Experimental results show that multi-source information is effective for stock movement prediction task and BERT achieves the best results for the representation of news headlines information.

Contents

Chapter 1

Introduction

1.1 Background

Stock as an investment method was born in the Netherlands more than 400 years ago. It is central to the global economy. Companies raise funds dispersed in society by issuing stocks to expand their production and operation scale in the stock market. Investors obtain partial ownership of the company by purchasing stocks and share risks and profits with the company. With the rapid economic development, stocks have gradually become an essential tool for corporate financing and personal financial management. Changes in stock prices directly affect the stability of the financial market. Hence, accurately predicting the trend of stock prices can effectively avoid investment risks and increase wealth.

News reported on social media will affect people's financial decision-making [1], and people's decision-making will affect their individual trading behavior [2], which will eventually lead to fluctuations in the financial market. For example, On June 16, 2021, the sports section of Reuters under the title "Drink water! -Ronaldo removes Coca-Cola bottles at Euro press conference"[1] reported that the famous football star Cristiano Ronaldo moved two bottles of Coca-Cola aside and indicating that he only drinks water. Then the stock price of Coca-Cola dropped from $56.10 to $55.22 after this reported movement. On the next trading day, Coca-Cola's stock price even reached a lower

[1] https://www.reuters.com/lifestyle/sports/drink-water-ronaldo-removes-coca-cola-bottles-euro-press-conference-2021-06-15/

price at $54.95. Therefore, news and content in social media have proven to directly or indirectly affect financial market fluctuations [3, 4].

In addition, since the stock market is composed of many companies, the prosperous relationship between stocks may contain many valuable clues in stock prediction. For example, In 2020, the stock price of the new energy sector has risen fiercely. Tesla and Nio are both in the new energy vehicle sector. At the end of 2020, Tesla has risen by seven times, and Nio has risen by more than eleven times compared with their stock price at the beginning of 2020. Therefore, integrating the relationships between companies should also be of great help to the stock trend prediction task.

1.2 Problem statement

In most cases, the methods used by stock traders and scholars to analyze stock price trends are divided into two categories: fundamental analysis and technical analysis. Fundamental analysis, first of all, is to understand the company's corporate financial data. In addition, it is necessary to understand external information such as the macroeconomic and microeconomics, the prospects of related industries, and the status of relevant companies in the industry. Technical analysis uses historical stock market data to study the past and current behavior of the market to predict future price trends. The technical indicators are mainly composed of data such as stock price, trading volume, or index. Technical analysis only cares about the changes in the stock itself.

In the view of data analyzed by these two methods, fundamental analysis mainly uses textual information, while technical analysis mainly deals with numerical information. Traditional stock market prediction tasks were usually use single-source information such like stock historical data or textual information of related companies [5, 6]. However, it has been proved many factors in the stock market will impact the stock movement, such as stock historical data, national laws and policies, news and public opinion, the company's operating status [7, 8]. Hence, it is not difficult to imagine fusing information from multiple sources can offer advantages in stock prediction tasks [9].

In our study, we will compare the performance of different text representation methods to capture news semantics. In addition, we will propose a solution to integrating historical price data, knowledge graph of company relationships, and financial news related to

corresponding companies on stock prediction task. Specifically, the task can be modeled as follows:

As we all know, every trading day in the stock market will generate a large amount of transaction data, including opening price, highest price, lowest price, closing price, trading volume and so on. The closing price is the transaction price at the last moment of a trading day. If no transaction occurs on that day, the closing price of the previous trading day will be used. Looking at it continuously, on the day t, for a specific stock i, the closing price within a period is a sequences data. The closing price sequence in the past T days can be recorded as:

$$P_{i,t} = (p_{i,t-T}, ...p_{i,t-2}, p_{i,t-1})$$

In addition to the historical price of stocks, the news text related to the stock market is also an important influencing factor. Similarly, we regard the news text of stock i in T days as a text sequence, which is recorded as:

$$D_{i,t} = (d_{i,t-T}, ...d_{i,t-2}, d_{i,t-1})$$

For any stock i on the t-th trading day, the fluctuation of the closing price and the opening price of the day is used as the method to determine the rise and fall of the stock. It is defined that on the t-th trading day, the rise and fall of the stock i is recorded as $y_{i,t}$, and when the closing price is higher than the closing price of the previous trading day, $y_{i,t}$ is 1, and when the closing price falls or remains the same as the closing price of the previous trading day, $y_{i,t}$ is 0.

$$y_{i,t} = \begin{cases} 1, if p_{i,t} > p_{i,t-1} \\ 0, if p_{i,t} > p_{i,t-1} \end{cases}$$

Hence, our task is for a certain stock i, giving the stock closing price sequence and news sequence in the past T days, and predicting the value of the stock price $y_{i,t}$ on the t day as 0 or 1.

1.3 Research questions

In this thesis, we conduct experiments to answer the following research questions:

- How is the utility of different sentence-level text representation methods in stock prediction tasks? Which method is a better solution for obtaining semantic information in news headlines?
- Do multi-source data like historical transaction data, company relations enhance the performance of solution compared to single source information in stock prediction tasks? Which indicator improves the solution the most?

1.4 Main contributions

In response to the previously mentioned questions, the key contribution of our thesis can be summarized as follows.

- We constructed two news-labeled datasets, including financial news data and a company relationship knowledge graph.
- We compared the performance of different text representation methods for obtaining the semantic information of news headlines.
- We propose a model to integrate three stock price indicators (relevant financial news, historical stock data, company relationship knowledge graph) to predict stock price movement.
- We evaluated the effectiveness of adding a knowledge graph with real-world stock data from the NYSE/NASDAQ stock trading market and compare the influencing indicator of various source data.

We organized the following chapters of this thesis as follows. We introduce previous literature and present text mining and related technologies used in stock movement analysis in the "Related works" section. Describe the data collection and data construction process of the data sources used in the experiment in the "Data collection/construction" section. We give an overview of the workflows and techniques to verify the effectiveness

of textural feature representation and briefly describe how to integrate different kinds of stock-related information sources on stock prediction tasks in the "Methods" section. Later, the experiment and result analysis will be introduced in the "Experiment and results" section. The "Conclusion and future work" summarizes the thesis and introduces the follow-up work direction.

Chapter 2

Background

In this section, we review the current research status of stock price prediction based on data from different information sources and describe some of the methods used in our workflow.

2.1 Prior work on stock trend prediction

- **Studies based on textual information.**With the rapid development of social networks in recent years, Twitter and forums have provided us with a large amount of text data. Previous research found that news text information can affect the price changes in the stock market [10]. Kearney et al. [11] summarized and compared dictionary-based approaches and machine learning approaches as the two most commonly used methods in textual sentiment analysis in finance. Meanwhile, they also surveyed five financial models to test whether and how textual sentiment impacts the market. Based on the mood of Twitter, Huang et al. [12] investigated CNN and DNN based models to predict the index (S&P500, NYSE Composite, DJIA, NASDAQ Composite) trends of the next trading day. They point out that Twitter moods contribute to the prediction performance and CNN based model has achieved better results. In addition to the news headline, Liu et al. [13] also considered news content that is not reflected in the headline as an essential information provider. They proposed a hierarchical complementary attention network (HCAN) to capture valuable information in news content.

- **Studies based on multi-source information.**The most common combination of multiple information sources is textual information and historical trading information of stocks in previous studies [14]. In addition, there are some other sources of information used for stock price prediction tasks. Akita et al. [14] use paragraph vector to represent newspaper articles and combine the opening prices of ten companies in the same industry to improve the prediction performance. Following this idea, Ding et.al [15] proposed to incorporate knowledge graphs that include relational knowledge and industry knowledge into the learning process and this results in encoded valuable background knowledge. Feng et.al. [16] propose a solution that combines the temporal evolution and company relationship network in stock prediction tasks. In the extension of this solution, Matsunaga et al. [17] constructed a knowledge graph of seven kinds of relationships between companies and trained them through TGC to mimic how investors make decisions and results show a significant increase in the return ratio and Sharpe ratio. Chen et.al. [18] constructed a graph of mutual investment between companies, where the edges are the shareholding ratios, as a means of data enhancement.

However, most of the existing research focuses on the combination of different text representation methods and prediction models. In this thesis, we use both news text data, historical stock price data and their corresponding knowledge graph data for prediction tasks. Then compare the ability of represent the semantics of news text among different representation methods.

2.2 Text representation methods

In this thesis, we use event embeddings (described in Chapter 4), Global Vectors for Word Representation (Glove) [19], and Bidirectional Encoder Representations from Transformers(BERT) [20] to represent texts first. There have been some tasks that use text representation methods such as BERT to use news text to predict stock trends [21], and their effectiveness has been verified. We continue to use these methods and compare with the classic event embedding method.

2.2.1 Glove

Glove is a word-level representation method based on global word-word frequency statistics. First, Glove builds a global word co-occurrence matrix from a word corpus. Assuming that the co-occurrence matrix is X, X_i represents the word i, and X_{ij} represents the number of occurrences of the word j in the context of the target word i. The model uses global information, word co-occurrence information state, and context information modeling during training process. If the context positions of two words appearing in the corpus are similar, the word vectors of the two words are similar, which means that the vector pairs of the words have a better semantic description. The loss function of the Glove model is:

$$J = \sum_{i,j=1}^{|V|} f\left(X_{ij}\right)\left(W_i^{\mathrm{T}} W_j + b_i + b_j - \log X_{ij}\right)^2$$

Where W_i (W_j) and b_i (b_j) are the word vector of the target word i (j) and the corresponding bias, and $f(X_{ij})$ is the weight function.

Glove is easier to parallelize compared to other word embedding methods, so the training time is relatively short, but it is necessary to construct a global word co-occurrence matrix before training. Hence, it will occupy a larger memory space of the training machine.

2.2.2 Transfer learning from pre-trained language models

Transfer learning [22] refers to pre-trained a neural network model on a known task, and then fine-tuning, using the trained neural network as the basis of a new specific purpose model. This technology was often used in the field of computer vision. In recent years, the success of pre-trained language models has proved the feasibility of learning potential semantic information from massive unlabeled texts, and there is no need to individually label large amounts of training data for each downstream natural language processing task. In addition, the success of pre-trained language models has also created a new paradigm for natural language processing research, that is, first use a large amount of unsupervised corpus for language model pre-trained, and then use a small amount of labeled corpus for fine-tuning on specific natural language processing tasks, including

text classification, sequence labeling, machine translation, sentiment analysis, automatic question answering, and text generation etc.

BERT is a transformer [23] based framework that represent the contextual relationship between words in the text. It is widely used in natural language processing tasks. BERT highlights two key points: bidirectional and transformers. In training the BERT model, there are two training strategies, one is Masked Language Model (MLM), which means before the word sequence input into the BERT model, 15% of the words in each sequence will be replaced with [MASK], the goal is to predict the words that are masked. Another strategy is Next Sentence Prediction, which introduces sentence relationships that are not in the labeling language model.

2.3 Knowledge graph embedding

In recent years, the construction and application of knowledge graph have been developed rapidly. Many knowledge data bases, such as Freebase, DBpedia, YAGO and Wikidata, have been created and successfully applied to many practical natural language processing tasks, ranging from semantic parsing and named entity disambiguation to information extraction and question answering. Knowledge graph embedding focus on mapping the content of the knowledge graph including entities and relationships to the continuous vector space. Inspired by word vectors, researchers consider how to map the entities and relationships in the knowledge graph to a continuous vector space, and include some semantic information, which can make it more convenient to operate the knowledge graph in downstream tasks.

A typical knowledge graph can be represented by triples (head entity, relationship, and tail entity). At present, there are some translation distance-based models that can embed the knowledge graph into a continuous vector space. The TransE [24] model is simple and effective to model one-to-one relationships. However, transE is not ideal for modeling complex relationships, such as one-to-N or N-to-N complex relationships, so many subsequent model variants have been proposed. TransH [24] effective to model the N-to-one relationship and TransR [25] maps entities and relationships to different semantic spaces, and the relationship is no longer a single, but multiple relationship spaces. Comparing in these models, we only need to model the one-to-one relationship

between companies to verify the effectiveness of the relation graph features. Hence, we use the transE model for graph embedding from company relationships.

2.4 Evaluation Metrics

2.4.1 Accuracy

Accuracy is one of common evaluation metrics in classification tasks. In our task, Accuracy is the proportion of correctly classified samples in the total samples. As shown in the following formula:

$$Accuracy = \frac{prediction\ correct}{total\ prediction}$$

But the accuracy rate cannot fully represent the model performance on a data set with uneven data. For example, a data set has 990 positive samples and 10 negative samples. Now the model predicts all the samples as positive samples, then we say that the accuracy of the model is 99%. Hence, it is not enough to use the accuracy rate to evaluate the quality of a model.

2.4.2 Precision & Recall

For a binary classification task, the prediction results can be divided into 4 categories as shown in the Table 2.1.

Prediction \ Label	1(Up)	0(Down)
1(Up)	True Positive (TP)	False Positive (FP)
0(Down)	False Negative (FN)	True Negative (TN)

Table 2.1: Binary classification result matrix

We can get the definition of precision: $Precision(Up) = \frac{TP}{TP+FP} Precision(Down) = \frac{TN}{TN+FN} Thedefinitionof Recall$:Recall (Up) = TP$\frac{}{TP+FN} Recall(Down) = \frac{TN}{TN+FP}$

2.4.3 F1 score

Precision and Recall are a pair of contradictory performance metrics. Hence, in order to better evaluate the performance of the classifier, F1-Score is used as an evaluation to measure the overall performance of the classifier. For a two-category confusion matrix, F1-score can be calculated by:

$$F1 = \frac{2 \text{ Precision } * \text{ Recall}}{\text{Precision } + \text{ Recall}}$$

In our experiment, we need to predict the rise and fall separately. Therefore, we use the macro average F1 as metrics which means calculate the F1 value of each class, and then calculate the average F1 score of both classes.

Chapter 3

Data collection/construction

Since we did not find a complete data set that meets the requirements of this research, we constructed the task data set ourselves. We obtain three kinds of data: historical stock trading data, financial news text data,and company relationship knowledge graph. The rest part of this section will describe these three source of data and the data we constructed.

3.1 Historical trading data

The New York Stock Exchange (NYSE) is the world's largest stock exchange, and the Nasdaq is the second-largest stock exchange. Specifically, the NYSE has high requirements for listed companies, which are generally relatively mature and large. The Nasdaq market listing standard is lower than that of the NYSE, so it is suitable for high-tech startups. Hence, we select listed companies at those two representative stock markets as our experimental data.

We use the Yahoo Finance API[1] to get the historical stock market trading data of NYSE and Nasdaq from October 1, 2006, to February 13, 2020. The historical trading data including the opening price, closing price, highest price, lowest price, and trading volume of each stock. Figure 3.1 shows the data format we obtained in this step. We obtain 2,736 and 3,043 stocks trading data from NYSE and Nasdaq, respectively. After that, we filtered out the stocks with the final closing price of no less than $5 per share to

[1] https://github.com/ranaroussi/yfinance

ensure that the stocks obtained are not penny stocks[2].In the end, we obtained 1,446 and 935 stocks data from NYSE and Nasdaq for our experiments.

Date	Open	High	Low	Close	Volume
2006-10-20	2.820357	2.856786	2.809643	2.855357	639413600
2006-10-23	2.856786	2.925000	2.848214	2.909286	832507200
2006-10-24	2.900357	2.917143	2.864286	2.894643	463212400
2006-10-25	2.905357	2.928571	2.893214	2.917143	485214800
2006-10-26	2.925000	2.950000	2.897500	2.935357	432756800

FIGURE 3.1: Trading data sample (APPL)

3.2 Knowledge graph

We use the company relationship knowledge graph released by Feng [3].This knowledge graph extracts the relationship between companies stored in the form of (subject, predicate, object) triples from Wikidata. In Wikidata, the knowledge base is composed of the item (Q followed by a number) and property (P followed by a number). Both intel ($Q248$) and Nvidia ($Q182477$) belong to the same specific industry ($P452$) and are represented in the knowledge graph as ($Q248$,$P452$,$Q182477$). Figure 3.2 shows the industry relationship between the two companies; if two companies are in the same industry, similar external factors will often affect their company's stock price trend.

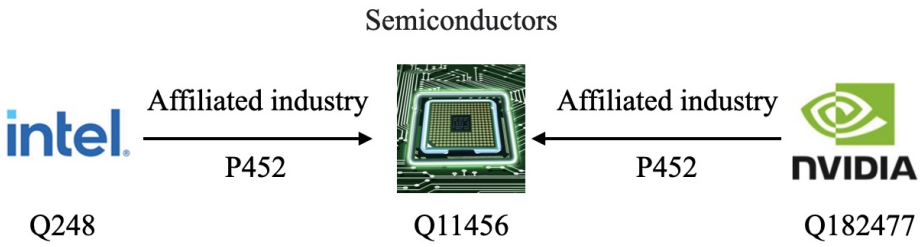


FIGURE 3.2: Some representative nodes in wikidata-based industry relation graph

[2]The United States Securities and Exchange Commission(SEC) define it as all stocks with a trading price of less than $5 per share, in order to remind investors to be cautious in investing in such high-risk stocks.

[3]https://github.com/fulifeng/Temporal_Relational_Stock_Ranking/tree/master/data

The data of Wikidata is in JSON format. In order to better understand the relationship of Wikidata, we converted the format into csv and used gephi for visualization. The Figure 3.3 shows that 293 companies belong to 33 industrial relationships. The same color cluster in the figure represents the companies in the same industry. The label of the node is the thicker of each companny in the stock market. It is not difficult to see that GOOG (Alphabet Inc.), AAPL (Apple Inc.), YNDX (Yandex N.V.), MSFT (Microsoft Corporation), and ADBE (Adobe Inc.) are all linked to yellow. Therefore, the company relationship graph extracted from Wikidata can be used in our experiments. However, it can also be seen that the industry relations here are not exactly the same as the information classified by the SEC.

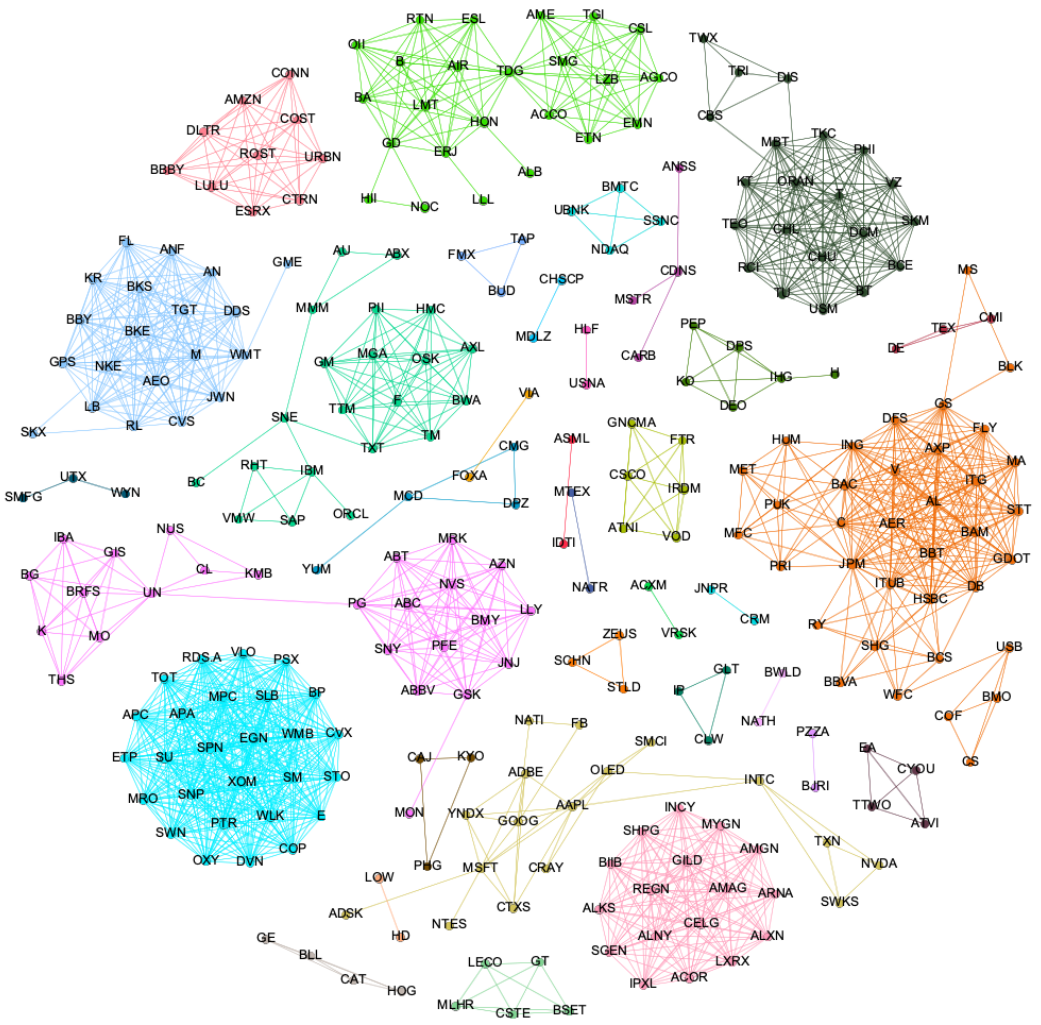
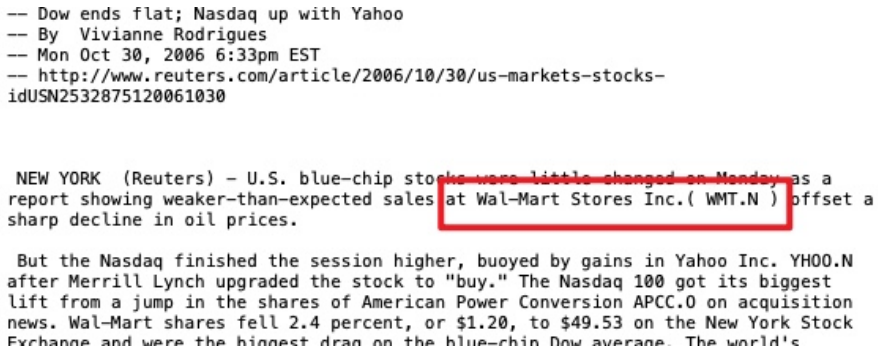


FIGURE 3.3: Visualization of industry relation graph

3.3 Financial news data

- The raw financial news data in task 1 are from Bloomberg and Reuters from October 20, 2006, to November 21, 2013. This dataset was created and released by Ding et al. [26].The data set consists of a total of 448,395 news articles, and each article lists the headline, author, publication time, original link, and news content. As shown in the red mark in Figure 3.4, for specific stock information in the news, its stock thicker will be marked when the news content mentions the company, so it can be convenient to align it with the company's stock price in the follow-up process. The distribution of headline is shown in Figure 3.5.


-- Dow ends flat; Nasdaq up with Yahoo
-- By Vivianne Rodrigues
-- Mon Oct 30, 2006 6:33pm EST
-- http://www.reuters.com/article/2006/10/30/us-markets-stocks-idUSN2532875120061030

NEW YORK (Reuters) - U.S. blue-chip stocks were little changed on Monday as a report showing weaker-than-expected sales at Wal-Mart Stores Inc.(WMT.N) offset a sharp decline in oil prices.

But the Nasdaq finished the session higher, buoyed by gains in Yahoo Inc. YHOO.N after Merrill Lynch upgraded the stock to "buy." The Nasdaq 100 got its biggest lift from a jump in the shares of American Power Conversion APCC.O on acquisition news. Wal-Mart shares fell 2.4 percent, or $1.20, to $49.53 on the New York Stock Exchange and were the biggest drag on the blue-chip Dow average. The world's

FIGURE 3.4: Sample news text in news data 1

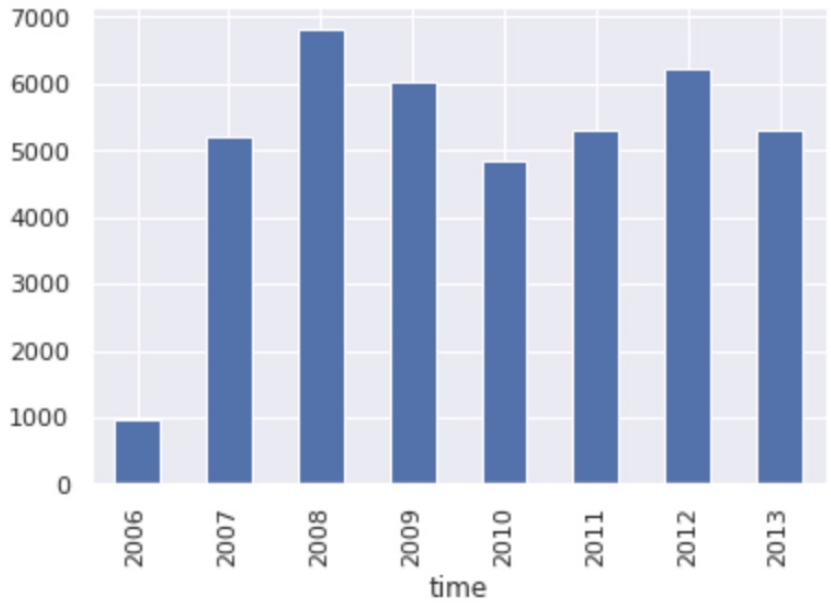


FIGURE 3.5: News distribution of the financial news data

- News archive of NYSE/NASDAQ which still has a price higher than $10 per share from October 02, 2008 to January 15, 2020. This data mixes 25% of Reuters financial news, 40% financial reviews on the website of zack investment, and company information from unknown sources. This data is available on the Kaggle website[4].

3.4 Dataset building

In our experiments, we compare the effects of different text representation methods and different factors on prediction task. Each factor corresponds to its separate data source. We are not able to get enough data after the first news data source is aligned with the company in the knowledge graph, in order to obtain sufficient training data, we switched to the second news data source to construct a second data set. The differences between this two datasets are shown in Table 3.1. The workflow of the construction of the two data sets is shown in the Figure 3.6. We use Dataset 1 to explore different text representation methods in prediction task. Dataset 2 study the influence of multi-source source information in stock prediction task.

	Dataset 1	Dataset 2
Data	Historical prices	Historical prices
	News headlines 1	News headlines 2
		Relation graph
News source	Reuters (financial reports)	Investing.com (financial reports, forum opinions)
Feature	1.Raw news headlines, stock symbols need to be manually extracted and aligned	1.Processed news headlines with stock symbols labels.
	2.For most of the headlines, subject, predicate, and object are complete	2.One or more of the subject, predicate, and object are missing in more than half of the headlines

Table 3.1: Differences between two datasets

[4] https://www.kaggle.com/gennadiyr/us-equities-news-data

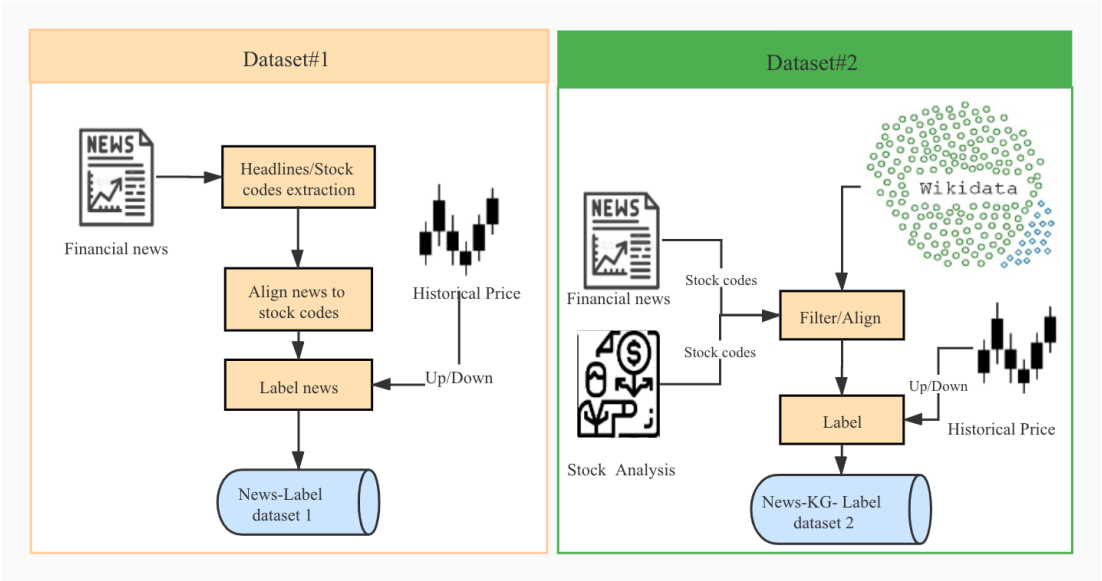


FIGURE 3.6: Construction workflow of two datasets

3.4.1 Dataset 1

In this step, we will build a task data set based on our existing data set. First, we need to construct the labels for the news. To align the news with its related stocks, we use the stock codes that appear in the text headline or content of the news as related stocks. Then we construct labels for news headlines, which is 1 if on the next day the stock price rises, and 0 if on the next day the stock price falls. We compare the open and close price of the stock on that specific day to determine its rise and fall. After that, we construct new-label pairs as the dataset for the embedding task. An example of news-label data sets are shown in Figure 3.7. The summary of information on dataset 1 is shown in Table 5.1.

```
LeBron James Revisits Cleveland as Mere Mortal: Scott Soshnick  1
Ford Tops `Government Motors' in Sales to U.S. After Avoiding Obama Rescue      0
Seagate Said to Have Rejected Western Digital Takeover Proposal 0
Seagate Said to Have Rejected Western Digital Takeover Proposal 0
Google's Bid for Groupon Drives Interest in Daily-Deal Sites    0
LivingSocial Arranges $175 Million Investment From Amazon.com   1
GM November Sales in China Rise 11% on Demand for Chevrolet, Buicks Cars        1
Goldman Sachs Should Split Into Three Units, Michael Price Says: Tom Keene      0
```

FIGURE 3.7: A example of news-label datasets

3.4.2 Dataset 2

Our goal is to study the influence of company relationship graph in the stock movement prediction task. Since the data in task 1 is the original news data, entity linking needs to align financial news with the company relationship knowledge graph. Since the primary purpose of this task is to explore the influence of the knowledge graph in stock price prediction, we choose prepossessed financial news data as the news source for this task. Each news in this data set is labeled with a company ticker. Hence, based on the original news data set and knowledge graph data set we have collected, we use the following steps to build the dataset for this task:

- There are a total of 215,447 news headlines of stocks traded on NYSE/NASDAQ in the raw historical financial news archive. Therefore, we first merge the knowledge graphs of NYSE and NASDAQ released by Feng [16], and then use the company's ticker to filter the news that appears in both the knowledge graph and financial news. We got 83314 financial news and opinions headlines and 1075 companies in 35 types of relationships in this step. The detailed filtered knowledge relation lists are shown in Appendices.
- Similar to task 1, label each news data based on the opening price and closing price.

Chapter 4

Methods

Figure 4.1 shows the overall framework of our experiment. The whole experiment is composed of three different kinds of data, constructed into two tasks. Among them, the yellow part represents the framework of task one, and the green part represents the framework of task two. The blue part represents the historical stock price data corresponding to the two tasks, respectively. We describe the methods and techniques we used in the rest of this section.

4.1 Historical price encoding

The historical stock price information is typical sequence data. In previous studies [27, 28], LSTM, as a variant of the RNN model, is a deep learning model used to solve the problem of vanishing gradients in long sequences. LSTM has three gates: update gate, forget gate, and output gate. The update and forget gate determine whether to update each element of the unit, which is helpful for prediction tasks that heavily rely on historical stock price information. Therefore, in this part, we also use LSTM to capture long-term dependencies.

The historical stock price data contains six features: opening price, highest price, closing price, lowest price, trading volume and the rise and fall of the day. For expression of rising and falling, we set the state of the first day of the sequence data to be represented by 0. In the remaining data, 1 means that the stock price rose on that day, and -1 means that the stock price fell.

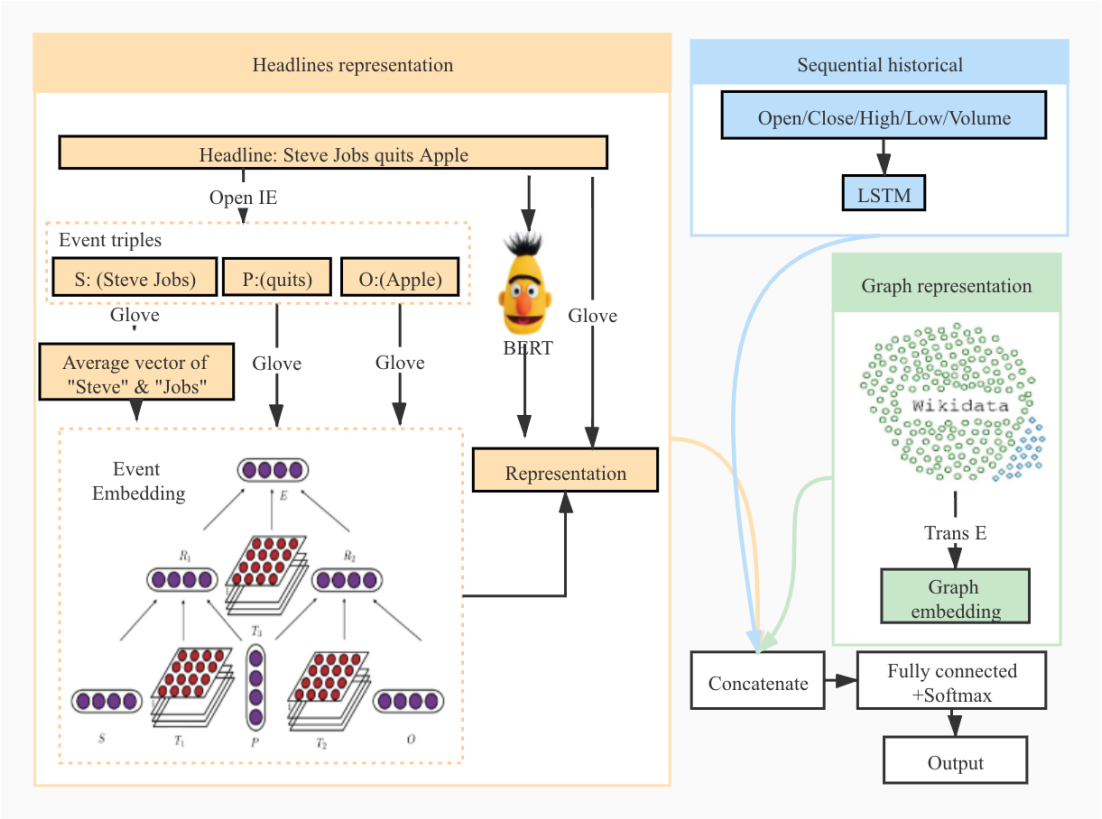


FIGURE 4.1: Overall framework of experiments

Before the data is fed to the LSTM network, we need to adjust the historical stock price data to the range of 0 to 1 through data normalization. We use the minmax method to normalize the price data by using the following formula:

$$z = \frac{x - min}{max - min}$$

Where x is the input vectors, max is the maximum value of the sample data, and min is the minimum value of the sample data.

Since historical stock price information is directly crawled raw data, there may be empty sequences in the data fed to the network. Here, we pad the sequence to the same length and use the mask matrix to distinguish between the non-padding and the padding parts. The purpose is to make the learning network only consider sentences of their actual length and ignore the useless padding part.

Finally, we feed stock historical data into the LSTM network, and then we take the final state of the hidden layer as the encoding of stock information.

4.2 Headlines representation

In the previous section, we modeled numerical data. In this section, we introduce the methods of representing financial news text information. In this process, we use event embeddings proposed by Ding [29], Glove and BERT to represent news headlines.

4.2.1 Event triple embeddings

4.2.1.1 Event triple extraction

As shown in the figure 3.4, the financial news data contains information such as headline, content, time, and corresponding URL. It proves to be more useful for predicting task news headlines than the main content [30]. Hence, we only deal with news headlines in our experiment.

First, we need to extract event information from financial news headlines. Traditional word embedding methods can only represent individual words discretely. In order to obtain the semantic information of the overall event in news headlines, we use Open Information Extraction (Open IE) [31] to extract the triplet of the news headline as the basic event representation. Specifically, the complete sentence described in the financial news headlines is extracted into corresponding triple (subject, predicate, objects). The triple information here is what we call event information.

We use the Open IE5 framework developed by Stanford [1] to extract event triples from financial news headlines. Table 4.1 shows several event information extracted from news headlines using OpenIE, the value of confidence indicates that the triplet conforms to the form of (subject, predicate,object). It can be seen from the table that for simple sentences (such as sentences 1-3), sentence will be divided into triples of subject, predicate and object directly. For some complex sentence containing multiple verbs (sentence 4), the extracted event may contain more than one set of triples.

[1] https://nlp.stanford.edu/software/openie.html

Subject	Predicate	Object	Confidence
1,Samsung Electronics says selling some of its shares in ASML			
Samsung Electronics	says	selling some of its shares in ASML	0.8684
2,Google challenges record EU antitrust fine in court			
Google	challenges	record EU antitrust fine in court	0.9324
3,Amazon Web Services announces new AI products			
Amazon Web Services	announces	new AI products	0.9436
4,Nvidia shares set record as Volta chips ensure future growth			
Volta chips	ensure	future growth	0.9266
Nvidia shares	set	record as Volta chips ensure future growth	0.9527

TABLE 4.1: Samples of events extracted from news headlines using OpenIE 5

4.2.1.2 Neural tensor network

Now we have the triple text representation of each news headline, The subject, predicate, and object of the triples correspond to (S, P, O) in the model respectively. In view of the fact that the result of initializing the input with pretrained word vectors is better than initializing the word vector randomly, we uses the pre-trained GLOVE (6B tokens, 400K vocab) [19] to generate 100 dimensions word vectors.

Follow the idea of Ding et al.[26] we use Neural Tensor Network to learn distributed financial events representations. The architecture of this model is shown in Figure 4.2. The input of the network is pretrained 100 dimensional word vectors of event triples and the output creates latent event vectors. As can be seen in Figure 4.1, subject, predicate and object often contain more than one word. We uses the average vector over pretained vectors of words to represent its corresponding event element.

Two tensors T_1 and T_2 are used to model the relationship between the Subject (S) and the Predicate (P) and Object (O). Specifically, R_1 models the relationship between the subject and the predicate through the following formula:

$$R_1 = S^T T_1^{[1:k]} P$$

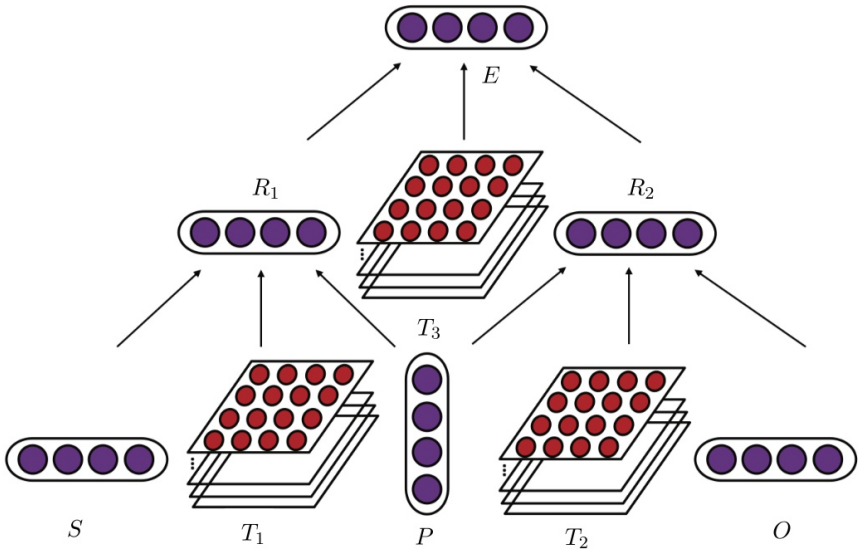


FIGURE 4.2: Event embedding architecture [29]

where T_1 is a k matrices tensor with a shape of 100×100 dimensions (same dimension as the input word vector). Similarly, R_2 and E are calculated in the same way as R_1.

4.2.1.3 Event embedding

For the event learning process, given a set of event triples $E = (S, P, O)$, First, suppose that the event triples should achieve a higher score in training process than the corrupted event triples. The corrupted event triples $E_c = (S_C, P, O)$ or $E_c = (S, P, O_c)$ were constructed by replace the A or O with a random word in the training process. We train the neural tensor network with max margin loss through the following formula:

$$L_e = max(0, 1 - f(E^i) + f(E_c^i))$$

where i represent the number of train events and corrupted events, respectively. $f()$ is the transformation of the neural tensor network.

4.2.2 BERT

We also use BERT to vectorize the headlines. Specifically, based on the input requirements of the BERT model, first we add [CLS] and [SEP] tokens to the beginning and end of all news headlines, respectively. Then, for sentences that exceed the set maximum length, we truncate them at the maximum length. For sentences shorter than

the set maximum length, we add empty tokens to make up the length. Then, convert the word into the id in the corresponding vocabulary. At this time, each tokens has a corresponding id in vocabulary. Subsequently, we add segment ids to distinguish two different sentences. Finally, we feed the two matrices generated into the model for forward propagation training to obtain the vectors of the headlines.

4.2.3 Glove

As mentioned in the background, Glove can represent each word in the news headline as a vector. In our paper, for news headlines, we directly use the average vector of words in all news headlines as our headlines representation methods.

4.3 Graph embedding

The knowledge graph is composed of a large number of fact triples, such as (China, Capital, Beijing) can represent knowledge in the real world. Graph embedding is the representation of entities and relationships in the knowledge graph in a low-dimensional space. In this section, we describe how to encode the company relationship knowledge graph in our prediction task. As a large semantic network that can be used to model the relationship between entities or concepts in the real world. Given the triple information of a company relationship graph $(Q, P, Q^{'})$, Q and $Q^{'}$ represent two company nodes respectively, and P represents the relationship between this two company nodes.

We use transE [24] as our graph embedding method. In the transE model representation, Q and Q1 are defined as the head entity and the tail entity, respectively. As shown in Figure 4.3, transE assumes that the sum of the head entity vector(h) and the relation vector(r) should be as equal to the tail entity vector(t) as possible. If it is a wrong triple, then their embedding does not satisfy this relationship. As shown in the following formula:

$$Q + P \approx Q^{'}$$

Similar to the loss function of event embedding, we use distance d to represent the score, the loss function represented by the knowledge graph is as follows:

$$L_{(h,r,t)} = max(0, 1 - d_{neg} + d_{pos})$$

where d is

$$d = ||h + r - t||$$

In this process, the head entity or tail entity is replaced with a random entity in the

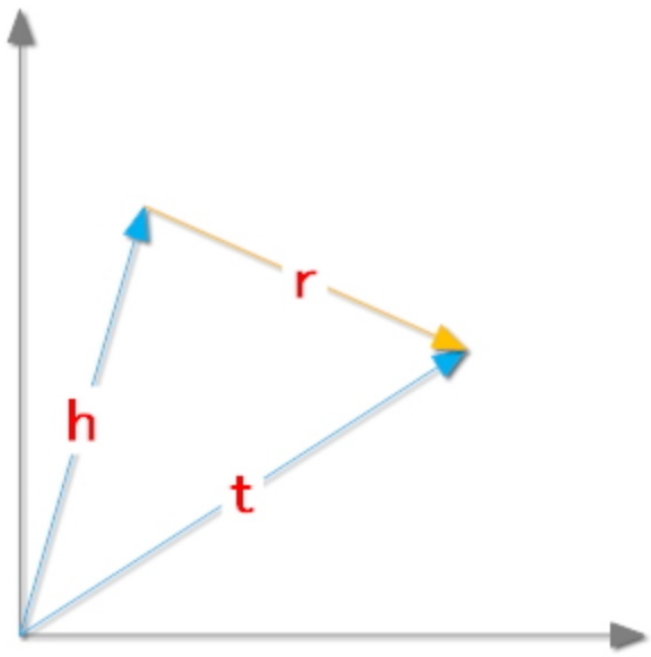


FIGURE 4.3: transE moedel

triple to obtain a negative sample. Since there are 35 kinds of relationships among 1107 companies in the data set we constructed. Referring to the transE code [2], we obtained the embedding of entities and relationships in the wikidata knowledge graph.

[2] https://github.com/wuxiyu/transE/blob/master/tranE.py

Chapter 5

Experiments and results

This section will describe the specific details of the experiments and the experimental results, and explains and analyzes the experimental results. We use pytorch as our deep learning framework. All experiments are run on a GTX1080ti GPU with 11G memory.

5.1 Event-driven stock movement prediction

In this task, we use different textual features to represent the event information contained in the headlines of financial news to compare the ability of different textual representations to capture the semantic information of financial events.

5.1.1 Text representation methods

The text representation method and prediction model used in the experiment are as follows:

- Glove. We use the pre-trained word vector namely glove.6B with 100 dimensions , which contains word vectors of 50d, 100d, 200d, and 300d commonly used English words, derived from 400K vocabulary in Wikipedia and Gigaword. We uses the average vector over pretained vectors of all the words in headlines.
- Event embedding. We use the event vector after neural tensor network training introduced in Chapter 3 as the text representation method.

- BERT. We used the pre-trained bert-base-uncased [20] python implementation with 12-hidden layer, 768 dimension.

After obtaining the representations of headlines and sequential historical prices, we connect it to a fully connected layer and softmax to obtain the classification result. Our neural network use Adam optimizer with initial learning rate of 0.001.

5.1.2 Experimental data

Now we have 106,521 raw Reuters news items. Since the news reported is global financial news, we extracted 23,376 news headlines that can correspond to stock codes through the method in Chapter 3. We use these data as our news text data. Table 5.1 show the split of news text datasets, since the news text data is the same as that of Ding et al., [26], for better comparison, we split the data set into the same time period.

Dataset#1 23,376	Up	12,108	51.80%
	Down	11,268	49.20%
	Training	Development	Test
Time	02/10/2006-18/06/2012	19/06/2012-21/02/2013	22/02/2013-19/11/2013
headlines	17,637	2,967	2,779

TABLE 5.1: Summary of information on dataset 1

5.1.3 Results

The results of different text representation methods on the task of stock price prediction are shown in the Table 5.2. The results show that financial news and stock price movements are indeed related. Our experimental results have achieved results close to Ding. At the same time, it can be seen that the performance of BERT is higher than event embedding in one run.

As a binary classification task, the accuracy of 59.90% can verify that the information of the news text is useful for stock price prediction. This is because the stock market price fluctuates greatly and depends on many factors. In addition, it can be seen that when we

use Glove to represent the headlines, only the word vector of the headline is averaged, and the result is lower than the average vector of the subject, predicate and object after the neural network training. Hence, for headline representation, segmentation into triples is an effective method.

Although it can be seen that news headlines are an effective factor in forecasting, it is not difficult to find that the effect is very limited. This is because the stock market price fluctuates greatly and depends on many factors. It is almost impossible to quantify all these factors. Relying on the features of news text alone is not enough to achieve good results. Therefore, in the next part, we will combine the indicators of multi-source information into stock prediction task.

Majority baseline	Accuracy: 0.5180		
Ding et al [26]	Accuracy: 0.5960		
Glove	Accuracy: 0.5689		
	Precision	Recall	F1 score
Down	0.5278	0.0679	0.1204
Up	0.5703	0.9532	0.7136
Replication of Event embedding [26]	Accuracy: 0.5822		
	Precision	Recall	F1
Down	0.5935	0.0573	0.1045
Up	0.5817	0.9709	0.7276
BERT	Accuracy: 0.5990		
	Precision	Recall	F1 score
Down	0.5777	0.2919	0.3878
Up	0.6050	0.8356	0.7018

TABLE 5.2: Results of different event representation methods

5.2 Multi-source information stock movement prediction

In this task, we explore the impact of different kinds of data source: company relationship knowledge graphs, financial news, and historical price data on stock price prediction task.

For the historical price information part of this task, we feed the past 100 days into the LSTM network. After that, we fine-tune the BERT model on news headline to get

the word embeddings. Finally, we concatenate historical price features and the BERT embeddings together with the graph embedding vector generated by trans E. After obtaining the concatenated vector, we connect it to a fully connected layer and softmax to obtain the classification result. This neural network also use Adam optimizer with initial learning rate of 0.001.

Through the various methods in the previous part, at this time we get the vector of historical stock price data, the vector of event embedding after extraction, the vector of word embedding of news headlines generated by pre-training BERT, and the graph embedding obtained using transE . In this step, we concatenate the vectors to get the prediction results. Among them, the dimensions of historical stock price information, news embedding, and graph embedding are 100, 768, and 50, respectively.

5.2.1 Experimental data

In the previous stock price news data, the news label only needs to be aligned with the stock code appearing in the news text to construct the data. The task here needs to align the previous label data with the nodes (companies) of the knowledge graph again. The Reuters news text alignment data volume is less than 5,000. Therefore, we choose the preprocessed data set 2 of kaggle as the news text of this task.

In the end, we aligned the stock historical trading data, knowledge graph, and news headlines, and finally obtained 83313 dataset. Table 5.3 shows the split of news in dataset 2.

Dataset#2 83,313	Up	43,656	52.4%
	Down	39,657	47.6%
	Training	Development	Test
Time	02/08/2008-14/04/2016	15/04/2016-23/11/2018	24/11/2018-13/02/2020
headlines	66,651	8,331	8,331

Table 5.3: Summary of information on task 2 data

5.2.2 Models

- Historical price. Only use historical stock price (100 days) information fed into LSTM layer.
- News embedding. Only use pretrained base bert of news vector as input.
- News embedding & Knowledge graph embedding. Concatenate news vector and the company knowledge graph embedding vector as input. The embedding size of transE is set to 50 dimensions.
- News embedding & Historical price. Concatenate news vector and historical price feature as input.
- Multi-source fusion. Event embedding+ sequential data+ company relation graph.

His	News	Graph	Model & Label	Precision	Recall	F1 score
✓			Historical price	Accuracy: 0.5813		
			Down	0.5921	0.2107	0.3118
			Up	0.5809	0.9740	0.7278
	✓		News embedding	Accuracy:0.5617		
			Down	0.5000	0.1180	0.1909
			Up	0.5722	0.9090	0.7024
	✓	✓	News & Knowledge graph	Accuracy: 0.5808		
			Down	0.5373	0.1055	0.1764
			Up	0.5848	0.9327	0.7189
✓	✓		News embedding & Historical price	Accuracy: 0.5923		
			Down	0.5379	0.2964	0.3822
			Up	0.6090	0.8114	0.6958
✓	✓	✓	Multi-source fusion	Accuracy: 0.6115		
			Down	0.5784	0.3207	0.4126
			Up	0.6218	0.8269	0.7098

Table 5.4: Summary of results of task 2

5.2.3 Results

The experimental results of multi-source information stock trend prediction are shown in Table 5.4. As can be seen in it, using the embedding of historical stock price information and news text information alone as predictions, both have achieved an accuracy rate of more than 56%, which proves that they are both meaningful features in stock price predictions. When news and knowledge graphs are spliced together, the model's performance is improved, but by a small margin. However, when the three information sources of historical stock price, news data, and knowledge graph are concatenated into the model, the accuracy rate reaches the highest 61.15%.

In addition, it is not difficult to find that whether it is a single-factor prediction or a combination of multiple factors, the prediction of the label up has a high recall, and the prediction of the label down has a low recall.

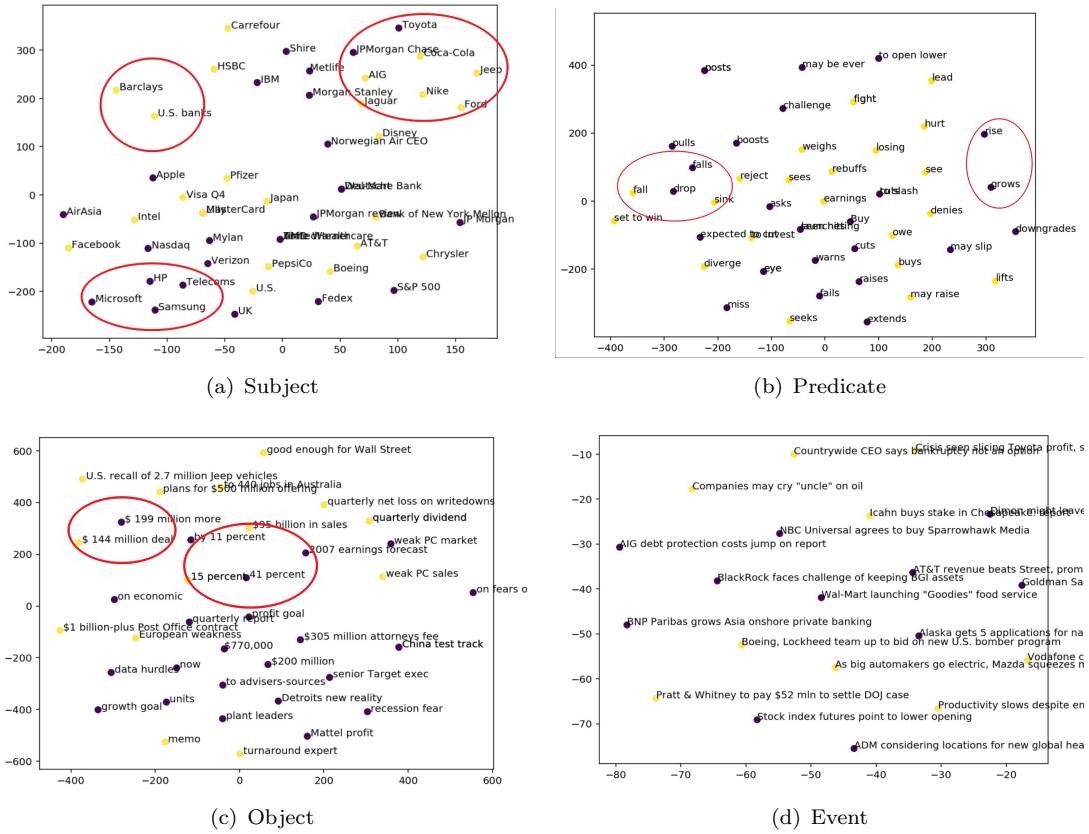
5.3 Post analysis

5.3.1 Embedding visualization

In the previous part, we got the results of stock price prediction. In this part, we will conduct a more intuitive analysis of various text representation methods. Specifically, first, as shown in the Figure 5.1, we use t-SNE [32] to visualize the top 50 subject, predicate, and object vectors and top 30 event vectors into a two-dimensional space. Among them, yellow and purple represent that the event label is down and up respectively.

It can be seen that companies in similar industries such as HP, Samsung, and Microsoft are clustered together. At the same time, the vectors expressing percentages in rise, grows, or drop and falls are clustered together. This shows that Glove vector or average vector over Glove vectors of words can obtain the semantics of subject, predicate and object.

For the event vector, the figure here can only show that a vector can be used to represent the global news headline at this moment, but it cannot intuitively see whether the semantic information similar to the event has been obtained. We will further analyze it in the following sections.

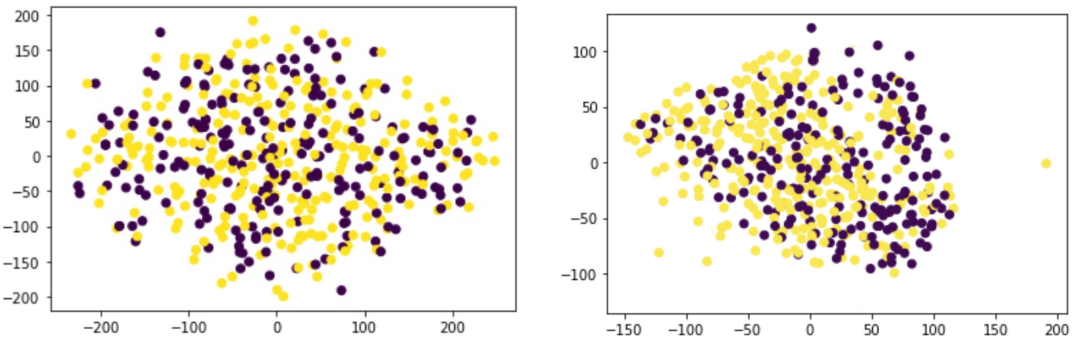


(a) Subject (b) Predicate

(c) Object (d) Event

FIGURE 5.1: The t-SNE visualization of different embeddings 1

Figure 5.2 shows t-SNE visualization of top 500 event embedding vectors and the news vector generated by BERT. It can be seen the yellow and purple events are separated a bit better in BERT.



(a) Event embedding (b) BERT

FIGURE 5.2: The t-SNE visualization of different embeddings 2

5.3.2 Event similarities

In this section, in order to specifically explore the connection between event embeddings, we conduct a post analysis of news headlines. Specifically, if the vector representing the event is mapped to a similar position in the vector space, it means that they have a high degree of similarity. Here, we use cosine similarity to measure the effectiveness of different text representation methods.

Table 5.5 shows that we manually select "UnitedHealth quarterly profit rises 3 percent" as the object headline, and select top 10 headlines with the highest cosine similarity with the object headline in event embedding and BERT respectively. UnitedHealth is the world's second-largest health insurance and health information technology company. However, among the 20 news headlines listed, only Metlife in event embedding belongs to the category of insurance companies, and the rest of the companies are divided into various industries. In addition, many news headlines contain "quarterly", "profit" "rise" "percent" and their similar words such as "earning","raises". The bold sentences are the same headline that are ranked in the top 10 similaritieswe can see that only one of the first ten sentences is the same. In addition, the headlines filtered by the BERT are more similar in semantics to the object headline. This is consistent with our previous experimental results.

Object headline: UnitedHealth quarterly profit rises 3 percent			Up
		News headlines	Label
Event Embedding	1	IBM may raise full-year software demand forecast	Up
	2	JPMorgan stays top of investment banking in first quarter	Up
	3	Hawaiian Airlines expects revenue growth from hotel, car rental bookings	Down
	4	MetLife sees 2010 operating earnings up 50 percent	Up
	5	**Visa authorizes $1.5 billion buyback, profit rises 89 percent**	Down
	6	Blackstone plans bid for Rio with Chinese	Up
	7	AIA IPO seeks up to $14.9 billion	Down
	8	Walmart eyeing Europe with new London team	Down
	9	Dubai could raise $1 billion from telecoms stake sales: J.P. Morgan	Up
	10	AT&T may bid for EchoStar by year's end	Down
BERT	1	Coca-Cola raises quarterly dividend by 10 percent	Up
	2	Eaton 4th-quarter profit rises 6 percent	Down
	3	Exxon sees '08 capital spending up 20 percent	Up
	4	**Visa authorizes $1.5 billion buyback, profit rises 89 percent**	Down
	5	ConocoPhillips beats estimates as output increases	Up
	6	Empire State Realty Trust IPO seen pricing at 13−15 per share	Up
	7	HP sees 2009 revenue up 5-6 percent	Up
	8	AIG hires KeyCorp exec to oversee finance, risk	Up
	9	AT&T sees 500,000 wireless net adds in second quarter	Up
	10	PepsiCo beats Street; CFO backs current portfolio	Up

Table 5.5: Top 10 headlines with the highest cosine similarity to the object headline

Chapter 6

Conclusion and future work

6.1 Conclusion

In this thesis, we compared the performance of different news text representation methods in the stock movement prediction task. On this basis, we found a solution that integrates multi-source information (historical transaction information, news text information, and knowledge graph information) in this task.

First, based on the raw news text data released by other researchers [26], we cleaned and constructed two data sets with stock price fluctuation labels. Dataset 1 contains high-quality news text information and its corresponding label. Including 23376 news headlines form Reuters. Dataset 2 contains 83,313 news headlines and their corresponding 33 different relationship graph data of 1,015 listed companies. One limitation of dataset 2 is that the quality of the news text is relatively low since it has been preprocessed.

For the text representation method, we tried to split the news headlines into triples (Subject, Predicate, Object) with OpenIE, and then fed it in a Neural Tensor network to obtain the event vectors [29]. We also tried pre-trained models like Glove and BERT to compare the text representation performance. The experimental results on dataset1 shows that BERT has achieved the best results, achieving an accuracy of 59.90% on the prediction task. After that, we selected the best performing BERT model and conducted a multi-source information fusion prediction experiment on dataset 2. The experimental results show that the three information sources are all effective information sources. In

this case of poor news data quality, an accuracy rate over 61% has been achieved. The result shows that historical stock price data contributes more to the model than the knowledge graph.

Finally, we used t-SNE to visualize the event vector into two dimension space and manually enumerated the similarity of some news headlines. It can be intuitively seen that BERT has achieved the best news text representation performance.

6.2 Future work

Although stock trend prediction has always been a relatively large research topic, since all the data sets in this experiment are constructed by ourselves, the quality of the data sets directly determines the subsequent experimental results. Therefore, a direction that needs to be improved in the future is that we need to crawl more high-quality news texts and corresponding knowledge graphs to obtain better data input. Secondly, in order to focus on the different text representation methods, we all use the simplest fully connected layer as the prediction in our experiments. Previous researchers have made many innovations in the prediction model [13, 33]. Therefore, further expansion in the direction of the prediction model is also a future direction.

In addition, except for the task of stock prediction, the method of event embedding using triples to construct global semantic information is very similar to the method of transE obtaining knowledge graph information through triples. Therefore, using language models to model knowledge graphs is also an interesting research direction. The most representative is K-BERT [34], which uses an independent TransE algorithm to obtain the entity vector, and then embeds the entity vector into BERT. It is also worthy of further exploration in the future.

Appendix A

Company relation

	Properties	Property code	Relation description in knowledge graph
1	country	P17	sovereign state of this item (not to be used for human beings)
2	operating system	P306	operating system (OS) on which a software works or the OS installed on hardware
3	described by source	P1343	mentioned in news article/subject of
4	legal form	P1454	legal form of an entity
5	industry	P452	specific industry of company or organization
6	award received	P166	award or recognition received by a person, organisation or creative work
7	different from	P1889	item that is different from another item, with which it is often confused
8	owner of	P1830	entities owned by the subject
9	owned by	P127	owner of the subject
10	use	P366	main use of the subject (includes current and former usage)
11	member of	P463	organization, club or musical group to which the subject belongs
12	headquarters location	P159	city, where an organization's headquarters is or has been situated
13	part of	P361	object of which the subject is a part
14	subsidiary	P355	generally a fully owned separate corporation
15	airline alliance	P114	alliance the airline belongs to
16	founded by	P112	founder or co-founder of this organization, religion or place
17	operating area	P2541	geographic area or jurisdiction an organisation or industry operates in, serves, or has responsibility for

	Properties	Property code	Relation description in knowledge graph
18	airline hub	P113	airport that serves as a hub for an airline
19	participant in	P1344	event in which a person or organization was/is a participant
20	located in time zone	P421	time zone for this item
21	parent organization	P749	parent organization of an organization, opposite of subsidiaries
22	platform	P400	platform for which a work was developed or released
23	business division	P199	organizational divisions of this organization
24	location of formation	P740	location where a group or organization was formed
25	stock exchange	P414	exchange on which this company is traded
26	followed by	P156	immediately following item in a series of which the subject is a part
27	chief executive officer	P169	highest-ranking corporate officer appointed as the CEO within an organization
28	named after	P138	entity or event that inspired the subject's name
29	country of origin	P495	country of origin of this item (creative work, food, phrase, product, etc.)
30	follows	P155	immediately prior item in a series of which the subject is a part
31	instance of	P31	that class of which this subject is a particular example and member
32	board member	P3320	member(s) of the board for the organization
33	product produced	P1056	material or product produced by a government agency, business, industry, facility, or process
34	item operated	P121	equipment, installation or service operated by the subject
35	source of income	P2770	source of income of an organization or person

Bibliography

[1] Anita Yadav, CK Jha, Aditi Sharan, and Vikrant Vaish. Sentiment analysis of financial news using unsupervised approach. *Procedia Computer Science*, 167:589–598, 2020.

[2] Doojin Ryu, Hyeyoen Kim, and Heejin Yang. Investor sentiment, trading behavior and stock returns. *Applied Economics Letters*, 24(12):826–830, 2017.

[3] G Pui Cheong Fung, J Xu Yu, and Wai Lam. Stock prediction: Integrating text mining approach using real-time news. In *2003 IEEE International Conference on Computational Intelligence for Financial Engineering, 2003. Proceedings.*, pages 395–402. IEEE, 2003.

[4] Robert P Schumaker and Hsinchun Chen. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM Transactions on Information Systems (TOIS)*, 27(2):1–19, 2009.

[5] Pegah Falinouss. Stock trend prediction using news articles: a text mining approach, 2007.

[6] Robert P Schumaker and Hsinchun Chen. A quantitative stock prediction system based on financial news. *Information Processing & Management*, 45(5):571–583, 2009.

[7] Xi Zhang, Siyu Qu, Jieyun Huang, Binxing Fang, and Philip Yu. Stock market prediction via multi-source multiple instance learning. *IEEE Access*, 6:50720–50728, 2018.

[8] You-Shyang Chen, Ching-Hsue Cheng, Chiung-Lin Chiu, and Shu-Ting Huang. A study of anfis-based multi-factor time series models for forecasting stock index. *Applied Intelligence*, 45(2):277–292, 2016.

[9] Zhifeng Guo, Kaile Zhou, Chi Zhang, Xinhui Lu, Wen Chen, and Shanlin Yang. Residential electricity consumption behavior: Influencing factors, related theories and intervention strategies. *Renewable and Sustainable Energy Reviews*, 81:399–412, 2018.

[10] Qing Li, Jinghua Tan, Jun Wang, and HsinChun Chen. A multimodal event-driven lstm model for stock prediction using online news. *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[11] Colm Kearney and Sha Liu. Textual sentiment in finance: A survey of methods and models. *International Review of Financial Analysis*, 33:171–185, 2014.

[12] Yifu Huang, Kai Huang, Yang Wang, Hao Zhang, Jihong Guan, and Shuigeng Zhou. Exploiting twitter moods to boost financial trend prediction based on deep network models. In *International Conference on Intelligent Computing*, pages 449–460. Springer, 2016.

[13] Qikai Liu, Xiang Cheng, Sen Su, and Shuguang Zhu. Hierarchical complementary attention network for predicting stock price movements with news. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1603–1606, 2018.

[14] Ryo Akita, Akira Yoshihara, Takashi Matsubara, and Kuniaki Uehara. Deep learning for stock prediction using numerical and textual information. In *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pages 1–6. IEEE, 2016.

[15] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Knowledge-driven event embedding for stock prediction. In *Proceedings of coling 2016, the 26th international conference on computational linguistics: Technical papers*, pages 2133–2142, 2016.

[16] Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems (TOIS)*, 37(2):1–30, 2019.

[17] Daiki Matsunaga, Toyotaro Suzumura, and Toshihiro Takahashi. Exploring graph neural networks for stock market predictions with rolling window analysis. *arXiv preprint arXiv:1909.10660*, 2019.

[18] Yingmei Chen, Zhongyu Wei, and Xuanjing Huang. Incorporating corporation relationship via graph convolutional neural networks for stock price prediction. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1655–1658, 2018.

[19] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[21] Qinkai Chen. Stock movement prediction with financial news using contextualized embedding from bert. *arXiv preprint arXiv:2107.08721*, 2021.

[22] Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, 2019.

[23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[24] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Neural Information Processing Systems (NIPS)*, pages 1–9, 2013.

[25] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[26] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Using structured events to predict stock price movement: An empirical investigation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1415–1425, 2014.

[27] Liheng Zhang, Charu Aggarwal, and Guo-Jun Qi. Stock price prediction via discovering multi-frequency trading patterns. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 2141–2149, 2017.

[28] Wei Bao, Jun Yue, and Yulei Rao. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PloS one*, 12(7): e0180944, 2017.

[29] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Deep learning for event-driven stock prediction. In *Twenty-fourth international joint conference on artificial intelligence*, 2015.

[30] Kira Radinsky, Sagie Davidovich, and Shaul Markovitch. Learning causality for news events prediction. In *Proceedings of the 21st international conference on World Wide Web*, pages 909–918, 2012.

[31] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D Manning. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 344–354, 2015.

[32] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[33] Shumin Deng, Ningyu Zhang, Wen Zhang, Jiaoyan Chen, Jeff Z Pan, and Huajun Chen. Knowledge-driven stock trend prediction and explanation via temporal convolutional network. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 678–685, 2019.

[34] Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. K-bert: Enabling language representation with knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2901–2908, 2020.