ATLANTIS PRESS

Advances in Computer Science Research, volume 92

International Conference on Mathematics, Big Data Analysis and Simulation and Modeling (MBDASM 2019)

A Semantic Path Based Approach to Match Subgraphs from Large Financial Knowledge Graph

Ziao Wang[1], Xiaofeng Zhang[2] and Yang Hu[3]
[1]Harbin Institute of Technology, Shenzhen, China
[2]Harbin Institute of Technology, Shenzhen, China
[3]Data Energy Union Technology Inc, Shenzhen, China

***Abstract*—In the past, people studied the stock market based on the assumption that the stock entity is known to be affected by the news. However, due to this assumption, these methods inevitably ignore the news without stock entities, and many news without stock entities will also have a significant impact on financial markets. In order to solve this problem, this paper proposes a subgraph matching algorithm based on semantic paths. Matching subgraphs on a knowledge graph that collects a large amount of stock market information and matching the affected stock entities from the semantic level can make a comprehensive analysis on Various news with or without entities. The main research work and achievements of this paper are as follows: First, starting with structured data, the paper complements semi-structured data and unstructured data to build a knowledge graph of the stock market and covering most of the stock market entities. Secondly, based on the analysis of LDA topic model, this dissertation extracts useful topics from financial news and constructs a news graph. A subgraph matching algorithm based on semantic path is proposed. From the knowledge graph, subgraphs matching with news graph are searched for mining the associated entities in the financial news. Finally, according to the result of subgraph matching, experiments and simulated investment are designed. The strategy achieved 15.96% excess return relative to the benchmark. The effectiveness of the subgraph matching algorithm based on semantic path is verified, and the feasibility of the algorithm in actual investment is proved.**

Keywords—knowledge graph; subgraph matching; stock prediction; related entities mining

I. INTRODUCTION

In 1970, Fama [1] proposed the Efficient Markets Hypothesis, which pointed out that in an effective market, people cannot obtain excess returns through known information. Financial markets are divided into three types by efficient market hypotheses: strong effective markets, semi-strong effective markets, and weak effective markets. However, when using effective markets hypothesis to explain financial phenomena, many scholars have encountered unexplained phenomena. For example, the actual rate of return in the Chinese market far exceeds the risk-free rate, and the equity risk premium is high, indicating that there are the "fair of equity premium" discovered by Mehra and Prescott [2] in the Chinese stock market. Rozeff [3] conducted an experiment on the US stock market index from 1904 to 1974. The experiment results show that the yield in January is significantly higher than other months. There are many phenomena that traditional finance cannot explain. The market is not completely effective, and the asymmetry of information will bring excess returns.

As early as 1971, Niederhoffer [4] showed that the volatility of financial market would be affected by news in the New York Times. Keown and Pinkerton [5] studied the impact of the company's earnings-related news on stock returns. Mitchell [6] pointed out that the Dow Jones announcement has a direct impact on several stocks. These early studies revealed the link between news and stock market movements. Later, more scholars were working to find out the impact of news on financial markets. For example, Cutler et al. [7] found that one-third of stock market movements were affected by news. Chan [8] monitored the monthly returns of companies included in the news headlines, and the study showed that the impact of negative news can last up to twelve months. All these studies have revealed that news reports have a certain degree of impact on the price of financial market targets. These studies are trying to understand the impact of literal information on the stock market, but their way of quantifying news is rough. The existing ways to quantifying news can be roughly divided into two categories, one is to record the number of times the company has been mentioned in the news, and the other is to record the positive and negative words in the company's related news, and thus to judge the impact of the news to the stock.

To overcome these problems, in the computer science field, people use text mining and machine learning techniques to quantify news text and predict stock prices. Some methods such as rule-based mining, KNN, and neural networks have been used to predict the price of stocks. Antweiler and Frank [8] extracted news from Yahoo Finance, using Naive Bayes and support vector machine algorithms to encode news as bullish, bearish or unaffected and use news information to predict market activity. Feuerriegel[9] and others used the LDA document theme generation model to automatically identify news topics, and studied the impact of different topics on stocks in different industries for the German stock market. Ding [10] et al. used a tensor neural network to extract the features of the subject-predicate relationship in the news headlines, and used neural networks to study the short-term and medium-term effects of events on stocks. Peng [11] and others combined the correlation graphs in the stock market and used neural networks to predict the rise and fall of stock price.

Most of the above research is aimed at study the impact of news on stocks, but they have ignored some issues, that is, the correspondence between news and stock entities are not always specific. The existing research can be divided into two types, one is to know the news corresponding to the stock, and then to study the trend of the specific stock in a certain time; another is to find keywords related to the stock from the title or content of the financial news, such as the company name, the stock code, etc. and thus correspond the keyword to the stocks, which has great limitations. In fact, many news does not mention specific companies or stocks, but they have a great impact on many stocks. This paper studies the mining of related entities in financial news. The main problem is how to identify the entities affected by financial news. According to the main problems, the research will be divided into two parts: First, the paper constructs the financial market knowledge graph, it can clearly express the intricate relationship in the financial market, and also provides an important tool for the second part of the research. In the second part, this article will show the method to find related financial entities when financial news occurs. In this part, subgraphs will be extracted from financial news and subgraph matching will be performed in the knowledge graph constructed in the first part of research. The main research framework of this paper is shown in Figure I.

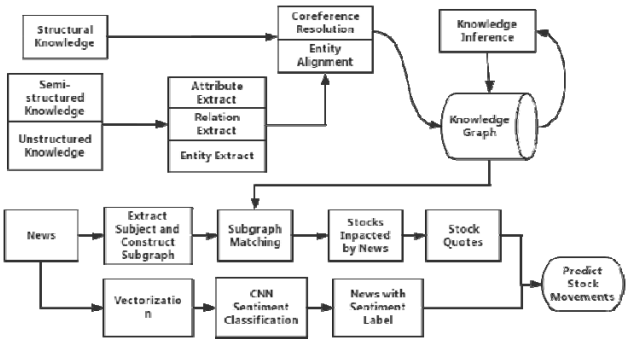

FIGURE I. RESEARCH FRAMEWORK

II. FINANCIAL KNOWLEDGE GRAPH FROM MULTIPLE DATA SOURCE

In this section, the detail of constructing the financial knowledge graph from multiple data source will be shown. There are two ways to construct a knowledge graph: top-down and bottom-up. The top-down method first defines the data pattern of the knowledge graph and then gradually refines the concept. Adding entities into the knowledge graph after forming a complete system. The bottom-up approach is just the opposite. It starts with the entities and gradually concretize abstractions. Neither method starts from scratch. The former uses some structured data and the latter uses existing entities in Wikipedia or Baidu Encyclopedia.

The overall process of constructing the knowledge graph in this paper is shown in the upper half of Figure I. Structured knowledge is used as the starting point to construct the data schema of the knowledge graph and then extract entities, relationships, and attributes from semi-structured and unstructured data. Finally, integrate knowledge graph using coreference resolution and entity alignment.

A. Data Source

Both structured data and unstructured data are used to construct the knowledge graph. Structured data refers to data stored in a relational database or an object-oriented database. The relational database uses a relational model, and its structure is a two-dimensional structure of table, which is very intuitive. The purpose of using structured data in this paper is to further expand other data based on these structured data. In order to ensure the accuracy of the knowledge graph, authoritative data sources are needed. In the field of financial and financial data, Wind Information has the highest market acceptance, and the data it provides is also the most recognized in the industry.

Semi-structured data includes open-link data and open knowledge base data, which are usually stored in graphs, most notably DBPedia [12], YAGO [13], and Freebase [14]. In the Chinese language field, Zhishi.me [15] is a knowledge base for extracting structured data from open encyclopedia data. It integrates three Chinese encyclopedias, including Baidu Encyclopedia, Interactive Encyclopedia and Wikipedia and it has over 5 million entities. This article will extract relevant knowledge from Zhishi.me as an extension of the knowledge graph.

Unstructured data mainly refers to text information. A large amount of text information on the internet can be used as the source of knowledge graph. The main source of text information in this paper is financial news. Financial news contains huge amount of entities, relationships and attributes, which is a good addition to the knowledge graph. This article has obtained a large amount of financial news by climbing financial websites, including Sina Finance, Netease Finance, etc. The total amount of news data is nearly 3 million.

B. Knowledge Extraction

Knowledge extraction is the most basic and crucial step in building knowledge graph. The key question is how to extract information from different data sources to obtain related entities. Structured information and open information extraction will be used to complete knowledge extraction.

According to the structured knowledge obtained, the data schema of the knowledge graph is defined. In knowledge graph, the nodes represent the entities. Therefore, the information collected in the above such as company, region, industry are defined as the node. The intrinsic property of an entity is defined as attribute, which is represented in the graph as a company's abbreviation, English name, stock code, and so on. The edges in the knowledge graph represents the relationships between entities, such as the relationship between a company and an industry sector.

As for unstructured data, entities, relationships and attributes are extracted from financial news. The first thing to do is the extraction of entities. The main technique used in this paper is the named entity recognition technology. The quality

of entity extraction has a great impact on the subsequent knowledge acquisition efficiency and quality. Therefore, in order to improve the accuracy of recognition, this paper constructs a financial entity dictionary. The dictionary includes information such as the full name and abbreviation of the domestic company, the name of the industry, and the name of the company's executives. The entity extraction from text corpus will generate many entities that have no relationship with each other. In order to get relevant semantic information, the relationship between entities are extracted from related corpus, and entities are connected through these associations. In order to form a structured knowledge system with network characteristics, this paper adopts the open domain extraction (OIE) framework proposed by Banko [16] et al. The prototype of this framework is TextRunner, which is based on self-monitoring learning method, the system inputs a part of the artificially labeled data for training, the training result is an entity relationship classification model, and then uses the model to classify the unstructured data, and trains the naive Bayesian model according to the classification result to identify the "entity-relationship-entity" triples.

C. Knowledge Fusion Based on Vector Space Model

Knowledge extraction can obtain associated entities and relationship information from unstructured and semi-structured information, but a large amount of error and redundant information may be included in these results, so it needs to be cleaned and integrated.

Entity linking is the operation of linking an entity object extracted from text to its corresponding entity object in the knowledge base. The basic idea of entity linking is to first select a set of possible matching entity objects from the existing knowledge base according to the extracted entity name, and then link the name with the corresponding entity by calculating the similarity between the entities. First, entity disambiguation is required. The referential item of a named entity can correspond to multiple entity concepts. Disambiguation needs to map the ambiguous reference item to the entity concept it refers to. For example, as shown in Figure II, based on the contextual information, "Apple" and "Jobs" are identified as "Apple (Company)".

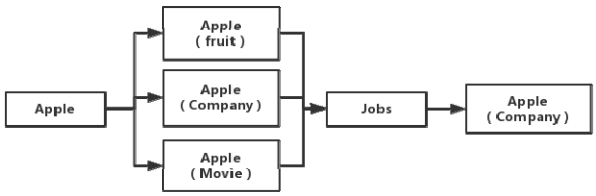

FIGURE II. ENTITY ALIGNMENT

This paper used the idea of Vector Space Model (VSM) to carry on entity alignment. The vector space model expresses text as a vector. A series of keyworks can be used to describe a sentence:

$$\mathrm{V(d)} = (t_1 w_1(d); \ldots ; t_n w_n(d)) \tag{1}$$

Where $t_i (i = 1,2, \ldots, n)$ are words that are different from each other, $w_i(d)$ is the weight of t_i in d, it can be expressed as the frequency of t_i in d:

$$w_i(d) = \psi(tf_i(d)) \tag{2}$$

The N sentences in the vector space can be described as a matrix, where any of which express is the weight of a word in the text:

$$N = \begin{pmatrix} & T_1 & T_2 & & T_l \\ D_1 & d_{11} & d_{12} & \cdots & d_{1l} \\ D_2 & d_{21} & d_{22} & & d_{2l} \\ & \vdots & & \ddots & \vdots \\ D_m & d_{m1} & d_{m2} & \cdots & d_{ml} \end{pmatrix} \tag{3}$$

After feature extraction, different weights must be given to each word because each word occupies a different degree of importance in the entity. The weight of each word in the vector is calculated using TF-IDF which is the product of the word frequency TF and the inverted document frequency IDF:

$$tfidf_{t,i} = tf_{t,i} \times idf_t \tag{4}$$

Where TF indicates the frequency at which a keyword appears and IDF indicates the number of documents and the logarithm of the quotient containing the word in the document:

$$\mathrm{IDF} = \log_2 \frac{|D|}{|w \in d|} \tag{5}$$

Where $|D|$ represents the number of all documents, $|w \in d|$ represents the number of documents containing the word w and the larger the TF-IDF is the more important of the word to the entire document.

By assigning weighted vector matrices, the similarity between vectors can be described by their corresponding vector angles, and the similarity formulas of documents D_1 and D_2 are:

$$\mathrm{sim}(D_1, D_2) = cos\theta = \frac{\sum_{k=1}^{n} w_k(D_1) \times w_k(D_2)}{\sqrt{(\sum_{k=1}^{n} w_k^2(D_1)) \times (\sum_{k=1}^{n} w_k^2(D_2))}} \tag{6}$$

In general, entity disambiguation first calculates the entity similarity. By the summary information extracting from Baidu Encyclopedia, constructing the feature vector, and measuring the co-occurrence degree of each feature word and entity

respectively, assign different weights through TF-IDF. Calculate the weight of the entity in the vector, and determine whether the disambiguation is needed by calculating the similarity between the two entity vectors.

III. SUBGRAPH MATCHING BASED ON SEMANTIC PATH

In this chapter, the process of constructing knowledge graph will be described in detail. A sub-graph matching algorithm based on semantic path will be proposed, a news graph will be constructed through LDA topic clustering and subgraph matching algorithm will be performed in the knowledge graph.

A. News Graph Construction Based on LDA Topic Clustering

The news graph is built by topic as node and semantic distance between different topic as edge where topic is obtained by LDA clustering. Based on LDA, the text can be abstracted to the topic level instead of just using words as features. It can express the semantic meaning of the text content. The schematic diagram of LDA topic clustering of news is shown in Figure III.

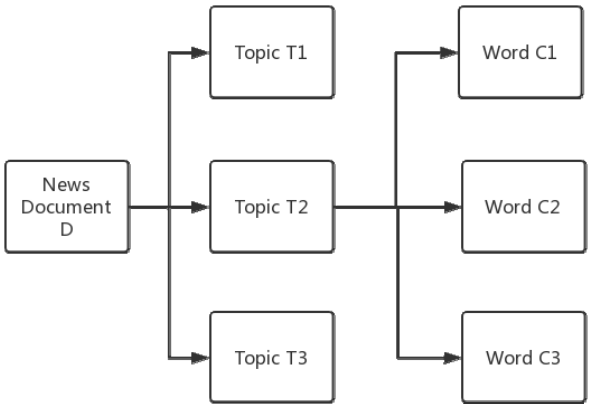

FIGURE III. LDA TOPIC CLUSTERING

The basic process of LDA topic clustering can be described as:

1) Select a document d according to the prior probability P(d)

2) Generating the topic distribution θ of document d from Dirchelet distribution α

3) Sampling from the polynomial distribution θ of the topic to generate the topic of the document d

4) Generating the word distribution φ of topic z from Dirichlet distribution β

5) Generating word w from the polynomial distribution φ

The LDA model cannot be solved directly because it contains multiple hidden variables. Generally, there are two commonly used parameter estimation methods for this problem. One is solved through variational method, and the other is by Gibbs sampling. The Gibbs sampling method takes more time in the solution process than the first variation method. According to the process of the LDA topic model, the training of the model will get a text distribution corresponding to the topic and the corresponding word distribution under the topic.

After completing the LDA topic model extraction of the news document, document D can be express as $D(c_1, c_2, .., c_n)$, where c_t represents the i-th node of the document and its physical meaning is the i-th topic of the news document. For the weights between news graph nodes $W(c_t, c_j)$, this paper proposes a weight measuring method named Weighted Jaccard Distance adjusted by cosine similarity:

$$W(c_t, c_j) = 1 - \frac{\sum_{a_t \in c_t, a_j \in c_j} \cos(\vec{a}_t, \vec{a}_j)_{\geq \tau} \left(w(a_t) + w(a_j)\right)}{\sum_{a_t \in c_t} w(a_t) + \sum_{a_j \in c_j} w(a_j)} \quad (7)$$

Where c_t and c_j are two random nodes in the graph, $(a_1, a_2, \ldots, a_m)$ represents the subject words and $\cos(\vec{a}_t, \vec{a}_j)$ represents the semantic distance between a_1 and a_j, this paper uses word2vec to vectorize the words and the cosine distance to represent the semantic distance of two words, $w(a_t)$ represents the weight of the subject word in the topic, it can be calculated through LDA topic clustering. Finally, the representation of a news graph is $D(V', E', W')$, where $V'(c_1, c_2, .., c_n)$ represents the nodes of the news graph, $E' \subseteq V' \times V'$ represents the edge of the news graph and $W'(c_{tj})$ represents weight of the edges. In the comparison of the number of topics extracted by the news, this paper finds that the three themes can better represent the news documents, and there is no excessive redundancy concept. Therefore, three vertex news graphs will be used through the entire paper. In order to better match the subgraphs in the knowledge graph, remove the edge with the smallest weight in the news graph, and represent the news graph as the shape shown in Figure IV.

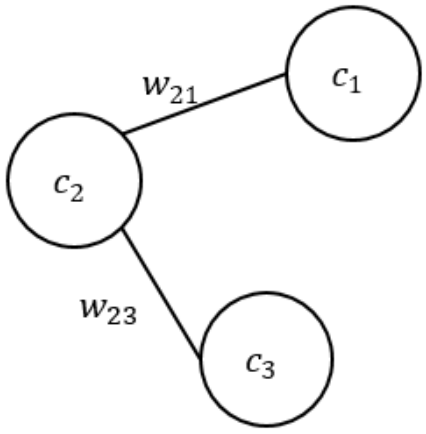

FIGURE IV. NEWS GRAPH

B. Subgraph Matching Algorithm Based on Semantic Path

The financial knowledge graph contains a large number of entities and complex relationships in the stock market. This

paper hopes to find the resonance of financial news and entity by means of matching news graph with knowledge graph. Therefore, this paper proposes a subgraph matching algorithm based on semantic path to complete the research from news to stock entity mapping.

1) The defination of subgraph matching based on semantic path problem

This part will define the subgraph matching problems based on semantic path. Continuing the representation of the previous section, the definition of subgraph matching based on semantic path:

For a news graph D with 3 vertices and a knowledge graph G, a matching subgraph X of G contains 3 vertices (v_1, v_2, v_3) in G, if the following conditions are met:

a) There is a mapping function f, for any random vertices c_i in D and v_i in G, $f(c_i) \sim v_i$ is true

b) For two edges (c_2, c_1) , (c_2, c_3) and their weights w_{21} , w_{23} in D, threre are two shortest path $B_1\ (f(c_2) \rightarrow f(c_1))$, $B_2\ (f(c_2) \rightarrow f(c_3))$ in G and their weights $W(B_1)$, $W(B_2)$, and $\frac{\frac{w_{21}}{w_{23}}}{\frac{W(B_1)}{W(B_2)}}$ is greater than the threshold τ.

Wherein, the mapping function is defined by the semantic distance of the node v_i in G to the node c_j in D. Since the concept node in the news graph D is composed by the keyword, wherein each keyword contains a weight, so the semantic distance SD can be calculated, and the mapping function is defined by the semantic distance.

$$SD(c_i, v_i) = \frac{\sum_{j=(1,2,\ldots,m)} \cos(\vec{v}_i, \vec{a}_j)}{m} \quad (8)$$

Where c_i is node in news graph D and can be represented as $(\vec{a}_1, \vec{a}_2, \ldots, \vec{a}_m)$, v_i is node in knowledge graph G.

The weight on the shortest path is defined as the average distance between the vertices:

$$W(B) = \frac{\sum_{1 \le i \le k-1} \cos(\vec{u}_i, \vec{u}_{i+1})}{k-1} \quad (8)$$

Where u_1 and u_k are the beginning and ending of the shortest path B who can be represented as $(u_1, u_2, \ldots, u_k)$.

2) The Algorithmn of Subgraph Matching Based on Semantic Path

According to the definition in the previous section, the candidate set that satisfies the condition for each vertex of the news graph will first be found out. And then calculate the shortest path between the three vertices for each vertex of the candidate set. According to the ratio of the shortest path weights and the closeness the weights in the news graph, a threshold is set, the subgraph that satisfies the threshold is the final matching subgraph. The subgraph matching algorithm based on the semantic path is described as follows:

Algorithm: subgraph matching based on the semantic path

Input: News Graph D, it vertices $V'(c_1, c_2, c_3)$, edge weights (w_{21}, w_{23}) and knowledge graph G

Output: the matching subgraph X in G of News Graph D

1. For node c_i in D, Find all nodes who satisfy condition $f(c_i) \sim v_i$ as candidates: $Candidate(c_1) = \{v_1^1, v_2^1, \ldots, v_m^1\}$, $Candidate(c_2) = \{v_1^2, v_2^2, \ldots, v_n^2\}$ $Candidate(c_3) = \{v_1^3, v_2^3, \ldots, v_k^3\}$;
2. For each Candidate(c_2):
3. For each Candidate (c_1):
4. For each Candidate (c_3):
5. find shortest path B_1, B_2 from c_2 to c_1 and c_2 to c_3
6. calculate weights of shortest path $W(B_1)$, $W(B_2)$
7. IF $\frac{\frac{w_{21}}{w_{23}}}{\frac{W(B_1)}{W(B_2)}} \ge \tau$:
8. find matching graph output(X)

IV. EXPERIMENT

This section introduces the specific details of the implementation of the subgraph matching algorithm based on semantic path, and designs experiment to verify the effectiveness of the algorithm. The process of the experiment is shown in Figure V. The news graph is extracted from the news corpus through LDA topic clustering. Then match the subgraph through the subgraph matching algorithm proposed in this paper, and use the matching result to simulate an investment in stock market.

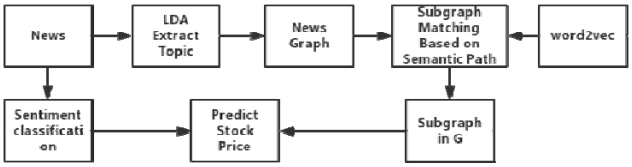

FIGURE V. EXPERIMENT PROCESS

A. Dataset

The experimental data used in this chapter are news text data, the knowledge graph constructed in the third chapter of this paper and the historical data of stocks. The news text data comes from web crawlers. This article has climbed nearly 3 million financial news from 2000 to 2015 from major financial websites. The attribute of the news is shown in Table I.

TABLE I. NEWS DATASET

	Attribute
time	2000-2017[a]
Total	3817209
Minimum length	55
Maximum length	30000
Mean length	1273

B. Experiment result

According to the semantic path-based subgraph matching algorithm implemented in this paper, in the case of giving a financial news, the algorithm can analyze the stock entities that the news will specifically affect. In this section, the subgraph matching algorithm based on the semantic path is shown, experiments are designed, other algorithms are compared, real-world investments are simulated and the value of algorithms in real-world investments are verified.

In order to verify the accuracy of the sub-graph matching algorithm based on semantic path, this paper first designs an experiment, and extracts the entities from the news as a classification problem, and uses the known classification method to compare with the method of this paper. This experiment selects 30 stock-related news as a data set, with pre-2017 news as a training set and 2017 news as a test set. The information about the news data set is shown in Table II.

TABLE II. COMPARATIVE EXPERIMENT DATASET

	Attribute
time	2015-2017[a]
total	47224
Training set	37387
Testing set	9837
Related stock number	30

The comparison classification algorithms used in this paper include SVM classification, decision tree (DT) classification, random forest (RF) classification, single-layer fully connected network (FCN) classification, naive Bayes(NB) classification, and convolutional neural network(CNN) classification algorithm. This paper uses commonly used classification evaluation indicators to evaluate the algorithm, including Accuracy, Precision, Recall and F1. The algorithm of this paper is based on the subgraph matching result of stock entity related to the news as the classification result, and compared with other methods. The experimental results are shown in Table III.

TABLE III. COMPARATIVE EXPERIMENT RESULT

	Accuracy	Precision	Recall	F1 Score
SVM	0.46	0.52	0.46	0.43
DT	0.27	0.30	0.27	0.27
RF	0.38	0.37	0.38	0.31
NB	0.42	0.51	0.42	0.42
FCN	0.27	0.47	0.27	0.32
CNN	0.45	0.28	**0.54**	0.37
OUR	**0.49**	**0.71**	0.48	**0.57**

From Table III, it can be found that the accuracy of the algorithm proposed in this paper exceeds SVM, decision tree, random forest, fully connected network, naive Bayes and CNN classification method. Although the recall rate is not as good as CNN classification, it is much better than other methods in precision and F1 value.

In order to verify the value of the proposed algorithm in the actual investment, this experiment selects the news of the whole year of 2014 and mark the news emotionally, extract the news graph, and perform the subgraph matching algorithm in the knowledge graph to find the stocks affected by the news. The specific details of the strategy designed in this paper are shown in Figure VI. On the day of the financial news happen, emotions are marked for each news, the news graph is extracted, and the subgraphs are matched in the knowledge graph to find out the stocks that have positive influence. The stock is bought at the market price after the opening of the next day, and the position is allocated equally for the stocks bought, and sold after one day.

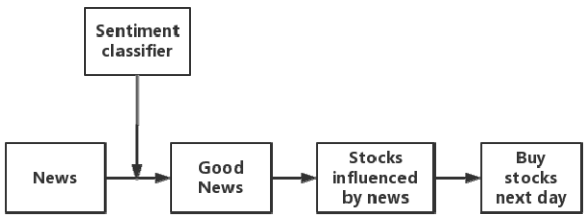

FIGURE VI. STRATEGY PROCESS

At the same time, in order to verify the effectiveness of the algorithm, a comparative experiment is designed. SVR (Support Vector Regression) [17] and GBRT (Gradient Boosting Regression Tree) [18] are used to design strategies that are often used for stock market forecasting. SVR and GBRT are used to regress against the market index and predict the next day's market trend. If the trend is positive, the market index will be held on the next day, otherwise the market index will be sold. We compare the market index returns, SVR strategy returns, GBRT strategy returns, and the benefits of the proposed algorithm for all A-shares during the corresponding period. In the case of holding a day to sell, the detailed benefits

of the algorithm and comparison strategy are shown in Table IV.

TABLE IV. EXPERIMENT RESULT

month	Our	SVR	GBRT	Stock index
1	-0.0233	-0.0170	-0.0076	-0.0548
2	-0.0132	0.0011	0.0095	-0.0107
3	-0.0229	-0.0179	-0.0240	-0.0150
4	0.0126	0.0046	0.0008	0.0058
5	-0.0137	-0.0023	-0.0096	-0.0010
6	-0.0049	-0.0004	0.0006	0.0040
7	0.0627	0.0629	0.0592	0.0855
8	-0.0106	-0.0153	-0.0145	-0.0051
9	0.0414	0.0245	0.0410	0.0482
10	0.0497	0.0231	0.0248	0.0234
11	0.2531	0.1458	0.1803	0.1198
12	0.2814	0.2766	0.2713	0.2581
total	**72.71%**	55.39%	62.02%	51.66%

It can be concluded from Table IV that the semantic map-based subgraph matching algorithm proposed in this paper exceeds the market and comparison algorithm in terms of revenue, and achieves 21.05% more than the market and 17.32% more than SVR. Compared with GBRT, the revenue is 10.69% higher than that of GBRT.

V. CONCLUSION

Based on the relationship between financial news and financial assets, this paper conducts research on related entities mining in financial news, crawls a large amount of financial news and constructs financial market knowledge graph, and proposes a sub-graph matching algorithm based on semantic path. The problem of the corresponding influence relationship between financial news and financial assets are solved. In order to verify the effectiveness of the proposed algorithm, this paper designs experimental simulations of financial market investment and has achieved good results. The main contributions of this paper are as follows:

(1) Established a financial market knowledge graph, and describe the intricate relationship of financial markets in the form of knowledge graph.

(2) A sub-graph matching algorithm based on semantic path is proposed to solve the corresponding influence relationship between financial news and financial asset targets.

ACKNOWLEDGMENT

This paper is partially supported by Guangdong Province Science Project No: 2017B090901022, the National Science Foundation of China under grant No.61872108, and Shenzhen Science and Technology Program under Grant No.JCYJ20170811153507788.

REFERENCES

[1] Fama, Eugene F. "Papers and Proceedings of the Twenty-Eighth Annual Meeting of the American Finance Association New York, N.Y. December, 28-30, 1969 || Efficient Capital Markets: A Review of Theory and Empirical Work." The Journal of Finance 25.2(1970):383-417.

[2] Mehra, R. , and E. Prescott . "The equity premium: a puzzle." Levine's Working Paper Archive 15.2(2010):145-161..

[3] Rozeff, Michael S. , and W. Kinney . "Capital market seasonality: The case of stock returns." Journal of Financial Economics 3.4(1976):379-402.

[4] Niederhoffer, Victor . "The Analysis of World Events and Stock Prices." The Journal of Business 44.2(1971):193-219.

[5] Keown, Arthur J. , and J. M. Pinkerton . "Merger Announcements and Insider Trading Activity: An Empirical Investigation." The Journal of Finance 36.4(1981):855-869..

[6] Mitchell, Mark L. , and J. H. Mulherin . "The Impact of Public Information on the Stock Market." The Journal of Finance49.3(1994):923-950.

[7] Cutler, David M., James M. Poterba, and Lawrence H. Summers. "What moves stock prices?." (1988).

[8] Antweiler, Werner, and Murray Z. Frank. "Is all that talk just noise? The information content of internet stock message boards." The Journal of finance 59.3 (2004): 1259-1294.

[9] Feuerriegel, Stefan, Antal Ratku, and Dirk Neumann. "Analysis of how underlying topics in financial news affect stock prices using latent dirichlet allocation." 2016 49th Hawaii International Conference on System Sciences (HICSS). IEEE, 2016.

[10] Ding, Xiao, et al. "Deep learning for event-driven stock prediction." Twenty-fourth international joint conference on artificial intelligence. 2015.

[11] Peng, Yangtuo, and Hui Jiang. "Leverage financial news to predict stock price movements using word embeddings and deep neural networks." arXiv preprint arXiv:1506.07220(2015).

[12] Auer, Sören, et al. "Dbpedia: A nucleus for a web of open data." The semantic web. Springer, Berlin, Heidelberg, 2007. 722-735.

[13] Suchanek, Fabian M., Gjergji Kasneci, and Gerhard Weikum. "Yago: A large ontology from wikipedia and wordnet." Web Semantics: Science, Services and Agents on the World Wide Web 6.3 (2008): 203-217.

[14] Bollacker, Kurt, et al. "Freebase: a collaboratively created graph database for structuring human knowledge." Proceedings of the 2008 ACM SIGMOD international conference on Management of data. AcM, 2008.

[15] Zhu, Jiangang, et al. "Building a Large-scale Software Programming Taxonomy from Stackoverflow." SEKE. 2015.

[16] Yates, Alexander, et al. "Textrunner: open information extraction on the web." Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations. Association for Computational Linguistics, 2007.

[17] Smola, Alex J., and Bernhard Schölkopf. "A tutorial on support vector regression." Statistics and computing 14.3 (2004): 199-222.

[18] Friedman, Jerome H. "Stochastic gradient boosting." Computational statistics & data analysis 38.4 (2002): 367-378.