TMUNLP at the NTCIR-15 FinNum-2

Yu-Chi Liang
Graduate Institute of Data Science
Taipei Medical University, Taiwan
m946108001@tmu.edu.tw

Yi-Hsuan Huang
Graduate Institute of Data Science
Taipei Medical University, Taiwan
m946108002@tmu.edu.tw

Yu-Ya Cheng
Professional Master Program in Data Science
Taipei Medical University, Taiwan
i906108009@tmu.edu.tw

Yung-Chun Chang
Graduate Institute of Data Science
Taipei Medical University, Taiwan
changyc@tmu.edu.tw

ABSTRACT

Machine learning methods for financial document analysis have been focusing mainly on the textual part. However, the numerical parts of these documents are also rich in information content. This paper presents our approach in Numeral Attachment in Financial Tweets (FinNum-2) at NTCIR-15. The purpose of this task is to determine whether there is a relationship between the target cashtag and the target number in financial tweets. We construct a model based on BERT-BiLSTM with attention mechanism, which is our main architecture of this task. In addition, we also add the results from a dependency parser and a CNN model to our main architecture. Our experimental results indicate that the BERT-BiLSTM with attention model has the best performance. More precisely, we obtain 87.02% in development set and 64.74% in test set in terms of F-score.

KEYWORDS

BERT, BiLSTM, Dependency grammars

1. Introduction

With the development of artificial intelligence (AI), major industries are investigating the possibility of applying these new technologies to domain-specific material. In order to improve service efficiency, the financial industry has developed a novel research field, financial technology (FinTech). FinTech can be applied in many places such as mobile payment, cloud platform and cloud big data, etc. As the popularity of FinTech topics continues to rise, it has now gone beyond previous examples and joined forces with natural language processing (NLP) technology [4].

By analyzing financial text data, we can obtain substantial and useful information related to finance. For instance, financial sentiment analysis can help us obtain expert opinions on stock market trends [5]. In addition, it can also use historical records of the contact process between employees and customers to analyze inter-industry relationships. [6]

However, to understand the details of financial data, we must rely not only on analysis of the content, but also the numbers that may contain important information. Therefore, more attention has been paid to the processing of digital information in recent years [1]. In the past, the first finance-related task was held at NTCIR-14 conference, called the Fine-Grained Numeral Understanding in Financial tweet (FinNum). The goal of this task was to classify the type of numbers in the tweets. However, it was insufficient to understand the meaning of numbers, and resulted in less satisfactory classification outcome. In order to further analyze the meaning of numbers for stocks mentioned in tweets, the NTCIR-15 conference holds a new task called Numeral Attachment in Financial Tweets (FinNum-2). FinNum-2 mainly explores whether there is a correlation between numbers and *cashtags* on financial social media data. Since there may contain multiple cashtags and multiple numbers in a tweet, we need to determine whether there is a correlation between the given cashtag and the number. Taking Figure 1 as an example, a financial document may contain more than one number. The number '3' represents the cashtag $ BAC was at 3 a share. Therefore, the number '3' is describing the cash label $BAC. In contrast, the number '2009' is not the modifier of $BAC due to the fact that it denotes the year was in 2009 and has nothing to do with the cash label $BAC.

Remember 2009? $BAC was at 3 a share. The people of Greece all want $NBR to open to get back to normal. Tired of banking notes on Notepads

Figure 1: The relationship between the number and the cashtag in a tweet.

FinNum-2 is a binary classification problem. There are five columns in the data. The five columns are financial tweet, target number, target number location, target cashtag and relationship. Given a financial tweet, a participant needs to determine whether the target number and target cashtag in the financial tweet are related. The outputs are either "1" or "0," where "1" means the

†Corresponding author. Fax: +886-2-6638-2736 ext. 1184 (Y.C. Chang).
E-mail address: changyc@tmu.edu.tw (Y.C. Chang).

target number and the target cashtag are related. On the contrary, "0" means the target number and the target cashtag are not related.

2. Methodology

Considering the situation where the original data consists of financial tweets, which may contain noise, we have two preprocessing steps before putting the data into the models. First, we replace the cashtag, target number and URLs in the tweet with "TICKER", "NUM" and "URL", respectively. Next, we remove the emoji in the tweet. With the increasing development of social media platforms, the use of emoji has also become popular. Users express emotions through emoji in tweets. Although emojis in a tweet can express emotions, they do not affect the recognition result. Therefore, we can safely remove emojis from tweets.

After preprocessing, we input the normalized data to the model. There are 3 major rounds in our model. Figure 2 shows detailed architecture of each round. The first round is our basic model architecture consisting of BERT-BiLSTM with attention mechanism [2]. In the second round, we add dependency grammar information to the basic model through a 10-dimensional feature vector. The last round is to add the BERT embedding matrix to the CNN model and connect them with the second-round model.

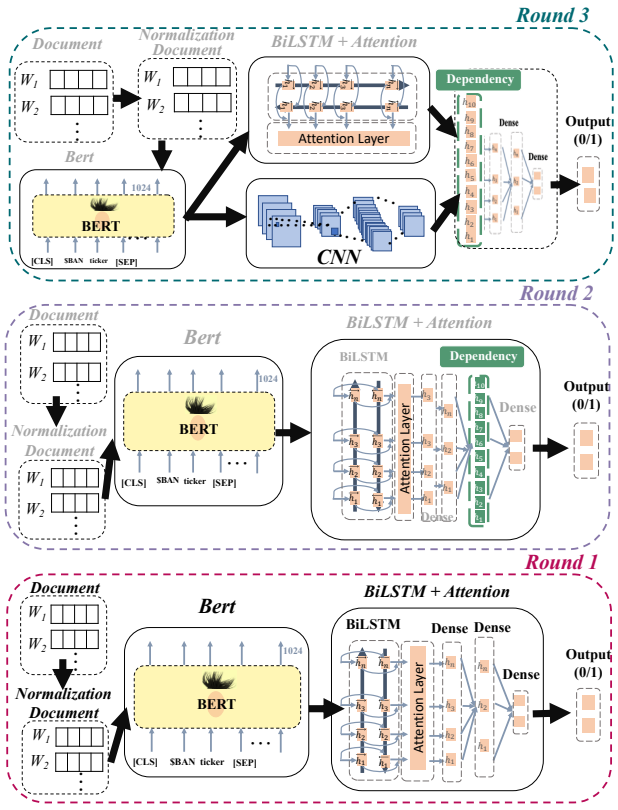

Figure 2: Model architecture of the three rounds

Round 1: BERT-BiLSTM with Attention Mechanism

Bidirectional Encoder Representations from Transformers (BERT) is a pre-trained language model that was trained using texts in an unsupervised manner.[10] The pre-trained models are available for download in the official BERT website[1]. In this task, we use the BERT-Large Uncased model. Uncased model requires that words should be converted to lowercase and the accent marks should be removed when using it. Through this model, we can convert the preprocessed financial tweets into 1024-dimensional vectors.

Bi-directional Long Short-Term Memory (BiLSTM) is a two-way concept that combines forward Long Short-Term Memory (LSTM) and backward LSTM. In addition to learning long-distance dependency, it can also better capture two-way semantic features.[8] Attention mechanism can help the model assign different weights to parts of the input and extract more critical information so that the model can make more accurate judgements. Therefore, placing the Attention layer on top of the BiLSTM layer can weigh the BiLSTM output results to improve performance.

In this round, the BERT embedding outputs are sent to the BiLSTM layer and then connected to the Attention layer. This architecture is our main architecture for this competition, and it is also the most effective of the three rounds.

Round2: Dependency Grammars-infused BERT-BiLSTM with Attention Mechanism

Dependency parser analyzes the relationship between each word in a sentence. There are two ways that we can find the dependency parser. The two methods are StanfordNLP and spaCy. In this task, we use the StanfordNLP toolkit as our dependency parser. Figure 3 shows the StanfordNLP dependency relationship in a sentence. The arrow that is connecting the words indicate that there is a dependency between the two words, and the relationship type is displayed above the arrow. We find the shortest path between the cashtag and the target number in each sentence through the dependency relationship. In the example in Figure 3, the original sentence can be shortened to "TICKER see 5." when the target number is 5. There are total of 19 relationships in all the shortest paths. Using these relationships, we can convert the shortest path into a 10-dimensional vector called the Dependency matrix. Specifically, we convert the relationships that appears in the shortest path into corresponding numbers. If the length of the shortest path is less than 10, we add zeros at the end until the dimension is 10. If the length of the shortest path is greater than 10, we truncate the first five relations and the last five relations. In this round, we add the 10-dimensional dependency matrix as extra

[1] https://github.com/google-research/bert

features to the model of the first round, and connect it before the last Dense layer. The purpose of this model is to learn more about the relationship between words in sentences through the dependency parser.

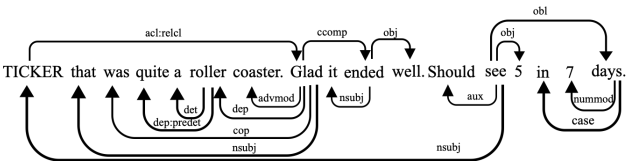

Figure 3: Example of StanfordNLP dependency relationship

Round3: Dependency Grammars-infused BERT-CNN-BiLSTM with Attention Mechanism

Convolutional Neural Network (CNN) [9] is constructed from multiple types of layers, such as a convolutional layer and a pooling layer. The convolutional layer captures the features between multiple consecutive words in a sentence through filters. The pooling layer takes the maximum value of the convolutional layer. It can transform sentences of different lengths into a fixed size matrix. CNN is widely used in image processing but it also has good performance in nature language processing. CNN is a feature extraction architecture and it can extract features from each paragraph in the documents.

We put the embedding matrix into the CNN model and connect it with the output of BiLSTM with Attention. The purpose of using CNN and BiLSTM is to combine the advantages of these two models to improve performance. The different from the second round is that this method connects the 10- dimensional dependency matrix after the Attention and CNN instead of the last Dense layer.

3. Results and Discussion

In the FinNum-2 task, there are a total of 10,340 sentences in the data set. Within it, 70% (7,187 entries) is set as the training set, 10% (1,044 entries) as the development set, and 20% (2,109 entries) as the testing set [3]. We use the macro F_1 score to evaluate the experimental results. Table 1 shows the results of two baseline methods (Majority and Caps-m) and the results of our three rounds. Our results outperform the majority method, no matter which round we compare. However, only round1 (TMUNLP -1) achieves better performance than both baseline methods. Within the experimental results of our three rounds, the round1 model, which uses the BERT embedding matrix as input into BiLSTM with Attention Mechanism has the best performance (87.02% in development set and 64.74% in test set).

In addition, as shown in Table 1, we found that there is a large difference between the performances of each of our models in predicting the development set and the test set. We postulate that the reason for this large difference in performance lies in the process of training the development model. During training, the test data is used as the validation data to find optimal model parameters in order to obtain better prediction results. However, when training for the test model, only a fraction of the training data is validated. Therefore, it is perceivable that a wide difference in development and test performance exists between these settings.

Table 1: Experimental results

Run	Macro-F1 score (%)	
	Development	**Test**
Majority	44.88	44.93
TMUNLP-3	**87.34**	58.40
TMUNLP -2	85.17	59.77
Caps-m	79.27	63.37
TMUNLP -1	87.02	**64.74**

In addition to the large difference in performance mentioned above, the three models that we propose are also worth exploring. In our expectation, the performance of round3 (TMUNLP-3) should be the best, followed by round2 (TMUNLP-2) and finally round1 (TMUNLP -1). It is surprising to see that the final result is contrary to what we have expected. Among these results, round1 (TMUNLP -1) has the best performance on both development and test sets. Compared to round1 (TMUNLP -1), the performance of round2 (TMUNLP -2) is 1.85% lower than that of round1 on development set and was 4.97% lower than round1 on test set. In our opinion, the reason that adding a 10-dimensional dependency matrix as additional feature to the model cannot improve the performance is as follows. Due to the fact that the tokens in each instance are separated by spaces to calculate the shortest paths, some of the target numbers may be included in a longer token and cannot be identified (e.g., '1.7' is a target number, but it's included in the '$1.5-1.7' token.). Therefore, the shortest path is replaced by the whole instance, which introduces misleading information. On the other hand, we speculate that the reason why round3 (TMUNLP -3) has the worst performance in all our results is that: the CNN model cannot fully capture the order and positional information of the words in the text sequence during the process of convolution and concatenation steps.

4. Conclusion

In this paper, we present a system that can distinguish whether there is a correlation between the numbers and cashtags on financial social media data. We compare three models, and the results show that the BERT embedding matrix as input into BiLSTM with Attention has the best performance, achieving about 64% in F_1 score. Future work mainly includes the following parts: (1) use other symbols, for instance, the dash "-" character, to cut sentences when calculating the shortest path; (2) design

novel model architectures to better incorporate the complex interactions between words in tweets.

ACKNOWLEDGMENTS

This research was supported by the Ministry of Science and Technology of Taiwan under grant MOST 107-2410-H-038-017-MY3, MOST 109-2410-H-038 -012 -MY2, and MOST 107-2634-F-001-005. In addition, this work was financially supported of the Higher Education Sprout Project by the Ministry of Education in Taiwan under grant DP2-109-21121-01-A-11, as well as funded by the grant of University System of Taipei Joint Research Program USTP-NTOU-TMU-109-03.

REFERENCES

[1] C. Chen, H. Huang, Y. Shiue and H. Chen. 2018. Numeral Understanding in Financial Tweets for Fine-Grained Crowd-Based Forecasting. 2018. IEEE/WIC/ACM International Conference on Web Intelligence (WI), Santiago, 136-143.

[2] Lung-Hao Lee, Yi Lu, Po-Han Chen, Po-Lei Lee and Kuo-Kai Shyu. 2019. NCUEE at MEDIQA 2019: Medical Text Inference Using Ensemble BERT-BiLSTM-Attention Model. In *Proceedings of the BioNLP 2019 workshop*. Italy, 528–532

[3] Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2019. Numeral Attachment with Auxiliary Tasks. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*. Paris, France.

[4] Chung-Chi Chen, Hen-Hsen Huang, Hsin-Hsi Chen. 2020. NLP in FinTech Applications: Past, Present and Future. arXive

[5] Sahar Sohangir, Dingding Wang, Anna Pomeranets and Taghi M. Khoshgoftaar. 2018. Big Data: Deep Learning for financial sentiment analysis. Journal of Big Data. 5(1),3. https://doi.org/10.1186/s40537-017-0111-6.

[6] Hiroki Sakaji, Ryota Kuramoto, Hiroyasu Mat-sushima, Kiyoshi Izumi, Takashi Shimada and Keita Sunakawa. 2019. Financial Text Data analytics framework for business confidence indices and inter-industry relations. In *Proceedings of the FirstWorkshop on Financial Technology and Natural Lan-guage Processing*. Macao, China. 40–46.

[7] Chung-Chi Chen, Hen-Hsen Huang, Hiroya Takamura, Hsin-Hsi Chen. 2019. Overview of the NTCIR-14 FinNum Task: Fine-Grained Numeral Understnding in Financial Social Media Data. In *Proceedings of the 14th NTCIR Conference on Evaluation of Information AccessTechnologies*.

[8] Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural networks 18, 5-6 (2005), 602–610.

[9] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Confer-ence on Empirical Methods in Natural Language Processing (EMNLP).*

[10] *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).*