TMUNLP at the NTCIR-15 FinNum-2

Yu-Chi Liang, Yi-Hsuan Huang, Yu-Ya Cheng, Yung-Chun Chang
Taipei Medical University, Taiwan
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
Graduate Institute of 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
Outline

A  Task Introduction

B  Methodology

C  Results

D  Discussion

E  Conclusion
Task Introduction

FinNum-2
FinNum is a task for fine-grained numeral understanding in financial social media data.

Task Definition
Given a target numeral and a cashtag, and we formulate the problem as a binary classification to tell if the given numeral is related to the given cashtag.

Datasets
Total: 10340
Training: 7187 (70%)
Development: 1044 (10%)
Testing: 2109 (20%)
Methodology

Normalization

1) Replace the cashtag, target number and URLs in the tweet with “TICKER”, “NUM” and “URL”, respectively.
2) Remove the emoji in the tweet.

2020/12/15
Round 1: BERT-BiLSTM with Attention Mechanism

1) We use the BERT-Large Uncased model. Through this model, we can convert the preprocessed financial tweets into 1024-dimensional vectors.

2) In this round, the BERT embedding outputs are sent to the BiLSTM layer and then connected to the Attention layer.
Round 2: Dependency Grammars-infused BERT-BiLSTM with Attention Mechanism
Round 2: Dependency Grammars-infused BERT-BiLSTM with Attention Mechanism

Example

Shortest Dependency Paths (SDP)

- **Dependency**
  Dependency parser analyzes the relationship between each word in a sentence.

- **Shortest paths**
  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.

“A TICKER that was quite a roller coaster. Glad it ended well. Should see 5 in 7 days.”

“TICKER see 5.”
Round 3: Dependency Grammars-infused BERT-CNN-BiLSTM with Attention Mechanism

[Diagram showing the flow of information through BERT, CNN, BiLSTM, and Attention layers.]
Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Development</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Majority</td>
<td>44.88%</td>
<td>44.93%</td>
</tr>
<tr>
<td>TMUNLP-3</td>
<td>87.34%</td>
<td>58.40%</td>
</tr>
<tr>
<td>TMUNLP-2</td>
<td>85.17%</td>
<td>59.77%</td>
</tr>
<tr>
<td>Caps-m</td>
<td>79.27%</td>
<td>63.37%</td>
</tr>
<tr>
<td>TMUNLP-1</td>
<td>87.02%</td>
<td>64.74%</td>
</tr>
</tbody>
</table>

2020/12/15
Large difference
There is a large difference between the performances of each of our models in predicting the development set and the test set.

The result is contrary to expectations
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.
Conclusion

1) **Use other symbols.** Ex.”-“
2) **TweeboParser:**
   A dependency parser for tweets.

Transformer BERT:
68.73%
Thank you !

*