Predict Stock Price Changes Using BERT with Financial News

source code: https://github.com/bamboochen92518/ADL-FINAL

方聿丞	張睿桓	陳竹欣	陳若瑜	鄭百里
b10902020	*b10902046*	*b10902005*	*b10902127*	*b10902101*

Abstract

This study aims to predict the stock price movements of the following day based on today's financial news. We employed three models provided by Hugging Face:

- bert-base-chinese
- hw2942/bert-base-chinese-finetuning-financial-news-sentiment-test
- hw2942/chinese-bigbird-wwm-base-4096-wallstreetcn-morning-news-market-overview-open-000001SH-v1

Fine-tuning was conducted using financial news data from January to October 2023, with model evaluation performed on financial news from November to December. Additionally, separate models were trained for each industry sector. Our findings reveal that the Computer Peripheral sector exhibited the best performance under the hw2942/bert-base-chinese-finetuning-financial-news-sentiment-test model, achieving an accuracy rate of 59.8%.

1. Introduction

Stock market investment involves four major aspects: fundamental analysis, technical analysis, chip analysis, and news analysis. Fundamental analysis relies on a company's financial statements to assess its capital structure, while technical analysis involves historical stock prices and various technical indicators to predict stock movements. Chip analysis observes trading records, but news analysis remains the most elusive aspect. Faced with the same news, interpretations vary among traders in the stock market, and each individual's level of information mastery differs. Therefore, the goal of this article is to predict stock price movements through financial news.

To simplify the problem, we exclude additional costs such as securities transaction tax and fees, focusing solely on whether the next closing price is higher or lower than the previous one. In this way, our labels are simplified to either 'rise' or 'fall,' transforming the problem into a binary classification issue.

Hugging Face's library conveniently provides functions for binary classification that we can directly utilize. In selecting pre-trained models, we tested three options:

- bert-base-chinese:
  without specific financial knowledge fine-tuning
- hw2942/bert-base-chinese-finetuning-financial-news-sentiment-test:
  with financial knowledge fine-tuning
- hw2942/chinese-bigbird-wwm-base-4096-wallstreetcn-morning-news-market-overview-open-000001SH-v1:
  with financial knowledge fine-tuning (capable of processing longer articles)

Regarding the industry sector selection, we opted for the Computer Peripheral sector. Compared to sectors like cement and food, the Computer Peripheral sector exhibits greater stock price fluctuations, which may be more conducive for model learning.

Finally, in the evaluation phase, we experimented with adjusting learning rates, batch sizes, and epochs to identify the best-performing model.

2. Related Works

2.1. bert-base-chinese [1]

This model has been pre-trained for Chinese, training and random input masking has been applied independently to word pieces (as in the original BERT paper).

As this model is simply the Chinese version of bert-base, without undergoing extensive fine-tuning with a substantial amount of financial news or related data, we initially assumed that our training data would be sufficient for the model to perform well after fine-tuning. However, the results indicate that our data is far from adequate. In Section 3.1, we will discuss the total amount of data we have collected.

2.2. hw2942/bert-base-chinese-finetuning-financial-news-sentiment-test [2]

This model is a fine-tuned version of bert-base-chinese on an unknown dataset.

While this model does not explicitly specify the data source used for fine-tuning, in Section 6, we observe that its performance surpasses that of bert-base-chinese. However, the model has a limitation. Since it still relies on bert as the base model, its max position embeddings is constrained to 512. Given that news articles typically exceed this length, we resort to employing a sliding window approach with strides = (0.25×window size) to segment and process the news.

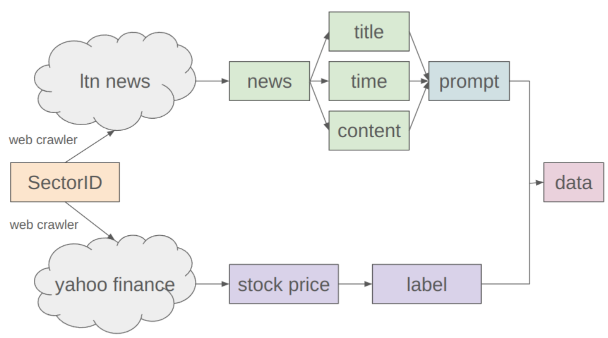
2.3. hw2942/chinese-bigbird-wwm-base-4096-wallstreetcn-morning-news-market-overview-open-000001SH-v1 [3]

This model is a fine-tuned version of Lowin/chinese-bigbird-wwm-base-4096 on the dataset of Wallstreetcn Morning News Market Overview with overnight index (000001.SH) movement labels.

Compared to the previous model, this model allows max position embeddings to be set at 4096, enabling it to handle longer input. In Section 6, We observe that in the absence of fine-tuning, the performance of the third model surpasses that of the second. However, after fine-tuning, the second model exhibits better performance with a more significant improvement, while the third model, at times, experiences a regression.

3. Data Collect and Preprocess

3.1. Data Collection



Picture 1. how we collect data

In our designed API, simply inputting the SectorID initiates a process where we first gather information on companies within that sector from Yahoo Finance [4]. Subsequently, we download relevant news from Liberty Times Net [5] and historical stock prices from Yahoo Finance [6] for the respective year. The collected news data includes fields such as 'title,' 'content,' 'time,' 'stock name,' and 'stock code.' Utilizing the 'time' field, we further retrieve the closing prices after the ex-dividend and ex-rights dates by cross-referencing with historical stock prices.

To simplify the problem, we exclude additional costs such as transaction taxes and fees, focusing solely on whether the stock rises or falls. If the next closing price is higher than the previous one, it is classified as a rise; otherwise, it is classified as a fall or no change. Figure 1 illustrates the schematic representation of our data collection process.

Ultimately, we collected 2,145 training data and 520 validation data.

3.2. Data Preprocessing

After gathering the data, we need to design prompts to concatenate information such as 'title,' 'content,' and 'time.' Since our task involves binary classification rather than text generation, providing examples is unnecessary. We opt for a zero-shot approach. In the end, the prompt we designed is

```
"[{time}] {title}\n 內文如下：{content}"
```

4. Train

In the training phase, we selected three pre-trained models for training: bert-base-chinese, hw2942/bert-base-chinese-finetuning-financial-news-sentiment-test, and hw2942/chinese-bigbird-wwm-base-4096-wallstreetcn-morning-news-market-overview-open-000001SH-v1. For fixed parameters, they include:

1) Shuffle Data:
Shuffling data is employed to introduce variability in each epoch and batch. When the input order remains the same, the content of each batch for every epoch also stays consistent. Compared to not shuffling data, the performance tends to be more rigid.
2) Max Position Embedding:
This parameter is predetermined by the pre-trained model. The first two models have a max position embedding of 512, while the third model has it set to 4096.
3) Optimizer:
We utilized the Adam optimizer with betas=(0.9, 0.999).

In the evaluation phase, we adjust parameters such as learning rate, batch size, and epochs to iteratively fine-tune and identify the best-performing model.

5. Evaluation

We experimented with four learning rates (5×10^{-6}, 1×10^{-5}, 2×10^{-5}, 3×10^{-5}) and four batch sizes (1, 2, 4, 8), running each combination for 10 epochs. Model performance is assessed using accuracy, calculated as follows:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

6. Result

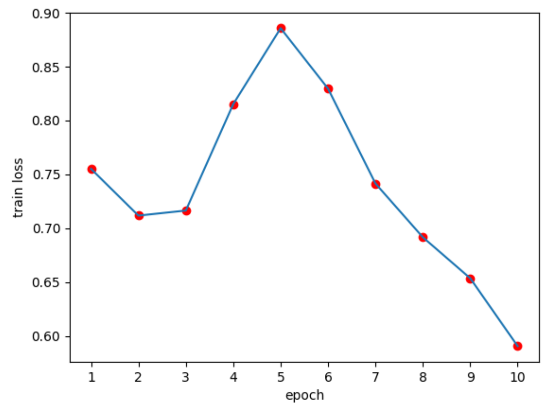
6.1. bert-base-chinese

In utilizing this model as a pre-trained model, its prediction results mostly consist of all 0s or all 1s, with an accuracy of 0.459 for predicting all 0s and an accuracy of 0.550 for predicting all 1s. Ultimately, the model performs best when the learning rate is set to 1e-05, batch size is 4, and epoch is 2, achieving an accuracy of 0.551. Refer to A.1 for detailed test results.

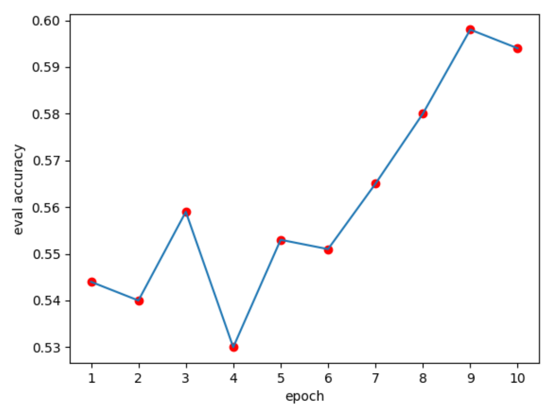
6.2. hw2942/bert-base-chinese-finetuning-financial-news-sentiment-test

Using this model as a pre-trained model, the accuracy without fine-tuning on the validation dataset is $275/520 \approx 0.528$. With fine-tuning, the model performs best when the learning rate is set to $2e-05$, batch size is 2, and epoch is 9, achieving an accuracy of 0.598. Refer to A.2 for detailed test results.

The learning curve for the configuration with a learning rate of $2e-05$ and batch size of 2 is illustrated in the following figure:



Picture 2. Train Loss v.s. epoch



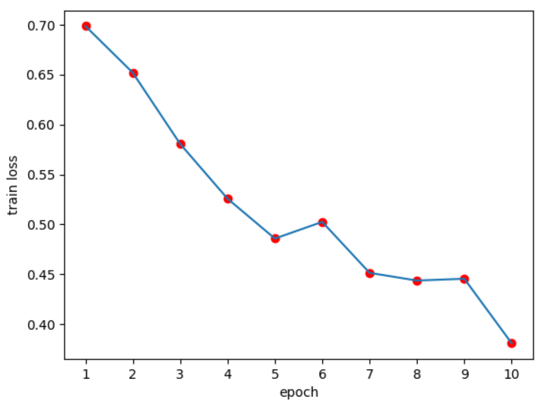
Picture 3. Evaluate Accuracy v.s. epoch

It is evident that after fine-tuning, our model exhibits a significant improvement, progressing from the initial accuracy of 0.528 to 0.598.

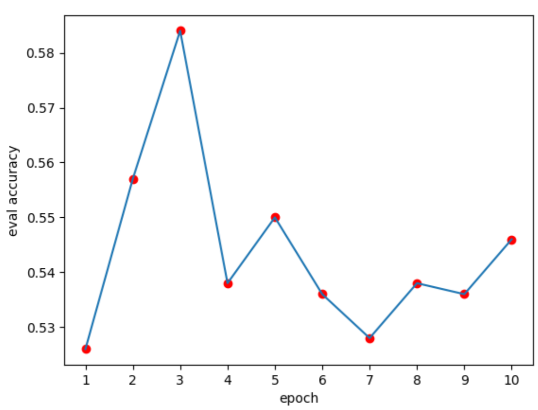
6.3. hw2942/chinese-bigbird-wwm-base-4096-wallstreetcn-morning-news-market-overview-open-000001SH-v1

Using this model as a pre-trained model, the accuracy without fine-tuning on the validation dataset is $295/520 \approx 0.567$. With fine-tuning, the model performs best when the learning rate is set to $1e-05$, batch size is 4, and epoch is 3, achieving an accuracy of 0.584. Refer to A.3 for detailed test results.

The learning curve for the configuration with a learning rate of 1e-05 and batch size of 4 is illustrated in the following figure:



Picture 4. Train Loss v.s. epoch



Picture 5. Evaluate Accuracy v.s. epoch

It is evident that after fine-tuning, our model did not show a significant improvement; in fact, in most cases, it performed worse. We attribute this to the fact that the training data for the pre-trained model is translated from the Wallstreetcn Morning News Market Overview. Apart from potential discrepancies introduced in the translation process, there are inherent differences in the writing styles of American and Taiwanese journalists. Consequently, fine-tuning did not lead to enhanced performance.

7. Conclusion

In conclusion, this report focused solely on predicting stock prices based on publicly available information in the news domain. Due to limitations in training data and the inability to access many non-public news sources that could impact stock prices, coupled with the omission of fundamental, technical, and chip analysis, the results are less than optimal. For more accurate stock price predictions, a comprehensive analysis incorporating fundamental, technical, chip, and news factors is essential.

This report serves as a starting point for future endeavors in stock price prediction, providing a convenient API for usage. Moving forward, we plan to continually gather new financial news and stock prices, integrating information from various aspects to develop a more comprehensive and accurate stock price prediction model.

References

[1] https://huggingface.co/bert-base-chinese

[2] https://huggingface.co/hw2942/bert-base-chinese-finetuning-financial-news-sentiment-test

[3] https://huggingface.co/hw2942/chinese-bigbird-wwm-base-4096-wallstreetcn-morning-news-market-overview-open-000001SH-v1

[4] https://tw.stock.yahoo.com/class/

[5] https://www.ltn.com.tw/

[6] https://finance.yahoo.com

[7] https://github.com/huggingface/transformers/blob/main/examples/pytorch/text-classification/run_classification.py

[8] https://huggingface.co/docs/transformers/main_classes/trainer

[9] https://huggingface.co/docs/transformers/tasks/sequence_classification

[10] https://huggingface.co/docs/datasets/v1.13.0/use_dataset.html

Appendix A. Evaluate Results

A.1. bert-base-chinese

When using bert-base-chinese as the base model, we did not test the scenario with a batch size of 8 due to memory limit exceeded.

learning rate $= 5 \times 10^{-6}$

	batch size			
epoch	1	2	4	8
1	0.459	0.509	0.511	–
2	0.459	0.548	0.494	–
3	0.459	0.488	0.488	–
4	0.459	0.507	0.511	–
5	0.459	0.484	0.488	–
6	0.459	0.486	0.486	–
7	0.459	0.465	0.515	–
8	0.459	0.492	0.478	–
9	0.465	0.490	0.494	–
10	0.531	0.475	0.488	–

learning rate $= 1 \times 10^{-5}$

	batch size			
epoch	1	2	4	8
1	0.459	0.540	0.540	–
2	0.459	0.540	**0.551**	–
3	0.459	0.459	0.501	–
4	0.459	0.540	0.492	–
5	0.459	0.540	0.488	–
6	0.459	0.542	0.486	–
7	0.459	0.517	0.478	–
8	0.459	0.5	0.467	–
9	0.459	0.528	0.469	–
10	0.459	0.528	0.478	–

learning rate $= 2 \times 10^{-5}$

	batch size			
epoch	1	2	4	8
1	0.459	0.459	0.540	–
2	0.459	0.540	0.540	–
3	0.459	0.459	0.459	–
4	0.459	0.540	0.540	–
5	0.459	0.540	0.540	–
6	0.459	0.540	0.459	–
7	0.459	0.540	0.540	–
8	0.459	0.459	0.459	–
9	0.459	0.459	0.459	–
10	0.459	0.459	0.540	–

learning rate $= 3 \times 10^{-5}$

	batch size			
epoch	1	2	4	8
1	0.459	0.540	0.459	–
2	0.459	0.540	0.540	–
3	0.459	0.459	0.459	–
4	0.459	0.540	0.540	–
5	0.459	0.540	0.540	–
6	0.459	0.540	0.459	–
7	0.459	0.540	0.540	–
8	0.459	0.459	0.459	–
9	0.459	0.459	0.478	–
10	0.459	0.459	0.463	–

A.2. hw2942/bert-base-chinese-finetuning-financial-news-sentiment-test

learning rate = 5×10^{-6}

	batch size			
epoch	1	2	4	8
1	0.555	0.55	0.532	0.557
2	0.536	0.513	0.548	0.55
3	0.501	0.475	0.475	0.501
4	0.513	0.503	0.507	0.484
5	0.534	0.471	0.498	0.480
6	0.544	0.463	0.521	0.494
7	0.525	0.517	0.494	0.492
8	0.534	0.480	0.519	0.476
9	0.526	0.488	0.503	0.484
10	0.532	0.478	0.511	0.496

learning rate = 1×10^{-5}

	batch size			
epoch	1	2	4	8
1	0.540	0.513	0.526	0.567
2	0.459	0.511	0.55	0.523
3	0.459	0.507	0.531	0.509
4	0.496	0.503	0.523	0.517
5	0.532	0.476	0.5	0.521
6	0.525	0.498	0.503	0.523
7	0.532	0.492	0.496	0.519
8	0.525	0.490	0.515	0.523
9	0.513	0.5	0.503	0.513
10	0.532	0.492	0.501	0.523

learning rate = 2×10^{-5}

	batch size			
epoch	1	2	4	8
1	0.459	0.544	0.578	0.559
2	0.459	0.540	0.578	0.536
3	0.459	0.559	0.507	0.509
4	0.459	0.530	0.540	0.517
5	0.459	0.553	0.530	0.507
6	0.459	0.551	0.553	0.528
7	0.459	0.565	0.553	0.517
8	0.459	0.580	0.526	0.528
9	0.459	**0.598**	0.526	0.521
10	0.459	0.594	0.525	0.528

learning rate = 3×10^{-5}

	batch size			
epoch	1	2	4	8
1	0.459	0.540	0.459	0.540
2	0.480	0.540	0.540	0.540
3	0.459	0.459	0.490	0.517
4	0.459	0.459	0.536	0.532
5	0.459	0.540	0.525	0.519
6	0.459	0.540	0.567	0.532
7	0.459	0.540	0.526	0.526
8	0.459	0.459	0.555	0.519
9	0.459	0.459	0.546	0.526
10	0.459	0.494	0.544	0.540

A.3. hw2942/chinese-bigbird-wwm-base-4096-wallstreetcn-morning-news-market-overview-open-000001SH-v1

learning rate = 5×10^{-6}

	batch size			
epoch	1	2	4	8
1	0.542	0.505	0.503	0.544
2	0.550	0.525	0.551	0.578
3	0.546	0.503	0.494	0.482
4	0.544	0.521	0.526	0.496
5	0.525	0.511	0.530	0.457
6	0.528	0.521	0.494	0.480
7	0.528	0.536	0.492	0.471
8	0.523	0.525	0.513	0.476
9	0.503	0.546	0.511	0.482
10	0.509	0.540	0.511	0.482

learning rate = 1×10^{-5}

	batch size			
epoch	1	2	4	8
1	0.459	0.507	0.526	0.538
2	0.459	0.532	0.557	0.561
3	0.569	0.530	**0.584**	0.467
4	0.555	0.551	0.538	0.463
5	0.55	0.553	0.55	0.476
6	0.571	0.526	0.536	0.469
7	0.538	0.540	0.528	0.473
8	0.569	0.540	0.538	0.467
9	0.559	0.513	0.536	0.465
10	0.551	0.515	0.546	0.461

learning rate = 2×10^{-5}

	batch size			
epoch	1	2	4	8
1	0.459	0.54	0.503	0.538
2	0.459	0.540	0.532	0.557
3	0.459	0.459	0.526	0.519
4	0.459	0.540	0.521	0.507
5	0.459	0.459	0.488	0.501
6	0.459	0.459	0.521	0.523
7	0.459	0.540	0.515	0.526
8	0.459	0.459	0.509	0.523
9	0.459	0.459	0.525	0.528
10	0.459	0.459	0.525	0.509

learning rate = 3×10^{-5}

	batch size			
epoch	1	2	4	8
1	0.459	0.55	0.459	0.525
2	0.540	0.540	0.540	0.540
3	0.459	0.465	0.573	0.576
4	0.459	0.540	0.548	0.534
5	0.459	0.540	0.521	0.536
6	0.459	0.459	0.515	0.542
7	0.459	0.540	0.536	0.534
8	0.459	0.459	0.534	0.55
9	0.459	0.459	0.534	0.536
10	0.459	0.459	0.515	0.538