Stock Selection Alpha-Factor for Chinese Stock Market Based on Sentiment Analysis

Yiyang Zhang
zhyiyang@umich.edu

Abstract

In this project, we collected the news for stocks in SSE 50 Index of the Chinese stock market and use the sentiment analysis to do the multi-classification on the related news. The news will be classified to five different labels and each stands for different market sentiment related to the stocks. In this multi-class sentiment analysis task, we used BERT model and LSTM model. Then, we mark their prediction result on the new incoming news with different score based on the label and do the ranking based on the scores. Then, we have got several Rank factors based on this method and test its effectiveness by comparing its performance with the benchmark portfolio based on SSE 50 Index. The conclusion is the BERT model and LSTM model performs similarly on this task with accuracy around 75% and the Rank factors we got are effective Augmentation Index Factors. The IR of this ranking method factor is 0.459 and the IC is about 0.044, which means the factors coming from this ranking method has a relative strong ability to gain excess return.

The collected data, relevant codes and all the relevant results can be found on GITHUB at https://github.com/YiyangZhang0201/EECS595-FINAL

1 Introduction

In this project, we will try to address the application of the NLP method to quantitative finance. As we all know, social media and news information will influence people's decisions on the financial market. So, this kind of information has great value when investors make their trading decisions and build their trading systems. The problem we will address is applying the sentiment analysis method of NLP to the information related to the associated stocks to find the markets' common sentiment on the related stocks. And trying to see if this result from the sentiment analysis can be reshaped to an effective factor in the quantitative trading strategy and gain an excess return.

For many years, stock return forecast has been a very important research topic. Many studies use statistical modeling or machine-learning methods, such as Support Vector Machine (SVM), from historical data and then predict the future changes of the stock. In recent years, because of the innovation of technologies, computing power has been greatly improved. Related neural-network models of deep learning have been accelerated and available for many successful applications in different fields, along with a large amount of data training. This success helps us solve and deal with many forecast applications. As we know, stock data is time-series data. It can be used for deriving patterns and identifying trends using deep learning models. Perhaps these trends are too complicated to be understood by humans or other conventional computer processes. The RNN model and LSTM model are very suitable for this sort of tasks. However, the gap between academic and financial industry is that due to the black-box of the neural network results, in real word trading, machine-learning and deep-learning techniques are seldom used directly in to the price or return prediction task of the stocks, since they are very hard to find financial or economic meanings. The most common way that they are used is to build or find the effective alpha-factors with explicit financial or economic meanings that can be really helpful for the Multi-factor trading strategies. And that is also one of the reason why we chose to build a factor to test its effect on stock market in this project.

And with the latest advances in deep learning for natural language processing. More and more researchers use deep learning techniques to recognize the sentiments in the content of the text or descriptive messages. One can instantly understand the stock investor's views on the stock market by recognizing the content of the forum posts and grasping the trend of stock investment.[1] The performance of the stock market is also affected by news which is a public and global information source for everyone. So, as 80& of the investors in the Chinese stock market are individual investors, by recognizing the sentiments in the news will become important vectors for predicting the stock prices' moving direction. Therefore, this project we will combine stock price history, LSTM neural network model and pre-training the Bidirectional Encoder Representations from Transformers (BERT) model to recognize the sentiment as input vectors from news and forum posts for an individual stocks and reshape them into alpha-factors and test their performance on back-testing.

2 Related Works

This section will describe researches that were related to applying the sentiment analysis method in to the quantitative finance field.

In year 2016, Joshi, K., Bharathi, Rao, J (1) developed a system which collects past tweets, processes them further, and examines the effectiveness of various machine learning techniques such as Naive Bayes Bernoulli classification and Support Vector Ma-chine (SVM), for providing a positive or negative sentiment on the tweet corpus. Subsequently, they employ the same machine learning algorithms to analyze how tweets correlate with stock market price behavior. Finally, they examine our prediction's error by comparing our algorithm's out come with next day's actual close price. Overall, the ultimate goal of this project is to forecast how the market will behave in the future via sentiment analysis on a set of tweets over the past few days, as well as to examine if the theory of contrarian investing is applicable. The final results seem to be promising as they found correlation between sentiment of tweets and stock prices. And this result gives us the fundamental idea and reliance of using sentiment analysis into the stock market.

Nan Jing, Zhao Wu and Hefei Wang (2021) (3) propose a hybrid model that combines a deep learning approach with a sentiment analysis model for stock price prediction. they employ a Convolutional Neural Network model for classifying the investors' hidden sentiments, which are extracted from a major stock forum. they then propose a hybrid research model by applying the Long Short-Term Memory (LSTM) Neural Network approach for analyzing the technical indicators from the stock market and the sentiment analysis results from the first step. And in Ko, C. R., Chang, H. T. (2021) (2) they utilized multiple factors for the stock price forecast. The news articles and PTT forum discussions are taken as the fundamental analysis, and the stock historical transaction information is treated as technical analysis. The state-of-the-art natural language processing tool BERT are used to recognize the sentiments of text, and the long short term memory neural network (LSTM), which is good at analyzing time series data, is applied to forecast the stock price with stock historical transaction information and text sentiments. According to experimental results using our proposed models, the average root mean square error (RMSE) has 12.05 accuracy improvement.

However, all of the previous work did not test the sentiment analysis's effect by using the ranking stock selection method, which is one of the mostly effective and common way used in the real industry. So, in this project, I will use this method in the back-testing.

3 Methodology

3.1 Data Processing

3.1.1 Data Collection

The data collection for this project contains two parts, the news data parts and the stock price parts.

For the news data, since there is no public available dataset, we did the collection by writing a python crawler, which will automatically get the required data from the website JINRONGJIE. This is a website that contains all the news for the stocks for A-share in China. By using the python crawler, we got all the news about the stocks in SSE 50 Index since 2019-01-01 to 2021-12-01. And use the data from 2019-01-01 to 2021-05-31 to train and valid the model, and use the data from 2021-06-01 to 2021-11-30 to simulate the factors' performance under the unknown market situation.

For the data of the stock prices, the data is collected by directly exporting from the Wind Financial Terminal, which is the most authority financial terminal in China. The data contains all the stock price data for stocks in SSE 50 Index from 2019-01-01 to 2021-05-31.

3.1.2 Preprocessing of news data

The preprocess of the data has the following steps:

1 Manually label the data, in this process . I first divided these news into categories such as Financial Reports, Legal Reports, Regular Announcements and so on. And each

type has different strength on the stock price. Then, with in each categories, I will give them pos/neg label before the strength based on my own sentiment.

2 Remove the irrelevant words such as stock names. And then do the word segmentation and remove the stop words that we built for this task.

3 Do the word embedding and turn the labels into one-hot type.

3.2 LSTM Model

The LSTM neural network is one of the derivatives of RNN. It not only improves the lack of long-term memory of RNN, but also prevents the problem of gradient disappearance.The LSTM neural network can dynamically learn and determine whether a certain output should be the next recursive input. Based on this mechanism that can retain important information, it provides a good reference and application when constructing a predictive model for this study.The LSTM neural network has a new structure called memory cell. The memory cell contains four main components: input gate, Forget gate, Output gate and Neurons, through these three gates, decide what information to store and when to allow reading, writing and forgetting. Following figure illustrates how data flows through the storage unit and is controlled by each gate.

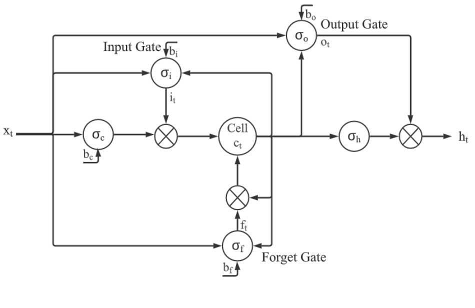

Figure 1: Demonstration of the LSTM model

In the figure, x_t is the input vector in time t, h_t is the output vector, c_t stores the state of the union, i_t is the vector of input gate, f_t is the vector of forgotten gate, o_t is the vector of output gate, and $\sigma_i, \sigma_f, \sigma_o, \sigma_c, \sigma_h$ are activation functions.

3.3 BERT Model

The structure of the BERT model is a multi-layer bidirectional transformer encoder. The transformer was a deep learning model using encoder and decoder for translation task. The BERT model took the advantages of the encoder part in transformer. Figure shown below is the BERT model diagram that takes $E_1 E_2 \ldots, E_n$ as inputs. They could be words or special symbols. After computed through multi-layer bidirectional transformer encoder, $T_1 T_2 \ldots, T_n$ are the output vectors. The BERT model utilized multi-layer bidirectional transformer encoder, they will map a word into different vector according to different context around the input word. In other words, the BERT model is a model that will more precisely map a word to the word vector according the context. Unlike previous language representation models, BERT pre-training a deep bidirectional language representation model based on the upper and lower semantics of all layers.

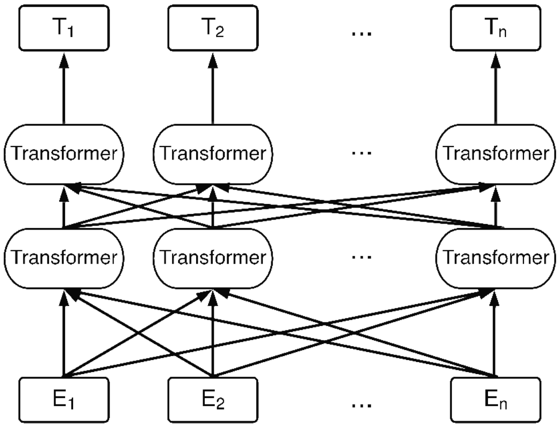

Figure 2: The BERT pre-training model based on bi-direction transformer encoders

And here shows the demonstration of BERT classification task in this project.

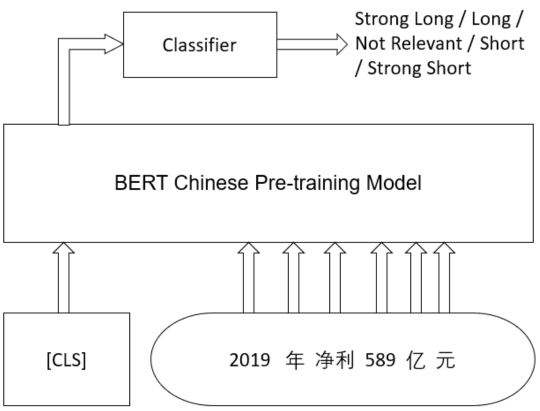
Classifier
Strong Long / Long / Not Relevant / Short / Strong Short
BERT Chinese Pre-training Model
[CLS]
2019 年 净利 589 亿 元

Figure 3: Demonstration of BERT classification task

3.4 Backtrading System

The backtesting is built based on the following basic assumptions.

- The transection cost is 0 in the market.
- The market is fully liquid, which means all the sell orders and buy orders will be fulfilled immediately.
- The news data classification model we choose to use is LSTM model.

With above assumptions, we can start to build the back-testing system. In this back-testing system, we will do the back-testing based on the following steps:

1. Prepare the required news data for each stock, and input the news data into the LSTM model, get the sentiment analysis result. Here we choose the LSTM model because of the computation source limitation.

2. Put the sentiment analysis result of the stock in to the score system and get the score for today. The classification label are scored by following logic:

   [0] Stands for Strong Short label, which means the news on this stock will have a very bad effect on the stock price, will be scored -2

   [1] Stands for Short label, which means the news on this stock has bad effect on the stock price, but the effect is not serious, will be scored -1

   [2] Stands for Not Relevant label, which means the news on this stock is irrelevant to the stock price or there is no news today, will be scored 0

   [3] Stands for Long label, which means the news on this stock is optimistic and will have a positive effect on the stock price, will be scored 1

   [4] Stands for Strong Long label, which means the news on this stock is really optimistic and will have a very significant positive effect on the stock price, will be scored 2 Then, as the news on the stock will have continuous influence on the stock price and this effect will fade as time elapse, we use the following way to include this effect in the scoring system.

   $$S_t = \frac{1}{7} \times \sum_{i=0}^{6} s_{t-i} + s_t$$

   where S_t denotes the final score for stock at time t and s_t denotes the original score that the stock got at time t.

3. Rank the stock based on the final score S_t for each of them at time t. Choose the Top 1 stock, Top 3 stocks and Top 5 stocks as the trading assets for factor Rank1, Rank3 and Rank5 at time t. We will buy the stock at the close time of market based on news at time t and sell all the stocks we bought at time $t-1$.

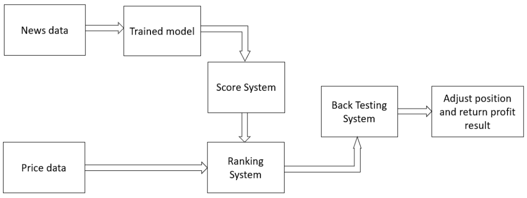

Figure 4: The flow chart of back-testing system

The benchmark portfolio we built is the selected stocks in SSE 50 Index and take equal weight on them. There is no trading strategy on them, we just buy and hold them during the test period.

3.5 Factor Evaluation

Since in the back-testing, we use the score rank to choose certain amount of stocks to do the trading. There is no direct return from the stocks. So, the IC (Information Coefficient) of the factors cannot be computed directly. We will calculate the IC in the following way: Denote S_{it} is the final score time series of the stock i and r_{it} to be the return time series of stock i,

$$IC_i = corr(S_{it}, r_{it})$$

where $corr$ denote Pearson product-moment correlation coefficient.

Then, IR (Information Ratio) is calculated by

$$IR = (\frac{1}{N}\sum_{i=1}^{N} IC_i)/std(\bar{IC})$$

where N denotes the total number of stocks and $\bar{IC}$ denotes the IC samples got from all the stocks.

4 Experiments and Results

4.1 NLP Part

In this part, we will compare the classification part result got by model BERT and LSTM.

4.1.1 BERT Model

In this part, the pre-trained model we chose to use is "bert-base-chinese", and we did the following settings: Batch Size = 32, Number of Labels = 5, Epochs = 3.

And after 3 Epochs, we can see from the output result that the classification accuracy on the test dataset reach 77%.

4.1.2 LSTM Model

In the LSTM Model, we built the model as shown in the following,

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 250, 100)	100000
spatial_dropout1d (SpatialDr	(None, 250, 100)	0
lstm (LSTM)	(None, 100)	80400
dense (Dense)	(None, 5)	505
Total params: 180,905		
Trainable params: 180,905		
Non-trainable params: 0		

And the test result we got can be shown in the following table.

	precision	**recall**	**f1-score**	**support**
0	0.64	0.79	0.71	38
1	0.61	0.65	0.63	55
2	0.74	0.77	0.76	162
3	0.67	0.57	0.62	148
4	0.81	0.81	0.81	97
accuracy			0.73	500
macro avg	0.71	0.74	0.72	500
weighted avg	0.73	0.73	0.73	500

And the train accuracy, test accuracy, train loss and test loss can be seen in the following figures.

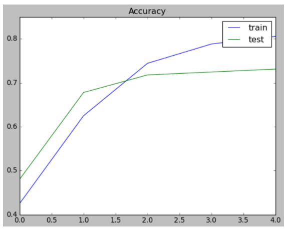

Figure 5: LSTM Accuracy

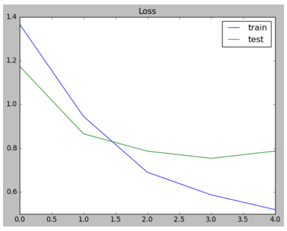

Figure 6: LSTM Loss

From the figures and results we can see that, after 3 Epochs, we can see that although the validation accuracy and train accuracy keeps rising, but the seep has a sharp drop. And for the loss function, after Epoch 4, we can see that the test loss start to rise, which means we should stop to avoid the potential over-fitting.

The final classification accuracy result of LSTM model we built is 73%.

So, the BERT model's sentiment analysis result on this multi-class classification task is slightly better than LSTM.

4.2 Back-testing Part

For the back-testing part, the result of our factors are very promising and the following figure shows the direct back-testing result of factors (Rank1, Rank2 and Rank3) to the benchmark portfolio.

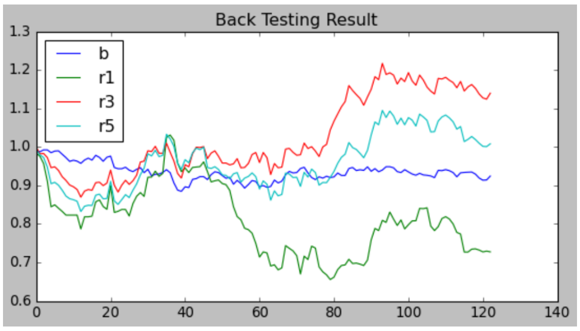

Figure 7: The backtesting result of different factors compared to benchmark

Besides, the IC of the factors is about 0.044 and the IR is about 0.459, since IC above 0.05 and IR above 0.5 means that the factor has the stable strong ability to gain excess return. Out factors' result is very close to this standard, which shows their effectiveness.

And from the figure we can see that the expect the factor Rank1, factor Rank3 and factor Rank5 both gained an obvious excess return, which is 51.6% per year and 20.1% per year respectively. The potential reason that Rank1 factor did not gain an excess return is that the accuracy of the prediction will influence it more than the multi-stocks factor such as Rank3 and Rank5.

5 Conclusion and Future Work

By the above results, we can have that the following conclusion:

- The BERT model and LSTM model performed similarly on the multi-class sentiment analysis model on the data of the financial news. Both of them get an accuracy rate on of 75% the test data.

- The Rank1, Rank3 and Rank5 factors we built based in sentiment analysis method are effective Augmentation Index Factors, which will increase the volatility and the trend of benchmark portfolio. The best Rank3 factor brought us excess return of 51.7% per year.

- Compared to the benchmark portfolio built on SSE 50 Index, we can have that the revenue of Rank1, Rank3, and Rank5 factor experienced an rise and fall, it means there might be an optimal stock selected number for different stock pools.

- The Chinese Stock market has the feature of both Semi-strong efficient market and Strong efficient market. This can help to illustrate the reason of the unexpected revenue drawdown of the factor we built expect the wrong classification.

In the future, the further works can be done based on this are potentially be:

- Use more models and collect more data to do the experiments to see if the accuracy rate can be improved.

- Build Rank10 factor and try some other methods in ranking to see if the factor build with more stocks selected or other ranking methods can improve the performance of the quantitative trading strategy in this way.

- Do some further experiments on the bigger benchmark portfolio. And see if a bigger benchmark portfolio can bring us a better result.

- Try to combine the factor we built with some other factors, to see if it's drawbacks can be offset by them and forms a more powerful stock selection factor.

5.1 References

References

[1] Joshi, K., Bharathi, Rao, J. (2016). STOCK TREND PREDICTION USING NEWS SENTIMENT ANALYSIS. https://github.com/gandalf1819/Stock-Market-Sentiment-Analysis.

[2] Ko, C. R., Chang (2021). LSTM-based sentiment analysis for stock price forecast. PeerJ Computer Science 7 1–23. https://doi.org/10.7717/peerj-cs.408 .

[3] Nan Jing, Zhao Wu, Hefei Wang. (2021). A hybrid model intergrating deep learning with investor sentiment analysis for stock price prediction.