Optimization of Machine Learning Based Stock Prediction

Junjie Guo*

International Education College, Henan University, Kaifeng, China

*Corresponding author: swlee143@gmail.com

Keywords: Stock Prediction, Quantitative Trading, Machine Learning.

Abstract: With the development of artificial intelligence, the core machine learning algorithms in artificial intelligence are becoming more and more mature in the application of finance. However, there are some common problems in most applications, such as over-reliance on native machine learning models, viewing the problem from a machine learning perspective rather than a financial perspective, and lack of optimization for financial applications. These problems inspire us to use and optimize models from a financial perspective. In this paper, we address the current problems in predicting stocks using machine learning for optimization and propose a formula to measure the degree of dispersion of prediction results called Dispersion Degree of the False Sample (DDFS). We give a quantitative and qualitative analysis of the optimization problem. The results indicate that our work can improve the efficiency of the model usage in quantitative trading and deepen the understanding of machine learning in finance.

1. Introduction

The task of stock prediction is highly complex. One of the main reasons is that for different stocks, people cannot figure out exactly the factors that affect stock price movements [1][2]. And in realistic prediction algorithms, researchers are often limited by the structured data of the stocks. These structured data may not work well as factors that affect the stock price movements. There are already studies that are using unstructured data for prediction, which are mainly news. Ideally, for each particular stock people would analyze the important factors that influence its movement and then quantify the data associated with these factors so that they can be used as inputs. Most of the current studies using news as variables and process all the data within a certain time window accordingly and do not carefully screen for positive, bearish data and neutral news [3][4].

Machine learning and deep learning, which are core components of the artificial intelligence field, have had very mature applications in many fields in recent years. But there has been a great controversy for the application of AI in the financial field. First, some achievements are widely recognized, for example, accounting robots can process financial data with great speed and accuracy, and according to related studies, an accounting robot can replace dozens of senior accountants [5]. However, many practitioners are skeptical about the use of machine learning and deep learning algorithms to predict financial market trends and perform quantitative trading. This is because the neural networks in machine learning and deep learning are considered as "black boxes", which means that practitioners do not trust the results given by the "black boxes" [6]. The main reason is that the process of neural networks cannot be well explained by the researchers concerned. As a result, practitioners in the mainstream of quantitative trading rarely use machine learning algorithms. Another concern is that recent studies have shown that neural networks in machine learning and deep learning can be implanted with back-end Trojan horses, leading to complete failure of the algorithms [7][8]. If a program that uses machine learning for quantitative trading is implanted with such a Trojan horse, it will lead to complete failure of the program or even reverse operations, which can cause huge losses to investors and even paralysis of financial markets [9][10].

The study of stock prediction has spawned a very large number of models in recent years. The main algorithms currently used to predict stocks are traditional machine learning algorithms such as Navie Bayes, Random Forests, Binary trees, and Gradient Boosters. Deep learning algorithms mainly

include CNN, LSTM, GRU, and so on. Generally speaking, deep learning models are deeper and have higher model complexity. From a large number of research results, it is known that machine learning algorithms can be maturely applied to prediction, but most of the native models of deep learning are not inherently applicable to prediction, and most deep learning models are applicable to computer vision and natural language processing [11][12]. However, previous studies have shown that variants of deep learning can also be used well for prediction.

In general, the optimization problem in machine learning focuses on the optimization of models, hyperparameters, activation functions, and raw data [13][14]. Since the native model has been shown to work well for forecasting after extensive experiments, changing the structure of the original model in financial forecasting is not very useful and is likely to lead to systematic errors due to incompatibility. Therefore, this paper focuses on the optimization of hyperparameters, the selection of activation functions, the processing of raw data, and the measurement of prediction results.

Confusion Matrix, Precision, Recall, and ROV are common measurements of prediction results in machine learning.[15]. These methods measure the performance of the prediction, precisely the degree of accuracy. These metrics and methods are equally important in predicting financial prices and are important ingredients in measuring the results.

In this paper, we draw a conclusion based on observations of financial markets and financial trading and combined with some practical situations in quantitative trading - the dispersion of results plays a critical role in the practitioner's analysis and has a significant impact on the development of the corresponding quantitative trading strategies. To demonstrate its importance more visually, we give an example in 3.4.

2. Literature Review

Stock prices have long been an important area of research in finance. Stocks have been an area of high interest both in the industry and in academia. Capital asset pricing models, efficient market hypothesis, and factor models in finance have provided strong models and theoretical bases for the study of stock markets [16][17][18]. Models like factor models aim to study those important factors in the market that can affect a certain range of financial markets, and the stock market is the main object of study. Of course, finding strongly correlated factors is very difficult, and different factors tend to show different correlations in financial markets of different countries and regions [19][20].

The prediction of stocks has been studied a lot in the last century. In the early years of research, the main object of forecasting capital asset prices was stocks, because at that time they were the most important investment object in the capital market. At that time, forecasts could only be made on the basis of the mathematical and statistical theories available and well established, such as regression analysis [21]. Of course, simple regression analysis was not able to predict well a highly stochastic subject like stock prices. Therefore, for a long time, people did not make great breakthroughs in capital asset price prediction.

Then, the rise of machine learning brought new algorithms to forecasting. One of the major advantages of machine learning is that it can learn a large amount of data and find certain correlations from the data, so machine learning is very powerful for processing data. Data such as financial time series can also be predicted by machine learning algorithms. Numerous algorithms have achieved excellent results, such as plain Bayes, random forests, gradient boosters, and binomial trees [22] [23] [24].

Deep learning has achieved excellent results in recent years in computer vision, natural language processing. Strictly speaking, deep learning is a branch of machine learning [25]. One of the major features of deep learning is the ability to make models larger and deeper. This also means that deep learning algorithms can handle huge amounts of data and, due to the recent increase in computer computing power, deep learning can perform specified tasks faster. Intuitively deep learning is better than machine learning at predicting capital asset prices. However, one point to emphasize is that the native models of deep learning are basically suitable for the image and natural language processing and do not perform as well as machine learning in some models in terms of processing time series.

With these powerful algorithms, researchers can use existing models to achieve far better results

in prediction than traditional mathematical and statistical methods. But people are not satisfied with the current achievements, so people started to investigate whether we can use some unstructured data, such as news [26][27], instead of using traditional structured data, such as the opening price and closing price of stocks, when predicting capital asset prices. Based on the existing financial theory and the actual situation of the financial market, it is widely believed that news as a kind of data can be used to predict financial time series. This is because both financial theory and people's market intuition generally agree that news has a significant impact on financial markets and that financial market fluctuations may often be influenced by news or even by certain important people [28].

3. Method

3.1 Data Cleaning

Data preprocessing is a key step to improve the efficiency of the model. Data cleaning is a necessary step, and in this paper, we perform the usual data cleaning work - filling nulls and removing outliers. We take the average of the 100 non-null values before and after the null values to fill the null values. Outliers are usually problems that occur during the data acquisition process, so we perform outlier handling by filling the nulls.

One feature of stock data that has been overlooked in many studies is that stock price data often has too large a spread over volume data. This means that the learning process may contain a lot of noise, and the noise is likely to cause overfitting. This means that the signal-to-noise ratio of the stock data is low, which can lead to non-robust results. Normalization means that the raw data are all mapped between 0 and 1 to facilitate data processing. In this paper, the closing prices of stock data are nonlinearly log-normalized. The log normalization formula is shown in (1). where X* denotes the data after normalization, and MAX(X) refers to the sample with the largest value in the data set. Normalization is used very frequently in machine learning and deep learning algorithms and has been shown to be effective in reducing noisy data and reduce overfitting [29][30]. However, normalization has been used less frequently in stock prediction.

$$X* = \frac{\log_{10}(X)}{\log_{10}(\mathrm{MAX}(X))} \tag{1}$$

In this paper, we choose the data of 2 stocks for prediction. They are AAPL and MMM. The data include a total of 3000 trading days and the time span is from 2006-1-3 to 2017-11-30. the overview of these two data sets is shown in Table 1.

Table.1.The overview of two stock data set

Stocks	Average Close Price	Range	Variance	Days of increase	Days of decline
AAPL	63.95	169	1919.48	1580	1420
MMM	111.02	201.31	1886.66	1605	1395

Empirically, 80% and 20% of the data are used as the training and test sets, respectively. That is, 2400 data in the dataset are used as the training set and 600 data are used as the test set.

3.2 Prediction Using Machine Learning

In this paper, we use four algorithms from machine learning (SVM, Naive Bayes) and deep learning (CNN, LSTM) for prediction.

Parsimonious Bayes is a simple but powerful prediction algorithm, and there have been many studies in machine learning that have demonstrated the efficiency of Parsimonious Bayes. Parsimonious Bayes is proposed based on Bayes' theorem, and Parsimonious Bayes assumes that the feature conditions are independent of each other. In the dichotomous classification problem in this paper let be the data set, the set of feature attributes corresponding to the data set is, and the set of categories is. The Bayesian formula shown in (2) yields the probability that a sample data belongs to category Y shown in formula (3).

$$P(Y|X) = \frac{P(Y)P(X|Y)}{P(X)}, \tag{2}$$

$$P(yi|x1, x2, \ldots, xd) = \frac{P(yi)\prod_{j=1}^{d} P(xj|yj)}{\prod_{j=1}^{d} P(xj)}, \tag{3}$$

SVM is a supervised learning algorithm and is a linear classifier that performs binary classification of data. Given the input data and the learning target in the classification problem, the feature space formed by the input data is, and the learning target is a binary variable. In SVM, if the feature space of the input data exists as the decision boundary of the hyperspace so that the learning target is separated according to the positive class from the negative class and the distance from any sample point to the plane is greater than or equal to 1. The decision boundary is shown in formula (4) and the distance from the point to the plane is shown in formula (5). The conditions to be satisfied by the classification are shown in formula (6).

$$\omega^T X + b = 0, \tag{4}$$

$$yi(\omega^T Xi + b) \geq 1, \tag{5}$$

$$\begin{cases} \omega^T Xi + b \geq +1 \Rightarrow yi = +1 \\ \omega^T Xi + b \leq -1 \Rightarrow yi = -1 \end{cases}, \tag{6}$$

CNN is a representative algorithm of deep learning, Alex used CNN in 2012 to achieve the best results in the field of image recognition at that time [31], since then CNN has been widely used as a powerful algorithm in the fields of computer vision, natural language processing, etc. CNN has three layers of neural networks: convolutional layer, pooling layer, and fully connected layer, in addition to input and output layers. LSTM is a temporal recurrent neural network that has shown powerful performance in time series prediction. In this paper, we set the instantaneous deactivation rate to 30% for both CNN and LSTM. In the premise of binary classification in this paper, the binary cross-entropy loss function is used as the loss function in this paper, as shown in formula(7).

$$L = -\sum_{i=1}^{N} y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)}) \tag{7}$$

The activation function is chosen as leaky relu, which has been shown to accelerate the convergence speed well and prevent gradient explosion and overfitting [3][32]. Leaky Relu is shown in formula (8). the structure of CNN and LSTM are shown in Figure 1. and Figure 2. respectively.

$$\text{LeakyRELU} = \begin{cases} x, x > 0 \\ \lambda x, x \leq 0 \end{cases} \tag{8}$$

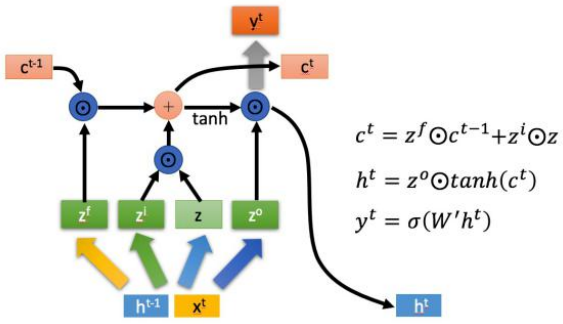

Figure 1. The structure of LSTM

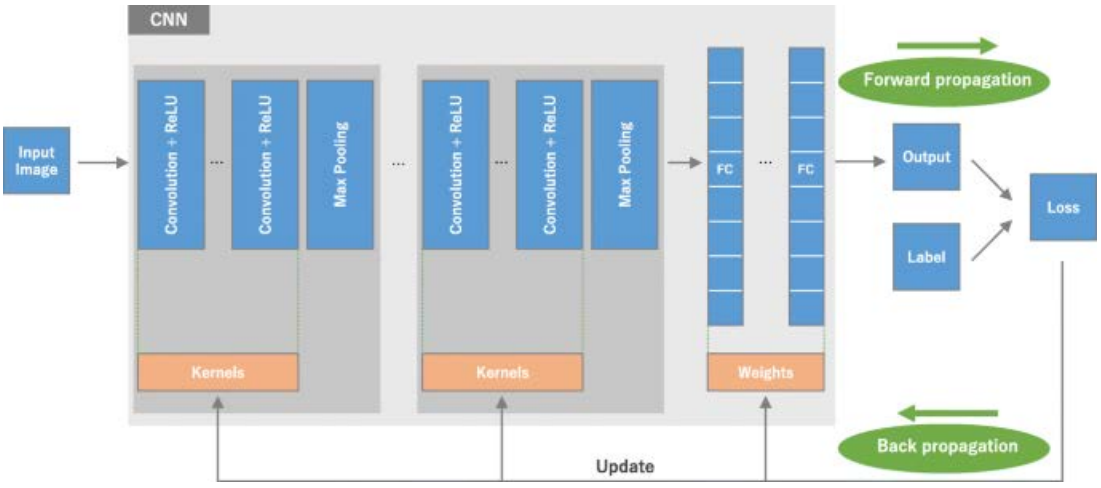

Figure 2. The structure of CNN

3.3 The formula for evaluating prediction result

In this paper, we consider that the distribution of error samples plays an important role in the formulation of trading strategies, so we give a method to measure the dispersion of error samples. This is defined as follows.

In the test set, a represents the accuracy of the prediction and 1-a represents the error of the prediction. Define$(I_1, I_2, I_3, \ldots, I_n)$as the set of error sample points, and $I_n - I_{n-1}$ is the Euclidean distance from the nth error sample point to the n-1th error sample point, where n is an even number and n-1 is an odd number, $B_n$is the first sample point in the test set, and is the last sample point in the test set. Where $B_n - B_{n-1}$ refers to the Euclidean distance from the last sample point to the first sample point. The formula is shown in (9).

$$\alpha = \frac{(B_n - B_1) - \sum_2^n (I_n - I_{n-1})}{a} \left(n \text{ is even}, \text{ n-1 is odd}\right) \quad (9)$$

The a in the formula acts as a scaling factor, 0 < a < 1. That is, when the numerator is the same, the larger the a the smaller the value of the formula, and vice versa, the larger the value of the formula. The denominator part does not affect the relative results of the formula; the denominator part is intended to take the accuracy of the prediction into account.

3.4 A Case related to the formula

In order to get a better sense of the distribution of the different false samples, in this paper we manually assume two extreme cases of the false sample distribution. First, the accuracy of the prediction is assumed to be 80%. In the first case, it is assumed that 20% of the error samples are fully aggregated, and in the second case, it is assumed that 20% of the error samples are uniformly distributed. The number of sample points in both cases is assumed to be 1000. In 4.2, we give a comparison with the normal prediction situation.

3.5 Precision and Recall rate

Precision and Recall rate are two measures of the outcome of the classification problem. The recall rate is calculated based on the confusion matrix. The confusion matrix is a situation analysis table in machine learning that summarizes the prediction results of a classification model in the form of a matrix that summarizes the records in the dataset according to two criteria: the true category and the category judgment predicted by the classification model. The confusion matrix is shown in Figure 3.

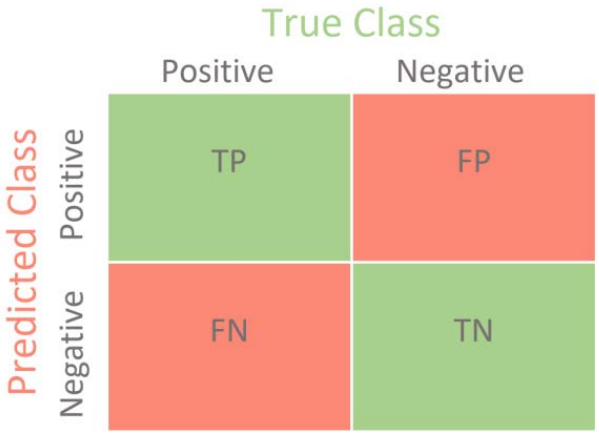

Figure 3. Confusion Matrix

The difference between FP and FN is that FP refers to the number of samples with positive predictions and positive true values, and FN refers to the number of samples with negative predictions but positive true values. The formulas for precision and recall are shown in formula (10) (11) respectively.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{10}$$

$$\text{RECALL} = \frac{\text{TP}}{\text{TP+FN}} \tag{11}$$

4. Result

4.1 The prediction result

Table.2 gives the prediction results, which include Precision and Recall. first of all, it can be seen from the table that the difference between precision and recall is not very big, and this paper mainly uses both metrics in order to measure the results more reasonably and objectively. One possible reason is that SVM is a more complex algorithm for classification problems and can better handle classification problems. LSTM has been shown to be an extremely effective algorithm for processing time series problems, while CNN excels in computer vision and natural language processing tasks.

Table 2 Precision and Recall

Methods	Precision=TP/TP+FP	Recall=TP/TP+FN	Precision-Recall
NB	56.03%	57%	-0.97%
SVM	58.66%	61.50%	-1.84%
CNN	58.52%	57.40%	1.12%
LSTM	60.13%	61.59%	-1.46%
Overall Result	Best in Precision	Best in Recall	Best Overall
	LSTM	LSTM	LSTM

4.2 The result of the case

Table.3 gives the results of the example mentioned in 4.3. Since we assume that there are 1000 sample points, $B_n - B_1$ is 1000, and $\sum_2^n (I_n - I_{n-1})$ in the formula is the part that needs to be calculated, and it can be seen from the table that there is a big difference between the values of this part in the two examples. From the final results, the gap between these two examples is further increased, and the difference between them is mainly the difference of $\sum_2^n (I_n - I_{n-1})$. Overall, it can be learned that the more aggregated the error sample is, the larger the result of this formula is, and vice versa.

Table.3. The result of the cases

Formulas	Case One	Case Two
$B_n - B_1$	1000	1000
$\sum_{2}^{n}(I_n - I_{n-1})$	100	500
$\frac{(B_n - B_1) - \sum_2^n (I_n - I_{n-1})}{a}$	1125	625

4.3 The evaluation based on our formula

Table.4 gives the prediction results of the four algorithms we chose based on the measurement of our proposed formula. Where we use Recall to calculate the result of the formula where a=Recall. It can be seen from the table that the results of the four algorithms do not differ much, which indicates that the algorithms do not have bugs or other errors during the operation. Based on the conclusions in 5.2, we can conclude that NB and CNN predict a higher aggregation of erroneous samples than SVM and LSTM. overall, our whole training process and results do not show any problems.

Next, we will analyze the significance of the results obtained based on this formula and the considerations in the use of the formula. First, if there is a large difference between the prediction results of different algorithms, it may mean that some algorithms are overfitted during the training process or that the prediction results are less feasible due to the excessive noise in the original data. Second, the results based on our formula are not an absolute measure of prediction performance but are intended to provide a measure of the aggregation of error samples. Third, this formula can quickly help one to determine how well the model is trained and run and is very efficient to use.

Table.4. Evaluation of prediction result based on our formula

Formulas	NB	SVM	CNN	LSTM
$B_n - B_1$	3000	3000	3000	3000
$\sum_{2}^{n}(I_n - I_{n-1})$	580	611	550	552
$\frac{(B_n - B_1) - \sum(I_n - I_{n-1})}{a}$	4245.60	3884.55	4268.29	3974.67

5. Conclusion

In this paper, we address the problems in machine learning based algorithms for stock prediction and propose to view and use machine learning models from a financial perspective, rather than copying methods from other fields. Specifically, we propose some optimizations in terms of the financial perspective. In the data cleaning phase, we advocate using log normalization to address the problem of excessive noise and variance in financial structured data. In terms of measuring the results, we propose the formula of the Dispersion degree of the false sample. Through the results, we find that this formula can be very helpful to find some problems in the training process and help people to measure the prediction results more reasonably and objectively.

References

[1] Engle R F, Ghysels E, Sohn B. Stock market volatility and macroeconomic fundamentals[J]. Review of Economics and Statistics, 2013, 95(3): 776-797.

[2] Giglio S, Maggiori M, Stroebel J, et al. Inside the mind of a stock market crash[R]. National Bureau of Economic Research, 2020.

[3] Vui C S, Soon G K, On C K, et al. A review of stock market prediction with Artificial neural

network (ANN)[C]//2013 IEEE international conference on control system, computing and engineering. IEEE, 2013: 477-482.

[4] Pahwa N, Khalfay N, Soni V, et al. Stock prediction using machine learning a review paper[J]. International Journal of Computer Applications, 2017, 163(5): 36-43.

[5] Ng C, Alarcon J. Artificial Intelligence in Accounting: Practical Applications[M]. Routledge, 2020.

[6] Rai A. Explainable AI: From black box to glass box[J]. Journal of the Academy of Marketing Science, 2020, 48(1): 137-141.

[7] Tang R, Du M, Liu N, et al. An embarrassingly simple approach for trojan attack in deep neural networks[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020: 218-228.

[8] Hutson M. Hackers easily fool artificial intelligences [J]. 2018.

[9] Chatzis S P, Siakoulis V, Petropoulos A, et al. Forecasting stock market crisis events using deep and statistical machine learning techniques[J]. Expert systems with applications, 2018, 112: 353-371.

[10] Tabar S, Sharma S, Volkman D. A new method for predicting stock market crashes using classification and artificial neural networks[J]. International Journal of Business and Data Analytics, 2020, 1(3): 203-217.

[11] Islam S M S, Rahman S, Rahman M M, et al. Application of deep learning to computer vision: A comprehensive study[C]//2016 5th International Conference on Informatics, Electronics and Vision (ICIEV). IEEE, 2016: 592-597.

[12] Deep learning in natural language processing[M]. Springer, 2018.

[13] Bengio Y, Lodi A, Prouvost A. Machine learning for combinatorial optimization: a methodological tour d'horizon[J]. European Journal of Operational Research, 2020.

[14] Yang L, Shami A. On hyperparameter optimization of machine learning algorithms: Theory and practice [J]. Neurocomputing, 2020, 415: 295-316.

[15] Novaković J D, Veljović A, Ilić S S, et al. Evaluation of classification models in machine learning [J]. Theory and Applications of Mathematics & Computer Science, 2017, 7(1): 39–46-39–46.

[16] Rossi M. The capital asset pricing model: a critical literature review[J]. Global Business and Economics Review, 2016, 18(5): 604-617.

[17] Gabriela ğiĠan A. The efficient market hypothesis: Review of specialized literature and empirical research[J]. Procedia Economics and Finance, 2015, 32: 442-449.

[18] Bundoo S K. An augmented Fama and French three-factor model: new evidence from an emerging stock market[J]. Applied Economics Letters, 2008, 15(15): 1213-1218.

[19] Connor G, Linton O. Semiparametric estimation of a characteristic-based factor model of common stock returns[J]. Journal of Empirical Finance, 2007, 14(5): 694-717.

[20] Taneja Y P. Revisiting fama french three-factor model in indian stock market[J]. Vision, 2010, 14(4): 267-274.

[21] Refenes A N, Zapranis A, Francis G. Stock performance modeling using neural networks: a comparative study with regression models[J]. Neural networks, 1994, 7(2): 375-388.

[22] Alkubaisi G A A J, Kamaruddin S S, Husni H. Stock Market Classification Model Using Sentiment Analysis on Twitter Based on Hybrid Naive Bayes Classifiers[J]. Computer and Information Science, 2018, 11(1): 52-64.

[23] Khaidem L, Saha S, Dey S R. Predicting the direction of stock market prices using random forest [J]. arXiv preprint arXiv:1605.00003, 2016.

[24] Wang Y, Guo Y. Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost[J]. China Communications, 2020, 17(3): 205-221.

[25] Ongsulee P. Artificial intelligence, machine learning and deep learning[C]//2017 15th International Conference on ICT and Knowledge Engineering (ICT&KE). IEEE, 2017: 1-6.

[26] Shah D, Isah H, Zulkernine F. Stock market analysis: A review and taxonomy of prediction techniques [J]. International Journal of Financial Studies, 2019, 7(2): 26.

[27] Mohan S, Mullapudi S, Sammeta S, et al. Stock Price Prediction Using News Sentiment Analysis[C]//2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService). IEEE, 2019: 205-208.

[28] Si J, Mukherjee A, Liu B, et al. Exploiting topic based twitter sentiment for stock prediction[C]//Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2013: 24-29.

[29] Jo J M. Effectiveness of normalization pre-processing of big data to the machine learning performance [J]. The Journal of the Korea institute of electronic communication sciences, 2019, 14(3): 547-552

[30] Singh D, Singh B. Investigating the impact of data normalization on classification performance [J]. Applied Soft Computing, 2020, 97: 105524.

[31] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks [J]. Communications of the ACM, 2017, 60(6): 84-90.

[32] Xu B, Wang N, Chen T, et al. Empirical evaluation of rectified activations in convolutional network[J]. arXiv preprint arXiv:1505.00853, 2015.