Proceedings of Business and Economic Studies, 2021, Volume 4, Issue 3
http://ojs.bbwpublisher.com/index.php/PBES
ISSN Online: 2209-265X
ISSN Print: 2209-2641

Quantitative Stock Selection Model Based on Long-Short Term Memory (LSTM) Neural Network

Xiao Wu*, Yanqiu Tang

School of Mathematics and Statistics, Zhaoqing University, Guangdong Zhaoqing, 526061, China

Corresponding author: Xiao Wu, jxwuxiao@126.com

Abstract: This article attempted to construct a multi-factor quantitative stock selection model, analyze the financial indicators and transaction data of listed companies in detail via the big data statistical test method, and to find out the alpha excess return relative to the market in the case of short stock index futures as a hedge in the Chinese market.

Keywords: Multi-factor; Validity test; Stock selection model; Quantitative strategy

Publication date: June 2021; ***Online publication:*** June 30, 2021

1. Research Background and Significance

With the continuous development of artificial intelligence and big data, various deep learning network models and machine learning algorithms such as decision trees, genetic algorithms, support vector machines, logistic regression, etc. have been applied to stock market forecasting.[1][2] Among them, the convolutional neural network model is better than the traditional machine learning algorithms in stock prediction research.[3]

LSTM's recurrent neural network has the characteristics for processing and predicting important events with long intervals and delays in time series. In recent years, it has proved to be outstanding in many fields. The recurrent neural network is mainly used to describe the relationship between the current and previous input data. Its memory ability allows the retain of information before the input and use that information to influence the values and development trends of the subsequent data. The LSTM model is applied to the prediction of stock volatility. With the increase of historical stock data, the effect from the predictions of the LSTM network model tends to be stable.[4] In addition to that, the LSTM network model is also applied to the prediction of stock returns by Chen, and he compared the impact of different input features on the accuracy of the model's prediction.[5]

This paper applied the LSTM network to stock selection signals, determined the appropriate number of layers of LSTM network and the number of hidden neurons in its feedforward network layer, as well as analyzed effective LSTM neural network quantitative stock selection models. Using Python software, the stock selection signals of existing multi-factor strategies are analyzed and predicted. The real and predicted values are compared, and the influence of different LSTM networks on the stock selection signals is verified.[6]

2. Model Ideas

Firstly, this article referred to the technical and fundamental information in the Shanghai and Shenzhen 300 index stock pool from the year 2017 to 2021 to select effective factors. This is done mainly through two indicators which are the information coefficient (IC) mean and IC_IR (information ratio) value to screen the factors and synthesize them, then, building a multi-factor stock selection model, and finally attaining the result of stock selection signals.

Secondly, the Keras framework is used to establish the LSTM neural network model and the stock selection signals of the technical factors that reflect the rise and fall of the market. The derived technical factors and the multi-factor model are used as the input layer while the stock selection signals predicted by the LSTM neural network model are used as the output layer. The test is then repeated and the model parameters are optimized to establish an appropriate number of hidden layers and neurons in the middle.

Finally, the stock selection signal output is filtered out of the original multi-factor stock selection model and a quantitative stock selection model based on the LSTM neural network is established. The model is then backtested and used in the multi-factor selection. A comparison of the backtesting results of the stock model is done to verify the empirical effects of the quantitative stock selection model based on the LSTM neural network.

3. Multi-factor Stock Selection Model

The process of establishing a multi-factor stock selection model is divided into five steps which is selecting candidate factors, comparing and screening factors, factor preprocessing, multi-factor synthesis, and establishing a multi-factor stock selection model.[2]

3.1. Selection of candidate factors

The candidate factors selected in this article are price-to-book ratio (pb), price-earnings ratio (pe), market-to-sales ratio (ps), market capitalization (float_mv), and momentum.

3.2. Factor comparison and screening

IC mean is the correlation between the predicted value and actual value of the measurement variable. IC_IR is the average IC after variance standardization. Generally, |mean(IC)|>0.02 can be used as a stock selection factor to determine the effectiveness of profit forecasting. If |IC_IR|>0.6, the factor was considered to be relatively stable. After calculating the average IC and IC_IR value of each factor on the 5th, 15th, and 30th day, the above five factors were then compared and filtered.

3.3. Factor preprocessing method

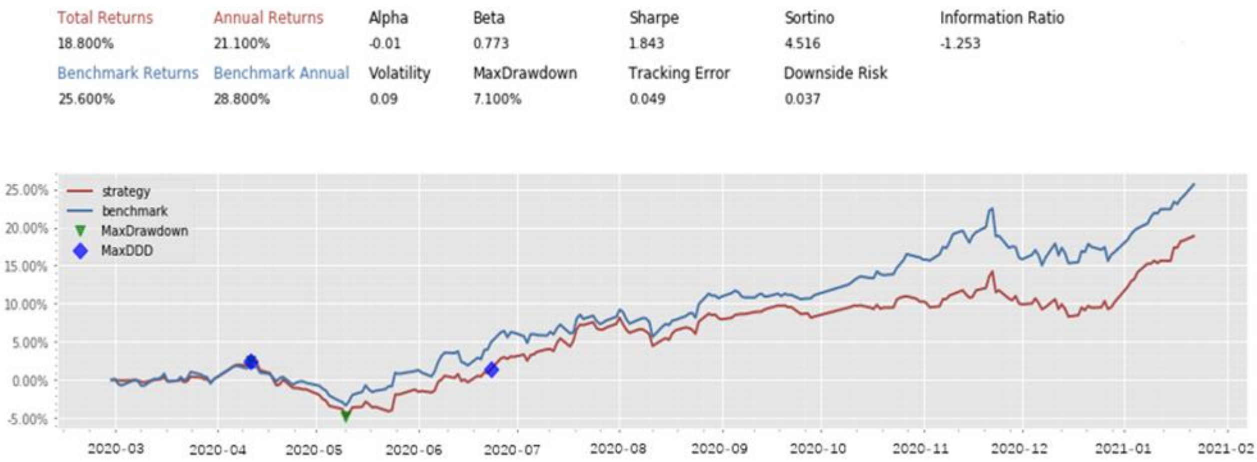
First, the direction of the factors was adjusted. According to previous analysis, the relationship between these factors and stock returns (IC value) in several holding periods were all negative. Hence, adjusting to a positive correlation should be done first, then the factors are extracted, and standardization (z-score) is done.

3.4. Multi-factor combination method

There is a general processing in combining the screened factors. When there is redundancy between the factors, whereby there is relatively strong homogeneity, Schmidt orthogonalization should be first performed on the factors, and then, the orthogonalization residuals are used as factors. This article attempted to assign equal weightage for the factors and to determine the best IC value. Hence, a multi-factor model was constructed with equal weights.

3.5. Building a multi-factor stock selection model

From April 2017 to August 2020, the synthetic factor was tested on a monthly basis in the CSI 300 stock pool. After the test, the quintile returns of the equal-weighted factors and the absolute return results of the strategy were obtained. The following were the parameters: The historical stock positions were empty, the standard fees for buying and selling were 8/10000, the slippage was empty, stocks of suspended listed companies, stocks with price limits, and stocks other than the Shanghai and Shenzhen 300 index components were excluded. The backtesting period was from February 28, 2020 to January 31, 2021 whereby the positions in the CSI 300 stock pool were adjusted monthly and backtested with the engine of Mikai. The results of the strategy backtest are shown in Figure 1.



Total Returns	Annual Returns	Alpha	Beta	Sharpe	Sortino	Information Ratio
18.800%	21.100%	-0.01	0.773	1.843	4.516	-1.253
Benchmark Returns	Benchmark Annual	Volatility	MaxDrawdown	Tracking Error	Downside Risk	
25.600%	28.800%	0.09	7.100%	0.049	0.037	

Figure 1. Multi-factor strategy backtest result graph

4. Quantitative Stock Selection Model based on LSTM Neural Network

4.1. Selection factor

This article selected technical factors that reflected the characteristics of the market and designed new derivative factors. The selected factors are shown in Table 1. The factors that mainly affected the rise and fall of individual stocks (relative and absolute) were considered. Relative ranking changes of factors (changes in cross-sectional rankings), changes in absolute values of factors (changes in time), changes in the spatial ranking of the system and changes in time, etc. were also included.

4.2. Forecast target setting

A signal filter was used to target whether the stock picking signal of a certain strategy is right or wrong. The correct signal operation of a strategy was recorded as “True,” and the wrong operation was recorded as “False”. For example, when stock picking is held in a specific period, the rise in the period is marked as 1, the fall is marked as -1, and the result of the third quarter multi-factor strategy stock selection is the target.

4.3. Dividing the data set

A 120-day data was used to predict the target, the feature data according to the time step were reorganized, and the data set was divided; 0.75 was occupied by the training set while 0.25 was occupied by the test set. The training set label data were sequenced into one-hot encoding (one-hot vector: 1- [0,1],0-[1,0]) and the input layer data were standardized. The z-score method was used to normalize the feature data and predict the conversion and reconstruction of the target. In regard to the training and the test set data, there were

39684 rows in total with 120 as the period and 53 data features as shown below:

Feature data set shape (dataX.shape) = (39684, 120, 53);
Label data set shape (dataY.shape) = (39684, 1);
Training feature data set shape (train_dataX.shape) = (29763, 120, 53);
Training label data set shape (train_dataX.shape) = (29763, 1);
Test feature data set shape (test_dataX.shape) = (9921, 120, 53);
Test label data set shape (test_dataX.shape) = (9921, 1).

4.4. Training process of LSTM model

(1) This paper adopted the categorical cross-entropy function to calculate the model error.
(2) The rejection rate of each layer of network nodes was determined. In order to prevent overfitting, it was set to 0.2 after testing.
(3) The activation function of the LSTM module was determined using Softmax.
(4) The iterative update method of the weight parameters was determined. This article used the Adam algorithm to optimize the LSTM neural network.
(5) The number of epochs of the entire training set sample and the number of samples for each batch of model training (batch size) were determined.

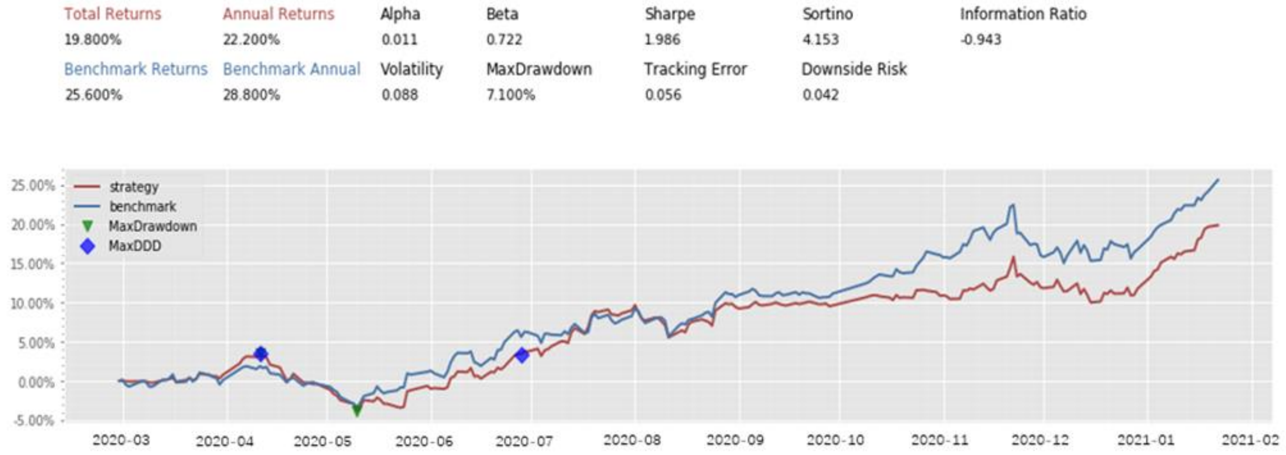
After the grid search optimization test, the parameter setting table was obtained as shown in Table 1:

Table 1. Parameter setting table

Setting parameters	Description	Settings
xrange	Settings	120
n_hidden	Number of neural network nodes in each layer	150
drop_prob	Dropout regularization ratio	0.2
num_layers	Number of middle layers of the model	3
batch_size	Number of samples trained for the model in each batch	20
epochs	Training time of all training set samples	20
class_weight	Weight of each category	Equal rights
activation function	Activation function	Softmax

4.5. Constructing LSTM neural network quantitative stock selection model

The optimal factor parameters trained by the LSTM neural network from 2017 to 2020 were used to predict the stock selection signals from 2020 to 2021 whereby the stock selection signals of the original multi-factor strategy were filtered, and the LSTM neural network quantitative stock selection model was then established. The following were the strategy parameters: The historical positions of stocks were all empty, the standard fees for buying and selling were 8/10,000, the slippage was empty, stocks of suspended listed companies, stocks with price limits, and stocks other than the Shanghai and Shenzhen 300 index components were excluded. The backtest period was from February 28, 2020 to January 31, 2021 whereby the positions in the CSI 300 stock pool were adjusted monthly and backtested with the engine of Mikai. The backtest results are shown in Figure 2.



Total Returns	Annual Returns	Alpha	Beta	Sharpe	Sortino	Information Ratio
19.800%	22.200%	0.011	0.722	1.986	4.153	-0.943
Benchmark Returns	Benchmark Annual	Volatility	MaxDrawdown	Tracking Error	Downside Risk	
25.600%	28.800%	0.088	7.100%	0.056	0.042	

Figure 2. LSTM neural network quantitative stock selection model backtest

5. Model Evaluation

The comparison of the results of the models are shown in Table 2.

Table 2. Comparison of the strategy results of the two models

Stock Selection Model	Annualized rate of return	Alpha	Sharpe ratio	Volatility	Maximum drawdown
Multi-factor model	21.1%	-0.01	1.843	0.09	7.1%
LSTM neural network	22.2%	0.011	1.986	0.088	7.1%

The annualized rate of return and Sharpe ratio of the original multi-factor model were 21.1% and 1.843 respectively. The annualized rate of return and Sharpe ratio of the improved strategy by filtering using the LSTM neural network model increased to 22.2% and 1.986 respectively. In addition, the alpha value of the strategy had also increased from -0.01 to 0.011. Therefore, the training and learning of the original stock selection signals through the LSTM neural network should be encouraged to achieve a more optimized effect. This model uses the input technology factors and stock selection signals of the original multi-factor stock selection model, introduces the dropout mechanism in the LSTM neural network, test the appropriate number of LSTM neural network layers and the number of hidden neurons in the feedforward network layer, as well as scientifically evaluate future stock selection signals. In regard to analyzing and predicting, the stock selection signals obtained by the final LSTM neural network model training can improve the performance of the multi-factor stock selection model.

Funding

Supported by National Natural Science Foundation of China (11961005); Guangdong Province General University Characteristic Innovation Project (2018KTSCX253).

Disclosure statement

The author declares no conflict of interest.

References

[1] Chen G, 2015, Quantitative investment analysis. Economic Management Press.

[2] Ding P, 2016, Quantitative investment: strategy and technology. Publishing House of Electronics Industry**.**

[3] Ouyang J, Lu L, 2011, Application of comprehensively improved BP neural network algorithm in stock price prediction. Computer and Digital Engineering, (2): 57-59.

[4] Chen W, 2018, Comparative study of Shanghai stock exchange index volatility prediction effect based on deep learning. Statistics and Information Forum, 33(5): 99-106.

[5] Chen K, Zhou Y, Dai F, 2015, A LSTM-based method for stock returns prediction: a case study of China stock market. IEEE International Conference on Big Data, IEEE Press, : 2823-2824.

[6] Cai L, 2017, Quantitative investment: using Python as a tool. Publishing House of Electronics Industry.