Asset allocation based on LSTM and Black-Litterman model

Haixiang Yao[a, b, c], Xiaoxin Li[a*], Lijun Li[d]

[a] *School of Finance, Guangdong University of Foreign Studies, Higher Education Mega Center, Panyu District, Guangzhou 510006, P.R. China*

[b] *Southern China Institute of Fortune Management Research, Guangdong University of Foreign Studies, Higher Education Mega Center, Panyu District, Guangzhou 510006, P.R. China*

[c] *Institute of Financial Openness and Asset Management, Guangdong University of Foreign Studies, Higher Education Mega Center, Panyu District, Guangzhou 510006, P.R. China*

[d] *Library, Guangdong University of Foreign Studies, Higher Education Mega Center, Panyu District, Guangzhou 510006, P.R. China*

[*] Corresponding authors

Email address: xiaoxinbill2@gmail.com (X. Li)

Asset allocation based on LSTM and Black-Litterman model*

Abstract

We propose a novel approach to generating the expected returns of investor's views. Specifically, we use the LSTM to train a classification model to forecast asset price trends. Next, according to the prediction results, we construct investor's views in the Black-Litterman (BL) model. This approach can efficiently reduce the sensitivity of the BL model to investor's views, making the BL model not concentrate portfolio weights on a few assets. Our findings suggest that based on the investor's views derived by LSTM, the portfolios of the BL model outperform mean-variance, equally weighted, and market portfolios.

Keywords: LSTM, Black-Litterman Model, Portfolio Optimization, Stock Prediction

JEL Classification: G11, G17, C45

* This paper was supported by the National Natural Science Foundation of China (Nos. 71871071, 72071051); the Key Program of the National Social Science Foundation of China (No. 21AZD071).

1. Introduction

The mean-variance (MV) model (Markowitz, 1952) is a seminal work in the portfolio optimization field. However, the MV model has drawbacks in practice: an unintuitive, highly concentrated portfolio, input sensitivity, and estimation error maximization (Idzorek, 2007). In order to overcome these disadvantages, Black and Litterman (1992) propose the BL model to estimate expected returns, combining the implied excess equilibrium returns with the investor's subjective views to form new expected returns (Bessler et al., 2017). The subjective views represent investor's opinions about expected returns of assets. The subjective views generation in the BL model has received substantial academic attention. Many prior studies have used the AR or GARCH model to derive investor's views (Beach and Orlov, 2007; Harris et al., 2017; Fernandes et al., 2018). Recent studies have used machine learning algorithms to predict asset expected returns and produce investor's views (Pyo and Lee, 2018; Kara et al., 2019). However, the returns of financial assets have non-stationary and loud noise characteristics. It is difficult to predict the specific price or return accurately. The advantages of the BL model are difficult to exploit based on imprecise views.

Long Short Term Memory (LSTM) networks[1] are deep recurrent network architectures proposed by Hochreiter and Schmidhuber (1997). LSTM is designed to deal with long-term dependency problems: error signals flowing backward in time tend to blow up or vanish (Yu et al., 2019). As a powerful recurrent network architecture, LSTM is widely applied in the field of text classification, speech recognition, and financial time series forecast. Many studies have used LSTM to predict stock price or trends in recent years (Nelson et al., 2017; Borovkova and

[1] The details of the LSTM algorithm are presented in the Appendix.

Tsiamas, 2019). Compared to predicting a specific price of assets, the characteristics of price trend is easier to capture. Thus, the primary purpose of this paper is to use LSTM networks to predict price trends and quantify investor's views in the BL model.

In this paper, we develop a novel approach to generating investors' s views based on LSTM networks, which is different from Barua and Sharma (2022). First, we use the LSTM networks to predict the China Securities Index (CSI) 300 industry indices price trends. Then we select the indices predicted to rise in the next month, sort them by descending order using the Cross-Entropy Loss[1] and assign rating values. Finally, we form the expected returns of investor's views and construct portfolios of the BL model. Our findings suggest that based on the investor's views derived by LSTM, the portfolio of the BL model outperforms the MV model and other portfolios. In addition, we make a robustness test for the parameters d of rating values and c of expected return in the BL model, finding that our empirical results are robust, and these parameters will significantly impact the confidence level of investor's views in the BL model.

The main contributions of this study are two folds. First, we use LSTM networks to predict the price trends of CSI 300 industry indices and provide an efficient approach to generate the expected return of investor's views in the BL model. This approach can efficiently reduce the sensitivity of the BL model to investor's views, which makes the BL model not concentrate portfolio weights on a few assets. In addition, it can make the BL model portfolio assign more weight to better assets. Second, our approach uses deep learning technology to provide asset management companies with effective asset allocation solutions.

The rest of this paper is organized as follows. In section 2, we introduce the BL model.

[1] The formula for Cross-Entropy Loss presents in the Appendix.

Section 3 discusses research design and data. The empirical results are shown in section 4, and section 5 concludes.

2. The Black-Litterman Model

Black and Litterman (1992) propose a reverse optimization approach to obtain the implied excess equilibrium return Π, regarded as the prior distribution of expected return. Given the market or benchmark weights w^*[1], the implied excess equilibrium return Π can be calculated as follows (Bessler et al., 2017):

$$\Pi = \lambda \Sigma w^*, \tag{1}$$

where λ is the investor's risk-aversion coefficient, and Σ is the covariance matrix of excess returns. In line with He and Litterman (2002), we set λ to be 2.5.

The BL model combines the implied excess equilibrium returns and investor's views to form a new estimate of expected returns, which is written as (Idzorek, 2007; Silva et al., 2017):

$$\hat{\mu}_{\mathrm{BL}} = [(\tau\Sigma)^{-1} + P^T\Omega^{-1}P]^{-1}[(\tau\Sigma)^{-1}\Pi + P^T\Omega^{-1}q], \tag{2}$$

$$\Sigma_{\mathrm{BL}} = \Sigma + [(\tau\Sigma)^{-1} + P^T\Omega^{-1}P]^{-1}, \tag{3}$$

where τ is a scalar that reflects the uncertainty of implied excess equilibrium returns, Ω is a diagonal covariance matrix that measures the uncertainty of an investor's views, P is a matrix that identifies the assets involved in the views , and q is a vector of the expected returns of each view.

According to Idzorek (2007), we set the scalar τ to be 0.025 and use absolute terms to express investor's views, so the P is an identity matrix. Additionally, Ω is a diagonal matrix as below:

$$\Omega = \begin{bmatrix} (p_1\Sigma p_1^T) * \tau & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & (p_k\Sigma p_k^T) * \tau \end{bmatrix}, \tag{4}$$

where p_k is the *k*th row vector of the matrix P.

Given the $\hat{\mu}_{\mathrm{BL}}$ and the Σ_{BL}, we construct the maximization problem:

$$\max_{w} \, [w^T\hat{\mu}_{\mathrm{BL}} - \frac{\lambda}{2}w^T\Sigma_{\mathrm{BL}}w]. \tag{5}$$

[1] Black and Litterman (1992) assume that w^* are the results of MV model optimization.

The optimal portfolio weights w_{BL}^{*} is calculated as follows:

$$w_{\mathrm{BL}}^{*} = (\lambda\, \Sigma_{\mathrm{BL}})^{-1} \hat{\mu}_{\mathrm{BL}}. \tag{6}$$

3. Empirical design

3.1 Data

We collect the monthly closing prices of ten China Securities Index (CSI) 300 industry indices and twelve technical analysis indicators from the Wind Database. The period ranges from January 2005 to March 2022. These technical analysis indicators as commonly used by prior studies (Li et al., 2017; Kara et al., 2011). Table 1 presents details of technical analysis indicators in this paper.

Table 1

The technical analysis indicators.

Serial Number	Name of Indicator
1	BIAS (Bias Ratio)
2	RSI (Relative Strength Index)
3	DPO (Detrended Price Oscillator)
4	MACD (Moving Average Convergence Divergence)
5	ROC (Rate of Change)
6	MA (Moving Average)
7	MTM (Momentum Index)
8	VHF (Vertical Horizontal Filter)
9	SOBV (Energy Tide Index)
10	KDJ_K (Stochastic Indicator)
11	KDJ_D (Stochastic Indicator)
12	KDJ_J (Stochastic Indicator)

Notes: These twelve technical analysis indicators are derived from the Wind database.

Table 2

The descriptive statistics.

Industry index name	Observation	Mean (%)	Std (%)	Min (%)	Max (%)
Energy	207	0.55	9.48	-33.16	27.93
Materials	207	0.87	10.15	-27.65	34.76
Industrials	207	0.73	9.13	-29.51	35.45
Consumer Discretionary	207	1.03	8.70	-27.22	35.37
Consumer Staples	207	1.78	8.57	-24.47	31.94
Health Care	207	1.38	8.69	-24.45	44.62
Financials	207	1.08	9.25	-29.86	41.25
Information Technology	207	0.68	9.75	-25.37	31.79
Telecommunication Services	207	0.61	9.25	-28.43	43.28
Utilities	207	0.50	8.19	-23.84	37.68

Notes: This table reports the descriptive statistics of monthly excess returns of ten CSI 300 industry indices. The sample period ranges from January 2005 to March 2022.

In addition, we use the CSI 300 index return and RMB one-year deposit rate as the proxies of the market portfolio and the risk-free rate. Table 2 reports descriptive statistics for ten CSI 300 industry indices' monthly excess returns.

3.2 Research design and the formation of investor's views

We use LSTM networks to form investor's views. Fig. 1 presents the procedure for generating the expected returns of investor's views. In the first stage, we use the closing prices of ten CSI 300 industry indices to construct labels "1" and "0", where the label "1" indicates the price will rise in the next month, and the label "0" does the opposite.

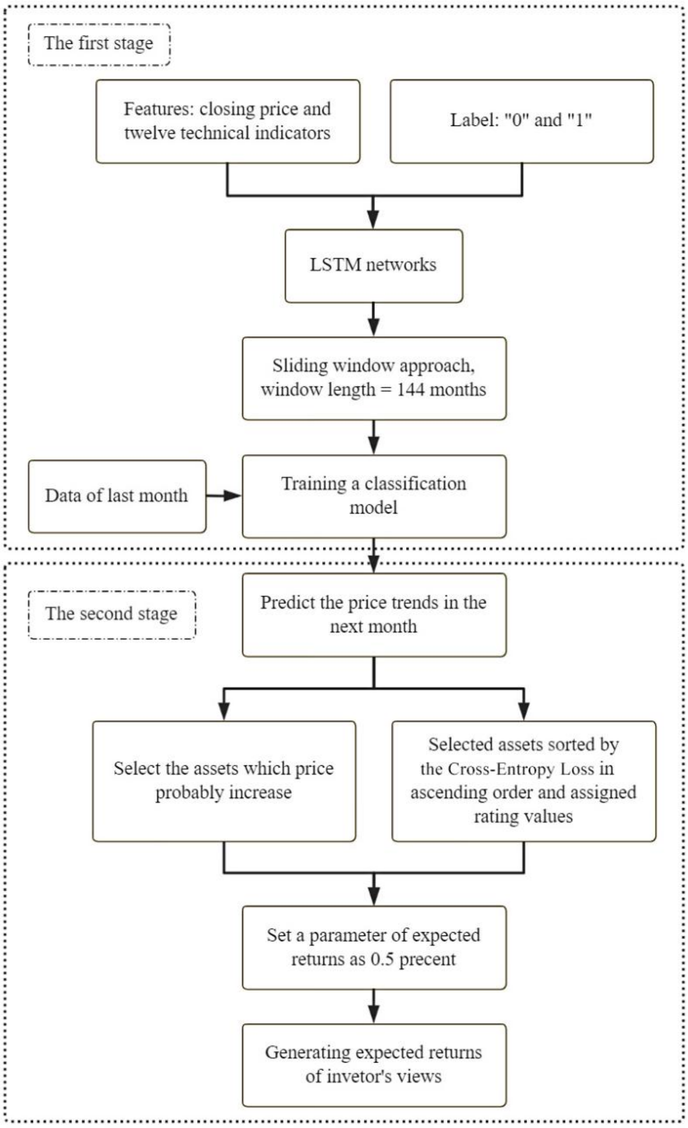

Fig. 1. The approach to generating expected returns of investor's views.

Notes: This figure shows the procedure for generating investor's views. Specifically, this procedure could divide into two-stage: the first stage is training a classification model to predict the CSI 300 industry indices price trends in the next month, and the second stage is generating expected returns of investor's views based on the prediction results.

Next, we use the sliding window approach to split the data set and train a classification model by considering the closing prices and twelve technical indicators. Fig. 2 shows the sliding window approach in detail.

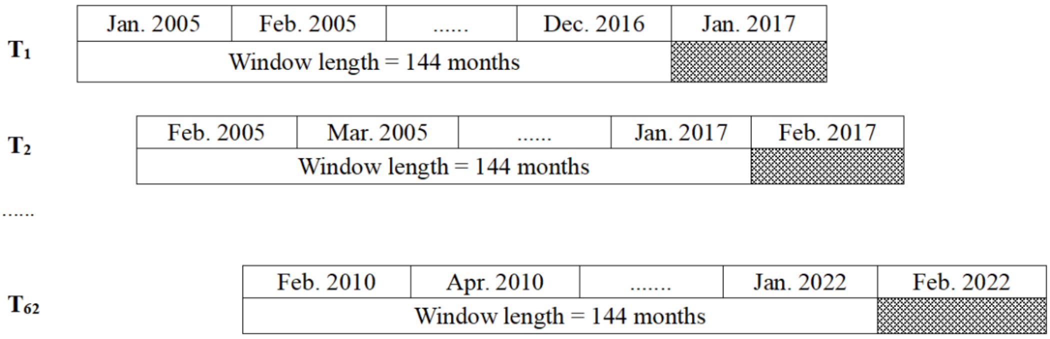

Fig. 2. The sliding window approach.
Notes: The window length is fixed to be 144 months. This figure shows the sliding window approach to splitting data sets and training a classification model. For example, assuming that we were in January 2017, firstly, we used the data set from January 2005 to December 2016 as a training set to train the classification model; secondly, we used the data in January 2017 as testing set to predict the price trends of CSI 300 industry indices in the next month. Then, repeat the process above until February 2022.

In the second stage, the CSI 300 industry indices, predicted to rise in the next month, will be selected as an assets pool, as investors tend to pay more attention to the assets with a greater likelihood of an increase in the next month. We use Cross-Entropy Loss to sort these assets by descending order and assign rating values. The rating values rv are calculated by:

$$rv = 1 + (n - 1) \times d, \tag{7}$$

where n denotes the serial number of selected assets, and d is a scale to control the difference in the rating values.

Another parameter in the BL model is the expected returns of views. In order to exert sufficient impact on the portfolio, according to Silva et al. (2017), we set a parameter c of expected return as 0.5 percent. In the final step, the expected return of the investor's views q is obtained as below:

$$q = rv \times c. \tag{8}$$

In Table 3, we provide a simple example to illustrate how to generate the expected returns of investor's views and the advantages of our approach. Table 3 suggests that rating values can control the difference in views' expected returns, which helps reduce the views' sensitivity and prevent the BL model's portfolio from concentrating on a few assets. In addition, according to the rating values, we know which one of the assets is better than the others and assign higher views' expected returns to better assets, which makes the BL model assign more portfolio weights to better assets.

Table 3

A simple example of generating the expected returns of investor's views.

Industry name	l	rv	q (%)	rv	q (%)
		d = 0.5		d = 1	
Health Care	0.6908	1.00	0.50	1.00	0.50
Information Technology	0.6842	1.50	0.75	2.00	1.00
Utilities	0.6641	2.00	1.00	3.00	1.50
Financials	0.6367	2.50	1.25	4.00	2.00
Consumer Discretionary	0.6161	3.00	1.50	5.00	2.50
Telecommunication Services	0.6071	3.50	1.75	6.00	3.00

Notes: This table reports how to generate the expected returns of investor's views q under different rating values rv. The l denotes Cross-Entropy Loss used to measure prediction errors. The parameter d is used to control the difference in rv. In addition, the parameter c of expected return sets 0.5 percent. Assuming that we were in January 2017, we used the LSTM networks to train a classification model and selected six CSI 300 industry assets as investment targets. Then we sorted these assets in descending order according to Cross-Entropy Loss l and assigned rating values. Finally, we generated the expected returns of investor's views q according to equation (8).

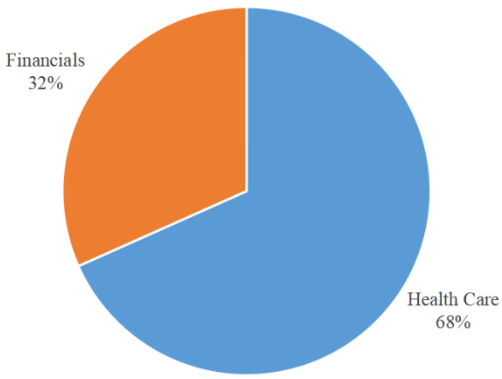

Fig. 3. The benchmark weight w^*.

Notes: This figure shows the benchmark weight w^* in Equation (1).

To analyze how the portfolio changes, we present the benchmark weights w^* in Fig. 3. We also show the weights of the BL model w^*_{BL} under different rating values (d = 0.5, 1) in Fig. 4

and 5, respectively. It suggests that the weights of the BL model adjust more toward investor's views under higher rating values (d = 1). Therefore, we can use rating values to control the strength of investor's views to exert more sufficient impacts on the BL model.

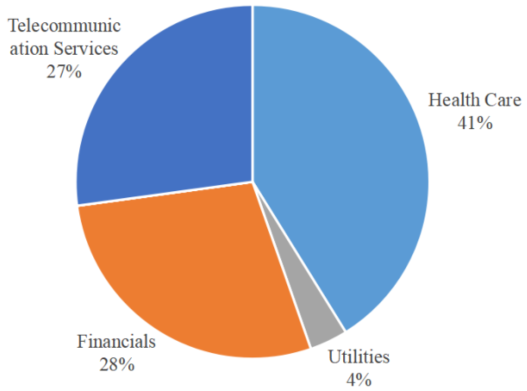

Fig. 4. The weight of the BL model w^*_{BL} (d = 0.5).

Notes: Under parameter d sets to 0.5, this figure shows the weight of the BL model w^*_{BL} in Equation (6).

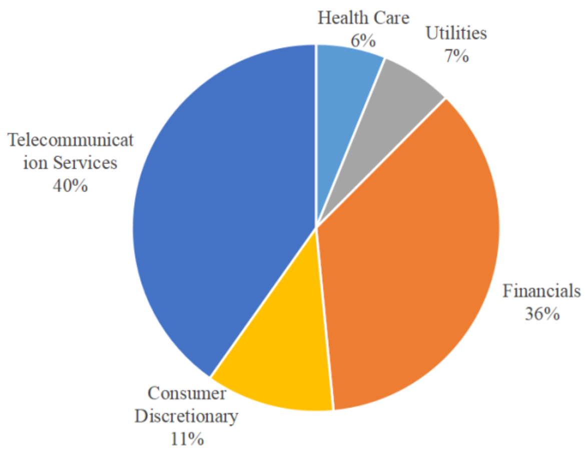

Fig. 5. The weight of the BL model w^*_{BL} (d = 1).

Notes: Under parameter d sets to 1, this figure shows the weight of the BL model w^*_{BL} in Equation (6).

Collectively, our approach to generating the expected returns of investor's views has three advantages. Firstly, it can reduce the BL model's sensitivity to investor's views. Therefore, the BL model's portfolio will not overly concentrate on a few assets. Secondly, it makes the BL model assign more portfolio weights to better-performed assets. Thirdly, we can control investor's views confidence levels according to rating values.

4. Empirical results

4.1 Portfolios performance

We use LSTM networks and a sliding window approach to predict the price trend of ten CSI 300 industry indices over the period from February 2017 to March 2022. Next, we quantify the expected returns of investor's views and construct a portfolio based on the BL model. Table 4 reports the out-of-sample performance of the constructed portfolios. The CSI 300 index represents the market portfolio (MKT). The equally weighted portfolio (EW) is each CSI 300 industry index with an equal weight of 10%. The MV portfolio is constructed by the MV model. In addition, LSTM-EW and LSTM-MV only use assets whose price is predicted to rise in the next month by LSTM and construct a portfolio based on equal weight and the MV model. Finally, LSTM-BL constructs a portfolio by the BL model according to investor's views derived from LSTM.

Table 4 shows that the performance evaluation of LSTM-BL is the best among all portfolios, with the highest annualized average excess return and Sharpe ratio. In addition, LSTM-BL has the lowest maximum drawdown. LSTM-EW and LSTM-MV respectively outperform EW and MV, which suggests the classification model trained by LSTM networks can gain an efficient prediction performance. Fig. 6 presents the cumulative excess returns of portfolios during the period from February 2017 to March 2022. We can conclude consistently with Table 4. In addition, it suggests that LSTM-BL yields higher cumulative excess returns than other portfolios, obtaining about 197.86% cumulative excess returns.

Table 4
The performance evaluation of portfolios.

	MKT	EW	MV	LSTM-EW	LSTM-MV	LSTM-BL
Excess return (%)	4.21	6.06	21.87	22.08	32.36	45.79
Standard deviation (%)	16.52	16.10	24.93	16.64	25.07	20.98
Sharpe ratio	0.25	0.38	0.88	1.33	1.29	2.18
Maximum drawdown (%)	77.22	73.01	25.60	12.68	15.97	6.94

Notes: This table shows different portfolios' performance. Our investment targets are ten CSI 300 industry indices over the period from February 2017 to March 2022. The excess return is the average annualized excess return. Other results are also annualized.

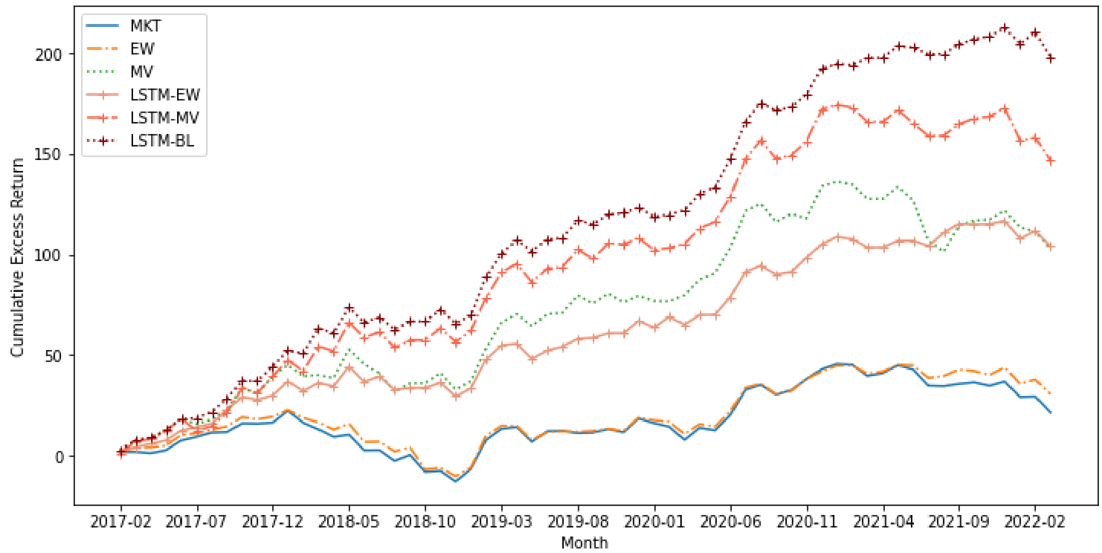

Fig. 6. The cumulative excess returns of portfolios.

Notes: This figure shows the cumulative excess returns of different portfolios from February 2017 to March 2022.

Table 5

The performance evaluation of long-only, short-only, and long-short portfolios.

	LSTM-EW	LSTM-MV	LSTM-BL
Panel A: Long-only Portfolios			
Excess return (%)	22.08	32.36	45.79
Standard deviation (%)	16.64	25.07	20.98
Sharpe ratio	1.33	1.29	2.18
Maximum drawdown (%)	12.68	15.97	6.94
Panel B: Short-only Portfolios			
Excess return (%)	-1.18	8.18	-15.33
Standard deviation (%)	20.32	26.49	22.41
Sharpe ratio	-0.06	0.31	-0.68
Maximum drawdown (%)	194.23	50.18	787.59
Panel C: Long-short Portfolios			
Excess return (%)	23.52	22.51	70.93
Standard deviation (%)	14.52	22.23	19.11
Sharpe ratio	1.62	1.01	3.71
Maximum drawdown (%)	17.68	13.83	3.70

Notes: Long-only portfolios hold only long positions on CSI 300 industry indices whose prices are expected to rise in the next month, while the short-only portfolios do the opposite. Long-short portfolios hold long positions and short positions.

Table 5 reports the out-of-sample performance of long-only, short-only, and long-short portfolios. It shows that the long-only, short-only, and long-short portfolios of LSTM-BL outperform LSTM-EW and LSTM-MV. Moreover, the long-short portfolio of LSTM-BL is better than the long-only portfolio. It further suggests that our approach to generating expected returns of investor's views in the BL model is efficient. Fig. 7 presents the cumulative excess returns of long-short portfolios. It reveals that the long-short portfolios of LSTM-BL can obtain the highest cumulative excess returns.

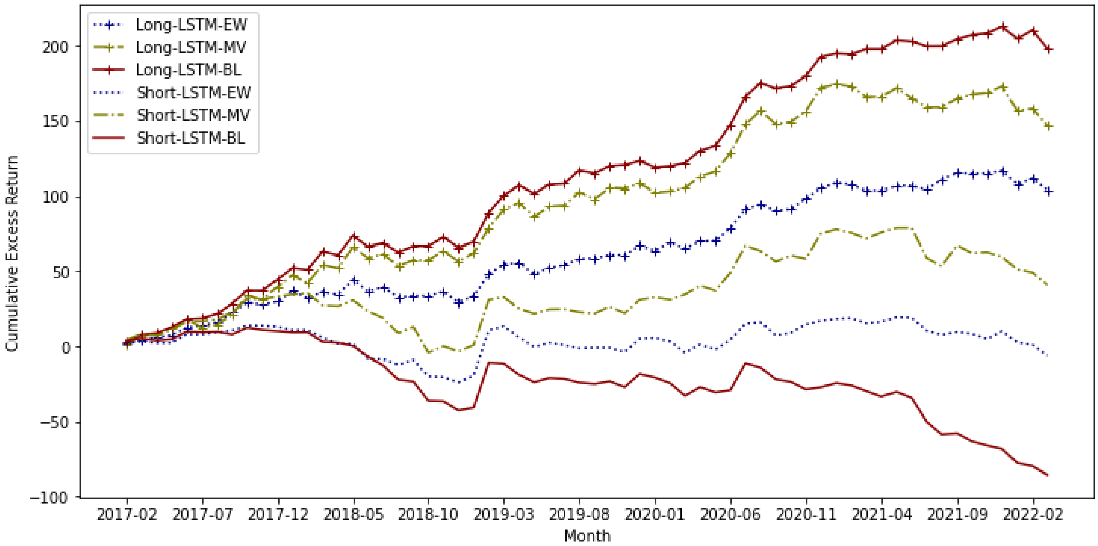

Fig. 7. The cumulative excess returns of long-short portfolios.

Notes: This figure shows the cumulative excess returns of long-short portfolios over the period from February 2017 to March 2022. The excess return of the long-short portfolio is the difference between the long-only and short-only portfolios' excess returns.

4.2 Robustness test

Table 6 reports the robustness test results of parameters d and c. Compared to the results of Panel A and B with those of Panel C, the LSTM-BL outperforms benchmark portfolios under different parameters. It suggests that the portfolio evaluation of the LSTM-BL is robust under different parameters d and c. Panel A shows that parameter d significantly impacts investor views' confidence levels in the BL model. The higher rating values exert more sufficient impacts on the BL model. Panel B shows similar results that the expected return parameter c also impacts investor views' confidence levels. Within an appropriate range, as the expected return parameter increases, the confidence level of investor's views increases, making the BL model pay more attention to investor's views.

Table 6

The robustness test of parameter d and c.

The parameter	Excess return (%)	Standard deviation (%)	Sharpe ratio	Maximum drawdown (%)
Panel A: The portfolio performance of the LSTM-BL under different parameter d.				
0.1	33.30	22.80	1.46	12.62
0.3	37.70	22.72	1.66	10.02
0.5	39.82	22.20	1.79	8.91
0.7	43.10	21.61	1.99	7.98
1	45.79	20.98	2.18	6.94
3	53.68	19.88	2.70	5.61
5	55.98	19.88	2.82	5.43
Panel B: The portfolio performance of the LSTM-BL under different parameter c (%).				
0.1	35.33	23.06	1.53	11.41
0.3	42.83	22.18	1.93	8.50
0.5	45.79	20.98	2.18	6.94
0.7	49.85	19.74	2.52	5.63
1	52.44	19.80	2.65	5.71
3	55.74	21.12	2.64	5.45
5	54.31	21.34	2.55	5.56
Panel C: The performance of benchmark portfolios.				
MKT	4.21	16.52	0.25	77.22
EW	6.06	16.10	0.38	73.01
MV	21.87	24.93	0.88	25.60
LSTM-EW	22.08	16.64	1.33	12.68
LSTM-MV	32.36	25.07	1.29	15.97

Notes: This table shows the robustness test for parameters d and c. The first column is the value range of the parameter. For example, the parameter d ranges from 0.1 to 5 in Panel A; the parameter c (%) ranges from 0.1 to 5 in Panel B. The parameter c of expected returns sets 0.5 percent in Panel A, and the parameter d sets 1 in Panel B. In addition, Panel C is the benchmark portfolios.

5. Conclusion

This paper uses LSTM networks to train a classification model to predict the price trend of CSI 300 industry indices over the period from February 2017 to March 2022. The prediction results are used to generate the expected returns of investor's views and construct portfolios based on the BL model. Our findings suggest that the BL model based on investor's views derived from LSTM outperforms other portfolios. In addition, the prediction performance of the classification model trained by LSTM networks is efficient. Finally, we find that the parameters of rating values and expected return significantly impact the confidence level of investor's views in the BL model.

References

Barua, R., & Sharma, A. K. (2022). Dynamic Black Litterman portfolios with views derived via CNN-BiLSTM predictions. Finance Research Letters, 103111.

Beach, S. L., & Orlov, A. G. (2007). An application of the Black–Litterman model with EGARCH-M-derived views for international portfolio management. Financial Markets and Portfolio Management, 21(2), 147-166.

Bessler, W., Opfer, H., & Wolff, D. (2017). Multi-asset portfolio optimization and out-of-sample performance: an evaluation of Black–Litterman, mean-variance, and naïve diversification approaches. The European Journal of Finance, 23(1), 1-30.

Black, F., & Litterman, R. (1992). Global portfolio optimization. Financial analysts journal, 48(5), 28-43.

Borovkova, S., & Tsiamas, I. (2019). An ensemble of LSTM neural networks for high-frequency stock market classification. Journal of Forecasting, 38(6), 600-619.

Fernandes, B., Street, A., Fernandes, C., & Valladão, D. (2018). On an adaptive Black–Litterman investment strategy using conditional fundamentalist information: A Brazilian case study. Finance Research Letters, 27, 201-207.

Harris, R. D., Stoja, E., & Tan, L. (2017). The dynamic Black–Litterman approach to asset allocation. European Journal of Operational Research, 259(3), 1085-1096.

He, G., & Litterman, R. (2002). The intuition behind Black-Litterman model portfolios. Available at SSRN 334304.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

Idzorek, T. (2007). A step-by-step guide to the Black-Litterman model: Incorporating user-specified confidence levels. In Forecasting expected returns in the financial markets (pp. 17-38). Academic Press.

Kara, M., Ulucan, A., & Atici, K. B. (2019). A hybrid approach for generating investor views in Black–Litterman model. Expert Systems with Applications, 128, 256-270.

Kara, Y., Boyacioglu, M. A., & Baykan, Ö. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. Expert systems with Applications, 38(5), 5311-5319.

Li, B., Lin, Y. & Tan, W. X. (2017). ML-TEA: A set of quantitative investment algorithms based on machine learning and technical analysis. Systems Engineering - Theory & Practice, 37(5), 1089–1100. (in Chinese)

Markowits, H. M. (1952). Portfolio selection. Journal of finance, 7(1), 71-91.

Nelson, D. M., Pereira, A. C., & De Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. In 2017 International joint conference on neural networks (IJCNN) (pp. 1419-1426).

Pyo, S., & Lee, J. (2018). Exploiting the low-risk anomaly using machine learning to enhance the Black–Litterman framework: Evidence from South Korea. Pacific-Basin Finance Journal, 51, 1-12.

Silva, T., Pinheiro, P. R., & Poggi, M. (2017). A more human-like portfolio optimization approach. European Journal of Operational Research, 256(1), 252-260.

Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. Neural computation, 31(7), 1235-1270.

Appendix: LSTM algorithm.

The LSTM algorithm at each time step is specified as follows:

$$f_t = \sigma(W_{fh}h_{t-1} + W_{fx}X_t + b_f),$$

$$i_t = \sigma(W_{ih}h_{t-1} + W_{ix}X_t + b_i),$$

$$o_t = \sigma(W_{oh}h_{t-1} + W_{ox}X_t + b_o),$$

$$\tilde{c_t} = tanh(W_{ch}h_{t-1} + W_{cx}X_t + b_c),$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tilde{c_t},$$

$$h_t = o_t \otimes tanh(c_t), \tag{9}$$

where f_t, i_t, and o_t denote the forget gate, the input gate, and the output gate, respectively. The forget gate decides what information from a past memory cell c_{t-1} should be stored; the input gate decides to store what new information from a candidate memory cell $\tilde{c_t}$; and the output gate decides the information output of memory cell c_t. In addition, h_t denotes the hidden states. W, X_t, and b are weights, inputs, and deviation. The operator $\otimes$ denotes the pointwise multiplication of two vectors. σ and $tanh$ are sigmoid and tanh activation function, respectively;

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \tag{10}$$

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{11}$$

We use Cross-Entropy Loss for a binary classification problem to estimate prediction error. The binary Cross-Entropy Loss function l is given by:

$$l = -\frac{1}{N}\sum_{i=1}^{N}[y_i \times log\hat{y}_i + (1 - y_i) \times log(1 - \hat{y}_i)], \tag{12}$$

where N is the number of training examples, y_i is the label of training example i, and $\hat{y}_i$ is the prediction result of LSTM networks.