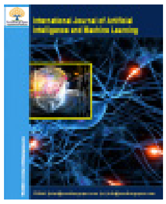
https://doi.org/10.51483/IJAIML.3.2.2023.20-27 ISSN: 2789-2557

SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/



Research Paper Open Access

Predicting Style Factor Returns and Group/Sector Returns Using Long and Short-Term Memory ("LSTM") Deep Learning Neural Networks

Sanele Makamo[1*]

[1]Benguela Global Fund Managers, 3rd Floor, Rivonia Village, Cnr Rivonia Boulevard, Johannesburg, Guateng 2191, South Africa. E-mail: sanelemakamo26@gmail.com

Article Info

Volume 3, Issue 2, July 2023
Received : 15 June 2023
Accepted : 10 July 2023
Published : 25 July 2023
doi: 10.51483/IJAIML.3.2.2023.20-27

Abstract

LSTM is a special type of Recurrent Neural Networks (RNN) with a broad range of applications including time series analysis, document classification, speech, and voice recognition. In this study we employ LSTM for predicting out-of-sample style factor returns and group/sector stock returns derived from the countries, industries, and style explanatory variables of a cross-section factor model. The data considered for the analysis is from September 2013 to June 2023 (approx. 10 years) of 4 style factor returns and 9 stock market groups/sectors for the South African stock market. One of the challenges of using factor models to forecast returns is the assumption that the prior consecutive observations are independent of each other as a result they do not account for the previous observations. Deep learning models like the LSTM are more accurate in predicting these sources of expected returns with their time-series behavior they can accurately predict markets where the effects of multiple market variables have interdependence. The results show that LSTM model is a powerful tool that can be used to predict returns which can help investors and portfolio managers who make investment decisions by grouping stocks into style, countries, or sectors.

Keywords: *Neural Networks, LSTM, RNN, Investment style analysis, Factor returns, Stock returns, Deep learning, Machine learning*

1. Introduction

Factor returns are usually estimated using a cross-sectional regression. In the asset pricing framework, each factor can be used to represent a portfolio for which the return exactly replicates the payoff to the factor. (Manchero, 2010).

In a paper by Menchero (2010) on the characteristic of factor portfolios, global equity factor models normally make use of countries, industries, and styles as explanatory variables in the cross-section regression model. In addition, a market factor is often included to capture the overall effect of the equity market. Country and industry factors are usually treated as indicator variables. That is, the stock is assigned an exposure of 0 or 1,

* *Corresponding author: Sanele Makamo, FBenguela Global Fund Managers, 3rd Floor, Rivonia Village, Cnr Rivonia Boulevard, Johannesburg, Guateng 2191, South Africa. E-mail: sanelemakamo26@gmail.com*

depending on whether it belongs to the country or industry under consideration. Style exposures, by contrast, typically are distributed in a continuous fashion with mean 0 and standard deviation 1.

This paper focuses on the future prediction of the factor returns derived from the countries, industries, and style explanatory variables. Data were collected for the groups from 10 years of historical records. The value predictions are created for 1, 2, 3, 4, 5 and 6 months in advance. Long Short-Term Memory (LSTM) machine learning algorithms was utilized for prediction of future values of the historical 4 style factor returns and 9 stock market groups/sectors for the South African stock market.

Nabipour *et al.* (2020) compared different deep learning models and other machine learning algorithms yields and found that the LSTM as the best structure for predicting stock market returns for multiple forecast windows using different regression loss measures like the MSE.

In their paper, Fischer and Kraus (2017) used LSTM networks for predicting out-of-sample directional movements for the constituent stocks of the S&P 500 from 1992 until 2015. They found that LSTM networks to outperform memory-free classification methods.

In their paper Abe and Nakayama (2018) investigated the performance of a deep learning models to predict one-month-ahead stock returns in a cross-section in the Japanese stock market. They trained the model by using the latest 120 sets of training from the past 10 years. They investigated 8 patterns of deep neural networks (DNN) with 8 layers (DNN8) and with 5 layers (DNN5). Their results indicated that the DNN with greater numbers of layers could increase representational power by repeating nonlinear transformations and improve the prediction accuracy of the cross-sectional stock returns.

Chen and He (2018) in their study on deep learning methods proposed a 6-layer convolution neural network (CNN) to predict stock prices in the Chinese stock market. They used a binary classification predict the movement of stock prices instead of regression i.e., one indicating that the stock price is up and zero indicating the stock price is down. The results showed that the CNN model is robust and can make prediction for a 1D sequential data.

Artificial Neural Networks (ANN) are either single or multi-layer networks that are fully connected. In each layer, each node is connected to every other node in the next layer. If there is an increase in the number of hidden layers, can lead to an even deeper network. Figure 1. Shows a sample of ANN with an input and output layer along with two hidden layers.

Figure 2 below illustrates the relationship of ANN nodes. This can be either of the hidden or output nodes. Since a node takes the weighted sum of the inputs which is added to the bias value and then passes it through an activation function which is usually non-linear. Firstly, the procedure moves from the input to the output, and lastly, the final output is obtained by repeating this process for all nodes. That is, the learning process of weights and biases associated with all nodes for training the neural network.

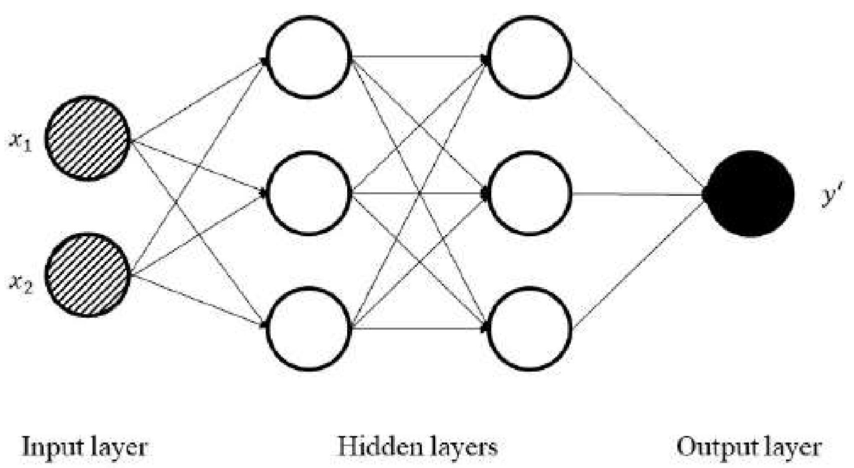


Figure 1: Schematic Illustration of Artificial Neural Networks (ANN)

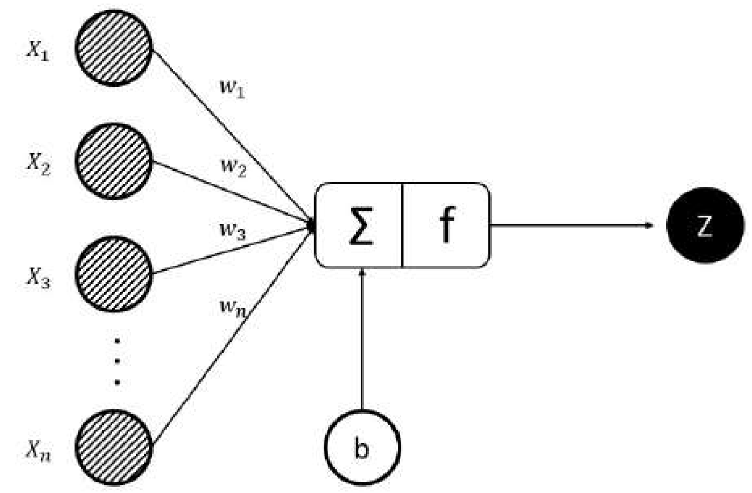


Figure 2: Illustration of a Relationship Between Inputs and Outputs of ANN

The following Equation (1) shows the relationship between nodes, weights, and biases. Here the weighted sum of inputs for a layer passed through a non-linear activation function to another node in the next layer.

$$Z = f(x \cdot w + b) = f\left(\sum_{i=1}^{n} x_i^T w_i + b\right) \quad ...(1)$$

where $X_1, X_2, X_3, ..., X_n$ are inputs vectors and $w_1, w_2, w_3, ..., w_n$ are weights. n is the number of inputs for the final node, f is an activation function and z are the output (Nabipour *et al.*, 2020).

The training process for calculating weights or biases is completed by some rules: initialize the weights or biases for the nodes randomly, perform a forward pass by the current weights or biases by calculating each node output, compare the final output with the actual target, and modify the weights or biases consequently by gradient descent with the backward pass technique also called the backpropagation algorithm (Nabipour *et al.*, 2020).

The Recurrent Neural Network (RNN) is one of the extensively used neural networks for a myriad of processes (Figure 3). Normally, in a neural network, the output is obtained when the input is processed through several layers. The assumption is that the prior consecutive inputs are independent of each other. However, this assumption may not correct in all instances of the process. If we consider the prediction of the stock market at a certain time, it is very important to account for the previous observations (Nabipour *et al.*, 2020).

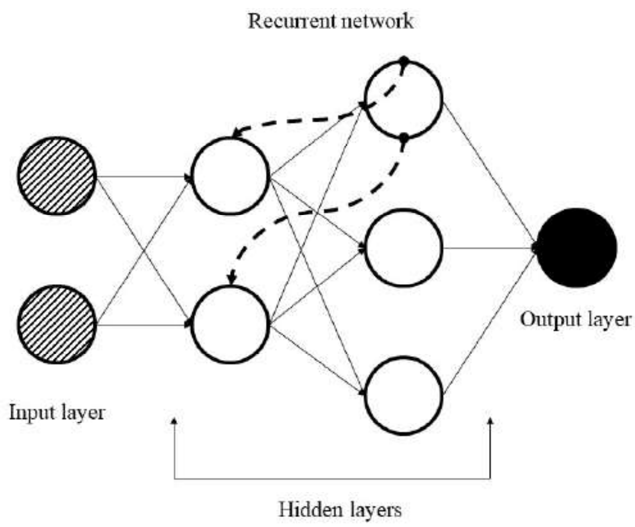


Figure 3: Illustrates of the RNN

This accountability is taken care by the RNN which has multiple neurons to create a network. Each neuron in RNN has a time-varying activation function and each connection between nodes has a real-valued weight that can be modified at each step. According to general architecture, the output of the node (at time $t - 1$) will be passed to the input (at time t) and add the data of itself (at time t) to make the output (at time t); recurrently exploiting the neuron node to flow multiple node elements to create RNN.

The following Equations (2) and (3) show the recursive formulas of the RNN,

$$h_t = \tanh\left(W_t h_{t-1} + W_x x_t\right) \quad ...(2)$$

$$y_t = W_y h_t \quad ...(3)$$

where y_t, h_t, x_t, and W_t are output vector, hidden layer vector, input vector, and weighting matrix respectively.

LSTM is a special type of RNN with a broad range of applications including time series analysis, document classification, speech, and voice recognition. In comparison with the feedforward Artificial Neural Networks (ANNs), the predictions are evaluated through RNNs are dependent on previous estimations. In real, RNNs are not applied widely because they are subject to a few shortcomings which cause impractical runs. However, the difference between LSTM and RNN is that every neuron in LSTM is a memory cell. The LSTM links the subsequent information to the current neuron. Every neuron has three gates (input gate, forget gate, and output gate) (Nabipour *et al.*, 2020).

Through the internal gate, the LSTM can solve the long-term dependence problem of the data. LSTM architecture includes forget gate, input gate, and output gate. The forget gate controls disposing information from the cell, and below Equations (4) and (5) show its related formulas where h_{t-1} is output at the prior time $t - 1$, and x_t is input at the current time into Sigmoid function $S(t)$.

$$f(t) = \sigma\left(W_f \cdot \left[h_{t-1}, x_t\right] + b_f\right) \quad ...(4)$$

$$S(t) = \frac{1}{1 - e^{-t}} \quad ...(5)$$

All W and b are the weight matrices and bias vectors that require to be learned during the training process. $f(t)$ defines how much information will be remembered or forgotten. The input gate defines which new information remember in cell state by the below Equations:

$$i_t = \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right) \quad ...(6)$$

$$\hat{C}_t = tz_t\left(W_c \cdot \left[h_{t-1}, x_t\right] + b_c\right) \quad ...(7)$$

In Equations (6) and (7) the value of i_t is generated to determine how much new information cell state need to be remembered. A tan$h$ function gains an election message to be added to the cell state by inputting the output (h_{t-1}) at the prior time ($t - 1$) and adding the current time t input information (x_t).

$$C_t = f_t \times C_{t-1} + i_t \times \hat{C}_t \quad ...(8)$$

In Equation (8) C_t gets the updated information that must be added to the cell state. The output gate defines which information will be output in cell state. The value of z_t is between 0 and 1, which is employed to indicate how many cells state information that need to output the following Equation (9):

$$z_t = \sigma\left(W_z \cdot \left[h_{t-1}, x_t\right] + b_z\right) \quad ...(9)$$

The result of h_t is the LSTM block's output information at time t, expressed by the below Equation (10):

$$h_t = z_t \times \tanh\left(C_t\right) \quad ...(10)$$

In Machine Learning (ML), the rate at which the machines learn is an important parameter in the optimization method that searches out the step size at each iteration while using the loss function to minimize the loss of information as it moves forward. Since RNN and LSTM networks exhibit a time-series behavior, the datasets are arranged to include features of several days. In contrast to ANN model, all parameters but epochs are constant, the variable parameters are several days included in the training dataset and respective epochs. So, if the number of days in the training set is increased, similarly, the number of epochs is increased to train the models with an adequate number of epochs (Nabipour *et al.*, 2020).

2. Data and Methodology

This study aims to make a short term prediction for the style factor returns and group/sector returns obtained from a cross-section regression factor model, the data considered for the analysis is from September 2013 to June 2023 (approx. 10 years) of four factor returns and nine stock market groups/sectors, style factors include, financial risk, business risk, market risk, while groups/sectors include consumer staples, communication services, consumer discretionary, energy, financials, health care, industrials, information technology, materials.

The initial step before using the data, we first use data cleaning process, which is a process of detecting and removing inaccurate variables from the datasets. We make use of the interquartile range (IQR score) to detect outliers and modify the dataset. We use the min-max scaler which is data preprocessing technique used in machine learning and data analysis to scale and normalize the features of a dataset. This technique transforms the original features such that they are all within a specific range, normally between 0 and 1. This data preprocessing technique is used in machine learning to create a training model less sensitive to the scale of variables.

In this study, the LSTM model is evaluated by training the dataset to predict the target value for 1, 2, 3, 4, 5 and 6 months ahead. Since the LSTM networks exhibit a time-series behavior, the datasets are arranged to include features of several days and the variable parameters are several months is included in the training dataset and respective epochs. So, if the number of months in the training set is increased, similarly, the number of epochs is increased to train the LSTM model with an adequate number of epochs(computations).

For the LSTM model we use 60 neurons on the first layer of LSTM, 120 neurons on the second layer of LSTM. Between the LSTM layers there are hidden layers with dropout of 30% (0.3) neurons. And lastly, we have Dense with 20 neurons as well as just one neuron which is our result on the last step of the model.

We are going to teach our model during 300 epochs (computations) with batch size on each epoch equal to 8. We have added an activation function to the Dense layer with 20 neurons. The activation functions introduce non-linearity to the model and can help the network learn complex patterns in the data. We use the MSE as a parameter for loss function and optimizer 'adam' which is the most popular in tasks of stock price prediction.

For the evaluation measure we use the Mean Squared Error (MSE) which measures the quality of the predictors, and its value is always non-negative (values close to zero are better). The MSE is the second moment of the error (about the origin), and includes both the variance of the prediction model, that is, how widely spread the predictions are from data sample to another and its bias, that is, how close the average predicted value is from the observation.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(A_t - F_t\right)^2 \qquad ...(11)$$

where (A_t) is the observed value and (F_t) is the prediction value and n are the number of samples.

We also make use of the Relative Root Mean Square Error which is a model evaluation measure used in ML which is the standard deviation of the prediction errors in regression work. The prediction errors or residuals show the distance between real values and a prediction model, and how they are spread out around the model. RRMSE is a square root of the average squared differences between predictions and actual observations.

$$RRMSE = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{A_t - F_t}{A_t}\right)^2 \qquad ...(12)$$

Lastly, we make use of the Mean Absolute Percentage Error (MAPE) which is employed to assess the performance of the prediction methods. MAPE is also a measure of prediction accuracy for forecasting methods in the machine learning area, it commonly presents accuracy as a percentage.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \times 100 \qquad ...(13)$$

3. Results

The Long-short Term Memory (LSTM) model was employed in the prediction of the four style factors returns and eight stock market groups/sectors. For the purpose, prediction experiments for 1, 2, 3, 5, and 6 months in advance of time are conducted. The following Tables 1 and 2 show the best parameters where a minimum prediction error is obtained. The LSTM used as a stand-alone deep learning model shows its prediction with relatively low errors in predictions. Among the style factors only the business has 4.03% error (RRMSE) or 14.5% error (MAPE). The rest of the factors are below 2% (RRMSE) forecast error. While for group/sector returns all forecast errors are below 3% (RRMSE) with exception to consumer staples which stands at 8% (RRMSE) forecast error. Overall, the results indicate a better goodness of fit as the lower RRMSE values indicate better model performance. Also suggests that the model's predictions are quite accurate, as the relative error is relatively small. Figure 4 shows the graphical plot of the actual and predicted factor returns for different styles. While Figures 5 and 6 shows the actual and predicted factor returns for different groups/sectors.

Table 1: Style LSTM Forecast

Factor Returns	Epochs	N Month's Forecast	MSE	RRMSE	MAPE
Market	300	6 months	136.26	0.0171	9.47%
Financial Risk	300	6 months	43.93	0.0056	6.14%
Business Risk	300	6 months	897.3	0.0403	14.49%
Market Risk	300	6 months	91.9872	0.0111	8.66%

Figure 4: Style LSTM Forecast

Table 2: Group/Sector LSTM Forecast

Group/Sector Returns	Epochs	N Month's Forecast	MSE	RRMSE	MAPE
Consumer Discretionary	300	6 months	43.436	0.00674	6.46%
Consumer Staples	300	6 months	497.83	0.0792	20.96%
Financials	300	6 months	176.6	0.02052	10.67%
Health Care	300	6 months	118.25	0.0176	10.21%
Industrials	300	6 months	202.13	0.0282	12.69%
Information tech	300	6 months	192.73	0.0235	11.44%
Materials	300	6 months	69.538	0.0117	7.41%
Energy	300	6 months	34.878	0.00379	4.59%
Communications services	300	6 months	114.06	0.0198	9.55%

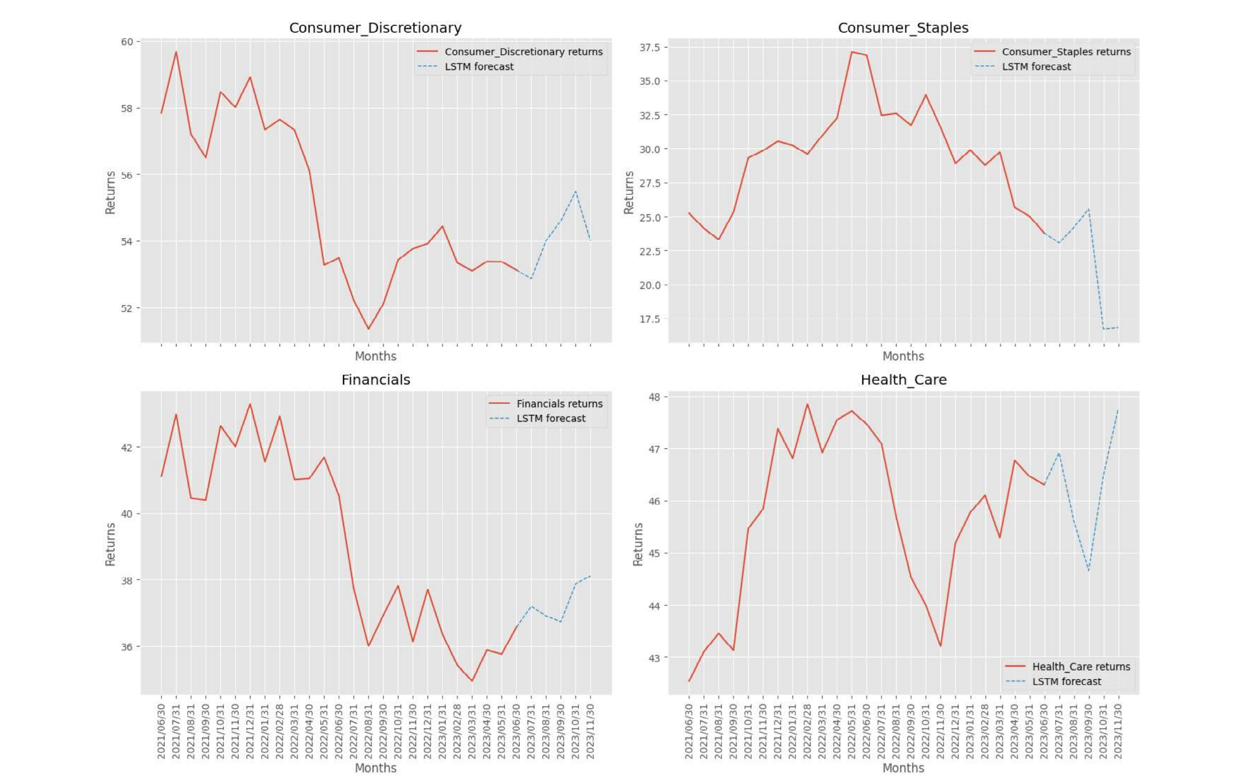


Figure 5: Group/Sector LSTM Forecast

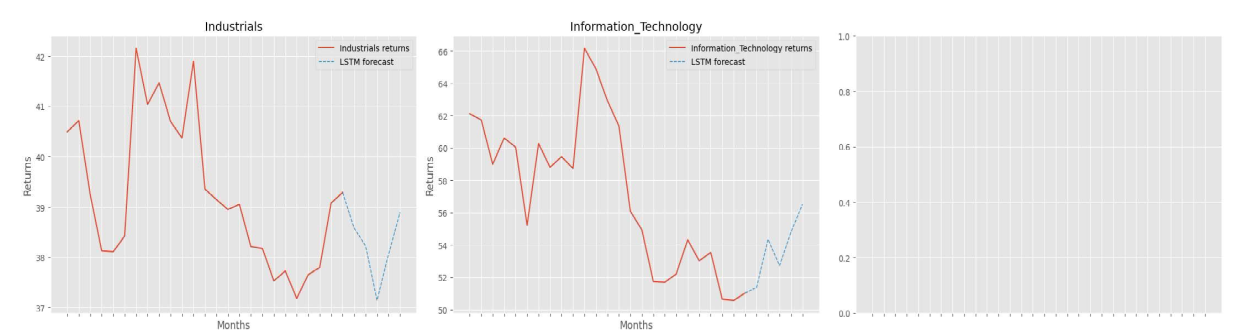


Figure 6: Group/Sector LSTM Forecast

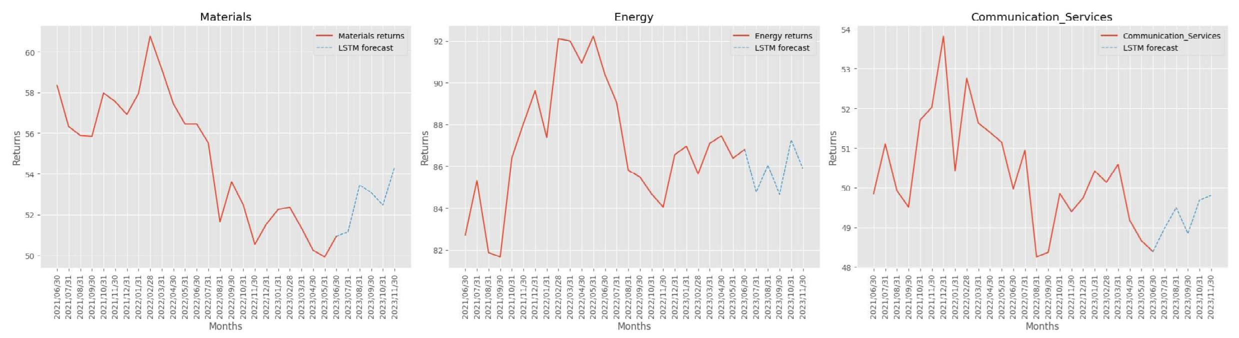


Figure 6 (Cont.)

4. Conclusion

There are various dimensions from which portfolio managers and investors can differentiate stocks. In most cases, portfolio managers often find it easier if they can combine several of these views. Factor models are constructed to serve this purpose, they can detach the effects of multiple variables acting in interdependence with one another. Portfolio managers also use these models to identify for sources of expected returns and controlling portfolio exposures. Deep learning models are more accurate in predicting these sources of expected returns with their time-series behavior they can accurately predict markets where the effects of multiple market variables have interdependence. In this paper we showed that the LSTM model can accurately predict style factor returns and group/sector returns derived from a cross-section regression model. This study seeks to contribute to the limited research on the prediction of stock market returns. Amongst the limitations of the study is that we only used one deep learning model, the LSTM model. More results can be obtained on the accuracy of the LSTM model by comparing LSTM model with other existing deep learning models.

References

Abe, M. and Nakayama, H. (2018). Deep Learning for Forecasting Stock Returns in the Cross-Section. In Advances in Knowledge Discovery and Data Mining: 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, VIC, Australia, June 3-6, 2018, Proceedings, Part I 22, 273-284. Springer International Publishing.

Chen, S. and He, H. (2018). Stock Prediction Using Convolutional Neural Network. IOP Conference Series, *Materials Science and Engineering*, 435, 012026. https://doi.org/10.1088/1757-899X/435/1/012026.

Fischer, T. and Krauss, C. (2018). Deep Learning with Long Short-term Memory Networks for Financial Market Predictions. *European Journal of Operational Research*, 270(2), 654-669. https://doi.org/10.1016/j.ejor.2017.11.054.

Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., Salwana, E. and S., S. (2020). Deep Learning for Stock Market Prediction. *Entropy*, 22(8), 840. https://doi.org/10.3390/e22080840.

Menchero, Jose. (2010). Characteristics of Factor Portfolios, MSCI Barra Research Paper No. 2010-08. http://dx.doi.org/10.2139/ssrn.1601414

Cite this article as: Sanele Makamo (2023). Predicting Style Factor Returns and Group/Sector Returns Using Long and Short-Term Memory ("LSTM") Deep Learning Neural Networks. *International Journal of Artificial Intelligence and Machine Learning,* 3(2), 20-27. doi: 10.51483/IJAIML.3.2.2023.20-27.