INVESTIGATE THE APPLICATION OF ATTENTION IN STOCK INDEX TIME SERIES PREDICTION

Yao Gu, Zijian Zhao, Yuxiang (Alvin) Chen

Northwestern University, USA

ABSTRACT

The application of deep learning approaches to stock price prediction has been a popular but highly challenging tasks for researchers. Attention [1, 2], a very useful mechanism as the length of the input data increases, has been proved useful for various natural language processing tasks but underutilized in current stock prediction models. This study presents a comparison among three models: 1. LSTM Encoder-Decoder Network; 2. LSTM Encoder-Decoder Network with Attention; 3. Transformer. They are trained to perform the same task - predicting the future S&P500 market index move with historical data. The performance of three models are compared through three metrics - R, TheiU and MAPE, the final quantitative comparisons demonstrate the superiority of using attention in this task.

1. INTRODUCTION

Stock index value prediction is a highly challenging study for both research and industry. Applying Deep Learning techniques to financial time series datasets has been a popular methodology. Existing prediction models are mostly based on Multi-Layer Perceptrons, Recurrent or Convolutional Neural Networks[3, 4, 5]. However, Attention is a very useful mechanism as the length of the input data increases but underutilized in current stock prediction models. Stock index prediction is a challenging problem because stock index datasets have the characteristics of irregularity and complex inner-connections, which is a big barrier for learning the trends of the datasets.

We built three deep learning models with different extent of application of attention to mine the same financial time series datasets and predict stock index price moves and evaluate their performances. The three models we chose are 1. LSTM Encoder-Decoder Network; 2. LSTM Encoder-Decoder Network with Attention; 3. Transformer.

This research problem is an important one because we can verify what extent of attention mechanism is the most appropriate for financial time series prediction and effective model construction. No research has been conducted with a focus on comparing these three models with different attention applications. The conclusion will be meaningful for future new model development. This is also a rewarding problem to solve because mining huge amounts of financial datasets can bring a lot of benefits for investing judgements.

2. METHODOLOGY

2.1. Long-short term memory Encoder-Decoder model

For the baseline, we used a LSTM based model. It's a standard seq2seq architecture, in which two recurrent neural networks work together to transform one sequence to another. The encoder condenses an input sequence into a hidden vector, and the decoder unfolds that vector into a new sequence. In our settings, the encoder is a 3-layer LSTM whose inputs are 30 days of stock data, the decoder is a single LSTM cell which will be used repeatedly to generate the stock index of the future 5 days. The optimizer is an Adam optimizer with default parameters, we used MSE loss to train the model.

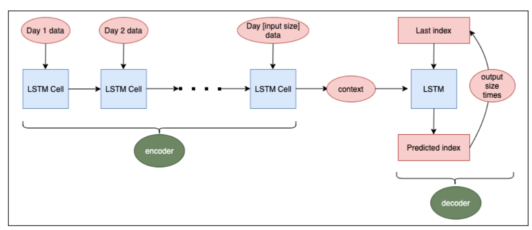

Fig. 1. LSTM model

2.2. Long-short term memory Encoder-Decoder model with Attention

The attention model basically has the same architecture as the LSTM model, but it has one more attention layer which can calculate the weight using the encoder output. In our model we used Bahdanau Attention. The other settings are the same as the first LSTM model, a defualt Adam optimzier and MSE criterion.

2.3. Transformer Model

Our transformer model is build based on the paper Transformer [1]. We replace the embedding layer of NLP tasks with a 1-D convolutional layer for projecting the time-series input into a length=d_model vector. Our transformer model uses 6 layers transformer encoder and 3 layers decoder, 8 heads self-attention, and d_model=512. We use an SGD optimizer with CosineAnnealing learning rate decay and MSE loss to train the model.

3. DATA DESCRIPTION

We used the dataset used in [3] and we decided to use daily data of S&P500 index between 2008/07/02 and 2016/09/30. There are 2079 daily data points. Here is the link to our dataset source:

https://github.com/timothyyu/wsae-lstm/tree/master/data/raw

We select three sets of input variables: 1. 5 standard basic trading data from stock market which includes Open, Close, High, Low and Daily Trading Volume 2. 10 commonly used technical indicators index time series data. 3. 4 macroeconomic variable. These input variables cover common information associated with index price moves.

4. EXPERIMENT METHOD

4.1. Train & Prediction Approach

We decided to use 30 days data as inputs to predict the 31-th day's close price move percentage. Thus, we grouped data into separate 31 day groups. Each group is incremented by 1 day.

We separated the data into three sets - 80% for training set, 10% for validation set and 10% for test set. training set and validation set are generated by randomly split, and test set contains the data from specific date.

We applied subsection prediction method [6]. This method has three parts. Firstly, we used training set to train the model. Secondly, we used the validation set to verify the optimal model setting. Lastly, we used testing set to measure the performance of the model.

4.2. Performance Measure

We chose three classical indicators used by previous financial time series papers([3, 7]) - R, MAPE and Theil U as the common indicators used for examining all our three models' performance. MAPE is a measure of error size. R is a measure of the linear correlation between two variables. Theil U is a relative measure of the difference between two variables. It has more weight to large errors because it squares the difference. The equations are presented below:

$$\text{MAPE} = \frac{\sum_{t=1}^{N} |\frac{y_t - y_t^*}{y_t}|}{N} \tag{1}$$

$$\text{R} = \frac{\sum_{t=1}^{N} (y_t - \overline{y_t})(y_t^* - \overline{y_t^*})}{\sqrt{\sum_{t=1}^{N} (y_t - \overline{y_t})^2 \sum_{t=1}^{N} (y_t^* - \overline{y_t^*})^2}} \tag{2}$$

	MAPE	R	Theil U
RNN*	0.018	0.847	0.012
WSAEs-LSTM*	0.011	0.946	0.007
LSTM	0.011	0.872	0.008
LSTM + Attention	0.0075	0.947	0.005
Transformer	**0.0055**	**0.980**	**0.0038**

Table 1. Prediction results camparison of different models for Close Price with MAPE, R, Theil U values (LSTM, LSTM+Attention, Transformer are our work).

$$\text{Theil U} = \frac{\sqrt{\frac{1}{N}\sum_{t=1}^{N} (y_t - y_t^*)^2}}{\sqrt{\frac{1}{N}\sum_{t=1}^{N} (y_t)^2} + \sqrt{\frac{1}{N}\sum_{t=1}^{N} (y_t^*)^2}} \tag{3}$$

In the three equations above, y_t is the true index close price percentage move, and y_t^* is the corresponding predicted value for it. Better performance is indicated by higher R, lower MAPE and lower Theil U.

5. RESULTS

Table 1 presents the performance metrics of two models from our main reference paper performing similar tasks report and our three models on our testing set.

In the table, the first two models with * sign are results from our primary reference paper. The later three models are out models. The results show that the traditional RNN model has the worst performance in all three metrics. The WSAEs-LSTM model performs a little better than our LSTM Encoder-Decoder model. But our LSTM + Attention & Transformer models perform better.

The most important finding is that, among the three models we developed, LSTM + Attention model is better than LSTM without Attention in all three metrics and Transformer model is even better than LSTM + Attention model in all three metrics. We can conclude that the application of attention improves model performance. The transformer model, which only applies attention, performs the best. This shows that complete application of attention without RNN could lead to the best performance.

Fig. 2 shows the prediction results of three models on the testing set and the true index close price (blue line).

From the figure we can see that the transformer model gives out the best prediction (orange line) that is very close to the true value line (blue line). Also, the LSTM + Attention model's predicted values (green line) are closer to true values than the LSTM model's predicted values (purple line). These observations match with the conclusion we got from the three performance metrics.

Thus, we conclude that among the three models, transformer model performs the best in predicting S&P500 index

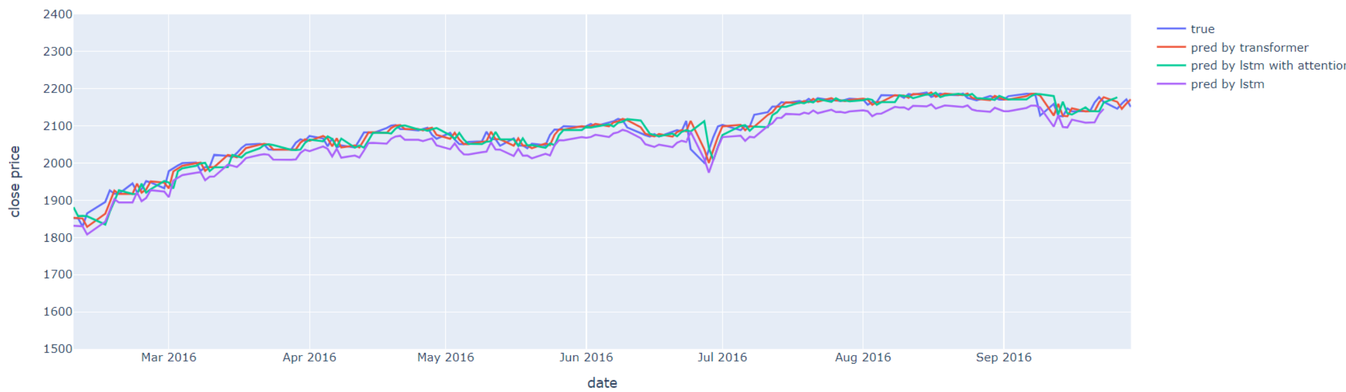

Fig. 2. Predictions of Close Price based on our three models

daily movement in our experiment setting. Although this does not mean transformer will always perform the best in any financial time-series prediction tasks (eg. different data freqeuncy, input variables), the finding still indicates that transformer has a high potential to be applied in financial time-series prediction tasks and should be payed more attention to in future similar researches.

In the next step, we would like to firstly, conduct similar experiments on other stock market indexes around the world such as HangSeng, Nasdaq to see whether our conclusion is consistent in different financial products. We currently have the appropriate dataset to do this so this will be our first step. Secondly, we would like to further develop our transformer model. Possible approaches include adding other components such as de-noising component (wavelet transformation, auto-encoders, etc.), adding more candidate input variables to provide more information, further tuning hyper parameters. Thus, we could come up with an actual daily adjusted index transaction strategy with good investment return.

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L. Kaiser, and Illia Polosukhin. Attention is all you need. *ArXiv*, abs/1706.03762, 2017. `https://arxiv.org/abs/1706.03762`.

[2] Jiayu Qiu, Bin Wang, and Changjun Zhou. Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PLOS ONE*, 15(1):1–15, 01 2020. `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0227222`.

[3] Wei Bao, Jun Yue, and Yulei Rao. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLoS ONE*, 12, 07 2017. `https://www.researchgate.net/publication/318991900_A_deep_learning_framework_for_financial_time_series_using_stacked_autoencoders_and_long-short_term_memory`.

[4] J. Liu, H. Lin, L. Yang, B. Xu, and D. Wen. Multi-element hierarchical attention capsule network for stock prediction. *IEEE Access*, 8:143114–143123, 2020. `https://ieeexplore.ieee.org/abstract/document/9159584`.

[5] Yaping Hao and Qiang Gao. Predicting the trend of stock market index using the hybrid neural network based on multiple time scale feature learning. *Applied Sciences*, 10(11):3961, Jun 2020. `https://www.mdpi.com/2076-3417/10/11/3961/htm`.

[6] Jacinta Chan Phooi M'ng and Mohammadali Mehralizadeh. Forecasting east asian indices futures via a novel hybrid of wavelet-pca denoising and artificial neural network models. *PLOS ONE*, 11(6), 06 2016.

[7] Zhiqiang Guo, Huaiqing Wang, Quan Liu, and Jie Yang. A feature fusion based forecasting model for financial time series. *PLOS ONE*, 9(6):1–13, 06 2014.