Long Short-term Memory Recurrent Neural Networks Models to Forecast the Resource Usage of MapReduce Applications

Yangyuan Li, Tien Van DO

Abstract: The forecasting of the resource usage of MapReduce applications plays an important role in the operation of cloud infrastructure. In this paper, we apply long short-term memory recurrent neural networks to predict the resource usage of three representative MapReduce applications. The Results show that the Long Short-term Memory Recurrent Neural Networks models perform higher prediction accuracy than persistence ones. Predictions of other usage parameters show similar accuracy with persistence one. The improper configuration parameters of Long Short-term Memory Recurrent Neural Networks possibly result in few of worse prediction.

Keywords: MapReduce application; resource usage parameters; LSTM-RNN model; forecasting

Introduction

MapReduce applications are developed to process big data [1,2] in public clouds and private clouds. Therefore, forecasting the resource usage of MapReduce application is crucially needed for cloud operators. Yan Ling et al. [3] predicted the total execution time for MapReduce applications with linear regression model and correction neural network model. Issa, J A et al. [4] proposed an estimation model to estimate total processing time versus different input sizes under a given processor architecture. H. Yang et al. [5] predicted the total execution time of a workload under different Hadoop configurations with support vector regression models. However, most of them only estimated the total execution time of MapReduce jobs.

In this paper, we apply multivariate long-short term memory recurrent neural networks (LSTM-RNN) [6] to forecast resource usage parameters (CPU usage(%), memory usage(%), read rate(MB/S) and write rate(MB/S)) of three MapReduce benchmark applications. LSTM-RNN is an evolutional version of recurrent neural networks that can effectively avoid gradient vanishing and exploding. The LSTM unit is composed of a memory cell state, an input gate, an output gate, and a forget gate. Moreover, the structure of LSTM-RNN is capable to learn long-term dependencies of time series data. We use LSTM-RNN to predict the resource usage of three applications (Wordmean, Grep, and Teragen). The first two applications calculate the average length of words and the matches to a regex in a text file, respectively.

The prediction with LSTMs Models

We use LSTM-RNN models with one hidden LSTM layer and one output layer. To choose a suitable configuration of LSTM-RNN, we identified the hyper parameters (epoch size, batch size, neurons number, time steps) by performing the tuning configuration prior to the training and the forecasting activity. Then we apply a one-shot method [7].

The experimental datasets of three applications are collected from the following scenario:

- Bare metal servers with an Intel CoreTM i5-4670 CPU 3.40GHz 4 cores, 16GB Kingston HyperX Black DDR3 1600MHz RAM and 250GB 7200RPM hard drive.
- Hadoop version 2.7.3 and MapReduce v2 in Ubuntu server 16.04.3 LTS, kernel 4.4.0-62-generic the block size is set to 512MB.

The root mean squared error (RMSE) [8] is used to evaluate the accuracy of prediction as it punishes large errors and results in a score that is in the same units as the forecast data. We establish a baseline performance for each usage parameter by developing a persistence model that provides a lower acceptable bound of performance on the test set. RMSE with the use of the persistence forecast (naive forecast) and LSTM-RNN forecast on the dataset is presented in Table 1.

The results show that the intensive resource usage parameters are able to obtain better predictive performance after using LSTM-GNN models.

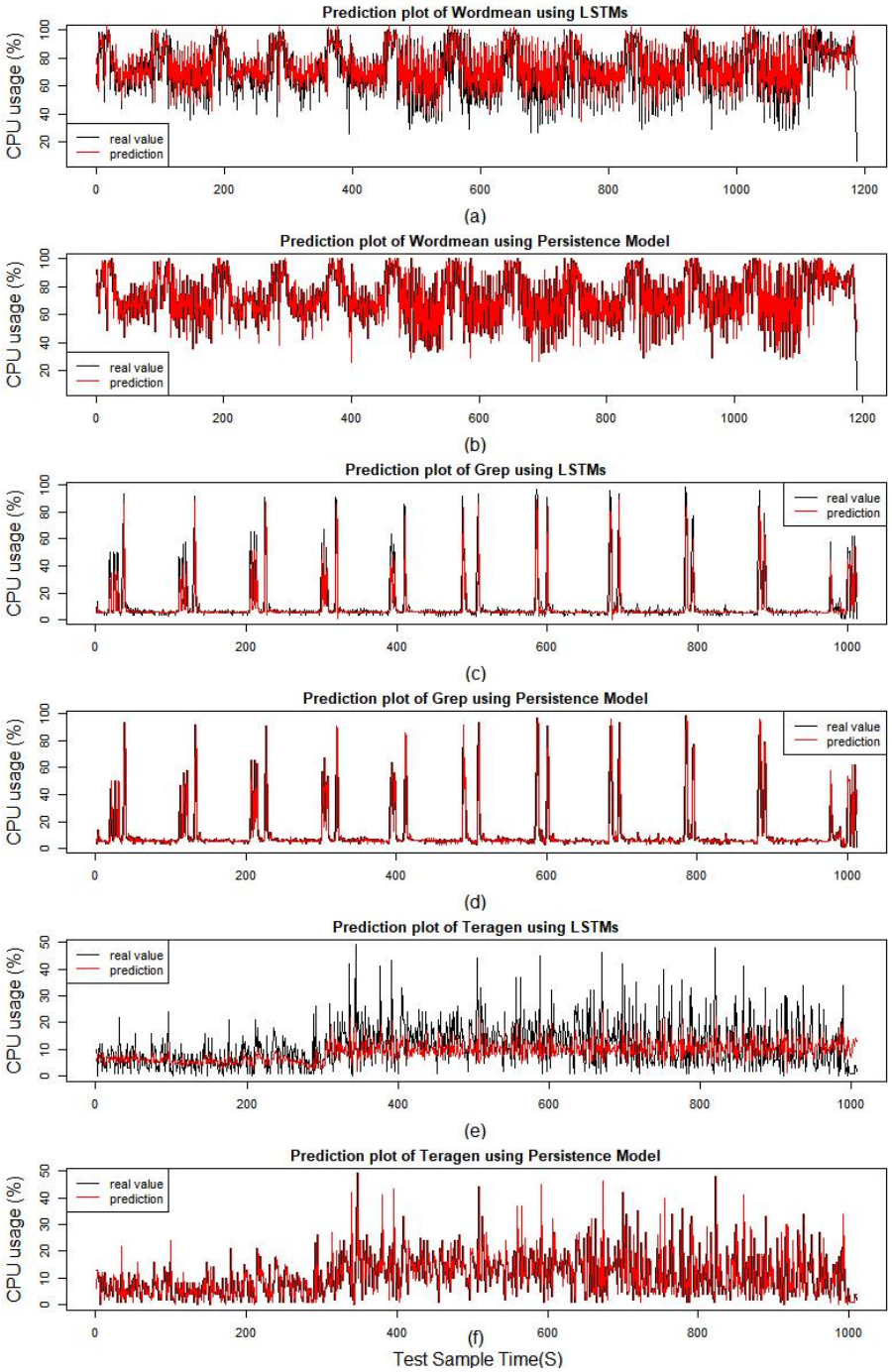

Figure 1: Forecasting comparison time series plot

Modeling Method /Application name	RMSE of CPU	RMSE of Memory	RMSE of Read rate	RMSE of Write rate
LSTM-RNN /Wordmean	14.133	0.371	2.608	0.137
Persistence model / Wordmean	25.080	0.386	3.120	0.207
Improvement rate after using LSTM-RNN	43.65%	3.89%	16.41%	33.82%
LSTM-RNN / Grep	12.492	0.393	2.889	0.120
Persistence model / Grep	13.581	0.406	2.700	0.166
Improvement rate after using LSTM-RNN	8.02%	3.20%	-7.00%	27.71%
LSTM-RNN / Teragen	8.198	0.103	0.055	5.986
Persistence model / Teragen	10.410	0.090	0.051	8.013
Improvement rate after using LSTM-RNN	21.25%	-14.44%	-7.84%	25.30%

Table 1: Prediction accuracy comparison

We draw the forecasting time series plot of CPU usage parameters between real value and prediction in Figure 1 to exhibit the forecasting performance of LSTM-RNN models. In Figure 1, the sub-figure [(a), (b)], [(c), (d)], and [(e), (f)] are used to show CPU usage time series comparison plot for Wordmean, Grep, and Teragen application respectively using LSTM-RNN models and persistence models. In Figure 1, the CPU usage forecasts of three applications with LSTM-RNN models show higher accuracy than predictions with persistence models. The persistence models only shift to right side 1 time-step.

Conclusions

We have applied LSTM-RNN model to forecast the usage parameters of MapReduce applications. The LSTM-RNN models show higher forecast accuracy than persistence models for the CPU usage prediction. The forecast accuracy for the rest of usage parameters show similar results with persistence models. Few of usage prediction get worse result possibly due to the improper configuration parameters of Long Short-term Memory Recurrent Neural Networks.

References

[1] V. K. Vavilapalli,A. C. Murthy,C. Douglas,S. Agarwal,M. Konar,R. Evans,T. Graves,J. Lowe,H. Shah,S. Seth,B. Saha,C. Curino, O. O'Malley, S. Radia, B. Reed, and E. Baldeschwieler. Yet Another Resource Negotiator, *Apache Hadoop YARN* , in Proceedings of the 4th Annual Symposium on Cloud Computing,p.5:1–5:16, 2013.

[2] A. S. Foundation, Apache hadoop. http://apache.hadoop.org. p. Last Published: 12/18/2017, 2017.

[3] Y. Ling, F. Liu, Y. Qiu, and J. Zhao. Prediction of total execution time for MapReduce applications, in 6th International Conference on Information Science and Technology, ICIST 2016, 2016, pp. 341-345.

[4] J. A. Issa. Performance Evaluation and Estimation Model Using Regression Method for Hadoop WordCount, IEEE Access, vol. 3, pp. 2784-2793, 2015.

[5] H. Yang, Z. Luan, W. Li, and D. Qian. MapReduce workload modeling with statistical approach,*Journal of Grid Computing*, vol. 10, no. 2, pp. 279-310, 2012.

[6] S. Hochreiter and J. Urgen Schmidhuber. LONG SHORT-TERM MEMORY, *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[7] S. Ta'asan, G. Kuruvila, and M. D. Salas. Aerodynamic design and optimization in one shot, in 30th Aerospace Sciences Meeting and Exhibit, 1992.

[8] T. Chai and R. R. Draxler. Root mean square error (RMSE) or mean absolute error (MAE)? - Arguments against avoiding RMSE in the literature, *Geoscientific Model Development*, vol. 7, no. 3, pp. 1247-1250, 2014.