Recurrent Neural Networks Application to Forecasting with Two Cases: Load and Pollution

Qing Tao[1], Fang Liu[1], and Denis Sidorov[2]

[1] School of Automation, Central South University, 932 Lushan S Rd, Changsha 410083, China
csuliufang@csu.edu.cn
[2] Energy Systems Institute of Russian Academy of Sciences, 130 Lermontov Str., Irkutsk 664033, Russia
contact.dns@gmail.com

Abstract. Forecasting problems exist widely in our life. Its purpose is to enable decision makers to make effective responses to future changes. The traditional prediction methods based on probability and statistics cannot guarantee the accuracy of multivariable dynamic prediction under the background of high randomness and big data. In recent years, with the improvement of hardware computing ability and the large-scale increase of training data, deep learning has been widely applied in the field of forecasting. This paper focuses on the analysis of the application of recurrent neural networks (RNN), an advanced algorithm in deep learning, in the forecasting task. The forecasting models based on long short-term memory (LSTM) and gated recurrent unit (GRU) were established respectively, and the real data of power load and air pollution were verified. Compared with traditional machine learning algorithms, the simulation proves the superiority of the forecasting model based on RNN.

Keywords: Deep learning, LSTM, GRU, Forecasting.

1 Introduction

Time series forecasting has a wide range of applications in weather, finance, transportation, industry, agriculture, etc. The forecasting of time series provides important guidance for decision makers to adopt appropriate strategies. To solve forecasting problem, some classical forecasting methods including exponential smoothing (ES), moving average model (MA), autoregressive integrated moving average model (ARIMA) [1], etc. were proposed. However, these traditional forecasting methods are simple in form, unable to explore the intrinsic relationship of a large number of data and provide high-precision prediction results. Deep learning has strong non-linear fitting and independent learning ability, which has been widely used in various fields. It is of great significance to apply deep learning method to time series forecasting.

Till now, various forecasting approaches have been proposed, which can be mainly classified into the traditional statistical models and artificial intelligence models. The

former develops for a long time and is mature at the same time. It mainly takes mathematical statistics as theoretical knowledge, and uses functions to model the relationships among various data in time series. It mainly includes regression model, exponential smoothing (ES) model, moving average (MA), autoregressive integrated moving average (ARIMA) and so on [2]. But such models cannot model non-linear and multivariate data, the accuracy is limited.

The artificial intelligence models can be classified into the shallow machine learning methods and deep learning based models. The typical machine learning forecasting methods includes support vector regression (SVR) [3], genetic algorithm (GA) [4], artificial neural network (ANN) [5], etc. Such methods have strong adaptive ability, autonomous learning ability and generalization ability for non-linear structures, and have great advantages over traditional methods. However, there are still many problems, such as slow learning speed, easy to fall into local optimum and so on.

With the development of big data technology and the rapid progress of hardware computing capacity, deep learning has been widely used in computer vision and natural language processing, and has also attracted much attention in the field of time series forecasting. Common methods include deep belief networks (DBN) [6], recurrent neural networks (RNN) and its variants like long short-term memory (LSTM) and gated recurrent unit (GRU) [7].

The main focus of the paper is to solve forecasting problem of deep learning models like LSTM and GRU. Therefore, two meaningful topics, air pollution forecasting and power system load forecasting, are selected for analysis. The data and input variables selection are described in Section 2. Section 3 describes the recurrent neural networks based forecasting model. Experiments and discussion are illustrated in Section 4. Finally, Section 5 gives the conclusion.

2 Data Description and Time Series Analysis

This section describes two data sets (Beijing PM2.5 dataset and Germany's electrical grid dataset) for training and testing forecasting model and identifies input variables by Granger non-causality test.

2.1 Beijing PM2.5 Dataset

This study uses Beijing PM2.5 data available on UCI machine learning repository, which contains the PM2.5 data and meteorological data in Beijing [8]. This dataset in hourly resolution covers data from January 2, 2010 to December 31, 2014, contains 8 attributes including PM2.5 concentration, dew point, temperature, air pressure, wind direction, wind speed, snowfall, and rainfall. The data set consists of 43800 examples, 30000 rows were used for training, 8000 rows for validation, and the remaining 5800 rows for testing. Fig. 1 shows the changes of PM2.5 in the dataset.

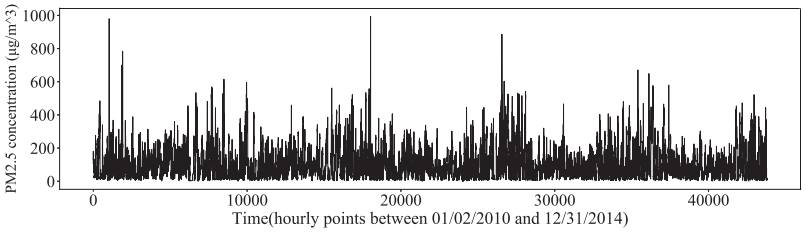


Fig. 1. Changes of PM2.5 concentration in Beijing PM2.5 dataset.

As shown in the Fig. 1, PM2.5 changes have no significant periodic characteristic, which brings sertain difficulties to the prediction. In order to make full use of effective information to make prediction, *Granger non-causality test* [9] is used to make variables selection, which can determine whether one time series can predict another. This article uses the convenient method provided by the *statsmodels* package[1], a statistics module in Python. There is a *Null hypothesi*s that: the series of *X*, does not Granger cause the series of *Y*. If the P-Value is less than a significance level (0.05) then one can reject the null hypothesis and conclude that the said lag of *X* is useful for the forecast of *Y*.

Table 1. Granger non-causality tests of PM2.5 and meteorological time series on Beijing PM2.5 dataset.

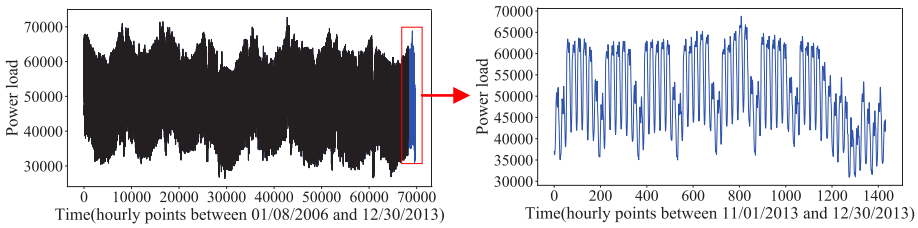
Variables	Dew point	Temperature	Pressure	Wind direction	Wind speed	Snow	Rain
P	0.0013	0.0217	0.7054	0.0000	0.0000	0.1468	0.0000

Granger non-causality tests of Beijing PM2.5 Dataset are shown in Table 1, one can find that P-value of pressure test is exceeds 0.05, indicating that pressure time series is not suitable for predicting PM2.5 concentration. Therefore, pressure is not considered as the input of the model in the prediction task.

2.2 Germany's Electrical Grid Dataset

For load forecasting, the real data of Germany's electric grid [10] was used, which contains 18 various features including average daily temperature in Hamburg (T1), Munich (T2), Stuttgart (T3), Bochum (T4), current load, an indicator of working days and holidays in Germany, day of week, day of year, time of day, load a day ago, load value an hour ago, load value a weekn ago, average load for yesterday, minimum load for yesterday, and exponential moving averages (EMA) with periods 12, 24, 48, 168 hours. The dataset duration starts from 2006-01-08 to 2013-12-30, total 69713 hourly examples, 60953 rows were used for training and validation, and the remaining 8760 rows for testing.

[1] http://www.statsmodels.org/stable/generated/statsmodels.tsa.stattools.grangercausalitytests.html



(a) Changes of load within eight years (b) Changes of load within two months

Fig. 2. Periodic characteristics of load changes.

The Fig. 2 above shows the changes of load on the whole data set. It can be seen that there are obvious year-wise and the week-wise periodic characteristics and it is a stationary time series. Therefore, period-based historical data (load values with time shift) can greatly provide prediction information. Similarly, Granger non-causality tests were conducted on the data set, and the results are shown in Table 2. One can find that P-value of "is holiday" test and "day of year" test are greater than 0.05. That is to say, they can't effectively improve forecasting performance, so in the case study, we eliminated these two variables.

Table 2. Granger non-causality tests of load and other time series on Germany's electrical grid dataset.

Variables	P	Variables	P	Variables	P
T1	0.0000	time of day	0.0000	yest.min	0.0000
T2	0.0000	yest.load	0.0000	EMA12	0.0000
T3	0.0000	last.hour.load	0.0000	EMA24	0.0000
T4	0.0000	last.week.load	0.0009	EMA48	0.0000
is.holiday	0.2018	day of year	0.1876	EMA168	0.0000
weekday	0.0000	yest.mean	0.0000		

3 Methodologies

The Recurrent Neural Networks (RNN) is a special neural networks developed for time processing and learning sequences [11], which can deal with the temporal relation of sequences data by memorize the previous information and apply it to the current input. But there are some drawbacks with simple RNN, like the vanishing gradient and exploding gradient, which makes it difficult for RNN to learn the long-term dependencies task. The general method to solve these problems is to change the structure of RNN, such as Long Short-Term Memory Unit (LSTM) and Gated Recurrent Unit (GRU). Both of them are an improved structure of RNN, which advantage is to overcome the problem of long-term dependencies in recurrent neural networks.

3.1 Long Short Term Memory (LSTM)

LSTM [12] can track long-term information through the gates it contains. The structure of the LSTM is shown in Fig. 3 (a), where *i, f* and *o* are the input, forget and output gate, respectively. *c* and $\tilde{c}$ denote the memory cell and the new memory cell content. In an LSTM unit, there are basically three gates, input gate, forget gate and output gate, which determine what information to store. These three gates are computed by:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad , \tag{1}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad , \tag{2}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad . \tag{3}$$

where σ is the logistic sigmoid function, x_t is the input, h_{t-1} is the output at the previous time, $W_{(\cdot)}$ and $U_{(\cdot)}$ are weight matrices which are learned, and $b_{(\cdot)}$ is the bias of each gate.

The memory cell is updated by the previous memory and the new memory content:

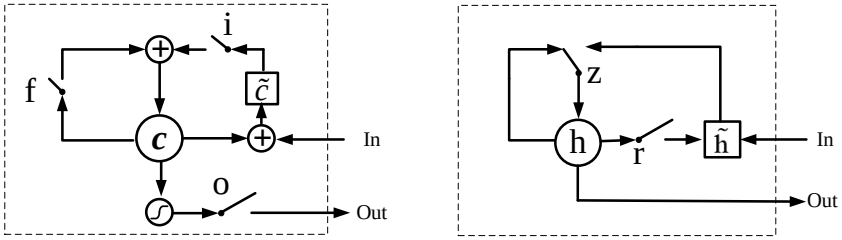
$$c_t = f_t c_{t-1} + i_t \hat{c}_t \quad , \tag{4}$$

where the new memory content is

$$\hat{c}_t = \sigma(W_c x_t + U_c h_{t-1} + b_c) \quad . \tag{5}$$

The next output of LSTM cell is computed by:

$$h_t = o_t \tanh(c_t) \quad . \tag{6}$$



(a) Long short-term memory (LSTM) (b) Gated recurrent unit (GRU)

Fig. 3 [14]. Illustration of (a) LSTM and (b) GRU.

3.2 Gated Recurrent Unit (GRU)

Similar to LSTM, gated recurrent unit (GRU) neural networks can also learn long-term dependencies [13]. Research shows that GRU has similar performance to LSTM, and it requires less computation [14].

The graphical illustration of GRU unit is shown in Fig. 3(b), where *r* and *z* are the reset and update gate, *h* and $\tilde{h}$ are the activation and the candidate activation. The GRU contains two gates, update gate and reset gate, the update gate defines which information to keep around, and the reset gate specifies how to combine the previous

state information with the new input information. The update gate and reset gate are computed by:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad , \tag{7}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad . \tag{8}$$

where σ is the activation function, x_t is the input, h_{t-1} is the previous activation, $W_{(\cdot)}$, $U_{(\cdot)}$ and $b_{(\cdot)}$ are weight matrices and bias of each gate.

The activation of GRU is updated by the previous activation h_{t-1} and the condidate activation $\hat{h}_t$:

$$h_t = (1 - z_t) h_{t-1} + z_t \hat{h}_t \quad , \tag{9}$$

where the condidate activation is

$$\hat{h}_t = \sigma(W_h x_t + U_h r_t h_{t-1} + b_h) \quad . \tag{10}$$

3.3 The Proposed RNN-Based Forecasting Method

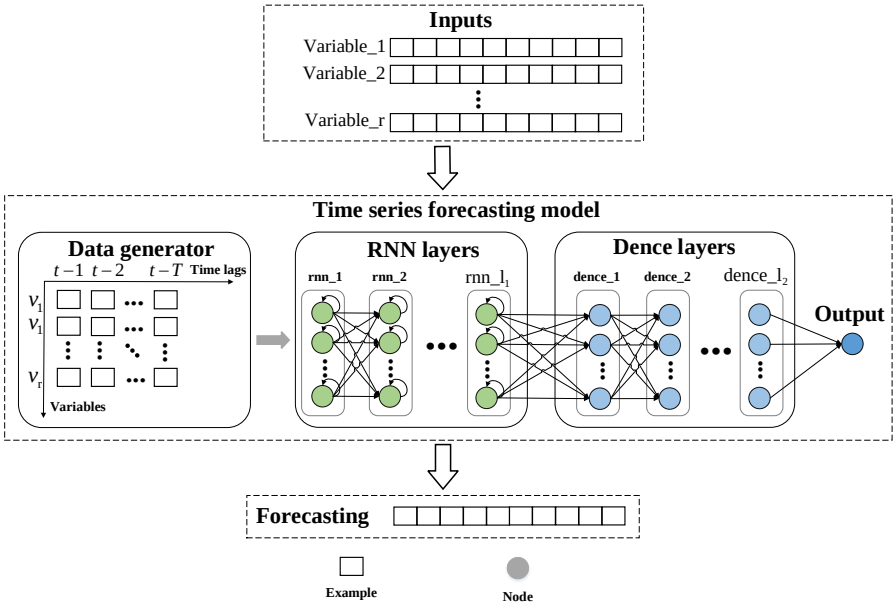


Fig. 4. The RNN-based forecasting model.

The forecasting model is shown in Fig. 4. For this model, the data generator generates examples with *T*-time lags by multivariable inputs. These examples are fed into RNN layers and full-connected layers, respectively. There is a one-neuron output layer at the end of the model, which generates the predicted value directly.

4 Experiments and Discussion

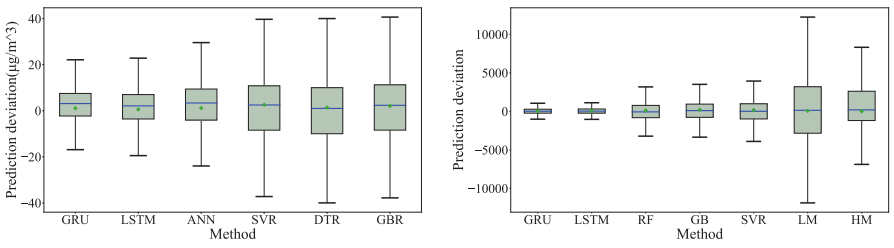
Three error evaluation metrics, mean absolute error (MAE), root mean square error (RMSE) and symmetric mean absolute percentage error (SMAPE) are used to evaluate the forecasting performance [15]. Deep learning models like LSTM, GRU and ANN are trained by *Keras* platform with *TensorFlow* as backend. Traditional machine learning models such as SVR, DTR and GBR are trained by *Scikit-learn* machine learning library. Before training, each time series needs to be standardized by removing the mean and scaling to unit variance.

4.1 PM2.5 Forecasting

Four models were selected to evaluate the performance of the proposed models, i.e., support vector regression (SVR), decision tree regression (DTR), gradient boosting regression (GBR), and artificial neural networks (ANN). Some settings of the experiment are as follows, the batch size of training is 50, the optimizer is *RMSprop*, and the epochs of training is 50. The errors evaluation of PM2.5 forecasting are shown in Table 3. It can be seen that the errors of GRU and LSTM are much smaller than those of SVR, DTR, GBR and ANN, indicating that the prediction accuracy of RNN-based model is higher than that of shallow machine learning model.

Table 3. Errors analysis of PM2.5 forecasting. Bold values indicate the smallest RMSE, MAE and SMAPE values.

Methods	Parameter setting	RMSE	MAE	SMAPE
SVR	kernel = 'rbf', C = 16, gamma = 0.1	27.7064	16.7608	0.2637
DTR	criterion = 'mae', max_depth = 8	28.8528	17.1611	0.2574
GBR	loss = 'ls'， learning rate = 0.08	27.7242	17.0031	0.2635
ANN	ann_1(500), ann_2(50)	15.1319	10.5044	0.2029
LSTM	lstm(500), dence(50)	13.2541	**8.6655**	**0.1617**
GRU	gru (500), dence(50)	**13.0131**	8.7306	0.1851



(a) PM2.5 forecasting (b) Load forecasting

Fig. 5. Boxplot of comparison models' forecasting deviation. The flatter the box and whisker lines, the more centralized the data.

In order to compare the performance of comparison models more intuitively, the

prediction deviation for each model is calculated by subtracting the observed values from the predicted results. The boxplot of PM2.5 predicted deviations is shown in Fig. 5(a). It is obvious that GRU and LSTM have similar performance. Compared with ANN, SVR, DTR and GBR, their prediction deviation are smaller, indicating that the RNN-based methods have obvious advantages.

4.2 Load Forecasting

For load forecasting, we predict the load after 24 hours based on the current data. In our recent paper [10], the relevant experiments on this data set were conducted, including support vector regression (SVR), random forest (RF), gradient boosting decision trees (GB) and multiparametric regression (linear model, LM), and we took these results as benchmarks of our models. Furthermore, a non-machine learning baseline is applied in medel comparison: historical model (HM). In our case, the load time series can be assumed to be periodical with a daily period. Thus a historical model would be to always predict that the load 24 hours from now will be equal to the load right now.

Different from the previous study, the inputs of proposed models eliminates two unrelated variables as described in Section 2.2. Besides, inputs not only contains all the variables at the current moment, but also contains the historical data with time lags of 3, which makes our models obtain more input information. In this way, the memory ability of RNN is fully utilized to get better forecasting performance.

GRU and LSTM are trained by 50 epochs to ensure convergence, the batch size is set 50. Additionally, *Adam* was selected as the optimizer with the loss function of MAE.

Table 4 clearly compares the forecasting results of 7 models. The most striking result is that the forecasting results of GRU and LSTM outperform other models. As shown by the forecasting deviation boxplot of different forecasting models (see Fig. 5(b)), compared with other reference models, the prediction accuracy of GRU and LSTM are greatly improved, for the prediction deviations are closer to 0. This is mainly attributed to RNN's ability to process sequences and the inputs of historical data with specified time lags can provide more useful information.

Table 4. Errors analysis of load forecasting. Bold values indicate the smallest RMSE, MAE and SMAPE values.

Methods	Parameter setting	RMSE	MAE	SMAPE
SVR [10]	RBF kernel, C=32, γ=0.055	2441.1308	1531.4336	0.0327
RF [10]	mtry:4	2115.1549	1244.3625	0.0265
GB [10]	Interaction.depth=9, Shrinkage=0.1, n.minobsinnode=10	2142.7626	1290.7911	0.0278
LM [10]	/	4752.5047	3715.8092	0.0786
HM	/	6138.9287	3950.0712	0.0850
LSTM	lstm_1(300), lstm_2(300), dence(16)	535.1583	377.9184	0.0079
GRU	gru_1(300), gru_2(300), dence(16)	**517.9470**	**364.1986**	**0.0076**

In terms of the comparison analysis above, it is sufficiently demonstrate that RNN-based models show better forecasting performance than traditional machine learning. To be specific, GRU and LSTM have similar performance. In terms of pollution prediction, they are close to each other. While for load prediction, GRU is slightly better than LSTM.

5 Conclusion

In this paper, RNN application to forecasting are conducted. In view of two cases of air pollution forecasting and power load forecasting, forecasting models based on GRU and LSTM are proposed respectively. Compares with the existing research, the exprimental results show that the RNN-based model outperform traditional machine learning models on real datasets, which cover periodic/aperiodic and stationary/non-stationary data. It is to say that the application of RNN to forecasting tasks has great advantages.

The next work will be focused on model ensembling for forecasting, which combines deep learning with shallow learning by optimizing weights, such that the forecasting performance can be further improved.

Acknowledgments. This work was supported in part by the NSFC-RFBR Exchange Program under Grants 61911530132/19-58-53011, in part by the Fundamental Research Funds for the Central Universities of Central South University under Grant 2019zzts567, and in part by National Natural Science Foundation of Hunan Province of China under Grant 2018JJ2529.

References

1. Box, G.E.P., Jenkins, G.: Time Series Analysis, Forecasting and Control. Holden-Day, Amsterdam (1976)
2. Zhang X., Shen F., Zhao J., Yang G.: Time Series Forecasting Using GRU Neural Network with Multi-lag After Decomposition. In: Liu D., Xie S., Li Y., Zhao D., El-Alfy ES. (eds) Neural Information Processing. ICONIP 2017. Lecture Notes in Computer Science, vol 10638. Springer, Cham (2017)
3. Abuella, M., Chowdhury, B.: Solar Power Forecasting Using Support Vector Regression. In: Proceeding of the American Society for Engineering Management 2016 International Annual Conference (2016)
4. Chuentawat, R., Kan-ngan, Y.: The Comparison of PM2.5 forecasting methods in the form of multivariate and univariate time series based on Support Vector Machine and Genetic Algorithm. 2018 15th International Conference on Electrical Engineering/Electronics, Comput-er, Telecommunications and Information Technology (ECTI-CON), Chiang Rai, Thailand, pp. 572-575 (2018)
5. Zhang, G.Q., Patuwo, B.E., Y. Hu, M.: Forecasting with articial neural networks: The state of the art. International journal of forecasting, 14(1):35-62 (1998)

6. Zhang, X., Wang, R., Zhang, T., Zha, Y.: Short-term load forecasting based on an improved deep belief network. 2016 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE), Chengdu, pp. 339-342 (2016)
7. Petneházi, G.: Recurrent Neural Networks for Time Series Forecasting. *arXiv preprin*t arXiv: 1901.00069 (2019)
8. Liang, X., Zou, T., Guo, B., Li, S., Zhang, H., Zhang, S., Huang, H., Chen, S. X.: Assessing Beijing's PM2.5 pollution: severity, weather impact, APEC and winter heating. Proceedings of the Royal Society A, 471, 20150257 (2015)
9. Greene: Econometric Analysis, http://en.wikipedia.org/wiki/Granger_causality
10. Sidorov, D., Tao, Q., Muftahov, I., Zhukov, A., Karamov, D., Dreglea, A., Liu, F.: Energy balancing using charge/discharge storages control and load forecasts in a renewable-energy-based grids. *arXiv preprint* arXiv:1906.02959 (2019)
11. Hopfield, J. J.: Neurons with graded re-sponse have collective computational properties like those of two-state neurons. Proceedings 0f the national academy of sciences, vol. 81, no. 10, pp. 3088-3092 (1984)
12. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation, vol. 9, no. 8, pp. 1735-1780 (1997)
13. Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representationsusing RNN encoder-decoder for sta-tistical machine translation. *arXiv preprint* arXiv:1406.1078 (2014)
14. Chung, J.Y., Gulcehre, C., Cho, K.H., Bengio, Y.: Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv preprint* arXiv: 1412.3555 (2014)
15. Tao, Q., Liu, F., Li, Y., Sidorov, D.: Air Pollution Forecasting using a Deep Learning Model based on 1D Convnets and Bidirectional GRU. In: IEEE Access. doi: 10.1109/ACCESS. 2019.2921578 (2019)