Time series prediction using recurrent model

Rubyat Ul Hasnat

Dept of Applied Mathematics for Networking and Data Science
Hochschule Mittweida University of Applied Sciences - Mittweida, Germany

1 Introduction

Predicting future events based on historical activities is the basic idea of time series data. The basic assumption is that the events that have impacted the past will continue to impact in future and based on this assumption we try to forecast the future [1]. The motivation for prediction is reducing future uncertainty, especially due to the volatility of some phenomena. Some examples from a range of industries to make the notions of time series analysis and forecasting more concrete i.e. forecasting the closing price of a stock each day, the average price of gasoline in a city each day, the birth rate at all hospitals in a city each year, whether an EEG trace in seconds indicates a patient is having a seizure or not, weather report, future market movement etc.

With the improvement of database system and cloud computing we are getting huge amount of data. Data can be represented in different formats, from the most basic, such as numeric and nominal, to more complex ones, such as audio and video. Selecting a suitable machine learning model for a specific problem can be a difficult task. There is no single network architecture or learning method which can be adequately used for all domains because each has its own pros and cons. Studies have been done over decades to improve model from feedforward network to recurrent neural network (RNN) and many techniques have been implemented in RNN to improve the functionality for working with time series data [2] [3] [4]. The commonly used feed-forward neural network learning through back propagation performed well but can be improved by using recurrent model [5]. The learning algorithm of RNN allows the network not only remembering the pattern in a static group of units, the networks incorporate the data that must be remembered into their functioning in such a way that there is no static pattern to represent it [6]. Some studies have been done in RNN to minimize the training duration such as using wavelet transformation and also keeping the performance better [7].

With conventional Back-propagation Through Time or RTRL (Real Time Recurrent Learning) tend to either blow up (exploding gradient) or vanish (vanishing gradient) depends on the size of the weights. Long Short- Term Memory (LSTM) and Gated Recurrent Unit (GRU) are specially designed with gated architecture that can deal with the issue [8]. LSTM recurrent networks have been implemented to stock data and the result shows that the model can forecast effectively and still many improvements like extracting more feature value can be done to make the accuracy very high [9]. Comparison has been done through experiments such as detecting anomaly in time series data between RNN and LSTM and the result shows that LSTM perform promisingly in both small term as well as long term temporal dependencies [10]. Existing linear model like ARMA, ARIMA cannot describe the stochastic or nonlinear data like traffic flow where LSTM GRU perform significantly[11].

The rest organisation of this paper as follows we will discuss about fundamental of time series data, Structure of simple RNN and comparison with classical feedforward network, some other improved RNN like BRNN (Bidirectional Recurrent Neural Network), gradient vanishing and gradient exploding problem faced by RNN can be solved with LSTM, architecture of LSTM and some applications and comparisons, finally the conclusion with mentioning the scope for time series prediction using recurrent model.

2 Fundamentals of Time Series Data

The observations or the data that have been observed from a successive time period where the time period is at uniform time interval are called time series data. A time series data X of size k can be written as an ordered sequence of observation over the time t, i.e., $X = (z_1, z_2, ..., z_k)$ where $z_t \in \Re$.

The time series values which can be formulated by a mathematical function $y = f(time)$ is called deterministic. When the time series value combined up with a random term ϵ, $y = f(time, \epsilon)$, the series is called stochastic or non-deterministic.The another type of time series data is stationary time series data where the statistical properties such as mean, variance, correlation etc. remain constant over the time.

Time series data have the major three components **Trend**, **Seasonality** and **Residue**. Trend is a long-term increase or decrease in the data. For example increasing trend can be found in the real time data such as gradual change of consumption habit and demands for technologies etc. In the same manner decreasing trend can be found in mortality rate and unemployment etc. Seasonality is the event cycle that the variations repeat with a constant interval. For example the sale of air conditions increase during the summer and warm clothing in winter. On the other hand Residue is the non predictable short term fluctuations from the normal. Such as natural disasters, terrorist attacks and strike causes such instability. [12]

3 Recurrent Neural Network for Time Series Prediction

Developing a learning algorithm that can utilize the full computational power of neural network is a major concern. Much progress have been done with traditional feedforward neural network. But some problems that feedforward neural network can not deal with such as attractor dynamics and the ability to store information for later use. To deal with time variant input researchers have focused on nontrivial ways. To deal with this issue a general framework (Back

Propagation Through Time) was laid out by Rumelhart, Hinton, and Williams (1986) that unfolds the recurrent network into a multilayer feedforward network that grows by one layer by each time step. The weakness of this approach is the growing memory requirement when the given training sequence is arbitrarily long. Learning algorithm such as The Basic Algorithm, Real-Time Recurrent Learning and Teacher-Forced Real-Time Recurrent Learning are proposed which shows that these algorithms can train completely recurrent, continually updated networks to learn temporal tasks[6].

3.1 Architecture of RNN

We know the structure of a multiple layer network MLP with three layers namely input layer, hidden layer and output layer.

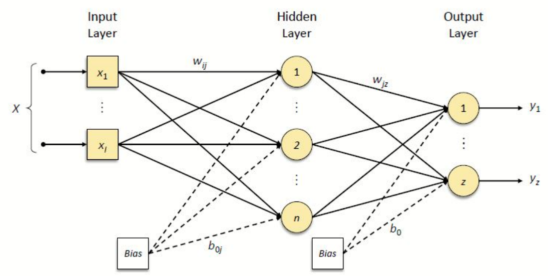

Fig. 1: Structure of MLP with single hidden layer [12]

The output of the model in Fig. 1 considering a single neuron in the output layer can be expressed as follow:

$$y = f(\sum_{i=1}^{n} w_j f(\sum_{i=1}^{l} w_{ij} x_i + b_{0j}) + b_0)$$

We can categorise the Perceptron and MLP models as feed-forward neural networks because it flows only one direction: from input to output. But Structure of a RNN can move in different directions.

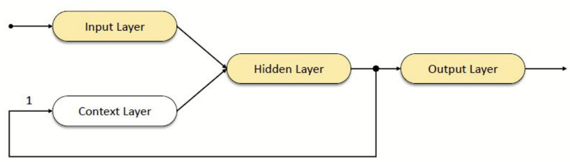

Fig. 2: Structure of RNN, Recurrent between hidden and context layer [12]

The fig. 2 illustrates the simple recurrent neural network, the hidden layer state at a given time is conditioned on it's previous state by a context layer. This recursion is known as short term memory that allows network to store complex signals for arbitrarily long time periods [12].

4 Comparison between RNN and Classical Feed-forward Neural Network in Time Series Prediction

To compare feed-forward and recurrent neural network we observe the performance while both model used Mackey-Glass nonlinear chaotic time series. Which is known as benchmark test whose elements are hard to predict. Results showed RNN expectedly achieved better result in all tests [5].

Time series data such as posses moving average components, state dependent, or have trends can be resulted better by using recurrent neural network over feed-forward network. Experiments have been done on load forecasting data and result shows that recurrent neural network provides significantly better model and corresponding prediction, than a feed forward network[2].

We have observed the performed experiments using three time series such as the logistic map, sunspot data and the laser time series. Traditional feed-forward neural networks have disadvantages while predicting multi step in the time series data from the current state. While MSRN which based on imposing a special learning phase for the purpose of long-term prediction using the predicted outputs as input variables and the MSRN proposed in this paper improved the performance in multi-step prediction[13].

5 Techniques Implemented on RNN to Enhance Performance

5.1 [14]Technique 1:

The Real Time Recurrent Learning based on gradient descent method is similar to the back-propagation algorithm. And it has the same major draw back that the learning is slow. This slowness due to lack of knowledge of right step size to be taken to the global minima and if the step size is chosen to be large then there is fear of divergence.Another drawback is in the prediction context. If the adjacent samples of the time series have a highly non-linear relationship, the recurrent neural network cannot perform the best. The technique for this scheme is we first train the network with RTRL. This training is done over a portion of the time series and then this trained network is used to predict the next sample and in this manner we continue the training till the training is exhausted enough. This method is interpreted as one epoch and after one epoch weight are made. This procedure is continued iteratively.The training is performed by “teacher-force”. After train the employing “teacher- force” here they adopt the cascade correlation algorithm and the prediction performance improved considerably.

5.2 [7]Technique 2:

From the past experience we come to an observation that recurrent neural network RNN for modeling several minutes of data would take more than hour to train completely. A discrete wavelet transform (DWT) is applied to reduce the time for training. The target of DWT is to approximate as closely as possible any function in homgeneoussobolev space, by summing interpolation function at several scale levels.By sub sampling the raw data and interpolating with the scaled cubic spline wavelet basis, we obtain the wavelet coefficients at each levels.

The method of using the wavelet coefficients instead of raw data in the RNN the computation time in trend prediction time is reduced from hours to minutes and the prediction accuracy remains satisfied.

5.3 [4] Technique 3:

The RNN network differs from MLP in that manner that RNN architecture can make use of all the available input information up to the current time $t_c(i.e., X_t, t = 1, 2, ..., t_c)$ to predict y_{t_c}. Future information after t_c is useful for prediction. RNN can predict y_c for input information up to X_{t_c+M} where M is the certain number of time frames. When we try to capture available all future information by increasing the value of M and when the M is too large the RNN prediction results drop. To overcome this issue Bidirectional Recurrent Neural Network BRNN can be trained using all available input information in the past and future of a specific time frame. The structure of the BRNN is a splitted RNN where one part is responsible for the positive time direction (forward states) and another part is responsible for the negative time direction (backward steps). Output from forward states are not connected with the input of backward states and vice versa. SO another way to imagine the BRNN is without the backward state it is a simple forward RNN. The training of BRNN is completed by three main steps i.e, Forward pass, Backward pass and Update weights.

6 Long Short-Term Memory

In the study of RNN we have observed that RNN can keep of arbitrary long term dependencies in the input sequences. The problem occurred when we train the network through back propagation. The gradient value which is back-propagated can be either vanished or exploded. The gradient based method called LSTM can learn to bridge minimal time lags in excess of 1000 discrete time steps by enforcing constant error flow through "constant error carrousels" with in special units. LSTM enables to learn also solving complex, artificial long time lag tasks that have never been solved by previous RNN algorithm [8].

6.1 Architecture of LSTM [11]

The typical LSTM network architecture is in Fig.3 mainly has 4 gates. Those gates are input gate, input modulation gate, forget gate and output gate. Input gate takes a new input point and from outside and process newly coming data. Memory cell input gate takes input from output and of the LSTM cell in the last iteration. Forget gate learn when to forget the result and thus selects the optimal time lag for the input sequence.

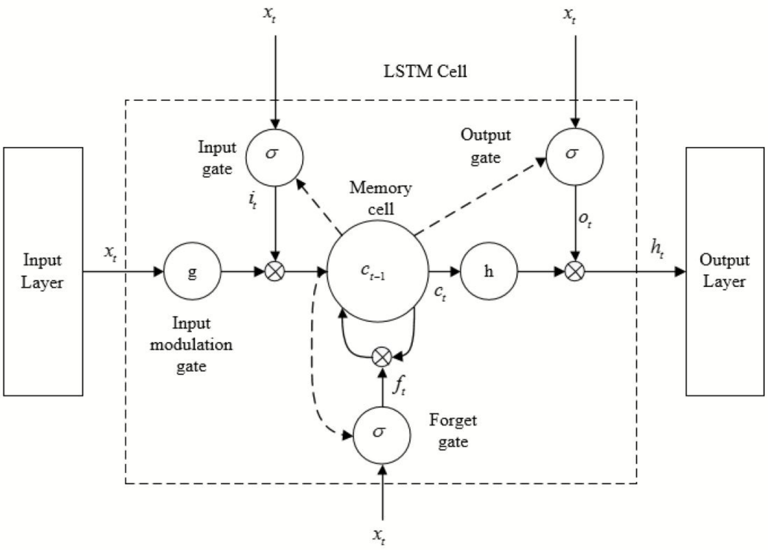

Fig. 3: Structure of LSTM neural network cell [11]

Let us consider time input series as $X = (x_1, x_2, ..., x_n)$, hidden state of memory cell as $H = (h_1, h_2, ..., h_n)$, output time series as $Y = (y_1, y_2, ..., y_n)$

LSTM computation as follows

$$h_t = H(W_{hx}x_t + W_{hh}h_{t-1} + b_n)$$

$$p_t = W_{hy}y_{t-1} + b_y$$

The hidden state of the memory cell computation formula are following

$$i_t = \sigma(W_{ix}x_t + W_{hh}h_{t-1} + W_{ic}c_{t-1} + b_i)$$

$$f_t = \sigma(W_{fx}x_t + W_{hh}h_{t-1} + W_{fc}c_{t-1} + b_f)$$

$$c_t = f_t * c_{t-1} + i_t * g(W_{cx}x_t + W_{hh}h_{t-1} + W_{cc}c_{t-1} + b_c)$$

$$O_t = \sigma(W_{ox}x_t + W_{hh}h_{t-1} + W_{0c}c_{t-1} + b_0)$$

$$h_t = o_t * h(c_t)$$

Where σ stands for the standard sigmoid function [11].

7 Time Series Prediction Using Long Short- Term Memory Network

Big data has made the development of prediction algorithm more intelligent. That causes gradually improvement in algorithm designing from traditional linear prediction to more deep learning prediction algorithm. Stock exchange data is one of the most complex data where we can check that how LSTM perform. Experiments showed that the LSTM algorithm perform better in prediction and has smaller errors. Though the advantage of the LSTM is that it requires longer training period and it requires a large number of sample data set [15].

The efficacy of LSTM is demonstrated on four data sets: ECG, space shuttle, power demand and multi-sensor engine data set. We observed anomaly detection by using LSTM on these four data sets. Results shows that (i) stacked LSTM networks are able to learn higher level temporal patterns without prior knowledge, (ii) in normal time series behaviour it can be used for anomally detection. Also we can compare the LSTM-AD with RNN-AD and the result shows LSTM-AD perform better or similar. Though the LSTM-AD is more roboust then RNN-AD[10].

In Intelligent Transportation System traffic flow prediction is important. We apply our Long Short- Term Memory neural network (LSTM, Gated Recurrent Unit (GRU) and Auto Regressive Integrated Moving Average ARIMA model for the prediction. Both LSTM and GRU perform better than ARIMA. Even though GRU in this data set performed a little better than LSTM and also converged faster[11].

8 Conclusion

In this paper we have discussed about time series data and the types of time series data. We have discussed about the very classic feed-forward neural network and from this we understand the necessity of recurrent neural network and examine the performance of RNN and many studies that have been done for the development of efficiency of RNN. That lead us to the state of the art LSTM. Researchers are still working for the improvement of LSTM prediction accuracy.

References

[1] M. Kulahci D. C. Montgomery, C. L. Jennings. Introduction to time series analysis and forecasting. *Wiley Series in Probability and Statistics, Wiley, New Jersey, USA, 2 edn.*, 2015.

[2] Jerome Connor, Les E Atlas, and Douglas R Martin. Recurrent networks and narma modeling. In *Advances in Neural Information Processing Systems*, pages 301–308, 1992.

[3] Sui-Lau Ho, Min Xie, and Thong Ngee Goh. A comparative study of neural network and box-jenkins arima modeling in time series prediction. *Computers & Industrial Engineering*, 42(2-4):371–375, 2002.

[4] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[5] Danko Brezak, Tomislav Bacek, Dubravko Majetic, Josip Kasac, and Branko Novakovic. A comparison of feed-forward and recurrent neural networks in time series forecasting. In *2012 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr)*, pages 1–6. IEEE, 2012.

[6] Ronald J. Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280, 1989.

[7] Fu-Chiang Tsui, Mingui Sun, Ching-Chung Li, and R. J. Sclabassi. Recurrent neural networks and discrete wavelet transform for time series modeling and prediction. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 3359–3362 vol.5, May 1995.

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[9] Siyuan Liu; Guangzhong Liao; Yifan Ding. Stock transaction prediction modeling and analysis based on lstm. *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2018.

[10] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, and Puneet Agarwal. Long short term memory networks for anomaly detection in time series. In *Proceedings*, page 89. Presses universitaires de Louvain, 2015.

[11] Rui Fu, Zuo Zhang, and Li Li. Using lstm and gru neural network methods for traffic flow prediction. In *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 324–328. IEEE, 2016.

[12] Antonio Rafael Sabino Parmezan, Vinicius MA Souza, and Gustavo EAPA Batista. Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model. *Information Sciences*, 484:302–337, 2019.

[13] Inés M Galván and Pedro Isasi. Multi-step learning rule for recurrent neural models: an application to time series forecasting. *Neural processing letters*, 13(2):115–133, 2001.

[14] S. S. Rao and V. Ramamurti. A hybrid technique to enhance the performance of recurrent neural networks for time series prediction. In *IEEE International Conference on Neural Networks*, pages 52–57 vol.1, March 1993.

[15] Fei Qian; Xianfu Chen. Stock prediction based on lstm under different stability. *2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, 2019.