

Analysis of Time Series Prediction using Recurrent Neural Networks

Gaurav Yadav
Department of Computer Science
Chandigarh University
Mohali, Punjab, India

Richa Vasuja
Assistant Professor
Department of Computer Science
Chandigarh University
Mohali, Punjab, India

ABSTRACT

Time series prediction is the heart of forecasting data that is based on past information of any particular dataset, recurrent neural network combines with the time series algorithm and provide much reliable outcomes having high matching efficiency with actual real-time results as the combination of RNN and time series can predict almost anything which has time as its managing factor as RNN has ability to iterate efficiency with time feeds, it tells about the trend of future and it is particularly important in prediction of crucial data such as weather forecast or financial data, because proper forecast can provide the vital help and safety or the advancement for the change, though the future is uncertain but people must know their future as near it could be to the future. Based on the research this paper contains analytical data of recurrent neural network and its use with time series alongside the experimental data analysis of weather forecast and financial forecast data.

General Terms

Artificial Neural Networks, Model Training, Back-Propagation, Forecasting.

Keywords

Time series, forecasting, moving averages, RNN, GRU, long short term memory, Auto regressive Model.

1. INTRODUCTION

Forecasting is the approach for predicting the estimated future trend of any particular dataset such as forex market data or financial data by historical inputs which after going through scientific iterations of computerized simulation with neural networks, recurrent neural networks architecture in this particular analysis provides the efficient and reliable outcomes for the future. Forecasting just predict the data for future by analyzing the outcomes from the past and time series algorithm makes it easy as it provides computation standard for forecasting but use of neural networks alongside time series sequence is the huge efficient leap for providing the correlated result for the future with high efficiency and minimum considerable analytical time.

Data collected from sequential observations over time is extremely common with recorded changes and anomalies' affecting the motion set of ongoing data and makes future predictions much common to changes relating to the past changes and helps to figure out the mood of the traders in financial market in particular or for another forecasting applications such as rainfall forecast or weather forecast past data is recorded and feed through the neural network running with the time series algorithm which helps to predict the future values for the predictive iterations, time series study has virtually endless applications. In meteorological application, the daily high and low temperature with averages

changes to the current values alongside the past data for particular, annual rainfall data, drought indices and wind speed and direction averages and other crucial data is observed and recorded to make the efficient forecast for oncoming days and changes, in financial market, daily open and close positions, daily averages change to the spread and liquidity of market and yearly changes affected by the traders mood or market changes, biology also the application for time series prediction, the data of milliseconds electrical activity of heart is analyzed for keeping the track of patient, so these are some noticeable application of time series sequence though it has virtually endless applications with improved computational ways such as introducing the concepts of neural network alongside the concept of machine learning and deep-learning. Recurrent Neural Network (RNN) is the architecture of Artificial Neural Network (ANN) in which connections are found between nodes form a direct graph along the spread sequence of the values. The direct graph to the sequence allows the recurrent neural network to have periodic dynamic behavior for the time sequence, which proves to be fundamental for time series sequence and with machine learning the result efficiency of RNN is improved and possess automaticity to the predictions. Recurrent neural networks were developed in 1980s but found the regular application after the advancement in the concepts of machine learning and deep learning for patten recognition and speech synthesizers but use of neural network for time series sequence prediction is new and for the future.

One important fundamental of neural networks – namely their ability to predict arbitrary non-linear functions, is of high value in time series processing and promises widespread applications, especially in the subfield of forecasting [1]. Times series analysis general purpose is twofold; to understand the stochastic mechanism that provides an observed series and to predict the future values of the series based on the outcomes of the recorded past values of the series or the values of related series which could affect the values of the ongoing series for the future predictions. A unbelievable feature of models of time series is that we usually cannot determine the basis of predictions is independent of widespread common observed population of datasets.

2. ANALYTICAL METHOD

Time series analysis mainly has two methods for determining the result of series prediction which are; frequency-domain method and time-domain method. Frequency domain method uses spectral analysis and wavelet analysis while auto-correlation and cross-correlation is used in time domain method of time series analysis by using filtered scaled correlation.

2.1 Frequency domain method

Frequency domain analysis methods allude to the examination of scientific capacities or signs concerning recurrence, as opposed to time. Put basically, a period area diagram demonstrates how a value changes after some time, while a frequency domain graph indicates the amount of the values exists in each given recurrence band over a scope of frequencies. A frequency domain portrayal can likewise incorporate data on the stage move that must be connected to every sinusoid with a specific end goal to have the capacity to recombine the frequent segments to recoup the first run through values.

2.2 Time domain method

Time domain method determine the mathematical approach of any gradient functions relating with physical changing factors or values and times series of financial or climate dataset against time, the plot data of time domain graph can be transformed into frequency domain method using Fourier transformation and vice-versa.

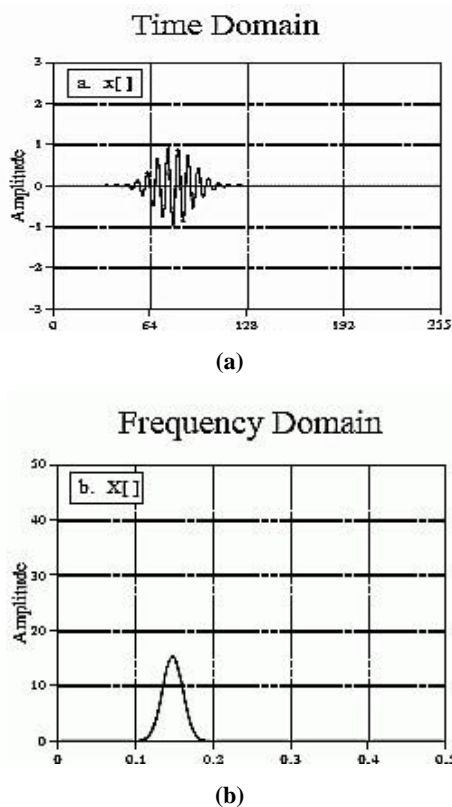


Fig 1: Time domain graphs (a), Frequency domain (b). [2]

Fourier transformation ensures the reliable graph conversion of time-space method (a) to frequency domain method (b) having the highest intervals and frequency to sample values respectively.

Later on time series have other methods for analysis which are parametric and non-parametric analytical methods to its dispense; parametric methods have stochastic values with stationary points such as regressions and Moving Averages(MA),weighed moving averages(WMA) or exponentially moving averages(EA). The goal of parametric method is to analyze the stochastic series. Linear and non-linear and single variant and multivariate determination is also applied in classification of the data of times series inputs for future feeding to neural network's environment.

3. TIME SERIES MODELLING

In dynamic programming the modeling is the main engine for determining the spread of result datasets as modeling describes about the principal classification of the data to help the future classification of data for other functions and feeds for error improvement alongside increasing the learning rate capacity of recurrent neural network. Time series can be classified mainly into two fundamental models, Deterministic model and Stochastic model

3.1 Deterministic Model

In this model the values of variables; univariate or multivariate, is uniquely determined by the set of previous value sets and by the parameters of the model, deterministic model can be viewed mathematically as,

$$\alpha(t) = \beta(t), \quad (1)$$

$$\alpha t = \beta(\alpha t-1, \alpha t-2, \dots, \alpha t-n) \quad (2)$$

Deterministic model's behavior is determined by the given set of series values sets. In environmental application of time series previous filtered data of geostatic pressure and temperature, movement average change over a particular period of time is collected and fed to the recurrent neural network simulations for training over the recognized datasets so that it could improve itself and can predict the efficient, reliable information for the future.

Time series modeling also summarizes the predicative result datasets for RNN iterations which uses LSTM and GRU-RNN for analysis the datasets to provides the near about to actual values of the previous continued multivariate dynamic pre-cycle of datasets and averages.

3.2 Stochastic Model

Probability distributions ensures the randomness of a widespread series of values and it is the main concept of this stochastic model as it is not determined by any particular unique univariate value but by the function of randomness. If we are having a starting value for any given multivariate dataset then there could be many iteration which could be much more efficient than any particular chosen iteration. Moving Averages (MA) of the forex currency pair is commonly predicted by analyzing the regressive approach of previous moving data and charts; Moving Average is the method of determining the time series predictions by value by value iterations transitions carried out with the calculation of particular values as to provide for statistical average.

The moving average model is like

$$\mu t = (\alpha + \beta t + \theta_1 \beta t + \dots + \theta_q \beta t - q) \quad (3)$$

α is the mean of the series, θ_1 to θ_q are the parameters of the model, $\beta t, \beta t-1, \dots, \beta t-q$ are the noise error terms in (3), then the value of q is said to be MA of the model. There are many more model available for financial data such as Autoregressive model (AR) and Auto regressive Moving Average model (ARM) to determine the future value set for the dynamic datasets of any series.

3.3 Multivariate Time series Model

The concept of multivariate Model revolves around the concept of Vectors and multi-dimensional Matrices, bidirectional in particular, in this model multivariate observations values are made up of another sets of observed values of another time series and denoted as $x = [x_1, x_2, \dots, x_n]^T$, And the whole set of multivariate values as a matrix

contains the individual observed values from another series to fit in x.

This model is used for providing the training advancement to the recurrent neural network associated with the calculation of time series differentiating from other conventional algorithm such as cluster algorithm, this model is used for future predictions or forecasting the data but not for any structural and policy analysis as all the values set of this models are above the restrictions which are required for structural analysis and multivariate times series values are endogenous.

Below is the representation of the multivariate matrix.

$$X = \begin{pmatrix} X_{11} & X_{12} & \dots & X_{1n} \\ X_{21} & X_{22} & \dots & X_{2n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ X_{n-1} & X_{n2} & \dots & X_{nm} \end{pmatrix}$$

3.4 The state-space Model

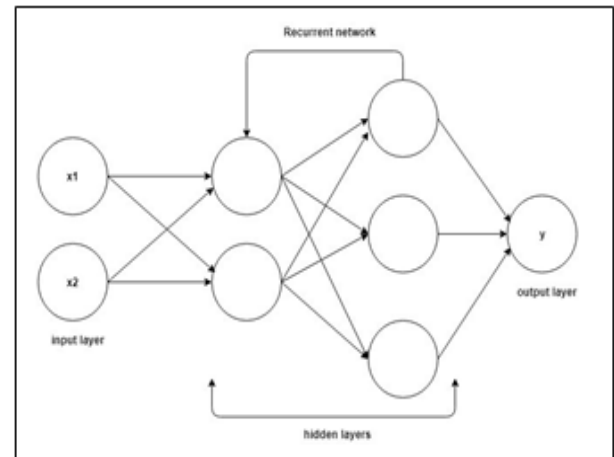
This model is determined by the smallest multivariate value of smallest subset of the system which could define the state of the entire system over a particular range of time. This model is used in particular if the key values of data which could affect the nodes and mapping efficiency or recurrent neural network system for the series trained over the past collection of various dataset values affecting the change factor of the time series sequence.

4. LONG-SHORT TERM MEMORY RECURRENT NEURAL NETWORK

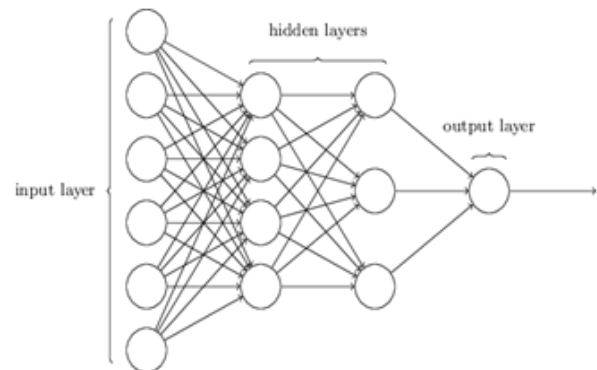
Whenever there is series of data and that temporal dynamics which relates the data is elegant than the spatial content of each individual time period [3], Recurrent Neural Network (RNN) is special architecture of Artificial Neural Network (ANN) because it is different in its behavior and working from other architectures of Neural Network as other neural network usually have feed forward flow system whereas coming with the contradictory behavior RNN possess cyclic flow among hidden layers of neural network. There are many networks available for Natural Language Processing (NLP) and Machine Learning (ML) but RNN deploys the strange ability of storing its inputs because it has its own memory, and because of their internal memory recurrent neural network becomes the most promising for providing sequential data and this makes recurrent neural networks best available for time series prediction as with proper analysis over time and with the concept of deep learning and machine learning for programming it becomes efficient and reliable for forecasting the data to make the human life as well as computation simple.

RNN is the outperforming algorithm behind the amazing achievements of deep learning in past few years, though there is huge remained to be found about the Artificial Super Intelligence (ASI) and Neural Networks but for the case like application of time series prediction it is based suited if one decides to implicate time series forecasting within the concepts of machine learning and automation. Recurrent neural networks are robust and powerful when it comes to handle the sequential variation of data because they can tell about what's coming next, because of their internal memory.

RNN has various types too but Long Short Term Memory (LSTM) is most efficient for time series prediction according to the analysis.



(a)



(b)

Fig 2: Recurrent neural network schematics (a) with back propagation, Feed Forward Neural Network (b).

Feed forward flow can be defined as the flow of iterations from input layers to output layer through the hidden layers as in Fig 2(b) until the programming introduce the back propagation for minimizing the gradient descent for the reduce error percentage and efficient learning rate cope up with training and flow among hidden layers of ANN. Neural networks has connections and each connections transferring the output from the hidden layers to the input from the input layer, transformation is done by a propagation function (5), and learning is done through the algorithms which sets the parameters for the neural networks handling flow and with multivariate grade descents or vanishing descent maximum efficiency can be achieved.

$$\alpha_i = \sum \beta_i(t) \mu_{ij} \quad (5)$$

Unlike neural network recurrent neural networks do not biased to feed forward flow of data RNN has unique recurrent network back propagating in Fig 2(a), they have tendency to store the temporal dynamic data to feedback or back propagate to another hidden layers or nodes to determine the forecasted value with maximum efficiency in result and vanishing grade descent for the network.

Recurrent Neural Network provides the predictive sequential data that other networks and algorithms cannot provide therefore they are used for providing the time series data with much efficient and automated way for the time series like

audio, video, speech, text, financial data, metrological data and many more considerable time series. The usual RNN has short term memory and it has cyclic flow of information through a loop to take the current decision in consideration along the current values and the whatever it has learned till that particular point and feeds back to the node for improving itself for future loop cycles and can remember each cycles decision and values to back propagate to itself which in case of feedforward or simple neural network is not possible because they cannot remember the values of every iterated cycle but recurrent neural networks remembers exactly that, therefore they are taken into consideration for this analysis of times series forecasting of sequential data because of its internal memory it remember outputs and take that output into consideration and loops it back to the network, Recurrent Neural Network adds the immediate past to the present and because of this phenomenon RNN has two inputs; the present cycle loop (n) data, (n-1) iteration data, and it can map many to many and many to one whereas feed forward neural networks has one to one mapping form, therefore RNN can determine what is coming next and due to the determination of immediate future it can do what others algorithms can't provide for the time series prediction analysis but with recurrent neural networks time series prediction becomes more automated and reliable then before as desired by the analysts.

4.1 Long Short Term Memory Network

Recurrent Neural Networks have internal memory but limited and for a long period of time within the cyclic iteration of the recurrent network the overflow and exiting problem arises to the iterations and to avoids that RNN needs to have considerable amount of operational memory to run throughout the complete iterations cycles to achieve to multilayered network of Long Short Term Memory Network (LSTM) is used as extension to recurrent neural network which in result provides the additional operational memory to RNN.

Long Short Term Memory Networks for time series prediction becomes much efficient and robust then single or multilayered RNN as alongside increasing the memory of RNN, LSTM enables RNN to remember their inputs over the long period of time and helps the network to learn in even long time lags between the decisions of each iterations for learning. LSTM holds the information in the memory so the recurrent network can read the information whenever any iteration changed anomaly is required over the complete cycled iterations till the compilations of the results datasets, memory of LSTM network having LSTM units can be referenced as a gated memory as it can read, write and even delete the information and with trained modules for programming this holds the ability of automation and efficient series data prediction without any un-normalized data or error in results dataset. LSTM can be trained to learn with vanishing grade descents with learning algorithm corresponding to the previous observed data of particular time series.

Long short term memory network consists of LSTM cells and the memory of these LSTM cells can be determined as the gated memory, which means LSTM unit cells gets to decides which information is relevant to the system and which sets of information is irrelevant to the system and which sets of information is to be ignored or removed. The behavior of gated memory cell is decided by the learning algorithms with analyzed weighs during model training, therefore LSTM recurrent neural network learn over iterations to decide itself what is important and what is to be skipped. LSTM networks memory cell have three usual considerations for the gated

memory cells; input,output, delete, which holds the grammatical behavior for iterated cycled values.

$$\alpha(q) = \mu q \div (\mu q + 1) \tag{6}$$

The gates in LSTM is in form of sigmoid function (6) ranging from 0 to 1, sigmoid function defines the nonlinearity in the neural network to enhance the learning capacity for the RNN model for time series prediction as the derivatives of sigmoid proves to be enhancement to the learning iterations.

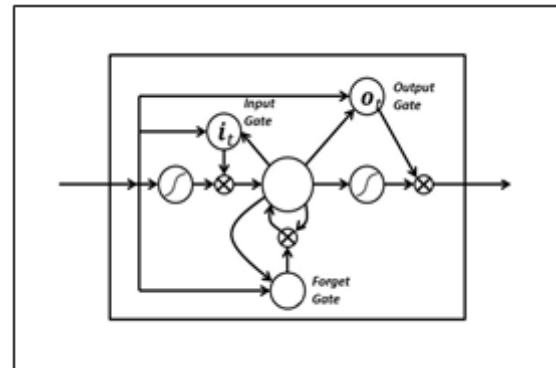


Fig 3: The gated memory of Long Short Term RNN[10]

The gates of LSTM cell are analog which determines the back propagation with it over the learning simulation periods or training period. The vanishing gradient descent which occurs when the value of gradient is too small and the learning time is too much or the model is just unable to learn which is usually arises when working with simple recurrent neural network but with LSTM gates memory recurrent network the problem of vanishing gradient is resolved as LSTM-RNN model keeps the gradient values steep enough so that the model can have minimum training period and high learning rate enhancing the prediction of predictive values over time.

5. GATED RECURRENT UNIT FOR TIME SERIES

Gated Recurrent Unit (GRU) is the simplified version of the LSTM Recurrent Neural Network which is preferred for this analysis of time series predictions because previous data collection could have missing values or values which were ignored by the iterations of normal RNN or LSTM recurrent neural network and with the missing values from the observed datasets of particular time series predictive value sets cannot much reliable as it can predictive values which could not be as near to the actual values if checked therefore, GRU recurrent neural network is preferred for this paper's analysis.

GRU takes in particular masking and time intervals for analyzing the missing or bad code values from the model and use these representations as to determine the missing values from the collected datasets over time and it efficiently engraves them into the complex model schematics so that GRU not only records from the widespread temporal relations of time series but also takes the missing values or strange analogies from the observeddataset of series to predict the near to perfect predictive result values for the time series forecast, this is the stronghold of GRU-RNN over the LSTM networks. In practical, this model suits best for time series prediction analysis because strange things can be seen in real-time which just happens and needs to be considered so that no error space could be there in future datasets to maintain the reliability of neural network and concept of automation and Machine Learning. GRU-RNN could be implemented in R, MATLAB or Python environment but for this analysis Python

environment is preferred as it has all the working modules at dispose and ease, keras [4] is used for the analysis for time series data of different datasets sequences forex market and temperature times series data in particular is taken into consideration to predict the forecast for particulars by using the recurrent neural network of type LSTM and GRU-RNN.

The main structural advancement of GRU is the introduction of reset and update gates to the LSTM recurrent neural network structure which has input, forget, output gates.

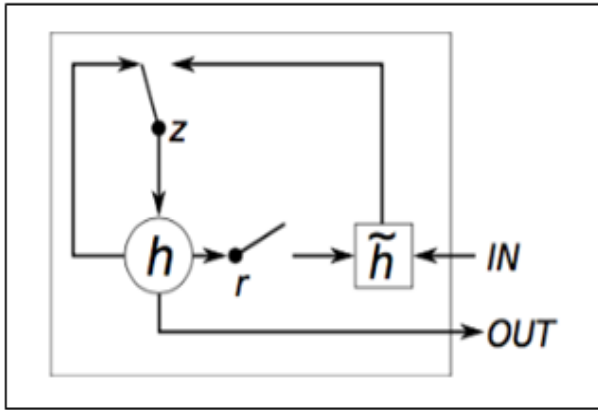


Fig 4: Gated Recurrent Unit with reset gate r and update gate [10]

The LSTM structural networks is followed for the generation of GRU so it could be considered that GRU will have all the capabilities of LSTM-RNN and in Fig (4), reset gate (r) and update gate (z) an advancement to the schematic of LSTM network and both having same computational purpose of analyzing the widespread relations of the observed data for the time series.

Functionality of Gated Recurrent Unit can be described through the reset gate as it is presented between the previous iteration cycle and the next iteration cycle to erase the previous iteration cycle decision, and then update gates gets to decide the amount of present iterations values to update the cell state. GRU exposed the whole cell unit state to the other units of the network and performs input and forget action through its reset gate only, which in practice makes it efficient for back propagation and steep gradient descents or vanishing gradient avoidance for better learning rate and efficiency over time.

In terms of training, GRU recurrent neural networks are used in this analysis as it is faster than LSTM-RNN because it has fewer parameters. Mathematically GRU can be determined as below

$$z_t = \alpha(w_z [h_{t-1}, x_t]) \quad (7)$$

z_t is a update gate, h_t is hidden state in (7)

$$r_t = \alpha(w_r [h_{t-1}, x_t]) \quad (8)$$

r_t reset gate and w is the current iteration's value 0 or in in (8)

$$\hat{h}_t = \tanh(w [r_t * h_{t-1}, x_t]) \quad (9)$$

\hat{h}_t candidate state x is the state of the gate cell (9)

$$h_t = (1 - z_t) * h_{t-1} + z_t * \hat{h}_t \quad (10)$$

h_t hidden state in (10)

Time series preferable application with RNN holds accountability over both LSTM as well as GRU-RNN but for this analysis GRU is taken into consideration as previous

paragraph states it is easy for training with less error correlation and can provides values even for missing or bad code data from the past collected datasets of a particular time series over a period of certain time.

6. ANALYTICAL RESULT

In this analysis the data was collected from sources and the environment for applying the Recurrent Neural Network; Long Short Term Memory as well as Gated Recurrent Unit structure of RNN was setup for analysis of the collected data from the referenced sources to predict the results for forecasting the possible or most matching probable values for actual values of time series of data.

6.1 Environment for analysis

The application of RNN needed the predefined modules which is used for training and simulating the epoch for prediction of time series near future dataset, machine learning's TensorFlow [5] and Keras with the coding environment of Python 3.6 language is used in this analysis and for hardware; Nvidia 940MX with 2 GB of dedicated memory and alongside machine learning framework running on window 10 OS and Intel i5 processor for this particular analysis is used though altered environment can also be used for application of RNN for Times series Prediction.

6.2 Data Rendering and pre-processing

Data was provides by the third party referenced source with particular parameters for forex currency pairs and famous cryptocurrency Bitcoin to USD moving price values for a time period of year 2017 to predict the price for September 2017 using RNN with time series and Bitcoin (BTC) and USD data is used in this analysis but use of this analysis could be performed over any forex currency pairs , commodity indexes , indices, stock market or for metrological index such as temperature which is taken into consideration of undefined location also analyzed in this analysis, aside quantitative data quality- based datasets are also collected from Google trends [6] for particular keyword in this analysis this keyword trend's data is collected for 'Bitcoin' to analyzed the mood of traders and areas affecting the market for Bitcoin and USD pair.

Data pre-processing was done over collected data of Bitcoin to USD exchange rate which includes formatting, completing missing values, cleaning and arranging the dataset in time dimension for BTC/ USD time series dataset, for Bitcoin some parameters of datasets collected are as; 4 features corresponding to years, the month and days of the week and month's day of the time series. Yahoo Finance [7], provided daily data with open and close, shifted close, high and low prices for 138 features and volumes of liquidity of 20 of the main stock market index is also collected and formatted for the analysis, 93 features of daily data formatted by opening, closing, high and low prices, volume of transitions and market capitalization of Bitcoin is rendered and the data was collected from Bitcoin Chart Application Programming Interface [8].

6.3 Long Short Term RNN approach for time series analysis

LSTM Recurrent Neural Network feeded with the data for the time series function after proper formatting through Keras library functions, the formatted data to LSTM-RNN consists of 1500 observations of daily data from May 2013 to September 2017 to predict the upcoming 30 days data through graphical representation and mimicking the actual dataset

graph to predicted graph in results for better representation of analysis and compiled result.

F is the defined feature space and R is the drop-out rate for RNN, integer values N, hidden layers is denoted by M and also for the number of neurons for each hidden layers. For better optimization for the analysis LSTM is aided with Nestov Adam optimizer (nadam), and the learning rate is set to default for Keras which is 0.002. With all these defined parameters LSTM-RNN have, 1 layer with linear activation function and an output layer of F unit; recurrent layers N with M neurons with drop-out rate of R and with hyperbolic activation function, with 1 output layer having 1 dense layer with liner activation function, are used in application of LSTM type of recurrent neural network. For Bitcoin to USD formatted data was imported to the keras and level_lstm was introduced to use the data for running LSTM to predict time series which provides the predicted price chart of Bitcoin to USD in comparison for real time actual price chart for particular which is shown in below output figures for LSTM.



Fig 5: Predicated chart to real chart using LSTM

This is the most mimicable efficient output for the time series for Bitcoin index using LSTM-RNN, this result was compiled with 1 recurrent layer with 100 neurons, and with the drop-out rate of 0.3 by using 9 features space in consideration, the output chart is prepared in Tensor board of TensorFlow which had Root Mean Squared Error (RMSE) of 28.45 and for next day RMSE analyzed to be 519.626 and for next upcoming week or 7 days from 1, September 2017 value for RMSE is noted to be 399.406. The result in Fig (5) is obtained with 100 batch size and same number of neurons with nadam optimizer [5] which had learning rate of 0.002 and the model is trained for analytical functionality with training period of 0.01:23.52596, these were the analytical parameters of output predicated value chart showed in Fig (5) corresponding to real value chart till the end of September 2017.

6.4 Gated Recurrent Unit RNN approach for time series

To compare the efficiency and reliability of RNN same data with modified features suited for GRU-RNN application is analyzed with GRU structure of RNN by using Keras sequential model with TensorFlow features representation by using the layer_gru method from Keras module in same environment and features space as of LSTM prediction.

The GRU network implementation with best efficiency and with best mimic rate to real values found to have 1 recurrent layer with 500 neurons and drop rate of 0.1 with 10 features in consideration and provides the RMSE of 273.92 and RMSE of next day and next week found to be 396.408 and 328.934 respectively.

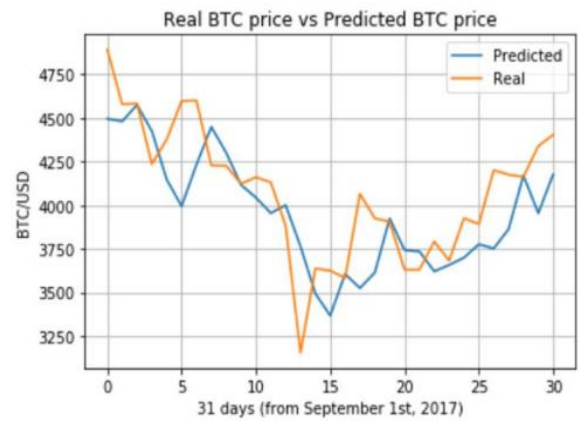


Fig 6: Predicated chart to real chart using GRU-RNN

The GRU approach also had nadam optimizer with 100 epoch or iterations the same learning rate of 0.002 and but found to have less training time from LSTM approach, which is 42 second to 61 second (LSTM training time).

GRU holds advantages over LSTM in training period time for model but both RNN models had not shown average accuracy for time series data chart prediction but the difference can be considered for both models if RMSE for next day (1 day) is considered for GRU which is 396.408 and RMSE of next day for LSTM is 519.626, therefore if both model is compared for making the most accurate prediction GRU holds advantages over LSTM but in case of weekly RMSE both model does not have much difference in their accuracies or RMSE values; which is shown in below comparison chart between LSTM and GRU application for analysis of dataset in Fig(7).

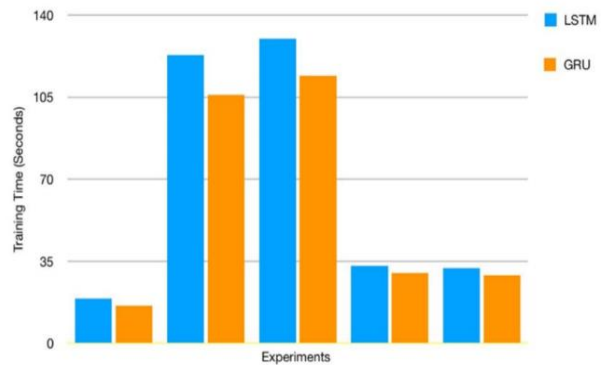


Fig 7: Training time comparison for LSTM and GRU

This analysis only considers quantitative parameters which do not includes sentiment analysis or say changes in markets due to news or climate conditions or any qualitative featured variable based changes as this analysis does not go deep into concepts of deep learning and artificial intelligence but to improve the analysis result and to ensure the better efficiency and reliability for forecasting the time series for Bitcoin index further Keras model methods is analyzed which do not used RNN for time series prediction.

Auto Regressive Integrated Moving Average (ARIMA) [9] with dynamic regression layers in Keras is also applied for concluding the comparison of RNN to ARIMA. ARIMA was feeded with same parameters and feature spaces as LSTM and GRU for same collected rendered data to predict the values for Bitcoin which found to have RMSE of 255.90 and had training time of 0.6169 sec. which is less than both the models but ARIMA doesn't holds accountability as its too volatile

and directional, nonetheless Artificial Neural Networks (ANN) are better in performance and reliability, though it can be a topic of argument over prediction of time series accuracy for RNN and ARIMA application but this analysis is done over a recorded past dataset with RNN and it has better performance over ARIMA and possess much mimic rate and reliability.

7. CONCLUSION

This paper shows the application of Recurrent Neural Network for prediction of Time series datasets by predicting the forecast of Bitcoin prices for 30 days which is quite defensible for this analysis with Artificial Neural Network used for particular and from the result it could be easily concluded that with ongoing research in the field of deep learning and artificial intelligence current recurrent neural networks provides much efficiency over usual time series forecasting such as cluster algorithm and moving averages analysis algorithm but this research introduces RNN and their types and reason for the use of RNN in time series prediction alongside the analytical results.

Both models of RNN shows not much differences in accuracy in predicating the prices for next 30 days (RMSE) but LSTM as well as GRU model describes the future trend of the market which is quite helpful for traders but in the prediction of 1 day of moving prices of Bitcoin considerable changes is to be seen as GRU-RNN has advantages over LSTM-RNN with better accuracy determined by RMSE of 396.408 to the LSTM's RMSE which is quite higher with value of 519.626 and shows efficiency in mimicking the actual real-time graph and GRU-RNN also holds advantages over LSTM-RNN in training time shown in Fig (7), GRU has less parameters than LSTM therefore it has less training time but due to its internal functionality of unit gates it is more efficient and reliable for time series prediction.

GRU structure of Recurrent Neural Networks is 12% faster than LSTM-RNN, by using the hyper parameters and features spaces and increased number of epochs for both RNN structures that produces best accuracy and reliability results GRU model always holds advantages over LSTM model in this analysis of time series prediction.

GRU model is shown in this analysis to provide better and reliable significations from other conventional ways of time series prediction which do not uses ANN. I believe that this analysis will be widely used for widespread application field

for time series prediction and I am convinced that there is a room for improvement of in time series prediction with the use of concepts of deep learning field and artificial neural networks, in addition to this analysis research, it could be improved by taking much efficient ways to collect data then using grid search as the formatting for parameter optimization.

The results and analysis presented in this analytical research should not be taken into consideration for investing or any investment advice as future will always be uncertain but can be analyzed.

8. REFERENCES

- [1] Georg Dorffner, Neural Networks for Time series prediction, 1996
- [2] Jeffrey Elman. "Finding Structure in Time". In: Cognitive Science 14 (Mar. 1990). DOI: 10.1016/0364-0213(90)90002-E.
- [3] K. Greff et al. "LSTM: A Search Space Odyssey". In: IEEE Transactions on Neural Networks and Learning Systems 28.10 (Oct. 2017), pp. 2222–2232. ISSN: 2162-237X
- [4] François Chollet, "Keras", 2015, In: Github Journal
- [5] TensorFlow, 2015, "TensorFlow: Large-scale machine learning on heterogeneous systems", <https://www.tensorflow.org/about/bib>
- [6] Google Trends. URL: <https://trends.google.com> (visited on 10/01/2018).
- [7] Yahoo Finance - Business Finance, Stock Market, Quotes, News. URL: <https://finance.yahoo.com/> (visited on 10/01/2018).
- [8] Bitcoincharts. URL: <https://bitcoincharts.com/> (visited on 10/01/2018).
- [9] A. Meyler, G. Kenny, and T. Quinn, "Forecasting Irish Inflation Using ARIMA Models," Technical Paper 3/RT/1998, Central Bank of Ireland Research Department, 1998.
- [10] <http://www.wildml.com/2015/10/recurrent-neural-network-tutorial-part-4-implementing-a-grulstm-rnn-with-python-and-theano/> (visited on 19/03/2019)