International Journal of Advanced Multidisciplinary Research
ISSN: 2393-8870
www.ijarm.com
(A Peer Reviewed, Referred, Indexed and Open Access Journal)
DOI: 10.22192/ijamr Volume 11, Issue 1 -2024

Research Article

DOI: http://dx.doi.org/10.22192/ijamr.2024.11.01.010

A Glimpse into Time: Harnessing AI for Accurate Time Series Forecasts

Sarthak Rout
Department of CSE,
Chandigarh University,
Mohali, Punjab,
21BCS6312

Ishpreet Kaur
Department of CSE,
Chandigarh University,
Mohali, Punjab,
21BCS6424

Keywords

Machine Learning,
Deep Learning,
ANN,
Time Series,
Forecasting

Abstract

This project aims to develop a model capable of predicting time series values for complex temporal data. This type of data prediction relies greatly on historical data and hence is a part of unsupervised machine learning. The inclusion of deep learning approaches inculcating neural networks aids in the development of strategies that produce outputs with higher precision than traditional time series prediction models. These strategies can effectively handle the complexity of weather forecasting, energy consumption, financial market analysis etcetera. Modern approaches like Artificial Neural Network perform comparatively better than other approaches like Long Short Term Memory.

I. INTRODUCTION

Time series data refers to attributes that depend majorly on time which can span manifold. In normal regular time data with intervals of approximately the same size, patterns are easy to predict. Pattern recognition in data oriented around time with unexpected and hard to predict events can be tackled with the help of time series prediction. Time series data evaluation has found its usage in various fields of technology, finance, agriculture, sciences and many interdisciplinary fields. We found multiple problems and corresponding solutions to time series data management that are handled by the implementation of deep neural networks.

Almost all data can be directed and adjusted with time as a major attribute which leads to the rising problem of time series data forecasting. The increasing complexity of the same needs to be handled with the application of deep learning methodologies in order to ensure the practical usage and effective and quicker solutions. These algorithms should also be subject to change and be competent enough to change and learn depending on the type of data it encounters because expecting any regular pattern is highly unlikely and hence could make the model prone to errors.

The dataset that was used is of two types and utilizes the contents of portfolio data and electricity consumption data. The first type of data that is portfolio data has been thoughtfully processed with the help of artificial neural network comprising of Recurrent Neural Network (RNN) and one of its most effective technique called Long Short Term Memory(LSTM). The prediction of this sequential data was brought about by incorporating a total of 172 neurons involving the input and hidden layers. The other dataset of electricity consumption had its first column represented by date and time in string format of 'yyyy-mm-dd hh:mm:ss'. The rest of the columns are float values denoting energy consumption in kW(Kilo Watts).

The main reason for choosing LSTM over other neural network architectures lies in the fact it has a better handling capabilities for Long Term Dependencies and focuses greatly on minimizing the vanishing Gradient problem. It provides ease in remembering smaller sets of data points and making decisions on that basis by adjusting the weights and biases involved. Unlike RNN, the smaller amount of information does not lose its context when passed after each timestep. LSTMs have a much more sophisticated parallelizable structure that aids in faster and better performance of the overall model.

The concept utilized in this paper is first, a normal LSTM model trying to handle the portfolio dataset and then a sinusoidal LSTM model that takes help from the mathematical sine wave function.

II. RELATED WORK

The paper takes inspiration from related works of various prominent authors such as Anastasia Borovykh[1], Neeraj Kumar[2], Gang Xiao[3], Ashu Jain[4] and more.

The first paper talks about Convolutional Neural Networks for integrating conditional information by Anastasia Borovykh, Sander Bohte, Cornelis W. Oosterlee[1].

The second paper talks about Artificial Neural Networks to predict weather patterns in complex temporal weather data by Neeraj Kumar and Govind Kumar Jha[2].

The third paper talks about Convolutional Neural Networks to capture intricate temporal patterns by Zhipeng Shen, Yuanming Zhang, Jiawei Lu, Jun Xu, Gang Xiao[3].

The forth paper talks about Neural Networks with hydrological insights by Ashu Jain, Avadhnam Madhav Kumar[4].

The fifth paper talks about hybrid model of ARIMA and neural networks by G.Peter Zhang[5].

The sixth paper talks about local linear wavelet neural network by Yuehui Chen, Bo Yang, Jiwen Dong[6].

The seventh paper talks about hybrid methodology by Cagdas Hakan Aladag, Erol Egrioglu, Cem Kadilar[7] to handle non-linear time series data.

The eighth paper talks about a fuzzy neural network by L.P. Maguire, B. Roche, T.M. McGinnity, L.J. McDaid[8] to handle a chaotic time series.

III. THEORY OF LSTM

Long short-term memory networks were invented by Hochreiter and Schmidhuber in 1997. They are fundamentally a part of Recurrent Neural Networks (RNN) but with improved functionality because of its shorter recollection of memory points and better contextualization after every timestep. The creation of weight matrices and biases is time-independent in LSTM architecture and hence they don’t change from timestep to another. The length of every sequence is calculated and utilized for the understanding of the computations that are expected out of each timestep. The dimensionalities of the weight matrices are equated with the help of the value of the input neurons and the hidden layer neurons in association with the bias at that particular step and they have been calculated and summarised as follows:

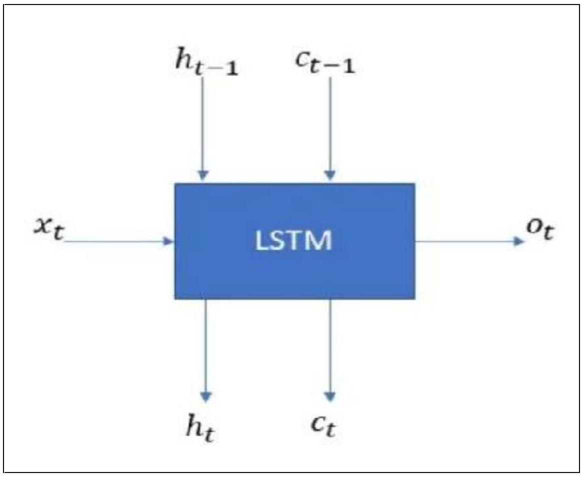

Fig. 1. Typical LSTM structure.

$$\mathbf{ft = \sigma g\ (Wf \times xt + Uf \times ht-1 + bf)}$$
$$\mathbf{it = \sigma g\ (W\times xt + U_{\text{¡}} \times ht-1 + b_1)}$$
$$\mathbf{Ot = \sigma g\ (W \times xt + Uo\times ht\ -1 + bo)}$$
$$\mathbf{c't = \sigma c\ (Wc \times xt + Uc \times ht\ -1 + bc)}$$
$$\mathbf{Ct = ft \cdot Ct\text{-}1 + it\ ct}$$
$$\mathbf{ht= Ot\sigma c\ (Ct)}$$

Note the following points for the above:

1. These equations get recomputed for each timestep and must be again calculated for the next timestep and this goes on for as many time steps that are present in the model.

2. As discussed before the weight matrices are time independent. Different outputs of all timesteps are summarized to be finally utilized on the same weight matrix.

IV. PROPOSED METHODOLOGY

The main goal of this paper was the creation of a LSTM model that is effectively able to handle sequential time series data and forecast on its basis different types of dataset such as energy consumption and financial data. It requires computational efficiency due to short condensed data point group. It provides a overall average accuracy of 92% due to its lower accuracy in vanilla LSTM for the portfolio data and higher accuracy in the sinusoidal LSTM for the electricity consumption data.

A. Preprocessing

The data is preprocessed by reshaping the data frame which is storing the CSV file. The dataset is split into train and test datasets in the accepted ratio of 4:1. The portfolio dataset has a total of 50 epochs which are consecutively run to achieve an optimal accuracy of about 67% initially and thereby moving towards 87%. The trained model is then loaded with almost 3000 data points to be run in 10 epochs and come very close to the actual data.

On the other hand, the energy consumption dataset has 1,40,000+ entries without any missing values thereby increasing the complexity of the overall dataset. This required more computation time, and the system was only able to successfully run 10 epochs with a batch size of 70 points at a time.

B. LSTM Modeling

The two models vary greatly not only because of the dataset they are handling but also the additional technique of the sinusoidal wave function that works with a total of 2 LSTM layers alternatively used with 2 dropout layers and finally one dense layer converging all the hidden layer outputs followed by the final activation layer which gives the final output.

Algorithm 1 Pseudocode for Performing Vanilla LSTM Modeling

```
// 1.Preprocessing

1.1 Train, test dataset splitting
1.2 Scaling of dataset according to accepted format
1.3 Specification of random state as 42

// 2. LSTM Modeling

2.1 Sequential model building
2.2 Input layer (64 neurons) with ReLu activation.
2.3 Hidden Layer(128 neurons) with ReLu activation.
2.4 Output Layer(64 neurons) with ReLu activation.
2.5 Final SoftMax activation followed by model compilation.
2.6 Mean Squared error calculation with Adam optimizer.
```

Algorithm 2 Pseudocode for Performing Sinusoidal LSTM Modeling

```
// 1.Preprocessing

1.1 Scaling with the help of MinMax Scalar.
1.2 Providing Window size of 50
1.3 Creating series by dropping certain values.
1.4 Train, test dataset splitting (80% and 20% respectively)

// 2. LSTM Modeling

2.1 Sequential model building.
2.2 LSTM layer of input shape (50,1)
2.3 Specifying dropout of 0.5
2.4 Another LSTM layer of shape 256
2.5 Another dropout of 0.5
2.6 Final Dense layer followed by linear activation
2.7 Adam optimizer-based Mean Squared error
```

V. RESULTS

The performance of our models have been evaluated on the basis of Loss Error function and accuracy calculation of the predictions made. Running 50 epochs resulted in a total of 0.15781 for the vanilla LSTM on the portfolio dataset the graph of which has been shown below comparing the true data and predicted data.

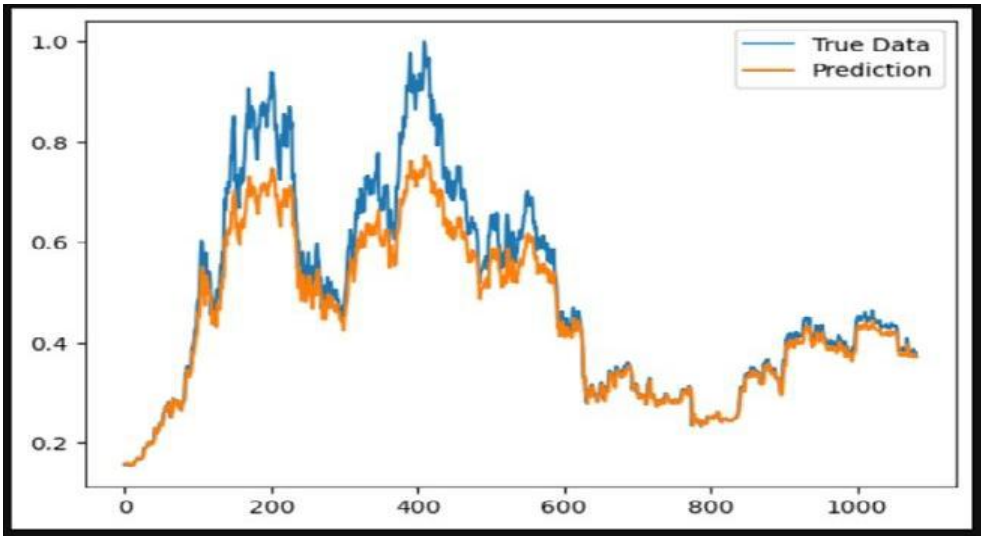

Fig. 2. Accuracy of vanilla LSTM

The electricity consumption dataset with multiple data points is processed by the sinusoidal LSTM which results in a more accurate forecasting outcome. The utilization of the mathematical function brings about a much more rounded and detailed gradient descent alteration which helps in precise updation of the weights and biases at each timestep and thereby providing an accuracy close to 96%. The sinusoidal wave function is a graph denoted by the sine curve that is given as follows:

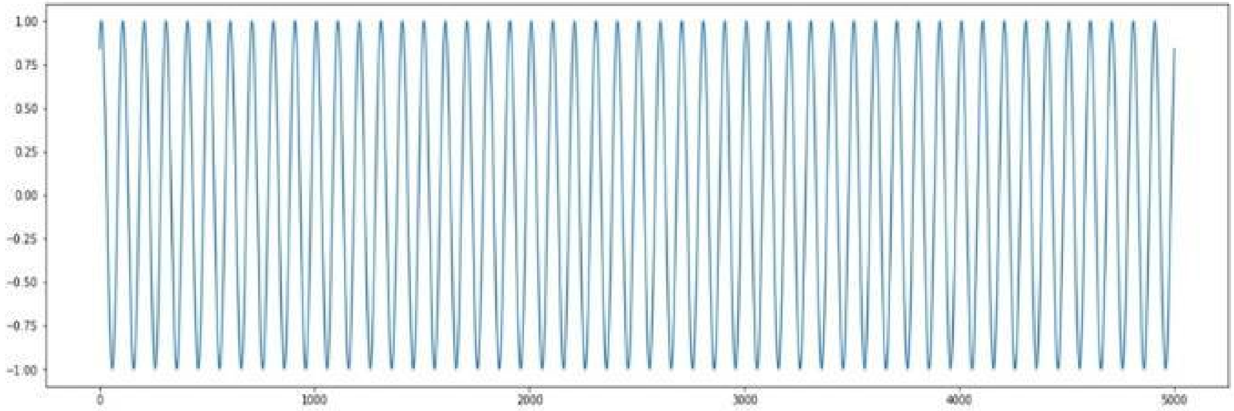

Fig. 3. Compact version of sinusoidal LSTM

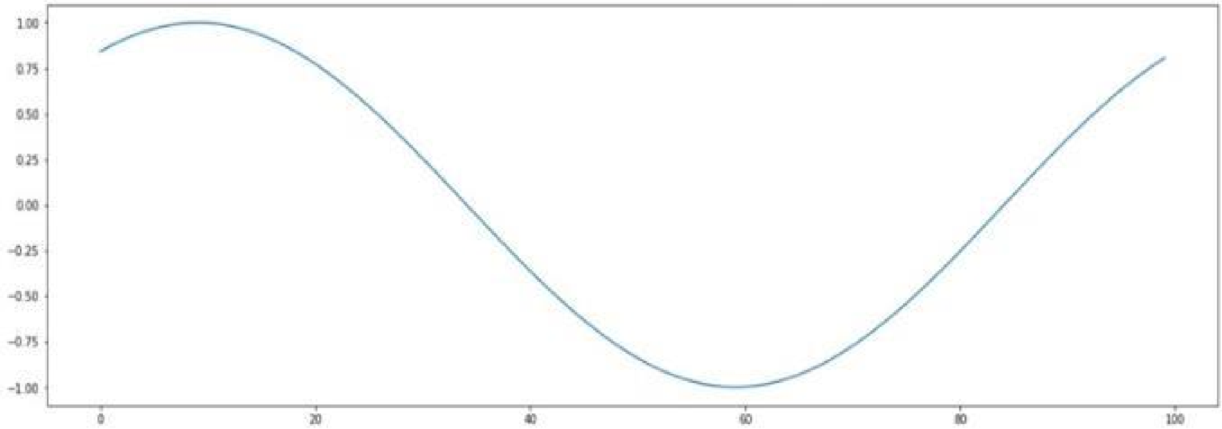

Fig. 4. Expanded version of Sinusoidal LSTM

The comparison of true data and the predicted data as given by sinusoidal LSTM is as follows:

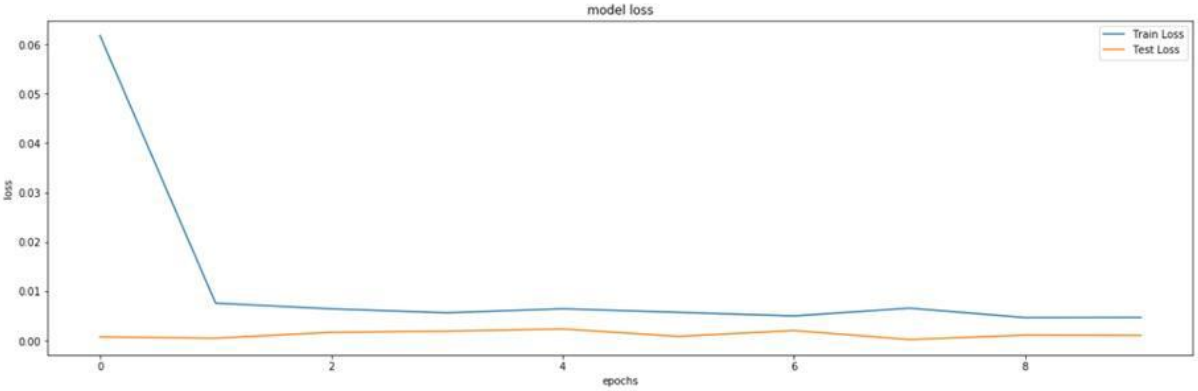

Fig. 5. Accuracy comparison of Sinusoidal LSTM

The loss of sinusoidal LSTM is 0.0010623. This shows that how advanced and accurate the sinusoidal LSTM can be over vanilla LSTM which outperforms recurrent neural networks when dealing with sequential time-series data.

VI. DISCUSSION

As discussed in the previous sections, there is great advantage in the strategic implementation of sinusoidal LSTM over vanilla LSTM when working with sequential time-series data that prioritizes quantity as well as quality of data records that must be handled with a specially curated complex batch relevant model that works with a looping mechanism along with alternate dropouts to reduce irrelevant attention drawn towards unnecessary and less important parameters that make their way into the dataset. This task of dataset specific modeling lacks generalization but makes up for it in the accuracy that it provides. We also learn and conclude that this type of modeling is computationally expensive and hence must be run-on high-performance machine to live up to their mark.

VII. CONCLUSION

In conclusion, the proposed project aims to develop a capable model which can extract valuable data from Time Series data and hence help out in creating accurate and substantial business and predictions. The incorporation of the various models ensures the quality of the forecasting to be optimal while being relatively simple to understand and implement. Versatility is what makes the project so attractive as a whole.

The entire paper and model can be summarized in three points which are as follows:

- A comparative study between RNN and LSTM by evaluating the contextualization of data points and specification of smaller batch sizes.
- A comparative study between vanilla LSTM and sinusoidal LSTM based on generalization factor and the speed at which the model compiles and finishes its recurrent epochs.
- Dataset evaluation based on ease of preprocessing and model implementation while considering the computation time and model accuracy.

ACKNOWLEDGMENTS

The authors express gratitude to Kaggle and GitHub repositories for their dataset contribution. The authors also thank the conceptualization and ideation provided by anonymous users for innovation in LSTM implementations.

REFERENCES

[1] Conditional Time Series Forecasting with Convolutional Neural Networks by Anastasia Borovykh, Sander Bohte, Cornelis W. Oosterlee https://arxiv.org/abs/1703.04691

[2] A Time Series ANN Approach for Weather Forecasting by Neeraj Kumar and Govind Kumar Jha https://www.academia.edu/download/63451733/3113ijctcm0220200528-69623-wzqd4u.pdf

[3] A novel time series forecasting model with deep learning by Zhipeng Shen, Yuanming Zhang, Jiawei Lu, Jun Xu, Gang Xiao https://www.sciencedirect.com/science/article/abs/pii/S0925231219304461

[4] Hybrid neural network models for hydrologic time series forecasting by Ashu Jain, Avadhnam Madhav Kumar https://www.sciencedirect.com/science/article/abs/pii/S1568494606000317

[5] Time series forecasting using a hybrid ARIMA and neural network model by G.Peter Zhang https://www.sciencedirect.com/science/article/abs/pii/S0925231201007020

[6] Time-series prediction using a local linear wavelet neural network by Yuehui Chen, Bo Yang, Jiwen Dong https://www.sciencedirect.com/science/article/abs/pii/S0925231205000457

[7] Forecasting nonlinear time series with a hybrid methodology by Cagdas Hakan Aladag, Erol Egrioglu, Cem Kadilar https://www.sciencedirect.com/science/article/pii/S0893965909001475

[8] Predicting a chaotic time series using a fuzzy neural network by L.P. Maguire, B. Roche, T.M. McGinnity, L.J. McDaid https://www.sciencedirect.com/science/article/abs/pii/S0020025598100269

Access this Article in Online	
	Website: www.ijarm.com
Quick Response Code	Subject: Computer Science
DOI:10.22192/ijamr.2024.11.01.010	

How to cite this article:
Sarthak Rout, Ishpreet Kaur. (2024). A Glimpse into Time: Harnessing AI for Accurate Time Series Forecasts. Int. J. Adv. Multidiscip. Res. 11(1): 83-89.
DOI: http://dx.doi.org/10.22192/ijamr.2024.12.01.010