LTE User behavior prediction Via LSTM

Mehrdad Tanha
Data Mining & Network planning
MTN Irancell
Tehran, Iran
Mehrdad.ta@mtnirancell.ir

Muhammad Adeel
Data Mining & Network planning
MTN Irancell
Tehran, Iran
muhammad.ad@mtnirancell.ir

***Abstract*—In this letter we want to use LSTM-RNN to analyze and predict the behavior of LTE users in mobile network. Predicting the user statistics and data usage on the whole mobile network is going to be studied. Analyzing and predicting the user charging data requests in LTE network and finally, finding abnormal base stations has been predicted via DNN networks. To accomplish that we considered aggregated LTE users under each base station and considered the behavior statistics as a time series. Ultimately, we forecasted the coverage index (CI) as a calculated metric to identify anomalies and determine the count of abnormal sites with fewer LTE CDRs compared to normal ones using LSTM on generated time series. This prediction can be used for planning and optimization teams to find and solve any abnormal site level parameters. Proposed method can have both the role of corrective actions on base stations to rectify any telecommunication issue on live network and also predictive action for forecasting network condition in site level scale to increase user satisfaction in advance.**

Keywords*—*time series prediction, deep learning algorithms, LTE users, look-back window, CDR forecasting, smart site planning, smart parameter tunning.

I. INTRODUCTION

In most of mobile networks, operators tend to provide higher throughputs and better user experience as higher throughputs means more interest in data usage and higher revenues for telecommunication operators. Therefore, they use KPI (Key profile indicators), CDR (Charging Data Record) and MDT (minimized drive test) to monitor network major/minor measures and optimize the network profiles accordingly. CDR is a bunch of data which provide us information about a chargeable event (e.g. time of call set-up, duration of the call, amount of data transferred, radio generation on the network which data transferred, etc.) for use in billing and accounting. So, by using number of users and data requests in each technology (2G, 3G, and 4G) we can have a map on the behavior and statistics of users in each area. Mostly, Subscribers' behavior and their coverage are primarily influenced by the geographical context and prevalent handset categories within a given region. In this paper, we used CDR for identifying regions and base stations (Nodeb & eNodeb) which have higher 3G service requests even though, they provide 4G coverage. It is worth to note that all of our input data sources consist of authentic live network data, with no data generation taking place. At the first step, we will analyze all of the input sources to find the coverage index which will be discussed in details later and finally predicting the network behavior in weeks ahead to identify them in advance and solve their issues for the better user experience.

After introduction of neural network and their application in different areas, in the last few years, new algorithms have reached the point that deep learning has outperformed than common ANN architectures in complex problems. The fact that DNNs are able to handle missing or noisy data and are capable of mapping complex functions, made them widely useful in many business and academic applications. Between these applications, Time Series Forecasting (TSF) has recently attracted the attention of many researchers which came to many applications such as: trend prediction, anomaly detection, capacity planning, incident prediction and so on. To apply AI models in our industrial application we decided to use Vanilla RNN as recurrent neural networks but RNNs are also have limitations in looking back in time as some memories may vanish or explode [1-2]. This problem was solved by introducing Long Short-Term Memory Recurrent Neural Networks (LSTM-RNN) [3-6]. LSTM networks are extensively employed in stock prediction due to their adeptness in analyzing data with temporal dependencies. They have demonstrated superior performance compared to other deep neural network (DNN) algorithms, particularly in extracting valuable insights from noisy time series data. [7-9]. The LSTM architecture is based on cell state and three gates calculated as (1): the forgotten (Ft), the input (It), and the output gates (Ot) which can weigh the information for each cells. This architecture will finally result in distinguishing any context changes and forgetting irrelevant data [10]. To be more precise, we have long-term memory (cell state) and short-term memory (hidden state) to remember the importance of data which modified based on results of LSTM gates. “Fig. 1” illustrates the LSTM structures.

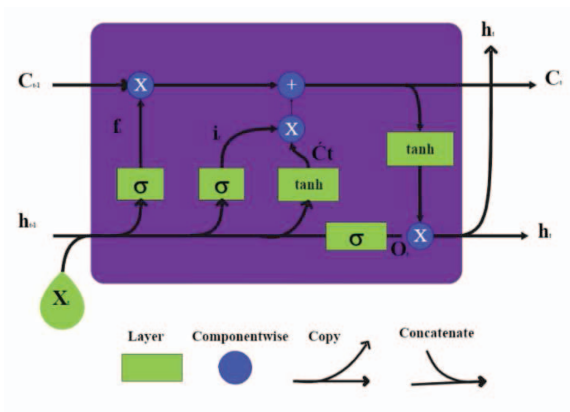

Fig. 1. LSTM Node architecture

As shown in the figure, we have input values at time t as X_t and hidden state from previous time step as H_{t-1} which after concatenation will be passed to sigmoid function to create forget and parts of input gates. The sigmoid function serves as

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00162

a gate or regulator, allowing only values between 0 and 1 to pass through. In this context, a value of zero indicates that no information should be added to the cell state (complete forgetting), while a value of one signifies high importance from a long-term memory perspective.

$$\begin{aligned} F_t &= \text{sigmoid}\,(W_t * X_t + U_t * H_{t-1} + b_t) \\ I_t &= \text{sigmoid}\,(W_i * X_i + U_i * H_{i-1} + b_i) \\ \acute{C}_t &= \tanh\left(W_j * X_j + \mathrm{U_j} * H_{j-1} + b_j\right) \\ C_t &= F_t \odot C_{t-1} + I_t \odot \acute{C}_t \end{aligned} \tag{1}$$

Where W and U are weight matrices learned for each step and $\odot$ denotes element-wise vector product.

Finally we have O_t and updated hidden state as below (2):

$$\begin{aligned} O_t &= \text{sigmoid}\,(W_o * X_t + U_o * H_{t-1} + b_o) \\ H_t &= O_t \odot \tanh(C_t) \end{aligned} \tag{2}$$

Where $tanh$ is the nonlinear tanh activation function used as an importance leverage of input and short term memory. The Symbol $\odot$ is used to denote Hadamard product.

Certainly, in addition to the classic LSTM, there are many other LSTM architectures that can integrate or connect different LSTM components to simplify, improve, or optimize the network based on the specific problem, the scale of input data, and processing capabilities. Various LSTM architectures are typically chosen depending on the dataset's structure, the number of features involved, and computational constraints. There isn't a one-size-fits-all approach for selecting a specific model across different applications; each model comes with its own set of advantages and disadvantages tailored to the specific problem at hand. Gated Recurrent Unit neural networks (GRU) as an example have simpler architectures and have shown successful performance in long sequence applications involving sequential/temporal patterns. GRU in architecture is almost same as LSTM but it is simpler and consists of two gates: reset gate and update gate and also cell state and hidden state in LSTM are merged to one hidden state in GRU [11-13]. This simplicity may raise the doubt that LSTM will outperform GRU, but many works have shown that beside of lower computational expenses, it can perform as good as LSTM in tasks such as: music modeling, speech signal modeling and natural language processing [14-15]. So, it would be interesting to study and compare both neural networks for our case as LTE users in live network can have seasonal and sequential patterns which can challenge both GRU and LSTM as popular recurrent neural networks in telecommunication.

Since user behavior in mobile networks varies significantly and can be directly influenced by factors such as seasons, network configuration adjustments, and events, we may accumulate valuable long-term data that could impact our present situation. Conversely, we may also possess outdated data from the past that is no longer relevant and should be discarded. To be more precise, we have non-stationary time series as the main properties change over time which consequently affect the mean and variance of our data. Therefore, there is always important memory in the past which is very valuable for our predictions. LSTM has demonstrated its effectiveness in handling sequential data patterns characterized by significant time steps between relevant data points. In our scenario, the generated CDRs for mobile users is computed and analyzed at various timestamps, with our objective being the prediction of user behavior in the near future. Many conventional machine learning methods often struggle to meet our requirements due to the sequential nature and temporal correlation inherent in time series data, which are crucial aspects of our forecasting task.

II. DATA AND EXPERIMENTS

Within mobile networks, various radio technologies like 3G, 4G, and the more recent 5G each come with their own specific coverage capabilities. In our context, different site scenarios present challenges where data requests predominantly originate in 3G rather than 4G due to coverage limitations, potentially leading to user dissatisfaction. Numerous factors contribute to this issue, including cell parameters, fluctuating user locations, variations in user density, and other environmental changes. Each site within the area, depending on its configuration, may experience actual coverage gaps, with the status of sites influenced by dynamic user densities and environmental shifts [Fig. 2].

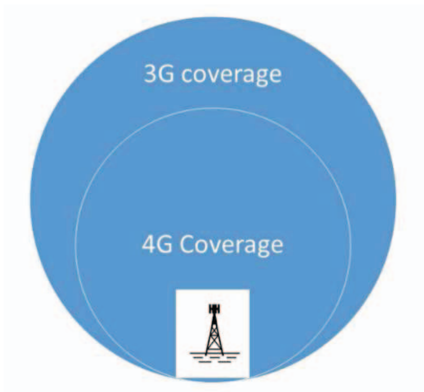

Fig. 2. Site coverage on U & L base station

Within expansive telecommunication networks, several crucial factors and network elements can directly or indirectly influence mobile users. These include the core network, transmission nodes, geographical features, artificial obstacles affecting coverage, and the user maturity index across various regions.

Our CDR database comprises 42 features and surpasses 100 million records per week. It covers the entirety of the network, encompassing various site configurations, geographical attributes, user handset classifications, and more. Data concerning the 5G network is excluded due to an insufficient number of deployed samples and NR users, rendering it impractical for analysis.

Initially, cell/sector-level data is aggregated to derive site-level CDR information, resulting in a reduction in the size of raw data by over ninefold. Given our focus on examining mobile user behavior within site-level scenarios, this process does not affect our outcomes. Therefore, CDR features primarily provide information on the number of old and new mobile SIM cards (USIMs) associated with each base station, user numbers per RAT type (XG) in site level, and the proportion of data requests for each RAT, among others.

In essence, within a live telecommunication network just like the one we analyze and similar database, we encounter numerous features that impact mobile user behaviors, some of which exhibit high correlations with others. Through the pre-processing phase and leveraging telecommunications concepts, we can identify and potentially eliminate redundant data. For instance, there exists a robust correlation between the subscribers of each RAT for every base station, potentially reducing the number of features. If a base station has, for instance, "75%" LTE subscribers and less than "20%" UMTS subscribers, the remaining users are likely utilizing the GSM network. Through correlation calculations across various features and leveraging telecommunications principles, we ascertain a strong correlation between 3G and 4G utilization. Specifically, users in different site locations with complete RAT configurations tend to favor 4G over 3G networks and are less inclined towards 2G networks. Furthermore, our database includes diverse utilization percentages for each RAT type at each site location, ranging from "1%" to "100%" utilization, which are undoubtedly correlated. This technical preprocessing facilitates faster and more efficient model convergence, resulting in a reduction of inputs for our CI calculation as the output. We illustrate a segment of the correlation heatmap in [Fig 3].

Additionally, we chose to categorize sites into regions based on the geographical characteristics of base stations, handset distribution within each province, and LTE user penetration rates. These regions have been carefully selected to reflect similarities in user behavior, site parameter settings, site distribution, and geographical layouts. This approach enhances the accuracy and generality of the model by accounting for regional variations.

Preprocessed data undergo analysis to compute our proposed metric (CI) for each site code in every region. However, before proceeding, we exclude base stations with zero or extremely low LTE traffic due to site unavailability or the inclusion of new integrated/license-limited base stations in the database. Consequently, we establish an observation period of four consecutive weeks on CI to identify abnormal samples for a site. Undoubtedly, sites experiencing LTE downtime not only hinder the model's ability to learn and predict feature relationships but also introduce misleading information.

We employ Python for our analysis and utilize Keras for predictive modeling in further steps. The calculated CIs for each site location per week will be incorporated into our database as the metric for abnormality. To determine problematic sites each week, we established Coverage Index Thresholds based on Table 1.

TABLE I. COVERAGE INDEX THRESHOLDS

Category	*Coverage Index Thresholds (CIT)*
Critical	Usage>50
High	30<Usage=<50
Medium	20<Usage=<30
Normal	Usage=<20

The generated output serves multiple purposes: it can be utilized for optimization based on severity levels and can also function as input for our supervised prediction model. In this model, preprocessed features and calculated coverage indices are used as X and Y variables, respectively.

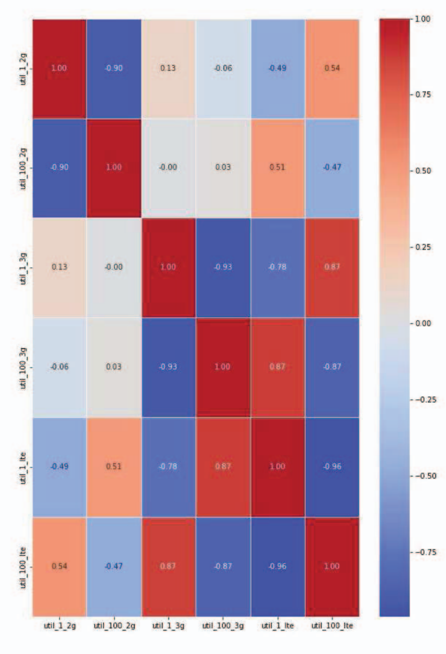

Fig. 3. Correlation heatmap between some of features

Due to the vast number of features and the extensive amount of data, we opt for employing Deep Learning algorithms to analyze and ultimately predict user CDR generations in radio technology. To validate the superiority of LSTM over GRU, we train two distinct models with an equal number of nodes versus varying numbers of epochs needed for training, then compare the loss of these models. Using Keras version "2.15.0", we initially set both LSTM and GRU with 50 nodes and a single Dense layer to forecast a single numerical value. As depicted in the figure, LSTM exhibits superior performance in our scenario, particularly with a lower number of epochs for training the model. [Fig. 4].

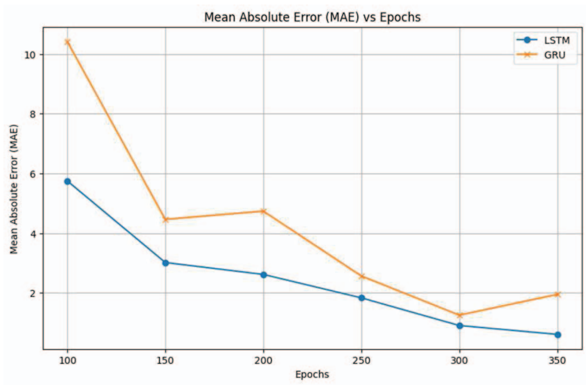

Fig. 4. Comparison of GRU and LSTM for LTE users

Given the comparison results and the time-series nature of our database, we opt for LSTM, which demonstrates advantages over other proposed DNN models such as GRU in our analysis. A Vanilla LSTM comprises a single hidden layer of LSTM units and an output layer employed for making predictions. This type of LSTM can be defined for univariate time series forecasting. The model is trained using the efficient Adam version of stochastic gradient descent and is optimized using the Mean Absolute Error (MAE) loss function. Adam serves as an optimization algorithm that can be utilized in lieu of the classical stochastic gradient descent approach to iteratively update network weights based on training data [16].

In determining the optimal activation function for our scenario, we conduct experiments with three different options: Sigmoid, hyperbolic tangent (Tanh), and rectified linear unit (ReLU). After analyzing the results over 100 epochs, we find that ReLU is the most suitable choice for our prediction task [Fig. 5].

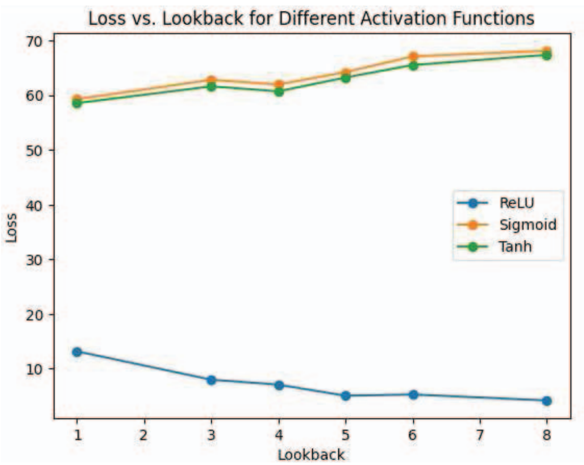

Fig. 5. Calculation of Loss for different activation functions in different look back window

Referring to our database specifications, we establish an observation period to identify temporary problematic sites. Since LTE license limitations or configuration change trials typically extend beyond a month, encompassing the time for providing licenses and rolling back trials, we find it pertinent to examine the impact of lookback periods on our model's performance. Considering our database details and technical considerations outlined, we investigate a range of six close lookback windows, namely "[1, 3, 4, 5, 6, 8]", to explore the relationship between the monitoring window and different lookback durations.

From a technical perspective, significant contextual changes in our database may occur approximately four weeks after the onset of an issue at each base station. If an issue persists for more than three weeks, it triggers an abnormality flag. The results illustrated in "Fig. 6" indicate that adopting a 5-week lookback window proves to be the most optimal choice for nearly all regions.

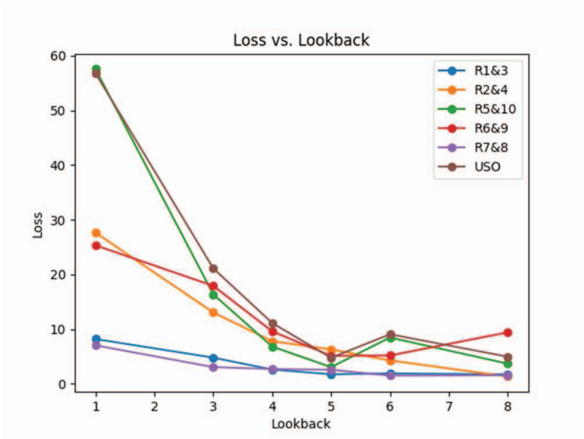

Fig. 6. Loss vs. Look-Back for different regions

In the last phase, to introduce challenges for our LSTM model, as illustrated in "Fig. 7", we deliberately assign low weights to certain undesired events by manipulating data in weeks 4 and 5. This is done to assess the resilience of the LSTM model designed for our specific objective.

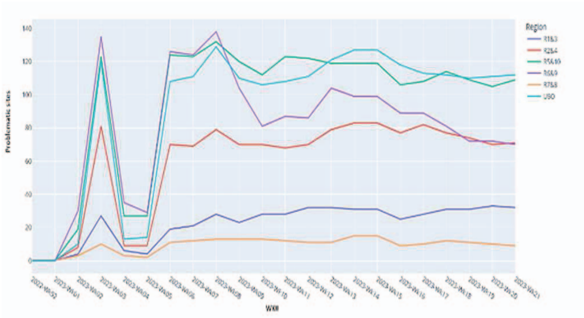

Fig. 7. Manipulated Time series on site abnormality

We train the model using current data and aim to forecast data for the following week. To assess the model's performance, we utilize Mean Absolute Error (MAE) for each week, which serves as a metric for the errors between predicted and actual values in our analysis. Both the predicted and actual values share the same scale of measurement (CI/CIT in our case), and it has been demonstrated that MAE can be the most suitable option for time series scale-dependent accuracy measurements [17-18]. After training the model and computing the errors, we find that the mean MAE is 2.33, which is deemed acceptable for our purposes.

Interestingly, as shown in "Fig. 8", the trained LSTM model accurately predicts values, with some regions, such as R5&10, achieving 100% correctness in their predictions, while other regions exhibit acceptable MAE for WK22. It should be noted that site numbers do not exhibit a discernible trend, and fluctuations typically occur based on various circumstances. Therefore, forecasting the network using classic machine learning algorithms is nearly impossible. The provided predictions are based on the LSTM model, which incorporates a lookback period of 5 weeks and forecasts one week ahead for planning and optimization purposes.

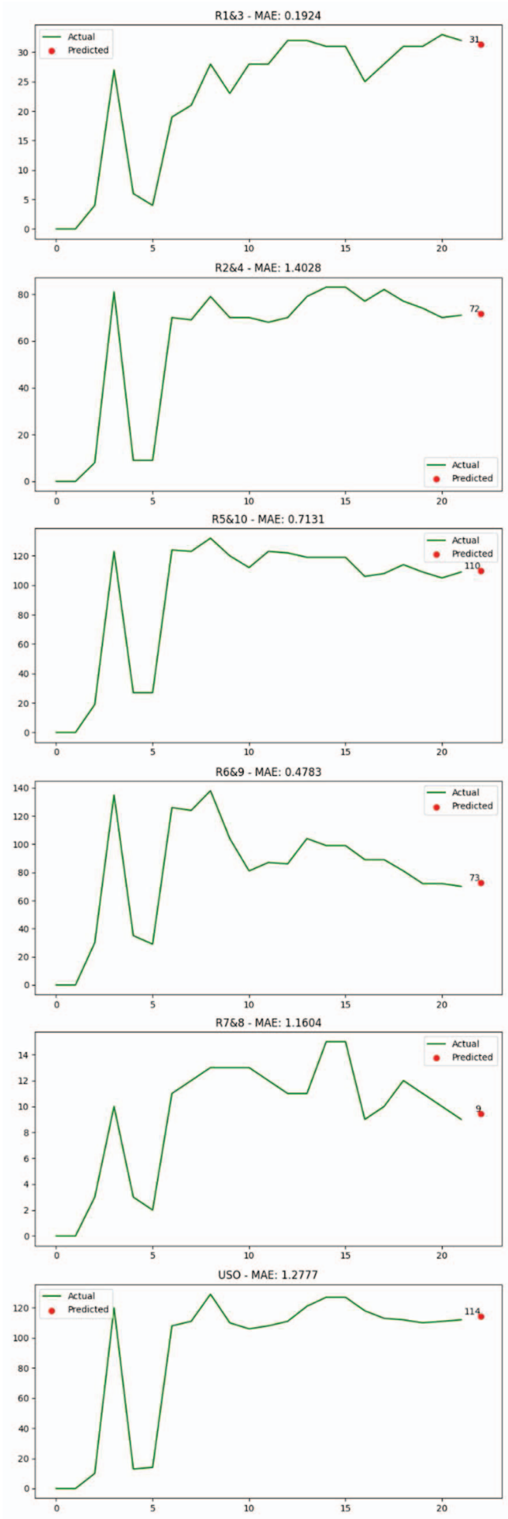

Fig. 8. Time series prediction per region via LSTM

In "Fig. 9", we present the actual output of our CDR analysis for the following week, utilizing real values from raw CDR reports. By comparing these actual values to our predicted values (WK22) as an example, we assessed the validity of our model.

We conducted weekly analysis and achieved MAE values ranging from approximately 2 to 5 for each week ahead which is completely acceptable for forecasting LTE user situation in the site level scenario. This analysis provided optimization owners with insights into the site situation and LTE user behavior several weeks in advance. The predictive outlook facilitated proactive actions that enhanced user experiences and furnished us with a heatmap of user statistics for intelligent site planning.

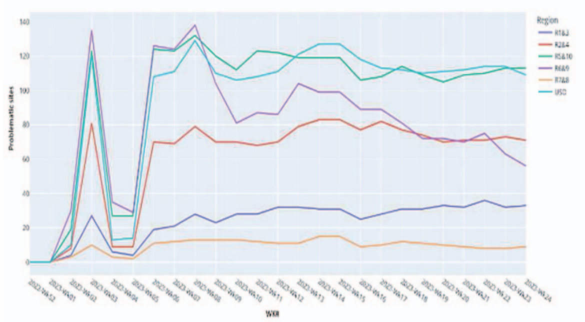

Fig. 9. Actual Time series on abnormal base stations

III. Conslusion

In this paper we evaluated the DNNs in the field of telecommunication and their possibility and accuracy in network analysis and smart optimization. Our results proved that there is a considerable potential in AI applications in telecommunication (smart planning and optimization). We utilized CDR reports to analyze and predict the LTE user behavior in site level scenarios. We categorized sites based on the behavior of LTE users and provided a scale of abnormality for them. In future, we plan to implement LSTM on user level point of view to forecast their behavior in service level and application level (application layer). it will be a considerable advantage in radio planning and link budget planning for special users having high revenue generation. We can implement smart cell and hardware expansions based on the dynamic user requests in each site location which is a leapfrog in the field of mobile site planning and real-time smart optimization.

References

[1] Josef Hochreiter. Untersuchungen zu dynamischen neuronalen Netzen. Master’s thesis, Institut fur Informatik, Technische Universitat, Munchen, (April 1991):1{71, 1991.

[2] Michael C Mozer. Induction of Multiscale Temporal Structure. In Advances in Neural Information Processing Systems 4, pages 275{282. Morgan Kaufmann, 1992.

[3] Felix A. Gers, J¨urgen Schmidhuber, and Fred Cummins. Learning to Forget: Continual Prediction with LSTM. Neural Computation, 12(10):2451{2471, oct 2000.

[4] Felix A. Gers, Nicol N. Schraudolph, and J¨urgen Schmidhuber. Learning precise timing with LSTM recurrent networks. Journal of Machine Learning Research (JMLR), 3(1):115{143, 2002.

[5] Sepp; Hochreiter and J¨urgen Schmidhuber. Long Short-Term Memory.Neural computation, 9(8):1735{1780, 1997.

[6] JA A P´erez-Ortiz, FA A Felix A. Gers, Douglas Eck, J??rgen U. Schmidhuber, Juan Antonio P??rez-Ortiz, FA A Felix A. Gers, Douglas Eck, and Jorgen U. Schmidhuber. Kalman filters improve LSTM network performance in problems unsolvable by traditional recurrent nets. Neural Networks, 16(2):241{250, 2003.

[7] Hossain, M.A., Karim, R., Thulasiram, R., Bruce, N.D.,Wang, Y.: Hybrid deep learning model or stock price prediction. In: 2018 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1837–1844. IEEE (2018)

[8] Siami-Namini, S., Namin, A.S.: Forecasting economics and financial time series: Arima vs. LSTM. arXiv preprint arXiv:1803.06386 (2018).

[9] Fischer, T., Krauss, C.: Deep learning with long short-term memory networks for financial market predictions. Eur. J. Oper. Res. 270(2), 654–669 (2018).

[10] Qi Lyu and Jun Zhu. Revisit long short-term memory: an optimization perspective. In Deep Learning and Representation Learning Workshop (NIPS 2014), pages 1{9, 2014.

[11] F. A. Gers and J. Schmidhuber, "Recurrent nets that time and count," Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium, Como, Italy, 2000, pp. 189-194 vol.3, doi: 10.1109/IJCNN.2000.861302

[12] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. ArXiv:1406.1078.

[13] Yao, Kaisheng & Cohn, Trevor & Vylomova, Katerina & Duh, Kevin & Dyer, Chris. (2015). Depth-Gated Recurrent Neural Networks.

[14] Ravanelli, Mirco; Brakel, Philemon; Omologo, Maurizio; Bengio, Yoshua (2018). "Light Gated Recurrent Units for Speech Recognition". IEEE Transactions on Emerging Topics in Computational Intelligence. 2 (2): 92–102.

[15] Su, Yuahang; Kuo, Jay (2019). "On extended long short-term memory and dependent bidirectional recurrent neural network". Neurocomputing. 356: 151–161.

[16] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).

[17] Willmott, Cort J.; Matsuura, Kenji (December 19, 2005). "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance". Climate Research. 30: 79–82.

[18] Rob J. Hyndman, Anne B. Koehler, :Another look at measures of forecast accuracy,International Journal of Forecasting,Volume 22, Issue 4,2006, Pages 679-688, ISSN 0169-2070.