Machine learning based call drop healing in 5G

JANANI RAMASWAMY MUDALIYAR
MASTER´S THESIS
DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING | LTH | LUND UNIVERSITY

Lund University

Department of Electrical and Information Technology Faculty of Engineering LTH

Master of Science Thesis

Machine learning based call drop healing in 5G

Author:
Janani Ramaswamy Mudaliyar

Academic Supervisor:
Professor Fredrik Tufvesson

Industry Supervisor:
Abhishek Saurabh

Examiner:
Professor Fredrik Rusek

A Collaboration between:

Department of Electrical and Information Technology

Ericsson AB

Abstract

Self-Organizing Network (SON) functions include self-configuration, dynamic optimization and self-healing of networks. In the era of 5G, mobile operators are increasingly exploring areas of SON through Machine Learning (ML) techniques. It is seen that 5G packet switched networks are often hit with radio link failures, an important Key Performance Indicator (KPI). Reasons for a dropped call ranges from a failed handover to coverage/capacity issues. In current networks, such issues are resolved by KPI analysis, but these metrics are not always service/user specific. The aim with this master's thesis work is to investigate how well ML techniques can be applied to predict a call drop in real-time networks.

In the thesis, ML techniques, namely neural networks and logistic regression were used to classify the link status. Initially, the parameters which characterize a link connection, e.g. the Reference Signal Received Power (RSRP), Block Error Rate (BLER) and similar parameters were investigated. This was followed by applying ML to the selected parameter(s) and classifying a bad link (with failure) from a good link (without failure), this was the first phase of the thesis. The next phase was forecasting a radio link failure before one occurs. This forms phase two of the thesis and the start of the self heal process where, counter measures could be taken to avoid a radio link failure. Counter measures for self-healing was not covered in the thesis. This thesis only focuses on phase one and two of predicting a radio link failure.

Popular Science Summary

Today with the increasing use of cellular networks, there's an expected sharp increase in network traffic. An interesting fact is that phone calls are getting longer. Call drops i.e. calls that are "dropped" (terminated) without any of the parties intentionally interrupting the calls are commonly experienced in networks. Avoiding call drops helps improve quality of service. Self organized network is a technology which helps simplify complex networks and ensures better service. Machine learning plays a key role in it. With human limitations in place, machine learning, a subset of artificial intelligence helps to visualize behaviours and see patterns beyond human recognition.

In the thesis, machine learning approaches are used to predict a radio link failure. A network may transfer (hand over), a user connection from the current cell to another cell, so that the user terminal will experience higher signal strength. This process is called a handover and an interruption here causes radio link failure. So it is important to minimize radio link failure in cellular networks for a better user experience. Deteriorating signal strength is one of the key indicators investigated in the thesis in determining a radio link failure. Real time data from live networks may seem unpredictable but machine learning approaches discussed in this thesis helps predict whether a connection would experience a link failure or not. There are a number of parameters causing a link failure, these parameters are inputs to the machine learning models and the result is a classified output which is - link failure or not. This thesis also involves a time series analysis which helps forecast a radio link failure much ahead of time. If a failure is forecast, it is possible to take counter action and prevent the same, this forms parts of the self healing process. This thesis mainly focuses on the classification and forecast of a radio link failure.

Acknowledgement

Firstly, I would like to convey my deepest gratitude to Ericsson AB and Bhavin Patel, my line manager at Ericsson for letting me write my thesis there. The inspiring environment and constant professional support from the systems team, especially Joel, Per, Niclas, Jonas has helped me accomplish this assignment.

I'm ineffably indebted to Fredrik Tufvesson and Abhishek Saurabh, my supervisors at LTH and Ericsson respectively for their constant valuable guidance, feedback throughout the thesis.

Last but not the least I would like to thank my family, parents Vijayalakshmi and Ramaswamy for their prayers and giving me this opportunity to pursue a master's. Further, I would also like to thank all my thesis colleagues for the refreshing coffee breaks and keeping me in good company.

- Janani Mudaliyar

Table of Contents

1 Introduction **1**
1.1 Background and motivation 1
1.2 Purpose and aims 1
1.3 Methodology 2
1.4 Limitations 2
1.5 Related work 2
1.6 Challenges 4

2 Technical background **5**
2.1 Beam Management 5
2.2 An overview of the simulator 7
2.3 RLF triggers as observed on the simulator 7
2.4 User scenarios 8
2.5 Log specifications 11

3 Machine Learning Approach - Phase 1 **15**
3.1 Introduction to Machine learning 15
3.2 Unsupervised Learning 15
3.3 Reinforcement learning 15
3.4 Supervised Learning 16
3.5 Choosing an apt ML model 16
3.6 Artificial Neural Networks 16
3.7 Multi-layer Perceptrons 17
3.8 Logistic regression 18

4 Results and analysis - Phase 1 **21**
4.1 Neural networks 21
4.2 Logistic regression 23

5 Machine Learning Approach - Phase 2 **25**

5.1 Time series analysis 25
5.2 Seasonal autoregressive integrated moving average 26
5.3 Long Short Term Memory 27

6 Results and analysis - Phase 2 **29**
6.1 Seasonal autoregressive integrated moving average 29
6.2 Long Short Term Memory 31

7 Conclusions and future work **33**

References **35**

List of Figures

1.1 Block diagram of the two phases in this thesis 3

2.1 Beam management 6
2.2 User scenario - SCM 9
2.3 User scenario - circular mover 10
2.4 User scenario - straight mover 10
2.5 A snip of the log files showing 12 "beamIndex" i.e. wide beams 12
2.6 RSRPs of different SSB beams when RLF occured 13
2.7 RSRP for the SCM scenario 13
2.8 RSRP for the circular mover scenario 14
2.9 RSRP for the straight mover scenario 14

3.1 Block scheme of supervised learning process, showing the output dataset being provided to the input dataset for training purpose 16
3.2 Logistic function graph on the x-interval (-10,10) 19

4.1 An example of the rolling window with window size 3 21
4.2 NN model topology with two hidden layers 22
4.3 NN classification model with 97.5 % accuracy 22
4.4 NN model confusion matrix with true and false positive values (test data) 23
4.5 Logistic regression model with 94% accuracy 24
4.6 Logistic regression model confusion matrix with true and false positive rates (test data) 24

5.1 LSTM 28

6.1 Example of non-stationary process in the straight mover scenario 29
6.2 Example of ACF and PACF for the straight mover scenario . . 30

6.3 Forecast results for a straight mover scenario 30
6.4 Forecast results for a straight mover scenario 31

Chapter 1

Introduction

1.1 Background and motivation

By the end of 2019, there were over 13 million 5G subscribers globally and by the end of Q1 2020, about 10 percent of 4G operators launched 5G networks [1]. 5G is not just another faster G but it is a platform for next generation technology standard for cellular networks. One important factor is to not only focus on use cases but also zoom in on relevant use places. Ensuring coverage would mean avoiding Radio Link Failure's (RLF's). Establishing good coverage also helps innovate network deployment strategy and it's then possible to build networks with precision. Customer satisfaction is the result of a well planned deployment.

Technology is the key to global economic and sustainable development. As a part of current trends with a 5G boost, the information and communications technology industry is clearly driven by intelligent algorithms which helps visualize beyond ordinary solutions. 5G is the theme for 2020 with a clear ramp up of device numbers, emerging early use cases and new technologies such as ML, gain traction.

Arthur Samuel, the pioneer of ML defines machine learning as an application of artificial intelligence that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. There has been research where ML modeling is chosen for data centric networks [2]. However, there is no extensive research done on whether ML would be a good choice in predicting and healing a call drop in comparison to the traditionally prevalent solutions. Hence we aim to investigate this in the thesis.

1.2 Purpose and aims

With millimeter wave (mmW) frequencies, the problem of RLF is more prominent. This is because the high frequencies do not penetrate surfaces easily making it prone to link failures.

Initially the thesis aims at finding parameters which characterize a link connection, for e.g. the Reference Signal Received Power (RSRP), block error rate and similar parameters. The aim with this master's thesis work is then to investigate how well ML techniques can be applied to predict and heal a call drop in packet switched networks. Ultimately, predicting link failures would help build stable, high quality networks, as expected from 5G.

1.3 Methodology

Trouble Reports (TRs) indicating radio link failures were evaluated in comparison to normal reports with no reported radio link failure. Evaluating TRs was found to be insufficient to observe transitions in a call leading to a radio link failure. So, there was a need to run simulations. The simulations were performed using an Ericsson simulator named Redhawk, which simulates radio link failure scenarios. Data for ML were collected by running these simulations.

The common set of characteristic parameter(s) of radio link failure's were found from the logs. ML was then applied to the selected parameter(s) to observe how well ML classifies a bad link with failure from a good link. This classification forms the first phase of the thesis. The second phase was forecasting a radio link failure. In the thesis, a time series analysis was performed on the chosen parameter(s) to forecast a RLF. The time series model would forecast a link failure much ahead of time thus helping to prepare for poor network conditions. Refer figure 1.1 for the block diagram denoting the two phases.

1.4 Limitations

Once RLFs are forecast, fast recovery measures would help in overcoming the RLFs. A possible third phase should include a ML algorithm, which would help "self-heal" RLFs. The ML algorithms would be trained to avoid bad link scenarios and this aids in self-healing. The thesis is limited to phase one and two alone. The third phase would be more of an ambitious aspect of the thesis but is left for future work.

1.5 Related work

The author explored several literature sources related to machine learning and RLF. Increased call drops were observed in the summer months [3].

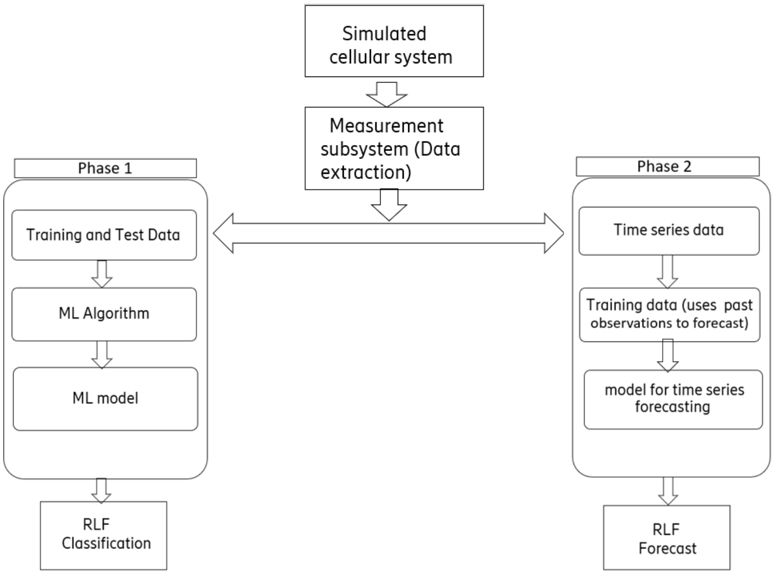

Figure 1.1: Block diagram of the two phases in this thesis

Time series models were used to study the behaviour of radio links during summer. This report also involved a detailed study of the dynamics that affect the performance of mobile communication networks (KPIs for example). The authors of the report found that RSRP was one of the most important feature for explaining variation in Dropped Call Rate (DCR) over time. The authors however collected data from public data repositories and not from live networks or simulations. This makes the results less effective to see patterns in real-time network.

In another report, deep learning based link failure mitigation techniques and ML models were proposed [4]. Signal level and quality measurements like RSRP, reference signal received quality, signal to interference plus noise ratio were the parameters investigated. Advanced ML models were used to classify a bad link from a good link. The authors however ran a simulation which involved only one scenario with a user moving along a straight line, while in reality it is possible to have multiple scenarios.

Tutorials on beam management in NR along with beam management procedures like P1, P2,P3 have been described in the references [5][6]. NR fundamentals required for the thesis was found in the book [7]. Detailed study of Neural network fundamentals can be found in [8][9]. These article provide sufficient information for understanding the various algorithms used to train neural networks and optimization techniques.

1.6 Challenges

One of the main challenges was to get relevant RLF occurring logs (containing relevant parameters) which would be the data for the ML model to train on. Another challenge was to choose an apt ML algorithm that meets our purpose. Live network data is currently not used for real-time predictions. The areas of 5G RLF predictions using ML remain unexplored and this means that there is a dearth of existing techniques to build upon.

Chapter 2

Technical background

2.1 Beam Management

To interpret parameters that characterize a link connection it is important to understand beam management. The ultimate task of beam management is, to establish and retain a suitable beam pair, i.e., a transmitter-side beam direction and a corresponding receiver-side beam direction that together provides good connectivity [7]. In NR, the signal reporting are based on either Synchronization Signal (SS) or Channel State Information (CSI) reference signals. Many radio products have a fixed number of wide beams, these wide beams also include a fixed number of narrow beams [6].

P1, P2, P3 are important procedures designed for beam management. The User Equipment (UE) measurements from these procedures are relevant to the thesis. UE is a mobile device used by an end user to communicate. gNodeB (gNB) is a terminology used in 5G for base transceiver station. P1 is the initial beam selection process. The gNB broadcasts SSBs [6]. The SS Block (SSB) beams are also called wide beams. The UE measures and selects a suitable beam and reports it to gNB.

P2 involves beam refinement and beam tracking for transmitter (gNB). In beam refinement the gNB transmits CSI reference signals (CSI-RS). The UE measures them and reports Reference Signal Received Power (RSRP) for CSI-RS. The gNB then selects a narrow beam and starts transmitting to the UE on the selected narrow beam. Beam tracking involves switching narrow beam as the UE moves. After beam refinement if the preferred CSI-RS resource changes, the gNB updates the selected beam and starts transmitting to the UE on the updated beam. Beam tracking between wide beams: the gNB transmits SSBs and the UE measures the SSB and reports RSRP for the SSB.

P3 involves beam adjustment for receiver (UE). The gNB fixes a beam and a beam sweep is performed on the UE side on reference signals. The UE measures and selects the best beam and uses the best beam for subsequent receptions. However, the UE does not report back to the gNB unlike in P1 and P2 [6]. Refer figure 2.1 for beam management processes.

This thesis is limited to using SSB RSRP measurements.

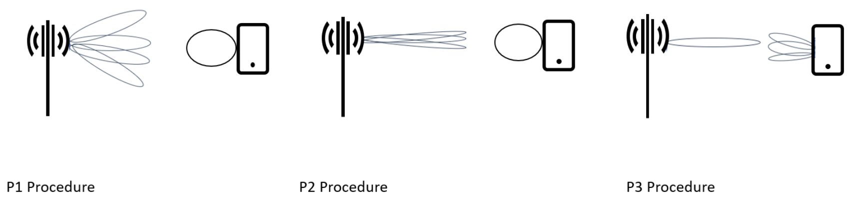

Figure 2.1: Beam management

2.1.1 Report configuration - Reference Signal Received Power (RSRP)

RSRP is the key indicator of the signal strength in 5G networks. The gNB transmits reference signals to the UE, the UE then measures the power in the received signal and reports back to the gNB. These reports of the received power are known as RSRP and are required when performing cell selection or handover. RSRPs are reported using the logarithmic power unit, dBm. The RSRP measurements are made on Cell-specific Reference Signal (CRS) in LTE (according to 3GPP TS 36.211). As mentioned in the above section, the RSRP reporting in NR are based on either SS or CSI-RS [7]. The CSI-RS measurements are made on the narrow beams where the traffic payload is transported.

2.1.2 Radio Link Control (RLC) retransmissions

The RLC protocol layer is a layer 2 protocol. It is located above the Medium Access Control (MAC) layer and below the Packet Data Convergence Protocol (PDCP) layer [8]. RLC, MAC and PDCP are user plane protocols. The data entity from/to a higher protocol layer is called Service Data Unit (SDU) and the entity from/to a lower protocol layer entity is called a Protocol Data Unit (PDU). RLC protocol delivers SDUs from PDCP layer to the RLC entity in the receiver. The RLC entity handles segmentation of SDUs, RLC retransmissions and duplicate removal [7]. In the following part, the focus is mainly on RLC retransmissions as it is used in the thesis.

Retransmission of missing PDUs is one of the main features of RLC. The RLC retransmission procedure ensures the delivery of error-free data to the higher layers [7]. This error-free delivery is possible because a sequence number is assigned to each SDU header. On evaluating the sequence numbers of the received PDUs, it is possible to detect the missing PDUs and request a retransmission [7].

RSRP and RLC retransmissions were the two input features for the machine learning models. The machine learning approaches used in this thesis are further described in chapter 3.

2.2 An overview of the simulator

New Radio (NR) services like low-latency and ultra reliable communication are emerging. To keep up with the requirements it is important to have good link connections. Call drops are usually caused by RLFs, which can be detected by the UE [9]. In principle, RLF occurs when a device moves out of coverage from a currently serving cell. In this case there's a need to re-establish the connection to a new cell or sometimes even a new carrier [7]. With ML, data is the key and like mentioned in the previous chapter, there was a need to run simulations to collect data and create a data set for ML. A data set is a term used to refer to a collection of data used for ML. The simulations were performed on an Ericsson simulator named Redhawk. The simulator was capable of generating high band traffic. The simulator already had a RLF occurring setup in place. The mobility enabled setup operated at mmW frequency (28 GHz), had a bandwidth of 200 MHz and had beam management. More changes were added in this basic setup to observe more RLF behaviours to build on.

2.3 RLF triggers as observed on the simulator

The Radio Resource Control (RRC) is a control plane protocol defined in 3GPP. One of the functions of RRC protocol is to control the RRC connection which is responsible for the communication between UE and gNB, along with setting up bearers and mobility [7]. Signaling Radio Bearers (SRBs) are used to transmit the RRC messages to the UE. The SRB is initially mapped to the common control channel during establishment of connection. Later, on establishing a connection, the SRB is mapped to the dedicated control channel[7]. Data Radio Bearer (DRB) is used to carry data related to Evolved Packet switched System (EPS) bearers. 'EPS Bearer' is a pipe line through which data traffic flows within EPS [10]. Proper SRB and DRB's RLC retransmission parameter settings can improve mobility performance [11].

In NR, the devices can be in three states viz. $RRC\_CONNECTED$, $RRC\_INACTIVE$, RRC_IDLE states. In the connected state, data transfer is possible as the RRC context is established, connection to the core network is also established. If the UE discontinues the RRC connection, there is no connection to the core network, the RRC context is not established, and the UE further progresses to RRC idle state [7]. In the RRC inactive state the RRC context is established and connection to the core network is also established. However no data transfer is possible. If a RLF occurs, the UE loses connection.

Listed below are the most common RLF triggers on the simulator.

- RLC delivery failure DRB - maximum number of transmissions reached.
- Delivery failure SRB - maximum number of transmissions reached.
- Random access channel failure - maximum number of preamble transmission attempts reached.
- Downlink out of synch detected in Layer 1.
- Handover procedure timed out.

The most commonly encountered RLF trigger in this thesis was : exceeding the maximum number of RLC retransmissions on DRB. The RRC connection reestablishment procedure gives the UE an option to try to re-connect if an RLF occurs. In a connected state, the device does not get to decide about handovers. It makes a decision based on the relative power of a measured SS block compared to the current cell, the device reports back the measurements to the network. The network then makes a decision as to whether or not the device is to handover to a new cell. This reporting is done using RRC signaling [7]. From the NR spec, the timer T310 was triggered everytime a link goes bad. In the simulator, the parameter named "ueRadioLinkFailureTimeT310" was set to 1000 ms. These are timers used by the UE in $RRC\_CONNECTED$, RRC_IDLE case. This timer played a key role in labelling the data set for ML. Details on the ML approaches used are described in the next chapter.

2.4 User scenarios

Inorder to collect data for ML, three different realistic user scenarios were created using simulations on Redhawk. For the sake of simplicity, the author chose to have a UE mounted on the centrally located base station. Refer figures 2.2, 2.3 and 2.4.

On the simulator, one of the parameters named "lossProbabilityFunction" gave an option to choose between different propagation models like Spatial Channel Model (SCM), line of sight model etc. It represents the path loss experienced by the signal between the base station and the UE. This function was set to "none" initially, but was changed to SCM later in the simulation so as to build the SCM user scenario. Following are the three distinct user scenario specifications.

1. SCM -$\pi/4$ mover

- Cell radius = 100 m.
- "lossProbabilityFunction" set to SCM.
- UE moves straight in an angle = -$\pi/4$ (with reference to the x-axis)
- Number of UE = 1.
- Number of sites = 7.
- Number of sectors per site = 3.

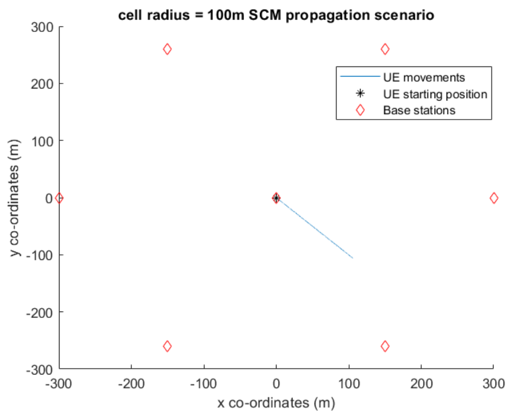

Figure 2.2: User scenario - SCM

2. Circular mover - low speed

- Cell radius = 225 m.
- Number of UE = 1.
- Number of sites = 7.
- Number of sectors per site = 3.
- UE moves in a circle.
- UE speed 0.833 m/s (moves slowly)

3. Straight mover - increased Speed

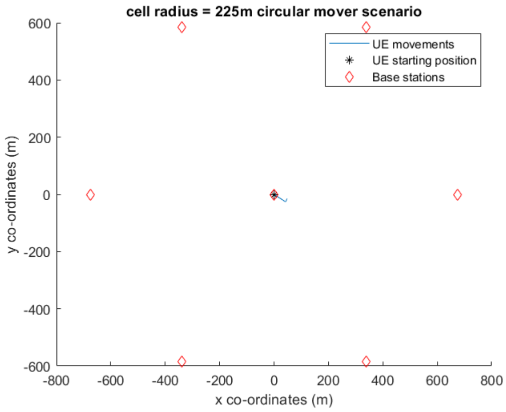

Figure 2.3: User scenario - circular mover

- Based on circular mover scenario.
- Number of UE = 1.
- Number of sites = 7.
- Number of sectors per site = 3.
- UE moves in a straight line along the x axis, positive direction.
- UE speed increased from 0.833 m/s to 10 m/s.

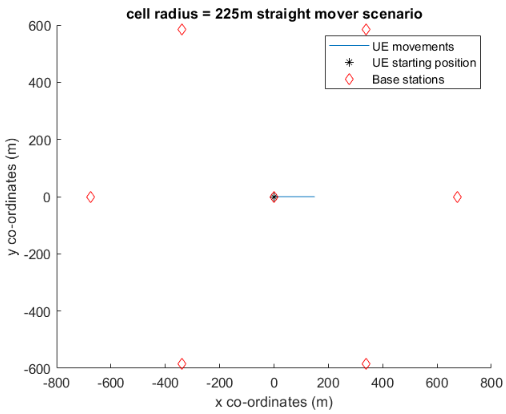

Figure 2.4: User scenario - straight mover

2.5 Log specifications

With the help of the simulator it was possible to have a deployment with 7 sites and 3 sectors per site. Thus a total of 21 cells were present in the data logs collected by running the above three scenarios, starting from "cellIndex" 0-20. Each of these cell indices had 12 "beamIndex" i.e. wide beams. Refer figure 2.5 The author was interested in analyzing the RSRP of the wide beams (SSB beam) that the UE measures as the best wide beam, on the current serving cell.

```
bestCellIndex=10008 bestRsrp=-133.643369654
bestCellIndex=20008 bestRsrp=-120.227237564
bestCellIndex=30009 bestRsrp=-136.871238916
bestCellIndex=40008 bestRsrp=-140.605935405
bestCellIndex=50008 bestRsrp=-140.879654833
bestCellIndex=60008 bestRsrp=-130.988817797
bestCellIndex=70008 bestRsrp=-129.727714408
bestCellIndex=80008 bestRsrp=-128.405478653
bestCellIndex=90008 bestRsrp=-127.185143557
bestCellIndex=100008 bestRsrp=-123.83088207
bestCellIndex=110009 bestRsrp=-134.40918892
bestCellIndex=120008 bestRsrp=-135.06183530
bestCellIndex=130005 bestRsrp=-123.58590773
bestCellIndex=140006 bestRsrp=-123.57899051
bestCellIndex=150008 bestRsrp=-105.91351338
bestCellIndex=160009 bestRsrp=-99.554357324
bestCellIndex=170007 bestRsrp=-102.63800615
bestCellIndex=180007 bestRsrp=-105.31767783
bestCellIndex=190009 bestRsrp=-144.23603464
bestCellIndex=200008 bestRsrp=-132.60355754
bestCellIndex=210008 bestRsrp=-133.97295598
ler cellIndex=0 beamIndex=10008 rsrp=-133.6
ler cellIndex=0 beamIndex=10007 rsrp=-134.7
ler cellIndex=0 beamIndex=10011 rsrp=-137.2
ler cellIndex=0 beamIndex=10009 rsrp=-138.7
ler cellIndex=0 beamIndex=10010 rsrp=-139.4
ler cellIndex=0 beamIndex=10004 rsrp=-141.9
ler cellIndex=0 beamIndex=10005 rsrp=-142.4
ler cellIndex=0 beamIndex=10006 rsrp=-143.2
ler cellIndex=0 beamIndex=10012 rsrp=-144.8
ler cellIndex=0 beamIndex=10002 rsrp=-145.2
ler cellIndex=0 beamIndex=10001 rsrp=-147.0
ler cellIndex=0 beamIndex=10003 rsrp=-152.9
ler cellIndex=1 beamIndex=20008 rsrp=-120.2
ler cellIndex=1 beamIndex=20009 rsrp=-121.9
ler cellIndex=1 beamIndex=20007 rsrp=-123.7
ler cellIndex=1 beamIndex=20011 rsrp=-125.3
```

Figure 2.5: A snip of the log files showing 12 "beamIndex" i.e. wide beams

In the simulator, the “bestRsrpLog” was enabled, which reported only the best wide beam RSRP of each cell. The parameter "bestCellIndex" is made up of the cellId 1-21 and wide beam index 1-12. Thus there were 21 "bestRsrp" logs at every point in time the wide beams were measured. It was possible to keep track of which cell the UE was in and then choose the RSRP related to that cell. It is assumed that the UE was on the best wide beam.

In the log file for the SCM scenario, two RLFs occurred at 1.5 and 3.4 seconds. Figure 2.6 shows the RSRP of the best SSB beams when RLF occurred. The RLC retransmissions were usually followed by a RLF. In some cases, there are retransmissions but the link recovers and the retransmissions are not followed by RLFs.

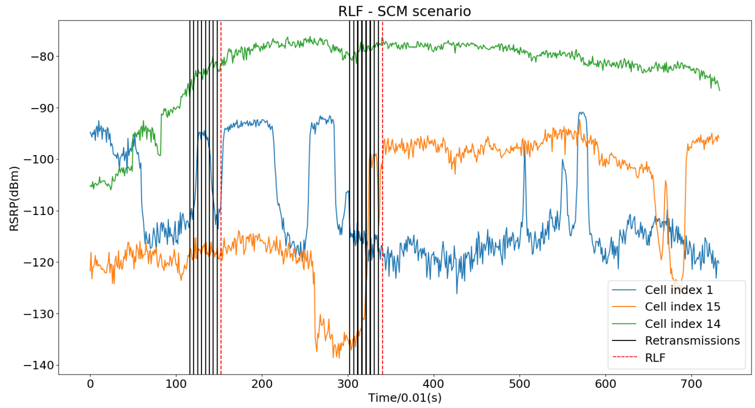

Figure 2.6: RSRPs of different SSB beams when RLF occured

RSRPs of the three user scenarios

On plotting the best wide beam RSRPs (see figures 2.7, 2.8, 2.9) of all the three user scenarios, the RSRP levels were initially low for about a second but eventually start improving. This is because the UEs were placed right on top of the central base stations as seen previously. In the circular mover scenario (figure 2.8) the UE moves with a very low speed hence there is little fluctuations in the power levels.

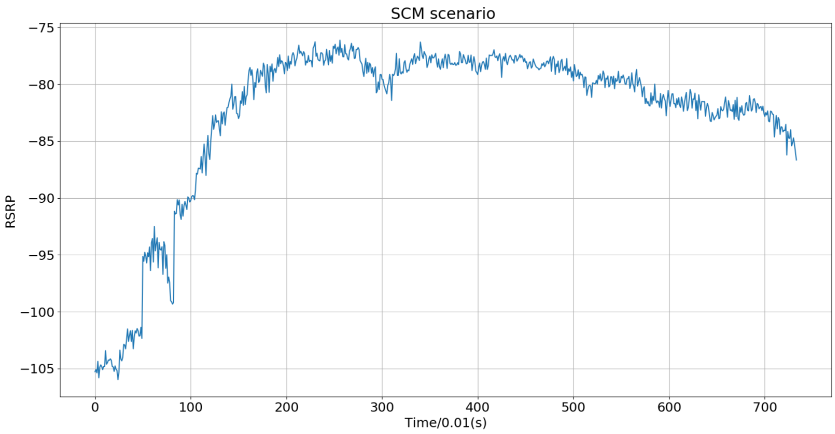

Figure 2.7: RSRP for the SCM scenario

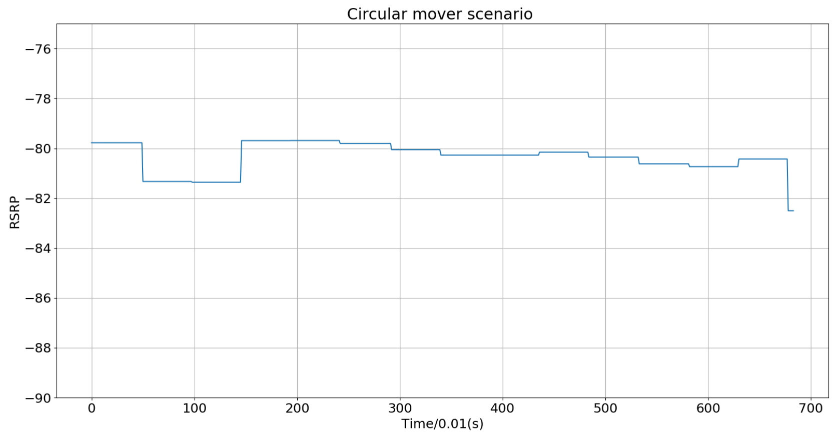

Figure 2.8: RSRP for the circular mover scenario

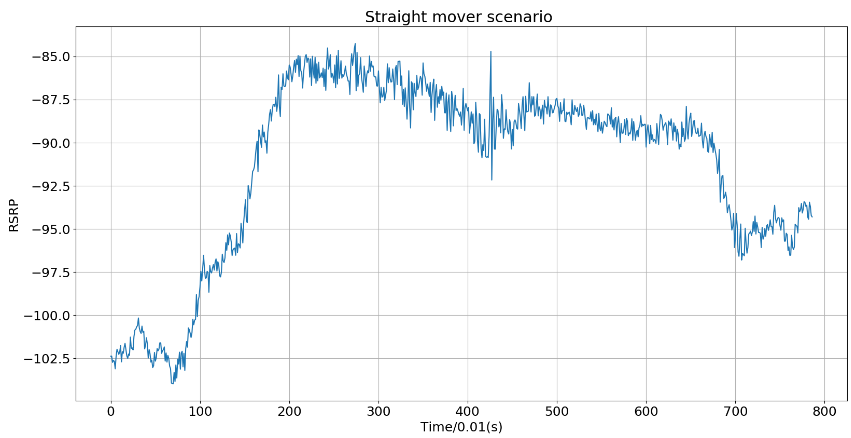

Figure 2.9: RSRP for the straight mover scenario

Chapter 3

Machine Learning Approach - Phase 1

3.1 Introduction to Machine learning

Machine Learning is an application of Artificial Intelligence (AI) for a future with 5G and beyond. It allows 5G networks to be more predictive in real-time. Machine learning promises to assist ICT developers to study traffic data, detect security threats and predict behaviour of networks. Because of this, ML enabled networks can become self-organizing, self-optimizing and self-healing. ML is broadly classified as unsupervised learning, reinforcement learning, and supervised learning.

3.2 Unsupervised Learning

In contrast to supervised learning, the unsupervised learning does not include pre-labeled outputs. This means there are no expected outputs or targets assigned. The model looks for patterns and learns from these patterns with minimum supervision. Hence the name *unsupervised learning*.

3.3 Reinforcement learning

Another branch of machine learning is the *Reinforcement Learning* (RL) which is seen as the science of optimal decision making. Similar to unsupervised learning, the reinforcement learning doesn't have a supervisor, but has a reward signal a.k.a feedback signal. The goal in RL is to maximize positive rewards through interaction between agents and environments [12]. In terms of RL, agent is represented as the brain who controls with the help of algorithms. An agent is able to take actions or influence its environment (data) and collect rewards. The agent gets both the observations and the reward coming in, thereby decision is made.

3.4 Supervised Learning

The goal of supervised learning is to learn and develop predictive models (rules) based on input and output relationship. Labeled data (usually actual outputs) are provided to help the learning process and enable a model to behave in a certain way, hence the name *supervised learning*. Refer figure 3.1. This thesis is based on supervised learning, as labeled data is used.

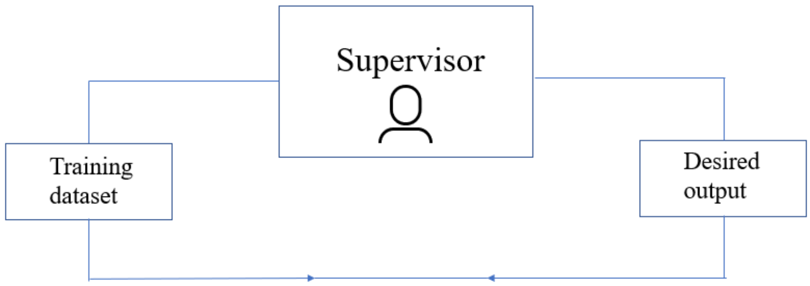

Figure 3.1: Block scheme of supervised learning process, showing the output dataset being provided to the input dataset for training purpose

3.5 Choosing an apt ML model

A significant finding in ML is that no single ML model is ideal or suitable for all problems [13]. The data is the key here and the performance of ML models also depend on a number of other parameters for instance, the use case, problem defined and the dimensions (number of input features) of the data set. Hence it is a good practice to compare the performance of different ML models to the most suitable one that fits the problem defined.

3.6 Artificial Neural Networks

There are three main types of Artificial Neural Networks (ANN), they are : Multi-layer Perceptrons (MLPs), Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). We now discuss MLP in detail as it is used in this thesis.

3.7 Multi-layer Perceptrons

The multi-layer perceptron model was used for binary classification in this thesis. Since there's no single ML model that is suitable for all problems, another supervised learning model namely logistic regression was investigated for comparison sake. Both the models are discussed further in this chapter. Multi-layer Perceptrons (MLPs) also known as deep neural networks are the most classical Neural Networks (NN). MLP is a supervised learning model which consists of an input layer, hidden layer(s) and output layer(s) connected by weighted connections.

The input is usually a vector multiplied by weights and added to a bias. Refer equation 3.1. Hidden layer(s) are the layers in between the input and output layers. Each of these layers consists of neurons or nodes. Typically, a neuron calculates the weighted average of its input, and this sum is passed through a nonlinear function, a.k.a activation function. The bias added to the input vector and weights, shifts the activation function by adding a constant to the input. Bias also makes it possible to fine-tune the output of the MLP.

Consider m as the number of dimensions for input. Given a set of features (input), $X = x_1, x_2, x_3 ... x_m$ and a target (output) $\hat{Y}$, MLP has the ability to learn the mapping between the output and the input. The model learns this mapping using an algorithm called back-propagation. Back-propagation adjusts the weights of the connections in a way to minimize the difference between actual output and desired output [14]. It also keeps on updating the weights repeatedly (learning process). As mentioned in the previous chapter, RSRP and RLC retransmissions were the two features (inputs) provided to the MLP model. MLP is suitable for a classification problem as in our case [15].

The hidden nodes in MLP are labeled as, $h = h_1^1, h_2^1, h_3^1 ... h_n^1$. These hidden nodes have superscript one as they represent the first hidden layer. i.e. h_1^1 represents the first node of the first hidden layer and so on. Similarly, the weight set for the first hidden layer is $W^1 = w_1^1, w_2^1 ... w_m^1$. As mentioned previously, the general equation is given by a dot product summation of the weights, input and adding a bias to it.

$$z = \sum_{i=1}^{m} w_i x_i + bias \tag{3.1}$$

Sigmoid activation was the chosen activation function for this thesis. It is defined as a smooth nonlinear function that is monotonically increasing and has an S shape [18]. The activation function is given by,

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{3.2}$$

The output of this activation function is in the range 0 to 1. This fulfills the requirement of binary classification and was hence the chosen activation function for this thesis. More about the sigmoid function is discussed in the next section. Further, z acts as an input to the activation function $\sigma(z)$, such that $h_1^1 = \sigma(z)$. Similarly it is possible to calculate the output for the second hidden node within h^1 i.e h_2^1 and so on. Once all outputs of the hidden layer(s) are calculated, they are used as inputs (call new input) to find $\hat{Y}$. The hidden layer weights are labeled as W^h. As seen before, the new input is represented as $h^1 = h_1^1, h_2^1, h_3^1..h_n^1$. So the final output is the dot product of the new input and hidden layer weights [16]. The general equation 3.1 with x_i as an input is now modified with the new input and can be written as follows.

$$z = \sum_{i=1}^{n} w_i^{h_1} h_i^1 + bias \tag{3.3}$$

From equation 3.3, the final output is:

$$\hat{Y} = f(z) \tag{3.4}$$

In short, each layer of MLP is found by calculating the values of the nodes in the previous layer and applying weights and bias. These values are then passed to the subsequent node and so on. It is also possible for a node to have several input weights [15]. Every node then applies an activation function to the input weights, creating a singular value for the given node. After the values in all the nodes (in a layer) have been calculated, the process repeats with the next set of weights [16].

The MLP was used in the phase one of the thesis to classify a good link from a radio link failure [17]. The performances, the prediction accuracy of each of this model are described further in this chapter. This here completes the method used in phase one of the thesis to classify a RLF.

3.8 Logistic regression

Logistic regression is commonly used in (binary) classification problems as in our case. In logistic regression, the target (dependent variable) must be a binary variable containing 1 (good radio link) or 0 (bad link) i.e. the logistic function $f(x)$ has two results, either 0 or 1. The logistic function takes any real input x, where $x \in R$, and outputs any number in the range 0 to 1. The logistic function is a S-shaped curve given by:

$$f(x) = \frac{L}{1 + e^{-k(x-x_0)}} \tag{3.5}$$

Refer figure 3.2. There are three important parameters to note. The parameter L is the maximum value of $f(x)$, it is set to 1 (standard). The parameter x_0 is the (x value) curve's midpoint. The parameter k is the steepness (logistic growth) of the curve [18]. Thus, x can be any variable and determining x is important in understanding logistic regression.

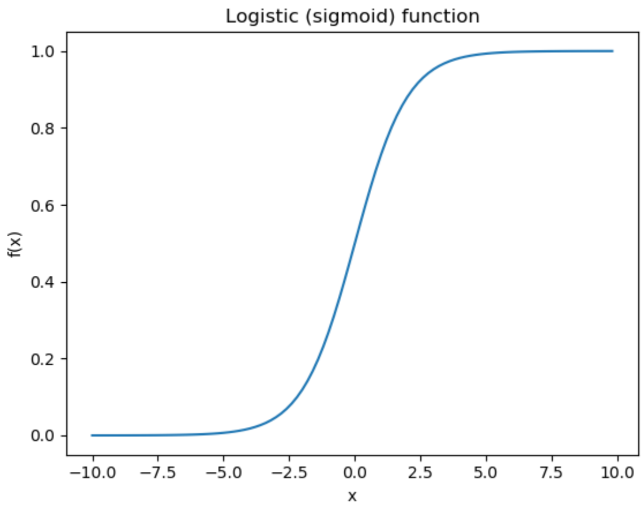

Figure 3.2: Logistic function graph on the x-interval (-10,10)

Both the models, the MLP and logistic regression were investigated in the phase one of the thesis to classify a bad link with failure from a good link without failure. The performances, the prediction accuracy of each of these models and the best pick amongst these two models are further stated in the next chapter.

Chapter 4

Results and analysis - Phase 1

4.1 Neural networks

The inputs to the neural network were provided using the rolling window algorithm. Initially a window of 2 was chosen, the input rows looked like {RSRPt0, RSRPt1, RETXt0, RETXt1}, {RSRPt1, RSRPt2, RETXt1, RETXt2}, {RSRPt2, RSRPt3, RETXt2, RETXt3} and so on. Here RSRPt0 and RSRPt1 are the first and second RSRPs' from the data set respectively. This is followed by RETXt0 and RETXt1 which are the first and second RLC retransmissions from the data set. Refer figure 4.1 for an example of rolling window.

Different window sizes were investigated and compared. A window of size 100 was used in the thesis for the two input features. With a window size of 100, the number of input nodes expand to 200 instead of 4 (when using the window size 2). Thus sampling the data set by rolling window facilitates the generation of more training samples.

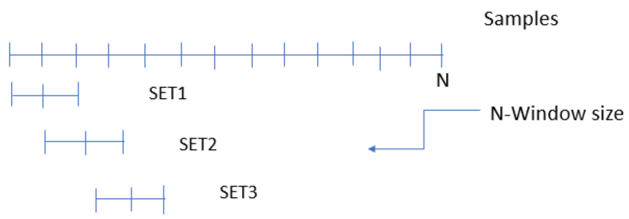

Figure 4.1: An example of the rolling window with window size 3

Each window is a training example for the neural network model to train on. The topology of the neural network used in this thesis was similar to figure 4.2. A total of 32 neurons in each of the 2 hidden layers were used.

The overall classification accuracy using NN was 97.5%. Refer figure 4.3. For the NN a total of 585 samples were chosen as unseen data (test data). From the confusion matrix in figure 4.4, True Negative (TN) + True

Positive (TP)/ False Negative (FN) + False Positive (FP) + TN + TP gives the total accuracy percent. From the confusion matrix, it is seen that all 91 samples are true negative i.e. correctly classified as having RLF. Out of the 585 samples, 97.1% of them were correctly classified as not having a RLF and 100% i.e. all of the data was correctly classified for having a RLF as seen from the confusion matrix.

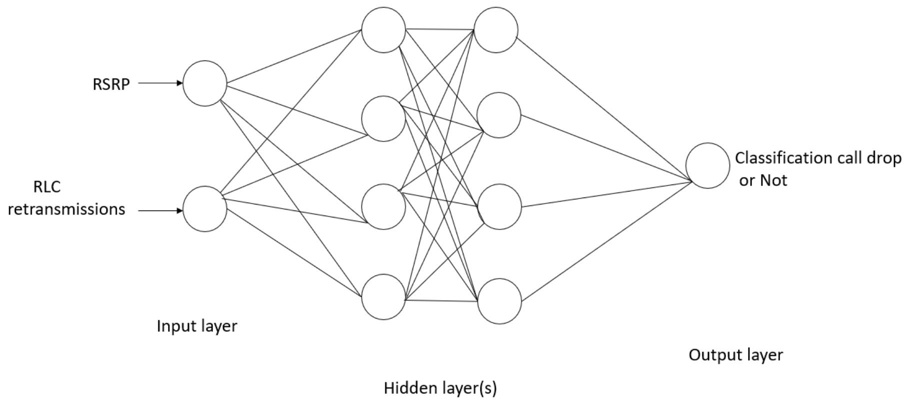

Figure 4.2: NN model topology with two hidden layers

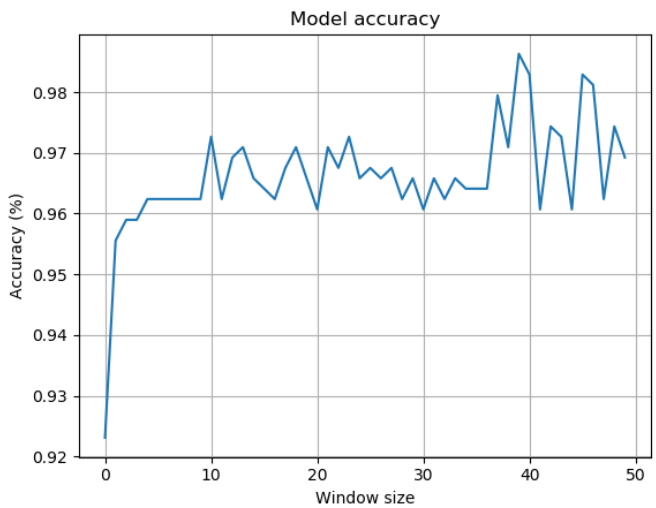

Figure 4.3: NN classification model with 97.5 % accuracy

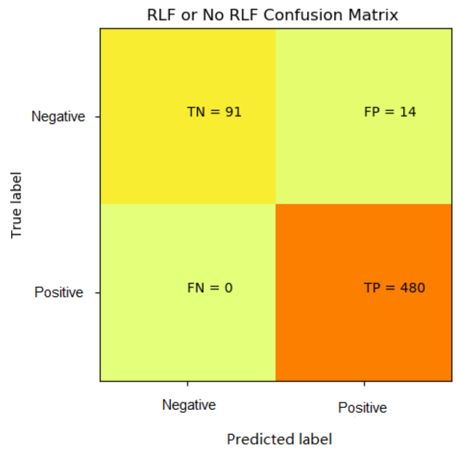

Figure 4.4: NN model confusion matrix with true and false positive values (test data)

4.2 Logistic regression

Similar to the NN, the inputs to the logistic regression model were provided using the rolling window algorithm, with a window size of 100. The overall classification accuracy using logistic regression was 94%. Refer figure 4.5. A total of 975 samples were chosen as test (unseen) data. Out of these, 96.2% were correctly classified as not having RLF and 88.8% were correctly classified as having RLF. Refer the confusion matrix in figure 4.6.

With the data set used in this thesis, the performance of NN model outshines the performance of logistic regression. This is because deep neural networks have the ability to capture and learn complex relations between the inputs much better than logistic regression. As data sets have non linearities involved, NN can deal with them easily whereas logistic regression fails to do so. But, on the other hand NN may be subjected to overfitting due to the use of hidden layers. Again, there are regularization algorithms to avoid overfitting.

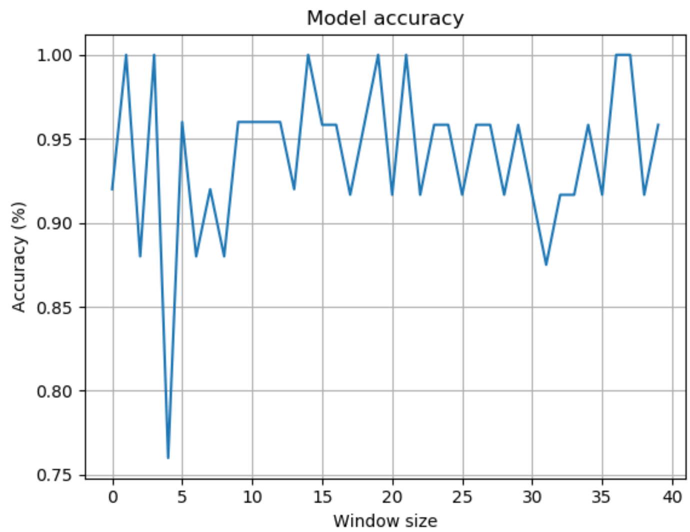

Figure 4.5: Logistic regression model with 94% accuracy

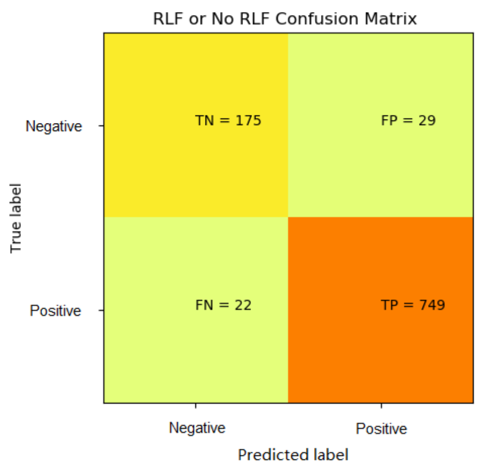

Figure 4.6: Logistic regression model confusion matrix with true and false positive rates (test data)

Chapter 5

Machine Learning Approach - Phase 2

5.1 Time series analysis

Time Series Analysis (TSA) is used for analyzing time series data i.e. data ordered in a sequence in time. This helps in extracting useful statistics and helps study other characteristics of the data [19]. In time series analysis, the time is the input on the x-axis (independent variable) and the goal is to forecast future values based on previous time series observations for e.g. stock market forecasting.

Time series analysis were used for the phase two part of this thesis to forecast the RSRP. There are two important aspects in Time Series (TS): stationarity and autocorrelation in the target variable.

5.1.1 Stationarity

A time series is said to be stationary if it has constant mean and variance, which do not change over time. There are two main reasons which make a TS non stationary, they are the seasonality and the trend. Seasonality is the periodic (seasonal) variations in the observations. For e.g. increase in foliage each year during autumn. Trends denote either an increasing or decreasing tendency of a TS data over time. Also, unlike seasonality, trends do not repeat periodically.

To check if a TS is stationary or not, Dickey-Fuller test is performed on the TS dataset. This test either supports or rejects the null hypothesis. The null hypothesis states that if $p > 0$, the null hypothesis is supported, the time series has a unit root and thus the process is non-stationary. If $p = 0$, there is strong evidence against null hypothesis and the null hypothesis is rejected. The process is then considered stationary as the data has no unit root.

The TS is further made stationary by taking a "first difference" i.e. shifting the TS and then checking the stationarity again. If the TS still supports null hypothesis, the process is repeated till the TS gets stationary.

5.2 Seasonal autoregressive integrated moving average

Seasonal Auto Regressive Integrated Moving Average (SARIMA) is a model that can fit any TS portraying non-stationarity and seasonality. It consists of AutoRegressive (AR) and Moving Average (MA) model in combination with order of integration (I).

For the MA(q) model, the parameter q represents the largest lag after which other lags are insignificant [20]. The parameter q can be found from the Auto Correlation Function (ACF) plot. To define ACF, let $\{y_t\}$ denote a time series given by:

$$\{y_t\} = \{...y_{t-1}, y_t, y_{t+1}, ...\} \tag{5.1}$$

The time series $\{y_t\}$ is (covariance) stationary if,

$$E[y_t] = \mu \text{ for all t} \tag{5.2}$$

$$cov(y_t, y_{t-j}) = E[(y_t - \mu)(y_{t-j} - \mu)] = \gamma_j \text{ for all t and j} \tag{5.3}$$

In equation 5.3, γ_j is called the j^{th} order or lag j autocovariance of $\{y_t\}$. The plot of γ_j against j is called the autocovariance function [21]. The autocorrelations (similarity between the observed samples as a function of time lags between them) of $\{y_t\}$ are given as,

$$\rho_j = \frac{cov(y_t, y_{t-j})}{\sqrt{var(y_t)var(y_{t-j})}} = \frac{\gamma_j}{\gamma_0} \tag{5.4}$$

From equation 5.4, the plot of ρ_j against j is called ACF. Refer chapter 6 for the ACF plot.

The AR(p) model is a regression of the time series, where the current value is based on the previous values with some time lags. It takes the parameter p which denotes the maximum value of lag after which all other lags are insignificant. The parameter p can be found from the Partial Auto Correlation Function (PACF) plot. The PACF is mainly associated with estimating the sequence of AR models [21], the AR(p) model is given by,

$$y_t - \mu = \phi_1(y_{t-1} - \mu) + ... + \phi_p(y_{t-p} - \mu) + \varepsilon_t \tag{5.5}$$

in lag operator notation this can be written as,

$$\phi(L)(y_t - \mu) = \varepsilon_t \tag{5.6}$$

In equation 5.6, $\phi(L) = 1 - \phi_1 L - ... - \phi_P L^p$. It can be shown that the AR(p) is stationary given the roots of the characteristic equation are outside the complex unit circle [21]. Refer equation 5.7.

$$\phi(z) = 1 - \phi_1 z - \phi_2 z^2 - ... - \phi_p z^p = 0 \tag{5.7}$$

The PACF is used to identify AR(p) models [22]. The estimation of the sequence of AR models can be done as,

$$\begin{aligned} z_t &= \phi_{11} z_{t-1} + \varepsilon_{1t} \\ z_t &= \phi_{21} z_{t-1} + \phi_{22} z_{t-2} + \varepsilon_{2t} \\ &\vdots \\ z_t &= \phi_{p1} z_{t-1} + \phi_{p2} z_{t-2} + ... + \phi_{pp} z_{t-p} + \varepsilon_{pt} \end{aligned} \tag{5.8}$$

here $z_t = y_{t-\mu}$ is the demeaned (mean-zero) data. The coefficients ϕ_{jj} for $j = 1...p$ (i.e., the last coefficients in each AR(p) model) are called the partial auto-correlation coefficients. For an AR(p) all of the first p partial autocorrelation coefficients are non-zero, and the rest are zero for $j > p$ [21]. It is possible to plot both ACF and PACF using the *plot_acf*() or *plot_pacf*() functions from the statsmodels library in Python.

The parameter d is the number of differences involved so as to make the TS stationary. Further, the seasonality component $s(P, D, Q, s)$ is added. Here s is the season's length i.e. the number of points where the signal is present as observed from ACF and PACF. The parameters P and Q are same as p and q mentioned previously but for the seasonal component s. Finally, D is the order of integration which is the number of differences required to remove seasonality from the TS [20]. Thus we get the SARIMA$(p, d, q)(P, D, Q, s)$ model.

The author wanted to check if there's seasonality or trends present in the TS that caused the non-stationarity and hence chose the SARIMA model since it is known to capture seasonality well.

5.3 Long Short Term Memory

LSTM is an artificial Recurrent Neural Network (RNN) which is commonly used for TSA. LSTM is capable of learning past data sequences (long term memory). Similar to MLP, these models train using the back-propagation algorithm. The architecture of LSTM comprises of three gates [23]:

5.3.1 Input gate

The function of this gate is to determine the input sequences which would be used to modify the memory. Sigmoid activation function chooses which values (0,1) can be let through the gates [23]. The tanh function prioritises

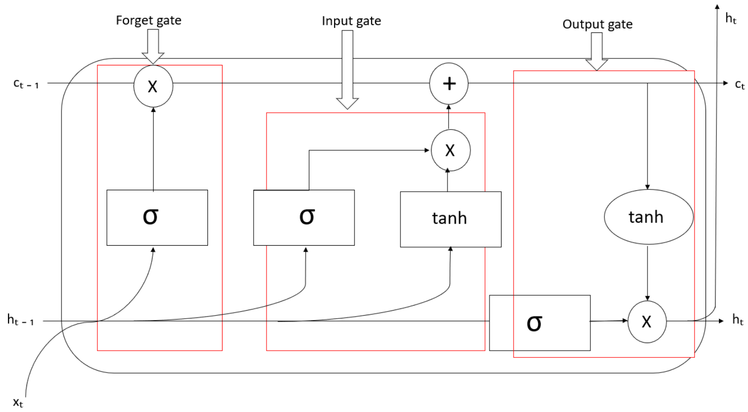

Figure 5.1: LSTM

these values, which are passed based on the weightage (ranging from -1 to 1). Refer figure 5.1.

$$
\begin{aligned}
i_t &= \sigma(W_i[h_{t-1}, x_t] + b_i) \\
C_t &= \tanh(W_c[h_{t-1}, x_t] + b_c)
\end{aligned} \tag{5.9}
$$

5.3.2 Forget gate

The sigmoid function checks the previous state (h_{t-1}) and the input(X_t) and outputs a number between 0 ("forgets" this) and 1 (keeps this) for each number in the cell state (c_{t-1})

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{5.10}$$

5.3.3 Output gate

The output is decided by the input and the memory of the block. The tanh function is multiplied with the output of Sigmoid.

$$
\begin{aligned}
o_t &= \sigma(W_o[h_{t-1}, x_t] + b_o) \\
h_t &= o_t \tanh(c_t)
\end{aligned} \tag{5.11}
$$

Chapter 6

Results and analysis - Phase 2

6.1 Seasonal autoregressive integrated moving average

For the straight mover scenario, the augmented dickey fuller test p value was found to be 0.244 as seen in figure 6.1. This supports null hypothesis and indicates non stationarity.

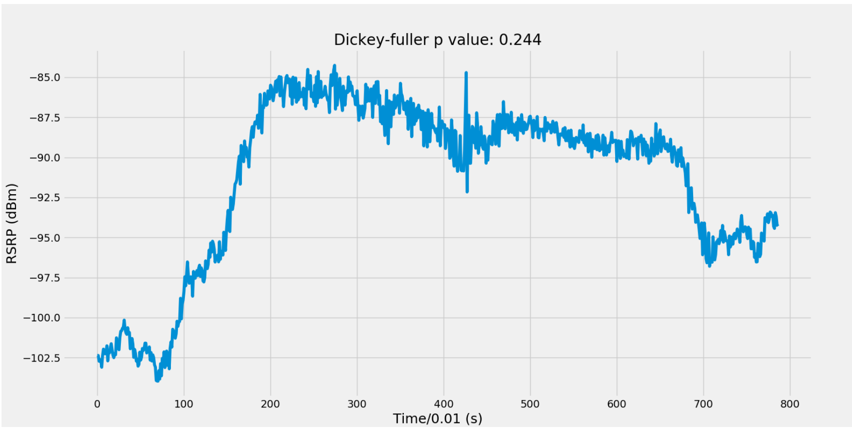

Figure 6.1: Example of non-stationary process in the straight mover scenario

For the SARIMA model, the value for p is determined by observing the Partial Auto Correlation Function (PACF) plot, which gives $p = 4$. The value of q can be found from the Auto Correlation Function (ACF) plot, which gives $q = 1$. Refer figure 6.2 for the PACF and ACF plots. The parameter d is the number of differences to make the TS stationary. The decaying plots of ACF and PACF denote that the TS has been made stationary.

The forecasted RSRP at 9 seconds is about -92.2 dBm. Refer figure 6.3. The Mean Absolute Percent Error (MAPE) is a measure used to find the accuracy of a forecast system. It is calculated as :

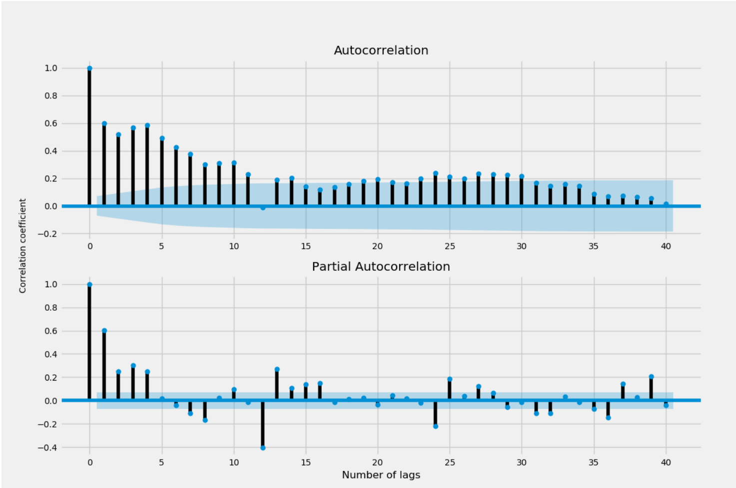

Figure 6.2: Example of ACF and PACF for the straight mover scenario

$$M = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right| \tag{6.1}$$

Where A_t is the actual value and F_t is the forecast value [22]. In this calculation, the absolute value is summed for every forecasted point in time and divided by n, which is the number of fitted points. Using SARIMA the MAPE is 2.43%

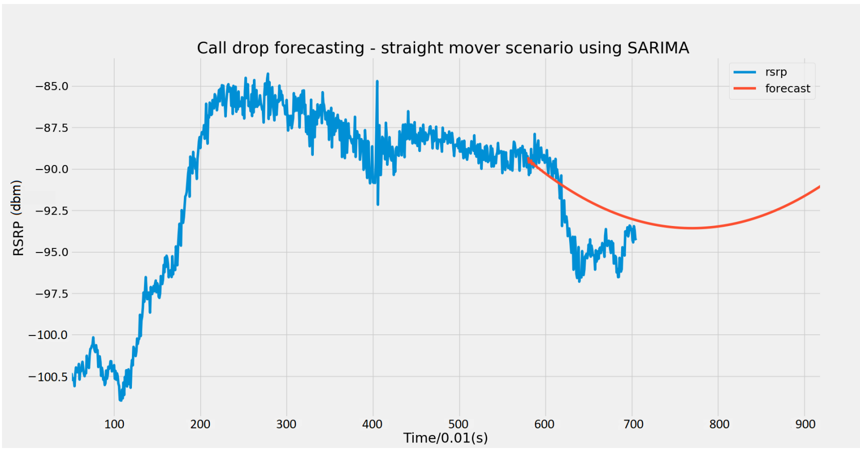

Figure 6.3: Forecast results for a straight mover scenario

6.2 Long Short Term Memory

The forecasted RSRP at time 10 seconds is about -93.5 dBm. The MAPE of predictions were found to be 2.81%. Thus the performance of SARIMA outshines the performance of LSTM. Though LSTM learns better, the data set involves some trends and seasons which the SARIMA seems to capture well. In the straight mover scenario the RSRP has a seasonal pattern occurring which has been captured better by SARIMA than LSTM.

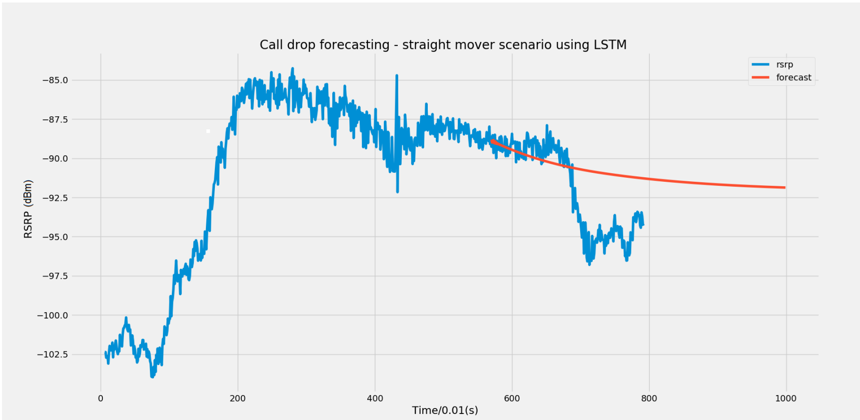

Figure 6.4: Forecast results for a straight mover scenario

Chapter 7

Conclusions and future work

This thesis motivated the need for investigation within the areas of RLF in 5G networks. The RLF problems are more prominent at mmWave frequencies. This is because the high frequency waves do not penetrate surfaces easily. As call drops are commonly caused by RLFs, they can be a major problem with 5G networks hence it is important to focus on getting high coverage in important spots. Ensuring good coverage would imply overcoming RLF.

This thesis falls under the umbrella of SON, where self healing is a functionality of SON. The aim with this thesis was to investigate how well ML can be used to classify and forecast a RLF ahead in time. This forms stepping stones for self-healing. In the networks of the future, such RLF's are expected to self heal. This can be done by either avoiding interference by applying a beam switch locally or by using an antenna tilt. The angle for the antenna tilt can be predicted by using advanced ML algorithms.

In this thesis, the author explored the two supervised learning algorithms namely the NN and logistic regression. It is however possible to investigate more ML algorithms and models to observe the behaviors. The drawback now is big NN and large storage requirements. Reducing the window size, means also shrinking the NN. The drawback is that storing a long window size per UE requires loads of memory. If that can be shrunk it is a huge gain. Especially if the performance of the NN is retained. To take one step further, it would be a good idea to consider evaluating either less or more neurons to see if it is possible to optimize inference but at the same time retain performance. Here, inference is the process of using a trained ML algorithm to make predictions.

With ML, data is the key and this thesis motivates the need for collecting relevant data by running Redhawk simulations. More scenarios can be generated by running these simulations and a data set with a combination of several of such scenarios can be created. This is because in reality there are several such scenarios occurring together at the same time.

RSRP and RLC retransmissions were the parameters investigated as triggers for a RLF as part of this thesis. More parameters like the BLER, HARQ can be explored and intelligent machine learning algorithms that can detect and report deviations from such ideal parameters can be used.

References

[1] C. Eklöf, J. Muller, and R. Zhohov, "The key to mobility robustness in 5G networks," Ericsson.com, May 21,2020.https://www.ericsson.com/en/blog/2020/5/the-key-to-mobility-robustness-5g-networks (accessed Jun. 12, 2020)

[2] T. S. Buda et al., "Can machine learning aid in delivering new use cases and scenarios in 5G?," in NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, Apr. 2016, pp. 1279–1284, doi: 10.1109/NOMS.2016.7503003. https://ieeexplore.ieee.org/abstract/document/7503003

[3] S. Vincenzi, "What causes increased dropped calls during summer?," Ericsson.com, May 15, 2020. https://www.ericsson.com/en/blog/2020/5/what-causes-increased-dropped-calls-during-summer (accessed Jun. 11, 2020).

[4] S. Khunteta and A. K. R. Chavva, "Deep Learning Based Link Failure Mitigation," in 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Dec. 2017, pp. 806–811, doi: 10.1109/ICMLA.2017.00-58.

[5] M. Giordani, M. Polese, A. Roy, D. Castor, and M. Zorzi, "A Tutorial on Beam Management for 3GPP NR at mmWave Frequencies," IEEE Commun. Surv. Tutorials, vol. 21, no. 1, pp. 173–196, 2019, doi: 10.1109/COMST.2018.2869411.

[6] N. Kumar, "Lean Beam Management for New Radio," p. 83.

[7] Dahlman, Erik, Parkvall, Stefan, Sköld, Johan, "5G NR: The Next Generation Wireless Access Technology." https://eds-b-ebscohost-com.ludwig.lub.lu.se/eds/

ebookviewer/ebook/bmxlYmtfXzE3MDMzNDVfX0FO0?sid=2b01ead4-68c4-4108-a233-6aa92187ab52@pdc-v-sessmgr05&vid=0&format=EB&rid=1 (accessed Aug. 03, 2020).

[8] "Radio Link Control," Wikipedia. Nov. 05, 2020, Accessed: Nov. 17, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Radio_Link_Control&oldid=987151704.

[9] X. Zhang, "Drop Call Optimization," LTE Optimization Engineering Handbook, pp. 216–243, 2017, doi: 10.1002/9781119158981.ch6. https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781119158981.ch6

[10] "ShareTechnote." http://sharetechnote.com/html/Handbook_LTE_EPS_Bearer.html(accessedNov.23,2020).

[11] "Radio Bearer - an overview — ScienceDirect Topics." https://www.sciencedirect.com/topics/engineering/radio-bearer(accessedAug.06,2020).

[12] G. Fahlén and J. Bill, "Machine Learning Technique for Beam Management in 5G NR RAN at mmWave Frequencies," p. 60. http://lup.lub.lu.se/luur/download?func=downloadFile&recordOId=9003743&fileOId=9004396

[13]]K. Tugrul, "Iteratively Finding a Good Machine Learning Model," Medium, Mar. 24, 2018. https://towardsdatascience.com/iteratively-finding-a-good-machine-learning-model-3eb5454208b1(accessedNov.19,2020).

[14] N. Kang, "Multi-Layer Neural Networks with Sigmoid Function— Deep Learning for Rookies (2)," Medium, Feb. 04, 2019. https://towardsdatascience.com/multi-layer-neural-networks-with-sigmoid-function

[15] "Neural Networks for Signal Processing." http://www.cnel.ufl.edu/courses/EEL6814/EEL6814.php(accessedSep.25,2020).

[16] Austin, "Classify Sentences via a Multilayer Perceptron (MLP)," Austin G. Walters, Jan. 03, 2019. https://austingwalters.com/classify-sentences-via-a-multilayer-perceptron-mlp/(accessedSep.22,2020).

[17] O. O. Erunkulu, E. N. Onwuka, O. Ugweje, and L. A. Ajao, "Prediction of Call Drops in GSM Network using Artificial Neural Network,"

Jurnal Teknologi dan Sistem Komputer, vol. 7, no. 1, pp. 38–46, Jan. 2019, doi: 10.14710/jtsiskom.7.1.2019.38-46.

[18] "Logistic function," Wikipedia. Sep. 09, 2020, Accessed: Sep. 28, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Logistic$_$function&oldid=977583860.

[19] "Time series," Wikipedia. Sep. 09, 2020, Accessed: Sep. 29, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Time_series&oldid=977544968.

[20] "The Complete Guide to Time Series Analysis and Forecasting — by Marco Peixeiro — Towards Data Science." https://towardsdatascience.com/the-complete-guide-to-time-series-analysis-and-forecasting-70d476bfe775(accessedOct.01,2020).

[21] "Time Series concepts." https://faculty.washington.edu/ezivot/MFTS1stEdition.htm(accessedNov.22,2020).

[22] "Partial autocorrelation function," Wikipedia. Jul. 15, 2020, Accessed: Nov. 22, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Partial_autocorrelation_function&oldid=967803127.

[23] A. Mittal, "Understanding RNN and LSTM," Medium, Oct. 12, 2019. https://towardsdatascience.com/understanding-rnn-and-lstm-f7cdf6dfc14e(accessedOct.02, 2020).

[24] "Mean absolute percentage error," Wikipedia. Nov. 18, 2020, Accessed: Nov. 19, 2020. [Online]. Available: https://en.wikipedia.org/wiki/Mean_absolute_percentage_error

ebookviewer/ebook/bmxlYmtfXzE3MDMzNDVfX0FO0?sid=2b01ead4-68c4-4108-a233-6aa92187ab52@pdc-v-sessmgr05&vid=0&format=EB&rid=1 (accessed Aug. 03, 2020).

[8] "Radio Link Control," Wikipedia. Nov. 05, 2020, Accessed: Nov. 17, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Radio_Link_Control&oldid=987151704.

[9] X. Zhang, "Drop Call Optimization," LTE Optimization Engineering Handbook, pp. 216–243, 2017, doi: 10.1002/9781119158981.ch6. https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781119158981.ch6

[10] "ShareTechnote." http://sharetechnote.com/html/Handbook_LTE_EPS_Bearer.html(accessedNov.23,2020).

[11] "Radio Bearer - an overview — ScienceDirect Topics." https://www.sciencedirect.com/topics/engineering/radio-bearer(accessedAug.06,2020).

[12] G. Fahlén and J. Bill, "Machine Learning Technique for Beam Management in 5G NR RAN at mmWave Frequencies," p. 60. http://lup.lub.lu.se/luur/download?func=downloadFile&recordOId=9003743&fileOId=9004396

[13]]K. Tugrul, "Iteratively Finding a Good Machine Learning Model," Medium, Mar. 24, 2018. https://towardsdatascience.com/iteratively-finding-a-good-machine-learning-model-3eb5454208b1(accessedNov.19,2020).

[14] N. Kang, "Multi-Layer Neural Networks with Sigmoid Function— Deep Learning for Rookies (2)," Medium, Feb. 04, 2019. https://towardsdatascience.com/multi-layer-neural-networks-with-sigmoid-function

[15] "Neural Networks for Signal Processing." http://www.cnel.ufl.edu/courses/EEL6814/EEL6814.php(accessedSep.25,2020).

[16] Austin, "Classify Sentences via a Multilayer Perceptron (MLP)," Austin G. Walters, Jan. 03, 2019. https://austingwalters.com/classify-sentences-via-a-multilayer-perceptron-mlp/(accessedSep.22,2020).

[17] O. O. Erunkulu, E. N. Onwuka, O. Ugweje, and L. A. Ajao, "Prediction of Call Drops in GSM Network using Artificial Neural Network,"

Jurnal Teknologi dan Sistem Komputer, vol. 7, no. 1, pp. 38–46, Jan. 2019, doi: 10.14710/jtsiskom.7.1.2019.38-46.

[18] "Logistic function," Wikipedia. Sep. 09, 2020, Accessed: Sep. 28, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Logistic$_$function&oldid=977583860.

[19] "Time series," Wikipedia. Sep. 09, 2020, Accessed: Sep. 29, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Time_series&oldid=977544968.

[20] "The Complete Guide to Time Series Analysis and Forecasting — by Marco Peixeiro — Towards Data Science." https://towardsdatascience.com/the-complete-guide-to-time-series-analysis-and-forecasting-70d476bfe775(accessedOct.01,2020).

[21] "Time Series concepts." https://faculty.washington.edu/ezivot/MFTS1stEdition.htm(accessedNov.22,2020).

[22] "Partial autocorrelation function," Wikipedia. Jul. 15, 2020, Accessed: Nov. 22, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Partial_autocorrelation_function&oldid=967803127.

[23] A. Mittal, "Understanding RNN and LSTM," Medium, Oct. 12, 2019. https://towardsdatascience.com/understanding-rnn-and-lstm-f7cdf6dfc14e(accessedOct.02, 2020).

[24] "Mean absolute percentage error," Wikipedia. Nov. 18, 2020, Accessed: Nov. 19, 2020. [Online]. Available: https://en.wikipedia.org/wiki/Mean_absolute_percentage_error

Printed by Tryckeriet i E-huset, Lund 2020

LUND UNIVERSITY

Series of Master's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2020-796
http://www.eit.lth.se