C-V2X Beam Prediction in the Era of Big Data Using Sequence to Sequence Model

by

Vivekanandh Elangovan

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical and Computer Engineering)
in the University of Michigan - Dearborn
2023

Doctoral Committee:

Professor Weidong Xiang, Chair
Associate Professor Sridhar Lakshmanan
Professor Hafiz Malik
John Locke, Molex LLC

Vivekanandh Elangovan

velango@umich.edu

ORCID iD: 0000-0002-8184-2204

Department of Electrical and Computer Engineering,
University of Michigan - Dearborn, USA.

Dedication

To my Wife, Daughter, and Parents

Shobana Kunagaran, Aivani Vivekanandh, Anika Vivekanandh, Elangovan Janakiraman and

Malathy Elangovan

Acknowledgements

The last five years of working towards my doctoral degree have been a great experience, and I would like to thank all the people who have contributed to it. To begin with, I would like to than Prof. Jayanthi who provided the fundamentals of Electronics and Communication while I was doing my bachelors at Thangavelu Engineering College – India. After my bachelor's, I received an opportunity to undertake my master's at Rochester Institute of Technology (RIT) where my advisor Dr. Chance Glenn, provided an eclectic ambiance to perform my research which motivated me to enhance my efficacy in research. After graduating with my master's from RIT, I started to work as Electrical Engineer at Anaren Microwave where I had the opportunity to work on consumer devices which works in ISM band. I would like to thank Dr. Hakan Partal, Dr. Ronak Gandhi who had provided me the insights to the research and how to approach and tackle an issue from research aspect. Soon after work, I had the opportunity to work with Dr. Jeff Reed from Virginia Tech and Dr. Dinesh Datla from Harris Corporation who provided the design process thinking for research and gave me an opportunity to work on 5G issue and submit my first conference paper. After I began my work in 2015 at Ford Motor Company, I encountered excellent benefits offered by the organization to support employees pursuing graduate studies. I immediately revived my plans of performing research and received an opportunity at University of Michigan under the mentorship of Prof. Weidong Xiang, funded by Mr. John Schneider (Director, Retired), Mr. Nicholas Colella (Sr. Manager) and Mr. John Van Wiemeersch (Supervisor, Retired) at Ford Motor Company. My role as a Research Engineer at Ford Motor Company has provided many insights regarding many features and functionalities of the automotive system.

I have had many insightful conversations with respect to wireless communication and enjoyed discussing my ideas with Technical Specialist Mr. John Locke and received excellent feedback on the implementation methods and research ideas. One such conversation triggered the core idea of the feature development of my doctoral dissertation. To execute the proof of concept, I received the Altair software for the EM simulation and the Real time testing is performed with the C-V2X chipset. I would like to thank my advisor Prof. Weidong Xiang, for his patience, excellent guidance, and support throughout the period. He understood that being a full-time employee and part time student, there are many hinderance and technical difficulties in work towards valuable research. His compassion and perception towards working students are commendable and pursuing a graduate study under his supervision is the obvious choice. My committee Dr. Sridhar Lakshmanan, Dr. Hafeez Malik and Mr. John Locke played a vital role in the sensitive phase of the thesis proposal by providing feedback and assisted me to shape up the scope of my work.

I would like to specially thank my spouse Ms. Shobana Kunagaran, for her support to pursue the challenge of doctoral studies in parallel with a full-time position and by taking care of an infant and raising them happy and strong. Last and most importantly, I would like to thank my family and friends for their affection and continuous support throughout my career. It is to all of them that I dedicate this thesis.

Table of Contents

List of Tables

List of Figures

List of Equations

List of Appendices

Abstract

The introduction of wireless connectivity to a world of automobiles brought many challenges which was transformed to many innovations. Researchers worldwide continuously strive hard to develop new technologies to improve the connectivity problems and enhance the user's comfort and enhance the safety of the users. Vehicle connectivity feature such as cellular connectivity for Wi-Fi connection provides better user convenience and Cellular-Vehicle to Everything connectivity provides safety features. Cellular Vehicle to Everything (C-V2X) has been an interesting technology which includes various connectivity methods such as Vehicle to Vehicle (V2V) connectivity, Vehicle to Network (V2N), Vehicle to Pedestrians (V2P), Vehicle to Infrastructure (V2I) and many more. The main goal of C-V2X system is to improve the safety of the vehicle and its surroundings. 3rd Generation Partnership Project (3GPP) has been working on standardizing the C-V2X which is referred as PC5.

Department of Transportation (DOT) and National Highway Traffic Safety Administration (NHTSA) governs the C-V2X system which issued Notice of proposed Rulemaking (NPRM) for the V2V communication which is based on the Dedicated Short-Range Communication (DSRC) defined in SAE J2735. A 360 degree "awareness" is expected from the V2V communication which provides the complete coverage for the vehicle with a range of 300 meters which leads to the adoption of omnidirectional antenna.

Omnidirectional radiational antenna provides the 360 degree "awareness" and provides us the 300-meter coverage, but it also increases the congestion factor, which has been regulated in SAE J2945/1. In a highly congested location where there are multiple vehicles present the

congestion factor is high i.e., there are high loads of data present everywhere and to reduce the congestion factor, the radiation power will be reduced. But, if the radiation power is reduced, it reduces the coverage requirement. To communicate for longer range without the requirement of reducing the power and increasing the congestion would be through beam i.e., beamforming.

The method to steer an array of antenna in an intended direction is called Beamforming. The radiated energy is concentrated into a narrow beam by adding the radio frequency (RF) signal either constructively or destructively based on the phase of the input RF signal. In various standards such as Wi-Fi and 5G, all the beams of the antenna perform scanning for each beacon interval (BI) and based on the various received signal, the optimum beam is chosen and adopted during the whole BI. If we implement the same methodology for our beam, we will get a medium or significant non-optimum selection of beam mainly due to the variation of Direction of Arrival (DoA) of multipath signals.

To avoid the non-optimal selection of beam, in this dissertation, a novel beam selection, “Intelligent Beam Selection” (IBS), was proposed, based on sequence-to-sequence machine learning prediction which enhances the selection of beam in real time with better accuracy compared to the traditional machine learning model.

In this research, IBS predicts the optimal beam to choose from various beams integrated to the vehicle as part of the C-V2X system. Deep learning (DL) models are developed by mapping the signal strength of the various antennas which are collected over the simulation and real drive scenario. The trained functional were utilized to predict the future beam of the vehicle, reflecting better signal reception without increasing the congestion factor. The IBS model is developed for the beam selection, but the model shall also be used for other time series feature prediction in real world scenarios.

Chapter 1 Introduction

1.1 Background

C-V2X solutions are designed to connect vehicles to nearly everything—including vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P), and vehicle-to-cloud (V2C) communication. In the device-to-device mode (V2V, V2I, V2P) operation, C-V2X does not necessarily require any network infrastructure. It can operate without a SIM, without network assistance and uses GNSS as its primary source of time synchronization.

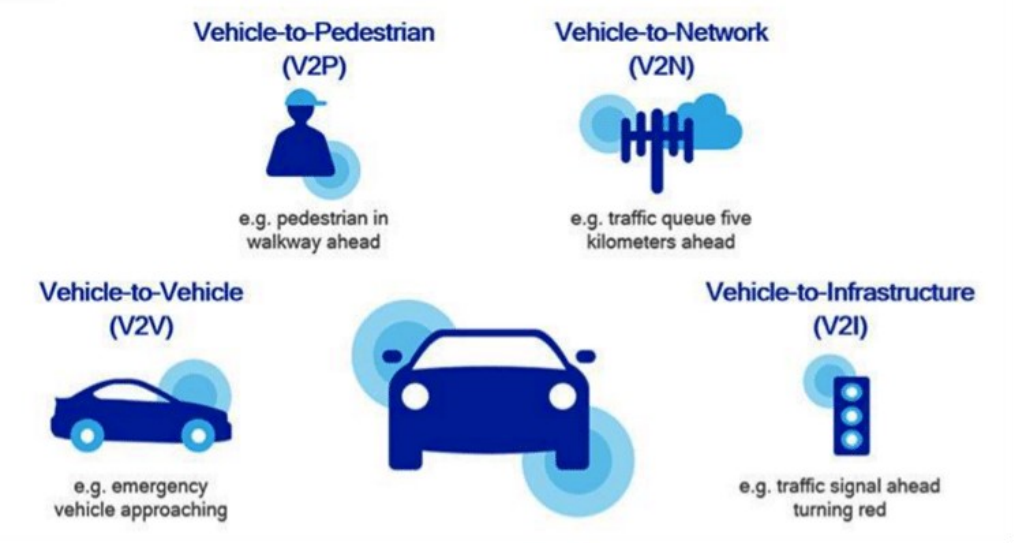


Figure 1Various Vehicle connectivity based on C-V2X System

The present standard dictates the bandwidth is 10MHz with just the Basic Safety message which is categorized into BSM Part 1 (vehicle size, position, speed, heading acceleration, brake system status) and BSM Part 2 (Varies upon event such as ABS activated and weather condition). The next generation which is the Advanced safety (Rel 14,15+) will have 70MHz bandwidth which

can support advanced safety services (e.g., higher bandwidth sensor sharing and wideband ranging/positioning) along with Basic safety message.

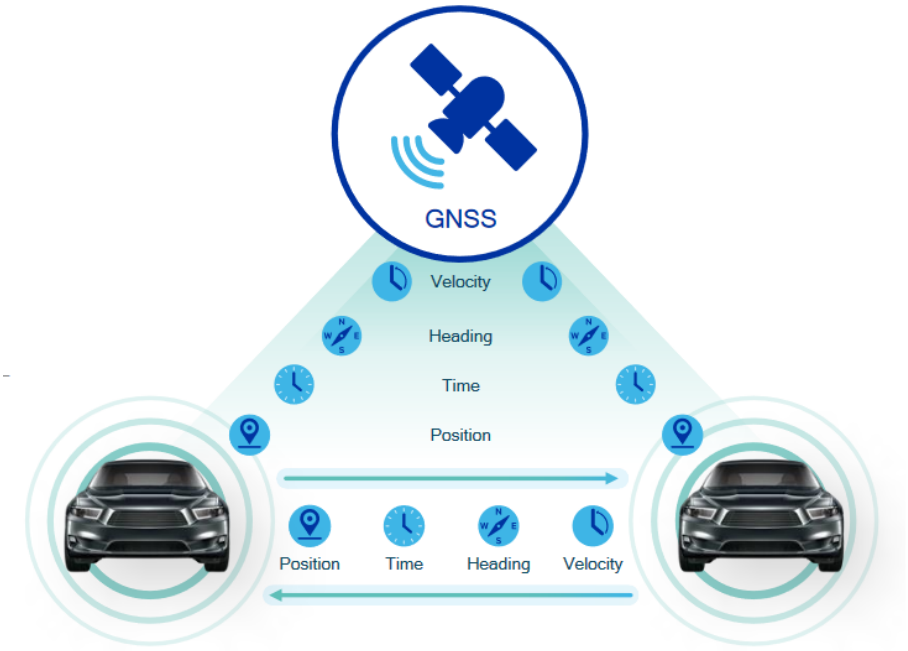


Figure 2 Basic Safety Message Information

The main goal of C-V2X system is to improve the safety of the vehicle and its surroundings [1] [2]. 3rd Generation Partnership Project (3GPP) has been working on standardizing the C-V2X which is referred as PC5. A more detailed explanation of C-V2X communication is provided in [3] [4] . Department of Transportation (DOT) and National Highway Traffic Safety Administration (NHTSA) governs the C-V2X system which issued Notice of proposed Rulemaking (NPRM) [5] for the V2V communication which is based on the Dedicated Short-Range Communication (DSRC) defined in SAE J2735 [6]. A 360 degree "awareness" is expected from the V2V communication which provides the complete coverage for the vehicle with a range of 300 meters

1.2 OEM's View

Many Automotive Original Equipment Manufacturer (OEM) has dedicated to implement C-V2X system and working on various sensor fusion techniques which shall be used for Advanced

Safety. OEM's want to use the C-V2X system for Do Not Pass Warning (DNPW) at 300-meter range and Lane Tracing Assist (LTA), Emergency Electronic Brake Light (EEBL), Lane Change Warning (LCW) at 150 meters.

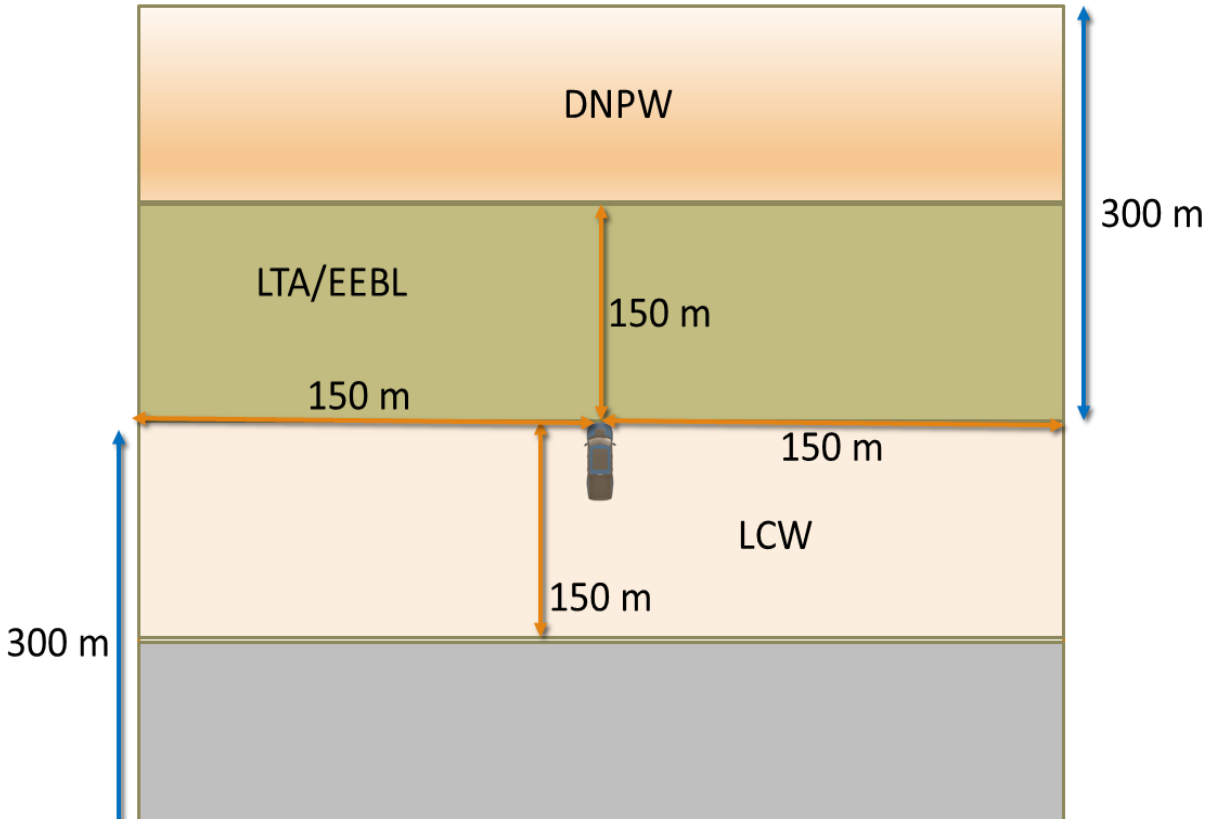


Figure 3 OEM's View of C-V2X coverage

To attain the 300-meter range without increasing the congestion factor one of the sensor fusion techniques is the usage of beamforming which are explored by various OEM's. By performing the beamforming, we would be able to achieve the longer range without increasing the congestion factor. The Beamforming will focus the radiation on certain angle based on the angle of the beam and the power is radiated towards that particular angle which increases the range and also reduces the Electromagnetic (EM) fields around the vehicle.

1.3 Machine Learning/Artificial Intelligence

Prediction of Wireless Channel has been a well-researched area and the usage of machine learning for predicting the wireless channel has been focused more due to bandwidth requirement and to reduce the noise factor. There has been various research such as Channel prediction on mmWave to reduce the RF chain, Denoising based on Vector AMP to reduce the noise in the channel. There has been research where the channel is converted to image as 2D image and predicting using Convolution Neural Network (CNN). Apart from the Neural network, there has been research in Markov Chain, Autoregressive Integrated Moving Average (ARIMA), Random Forest which are statistical prediction method. The best method to do a time series prediction is using Long Short-Term Memory (LSTM) Neural network which has produced consistent result for time series prediction. As the wireless channel is considered as time series model, LSTM has been the predominant focus in recent years.

Applications	Existing Works		ANN Tool	Data Analytics		RL
	Problems	Reference		Supervised	Unsupervised	
UAV	• UAV control.	• [91] • [94]	• FNNs. • FNNs	√		√
	• Position estimation. • UAV detection.	• [95] • [93]	• FNNs • RNNs.	√ √		
	• Resource allocation.	• [96] • [98]	• RNNs. • SNNs.			√ √
	• UAV deployment.	• [92] • [97]	• FNNs. • RNNs.	√ √		√
VR	• Head movement prediction. • Resource allocation. • VR content caching and transmission.	• [105] • [103], [104] • [107]	• RNNs. • RNNs. • DNNs.	√		√ √. √.
Caching and Computing	• Cache replacement.	• [109] • [110] • [111]	• FNNs. • DNNs. • DNNs.	√	√	√
	• Content popularity prediction. • Content request distribution prediction.	• [113] • [97], [114]	• FNNs. • RNNs.	√ √		
Multi-RAT	• Resource management.	• [115] • [101]	• DNNs. • RNNs.			√ √
	• RAT selection. • Transmission technology classification. • Multi-radio packet scheduling. • Mode selection. • Automatic root cause analysis.	• [116] • [117] • [118] • [120] • [62]	• CNNs. • FNNs. • RNNs. • FNNs. • RNNs.	√ √	√	√ √
IoT	• Model IoT as ANNs. • Failure detection. • User activities classification. • Tracking accuracy improvement. • Image detection. • Data sampling. • Entity state prediction. • Target surveillance.	• [121], [123] • [122] • [124] • [125] • [126] • [127] • [128] • [129]	• FNNs. • FNNs. • DNNs. • DNNs. • CNNs. • PNNs. • DNNs. • FNNs.	√ √ √ √	√ √ √ √	

Figure 4 Prediction Methods on Various Domains by M. Chen et.al

Time series prediction using Neural network has been long studied in many fields [7], such as stock forecasting [8], Weather forecasting [9], traffic flow forecasting [10], Global positioning prediction [11], Wireless Channel prediction [12] [13] , scenario identification [14] [15] [16]and most of the time series prediction follows the existing Neural network methodology such as ARIMA [17], Support Vector Regression (SVR) [18], traditional Artificial Neural Network (ANN) [19] and hybrid neural networks [20] [21]. Traditionally statistical methods such as ARIMA, exponential smoothing was often used for time series forecasting. Armstrong et.al [22] proposed 28 golden rules for time series prediction where ARMA and ARIMA is judged as the best time series prediction method. With the growth of Deep Neural network, there has been only a few time series classification algorithms have been proposed [23]. Wang et. al. [24] proposes a combination of Markov-LSTM where the multi-step Markov transition matrix is defined and then the LSTM is introduced to combine multiple first-order Markov chain.

Recently the Neural Machine Translation (NMT) has achieved state of the art performance using various methods such as Encoder Decoder [25], Encoder Decoder with Attention [26] [27] and Transformation [28]. These methods have been used by various researcher for language translation and these methods has been researched for time series prediction [29]. In the Encoder Decoder with Attention, the encoder and decoder are designed using various Neural Network such as Recurrent Neural Network (RNN), LSTM, Gated Recurrent Unit (GRU). The Attention is the key mechanism which provides improvement from Encoder Decoder model. The Attention mechanism provides information to which input sequence are relevant to each word in the output. Attention is proposed as a method to both align and translate.

Xu et. al. [30] proposed hard attention where it attends to exactly one input state for an output, [31] [32] shows a sequence-to-sequence prediction with the hybrid of hard and soft

attention. Elsayed et.al. [33] provides a modified hard attention called Saccader for vision by requiring only class labels for initial attention, whereas Papadopoulos et.al. [34] provides a multi-scaled hard-attention architecture for image classification. Sorokin et.al. [35] presents the "soft" and "hard" attention on Q learning which is based on feature extracted by CNN at different image regions, Deng et.al. [36] presents variational attention which is considered as an alternate to both "soft" and "hard" attention where the attention is set with tighter approximation bounds based on amortized variational interference, Malinowski et.al. [37] shows the use of hard attention by exploring various image attention mechanism to locate regions that are relevant to the question, Harvey et.al. [38] presents "hard" attention for image classification but based on the Bayesian optimal experimental design which helps in the speed up of the training process. The various presented methods are focused on vision, image and text-based classification and prediction. And these methods have proposed either a hybrid of "soft" and "hard" attention or focus on a single feature based on "hard" attention. There has not been much focus on the time series prediction and understanding the relationship between the time variables.

1.4 Problem Statement and Motivation

With the arrival of big data era, every industry would like to utilize the advantage of neural network for better prediction. Automotive industry has been focused on using advanced neural network for various reasons such as path prediction [39], language recognition [40] and many more in automated driving. C-V2X has been emerging technology within Automotive world which encompasses V2V connectivity, V2I, V2P and V2N. C-V2X communication is envisioned to enhance the safety of drivers, passengers, and pedestrians. C-V2X system is governed by the National Highway Traffic Safety Administration (NHTSA) and Department of Transportation (DOT). In 2017, the NHTSA and DOT issues Notice of Proposed Rulemaking (NPRM) for the

V2V communication by then V2V communication is like to be based on the DSRC defined in SAE J2735. The technology behind V2V communication expects an implementation of 360 degree "awareness" and a range of 300 meters where omnidirectional antennas are adopted.

Omnidirectional antenna gives a complete coverage of 300 meters but increases congestion factor, which is regulated in SAE J2945/1. In a highly congested vehicular location, a network experiences high data loads which requires reduced radiation powers. On the other hand, reducing power reduces the coverage. An effective way to communicate in longer range without increasing the congestion is implementing beam i.e., beamforming.

Beamforming is a technique in which an antenna array can be steered in a desired direction. The input RF signal is fed to the antenna array in parallel and signals are added constructively and destructively, depending on the phases, in such a way that they concentrate the energy into a narrow beam. In both Wi-Fi and 5G standards, during the antenna training phase of each beacon interval (BI) scanning is performed across all the beams and the optimum one is chosen and adopted during the whole BI. If the same method is performed in the C-V2X system, it will lead to medium or significant non-optimum selection of beam due to rapid variation of direction of arrival (DoA) of multipath signals.

There has been various research going on using Machine learning in Vehicular network [41], most of them focused on channel estimation [42], distance estimation [43], Vehicle trajectory [44] but very minimal in beam prediction [45] and only using traditional methods and nothing on beam prediction using deep neural network. Our research is focused on real-time beam prediction model.

Chapter 2 System Model

C-V2X system is yet to be implemented in real world by many OEM's as it is still in the early phase of specifications. There are constraints to get the data for the research aspect and due to the constraints of the available data, we decided to perform the data collection on our own. We also performed the simulation initially to understand the data pattern and build a beam forming antenna and collected the data for the research purpose.

2.1 Modelling for C-V2X Simulation

The simulations of wireless channels for C-V2X systems in the University Campus were built through the Altair WinProp software. The simulation modelled by WinProp either as empirical or semi-empirical models includes the main buildings, roads, and green fields of the campus which is close to 200 models.

Figure 5 shows the 3D view of the campus where the heights of the buildings are all set as 20 meters for simplicity with wall materials as concrete. The roads and parking lots was considered as concrete with thickness of 1cm. The vegetation areas were set to the heights of 1 meter for bushes and 3 meters for trees. Figure 5 also shows the simulation environment projected on top of the Google maps with drive paths. The simulation result includes RSSI, DoA, latency and gain for each multipath, all along the drive paths.

Figure 5 University Campus simulation Layout in 3D View and the drive Path

The simulation is at first pre-processed using FEKO WallMan program, which is a preliminary processing to perform 3D Intelligent Ray Tracing (IRT) in Altair WinProp simulation where complicated 3D propagation path including reflections, diffractions and scattering around building wedges, both horizontally and vertically [46]. The pre-processed program is used in FEKO ProMan program where the Ray tracing simulation is performed.

The transmitter antennas are placed at the heights of 15 meters above the ground transmitting at a frequency of 5.9GHz, the nominal channel for C-V2X systems. Omnidirectional antennas are adopted at transmitter as per system requirement and beamforming antenna arrays are only applied to vehicles (receiver), at heights of 1.5 meters above the grounds. At this time, the impact of vehicle to antenna array has not been considered since it is out of the scope of this effort.

Simulation provided us the flexibility to place the Roadside Unit (RSU) at different location and evaluate the power radiation pattern and how it impacts the signal strength across

various locations within the campus. Figure 6 shows the results of various RSU's placed at various location along the drive path and shows the power level across the campus.

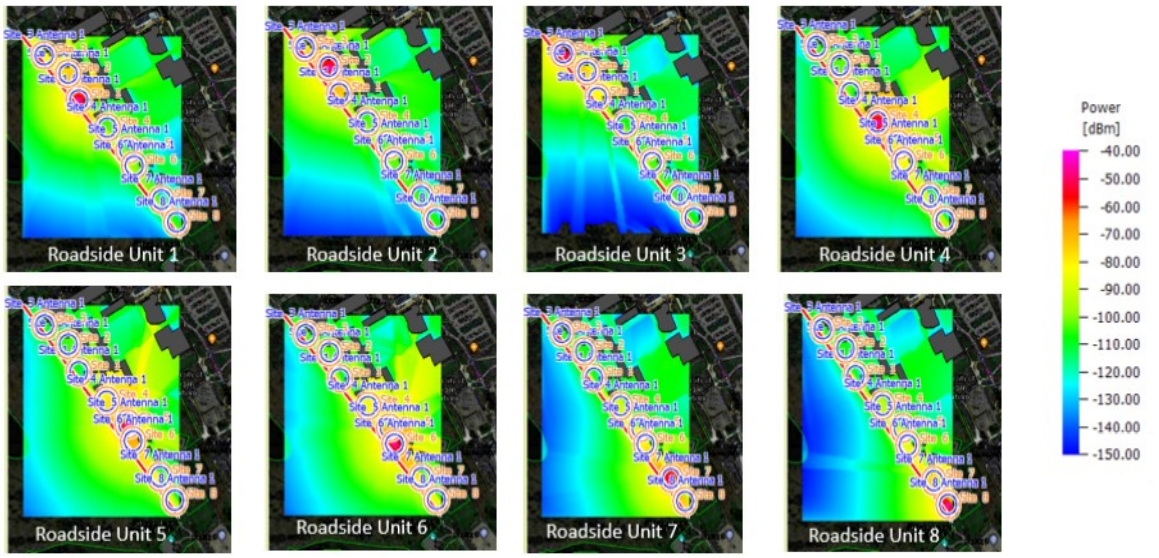


Figure 6 Simulation Result of Various RSU's location

From the simulation we can also under the multipath which the signal take. Each point on the desired receive location will have multipaths from the RSU. Based on the various paths, the angle of the signal to the desired received location is calculated i.e., the vehicle is moving from North to South and the angle of the signal is received at that location is calculated based on the vehicle direction. Figure 7shows an example of the various path the signal took between the RSU, and one receive location.

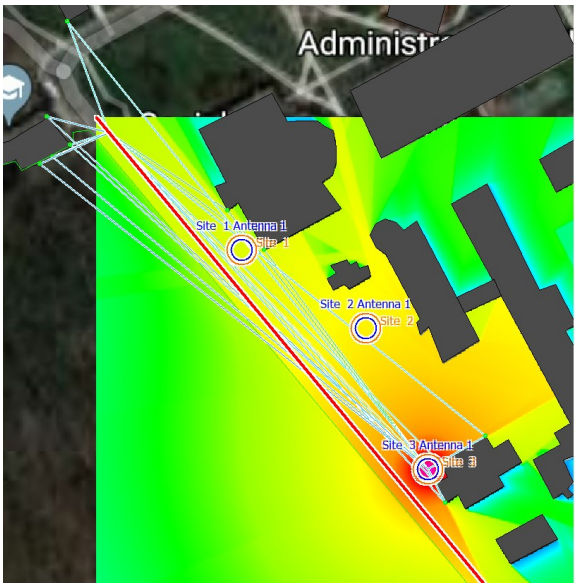


Figure 7Multipath of the Signal between RSU and the receive location

The simulation data were stored as .str files by WinProp for each transmitting antenna with a resolution of 1 meter. The str data is pre-processed through a Python program, of which the processing is shown in Figure 8. Finally, all the multipath signal data were stored in another file in JSON format.

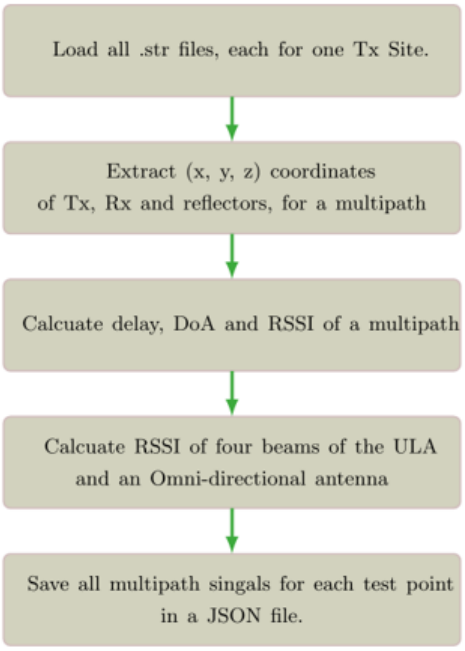


Figure 8 Data Extraction Flow Chart

2.2 Real time Data Collection

The real time data collection is performed by designing a 4-element uniform linear array (ULA) receiver antenna array built upon a 4x4 Butler matrix which will be used to collect the data for the machine learning algorithm so that we can achieve a real time beamforming selection for C-V2X system. Shown in Figure 9 is the system design of the 4x4 beamformer designed for the C-V2X system where we have four 5.9 GHz whip antennas, separated by quarter of wavelength (**λ**/4) which are connected to a 4x4 Butler to form a ULA. A switch box containing SPDT (ZFSWA2R-63DR+) and SP4T (ZSWA4-63DR) is used to select one of the outputs of the Butler switch. the signal between two adjacent antennas within the array creates a phase difference of ϕ= kdcosθ, where wave number (k) and Array Factor (AF) is given by Equation 1 and 2 respectively.

$$k = \frac{2\pi}{\lambda} \tag{1}$$

$$AF(\theta) = \left| \frac{1}{N} \sum_{n=0}^{N-1} e^{j(nkdcos\theta + \alpha_n)} \right|^2 \tag{2}$$

N is the total number of antennas
α_n is additional phase shift

For a broad side antenna array, the AF can be further written as Equation 3.

$$AF(\theta, \phi) = \left[\frac{sin(\frac{1}{2}N(kdsin\theta cos\phi + \beta)}{N\, sin(\frac{1}{2}N(kdsin\theta cos\phi + \beta)} \right]^2 \tag{3}$$

Finally, the beamforming radiation pattern is given by Equation 4.

$$BF(\theta, \phi) = AF(\theta, \phi)P(\theta) \tag{4}$$

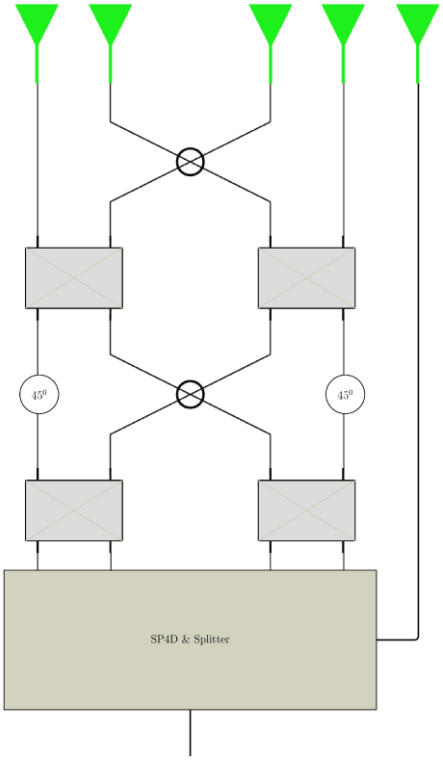


Figure 9 System Design of 4x4 Beamforming for C-V2X

The radiation pattern of the ULA is shown in Figure 10 where the radiation pattern for all the 4 ports is shown.

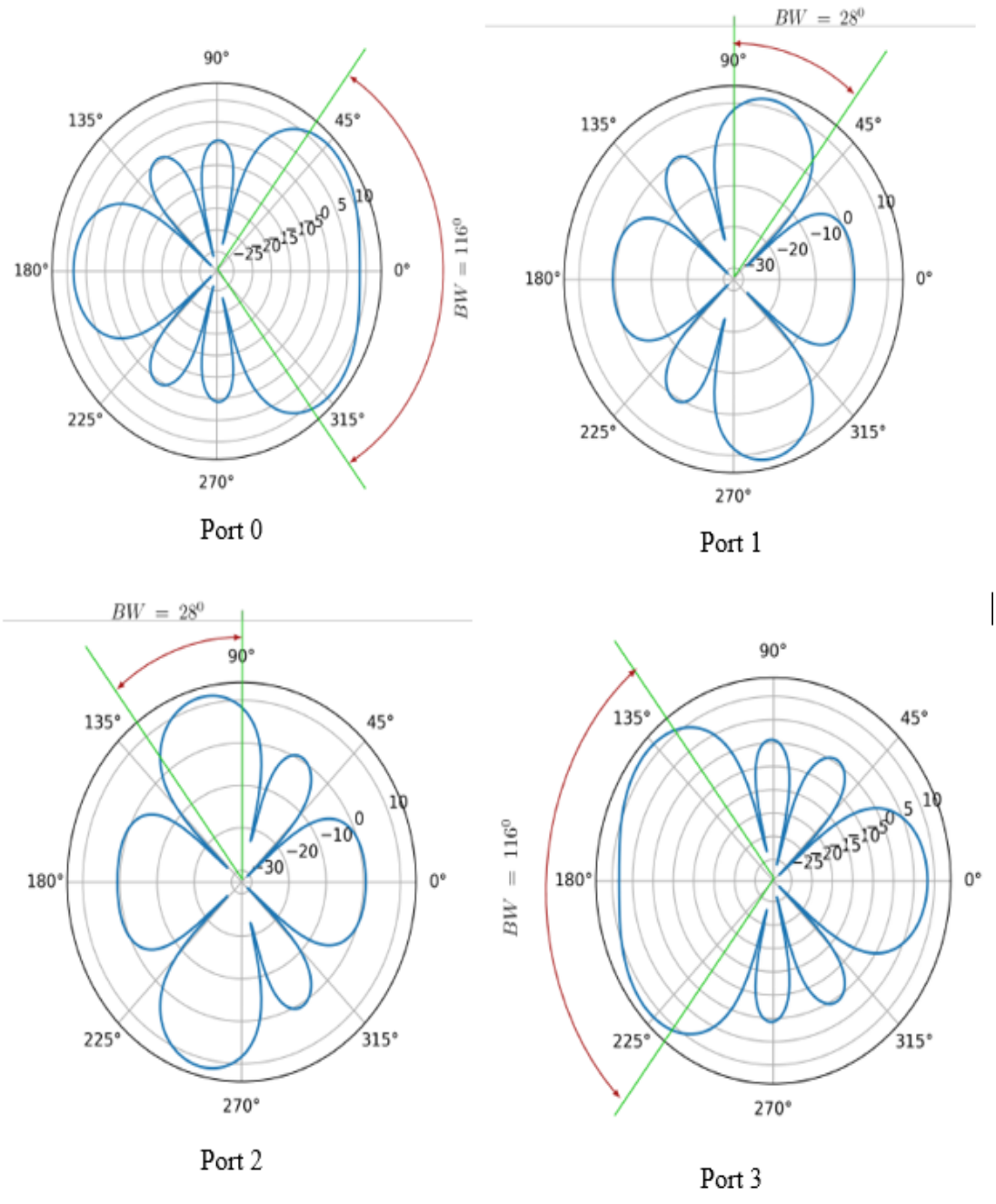
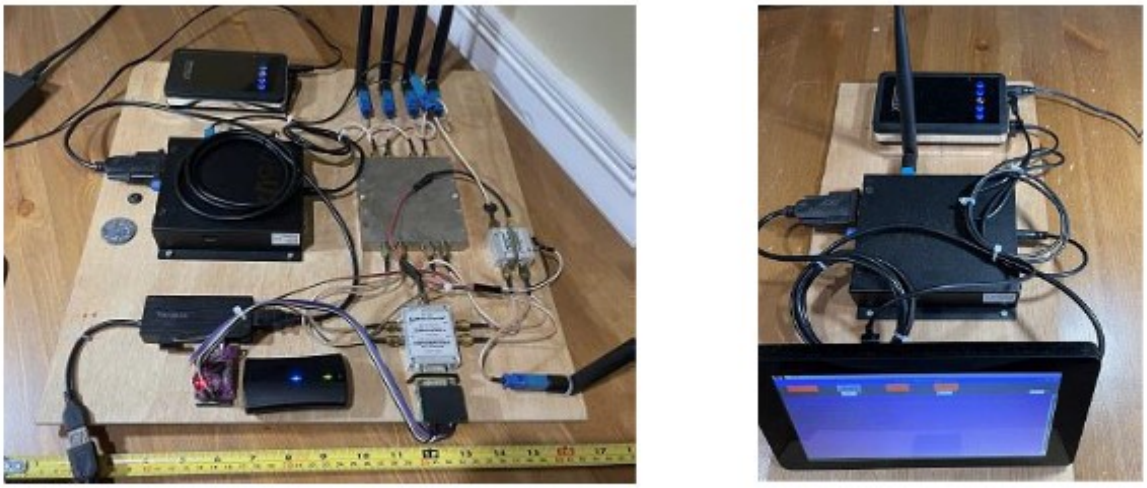


Figure 10 Radiation Pattern of 4 Beam Antennas

The receiver antenna is connected to the C-V2X onboard unit, and the receiver module also has a Raspberry Pi which is used to command the radio. Both the C-V2X onboard unit and the Raspberry Pi is powered using a portable battery (XTPower MP-10000). The receiver unit is placed on top of the car and the entire unit is shown in Figure 11. The transmitter which is placed

on a fixed location has a single antenna which is omni directional and connected to a C-V2X onboard unit and powered by a portable battery as shown in Figure 11.



Receiver Antenna Unit

Transmitter Antenna Unit

Figure 11 Receiver and Transmitter Antenna Unit

The test is performed at the university campus shown in Figure 11Figure 12, where the transmitter is placed on one of the parking decks (2nd floor) as shown in figure and the vehicle with the receiver module is driven around the campus and the data is collected throughout the campus which is used for the machine learning validation.

Figure 12 Google Maps of Campus with Tx Location

Chapter 3 Machine Learning Models

Ranging from Markov Chain to Machine learning [47] [48] [49] [50], there has been various methods being well-researched for Wireless Channel prediction. In particular, there has been more focus on Neural network research in wireless communication. Huang et al. presented deep learning models for DoA estimation based on multiple input multiple output (MIMO) antenna system using hybrid precoding. [51] speaks about interference alignment in a congested time varying network. Tuzi et al. presents how CNN has been used for estimating the distance in a DSRC system [52]. Sheng et al. studied an intelligent heterogenous 5G wireless network architecture by adopting a reinforcement learning (RL) for V2V, V2I system [53]. The research presented in [54] provides new sensing technique in V2V network for improved road safety. The authors of [55] show the method of using RL for vehicle detection. [56] shows an effective sequence to sequence time series prediction using hard attention. [57] presents a survey on the various use of deep learning in wireless network and [58] presents a tutorial on Neural networks-based machine learning for wireless networks. The research focused on [59] [60] [61] presents Vehicle to Vehicle communication model for varying channels.

Prior works are in wireless channel prediction and C-V2X links using single variate machine learning. There has been no work trying to reduce the congestion factor using machine learning or using new sensing technology based on multivariate system.

In this section, we discuss the implementation details of various machine learning and the training methodology including conventional and unconventional methodology and how we train and test our dataset using these various methods. We split the dataset into three sets such as:

- Training sets: 80% of data set
- Validation sets: 10% of data set
- Prediction sets: 10% of data set

In our implementation, for the given data set, we use a sliding window input so that we achieve maximum overlap of sequences and in our training method we use the guided training methodology. In the guided training we feed the actual data as the next input which aims to achieve faster convergence by guiding the model towards the local minima. Whereas during the validation and prediction we use the unguided methodology where we feed the predicted data as the next input as we don't have access to the actual data set during these stages.

3.1 Why Machine Learning Model?

The straightforward implementation for choosing the beam would be adaptive antenna selection i.e., scanning for the strongest signal on all the beams and sticking to a beam which has the strongest signal until the next Beam Interval. The adaptive antenna selection is implemented in Wi-Fi routers and is being used to extend the range of the signal and for better coverage. In the CV2-X system, the adaptive antenna selection implementation chooses to select a beam every 100msec i.e., every 4λ where λ is the wavelength of 5.9GHz (Change of beam interval is every 100 msec which translates to the length of 4λ). Considering a vehicle speed of 60 mile/hour, the distance moved in every 4λ i.e., 100 msec, approximately 3 meters. In the simulation, 3 meters reflects to 3 data points and a beam was chosen based on the next 3 consecutive data points. For example, if beam 1 is selected during the initial scan, the next three packets will be using beam 1 to receive the signal. Observed from simulation data, implementing adaptive antenna selection has an evident data loss resulting in only 29.41% accuracy. This motivates the effort to use the machine learning in predicting the beam, which aims to achieve an enhanced efficiency of data reception.

3.2 Conventional Models

There are many models which are being used for time series prediction such as ARIMA, VAR, LSTM, Encoder Decoder and so on. In this section, we will investigate the details of how these methods are being implemented and how we have approached to analyze the dataset using these models.

3.2.1 Persistence Prediction

The most "naïve" forecast which is the persistence algorithm or Walk-Forward validation. The persistence algorithm uses the value at the previous time step (t-1) to predict the expected outcome at the next time step (t+1).

3.2.2 ARIMA

In Autoregressive Integrated Moving Average (ARIMA) (p,d,q) model, AR(p) is used to forecast data through best fitting past p values in the sense of least mean square errors. Moving Average (MA) (q) uses q forecast errors instead of past values. The integrated processing in ARIMA is to eliminate non-stationarity of original time series data set through the d^{th} order of integration.

A time series is said to be stationary when its first and secondary moments, mean and variance do not dependent upon the time. In another words, there should not be any trends or seasonality shown in data series. If a data series is not stationary, it can be differentiated to be stationary. Differentiation helps to stabilize the mean of the time series by changing the levels of the time series and therefore eliminating their trends or seasonality.

The ARIMA system (p, d, q) is determined by using Partial Autocorrelation (PACF) and auto-correlation (ACF) of the data series. The general characteristics of theoretical ACF are shown in Table 1:

Table 1 Determination of ARMA system based on ACF's and PACF's

Model	ACF	PACF
AR(p)	Plot decays gradually toward zero	Plot cutoff at zero
MA(q)	plot cutoff at zero	Plot decays gradually towards zero
ARMA (p, q)	Plot decays gradually towards zero	Plot decays gradually towards zero

In our AR implementation we have a single variable input and single variable output i.e., the signal strength at time t is provided as input while as the signal strength at time t+1 is predicted. Guided training methodology is used during the training phase where the measured signal strength is provided as the next input. The order of the implementation is based on the best Akaike Information Criterion (AIC) which is an estimator of out-of-sample prediction error. Figure 13

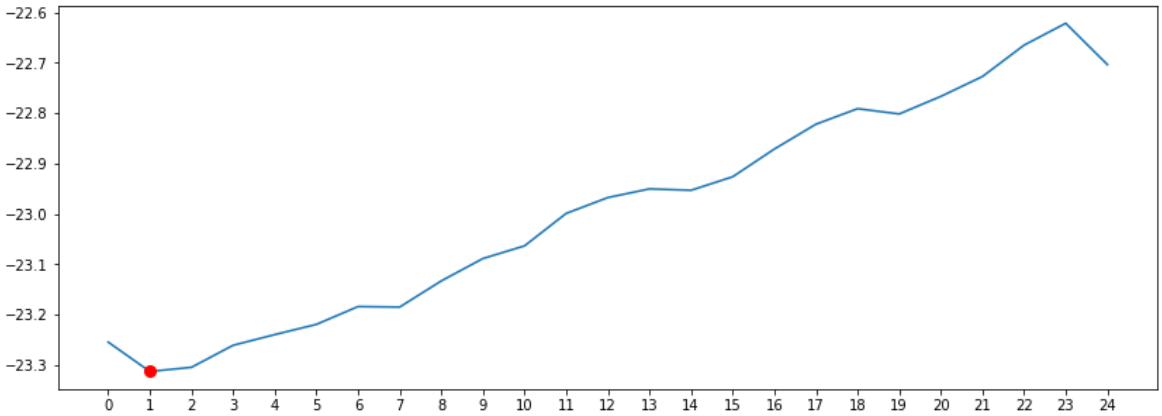


Figure 13 AIC Index Variation with Lags

shows such a result of a site antenna simulation where the best order is chosen from 25 lags. The lower AIC is generally "better" and in our dataset the AIC index is 2 which quantifies the goodness of fit and the simplicity of the model.

Based on our dataset, the q value is set to be zero, which translates to be an AR model. Our AR model is a univariate system i.e., signal strength from each individual beam is predicted individually by providing RSSI at t as input and predicting RSSI at t+1. The output from each model at their time sequence is thereafter filtered to select the strongest signal. Figure 14 illustrates the implementation method of AR with various univariate system with selection output.

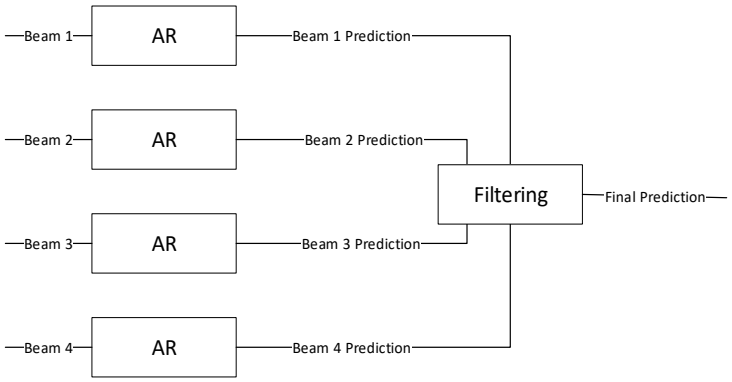


Figure 14 The implementation of AR Beam Prediction

3.2.3 VAR

A finite order Vector Autoregression (VAR) process with a finite order of MA error term is called as Vector Autoregressive Moving Average (VARMA) process. VAR model is a multivariate time series model containing a system of n equations of n distinct, stationary response variables as linear functions of lagged responses and other terms [62]. A VAR model is useful for the prediction of multiple variables i.e., multivariate using a single model.

For example, if there are two variables X_t and Y_t then the VAR equation is shown in Equation 5 and 6.

$$Y_t = \alpha_{10} + \alpha_{11}Y_{t-1} + \alpha_{12}Y_{t-2} \ldots + \alpha_{1p}Y_{t-p} + \theta_{11}X_{t-1} + \theta_{12}X_{t-2} + \cdots + \theta_{1p}X_{t-p} + \varepsilon_{1t} \quad 5$$

$$X_t = \alpha_{20} + \alpha_{21}Y_{t-1} + \alpha_{22}Y_{t-2} \ldots + \alpha_{2p}Y_{t-p} + \theta_{21}X_{t-1} + \theta_{22}X_{t-2} + \cdots + \theta_{2p}X_{t-p} + \varepsilon_{2t} \quad 6$$

The signals consist of line-of-sight (LOS) and multipath component (MPC) due to different reflection, diffraction and scattering with diverse DoA, which are received by one or more of the four beams. Such propagation is represented by the cross terms, θ in Equation 5 and 6, which offers the rationale to adopt VAR.

Both α and θ can be estimated using ordinary least squares. In VAR Model all the RSSI received by the four beams are provided as an input at an instant and the RSSI of individual beams are predicted, followed by a filtering module to select the strongest one. The diagram of the VAR model is shown in Figure 15, the order of which is the same as that of AR module.

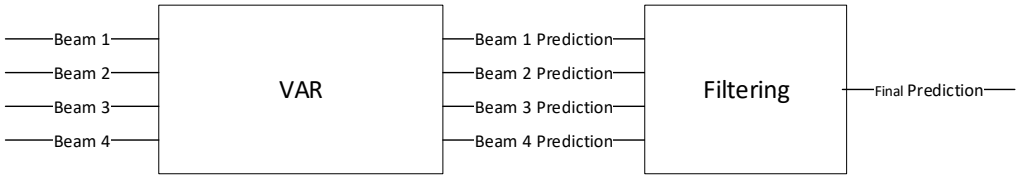


Figure 15 Implementation of VAR Beam Prediction

3.2.4 LSTM

LSTM is a popular model for processing sequential data based on the Recurrent Neural Network (RNN), where neurons are replaced by memory cells, each having:

- A Forget Gate – used to shape what information to be discard from the cell.
- An Input Gate – used to determine which values from the input to update the memory state.
- An Output Gate – used to decides what to output based on input and the memory of the cell.

The LSTM architecture is shown in Figure 16.

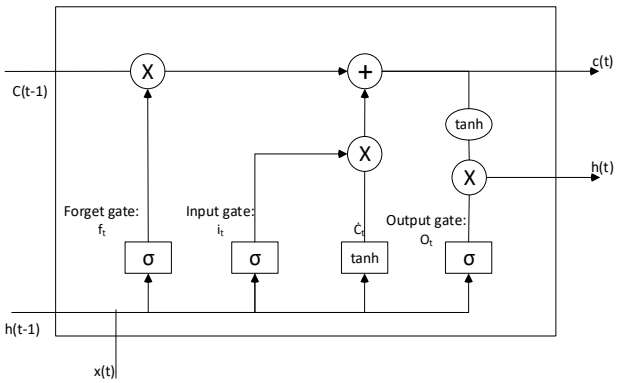


Figure 16 The illustration of LSTM Architecture

These gates Equation 7,8,9,10,11 and 12 offer RNN the capacity of learning long-term dependencies. By regulating the flow of information in and out of the cells, it helps RNN to remember only valuable information for long periods of time, hence enhance prediction capacity [63].

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t]^T + b_f) \quad 7$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t]^T + b_i) \quad 8$$

$$\dot{C}_t = tanh(W_C \cdot [h_{t-1}, x_t]^T + b_C \quad 9$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t]^T + b_o) \quad 10$$

$$C_t = f_t * C_{t-1} + i_t * \dot{C}t \quad 11$$

$$h_t = o_t * tanh(C_t) \quad 12$$

Where,

f_t is the forget gate.

i_t is the input gate.

o_t is the output gate.

C_t is the cell state.

$\dot{C}_t$ is the candidate for cell state.

h_t is the output of the cell and x_t is the input.

h_{t-1} is the output of the previous cell.

W_f, W_i, W_C, W_o are weights of forget gate, input gate, cell state and output gate, respectively.

b_f, b_i, b_C, b_o are biases of forget gate, input gate, cell state and output gate, respectively.

There are several variants of LSTM, but all have similar performance [64] and this paper adopted Vanilla LSTM. The LSTM implementation is a multivariate input and multivariate output system with the input sequence length of 10 and the output length of 4. The LSTM implementation adopted a time series generator from Keras module having 64 layers followed by a Dense layer of 32 using rectifier linear unit (ReLU) as activation function and a dense layer of 4 as output, as shown in Figure 17. The loss function used is mean squared error (MSE).

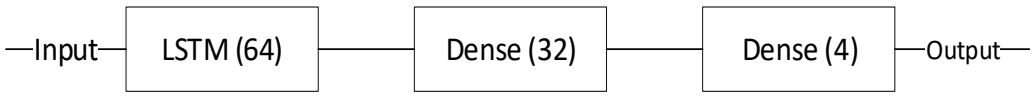


Figure 17 LSTM Multivariate System

The model is trained by using training data set and then validated by validation data set in an epoch of 1000, which trades off between the loss curve and overfitting. The signal strength of sequence from the 1st to t time slot are provided as input and the signal strength at the t+1 time slot is predicted, for the four beams and the strongest signal is chosen after passing the filtering module.

Table 2 Parameters of LSTM Model

Parameter Name	Parameter Value
API Module	Keras
LSTM Layer	64 layers
Dense Layer	32 layers
Activation Function	ReLU
Epoch	1000
Loss Function	MSE

3.2.5 Encoder Decoder with Attention

Encoder Decoder was developed to address the sequence-to-sequence machine translation with a set of input sequence and a set of output sequence. Attention is a mechanism that was developed to improve the performance of the Encoder Decoder RNN on machine translation. From a high-level, the Encoder Decoder model is comprised of two sub models.

- Encoder – The encoder will perform the act of stepping through the input series and encoding the entire sequence into a fixed length vector called context vector
- Decoder - The decoder will perform the act of stepping through the output series while reading from the context vector.

This approach has issues while decoding longer sequence and hence Attention is introduced.

- Attention - Instead of encoding the input sequence into a single fixed context vector, the attention model develops a context vector that is filtered specifically for each output time step.

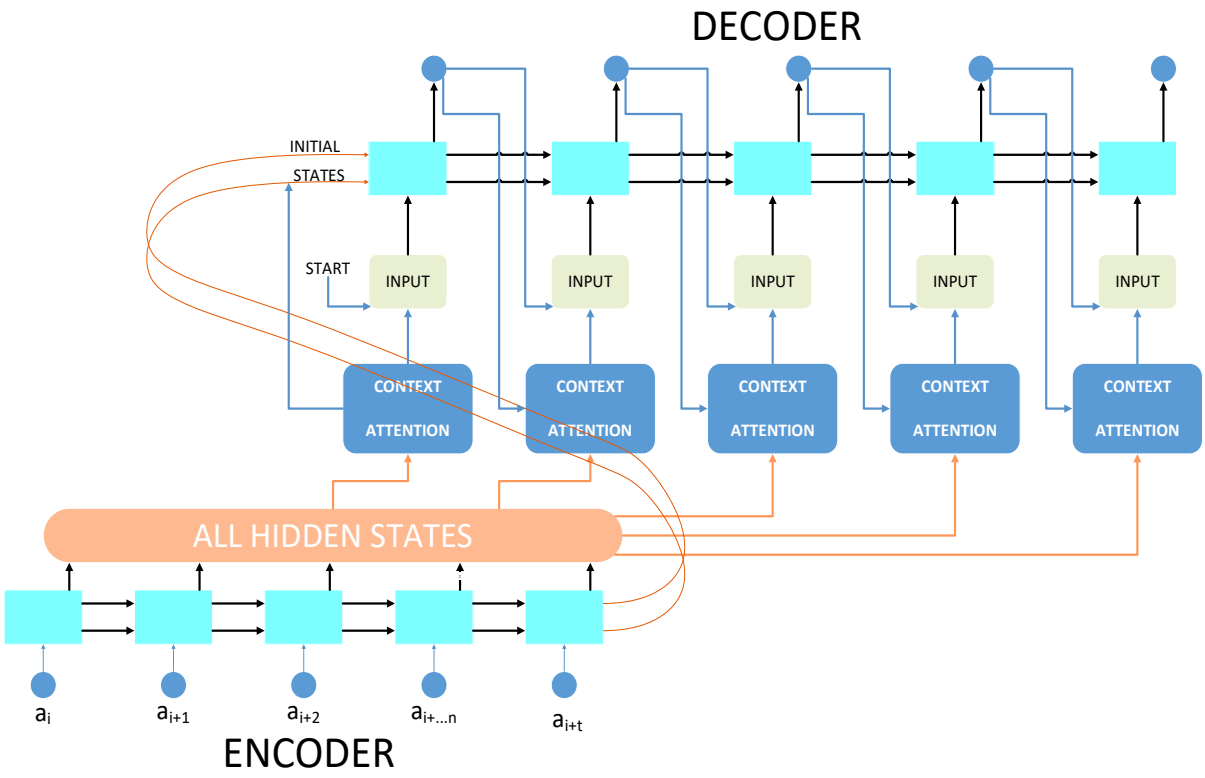


Figure 18 Model of Encoder Decoder with Attention

With the introduction of Attention as shown in Figure 18, the decoder output is more specifically focused which provides better prediction. The score is calculated in the Attention model which helps to relate the encoder's all hidden states and the previous decoder's output. The two important scores are proposed by Bahdanau 13 and Luong 14.

$$score\left(h_t, \bar{h}_s\right) = \vartheta_a^T \tanh(W_1 h_t + W_2 \bar{h}_s) \quad 13$$

$$score\left(h_t, \bar{h}_s\right) = h_t W \bar{h}_s \quad 14$$

where ht is the Encoder all hidden states and hs is the decoder output

The weights are learned during the backpropagation i.e., during the training. The weights are normalized and then the context vector is calculated in Equation 15.

$$c_t = \sum_s \alpha_{ts} h_t \quad 15$$

After calculating the context vector, we will concatenate the context vector with the previous decoder hidden state which will be the input for the next decoder output.

It shall be noted that during the score calculation, the weights are learned during the training i.e., the weights are set as random and then trained during the backpropagation. This method doesn't provide us any insight on how the weights are calculated and in the time series calculation this creates a randomness on the focus in the attention sub model.

3.3 Unconventional Models

This section will discuss the unconventional models such as combination system and using one model to train another model.

3.3.1 Long Short-Term Memory – Multivariate Input Univariate Output (LSTM MIUO)

The LSTM implementation of MIUO considers the correlation among various input and perform a better prediction on a single output variable. The model is the same as the LSTM system detailed in Table 2 where the output layer is 1, instead of 4 as shown in Figure 19.

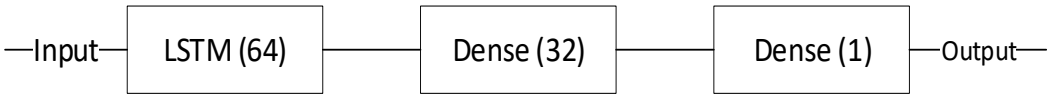


Figure 19 LSTM MIUO Beam Prediction

The LSTM MIUO model uses the same epoch as the LSTM model. It predicts the RSSI of each beam, and the strongest signal is filtered out. Figure 20 illustrates the implementation method of LSTM MIUO with a filtering module.

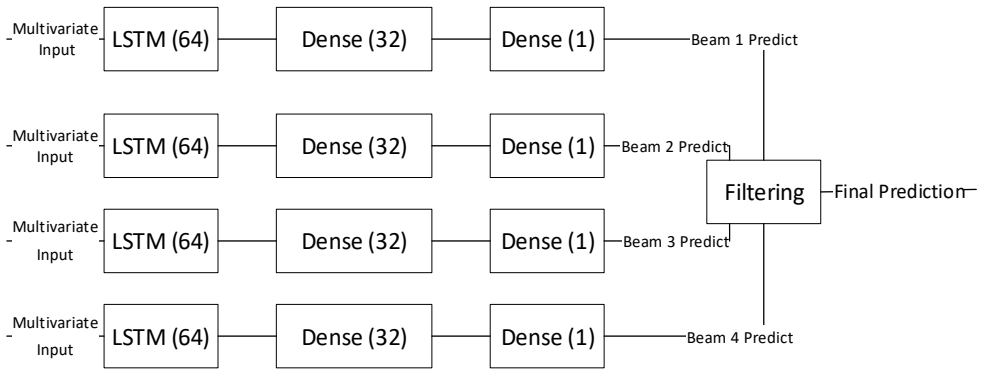


Figure 20 Implementation of LSTM MIUO Beam Prediction

3.3.2 Long Short-Term Memory with Vector Autoregressive (LSTM VAR)

The hybrid approach of LSTM with VAR improves the training process by extracting what VAR has learned during the VAR analysis and improves the prediction in multivariate LSTM System. The VAR model design is explained in section III where the model is trained and validated which leads to a well fitted VAR. In the implementation, VAR is used to improve the training of the LSTM multivariate system.

In the hybrid method, the two-step training procedure i.e., the fitter value from the VAR is adopted as an input to the LSTM model followed by data source. The training is based on different data set but from the same data source to anti overfitting. Moreover, dropout is adopted to avoid the catastrophic forgetting.

3.3.3 Long Short-Term Memory with Autoregressive (LSTM AR)

The hybrid approach of LSTM with AR improves the training process by using the AR prediction method. In the LSTM AR implementation, two step training procedure is adopted. The first step is based on the AR values and the next by the original data source. Same approach is adopted to avoid catastrophic forgetting. As the AR is a univariate system, the LSTM input in the first step is the combination of the 4 AR's forming a Multivariate system.

Chapter 4 Intelligent Beam Selection Model

Initially the combination of various machine learning model i.e., an ensemble approach is being tried to predict the signal strength of the beam.

4.1 Ensemble Approach

In our ensemble approach we categorized our algorithm to four different steps such as Training Block, Validation Block, Classification Block and finally Prediction Block as shown in Figure 21.

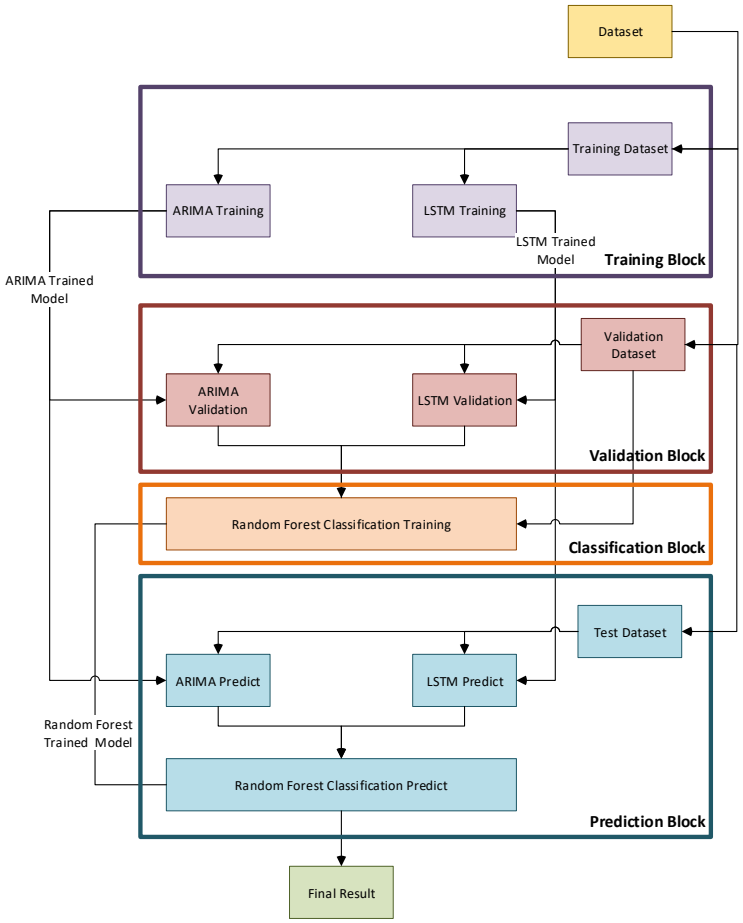


Figure 21 High level block diagram of ensemble time series prediction

4.1.1 Training Block

The training block is where the time series algorithm is trained. In our ensemble approach we train our data set using ARIMA and LSTM. And in our approach, we perform multi step prediction and not a single step prediction as it is shown previously that a single step prediction is like persistence prediction i.e., the prediction is more following the actual result rather than predicting. During the multi-step prediction, we use a window size of 10 and two hidden layers and each layer includes 50 artificial neurons.

From the dataset we use 80% of the value as the training set and input the entire training set to both ARIMA and LSTM prediction and train the model. We calculate the p, d and q value for the ARIMA during the training block and train the ARIMA based on the p, d and q values. The trained model is the base model for all the validation and prediction for future algorithm.

4.1.2 Validation Block

The validation block is where the trained model is validated with the validation data set which was another subset of the entire dataset. During the validation block if the validation result doesn't correspond to the actual dataset, the model needs to be retrained i.e., back to the training block where the parameters are tuned so that we can achieve better result in the validation block.

4.1.3 Classification Block

The classification block is where both the time series prediction from the validation block is trained, and the best prediction algorithm is chosen for each data. During the classification block, the predicted value from the validation block is sent to the classification algorithm along with the expected result i.e., if the ARIMA prediction is better than the LSTM prediction which is identified using standard error between the predicted output and the actual output and taking the absolute

value of the error. Based on the error i.e., minimal error, the output is classified as LSTM or ARIMA for each data.

For a given validation data x_i, the error is calculated for both the LSTM prediction, x_{iLSTM} and ARIMA prediction, x_{iARIMA}. If the error of $abs(x_{iLSTM} - x_i)$ is lesser than the error of $abs(x_{iARIMA} - x_i)$, then LSTM data is considered as valid output or if the error of ARIMA is lesser than the LSTM then the ARIMA data is considered as valid output. Table 3 shows the input format to the Classification block where the ARIMA validation result and the LSTM Validation result is fed as input to the classification algorithm and the expected output based on the error is fed as input for training the classification algorithm.

Table 3 Classification Block Input and Output

Input		OUTPUT
ARIMA	LSTM	Minimal Error
x_{iARIMA}	x_{iLSTM}	$Lowest\ of\ [abs(x_{iLSTM} - x_i)\ or\ [abs(x_{iARIMA} - x_i)$

4.1.4 Prediction Block

The prediction block is where the final prediction of our ensemble approach i.e. the test data is provided as an input to our model which was trained during our training block and the classification block. The predicted data from the ARIMA model and the LSTM model is fed to the classification model which provides us the result of our predicted data.

4.2 Attention with Transition States

In our model, we represent a transition matrix TM, which helps the model where to focus the attention when generating the next time sequence data. The transition matrix is probability of transition from one state to another state which shall be generated from the given data set i.e., given certain state what is the probability of moving to another state or staying in the same state. A method of representing the Transition states is shown through the matrix in Table 4. This transition probability values shall be used in the scores during the attention sub model which shall provide the information of where the focus needs to be for the decoder during the prediction of the tth time series.

Table 4 State Transition Matrix

State	a_i	a_{i+1}	a_{i+2}	…….	a_{i+t}
a_i	$P(a_i \mid a_i)$	$P(a_i \mid a_{i+1})$	$P(a_i \mid a_{i+2})$	…….	$P(a_i \mid a_{i+t})$
a_{i+1}	$P(a_{i+1} \mid a_i)$	$P(a_{i+1} \mid a_{i+1})$	$P(a_{i+1} \mid a_{i+2})$	…….	$P(a_{i+1} \mid a_{i+t})$
a_{i+2}	$P(a_{i+2} \mid a_i)$	$P(a_{i+2} \mid a_{i+1})$	$P(a_{i+2} \mid a_{i+2})$	…….	$P(a_{i+2} \mid a_{i+t})$
…..	…….	…….	…….	…….	…….
a_{i+t}	$P(a_{i+t} \mid a_i)$	$P(a_{i+t} \mid a_{i+1})$	$P(a_{i+t} \mid a_{i+2})$	…….	$P(a_{i+t} \mid a_{i+t})$

When the score is calculated based on Equation 16, the weights are determined based on the transition matrix TM.

$$score\left(h_{t,}\bar{h}_s\right) = h_t W \bar{h}_s \qquad 16$$

where W is the Transition Matrix

The weights are determined based on the encoder input time series $(a_i, a_{i+1}, a_{i+2} \ldots a_{i+t})$ data and the last predicted time series data b_{t-1}, where $b_{t-1} \subset (a_i, a_{i+1}, a_{i+2} \ldots a_{i+t})$. It would be the probability of $(a_i, a_{i+1}, a_{i+2} \ldots a_{i+t})|(b_{t-1})$i.e., $P(a_i, a_{i+1}, a_{i+2} \ldots a_{i+t}|b_{t-1})$

$$W = [P(a_i|b_{t-1}), P(a_{i+1}|b_{t-1}), P(a_{i+2}|b_{t-1}) \dots, P(a_{i+t}|b_{t-1}) \quad 17$$

The weight matrix is determined based on Equation 17. This provides us the insights on what is the highest probability of time series decoder output which is provided by the previous output and is known to the next decoder state. This also ensures that the conversion is not the traditional language prediction method which is a one-to-one translation. The Weight matrix provides us the time series prediction.

An example is shown in Figure 22 how the Weights W is chosen in the score calculation of the attention sub model. Considering the encoder input time series data with 4 sets of data as ai, ai+2, ai+1, ai and the first decoder loop output as bt-1 and as the decoder output is a subset of the input, we consider bt-1 as ai+3. Considering the input and output, the weights of the score would be P (ai | ai+3) = P4, P (ai+2 | ai+3) = P12, P (ai+1 | ai+3) = P8 and P (ai | ai+3) = P4 i.e., it would be P4 P12 P8 P4.

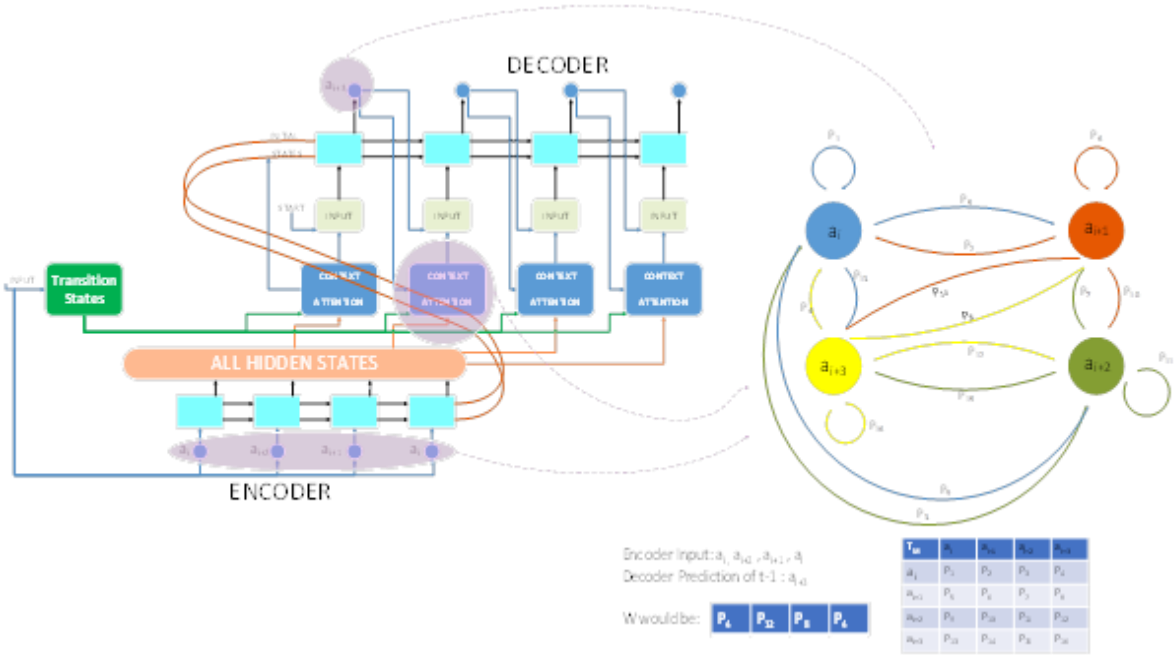


Figure 22 Encoder Decoder - Attention with Transitionl Matrix

In our model, the encoder part will act like traditional encoder, where it receives the input data and process it. It outputs its last hidden state along with the last cell state to the decoder as input. It also stores all its hidden state of every encoder block which shall be used in the context vector. The decoder initial input is sent by the encoder and the decoder runs in loops. At each time step, the decoder consumes its inputs and states and outputs its last hidden state and last cell state. Decoder uses its last hidden state as the next input to the attention sub model which shall process the data as an input to the next decoder time step. It also uses the last hidden state for the prediction for the current time step.

In the attention sub model, the encoder hidden state is used as one of the inputs for the score along with the weights from the transition matrix TM, and the decoder output. Using the score, the context vector is calculated which shall be concatenated with the decoder output and provided as an input to the next decoder state.

The transition matrix illustration is similar to the state space model, as both are time varying system. But the state space model has the ability to change the number of states, observation, disturbance i.e., a state space model is a dimension varying model and also the state space model can handle the system with nonzero initial condition. On the other hand, transition matrix proposed in this paper is not a dimension varying model incapable of handling the nonzero initial condition because the matrix will be skewed.

The adaptation of transitional matrix in principle is to add statistics information over long term data to attenuation and thus change attenuation from blind unsupervised learning to supervised or semi supervised learning. The transitional matrix and attenuation are added with tunable and time-varying weights during the training to achieve better performance.

4.2.1 Why Attention with Transition States

The attention mechanism has been developed to improve the performance on long input sequence and especially for image recognition and Natural Language Prediction. The idea behind the attention mechanism is its ability to access encoder selectively during the decoding process achieved by the context vector. The context vector defined by Equation 15 is calculated based on the score given by Equation 16 using the probability distribution as shown in Equation 18.

$$\alpha_{ts} = \frac{exp\ (score\ (h_{t,}\bar{h}_s))}{\sum_{s'=1}^{S} exp\ (score\ (h_{t,}\bar{h}_{s'}))} \tag{18}$$

In image classification and Natural Language Prediction, the weights in Equation 16 are calculated throughout back propagation during the training. In a time-variant system, the back propagation suffers from vanishing gradient problem. The LSTM uses the concept of Backpropagation Through Time (BPTT) to avoid the vanishing gradient problem, but the context

and attention block is not part of the LSTM structure and suffers from the vanishing gradient problem. To this end, the transition matrix is formulated to provide the statistical information over long term data for the score and thereafter context vector calculation.

4.3 LSTM Model Switching using Random Forest

When there is multivariable input, the input to the model can be multivariable and the prediction accuracy can be improved. The dependency between input variables will enhance the prediction accuracy. But in most scenarios, the LSTM model is kind of black box where the relationship between variables is not clearly defined. When we investigated the correlation between beams, we are not able to get a correlation factor which provides us the evidence that a certain combination of input would provide a better result.

In our model, we use the Random Forest based LSTM model choosing which enables us to choose which LSTM model to use for the prediction which leads to better results. During the training phase we train all the possible variations using LSTM as shown in Figure 23 and feed the error into Random Forest along with the multivariable as input to the Random Forest.

With the trained model, now we predict initially using the Random Forest. The input of the multivariable is fed to the Random Forest, which will predict which LST model to use for better accuracy.

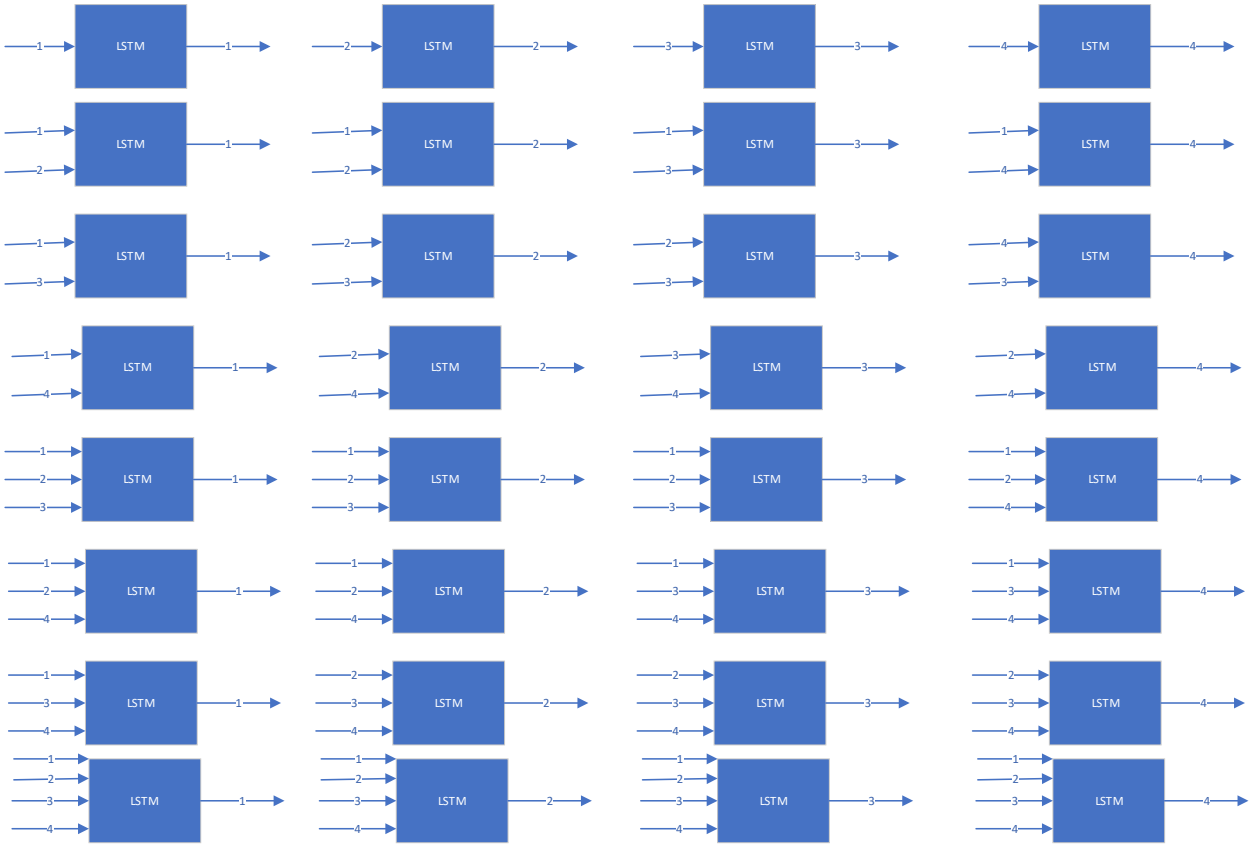


Figure 23 Various LSTM Combination

The model is designed as Training Model and Prediction Model where the training model can live in cloud which would be trained continuously as we collect data which would enhance the performance during prediction. And the prediction model can live in the vehicle and can be updated periodically by the cloud with its weights for the LSTM and the Random Forest prediction model.

Shown in Figure 24 and Figure 25 is the training model and the prediction model of the RF Based LSTM Model.

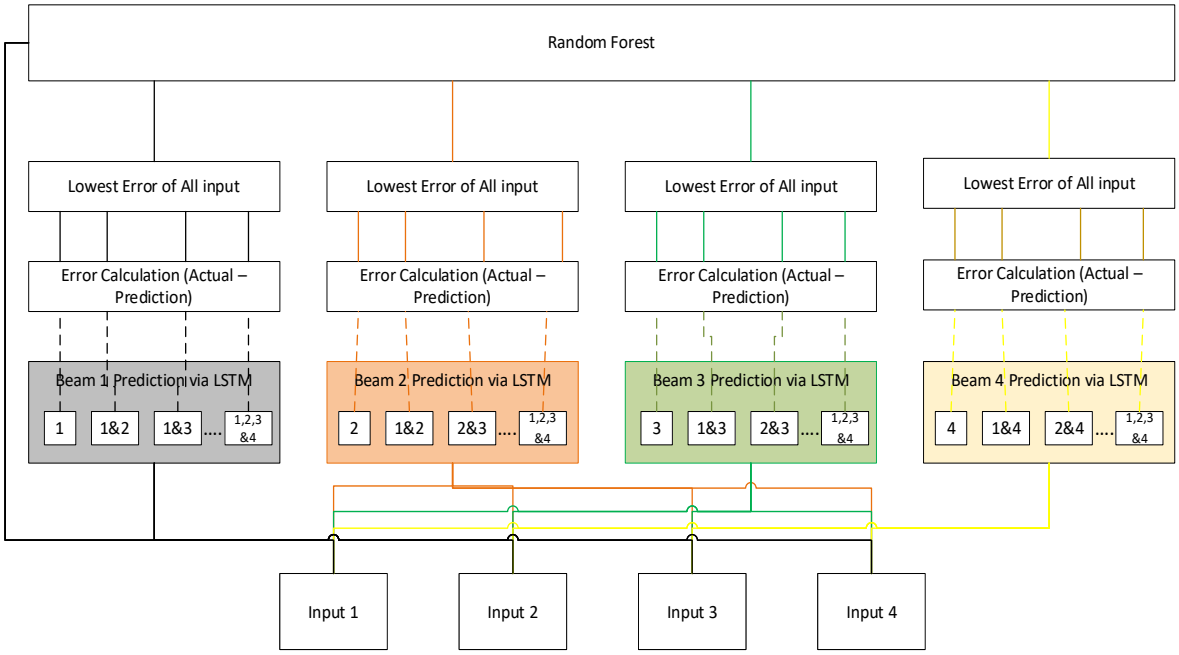


Figure 24 RF Based LSTM Training Model

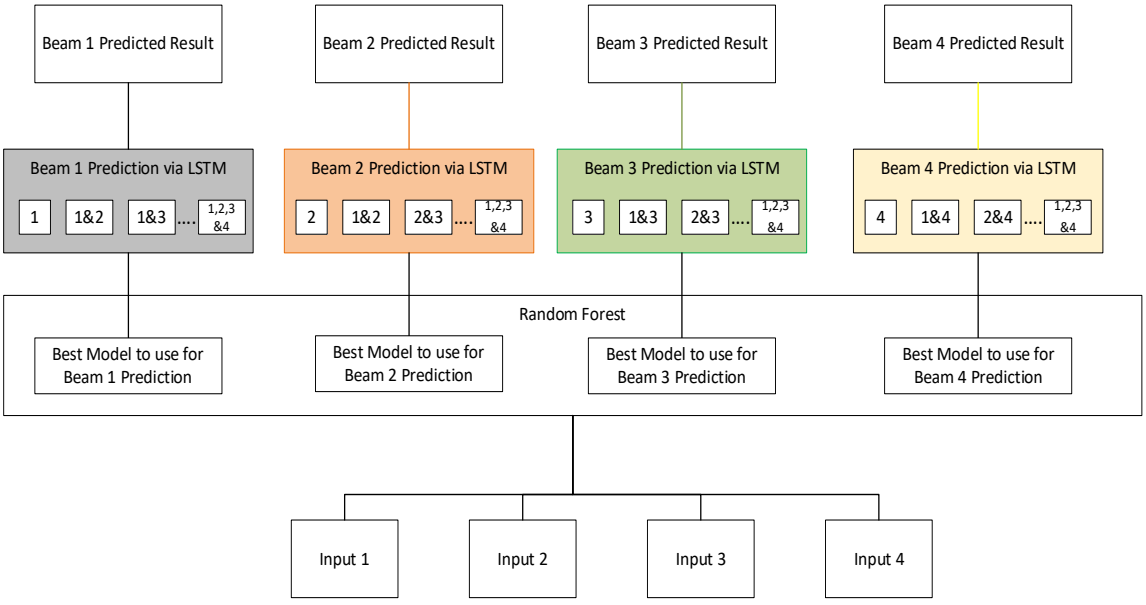


Figure 25 RF Based LSTM Prediction Model

During the training phase, a matrix is created with various combination of prediction output 1, 2, 3 and 4 with various inputs and this matrix is used for calculating the lowest error which would be fed as an input to the Random Forest.

Shown in Table 5 is the example of error matrix created based on various input and the various combinations.

Table 5 Error Matrix For Output 1

Input	Output	Error					
		[1,2…10]	[2,3…11]	[3,4….12]	[4,5….13]	…	[N-10, N-9…N]
1		Actual - Predicted	Actual - Predicted	Actual - Predicted	Actual - Predicted		Actual - Predicted
1&2		…	…	…	…		…
1&3		…	…	…	…		…
1&4	1	…	…	…	…		…
1,2,3		…	…	…	…		…
1,2,4		…	…	…	…		…
1,3,4		…	…	…	…		…
1,2,3,4		…	…	…	…		…

Like Table 5, an Error matrix would be created for all the Outputs which would create a huge matrix with various combinations of inputs and output for the various models which is shown in Figure 26.

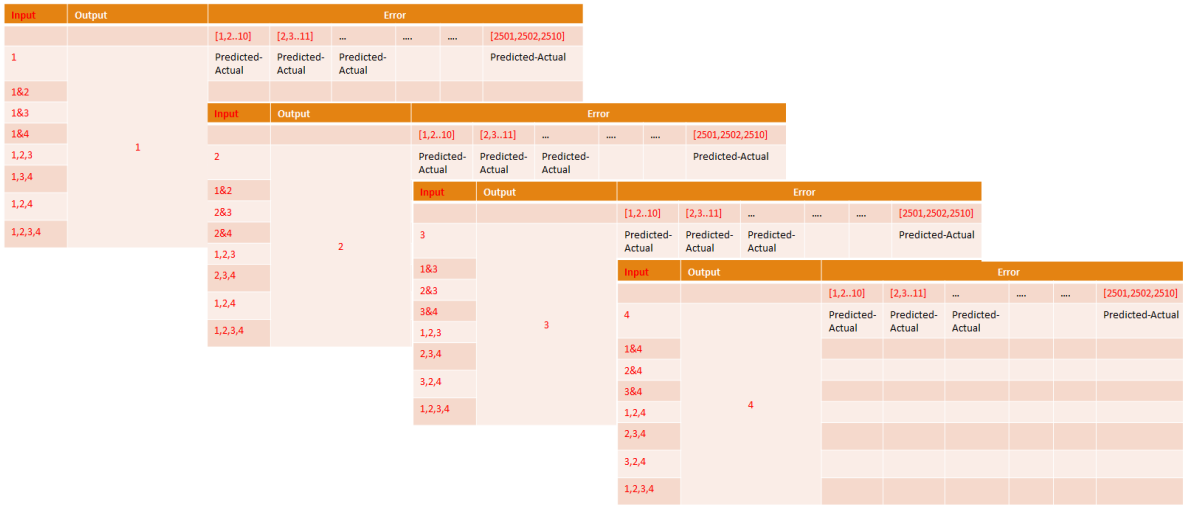


Figure 26 Error Matrix for all combinations

With the error matrix being created, we choose the lowest error from the input combinations and create another table which provides the lowest error from various combinations as shown in Figure 27. This error matrix creation with lowest error is performed for every output possible which leads to the final error matrix which would be the input to the Random Forest Model as shown in Table 6.

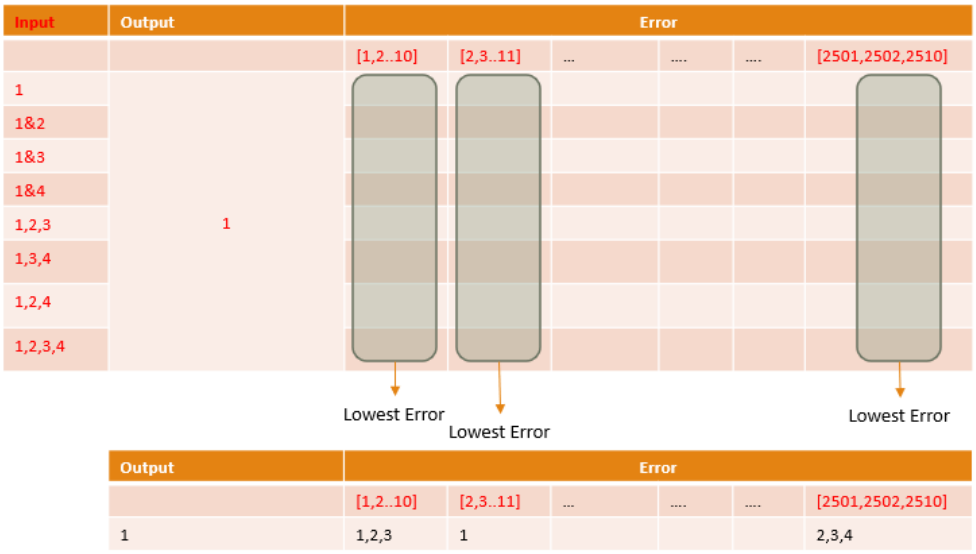


Figure 27 Errror Matrix creation with Lowest Error

Table 6 Final Error Matrix input for Random Forest

Output	Error			
	[1,2…10]	[2, 3…11]	…..	[N-10, N-9…N]
1	1,2,3	1	…..	2,3,4
2	1&2	1,2,3,4	…..	2,3,4
3	…..	….	….	….
4	…..	….	….	….

During the prediction, the input of various beams is fed as input to the random forest which looks at the various combinations of the beams and produces which combination is the best to predict which beam and provides that as the output. With the provided output, we can choose the right LSTM model and predict the signal strength which provides a better accuracy than feeding all the beam at all the time.

Chapter 5 Results and Validation

In this section we present the numerical results and analysis of the conventional, unconventional and our Intelligent Beam Selection Model. This section is split into Simulation analysis and the real-world data analysis where the simulation analysis investigates the comparison of the convention and the unconventional model and its effectiveness in time series prediction. In the real-world analysis, we will look at the performance results of some of the conventional model and our Intelligent Beam Selection Model.

5.1 Simulation Results and Analysis

In this section, we present experimental results that validate the applicability and robustness of the predication modules based on our simulation data. The metric used for the evaluation is the percentage of accuracy Equation 20 based on a 3dB margin. If the predicted result is within the 3dB threshold of the actual result Equation 19, the predicted result is considered as an accurate prediction, otherwise inaccurate.

$$Accurate\ prediction = \ |Predicton - Actual| \leq 3dB \tag{19}$$

Another metric is the mean absolute error (MAE), defined in Equation 21.

$$Accuracy = \frac{Total\ Number\ of\ accurate\ prediction}{Total\ Number\ of\ Prediction} \tag{20}$$

$$MAE_j = \frac{\sum_{i=1}^{n}(\hat{x}_{ij} - x_{ij})}{n} \tag{21}$$

5.1.1 Qualitative Analysis of Conventional and Unconventional Model

Figure 28 shows the performances of RSSI prediction generated by above predicators. Figure 28 (1) to (d) shows beam by beam prediction and Figure 28 (e) shows the strongest beam prediction. To test the robustness of the predication modules, Table 7 compares the variance of the prediction accuracy for Site 1.

Table 7 The Repetition test of Site 1

MODEL	Site 1 Accuracy (%)				
	Test 1	Test 2	Test 3	Test 4	Test 5
Adaptive Antenna Scanning	29.41	29.41	29.41	29.41	29.41
VAR	57.14	57.14	57.14	57.14	57.14
AR	58.49	58.49	58.49	58.49	58.49
LSTM	55.81	60.46	62.76	60.46	62.79
LSTM MIUO	72.09	81.39	69.76	72.09	69.77
LSTM VAR	53.48	58.13	62.79	60.46	58.13
LSTM AR	41.86	51.16	39.53	44.18	41.86

Seen from the Table 7, LSTM MIUO prediction achieves 72% efficiency compared to the 29%, 57%, 58%, 55%, 53% and 41% of adaptive antenna scanning, VAR, AR, LSTM, LSTM VAR, and LSTM AR respectively. Approximately ¾ of the packet will be received when adopting LSTM MIUO, whereas only $\frac{1}{3}$ of the packets would be received while using antenna scan method and only ½ of the packets would be received for VAR, AR, LSTM, LSTM VAR and LSTM AR method. The LSTM MIUO is proven to offer 24% improvement compared to any other method discussed here.

The MAE for a prediction across various beams is shown in Figure 29 where the LSTM MIUO has the lowest MAE compared to other methods.

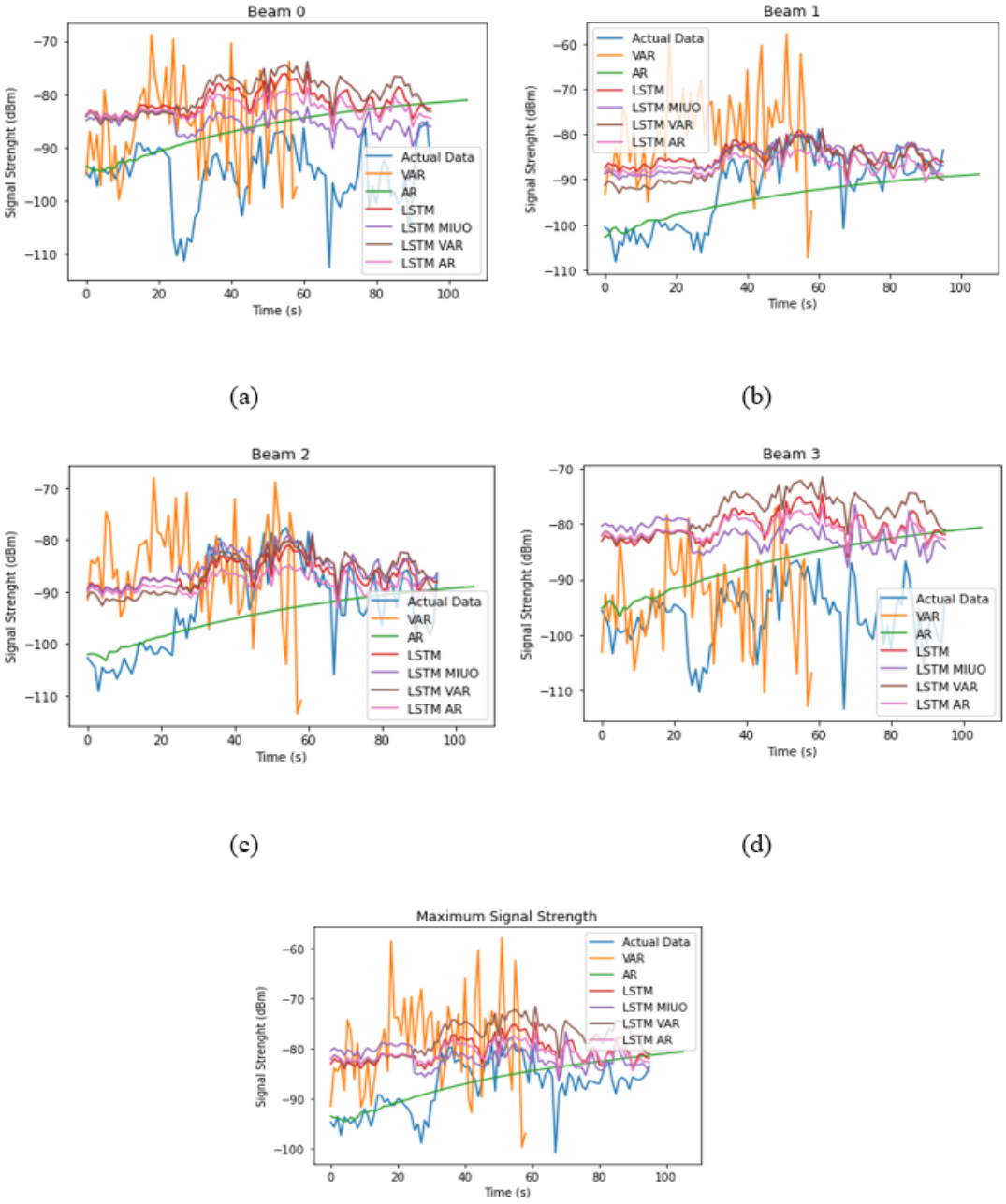


Figure 28 The Signal Strength Prediction for each beam and corresponding output

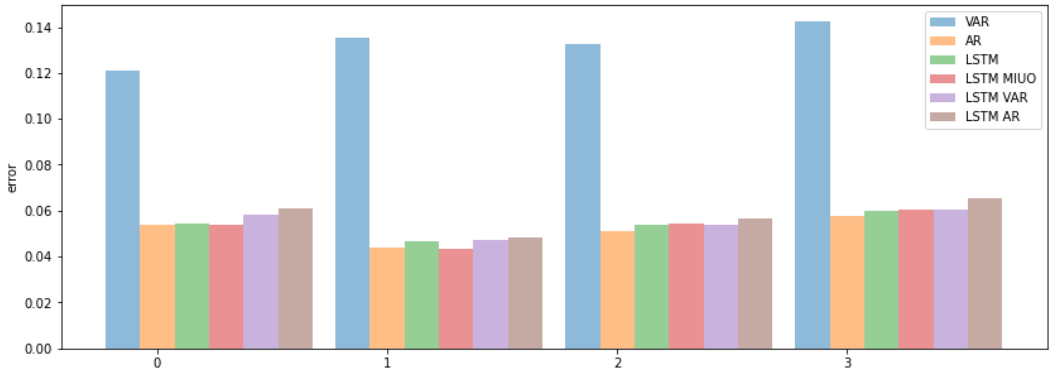


Figure 29 Comparison of MAE for Site 1 across various Methods

The performance of the LSTM MIUO starts during the training process where the loss function is calculated based on the prediction compared with the actual result and there by updating the weights accordingly. The final prediction is based on the weights assigned during the training set. As in the case of the LSTM MIUO, the weights are tuned to the individual signal strength of each beam i.e., a different set of weights is assigned to predict the signal strength of beam. Whereas in the LSTM system, the loss of the system is calculated based on all the signal strength across all the 4 beams combined and hence the weights are tuned based on all the signal strength at the same time which compromises on finding the best weights for all beams rather than individual in LSTM MIUO. Figure 30 shows the loss curves of the LSTM MIUO and the LSTM.

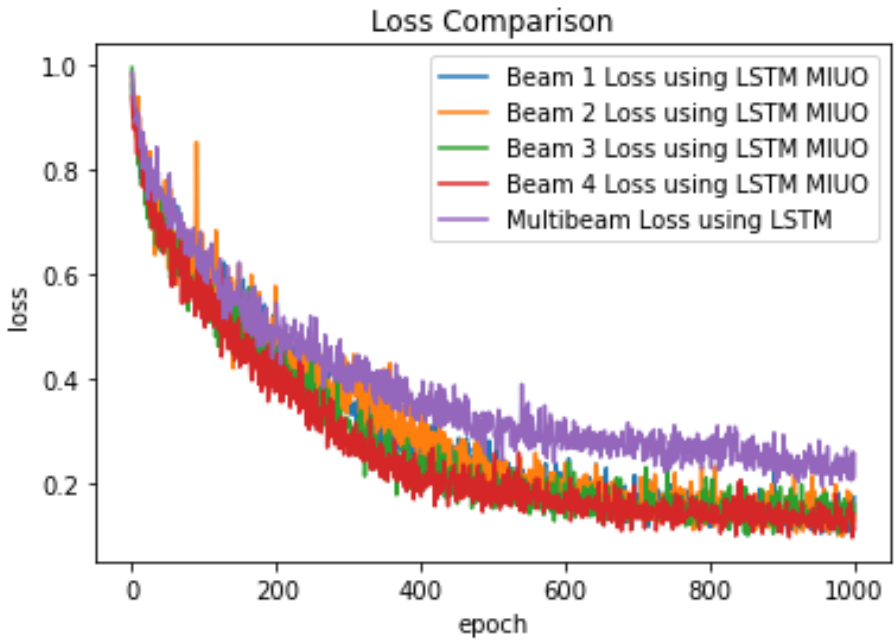


Figure 30 Loss of LSTM System between individual beam and the multibeam

It can be noticed that the loss plots of Figure 30 shows the variation in the loss. A small variation in the loss leads to better prediction efficiency since it tunes the weights significantly. The better fitted loss curve is expected to result in weights generating optimum output, which is the reason behind the performance gap between the LSTM MIUO and others. Table 8 shows the difference in the loss across various LSTM models i.e., the change of loss from the initial epoch to the final epoch calculated based on Equation 22.

$$Loss = |Highest\ Loss\ Value - Lowest\ Loss\ Value| \quad 22$$

Table 8 Loss Variation across all Models

MODEL	Beam	SITE 1 LOSS
LSTM	All Beam Combined	0.806
LSTM MIUO	Beam 1	0.853
	Beam 2	0.895
	Beam 3	0.859
	Beam 4	0.850
LSTM VAR	All Beam Combined	0.456
LSTM AR	All Beam Combined	0.118

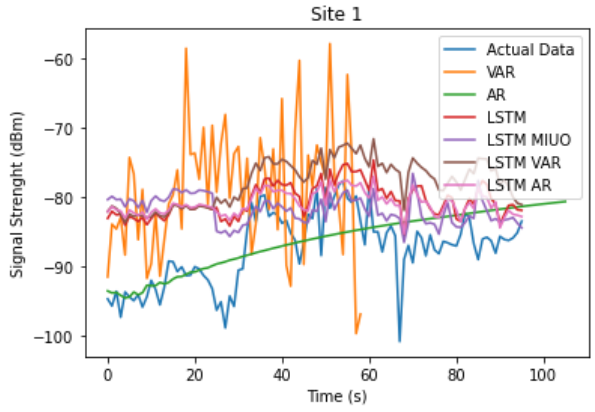
Table 8 shows the variation in the loss for all the models. The loss is more in the LSTM MIUO as it is tailored more towards each individual beam as well as the correlation among them, and thus resulting in superior to others during prediction.

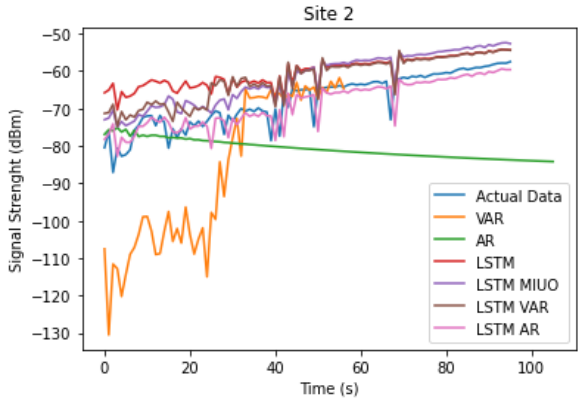
5.1.2 Quantitative Analysis of Conventional and Unconventional Model

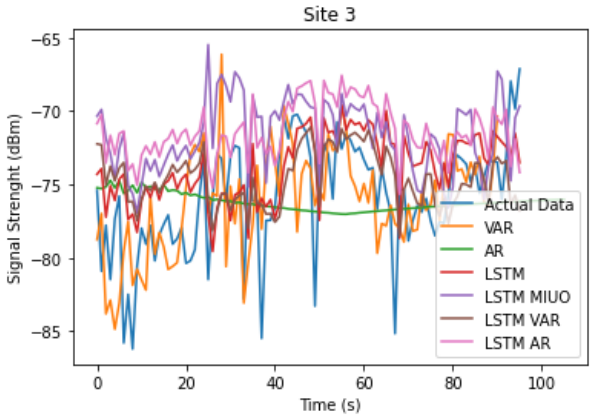
We also present the prediction results across various sites showing the performance of the LSTM MIUO. Figure 31 shows the highest signal strength prediction across the beams among varies sites located all around the campus. Table 9 list the percentage of accuracy across various models for various sites.

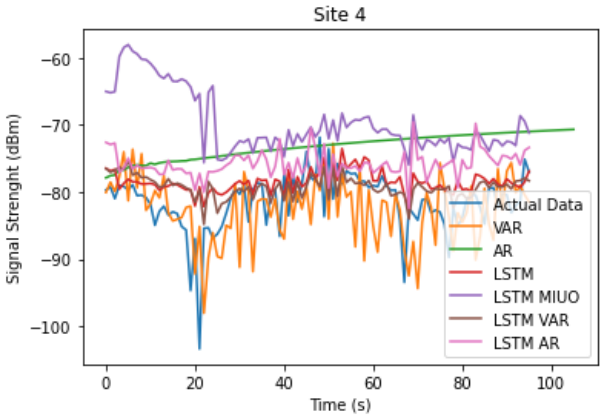
Table 9 Prediction Accuracy from Site 1 to Site 15

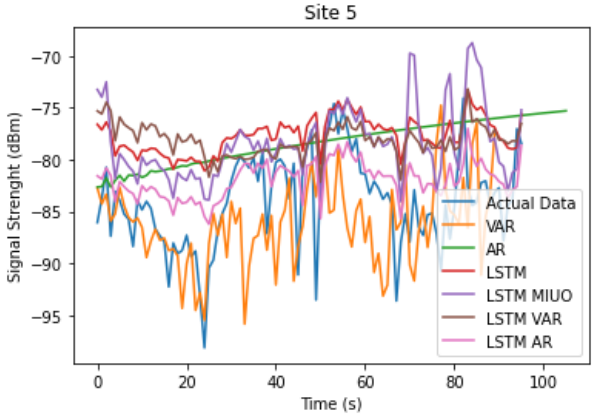
MODEL	Simulation Sites (Accuracy %)														
	Site 1	Site 2	Site 3	Site 4	Site 5	Site 6	Site 7	Site 8	Site 9	Site 10	Site 11	Site 12	Site 13	Site 14	Site 15
Adaptive Antenna scanning	29.41	35.29	41.17	23.52	29.41	26.47	38.23	38.23	38.23	17.64	32.35	41.17	20.58	50.0	32.35
VAR	57.14	95.83	64.1	55.31	63.0	76	65.30	80.95	69.56	71.42	63.04	60.78	66.0	76.47	66.66
AR	58.49	94.33	69.5	45.28	60.37	30.18	5.66	5.66	69.81	64.15	64.15	60.37	56.60	77.35	67.92
LSTM	60.46	97.67	55.81	41.86	51.16	58.13	6.97	74.41	79.06	69.76	67.44	55.81	53.48	72.09	60.46
LSTM MIUO	74.41	100	69.76	72.09	64.8	81.39	97.67	100	93.02	86.04	83.72	79.06	69.76	90.69	69.76
LSTM VAR	60.46	97.67	58.13	34.88	62.8	55.81	16.27	58.13	60.46	65.11	51.16	51.16	62.79	60.46	53.48
LSTM AR	51.16	95.34	53.48	44.18	53.5	51.16	25.58	46.51	62.79	60.46	41.86	48.83	69.66	74.41	51.16

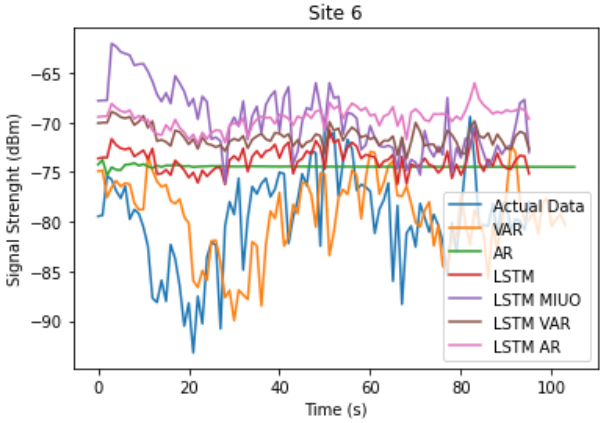


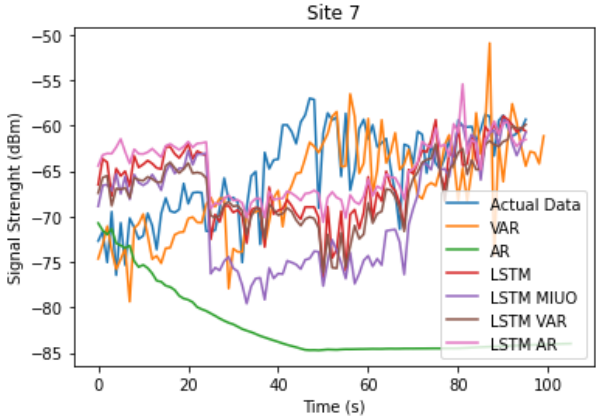


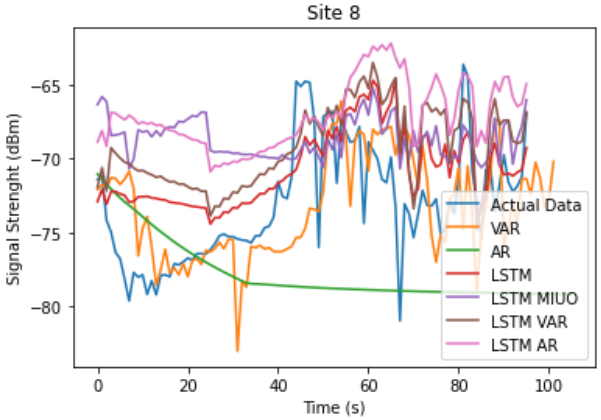

Site 3
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR

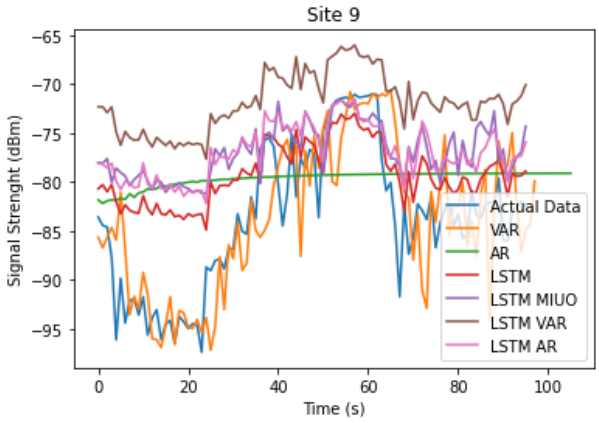

Site 4
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR

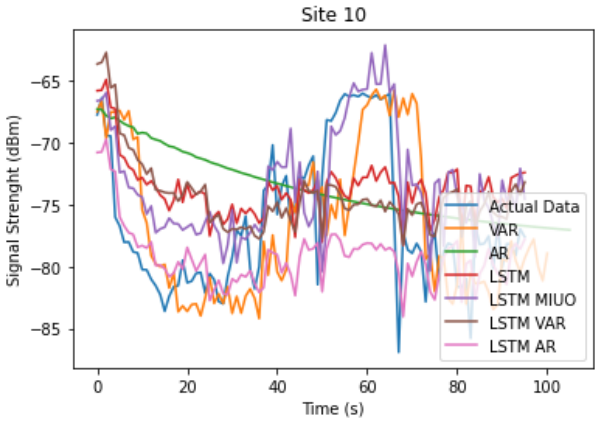

Site 5
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR

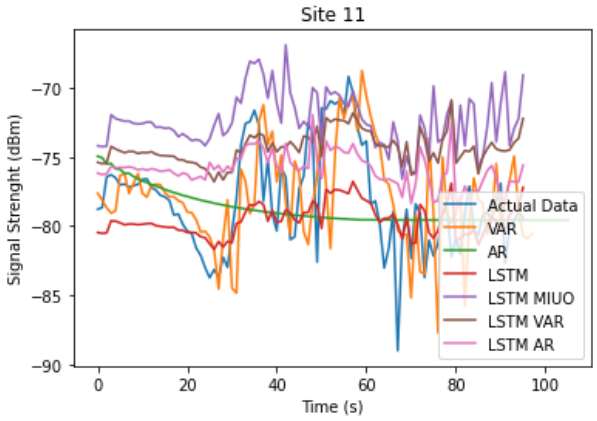

Site 6
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR

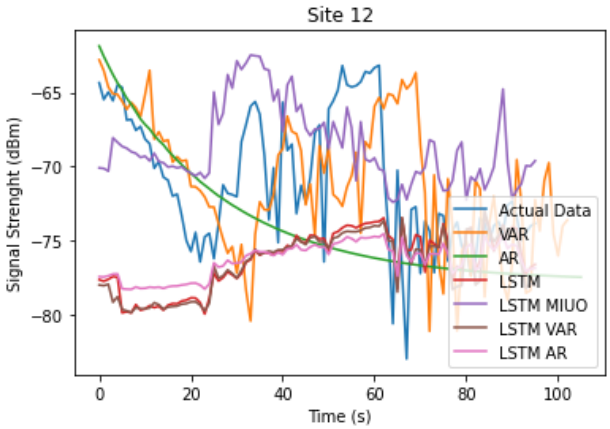

Site 7
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR

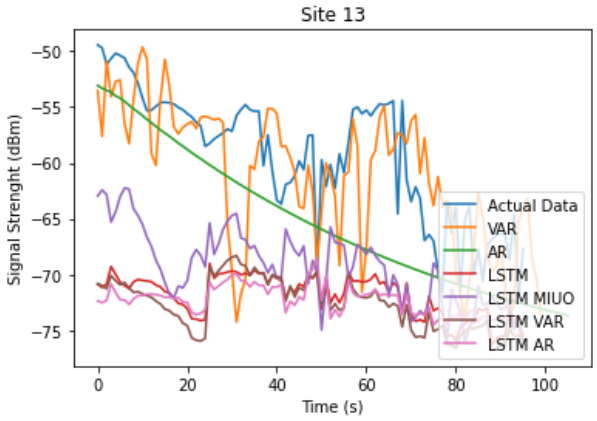

Site 8
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR

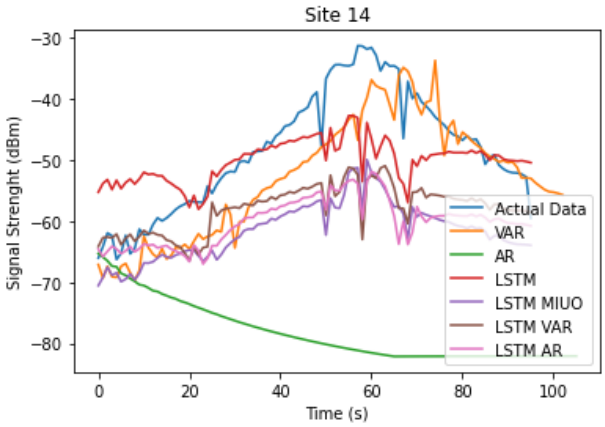

Site 9
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR


Site 10
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR


Site 11
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR


Site 12
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR


Site 13
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR


Site 14
Signal Strenght (dBm)
Time (s)
Actual Data
VAR
AR
LSTM
LSTM MIUO
LSTM VAR
LSTM AR

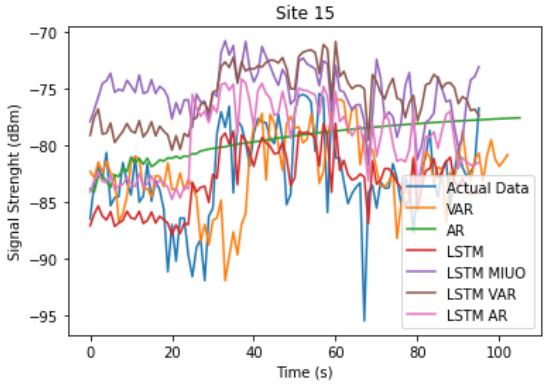


Figure 31 Display of Signal Strength across 15 sites

Seen from the Table 9 and the Figure 31, the LSTM MIUO outperform others across 15 sites. Antenna scanning method has the highest percentage error compared to any machine learning predicator resulting in two third packet loss as the beams are chosen once every 100ms.

The accuracy improvement of LSTM MIUO when compared with others is shown in Table 10 which is calculated from Equation 23. The comparison is between LSTM MIUO and the best prediction one of others. i.e., in Site 1 the LSTM MIUO is compared to AR. The results support our conclusion about performance of the LSTM MIUO.

$$\frac{|P_1 - P_2|}{\left[\frac{(P_1 + P_2)}{2}\right]} X\ 100 \qquad 23$$

where

P_1 is the accuracy percentage of LSTM MV

$P_2$is the accuracy percentage of best one of others.

Table 10 Enhancement of prediction accuracy when adopting LSTM MIUO

SITE	Accuracy	SITE	Accuracy	SITE	Accuracy
1	23.9%	6	6.8%	11	21.5%
2	2.3%	7	39.7%	12	26.1%
3	0.4%	8	21%	13	0.2%
4	26.3%	9	16.2%	14	19.7%
5	1.7%	10	18.5%	15	2.7%

5.1.3 Time Series Ensemble Analysis

In this section, we present experimental results that demonstrates the improved accuracy of the Time Series Ensemble (TSE) algorithm. The main metrics used for evaluation are the Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and Absolute Precision Error (APE). RMSE and MAE captures the absolute error, MAPE captures the percentage error and APE is the standard deviation of the set of measurements.

The power level (RSSI) recorded during the simulation is used as the dataset and the prediction is performed. The prediction is performed on three different models.

- ARIMA
- LSTM
- Time Series Ensemble (TSE)

The results of all the three models are compared to show the performance of the TSE method.

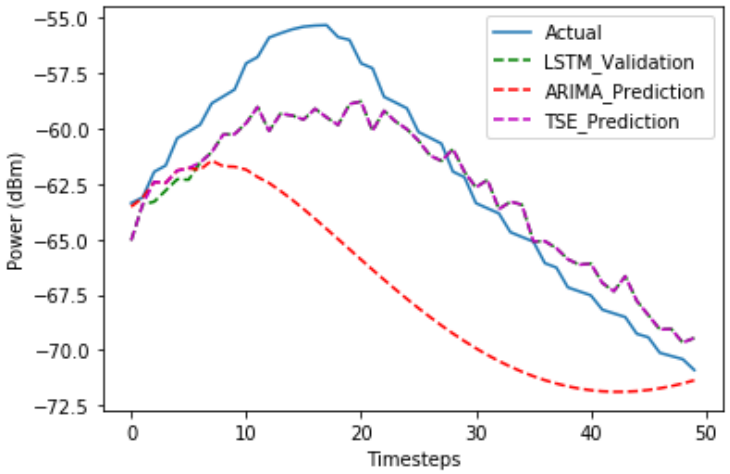


Figure 32 RSSI Prediction of TSE

Figure 32 shows the prediction result of the RSSI data over three different models and the actual result. It can be clearly noted that ARIMA prediction is better in a near short term and as the timesteps increases the ARIMA prediction error increases. Similarly, in the LSTM it is identified that the near short-term prediction error is higher than the ARIMA, but as the timestep increases the prediction gets better than ARIMA. In our TSE algorithm, the prediction is best in both near short term and in long term as the timestep increases, thus increasing the accuracy of the prediction and reducing the error from the actual value.

The analysis is also performed on the Angle of arrival (AoA) i.e., the angle which the signal is received by the vehicle. The Angle of arrival data is collected during the simulation and the prediction is performed. Evaluation has been performed previously for Angular information by [65]. Figure 33 shows the same performance as power level prediction where TSE method performs better than standard ARIMA and LSTM in both short term and long term.

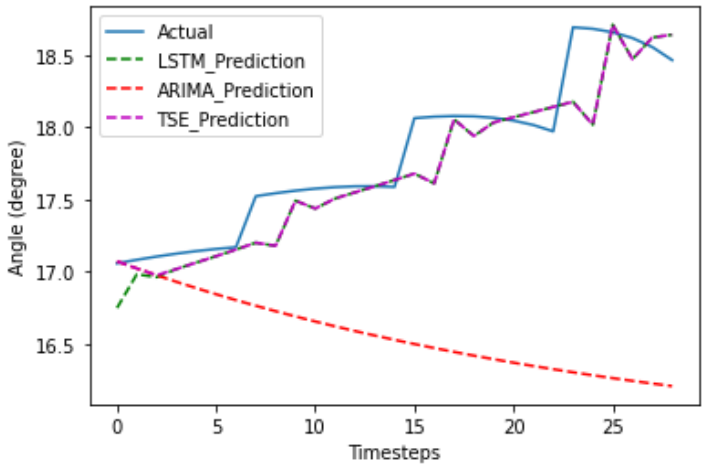


Figure 33 Angle of Arrival TSE Prediction

Table 11 shows the error calculation and the comparison between the ARIMA, LSTM and the TSE Algorithm. From the various error calculation, it can be proven that the TSE algorithm outperforms both ARIMA and the LSTM in time series prediction.

Table 11 TSE Error Comparison

Algorithm / Error	Angle Prediction			Power Level Prediction		
	ARIMA	LSTM	TSE	ARIMA	LSTM	TSE
RMSE	9.99	19.28	9.71	8.08	6.08	6.05
MAPE	25.15	35.22	25.48	9.83	7.82	7.79
APE	8.85	18.30	8.53	6.53	3.69	3.64
MAE	9.58	14.13	8.40	5.12	1.91	1.82

5.2 Theoretical Analysis

To validate the proposed model, we generated a theoretical data set of Antenna Beam 1 to 4 with a total data set length of 1500 with the following probability conditions.

Table 12 Theoretical Data Set Condition

Beam	Beam 1_i	Beam 2_i	Beam 3_i	Beam 4_i
Beam 1_{i-1}	0.1	0.2	0.3	0.4
Beam 2_{i-1}	0.4	0.1	0.2	0.3
Beam 3_{i-1}	0.3	0.4	0.1	0.2
Beam 4_{i-1}	0.2	0.3	0.4	0.1

Table 12 shows the condition of how the data set has been generated to validate this model. For example, if Beam 1 is present beam, the probability of next data to be Beam 1 to Beam 4 are 0.1, 0.2, 0.3 and 0.4 respectively.

The generated dataset is uniformly distributed i.e., if a random number is chosen as a prediction, there is a 0.25 probability that the random number is correct i.e., the accuracy is 25% If the transitional matrix is known and is still applicable to future dataset, maximum likelihood estimate can be adopted to achieve the best estimate. Based on the generated dataset the theoretical maximum likelihood is 0.4 i.e., 40% accuracy. This estimate is based on the factor that the previous estimation Beami is correct, or we provide the actual data (Beami) for every Beami+1 prediction. Whereas in the prediction method we always feed the predicted value to predict the next Beam i.e., Beami is predicted and the predicted Beami is fed as an input to predict Beami+1.

Simulation is performed to see the performance of the maximum likelihood where the input Beami is also predicted value which is considered as a known value to predict Beami+1 i.e., unguided methodology. The total dataset is 1500 and we considered the last 200 as the test data. The last known value i.e., dataset 1300 is Beam 3 which is considered as Beami to predict Beami+1. Based on the table Beami+1 would be Beam 2 due to 0.4 probability. For the next

prediction we used Beam 2 as the input and predicted Beam 1 based on 0.4 probability. This has been simulated and the accuracy is calculated as 26.5%. Figure 34 shows an example for the difference between guided and unguided methodology based on the Table 12 prediction. It's shown that in the guided methodology, the probability of next Beam is always based on the true data (Example Data) whereas in the unguided methodology, the probability of next Beam is based on previous estimate.

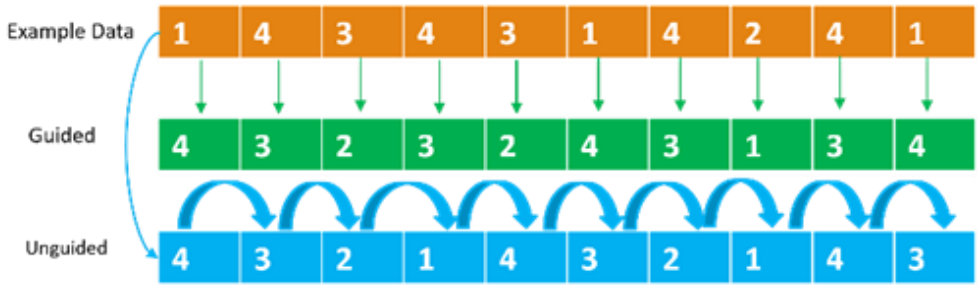


Figure 34 Guided Vs Unguided likelihood estimation

Based on the generated dataset, the analysis is performed on the most "naïve" forecast which is the persistence algorithm or Walk-Forward validation. The persistence algorithm uses the value at the previous time step (t-1) to predict the expected outcome at the next time step (t+1). We have also performed analysis on our proposed Attention with Transition model and compared with Encoder Decoder with Attention model, both Dot product and Luong's method of implementation. In the decoder model, during the prediction of the test data, the input provided to the attention sub model is the actual predicted values i.e., unguided methodology. Based on this method, the percentage of accuracy is calculated to show the improvement of results.

- Theoretical Random selection: 25%
- Maximum likelihood
- Theoretical Guided: 40%

- Unguided: 26.5%
- Walk-Forward Validation (Persistence Prediction): 8.82%
- Encoder Decoder with Attention (Dot product): 23.65 %
- Encoder Decoder with Attention (Luong's Method): 24.85 %
- Encoder Decoder with Attention (Attention with Transition): 28.35 %

It can be noted that in the theoretical maximum likelihood has 40% prediction accuracy, but it's a theoretical analysis and there are other factors which contribute to this method. We need to know the input to have the better prediction. When we compare the actual prediction model, the analysis showed significant improvement in the accuracy of prediction, where we see close to 12% (28.35 / 23.65) improvement than Encoder decoder with Attention method.

Along with the percentage of accuracy, we also performed Mean Squared Error, Mean Absolute Error (MAE) (Equation 24) and Mean Absolute Percentage Error (MAPE) (Equation 25) metric to see the performance of the proposed model. MSE captures the difference between the original the predicted value whereas MAPE captures the absolute error of the prediction and MAPE captures the percentage error.

$$MAE_j = \frac{\sum_{i=1}^{n}(\hat{x}_{ij} - x_{ij})}{n} \quad 24$$

$$MAPE_j = \frac{\sum_{i=1}^{n}\left|\frac{(x_{ij} - \hat{x}_{ij})}{x_{ij}}\right|}{n} \quad 25$$

Table 13 Performance Comparison of Theoretical Data Set

Error	MSE	MAE	MAPE
Walk-Forward Validation (Persistence Prediction)	1.30	0.90	0.47
Dot product	1.42	0.95	0.65
Luong's Method	1.41	0.93	0.62
Attention with Transition	0.99	0.74	0.38

Table 13 it can be noted that MSE, MAE and MAPE is lowest in our proposed method. The improved performance of the system is because the weights are determined by the transitional state matrix. During the attention part, the transitional state value provides input to the attention where the focus of the decoder should be. In the traditional encoder decoder with attention, the training part determines which encoder part the decoder should focus on, so that the decoder decodes the data based on the attention value. Whereas in our method, the transitional state provides input to the attention state which provides the focus to the decoder and providing the information of which encoder the attention or focus needs to be for the decoder so that the predicted value is similar to the actual value. By providing the attention weights the prediction results are much better than the traditional method.

The main motivation of the attention is at different steps, the decoder needs to focus on different source which are relevant at that step. The attention score is the "relevance" of the encoder state to the decoder state. The attention score transforms to attention output which is the weighted sum of the attention weights. The variability in attention score adds up for the attention output. The lesser in variability provides clear definition of which transition encoder to focus on.

When the attention score is taken closer look as shown in Figure 35, it can be noted that the variability of the attention score is very small in our Attention with transition method compared to the Luong's method. The variability of the attention score for the Luong's method is 237.4 with the lowest value to be -217.01 and the highest value to be -20.42 whereas in Attention with Transition the variability of the attention score is 30.9 with the lowest value to be -20.01 and the highest value to be 10.89. The reason for the variability is the weights being assigned randomly in the Luong's method whereas in our Attention with Transition method, the weights are determined based on the known data of transition which provides better relevance of the encoder to the decoder state. The attention score provides better capability for the decoder to focus on the right source and leading to better predictability.

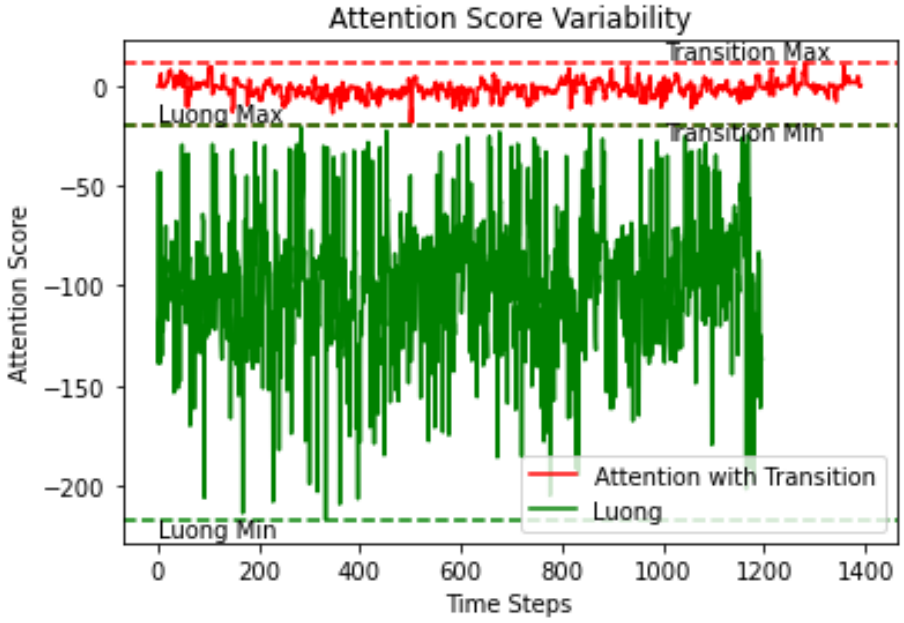


Figure 35 Attenuation Score Variability between Luong and Attention

5.3 Experimental Analysis

5.3.1 Qualitative Analysis

The experiment is performed over the collected data sample. The algorithm is compared with the Encoder decoder with Attention model, both Luong's and Dot product to show the improvement of our system compared to the Luong's method of implementation. The analysis is performed like the theoretical analysis and shows a consistent performance i.e., improved results in the Attention with Transition model on both theoretical and measured data.

- Walk-Forward Validation (Persistence Prediction): 25.76%
- Encoder Decoder with Attention (Dot product): 36.91 %
- Encoder Decoder with Attention (Luong's Method): 39.82 %
- Encoder Decoder with Attention (Attention with Transition): 42.44 %

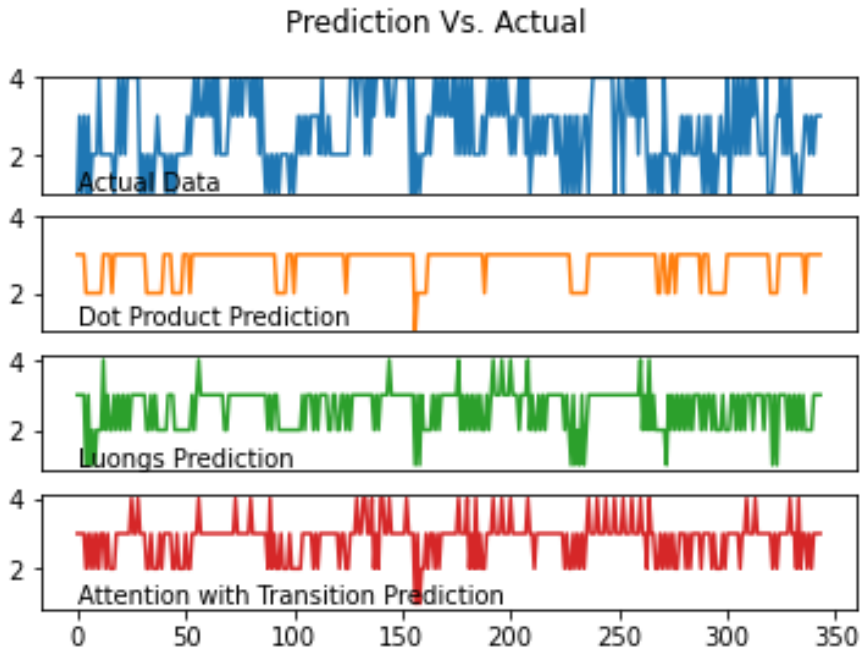


Figure 36 Comparison of Prediction results from Dot, Luong and Attention

Figure 36 shows the prediction results of various models. It can be noted that our proposed method has significantly better performance of predicting the beam compared to the traditional

Dot product method and the Luong's method. We show an improvement of 11.5% from the traditional dot product and 10.5% for the Luong's method.

The loss curve shown in Figure 37 indicates that the training is better and attains better stabilization quicker using our proposed model.

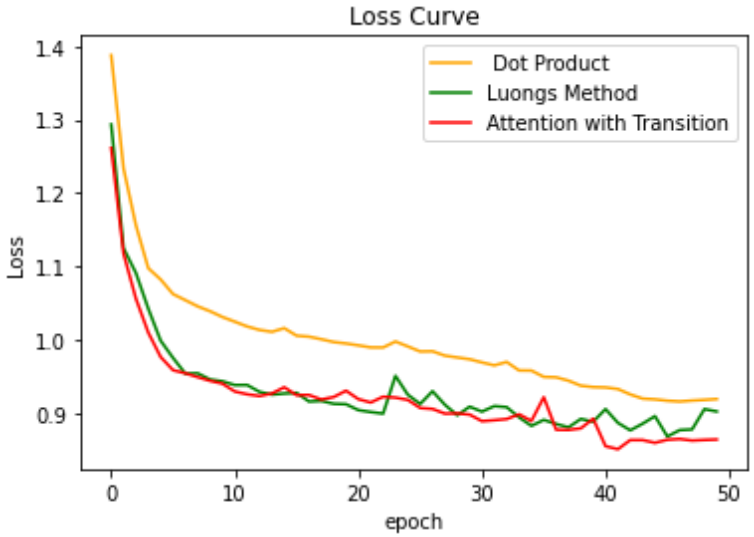


Figure 37 Comparison of Loss Curve

The Attention vector is the score of the corresponding value within the source sequence which tell the decoder what to focus on at each time step. A huge variability in the Attention score provides lower confidence in the decoder which results in choosing the wrong encoder to focus the prediction on. In our test data analysis, the variability of the attention score is considerably lower when compared with the Luong method as shown in Figure 38. The variability in attention score for Luong's prediction is 128.5 whereas the variability in attention score value for Attention with Transition prediction is 35.8, which provides us the better confidence of predicting the value by focusing on the right encoder during the prediction.

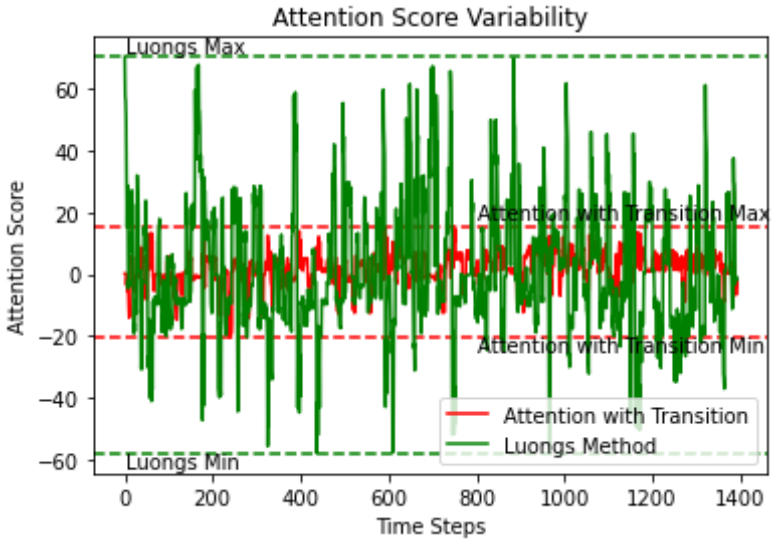


Figure 38 Attention Score Variability

The accuracy plots show in Figure 39 indicate the accuracies from the dot product, Luong's method, and attention with transition as 35.5%, 40.3% and 42.1% respectively. This is during the training phase over 50 epochs where the losses have achieved its lowest levels and the accuracies are at their peaks.

Figure 39 Comparison of Accuracy Curves

Along with the percentage of accuracy, we also performed MSE, MAE and MAPE metric to see the performance of the proposed model.

Table 14 Performance Comparison of Measured Data

Error	MSE	MAE	MAPE
Walk-Forward Validation (Persistence Prediction)	1.719	1.04	0.37
Dot product	0.92	0.72	0.40
Luong's Method	0.99	0.74	0.38
Attention with Transition	0.91	0.68	0.35

From Table 14, it can be noted that MSE, MAE and MAPE is lowest in our proposed method. The prediction results shows that Attention with Transition has a better prediction accuracy compared to other traditional prediction methods.

If a dataset is uniformly distributed, then the random selection of data will result in 25% accuracy i.e., if a data is chosen randomly the probability of getting the right Beam is 25%. Based on this, we can say that if the dataset is uniformly distributed, then the accuracy of random selection would be 25%. In our dataset, the Beam data are not uniformly distributed, and the accuracy will not be 25%. In this dataset, as shown in Figure 40, the total number of Beam 1 is 17% of the data set, Beam 2 is 37% of the data set, whereas Beam 3 is 21% of the data set and Beam 4 is 25% of the data set. If the random selection is Beam 1, the probability of getting it correct is 17% and if the random selection is Beam 2, the probability of getting it correct is 37% and so on with Beam 3 is 21% and Beam 4 is 25%. When this accuracy is compared with our prediction method, we should outperform these accuracies or else the random selection is a better method than the machine learning prediction. When we analyze our predicted data, the probability of Beam 1 prediction is 80% i.e., 80% of the Beam 1 prediction is correct whereas when we randomly choose there is a probability of only 17%. Similarly, the probability of Beam 2 is 55%

whereas the random selection is 37%, probability of Beam 3 is 29% whereas the random selection is 21% and the probability of Beam 4 is 68% whereas the random selection is 25%. Table 15 shows the prediction probability comparison between the random selection, and our prediction method which shows that our prediction method performs better than the random selection in all individual beam selection method.

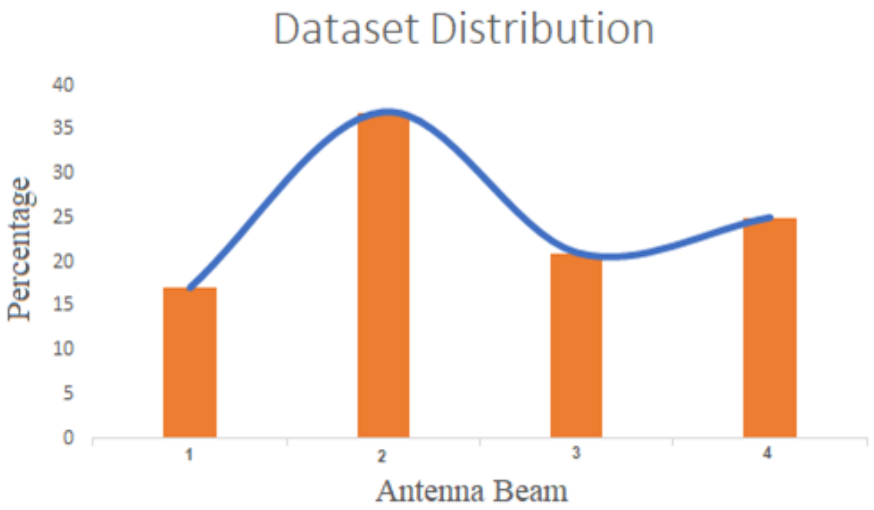


Figure 40 Dataset distribution percentage

Table 15 Random Slection Vs. Attention with Transition

BEAM	RANDOM SELECTION	ATTENTION WITH TRANSITION
Beam 1	0.17	0.45
Beam 2	0.37	0.47
Beam 3	0.21	0.31
Beam 4	0.25	0.48

During the test data prediction, instead of feeding the predicted values as input to the next decoder loop, if we provide the actual data to the next decoder loop i.e., guided methodology, the

accuracy percentage improves and provides us an accuracy of 46.9%. This method will provide better efficiency of prediction if we know the output values during the testing stage.

5.3.2 Quantitative Analysis

To validate the model across various dataset, we also collected data from different drive zones around the campus as shown in Figure 41 and shown the analysis of the various dataset across the different encoder decoder models.

The performance comparison of various zones is shown below. The analysis indicates that Attention with Transition (Our proposed) model performed better than the traditional Encoder decoder model.

Figure 41 Various drive Path (Left to Right) Zone 1, 2 and 3

Table 16 Performance comparison of Various zones

Drive Zone	Method	Accuracy (%)	MAPE	MSE
Zone 1	Persistence Prediction	25.76	0.37	1.71
	Dot product	36.91	0.40	0.92
	Luong's Method	39.82	0.38	0.99
	Attention with Transition	42.22	0.35	0.91
Zone 2	Persistence Prediction	23.48	0.47	2.30
	Dot product	26.12	0.46	1.52
	Luong's Method	26.75	0.50	1.46
	Attention with Transition	29.19	0.44	1.42
Zone 3	Persistence Prediction	25.51	0.53	2.44
	Dot product	27.19	0.66	1.41
	Luong's Method	29.38	0.54	1.32
	Attention with Transition	32.89	0.51	1.24

It shall be noted from Table 17 the performance improvement from the Dot product Vs. Attention with Transition and Luong's Method Vs. Attention with Transition. The performance improvement is calculated from the accuracy percentage as explained in Equation 26. The variance in the improvement as seen is dependent on the dataset. Based on our dataset the variance is between 10 to 12% improvement.

$$Performance\ Improvement = \frac{Accuracy\ of\ Attention\ with\ Transition}{Accurancy\ of\ Dot\ product\ or\ Luong\ Method} \quad 26$$

Table 17 Performance improvement comparsion

Drive Zone	Performance Improvement	
	Dot Vs Attention with Transition (%)	Luong Vs Attention with Transition (%)
Zone 1	11.43	10.60
Zone 2	10.91	10.91
Zone 3	12.09	11.19

The accuracy of the prediction depends upon the dataset and the prediction accuracy falls with the entropy of the dataset. The entropy provides the information about the randomness on the dataset and our model prediction result follows the entropy of the dataset as well. The entropy is calculated as shown in Equation 27.

$$H(X) = -\sum_{i=1}^{n} P(x_i) log_b P(x_i) \quad 27$$

The entropy is calculated for the theoretical data and for all the three measured zones and their corresponding accuracy is plotted in Figure 42. It shall be noted that as the Entropy increases the accuracy of prediction decreases which correlates to the Shannon theory.

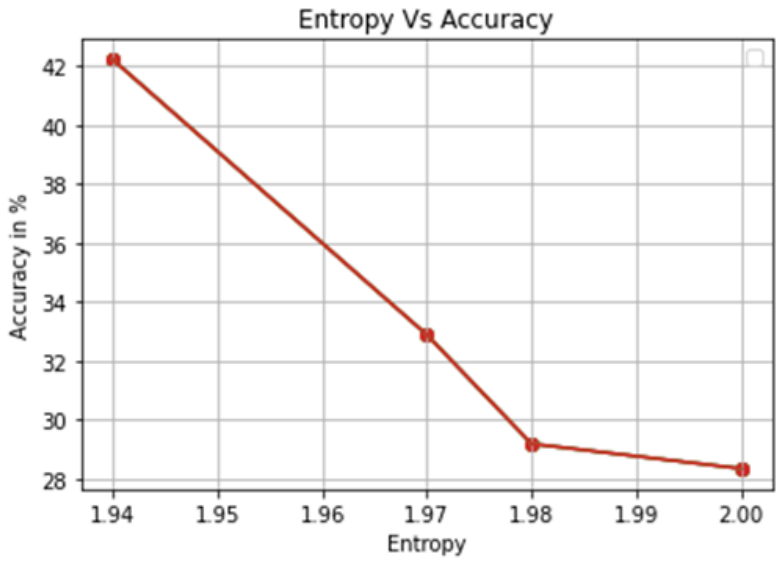


Figure 42 Entropy Vs Accuracy

5.3.3 RF Based LSTM

The prediction accuracy is calculated over the collected data and compared with the theoretical random selection which is selecting randomly one of the four beams. Based on the dataset which has been collected around University of Michigan – Dearborn campus the results shows that the RF based LSTM has a better prediction accuracy than a standard LSTM with multivariable input.

- Theoretical Random Selection : 25%
- LSTM : 31.92%
- RFLSTM : 35.54%

From the results, it is evident that the RF Based LSTM outperforms the standard method of prediction. Figure 43 shows the prediction result comparison between the RF based LSTM with the standard LSTM along with the baseline measurement.

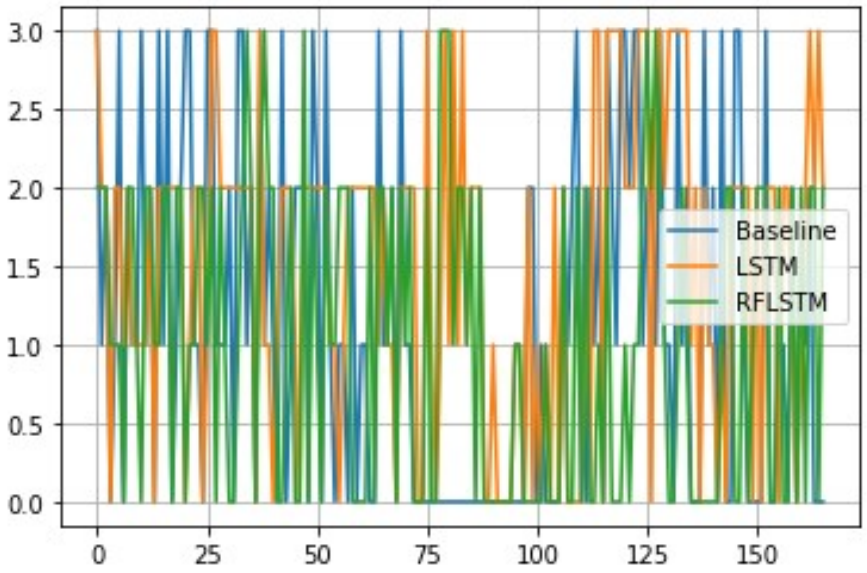


Figure 43 Comparision between RFBased LSTM and LSTM

To compare the performance improvement from the we use the Equation 26 which shows the performance improvement compared to other methods which is shown in Table 18.

Table 18 Performance Improvement of RFLSTM

Drive Location	Performance Improvement	
	Random Selection Vs. RFLSTM (%)	LSTM Vs. RFLSTM (%)
Campus	14.21	11.13

Chapter 6 Conclusion

This research proposes a novel beam selection named “Intelligent Beam Selection” (IBS) which enhances the real time prediction of beam by considering all the signal strength from various beams. A multi-agency LSTM beamforming prediction model is presented resulting in enhance performance than conventional ones including ARIMA and LSTM. The effectiveness of such model is verified through C-V2X simulation results in the University campus which resulted in prediction improvement of 24% compared to other prediction methods. The robustness of the performance of the LSTM MIUO model across various sites signals its widespread applicability and repeatability.

A new Encoder Decoder modified hard attention is shown resulting in enhance performance than the conventional one including Encoder Decoder with Attention (Dot product and Luong’s method). The effectiveness of such model is verified using actual test data which was taken at the university campus using the antenna array which was designed for this application. We hope that the results of this paper will encourage future work in using modified hard attention. We also expect that the modularity of the encoder-decoder approach combined with modified attention to have useful applications in other domains

Chapter 7 Future Work

This research was performed with real-time datasets and valid assumptions were taken to develop the concept of the new drive mode. The validation was conducted with specific limitations which produced best results as described in the Chapter 5. Hence in this chapter, the following ideas were proposed which could augment the deployment of the Intelligent Beam Selection (IBS).

7.1 Multi-variate System

This research is focused on the signal strength data, but there are other data which shall be used to improve the accuracy of the prediction system.

7.1.1 GPS Data

The GPS Data will provide information about the location of the vehicle with respect to the transmitter which would enhance the model predictability. By knowing the transmitter location and understanding the receiver location, the model could make a predetermination of the beam or use the previous prediction results and further train the model for better accuracy

7.1.2 Vehicle Speed

The Vehicle speed will provide the information of the doppler shift on the frequency of the signal. The vehicle speed provides a determination of how fast the beam selection needs to be made and how often the beams need to be switched. The vehicle speed will also provide information of the training data set of how often the data is collected and what is the corresponding distance between two data collected from the same beam.

7.1.3 Foliage Condition

Foliage plays a vital role in the Radio Frequency performance. The radiation of the signal varies a lot based on the Foliage condition. It has been noted by much research that the RF performance or radiation performance varies based on Foliage condition. The coverage region varies a lot based on the Foliage such as if the trees are blocking the radiation. This information will enhance the efficiency of prediction of any model.

7.1.4 Traffic

The traffic condition will help us to understand the congestion factor around the vehicle and the effect of noise on the receive signal. If the traffic is high in a location, the Signal to Noise Ratio (SNR) would be high due to the higher noise factor. The traffic condition also helps in training as it might be one of the feature inputs to the model which shall enhance the prediction accuracy.

7.1.5 Urban Vs Rural

Like the traffic condition, the drivable condition of the location such as Urban or Rural would provide information to the prediction model. In an urban scenario, the vehicle movement would be higher compared to the rural condition and the selection of beam would be better and also helps in the prediction model as the noise factor might be higher in the Urban compared to Rural.

7.1.6 Drive Terrain

Drive terrain would be a big impact to the prediction model such as mountain or uphill or downhill driving. Based on the drive terrain the radiation pattern changes and the accumulation of signal will have a huge impact on the signal strength. The Drive terrain would provide the

information why the signal strength varies a lot within a quick session, and it helps in the prediction model. And the drive terrain would provide the information that the beam would not be an optimal selection as the beam would focus more energy which might not be received by the source or the receiver. So, the model would make a prediction of switching to omnidirectional rather than beam in scenarios based on drive terrain.

7.2 Cloud Processing

The model can be stored in the cloud which would enable faster processing and computational time. The cloud model would have the ability to store large data set and improve the training scenario which leads to better prediction accuracy of the system. The cloud processing can also create multiple models based on multiple scenarios as explained in the multi-variate system under Future Work.

7.3 Transmitter Beam

The test has been performed considering the transmitter to be an omnidirectional antenna and the receiver to be a beamforming and the model has been trained in the vehicle to predict the right beam for the vehicle. The future work can also involve the transmitter to be a beamforming antenna which enables the range of the entire communication, and the model would be able to predict both the transmitter beam and the receiver beam. In this scenario, the model can be synchronized where the information is shared by the model from the transmitter to the receiver and vice-versa so that the prediction model can be enhanced by knowing what the Transmitter beam will be and the receiver beam would be.

Appendices

Appendix A: Simulation

A.1: Simulation Setup

The simulation is performed in Altair Winprop – ProMan software which provides the 3D simulation of the location. The simulation is performed in the University of Michigan-Dearborn campus with the prediction area covering the entire campus and the neighboring places to make sure the coverage data is not lost.

Before the execution of the simulation, the path and the material properties of the environment is setup using the Altair Winprop -WallMan software which allows us to set the heights of the properties around the campus. Shown in Figure 44 the various properties of the materials constructed during the simulation and the layout.

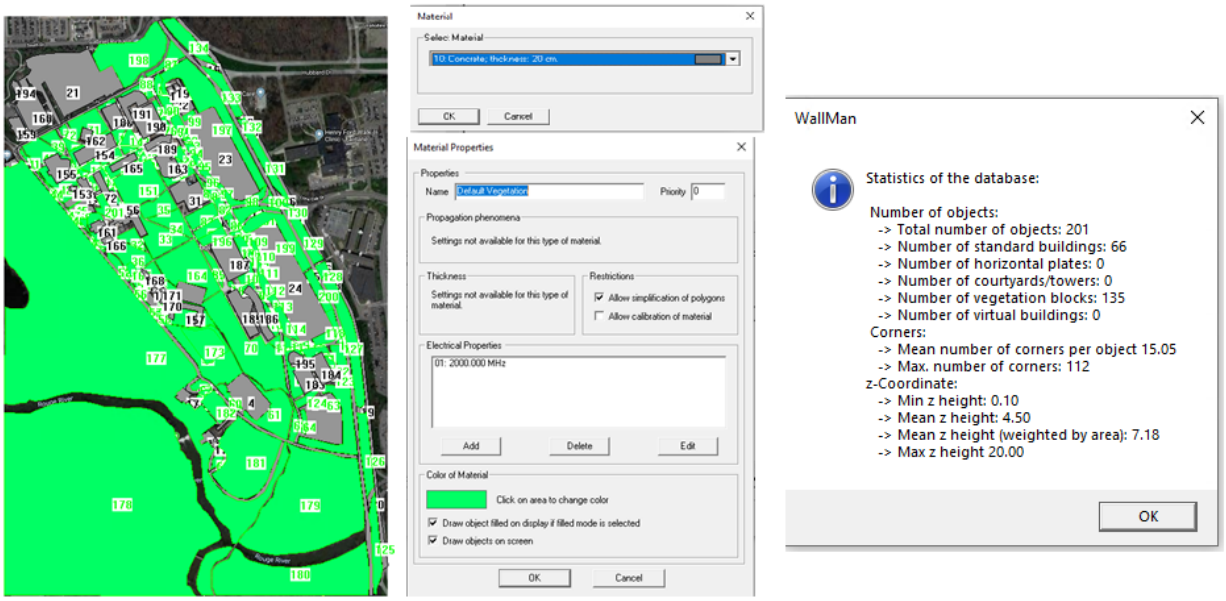


Figure 44 Altair-WallMan Simulation Setup

The simualtion is performed in the ProMan software with the results recoreded in ASCII format. The computational prediction results in the ASCII format are:

- Field Strenght
- Path Loss
- Line of Sight Analysis
- Delay Spread
- Minimum Path Delay
- Channel Impulse Response
- Propagation paths

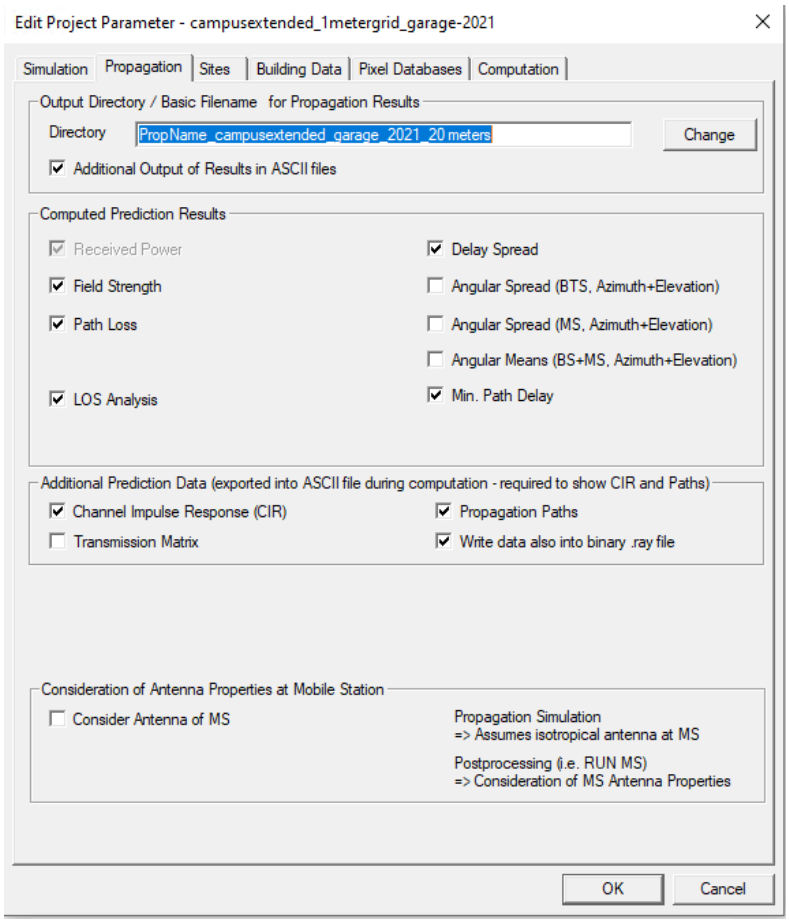
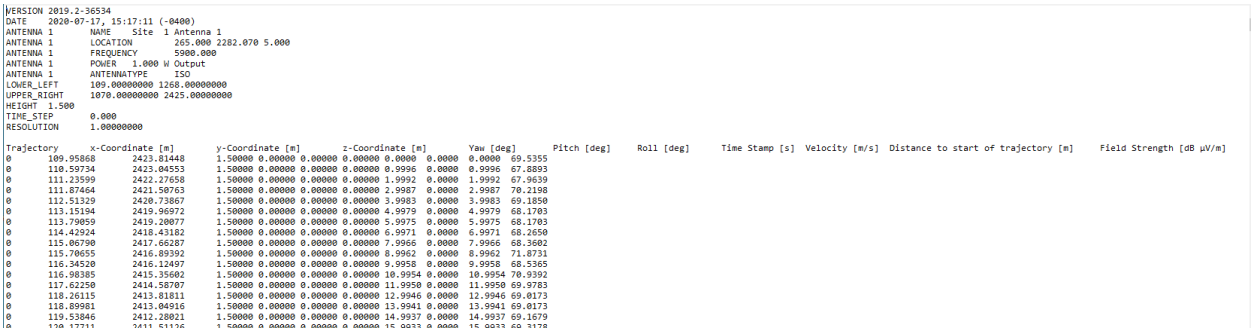


Figure 45 Project Parameter

A.2: Simulation Data Format

The results are stored in the ASCII format for better computation. The results contain the number of antennas used for the simulation along with their location, their frequency of operation, the transmit power and the receiver location with time stamp.



```
VERSION 2019.2-36534
DATE    2020-07-17, 15:17:11 (-0400)
ANTENNA 1       NAME    Site  1 Antenna 1
ANTENNA 1       LOCATION        265.000 2282.070 5.000
ANTENNA 1       FREQUENCY       5900.000
ANTENNA 1       POWER   1.000 W Output
ANTENNA 1       ANTENNATYPE     ISO
LOWER_LEFT      109.00000000 1268.00000000
UPPER_RIGHT     1070.00000000 2425.00000000
HEIGHT  1.500
TIME_STEP       0.000
RESOLUTION      1.00000000

Trajectory      x-Coordinate [m]        y-Coordinate [m]        z-Coordinate [m]        Yaw [deg]       Pitch [deg]     Roll [deg]      Time Stamp [s]  Velocity [m/s]  Distance to start of trajectory [m]     Field Strength [dB µV/m]
0       109.95868       2423.81448      1.50000 0.00000 0.00000 0.00000 0.0000  0.0000  0.0000  69.5355
0       110.59734       2423.04553      1.50000 0.00000 0.00000 0.00000 0.9996  0.0000  0.9996  67.8893
0       111.23599       2422.27658      1.50000 0.00000 0.00000 0.00000 1.9992  0.0000  1.9992  67.9639
0       111.87464       2421.50763      1.50000 0.00000 0.00000 0.00000 2.9987  0.0000  2.9987  70.2198
0       112.51329       2420.73867      1.50000 0.00000 0.00000 0.00000 3.9983  0.0000  3.9983  69.1850
0       113.15194       2419.96972      1.50000 0.00000 0.00000 0.00000 4.9979  0.0000  4.9979  68.1703
0       113.79059       2419.20077      1.50000 0.00000 0.00000 0.00000 5.9975  0.0000  5.9975  68.1703
0       114.42924       2418.43182      1.50000 0.00000 0.00000 0.00000 6.9971  0.0000  6.9971  68.2650
0       115.06790       2417.66287      1.50000 0.00000 0.00000 0.00000 7.9966  0.0000  7.9966  68.3602
0       115.70655       2416.89392      1.50000 0.00000 0.00000 0.00000 8.9962  0.0000  8.9962  71.8731
0       116.34520       2416.12497      1.50000 0.00000 0.00000 0.00000 9.9958  0.0000  9.9958  68.5365
0       116.98385       2415.35602      1.50000 0.00000 0.00000 0.00000 10.9954 0.0000  10.9954 70.9392
0       117.62250       2414.58707      1.50000 0.00000 0.00000 0.00000 11.9950 0.0000  11.9950 69.9783
0       118.26115       2413.81811      1.50000 0.00000 0.00000 0.00000 12.9946 0.0000  12.9946 69.0173
0       118.89981       2413.04916      1.50000 0.00000 0.00000 0.00000 13.9941 0.0000  13.9941 69.0173
0       119.53846       2412.28021      1.50000 0.00000 0.00000 0.00000 14.9937 0.0000  14.9937 69.1679
0       120.17711       2411.51126      1.50000 0.00000 0.00000 0.00000 15.9933 0.0000  15.9933 69.3178
```

Figure 46 Field Strength Data Format

All the data collected over simulation contains the following fields as shown in Table 19.

Table 19 Data field from Simulation

Field	Units
x-Coordinate	Meters
Y-Coordinate	Meters
z-Coordinate	Meters
Yaw	Degree
Pith	Degree
Roll	Degree
Time Stamp	Seconds
Velocity	Meter per second
Distance to start of trajectory	Meters
Field Strength	dB microvolt per Meter
Line of Sight	Status [1 or 0]
Path Loss	dB
Power	dBm

Appendix B: Drive Data

B.1: Real Data Format

The sample data collected at University of Michigan Dearborn campus is shown below in this Appendix. The data is collected for Beam 1, 2, 3 and 4 and the Omni directional antenna. Each Beam data has the time stamp in seconds, followed by index number and then followed by the signal strength in dBm.

Table 20 Beam 1 Sample Data Set

Time Stamp	Index	Signal Strength
2491.96	46132	-67
2492.71	46137	-62
2493.34	46142	-68
2494.15	46147	-67
2494.19	46152	-71
2494.22	46157	-68
2494.26	46162	-64
2494.49	46167	-68
2494.57	46172	-70
2495.02	46177	-61

Table 21 Beam 2 Sample Data Set

Time Stamp	Index	Signal Strength
2491.99	46133	-66
2492.84	46138	-66
2493.78	46143	-67
2494.16	46148	-70
2494.2	46153	-71
2494.23	46158	-71
2494.27	46163	-61
2494.5	46168	-64
2494.59	46173	-68
2495.42	46178	-62

Table 22 Beam 3 Sample Data Set

Time Stamp	Index	Signal Strength
2492.37	46134	-66
2492.86	46139	-65
2493.8	46144	-66
2494.17	46149	-70
2494.2	46154	-67
2494.24	46159	-70
2494.27	46164	-74
2494.52	46169	-67
2494.61	46174	-70
2495.44	46179	-61

Table 23 Beam 4 Sample Data Set

Time Stamp	Index	Signal Strength
2492.39	46135	-67
2492.88	46140	-72
2493.82	46145	-73
2494.18	46150	-70
2494.21	46155	-69
2494.25	46160	-64
2494.47	46165	-63
2494.53	46170	-70
2494.62	46175	-70
2495.46	46180	-62

Table 24 Omni Sample Dat Set

Time Stamp	Index	Signal Strength
2491.96	46132	-67
2492.71	46137	-62
2493.34	46142	-68
2494.15	46147	-67
2494.19	46152	-71
2494.22	46157	-68
2494.26	46162	-64
2494.49	46167	-68
2494.57	46172	-70
2495.02	46177	-61

References

[1] Q. Guo and B. Liu, "Simulation and Physical Measurement of Seamless Passenger Airbag Door Deployment," SAE, 2012.

[2] ETSI, "3GPP TS 36.213: Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures," 2017. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/136200_136299/136213/14.02.00_60/ts_136213v140200p.pdf. Accessed: Feb. 23. 2021.

[3] R. M.-M. a. J. Gozalvez, "Lte-v for sidelink 5g v2xvehicular communications: A new 5g technology for short-range vehicle-to-everything communications," IEEE Vehicular Technology Magazine, 2017.

[4] B. Toghi, M. Saifuddin, H. N. Mahjoub, M. O. Mughal, Y. P. Fallah, J. Rao and S. Das, "Multiple access in cellular v2x: Performance analysis in highly congested vehicular networks," in *IEEE Vehicular Networking Conference*, 2018.

[5] G. Tuzi, Z. Medenica and R. Miucic, "Using Convolutional Neural Networks for Distance Estimation between Dedicated Short-Range Communications Equipped Vehicles," in *IEEE 87th Vehicular Technology Conference (VTC Spring)*, Porto, 2018.

[6] SAE, "SAE J2735: Dedicated Short-Range Communications (DSRC) Message Set Dictionary," SAE, PA, 2016.

[7] J. G. De gooijer and R. J. Hyndman, "25 years of time series forecasting," *International Journal of Forecasting,* vol. 22, no. 3, pp. 443-473, 2006.

[8] G. Sismanoglu, M. Ali Onde, F. Kocer and O. K. Sahingoz, "Deep Learning Based Forecasting in Stock Market with Big Data Analytics," 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT), Istanbul, 2019.

[9] A. G. Salman, B. Kanigoro and Y. Heryadi, "Weather forecasting using deep learning techniques," in *International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2015.

[10] D. Xia, B. Wang, H. Li, Y. Li and Z. Zhang, "A distributed spatial–temporal weighted model on MapReduce for short-term traffic flow forecasting," in *Neurocomputing*, 2016.

[11] S. Liu, V. Elangovan and W. Xiang, "A Vehicular GPS Error Prediction Model Based on Data Smoothing Preprocessed LSTM," in *IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, 2019.

[12] A. Kulkarni, A. Seetharam, A. Ramesh and J. D. Herath, "DeepChannel: Wireless Channel Quality Prediction Using Deep Learning," *IEEE Transactions on Vehicular Technology,* vol. 69, pp. 443-456, 2020.

[13] X. Xu, R. Li, H. Rui, W. Lin, X. Liu and W. Cao, "Wireless Channel Scenario Recognition Based on Neural Networks," in *ITU Kaleidoscope: Connecting Physical and Virtual Worlds (ITU K)*, Geneva, 2021.

[14] Y. Zhou, Y. Wang, P. Liu, J. Yang, T. Ohtsuki and H. Sari, "Decentralized Learning-based Scenario Identification Method for Intelligent Vehicular Communications," in *IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, 2021.

[15] T. Chang, S. Jiang, Y. Sun, A. Jia and W. Wang, "Multi-bandwidth NLOS Identification Based on Deep Learning Method," in *15th European Conference on Antennas and Propagation (EuCAP)*, 2021.

[16] M. Yang, B. Ai, R. He, C. Shen, M. Wen, C. Huang, J. Li, Z. Ma, L. Chen, X. Li and X. Xhong, "Machine-Learning-Based Scenario Identification Using Channel Characteristics in Intelligent Vehicular Communications," *IEEE Transactions on Intelligent Transportation Systems,* vol. 22, no. 7, pp. 3961-3974, 2021.

[17] G. E. P. Box and D. A. Pierce, "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models," J. Am. Stat. Assoc., 1970.

[18] P.-F. Pai, K.-P. Lin, C.-S. Lin and P.-T. Chang, "Time series forecasting by a seasonal support vector regression model," *Expert Systems with Applications,* vol. 37, no. 6, pp. 4261-4265, 2010.

[19] B. Zhao, H. Lu, S. Chen, J. Liu and D. Wu, "Convolutional neural networks for time series classification," *Journal of Systems Engineering and Electronics,* vol. 28, no. 1, pp. 162-169, 2017.

[20] V. Elangovan, S. Liu and W. Xiang, "An Ensembled Approach to Time Series Prediction for Vehicle Communication," in *IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, 2021.

[21] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," Neurocomputing, 2003.

[22] J. S. Armstrong, K. C. Green and A. Graefe, "Golden rule of forecasting: Be conservative," *Journal of Business Research,* vol. 68, no. 8, pp. 1717-1731, 2015.

[23] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar and P.-A. Muller, "Deep learning for time series classification: a review," *Data mining and knowledge discovery,* vol. 33, no. 4, pp. 917-963, 2019.

[24] W. Harvey, M. Teng and F. Wood, "Near-optimal glimpse sequences for improved hard attention neural network training," arXiv preprint arXiv:1906.05462, 2019.

[25] V. O. L. Q. Sutskever I, "Sequence to sequence learning with neural networks," *Advances in neural information processing systems,* vol. 27, 2014.

[26] D. Bahdanau, K. Cho and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," arXiv:1409.0473.

[27] M.-T. Luong, H. Pham and C. D. Manning, "Effective Approaches to Attention-based Neural Machine Translation," arXiv:1508.04025.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017.

[29] S. Du, T. Li, Y. Yang and S.-J. Horng, "Multivariate time series forecasting via attention-based encoder–decoder framework," *Neurocomputing,* vol. 388, pp. 269-279, 2020.

[30] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proceedings of the 32nd International Conference on Machine Learning*.

[31] S. G. a. S. S. Shiv Shankar, "Surprisingly Easy Hard-Attention for Sequence to Sequence Learning," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, 2018.

[32] T. T. Z. G. L. J. J. S. W. a. C. Z. Shen, "Reinforced self-attention network: a hybrid of hard and soft attention for sequence modeling," arXiv preprint arXiv:1801.10296, 2018.

[33] G. F. Elsayed, S. Kornblith and Q. V. Le, "Saccader: Improving accuracy of hard attention models for vision," in *Advances in Neural Information Processing Systems*, 2018.

[34] A. Papadopoulos, P. Korus and N. Memon, "Hard-attention for scalable image classification," in *Advances in Neural Information Processing Systems*, 2021.

[35] I. A. S. M. P. A. F. a. A. I. Sorokin, "Deep attention recurrent Q-network," arXiv preprint arXiv:1512.01693, 2015.

[36] Y. Dang, Y. Kim, J. Chiu, D. Guo and A. M. Rush, "Latent alignment and variational attention," in *Advances in Neural Information Processing Systems*, 2018.

[37] M. Malinowski, C. Doersch, A. Santoro and P. Battaglia, "Learning visual question answering by bootstrapping hard attention," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[38] W. Harvey, M. Teng and F. Wood, "Near-optimal glimpse sequences for improved hard attention neural network training," arXiv preprint arXiv:1906.05462, 2019.

[39] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings and A. Mouzakitis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE Transactions on Intelligent Transportation Systems,* pp. 33-47, 2020.

[40] A. Luckow, M. Cook, N. Ashcraft, E. Weill, E. Djerekarov and B. Vorster, "Deep learning in the automotive industry: Applications and tools," in *IEEE International Conference on Big Data (Big Data)*, 2016.

[41] H. Ye, L. Liang, G. Ye Li, J. Kim, L. Lu and M. Wu, "Machine Learning for Vehicular Networks: Recent Advances and Application Examples," *IEEE Vehicular Technology Magazine,* vol. 13, no. 2, pp. 94-101, 2018.

[42] R. Sattiraju, A. Weinand and H. D. Schotten, "Channel Estimation in C-V2X using Deep Learning," in *IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 2019.

[43] G. Tuzi, Z. Medenica and R. Miucic, "Using Convolutional Neural Networks for Distance Estimation between Dedicated Short-Range Communications Equipped Vehicles," in *IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018.

[44] X. Chen, H. Zhang, F. Zhao, Y. Cai, H. Wang and Q. Ye, "Vehicle Trajectory Prediction Based on Intention-Aware Non-Autoregressive Transformer With Multi-Attention Learning for Internet of Vehicles," *IEEE Transactions on Instrumentation and Measurement,* vol. 71, pp. 1-12, 2022.

[45] W. Xiang, V. Elangovan and S. Lakshmanan, "A Real-Time Seq2Seq Beamforming Prediction Model for C-V2X Links," in *AI-enabled Technologies for Autonomous and Connected Vehicles*, Springer, 2023, p. 511.

[46] Altair Winprop, "User Guide," Altair, 2019.

[47] X. Wang, L. Gao, S. Mao and S. Pandey, "CSI-based fingerprinting for indoor localization:A deep learning approach," *IEEE Transactions on Vehicular Technology,* vol. 66, no. 1, pp. 763-776, 2017.

[48] L. T. Tan and R. Qingyang Hu, "Mobility-aware edge caching and computing in vehicle networks: A deep reinforcement learning," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 11, pp. 10190-10203, 2018.

[49] H. Huang, J. Yang, H. Huang, Y. Song and G. Gui, "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 9, pp. 8549-8560, 2018.

[50] G. Gui, H. Huang, Y. Song and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 9, pp. 8440-8450, 2018.

[51] Y. He, C. Liang, F. R. Yu, N. Zhao and H. Yin, "Optimization of cache-enabled opportunistic interference alignment wireless networks: A big data deep reinforcement learning approach," in *Proc. IEEE Int. Conf. Communication*, 2017.

[52] G. Tuzi, Z. Medenica and R. Miucic, "Using Convolutional Neural Networks for Distance Estimation between Dedicated Short-Range Communications Equipped Vehicles," in *IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018.

[53] Z. Sheng, A. Pressas, V. Ocheri, F. Ali, R. Rudd and M. Nekovee, "Intelligent 5G Vehicular Networks: An Integration of DSRC and mmWave Communications," in *International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, 2018.

[54] B. Kihei, J. A. Copeland and Y. Chang, "Automotive Doppler sensing: The Doppler profile with machine learning in vehicle-to-vehicle networks for road safety," in *IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Sapporo, 2017.

[55] R. Zhang, A. Ishikawa, W. Wang, B. Striner and O. K. Tonguz, "Using Reinforcement Learning With Partial Vehicle Detection for Intelligent Traffic Signal Control," *IEEE Transactions on Intelligent Transportation Systems.*

[56] V. Elangovan, W. Xiang and S. Liu, "A Real-Time C-V2X Beamforming Selector Based on Effective Sequence to Sequence Prediction Model using Transitional Matrix Hard Attention," *IEEE Access.*

[57] Q. Mao, F. Hu and Q. Hao, "Deep learning for intelligent wireless networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials,* vol. 20, no. 4, pp. 2595-2621, 2018.

[58] M. Chen, U. challita, W. Saad, C. Yin and M. Debbah, "Artificial Neural Networks-Based Machine Learning for Wireless Networks: A Tutorial," *IEEE Communications Surveys & Tutorials,* vol. 21, no. 4, pp. 3039-3071, 2019.

[59] H. Zhang, C. Huang, M. Gao, M. Yang and R. Chen, "A Time-Varying Clustering Algorithm for Channel Modeling of Vehicular MIMO Communications," in *XXXIIIrd General Assembly and Scientific Symposium of the International Union of Radio Science*, 2020.

[60] M. Yang, B. Ai, R. He, C. Huang, J. Li, Z. MA, L. Chen, X. Li and Z. Zhong, "Identification of Vehicle Obstruction Scenario Based on Machine Learning in Vehicle-to-vehicle Communications," in *IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020.

[61] C. Huang, A. F. Molisch, R. He, R. Wang, P. Tang, B. Ai and Z. Zhong, "Machine Learning-Enabled LOS/NLOS Identification for MIMO Systems in Dynamic Environments," *IEEE Transactions on Wireless Communications,* vol. 19, no. 6, pp. 3643-3657, 2020.

[62] Mathworks, "Vector Autoregressive Models," [Online]. Available: https://www.mathworks.com/help/econ/introduction-to-vector-autoregressive-var-models.html?s_tid=srchtitle . [Accessed 23. Feb. 2021].

[63] M. Roondiwala, H. Patel and S. Varma, "Perdicting Stock Prices Using LSTM," *International Journal of Science and Reserch,* pp. 1754-1756, 2017.

[64] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 28, no. 10, pp. 2222-2232, 2017.

[65] C. Huang, A. F. Molisch, R. Wang, P. Tang, R. He and Z. Zhong, "Angular Information-Based NLOS/LOS Identification for Vehicle to Vehicle MIMO System," in *IEEE International Conference on Communications Workshops (ICC Workshops)*, Shanghai, 2019.