IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 January 2023, accepted 28 January 2023, date of publication 1 February 2023, date of current version 6 February 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3241130*

RESEARCH ARTICLE

# A Real-Time C-V2X Beamforming Selector Based on Effective Sequence to Sequence Prediction Model Using Transitional Matrix Hard Attention

**VIVEKANANDH ELANGOVAN, WEIDONG XIANG, (Member, IEEE), AND SHENG LIU**
Department of Electrical and Computer Engineering, University of Michigan–Dearborn, Dearborn, MI 48187, USA
Corresponding author: Vivekanandh Elangovan (velango@umich.edu)

**ABSTRACT** For C-V2X systems, the selection of the best beam in a real-time mode becomes an increasingly critical and yet open topic. Most of the existing approaches adopt either conventional ARIMA or ANN. Recently, there has been research on adopting sequence-to-sequence (Seq2Seq) predictors with attentions to extract time series features and emphasis on critical information to achieve data prediction. In this paper, a Seq2Seq predictor integrating with a Transitional Matrix based Hard attention is presented and validated through an artificial test dataset with predefined transitional states. At first, the transition probability matrix is generated from previous time series data and fed into the ''hard'' attention module of Seq2Seq predictor to determine the weights during the training phase. Secondly, the presented Seq2Seq predictor was implemented and adopted to predict the best beams of a C-V2X beamforming selector built up by the authors. Experiments were conducted and captured data were used to validate the performance of the predictor. When compared with baseline models, the presented predictor can achieve an enhanced prediction accuracy in a gain of 10-12%.

**INDEX TERMS** Autoregressive integrated moving average (ARIMA), beamforming, cellular vehicle to everything (C-V2X), encoder, decoder, encoder decoder with attention, deep learning (DL), long short-term memory (LSTM), machine learning (ML), neural machine translation (NMT), hard attention, sequence to sequence, soft attention, time series prediction, transition matrix, wireless network.

## I. INTRODUCTION

Time series prediction using Neural network has been long studied in many fields [1], such as stock forecasting [2], Weather forecasting [3], traffic flow forecasting [4], Global positioning prediction [5], Wireless Channel prediction [6] and most of the time series prediction follows the existing Neural network methodology such as ARIMA [7], SVR [8], traditional ANN [9] and hybrid neural networks [10], [11]. Traditionally statistical methods such as ARIMA, exponential smoothing was often used for time series forecasting. Armstrong et al. [23] 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 [24]. Wang et al. [36] 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 [12], Encoder Decoder with Attention [13], [14] and Transformation [15]. These methods have been used by various researcher for language translation and these methods has been researched for time series prediction [16]. In the Encoder Decoder with Attention, the encoder and decoder are designed using various Neural Network such as RNN, LSTM, GRU. The Attention is the key mechanism which provides improvement from Encoder Decoder

The associate editor coordinating the review of this manuscript and approving it for publication was Sangsoon Lim.

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. [28] proposed hard attention where it attends to exactly one input state for an output, [29], [30] shows a sequence-to-sequence prediction with the hybrid of hard and soft attention. Reference [31] provides a modified hard attention called Saccader for vision by requiring only class labels for initial attention, whereas [32] provides a multi-scaled hard-attention architecture for image classification. Reference [33] presents the ''soft'' and ''hard'' attention on Q learning which is based on feature extracted by CNN at different image regions, [34] 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, [35] shows the use of hard attention by exploring various image attention mechanism to locate regions that are relevant to the question, [36] 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.

In a time series prediction such as Beamforming selection in a C-V2X system, where the beam must be chosen based on previous input, the above-mentioned methods does not provide any attention to the transitional values. In response to that, the contribution of this paper is as follows:

- We showed a ''hard'' but deterministic attention mechanism trainable by pre-determined transitional states
- We show how we can gain insight in ''which'' attention is focused on
- Finally, we demonstrate the effectiveness of our model by testing on actual measured data in the field, and the experiment results showed that our model has the best prediction performance compared to the baseline methods.

Wireless channel status and characteristics demonstrate similarity once the environment is in the same category such as urban, rural, residential, and hilly areas. The main feature of each category channels can be real-time learned and modelled by a transitional matrix and used as a foreknowledge to neural network.

The rest of the paper is arranged as follows: Section II gives the background about problem statement and the motivation, Section III expounds on the experimental measurement data, Section IV analyzes the overall architecture of the encoder decoder with attention based on transition states framework and describes the relevant theory and process details. Section V is the theoretical analysis and Section VI is the experimental analysis content and concluding the paper on Section VII.

## II. 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 [17], language recognition [18] and many more in automated driving. Cellular Vehicle to Everything (C-V2X) has been emerging technology within Automotive world which encompasses Vehicle to Vehicle (V2V) connectivity, Vehicle to Infrastructure (V2I), Vehicle to Pedestrians (V2P) and Vehicle to Network (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) [25] for the V2V communication by then V2V communication is like to be based on the DSRC defined in SAE J2735 [26]. 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 [27]. 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 [19], most of them focused on channel estimation [20], distance estimation [21], Vehicle trajectory [22] but very minimal in beam prediction [39] and only using traditional methods and nothing on beam prediction using deep neural network. Our research is focused on real-time beam prediction model.

## III. SYSTEM MODEL

Beamforming antenna arrays have attracted increasingly attention recently and well found their applications ranging from Wi-Fi, 5G and Internet of things (IoT). In this project, a 4-element uniform linear array (ULA) receiver antenna array built upon a $4 \times 4$ 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 1 is the system design of the 4 × 4 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 4×4 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 $\phi = \text{kdcos}\ \theta$, where wave number (k) and Array Factor (AF) is given by (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

$\alpha_n$ is additional phase shift

For a broad side antenna array, the AF can be further written as (3),

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

Finally, the beamforming radiation pattern is given by (4),

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

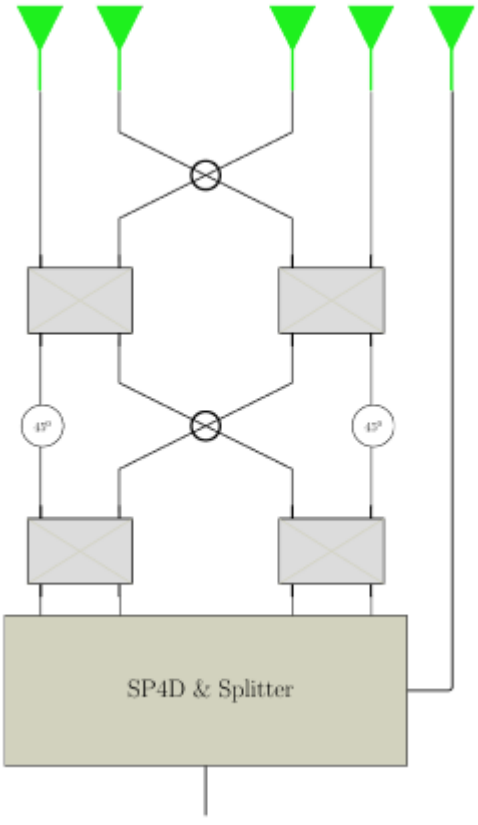


**FIGURE 1.** System design of 4 × 4 beamforming for C-V2X.

The radiation pattern of the ULA is shown in Figure 2 where the radiation pattern for all the 4 ports is shown.

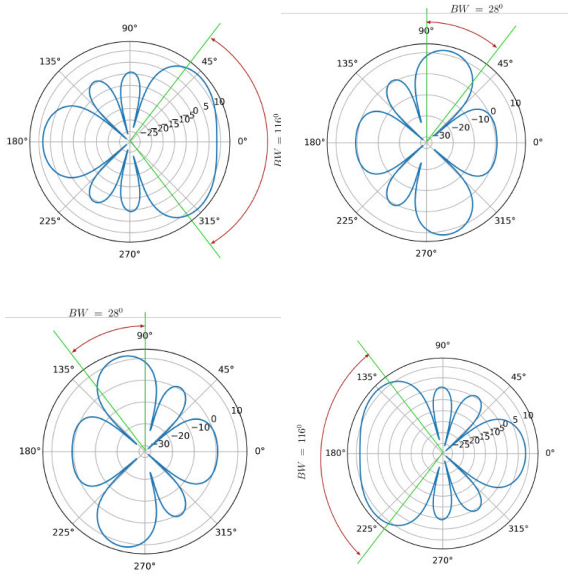


**FIGURE 2.** Radiation pattern of 4 antennas.

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 as shown in Figure 3. 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 below.

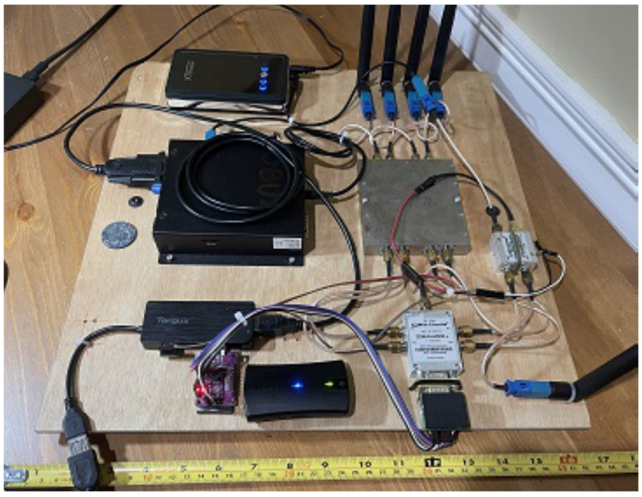

**FIGURE 3.** Receiver antenna unit.

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 4.

**FIGURE 4.** Transmitter antenna unit.

The test is performed at the university campus shown in Figure 5, 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 5.** Google maps of campus with the location of transmitter.

## IV. FRAMEWORK MODEL

In this section, we discuss the implementation details of machine learning and the training methodology. We split the dataset into three sets such as:

- Training sets: 80% of data set
- Prediction sets: 20% 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 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.

Before diving into the details of our framework model, we first brief the limitation of traditional standard beam selection technique.

### A. 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 4$\lambda$ where $\lambda$ is the wavelength of 5.9GHz (Change of beam interval is every 100 msec which translates to the length of 4$\lambda$). Considering a vehicle speed of 60 mile/hour, the distance moved in every 4$\lambda$ 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.

### B. 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

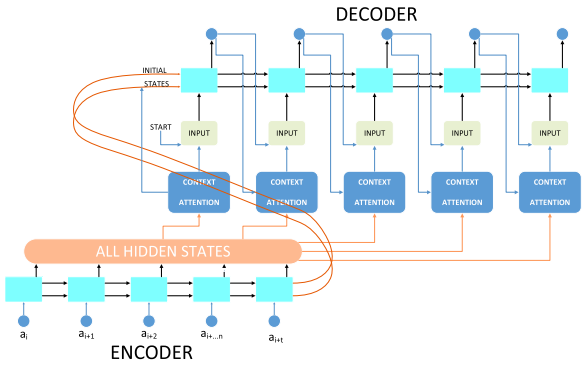


**FIGURE 6.** Model of encoder decoder with attention.

a context vector that is filtered specifically for each output time step.

With the introduction of Attention as shown in Figure 6, 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 (6) and Luong (5).

$$score\left(h_t, \bar{h}_s\right) = h_t W \bar{h}_s \quad \left[Luong's\ multiplicate\ style\right] \tag{5}$$

$$score\left(h_t, \bar{h}_s\right) = \vartheta_a^T \tanh\left(W_1 h_t + W_2 \bar{h}_s\right) \quad \left[Bahdanau's\ additive\ style\right] \tag{6}$$

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 (7).

$$c_t = \sum_s \alpha_{ts} h_t \tag{7}$$

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.

### C. 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 1. This

**TABLE 1.** State transition matrix.

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

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.

When the score (8) is calculated, the weights are determined based on the transition matrix TM.

$$score\left(h_t, \bar{h}_s\right) = h_t W \bar{h}_s \tag{8}$$

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 $(a_i, a_{i+1}, a_{i+2}\ldots a_{i+t}) \mid (b_{t-1})$ i.e., $P(a_i, a_{i+1}, a_{i+2}\ldots a_{i+t} \mid b_{t-1})$

$$W = [P(a_i \mid b_{t-1}), P(a_{i+1} \mid b_{t-1}), P(a_{i+2} \mid b_{t-1}) \ldots \ldots, P(a_{i+t} \mid b_{t-1})] \tag{9}$$

The weight matrix is determined based on (9). 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 7 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 ($a_i \mid a_{i+3}$) = P4, P ($a_{i+2} \mid a_{i+3}$) = P12, P ($a_{i+1} \mid a_{i+3}$) = P8 and P ($a_i \mid a_{i+3}$) = P4 i.e., it would be P4 P12 P8 P4.

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

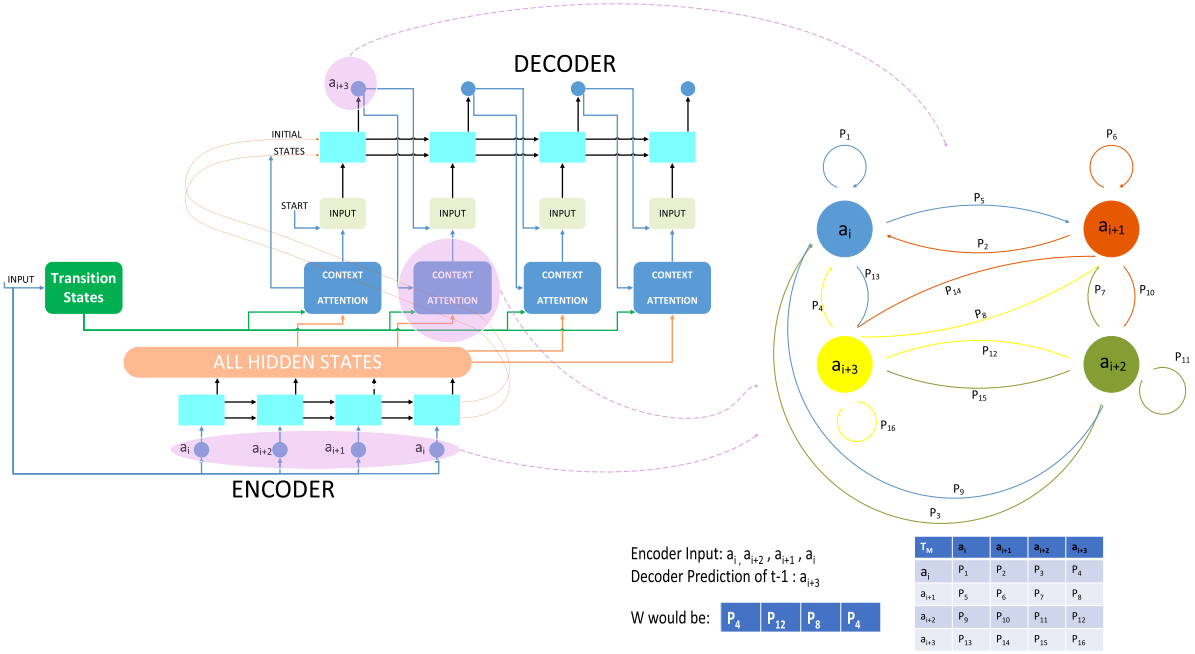


**FIGURE 7.** Example of encoder decoder – attention with transition matrix.

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.

### D. 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 (7) is calculated based on the score given by (8) using the probability distribution as shown in (10).

$$\alpha_{ts} = \frac{\exp(score\left(h_t, \bar{h}_s\right))}{\sum_{s'=1}^{S} \exp(score\left(h_t, \bar{h}_{s'}\right))} \tag{10}$$

In image classification and Natural Language Prediction, the weights in (8) 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 are formulated to provide the statistical information over long term data for the score and thereafter context vector calculation.

## V. 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 2.** 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 2 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 8 shows an example for the difference between guided and unguided methodology based on the Table 2 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.

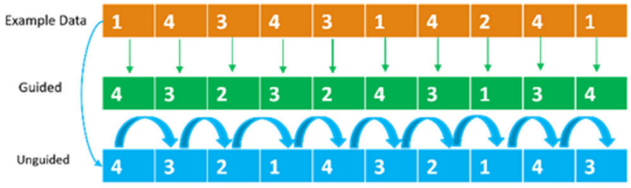


**FIGURE 8.** Guided vs unguided maximum likelihood estimation.

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 (MSE) (11), Mean Absolute Error (MAE) (10) and Mean Absolute Percentage Error (MAPE) (13) 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.

$$MAE_j = \frac{\sum_{i=1}^{n} (\hat{x}_{ij} - x_{ij})}{n} \tag{11}$$

$$MSE_j = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{x}_i)^2 \tag{12}$$

$$MAPE_j = \frac{\sum_{i=1}^{n} \left| \frac{(x_{ij} - \hat{x}_{ij})}{x_{ij}} \right|}{n} \tag{13}$$

**TABLE 3.** 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 |

From Table 3, 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 9, 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.

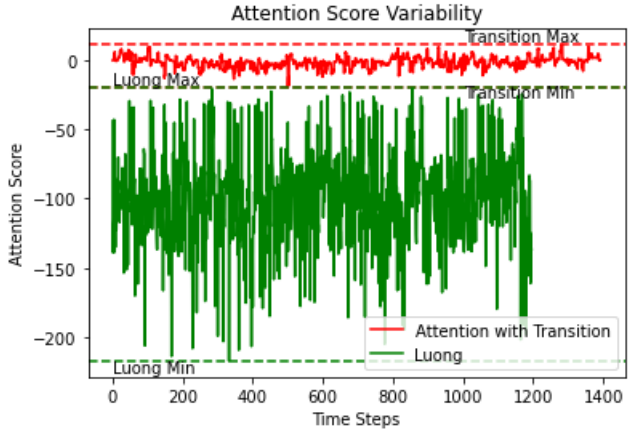


**FIGURE 9.** The attenuation score variability between Luong and attention with transition method.

The analysis is also performed using the actual measured data as explained in Section III.

## VI. EXPERIMENTAL ANALYSIS

### A. QUALITATIVE ANALYSIS

The experiment is performed over the collected data sample as described in Section III. 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 10 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.

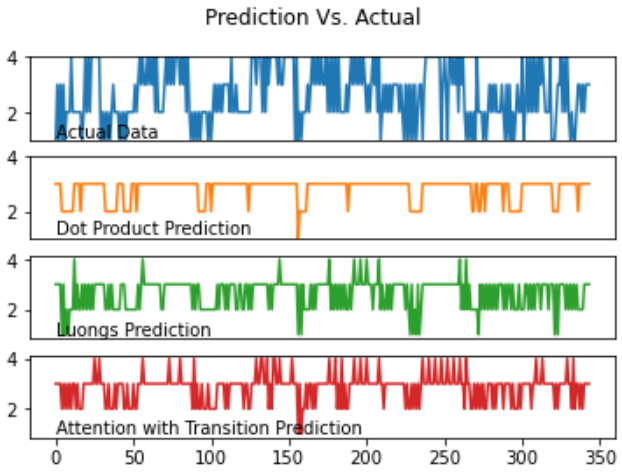


**FIGURE 10.** The comparison of the prediction results from the dot product, Luong's methods, attention with transition method with actual data.

The loss curve shown in Figure 11 indicates that the training is better and attains better stabilization quicker using our proposed model.

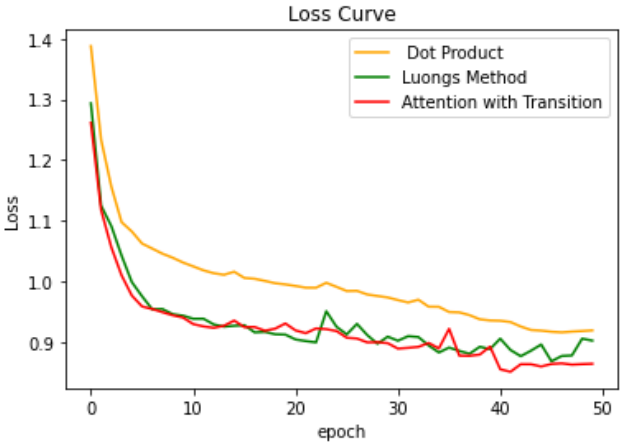


**FIGURE 11.** The comparison of the loss curves of the dot product, Luong's method, and attention with transition method.

The Attention vector is the score of the corresponding value within the source sequence which tell the decoder

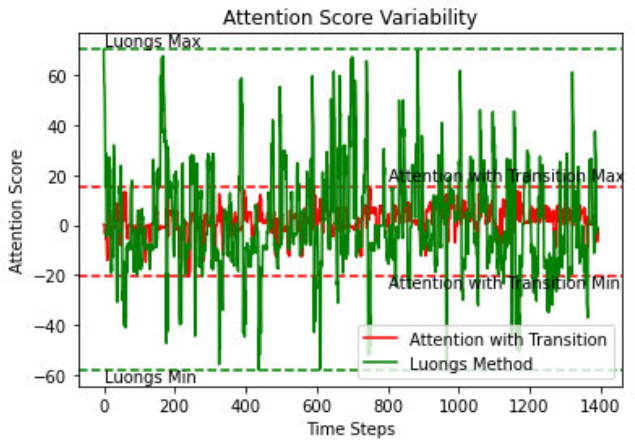


**FIGURE 12.** The attenuation score variability between the Luong's and attention with transition method.

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 12. 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.

The accuracy plots show in Figure 13 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.

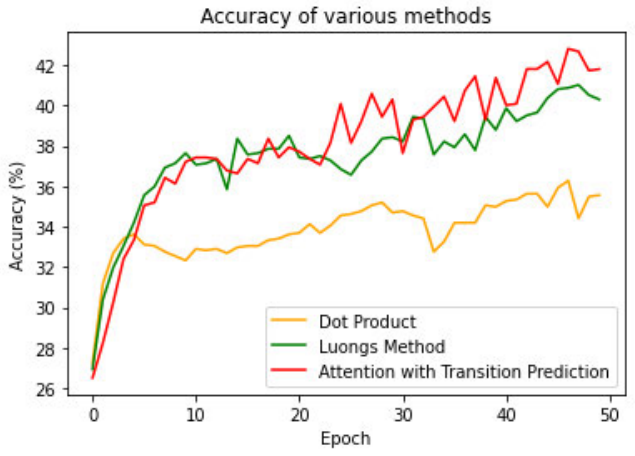


**FIGURE 13.** The comparison of the accuracy curves from the dot product, Luong's method, and attention with transition method.

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

**TABLE 4.** 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 4, 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 14, 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 5 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.

**TABLE 5.** Random selection 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 |

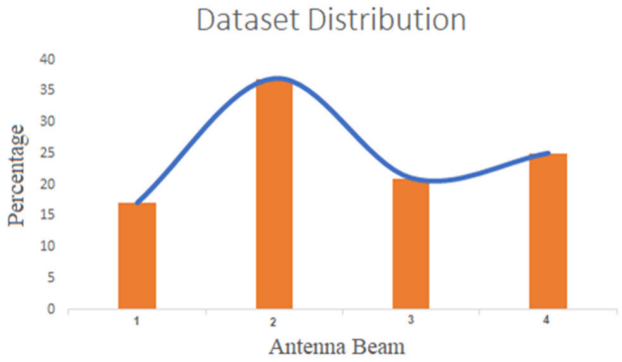


**FIGURE 14.** Data distribution percentage.

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.

### B. QUANTITATIVE ANALYSIS

To validate the model across various dataset, we also collected data from different drive zones around the campus as shown in Figure 15 and shown the analysis of the various dataset across the different encoder decoder models. Table 6 shows the performance comparison of various zones. The analysis indicates that Attention with Transition (Our proposed) model performed better than the traditional Encoder decoder model.

**FIGURE 15.** Various drive path (Clockwise) Zone 1, 2 and 3.

It shall be noted from Table 7 the performance improvement from the Dot product Vs. Attention with Transition and Luong's Method Vs. Attention with Transition. The

**TABLE 6.** 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 |

performance improvement is calculated from the accuracy percentage as explained in (14). The variance in the improvement as seen is dependent on the dataset. Based on our dataset the variance is between 10 to 12% improvement.

$$\textit{Performance Improvement} = \frac{\textit{Accuracy of Attention with Transition}}{\textit{Accurancy of Dot productorLuong Method}} \quad (14)$$

**TABLE 7.** Performance improvement comparison.

| 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

**TABLE 8. Performance comparison of 3rd party dataset.**

| Method | Accuracy Percentage (%) |
|---|---|
| Dot Product | 72.06 |
| Luong's Method | 99.28 |
| Attention with Transition | 99.53 |

calculated as shown in (15).

$$H(X) = -\sum_{i=1}^{n} P(x_i) log_b P(x_i) \tag{15}$$

The entropy is calculated for the theoretical data and for all the three measured zones and their corresponding accuracy is plotted in Figure 16. It shall be noted that as the Entropy increases the accuracy of prediction decreases which correlates to the Shannon theory.

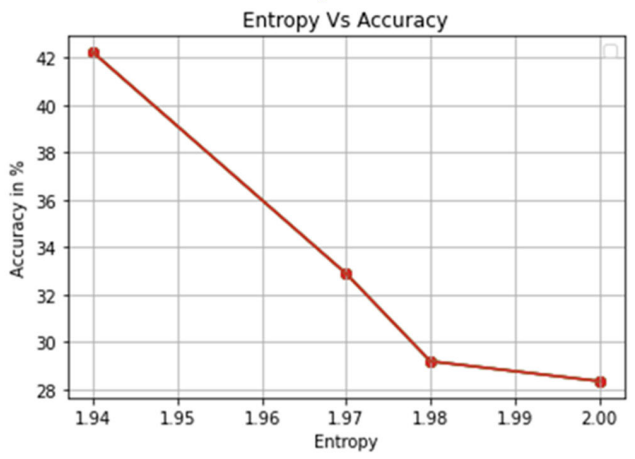


**FIGURE 16. Entropy vs accuracy.**

### C. REPOSITORY ANALYSIS

The dataset used to validate the model is the measured dataset around the campus. To validate the model on a 3rd party dataset, we use the Occupant Detection Data Set [37] from the UCI Machine Learning Repository database. The dataset contains the occupied status in a room i.e., if the room is occupied which is recorded as 1 or not which is recorded as 0. We trained the dataset using Dot product, Luong's method, and the Attention with Transition method to see the prediction accuracy. From the prediction results as shown in Table 8, we noticed that our method prediction result has slight improvement compared to the other method. The prediction dataset doesn't have much variability i.e., its either 0 or 1 and most of the values are in 0 and hence the improvement in accuracy is small considered to other prediction method as shown in Figure 17.

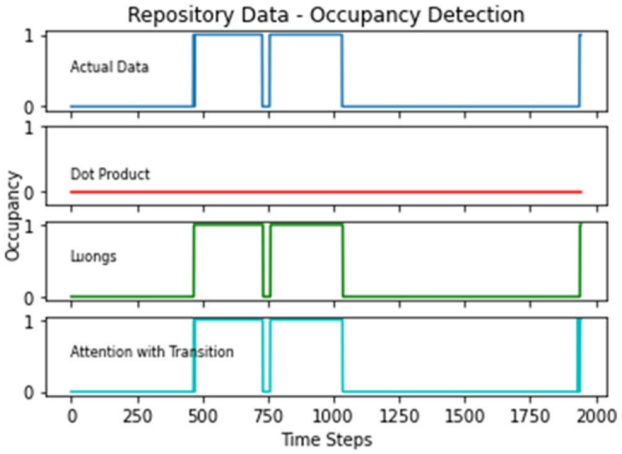


**FIGURE 17. Repository data – occupancy detection prediction comparison between Dot Product, Luong's method, and attention with transition method with the actual data.**

## VII. CONCLUSION

In this paper, 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. The future work would focus on multi-variate attribute to improve the accuracy of the prediction system.

## REFERENCES

[1] J. G. D. Gooijer and R. J. Hyndman, "25 years of time series forecasting," *Int. J. Forecasting*, vol. 22, no. 3, pp. 443–473, 2006.
[2] G. Sismanoglu, M. A. Onde, F. Kocer, and O. K. Sahingoz, "Deep learning based forecasting in stock market with big data analytics," in *Proc. Sci. Meeting Elect.-Electron. Biomed. Eng. Comput. Sci. (EBBT)*, Apr. 2019, pp. 1–4, doi: 10.1109/EBBT.2019.8741818.
[3] A. G. Salman, B. Kanigoro, and Y. Heryadi, "Weather forecasting using deep learning techniques," in *Proc. Int. Conf. Adv. Comput. Sci. Inf. Syst. (ICACSIS)*, Oct. 2015, pp. 281–285, doi: 10.1109/ICACSIS.2015.7415154.
[4] 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," *Neurocomputing*, vol. 179, pp. 246–263, Feb. 2016.
[5] S. Liu, V. Elangovan, and W. Xiang, "A vehicular GPS error prediction model based on data smoothing preprocessed LSTM," in *Proc. IEEE 90th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2019, pp. 1–5, doi: 10.1109/VTCFall.2019.8891454.
[6] A. Kulkarni, A. Seetharam, A. Ramesh, and J. D. Herath, "DeepChannel: Wireless channel quality prediction using deep learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 443–456, Jan. 2020, doi: 10.1109/TVT.2019.2949954.
[7] G. E. P. Box and D. A. Pierce, "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models," *J. Amer. Stat. Assoc.*, vol. 65, no. 332, pp. 1509–1526, Dec. 1970.
[8] P.-F. Pai, K.-P. Lin, C.-S. Lin, and P.-T. Chang, "Time series forecasting by a seasonal support vector regression model," *Expert Syst. Appl.*, vol. 37, no. 6, pp. 4261–4265, Jun. 2010.

[9] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu, "Convolutional neural networks for time series classification," *J. Syst. Eng. Electron.*, vol. 28, no. 1, pp. 162–169, Feb. 2017, doi: 10.21629/JSEE.2017.01.18.

[10] V. Elangovan, S. Liu, and W. Xiang, "An ensembled approach to time series prediction for vehicle communication," in *Proc. IEEE 94th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2021, pp. 1–6, doi: 10.1109/VTC2021-Fall52928.2021.9625341.

[11] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, Jan. 2003.

[12] I. Sutskever, O. Vinyals, and L. Qv, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 3104–3112.

[13] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.

[14] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," 2015, *arXiv:1508.04025*.

[15] A. Vaswani, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 6000–6010.

[16] 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, May 2020, doi: 10.1016/j.neucom.2019.12.118.

[17] 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 Trans. Intell. Transp. Syst.*, vol. 23, no. 1, pp. 33–47, Jan. 2022.

[18] A. Luckow, M. Cook, N. Ashcraft, E. Weill, E. Djerekarov, and B. Vorster, "Deep learning in the automotive industry: Applications and tools," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2016, pp. 3759–3768.

[19] H. Ye, L. Liang, G. Y. Li, J. Kim, L. Lu, and M. Wu, "Machine learning for vehicular networks: Recent advances and application examples," *IEEE Veh. Technol. Mag.*, vol. 13, no. 2, pp. 94–101, Jun. 2018, doi: 10.1109/MVT.2018.2811185.

[20] R. Sattiraju, A. Weinand, and H. D. Schotten, "Channel estimation in C-V2X using deep learning," in *Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS)*, Dec. 2019, pp. 1–5, doi: 10.1109/ANTS47819.2019.9117972.

[21] G. Tuzi, Z. Medenica, and R. Miucic, "Using convolutional neural networks for distance estimation between dedicated short-range communications equipped vehicles," in *Proc. IEEE 87th Veh. Technol. Conf. (VTC Spring)*, Jun. 2018, pp. 1–6, doi: 10.1109/VTCSpring.2018.8417732.

[22] 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 Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022, doi: 10.1109/TIM.2022.3192056.

[23] J. S. Armstrong, K. C. Green, and A. Graefe, "Golden rule of forecasting: Be conservative," *J. Bus. Res.*, vol. 68, no. 8, pp. 1717–1731, Aug. 2015.

[24] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Müller, "Deep learning for time series classification: A review," *Data Mining Knowl. Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[25] *Notice of Proposed Rulemaking (NPRM): Federal Motor Vehicle Safety Standards; V2V Communications*, National Highway Traffic Safety Administration (NHTSA), Department of Transportation (DOT), NHTSA, Washington, DC, USA, 2017.

[26] *Dedicated Short Range Communications (DSRC) Message Set Dictionary*, document SAE, SAE J2735 PA SAE, Mar. 2016.

[27] *On-Board System Requirements for V2V Safety Communications*, document SAE, SAE J2945/1, PA SEA, Apr. 2020.

[28] 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 *Proc. 32nd Int. Conf. Mach. Learn.*, vol. 37, pp. 2048–2057.

[29] S. Shankar, S. Garg, and S. Sarawagi, "Surprisingly easy hard-attention for sequence to sequence learning," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 640–645.

[30] T. Shen, T. Zhou, G. Long, J. Jiang, S. Wang, and C. Zhang, "Reinforced self-attention network: A hybrid of hard and soft attention for sequence modeling," 2018, *arXiv:1801.10296*.

[31] G. Elsayed, S. Kornblith, and Q. V. Le, "Saccader: Improving accuracy of hard attention models for vision," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 702–714.

[32] A. Papadopoulos, P. Korus, and N. Memon, "Hard-attention for scalable image classification," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 14694–14707.

[33] I. Sorokin, A. Seleznev, M. Pavlov, A. Fedorov, and A. Ignateva, "Deep attention recurrent Q-network," 2015, *arXiv:1512.01693*.

[34] Y. Deng, Y. Kim, J. Chiu, D. Guo, and A. Rush, "Latent alignment and variational attention," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 9712–9724.

[35] M. Malinowski, C. Doersch, A. Santoro, and P. Battaglia, "Learning visual question answering by bootstrapping hard attention," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 3–20.

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

[37] P. Wang, H. Wang, H. Zhang, F. Lu, and S. Wu, "A hybrid Markov and LSTM model for indoor location prediction," *IEEE Access*, vol. 7, pp. 185928–185940, 2019, doi: 10.1109/ACCESS.2019.2961559.

[38] M. Luis Candanedo and V. Feldheim, "Accurate occupancy detection of an office room from light, temperature, humidity and $CO_2$ measurements using statistical learning models," *Energy Buildings*, vol. 112, 15, Jan. 2016, pp. 28–39.

[39] 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* (Lecture Notes in Intelligent Transportation and Infrastructure), Y. L. Murphey, I. Kolmanovsky, and P. Watta, Eds. Cham, Switzerland: Springer, 2023, doi: 10.1007/978-3-031-06780-8_18.


**VIVEKANANDH ELANGOVAN** received the M.S. degree in electrical engineering from the Rochester Institute of Technology, Rochester, NY, USA. He is currently pursuing the degree with the Department of Electrical Engineering, University of Michigan–Dearborn, Dearborn, MI, USA. He is also working as a Research Engineer with Ford Motor Company. His research interests include wireless systems, computer networks, and application of machine learning to wireless communications.

**WEIDONG XIANG** (Member, IEEE) received the M.S. and Ph.D. degrees from Tsinghua University, Beijing, China, in 1996 and 1999, respectively. From 1999 to 2004, he worked as a Postdoctoral Fellow/Research Scientist at the Software Radio Laboratory (SRL), Georgia Institute of Technology, Atlanta, GA, USA. In 2004, he joined the Department of Electrical and Computer Engineering, University of Michigan–Dearborn (UMD), Dearborn, MI, USA, where he is currently a Professor. He established and led the Center for the Vehicular Communications and Network Laboratory, UMD, focusing on dedicated short range communications (DSRC), machine type communications (MTC), and LTE for high mobility applications and UWB positioning. His current research is widely supported by NSF, DoE, GM, Ford, LGE, CISCO Research, and other companies. He has published more than 100 technical papers in relevant international journals and conferences. His research interests include vehicular communications and networks, 5G, autonomous driving, the Internet of Things, and wireless control systems. He served as an Associate Editor/Editor for *IEEE Communications Magazine*, SAE Journals, and others.

**SHENG LIU** received the M.S. degree in electrical engineering from the University of Michigan–Dearborn, Dearborn, MI, USA, where he is currently pursuing the Ph.D. degree with the Department of Electrical, Electronics, and Computer Engineering. His research interests include wireless systems, vehicular communications, and machine learning applications applied on wireless communications.