VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JNANASANGAMA" BELAGAVI - 590 018

KARNATAKA

REPORT OF INTERNSHIP/PROFESSIONAL PRACTICE

Carried out in

EUNOIA LABS, Bangalore

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE & ENGINEERING

Submitted by:

MOHAMMED AFREED ULLA
[1CG20CS055]

INTERNAL GUIDE

Mrs. Rashmi C R M.Tech
Assistant Professor,
Dept. of CSE,
C.I.T, Gubbi, Tumkur.

EXTERNAL GUIDE

Mrs. Roopa K S
H R Manager
Eunoia Labs
Bangalore

HOD
Dr. Shantala C P Ph.D
Professor & Head,
Dept. of CSE
CIT, Gubbi

Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)
(NAAC Accredited & ISO 9001:2015 Certified Institution)
NH 206 (B.H. Road), Gubbi, Tumkur – 572216. Karnataka

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2023-2024

Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)
(NAAC Accredited & ISO 9001:2015 Certified Institution)
NH 206 (B.H. Road), Gubbi, Tumkur – 572216. Karnataka.

Partnering in Academic Excellence

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2023-2024

CERTIFICATE

This is to certify that the internship entitled **"TRAFFIC PREDICTION USING DEEP LEARNING"** has been carried out by **MOHAMMED AFREED ULLA - [1CG20CS055]** bonafide student of **CHANNABASAVESHWARA INSTITUTE OF TECHNOLOGY, GUBBI, TUMKUR**, in partial fulfillment of the requirement for the award of the degree **Bachelor of Engineering** in **COMPUTER SCIENCE & ENGINEERING** from the **Visvesvaraya Technological University, Belagavi** during the year **2023-2024.** It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report. The Internship report has been approved as it satisfies the academic requirements in respect of Internship/Professional practice prescribed for the said degree.

Signature of Guide	Signature of Internship Coordinator
Mrs. Rashmi C R **M.Tech** Assistant Professor, Dept., of CSE C.I.T, Gubbi.	**Mrs. Rashmi C R** **M.Tech.,** Assistant Professor, Dept., of CSE C.I.T, Gubbi.
Signature of HOD	Signature of Principal
Dr. Shantala C P **Ph.D** Professor & Head, Dept., of CSE C.I.T, Gubbi.	**Dr. SURESH D S** **Ph.D** Director & Principal C.I.T, Gubbi.

External Viva

Examiners Name	Signature with Date
1. ______________________________	______________________
2. ______________________________	______________________

Channabasaveshwara Institute of Technology
(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)
(NAAC Accredited & ISO 9001:2015 Certified Institution)
NH 206 (B.H. Road), Gubbi, Tumkur – 572216. Karnataka.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2023-2024

UNDERTAKING

I, **MOHAMMED AFREED ULLA** bearing **1CG20CS055**, student of **VIII Semester B.E.** in **COMPUTER SCIENCE & ENGINEERING, C.I.T, GUBBI, TUMKUR** hereby declare that the Internship carried out in **Eunoia Labs, Bangalore** and submitted in partial fulfillment of the requirements for the award of the degree **Bachelor of Engineering** in **COMPUTER SCIENCE & ENGINEERING** of the **Visvesvaraya Technological University, Belagavi** during the academic year 2023-2024.

Place: GUBBI **MOHAMMED AFREED**
ULLA
Date: **1CG20CS055**

Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)

(NAAC Accredited & ISO 9001:2015 Certified Institution)

NH 206 (B.H. Road), Gubbi, Tumkur – 572216. Karnataka.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2023-24

BONAFIDE CERTIFICATE

This is to certify that the Internship carried out in **Eunoia Labs, Bangalore** is a bonafide work of **MOHAMMED AFREED ULLA – [1CG20CS055],** student of **VIII** semester **B.E.- COMPUTER SCIENCE & ENGINEERING** from **Channabasaveshwara Institute of Technology, Gubbi, Tumkur,** in partial fulfillment of the requirements for the award of degree **B.E.,** in **COMPUTER SCIENCE & ENGINEERING** of **Visvesvaraya Technological University**, **Belgaum** during the academic year 2023-2024. It is certified that the Internship work carried out was under my supervision and guidance.

Guide

Mrs. Rashmi C R **M.Tech.,**
Assistant. Professor
Dept., of CSE
C.I.T, Gubbi.

ACKNOWLEDGEMENT

Several special people have contributed significantly to this effort. First of all, I am grateful to my institution, **Channabasaveshwara Institute of Technology, Gubbi**, which provides me an opportunity in fulfilling my most cherished desire of reaching my goal.

I, acknowledge and express my sincere thanks to our beloved Director & Principal, **Dr. Suresh D S**, for his many valuable suggestion and continued encouragement by supporting me in mt academic endeavors.

I, express my sincere gratitude to **Dr. Shantala C P**, Professor and Head, Department of CSE, for providing her constructive criticisms and suggestions.

I, extend my gratitude to my Internship guide **Mrs. Rashmi C R, Assistant Professor,** Department of CSE, for his/her guidance, support and suggestions throughout the period of this Internship.

I express my deep sense of gratitude to **Eunoia Labs, Bangalore** for giving such an opportunity to carry out the internship in their esteemed industry/organization.

I sincerely thank **Mrs. Roopa K S, H R Manager, Eunoia Labs, Bangalore** for exemplary guidance and supervision.

Finally, I would like to thank all the individuals who supported me directly and indirectly for the successful completion of this internship work.

MOHAMMED AFREED ULLA [1CG20CS055]

ABSTRACT

The project deals with traffic prediction using deep learning techniques that can be used to predict the traffic at particular junction. The pervious years traffic dataset have been used which contains the traffic data of four different junctions which ultimately provides the accuracy and mean square error. This prediction will be helpful for the people who are in need to check the immediate traffic state. The traffic data is predicated on a basis of current time hour. The system compares the data of all junctions and determines the most populated junction of the city. This project proposes the GRU model in order to predict the traffic using deep learning by importing Sklearn, Keras and TensorFlow libraries.

CONTENTS

CHAPTER 1

COMPANY PROFILE AND TRAINING

1.1 COMPANY PROFILE

Eunoia labs is currently in ‘stealth-mode’ developing products and services in the domain of ‘Education’. It is committed to Individualized and Contextual learning. Eunoia Labs has an experienced pool of experts and engineers with decades of domain experience and expertise in the fields of cloud Services, Open Source tools, ERP for enhanced teaching-learning process, Foss Consultancy and training.

Eunoia Labs - Training division supports working professionals, students and faculty members to adopt advanced software and technology knowledge. Eunoia Labs has the unique courses designed for different levels of engineers and covers theory + practical + projects and internship program to enhance the skills and experience.

Services

- FOSS Consultancy

Facilitate Adoption of Open-source Software by Educational Institutions for both Academia and Administration and Management. We provide Deployment, Training and Customization of various opensources such as OpenStack, Moodle, Odoo, OSTicket, Simulators and emulators, etc.

- Trainings

Customized training on State-Of-Art technologies such as Containers, GoLang, SDN-NFV, etc.

Products

- Sententia: Sententia is a web-based formal writing assistant. Unlike other tools, Sententia does not focus on just correcting the grammar.
- OSTICKET : Online Ticketing System
- Mapaka : Adaptive Assessment Platform. [Under Development]
- Aurora: Individualized Learning platform. [Under Development]
- Duende: Q&A Based Teaching-Learning Platform. [Under Development]

1.2 TRAINING

Week 1: In the lst week of internship, we were assigned with a project based on "Deep Learning" and introduced to a few basics of Python, Numpy, Pandas ,Matplotlib and Seaborns and other libraries for developing the project. we were addressed by the respective guides and explained some concepts of Python. We were introduced to the concepts of NumPy, Pandas in detail.

- **NumPy:** NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, 1/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.
- **Pandas:** It is a Python library used for working with data sets. It has functions for analysing, cleaning, exploring, and manipulating data. Pandas allows us to analyse big data and make conclusions based on statistical theories. Pandas can clean messy data sets, and make them readable and relevant.
- **Matplotlib:** It is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.
- **Seaborn:** Its plotting functions operate on data frames and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots.

Week 2: In the Second week, we were introduced to the concepts of Scikit-learn and TensorFlow in detail.

- **Scikit-learn:** Seikit-learn (formerly known as sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.
- **TensorFlow:** TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.

Week 3: we were assigned with the task - To work on the algorithms in the machine learning and deep learning such as Logistic Regression, Support Vector Machines, Decision tree, Random forest, K-nearest neighbor, GRU, LSTM etc. Later we decided which algorithm best suits with the high accuracy for the project. we were introduced to actual software's required for the development of the project.

I started collecting the datasets required for the project, choose one to Implement and started implementation of the project using selected algorithm and predict traffic probability for 4 different junctions. This is achieved by applying the GRU technique to a model that will have to be trained over a dataset containing the necessary attributes to predict the traffic probability.

Week 4: We implemented the GRU model and completed the project with required accuracy.

CHAPTER 2

INTRODUCTION

Deep Learning is one of the most important and popular emerging branches these days as it are a part of Artificial Intelligence (AI). In recent times, deep learning becomes an essential and upcoming research area for transportation engineering, especially in traffic prediction. Traffic congestion affects the country's economy directly or indirectly by its means. Traffic congestion also takes people's valuable time, cost of fuel every single day. As traffic congestion is a major problem for all classes in society, there has to be a small-scale traffic prediction for the people's sake of living their lives without frustration or tension. For ensuring the country's economic growth, the road user's ease is required in the first place. This is possible only when the traffic flow is smooth. To deal with this, Traffic prediction is needed so that we can estimate or predict the future traffic to some extent. In addition to the country's economy, pollution can also be reduced. The government is also investing in the intelligent transportation system (ITS) to solve these issues. The plot of this research paper is to find different deep learning algorithms and speculating the models by utilizing python3. The goal of traffic flow prediction is to predict the traffic to the users as soon as possible. Nowadays the traffic becomes really hectic and this cannot be determined by the people when they are on roads. So, this research can be helpful to predict traffic. Deep learning is usually done using anaconda software but in this paper, I have used the python program using command prompt window which is much easier than the usual way of predicting the data . In summary, the constructs of this paper consist of ten major sections. These are: Introduction, Purpose of Traffic Prediction, Problem Statement, Related Work, Overview, Methodology, Software Implementation and Conclusion with Future work.

2.1 Objectives

- Design and train Deep learning Model, such as LSTM to predict traffic conditions for 4 different junctions.
- Optimize model performance to enhance the accuracy of traffic forecasts.

2.2 Problem statement

In modern urban environments, traffic congestion is a significant issue affecting millions of people daily. The goal of this project is to develop a deep learning model to predict traffic congestion at four different junctions within a city.

CHAPTER 3

LITERATURE SURVEY

1. **Chen Liu and Yuxuan Sun, 2022**

   This study focuses on the application of transfer learning in traffic prediction, enabling models to adapt to different regions with limited data availability. The results highlight the potential of transfer learning to improve prediction accuracy in data-scarce environments.The authors introduce a novel approach using reinforcement learning to dynamically adjust prediction models based on real-time traffic conditions. This adaptive method enhances the responsiveness and accuracy of traffic predictions under varying conditions.

2. **Liyuan Zhang et al., 2022**

   This paper proposes a deep multi-task learning framework for traffic prediction, which simultaneously predicts traffic flow, speed, and density. The study demonstrates that multi-task learning can improve overall prediction accuracy by sharing information across related tasks. The authors investigate the use of transformer networks for traffic prediction, leveraging their capability to handle long-range dependencies in sequential data. The transformer-based model shows significant improvement in capturing complex traffic patterns.

3. **Rong Liu et al., 2023**

   This research introduces a deep generative adversarial network (GAN) for traffic prediction, aiming to generate more realistic traffic data for training purposes. The GAN-based approach enhances the robustness and accuracy of traffic forecasts.The authors present a hybrid model combining spatial-temporal attention mechanisms with convolutional LSTM networks (ConvLSTM) for accurate traffic flow prediction. The model effectively captures both spatial and temporal dependencies in traffic data.

4. **Hao Xu et al., 2023**

   This paper explores the application of federated learning for traffic prediction, allowing models to be trained across multiple decentralized datasets while preserving data privacy. The study demonstrates the feasibility and benefits of federated learning in traffic forecasting.

The authors propose an end-to-end deep reinforcement learning framework for adaptive traffic signal control and prediction. The framework learns to optimize traffic signals in real-time, leading to improved traffic flow and reduced congestion.

5. **Jing Liu et al., 2024**

   This study introduces a spatiotemporal attention-based neural network (STANN) for traffic prediction, which dynamically adjusts the attention weights based on the importance of different spatial and temporal features. The model outperforms traditional deep learning approaches in various traffic scenarios.The authors develop a deep ensemble learning approach using multiple heterogeneous models to enhance the robustness and accuracy of traffic predictions. The ensemble method aggregates the strengths of different models, providing more reliable forecasts under diverse conditions.

CHAPTER 4

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

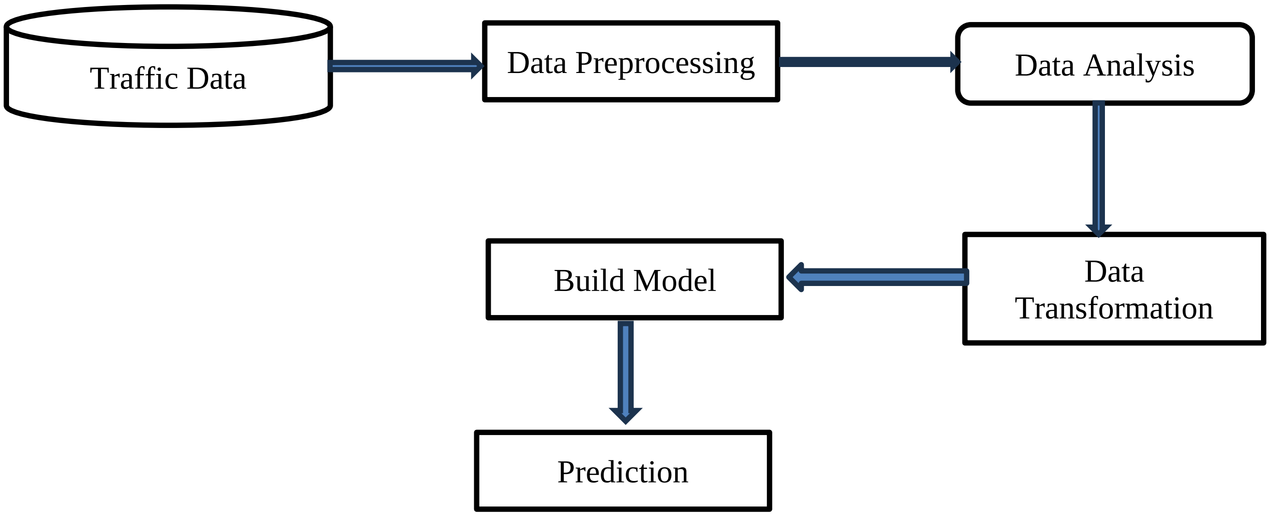


Figure 4.1: Architecture of the system

4.2 DATAFLOW DIAGRAM

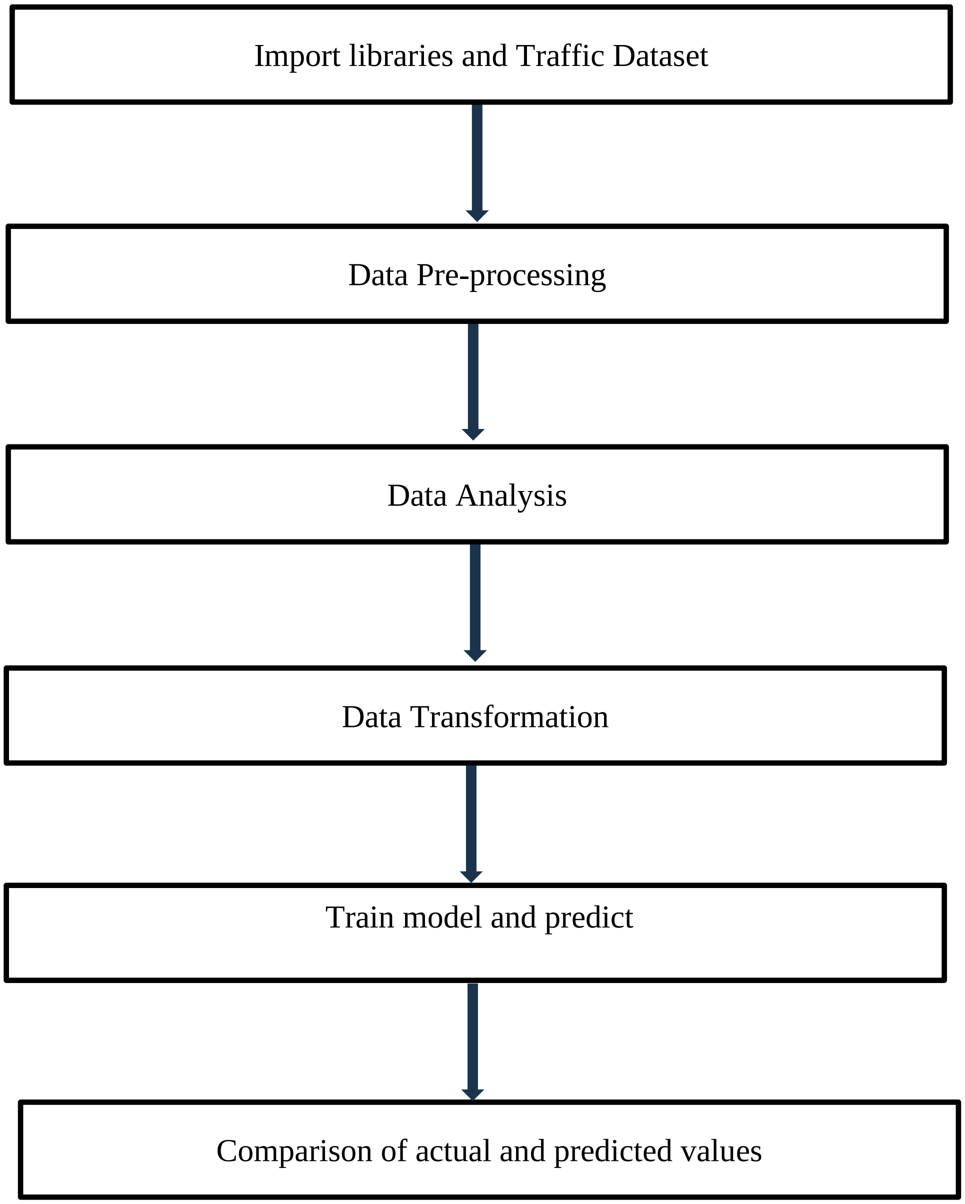

Import libraries and Traffic Dataset
Data Pre-processing
Data Analysis
Data Transformation
Train model and predict
Comparison of actual and predicted values

Figure 4.2: Dataflow diagram of proposed system

4.3 USE CASE DIAGRAM

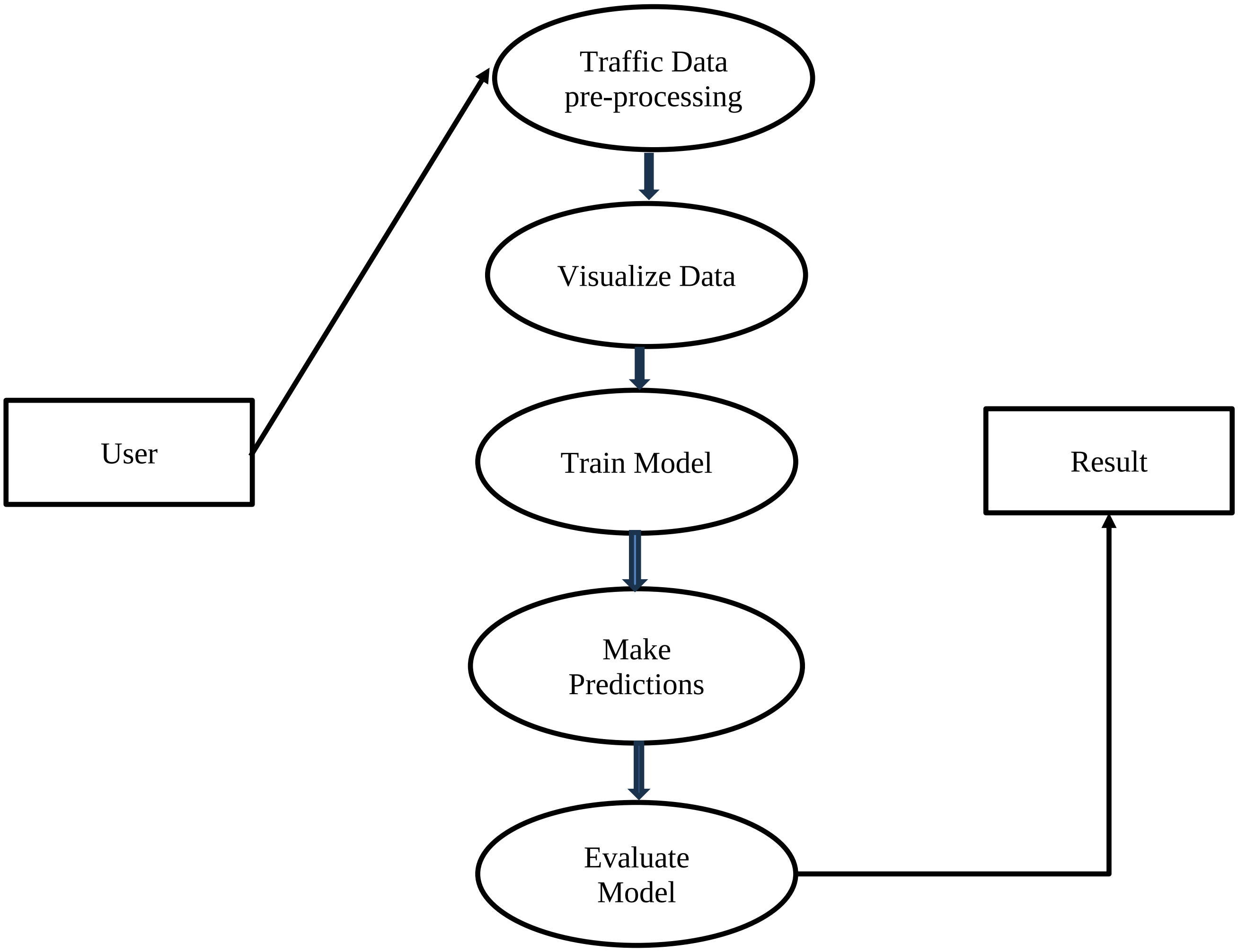


Figure 4.3: Use Case diagram of proposed system.

4.4 SEQUENCE DIAGRAM

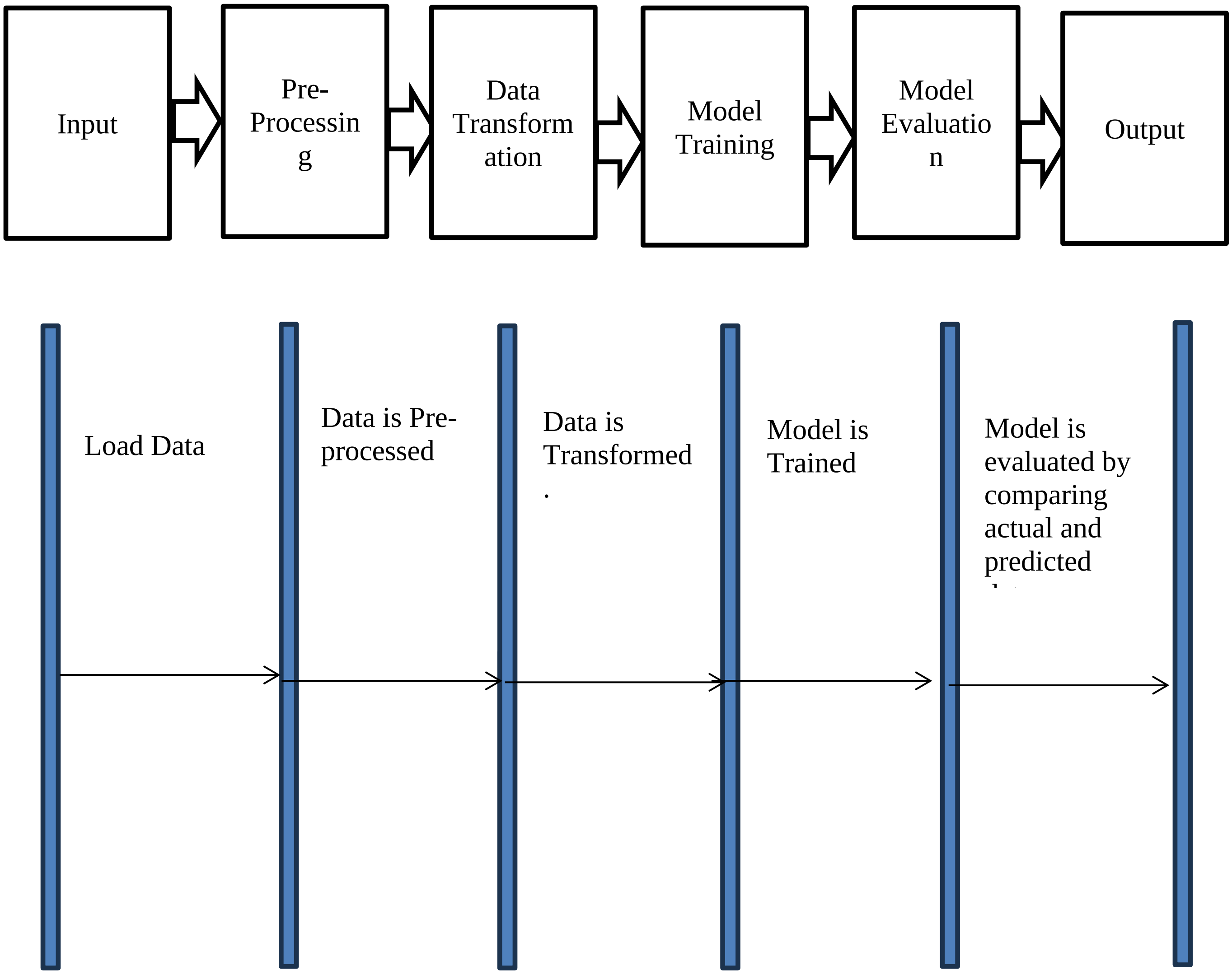


Figure 4.4: Sequence Diagram of proposed system

CHAPTER 5

METHODOLOGY

This section outlines the step-by-step process followed to develop a traffic prediction system using GRU(Gated Recurrent Unit) model. The methodology is structured into several key stages: data collection, preprocessing, exploratory data analysis, data transformation, model training, and evaluation.

5.1 Data Collection

The dataset is downloaded from Kaggle. The “traffic.csv” file contains total number of vehicles in 4 different junctions in different hours **.**The dataset consists of 48120 rows and 4 columns.

5.2 Data Preprocessing

Preprocessing is crucial to prepare the raw data for analysis and modeling. The following steps were performed:

- **Loading CSV File:** The dataset is loaded into a DataFrame using pandas.
- **DateTime Conversion:** The timestamp column is converted to a DateTime object to facilitate time-based operations.
- **Feature Extraction:** Time-based features such as day of the week, hour of the day, and month are extracted from the timestamp.
- **Column Dropping:** Non-essential columns are removed to streamline the dataset for analysis.

5.3 Exploratory Data Analysis (EDA)

EDA involves visualizing and understanding the dataset to identify patterns and relationships:

- **Data Visualization:** Various plots (line plots, histograms) are created to visualize traffic patterns over time.
- **Correlation Analysis:** A heatmap is generated to identify correlations between different features, helping to select relevant features for modeling.

5.4 Data Transformation

Data transformation ensures that the data is in a suitable format for training machine learning models:

- **Pivoting Data:** The data is pivoted to align traffic metrics with time.
- **Normalization and Differencing:** Data normalization is applied to scale the features, and differencing is used to make the time series stationary.
- **Stationarity Check:** The Augmented Dickey-Fuller (ADF) test is conducted to verify the stationarity of the time series data.

5.5 Data Splitting

The dataset is split into training and testing sets to evaluate model performance effectively:

- **Train/Test Split:** The data is divided into training and test sets based on a specific date cutoff.
- **Feature and Target Preparation:** Features and targets are defined for the training and test sets to feed into the model.

5.6 Model Training

The core of the methodology involves training the GRU model:

- **Model Definition**: GRU model is defined using Keras. The models include layers such as GRU layers, Dense layers, and Dropout layers.
- **Training:** The models are trained on the training data using a specified number of epochs and batch size. Mean Squared Error (MSE) is used as the loss function, and the Adam optimizer is employed for optimization.

5.6 Model Used

The classification model used in this project for traffic prediction is:

GRU(Gated Recurrent Unit) Model : In sequence modeling techniques, the Gated Recurrent Unit is the newest entrant after RNN and LSTM, hence it offers an improvement over the other two. Understand the working of GRU and how it is different from LSTM.

GRUs are very similar to Long Short Term Memory(LSTM). Just like LSTM, GRU uses gates to control the flow of information. They are relatively new as compared to LSTM. This is the reason they offer some improvement over LSTM and have simpler architecture.

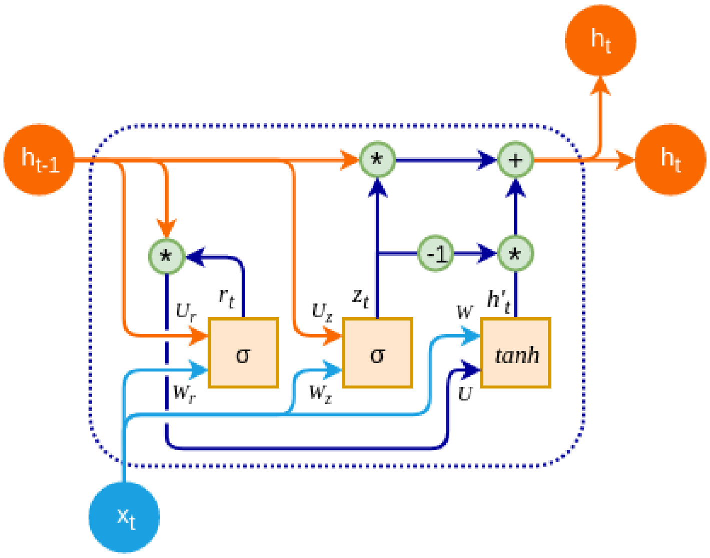


Figure : GRU Model

GRU has feedback connections, unlike conventional feed-forward neural networks. It can handle not only single data points (like photos) but also complete data streams (such as speech or video). GRU can be used for tasks like unsegmented, linked handwriting recognition, or speech recognition.

Architecture Of GRU : Here we have a GRU cell which more or less similar to an LSTM cell or RNN cell. At each timestamp t, it takes an input Xt and the hidden state Ht-1 from the previous timestamp t-1. Later it outputs a new hidden state Ht which again passed to the next timestamp. Now there are primarily two gates in a GRU as opposed to three gates in an LSTM cell. The first gate is the Reset gate and the other one is the update gate.

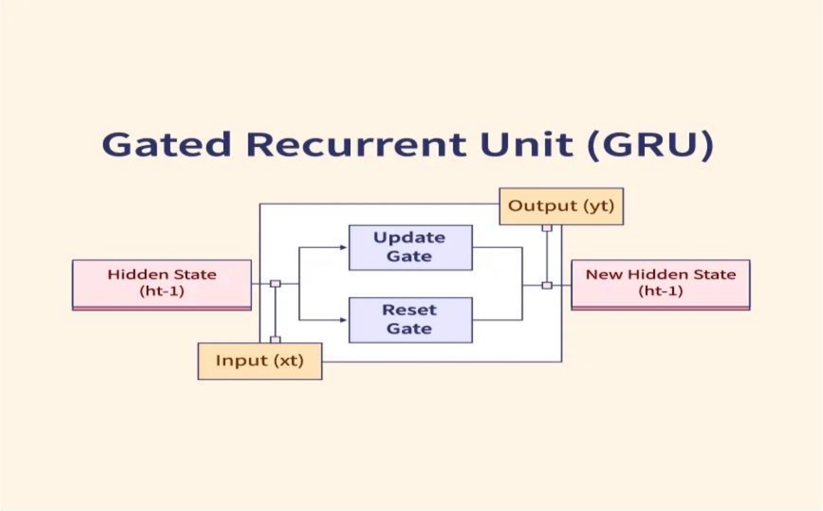


Figure : Architecture Of GRU

Reset Gate (Short term memory) The Reset Gate is responsible for the short-term memory of the network i.e the hidden state (Ht). Here is the equation of the Reset gate. If you remember from the LSTM gate equation it is very similar to that. The value of rt will range from 0 to 1 because of the sigmoid function. Here Ur and Wr are weight matrices for the reset gate.

Update Gate (Long Term memory) Similarly, we have an Update gate for long-term memory and the equation of the gate is shown below. The only difference is of weight metrics i.e Uu and Wu.

No protected cell state is used, hence the gates are applied directly to the hidden state. This difference makes the GRU more computationally efficient than the LSTM. The computation steps of a GRU are the following:

$$r_t = \sigma(W_{ir}x_t + b_{ir} + W_{hr}h_{t-1} + bhr)$$

$$z_t = \sigma(W_{iz}X_t + b_{iz} + W_{hz}h_{t-1} + b_{hz})$$

$$n_t = \tanh\left(W_{in}x_t + b_{in} + r_t * (W_{hn}h_{t-1} + b_{hn})\right)$$

$$h_t = (1 - z_t) * n_t + z_t * h_{t-1}$$

GRU Cycle:

The GRU cycle is divided into five steps:

- **Step1**: Define Network.

The first step is to create an instance of the Sequential class. Then you can create your layers and add them in the order that they should be connected. The GRU recurrent layer comprised of memory units is called GRU(). A fully connected layer that often follows GRU layers and is used for outputting a prediction is called Dense().

- model = Sequential()
- model.add(GRU(2))
- model.add(Dense(1))

- **Step2:** Compile Network.

Compilation is an efficiency step. It transforms the simple sequence of layers that we defined into a highly efficient series of matrix transforms in a format intended to be executed on your GPU or CPU, depending on how Keras is configured.

Compilation requires a number of parameters to be specified, specifically tailored to training your network. Specifically, the optimization algorithm to use to train the network and the loss function used to evaluate the network that is minimized by the optimization algorithm.

- model.compile(optimizer='sgd', loss='mean_squared_error')

- **Step3:** Fit Network.

Once the network is compiled, it can be fit, which means adapt the weights on a training dataset.

Fitting the network requires the training data to be specified, both a matrix of input patterns, X, and an array of matching output patterns, y.

The network is trained using the backpropagation algorithm and optimized according to the optimization algorithm and loss function specified when compiling the model.

The backpropagation algorithm requires that the network be trained for a specified number of epochs or exposures to the training dataset.

- history = model.fit(X, y, batch_size=10, epochs=100)

- **Step4:** Evaluate Network.

Once the network is trained, it can be evaluated.

The network can be evaluated on the training data, but this will not provide a useful indication of the performance of the network as a predictive model, as it has seen all of this data before.

We can evaluate the performance of the network on a separate dataset, unseen during testing. This will provide an estimate of the performance of the network at making predictions for unseen data in the future.

The model evaluates the loss across all of the test patterns, as well as any other metrics specified when the model was compiled, like classification accuracy. A list of evaluation metrics is returned.

- loss, accuracy = model.evaluate(X, y)

- **Step5:** Make Predictions.

Once we are satisfied with the performance of our fit model, we can use it to make predictions on new data.

This is as easy as calling the predict() function on the model with an array of new input patterns.

- predictions = model.predict(X)

5.7 Model Evaluation

The predicted values are tested for test data and evaluated.

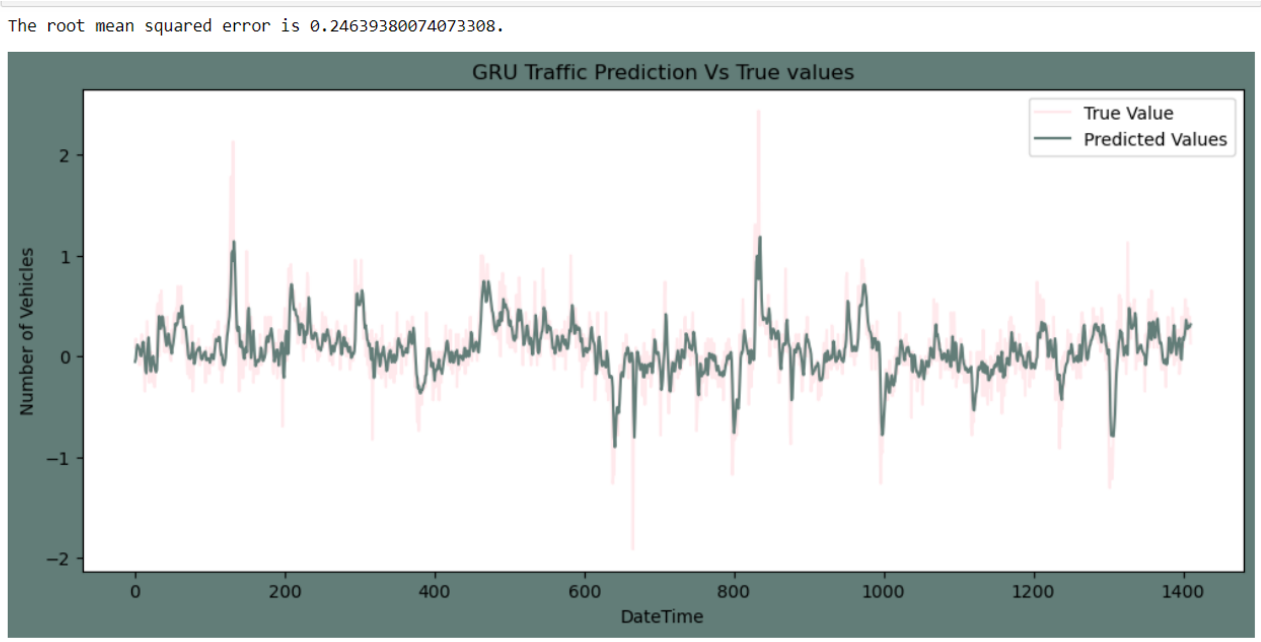
Accuracy: This is the proportion of correct predictions out of all the predictions made by the model. It is a commonly used metric for binary classification problems like water potability detection. However, accuracy can be misleading if the data is imbalanced (ie., one class is much more prevalent than the other), as a model that always predicts the majority class can have a high accuracy but not be useful.

Precision and recall: Precision is the proportion of correct positive predictions out of all positive predictions made by the model, while recall is the proportion of correct positive predictions out of all actual positive instances in the data. These metrics can be useful in cases where the cost of false positives or false negatives is high.

F1-score: This is a harmonic mean of precision and recall, and can be useful for imbalanced datasets where both precision and recall need to be considered.

CHAPTER 6

RESULTS



```
The root mean squared error is 0.24639380074073308.
```

Figure 6.1: Traffic Prediction at Junction 1.

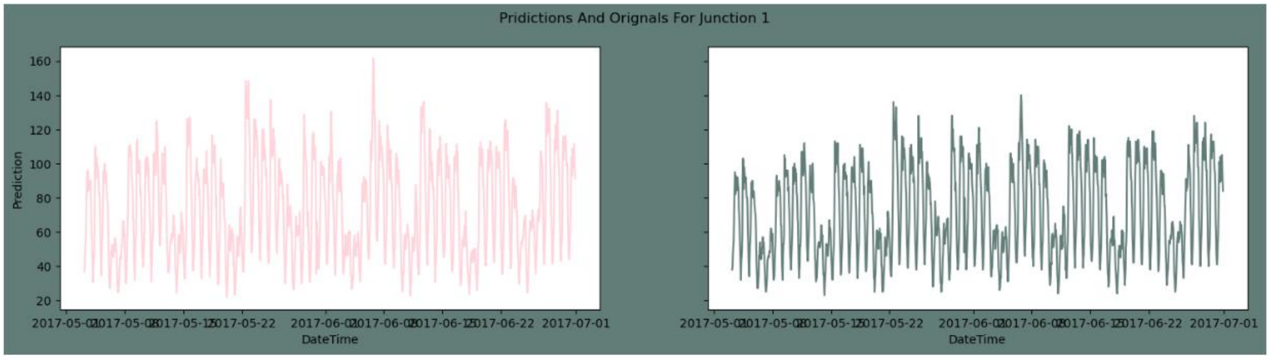


Figure 6.2: Comparison of prediction and actual results at junction 1.

```
The root mean squared error is 0.556422310749018.
```

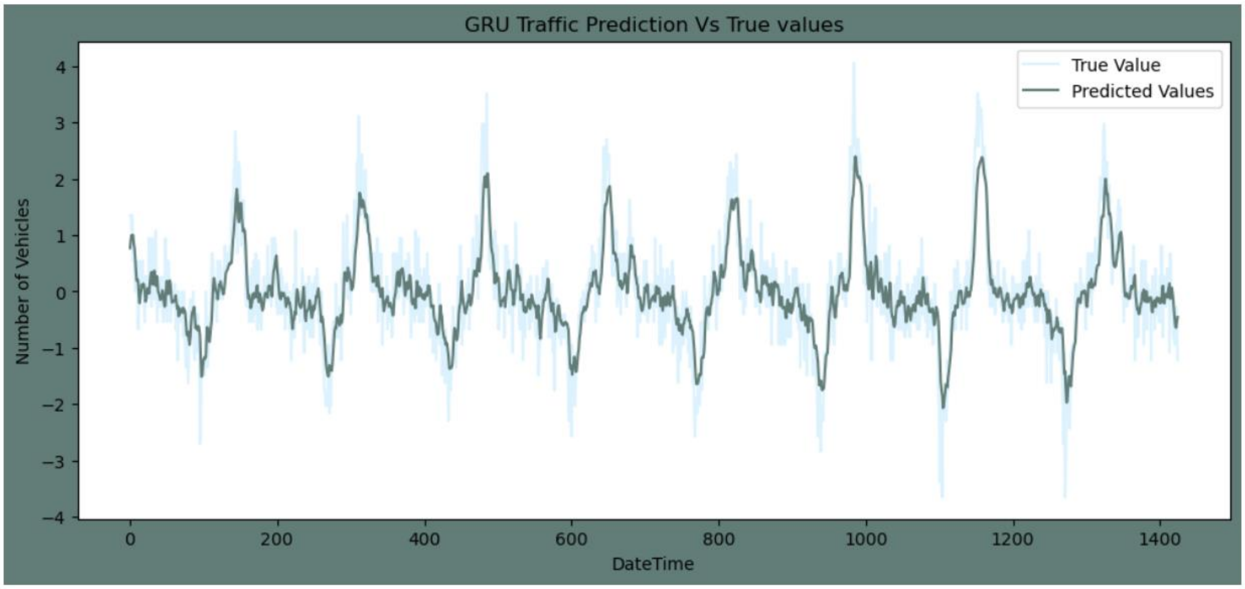


Figure 6.3: Traffic Prediction at Junction 2.

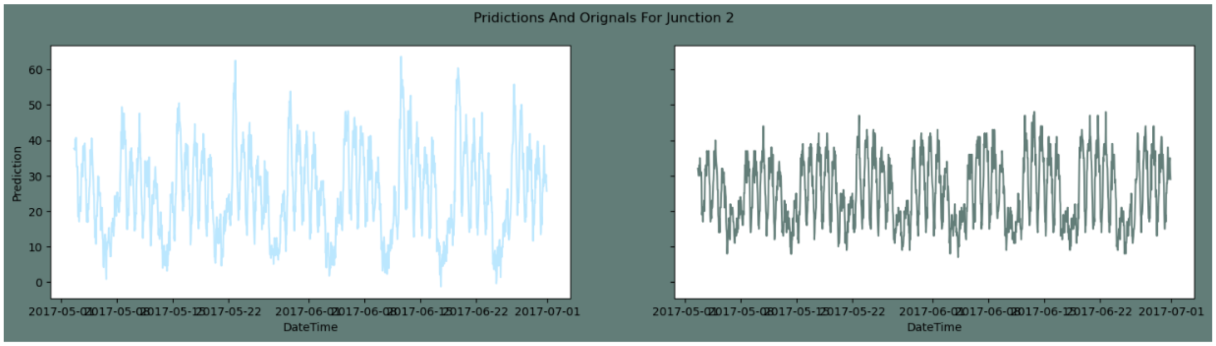


Figure 6.4: Comparison of prediction and actual results at junction 2.

```
The root mean squared error is 0.6097630487723705.
```

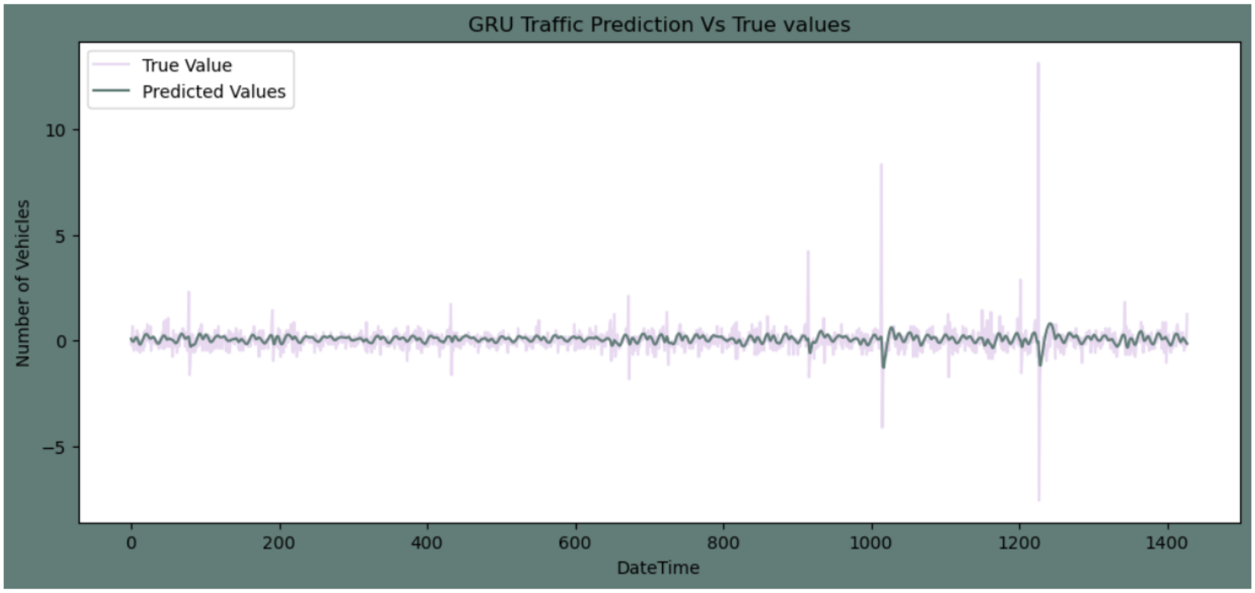


Figure 6.5: Traffic Prediction at Junction 3.

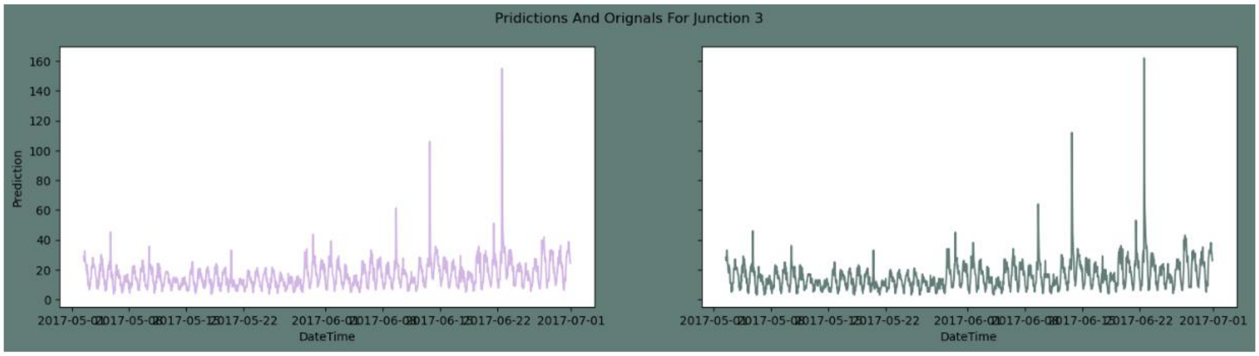
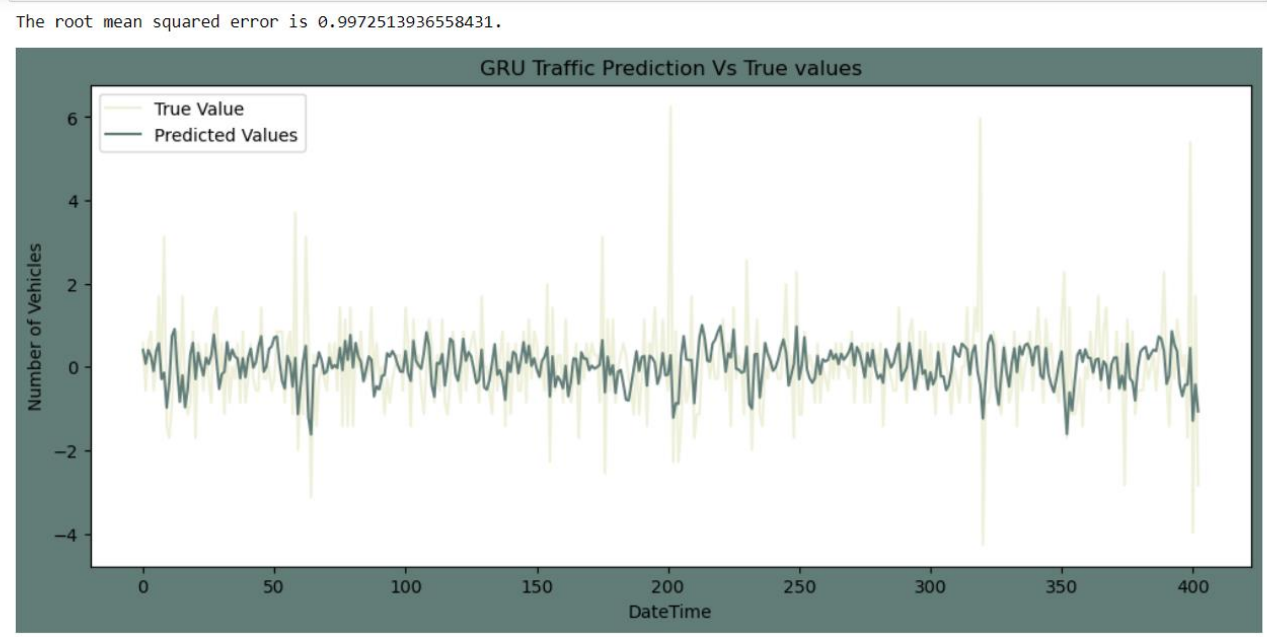


Figure 6.6: Comparison of prediction and actual results at junction 3.



```
The root mean squared error is 0.9972513936558431.
```

Figure 6.7: Traffic Prediction at Junction 4.

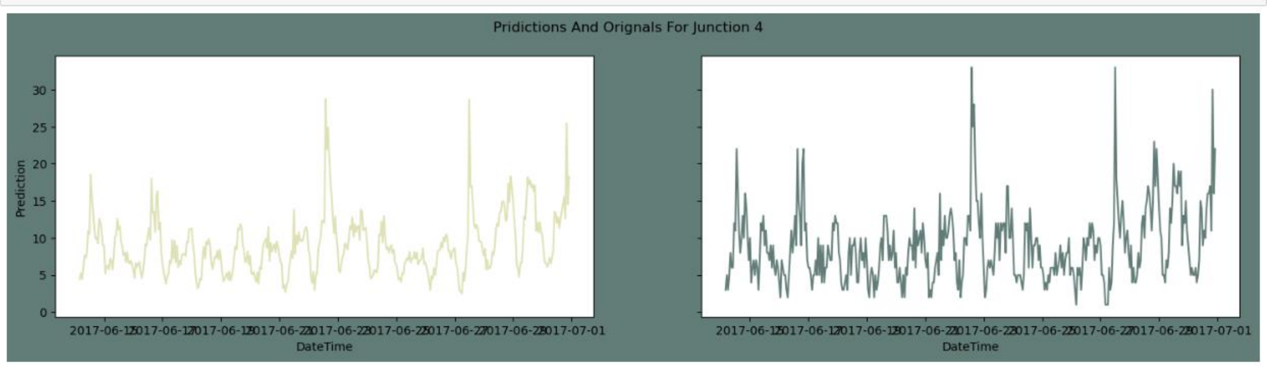


Figure 6.8: Comparison of prediction and actual results at junction 4.

CHAPTER 7

CONCLUSION

In the system, it has been concluded that we develop the traffic flow prediction system by using a deep learning algorithm. By using regression model, the prediction is done. The public gets the benefits such as the current situation of the traffic flow, they can also check what will be the flow of traffic on the right after one hour of the situation and they can also know how the roads are as they can know mean of the vehicles passing though a particular junction that is 4 here. The weather conditions have been changing from years to years. The cost of fuel is also playing a major role in the transportation system. So, this prediction can help judging the traffic flow by comparing them with these 2 years data sets. The forecasting or the prediction can help people or the users in judging the road traffic easier before hand and even they can decide which way to go using their navigator and also this will prediction will be also helpful.

REFERENCES

[1]. Vlahogianni, E. I., M. G. Karlaftis, and J. C. Golias. Optimized and Meta-Optimized Neural Networks for Short Term Traffic Flow Prediction: A Genetic Approach. Transportation Research Part C: Emerging Technologies, Vol. 13, No. 3, 2005, pp. 211–234.

[2]. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2845248/

[3]. https://jupyter.org/

[4]. Bao G, Zeng Z, Shen Y. Region stability analysis and tracking control of memristive recurrent neural network. Neural Netw. 2017;5(1):74-89.

[5]. Jiang, X., and H. Adeli.Dynamic Wavelet Neural Network Model for Traffic Flow Forecasting. Journal of Transportation Engineering, Vol. 131, No. 10, 2005, pp. 771–779

[6].Identification Traffic Flow Prediction Parameters AnuchitRatanaparadorn Department ofIndustrialEngineering,KasetsartUniversity,Thailand,AnuchitRatanaparadorn,SasivimolMeeampol ,ThaneeratSiripachana,Pornthep,Anussornnitisarn, 19-21 2013, zadar, Croatia, international conference

[7]. YuhanJia, Jianping Wu, and Ming Xu, Traffic Flow Prediction with Rainfall Impact Using a Deep Learning Method,Journal of Advanced Transportation, 2017.

[8]. Felix Kunde Alexander Hartenstein Stephan Pieper Petra Sauer, Traffic prediction using a Deep Learning paradigm, CEUR-WS.org, 2017

[9]. https://www.kaggle.com/fedesoriano/traffic-prediction-dataset

[10]. https://www.hindawi.com/journals/jat/2021/8878011/

[11]. https://www.shanelynn.ie/pandas-iloc-loc-select-rows-and-columns-dataframe/

[12]. https://matplotlib.org/2.0.2/api/pyplot_api.html

[13]. Ioannis LoumiotisRoad Traffic Prediction Using Artificial Neural Networks 2018

[14]. https://www.catalyzex.com/s/Traffic%20Prediction

[15]. https://www.geeksforgeeks.org/formatting-dates-in-python/

[16]. Amit Kumar Tyagi, Meghna Mannoj Nair, Deep Learning for Clinical and Health Informatics, in the book "Computational Analysis and Deep Learning for Medical Care: Principles, Methods, and Applications", 28 July 2021, DOI: https://doi.org/10.1002/9781119785750.ch5