FAST-NATIONAL UNIVERSITY OF COMPUTER & EMERGING SCIENCES-KARACHI CAMPUS

COURSE TITLE: COMPUTER NETWORKS

PROJECT TITLE: "A CYBERPHYSICAL SYSTEM FOR DATA-DRIVEN REAL-TIME TRAFFIC PREDICTION ON THE LAS VEGAS I-15 FREEWAY"

SUBMITTED TO:

DR FARRUKH SALIM

DATE OF SUBMITTAL: 5th MAY 2023

GROUP MEMBERS:

INSHA SAMNANI (20K-0247)

ISMAIL AHMED ANSARI (20K-0228)

TABLE OF CONTENTS

ABSTRACT:

Traffic congestion is increasing rapidly, highlighting the need for traffic management, and focusing on the importance of traffic influx outflux predictions. ITS (Intelligent Transportation Systems) have played a vital role in managing traffic congestion. As the problem of Traffic congestion in urban areas has erupted as a new problem, we have used a data-driven technique for accurate short-term predictions for traffic pattern identification using CPS (Cyber-Physical System) proposed on the Las Vegas I-15 Freeway. The CPS (Cyber-Physical System) divides into three layers Field layer, the Connectivity layer, and the cloud–like cyber layer. The layers work with deep neural network models with error recurrence. Periodic training is done in intervals to maintain prediction accuracy. The results of CPS (Cyber-Physical System) with DNN (Deep Neural Network) provide an effective solution for traffic management and highlight the potential of the model proposed in addressing real-world transportation challenges.

INTRODUCTION:

This report bids on traffic congestion issues and the need for traffic control management, such as economic, environmental, and health impacts on urban residents. These issues are necessary to address traffic snags in managing traffic congestion. Considering past studies, a significant remark of traffic prediction has been done but in a controlled scenario. Our research on the previous research gap has made a detailed and calcified execution capable of predicting traffic patterns in real time with extravagant results. The reason behind it is the incorporation of missing statistics in the data. In addition to it, periodic maintenance of prediction accuracy was absent. The report aims to analyze these issues, implementing two novel real-time traffic prediction models based on Neural Networks with data loss prevention techniques or error recurrence. The research proposal includes CPS (Cyber-Physical System) that adapts the real-time data from the sensors and a series avant-garde of deep learning-based models. The main objective of proposing a Cyber-Physical System in a nutshell is to provide a potential solution to address the traffic congestion problem.

Bottlenecks and congestion in traffic are essential factors that we assess in modern large-scale cities. Following their importance, Intelligent Transportation Systems (ITS) present an efficient solution to improve traffic congestion through DNN. Within ITS, traffic prediction is a crucial task for traffic management, adjustment of routes, and allocation of resources. Furthermore, on-the-fly validation and error recurrence was challenging while dealing with the predictions.

Our report's significance lies in providing a practical real-time solution that proposes real-time traffic predictions with data-driven techniques, such as DNN (Deep Neural Network), that improve prediction accuracy on sudden traffic changes. Concrete contributions are threefold:

- CPS (Cyber-Physical System).
- Sustained dataset from the Las Vegas I-15 freeway on traffic.

- Two avant-garde prediction models based on Neural Networks.

Finally, the scope of this paper has opened a new opportunity in real-time and unforeseen traffic flow predictions.

LITERATURE REVIEW:

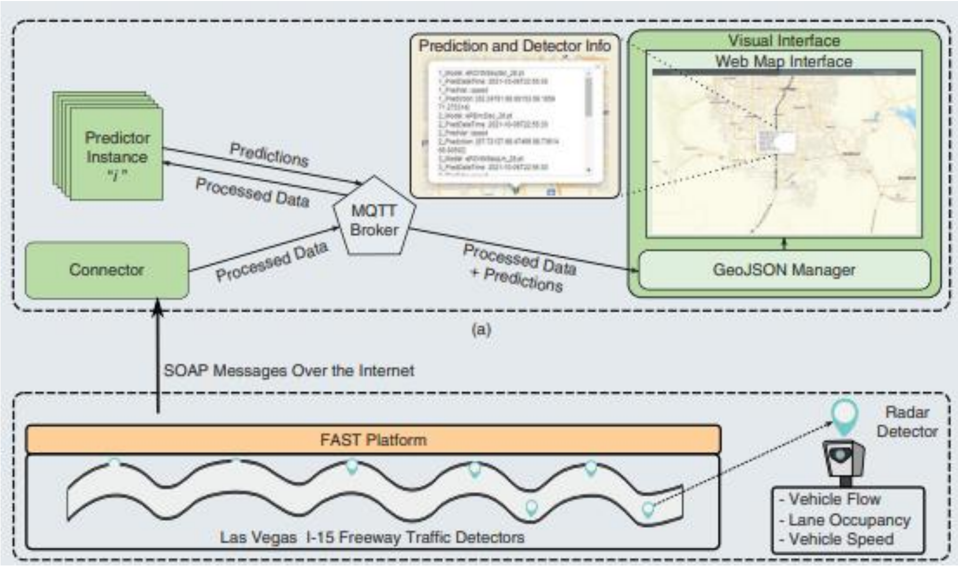
Traffic prediction is a momentous work for ITS (Intelligent Transportation Systems), as it helps to manage traffic flow and reduce congestion. Recent contributions to traffic predictions have focused on data-driven approaches for predicting traffic pattern is done using space-time spatiotemporal data. One such approach, the error feedback recurrent CNN (eRCNN), learns previous prediction errors to improve the subsequent predictions. Studies have shown that the eRCNN outperforms other models, such as Autoregressive Integrated Moving Average (ARIMA) and Support Vector Regression (SVR) that predicts vehicle speed, particularly in congested conditions.

Other approaches include a new GCNN structure called the Diffusion Convolutional Recurrent Neural Network(DCRNN), which captures the degree of spatial autocorrelation between independently measured values observed in geographical space by bidirectional random walks on the graph and temporal dependency by using an encoder-decoder structure. The Graph WaveNet (GWNet) model, based on the WNet model for raw audio generation, uses stacked blocks of dilated 1D convolutions and GCNNs to measure the hidden temporal and spatial dependencies for the traffic data. Comparative studies of these models show that the GWNet model surpasses all other models in predicting the speed of vehicles across prediction horizons of 15, 30, and 60 min. However, most of these studies have finite benchmark datasets and lack real-time update mechanisms for long-term operation. Moreover, most of the datasets on the approaches discussed above tend to be spatially large, because the dataset is for a short time interval, and traffic patterns were rarely included.

ARCHITECTURAL DESIGN:

I. INTRODUCTION:

- Explanation of CPS Architecture used in this work.
- An overview of the three layers: Physical, Connectivity, and Cyber have also been shown in the following diagram.
- Green box (platform services), Orange box (external data source), Gray diamond (internal communication broker.)
- The platform uses JSON (JavaScript Object Notation) and SOAP (Simple Object Access Protocol).



II. PHYSICAL LAYER: FREEWAY AND ARTERIAL SYSTEM OF TRANSPORTATION:

- Description of the FAST platform used for traffic monitoring and control.
- Details of the detectors deployed along four main freeways in Las Vegas.
- Mention of the classical variables used in data-driven traffic prediction.
- Explanation of spatiotemporal images and their benefits.
- Focus on a single segment of the I-15 freeway.

III. CONNECTIVITY LAYER: SIMPLE OBJECT ACCESS PROTOCOL:

- Use of SOAP for interacting with FAST.
- Explanation of SOAP's messaging protocol layer.
- Details of FAST's measurements exposed through SOAP.
- Proposal for implementing a local service for data handling and processing.

IV. CYBER LAYER: CONNECTOR, VISUALIZATION, AND PREDICTOR INSTANCES:

- Description of the cyber layer and its services.
- Breakdown of the cyber layer into three main parts.

1. CONNECTOR:

- The function of the connector in managing data ingestion from FAST.
- Explanation of MQTT as a publish-subscribe protocol with low overhead.
- Details of data transformation actions performed by the connector.
- Examples of data aggregation, variable translation, and resampling.
- Mention of a new "flow" variable generated from measurements of "volume".
- Explanation of data repair through filtering outliers and masking missing data points.

2. VISUALIZATION (FIGURE 1-A):

- Role of visualization in providing a user-friendly interface for data exploration.
- Description of the visualization service's features and functionalities.

3. PREDICTOR INSTANCES:

- Purpose of the predictor instances in generating traffic predictions.
- Details of the machine learning models used for prediction.
- Mention the importance of hyperparameter tuning and model validation.

V. CONCLUSION:

- Recap of the CPS architecture used in this work.
- Summary of the key features of each layer in the architecture.
- Importance of the proposed approach for traffic prediction in Las Vegas.

DATA DESCRIPTION:

I. QUALITY:

The quality of the dataset is most important for producing reliable quality scores. The prediction model sees the credibility of the datasets used for training, validations, and then testing on dataset, which means the higher the quality of data higher its accuracy, and the lower the quality lower the accuracy of the prediction.

II.DATASETS AND PURPOSES:

The researchers created three datasets with different purposes - one for each training, validation, and testing. These datasets were generated using historical traffic data from the I-15 northbound highway in 2018 and 2019.

III.SEASONAL TRAFFIC CHANGES:

With two years of historical traffic data, seasonal traffic changes could be considered while generating the datasets. This means that the datasets were created to mimic the real-world scenario where traffic patterns change throughout the year.

IV.FEATURES OF DATASETS:

The datasets were created using data from sensors that measured the flow of vehicles, road occupancy, and average speed of vehicles vehicle. These variables were used to generate the datasets. Traffic variables used in the study are described in the table below.

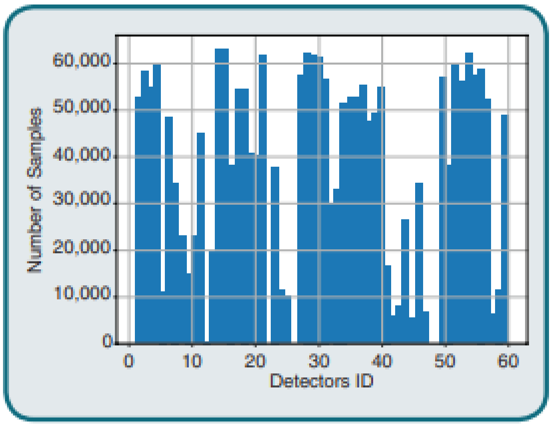
Variable	Range	Unit
Flow	0–3975	Vehicles/hour
Occupancy	0–66.5	Percentage vehicles/lane
Speed	0–85	Miles/hour

V.SAMPLING PERIOD:

The data used to create the datasets were collected at a sampling period of 15 minutes. This means that data were collected and recorded every 15 minutes, and these data points were used to create the datasets.

VI.RAW DATA CHALLENGES:

Prior to dataset generation, the researchers performed an initial analysis of the raw data and identified four primary issues, namely invalid measurements, data losses, unevenly sampled data, and missing data. It was imperative to address these problems. The figure below depicts the data available for each detector in the section of the I-15 freeway being examined.



1.INVALID MEASUREMENTS:

The FAST platform failed or shows invalid output on 16% of the data indicating a fault in collecting the data through the sensors. This data is eliminated from the model.

2.DATA LOSSES:

High data loss rates were detected by the sensors, and it became impracticable to use time series even after the data imputation techniques it wasn't possible. For these reasons, the detectors measuring 50% of the data were removed from the model and research.

3.UNEVENLY SAMPLED DATA:

Owing to communication flaws and data compression techniques implemented by FAST, the interval between samples was not consistently 15 minutes. To mitigate this problem, an algorithm that relied on nearest-neighbor data and linear interpolation was used to resample the database and establish a uniform sampling period.

4.MISSING DATA:

To address the issue of missing data, a data imputation technique was created with the aid of the miceforest library. This involved employing a random forest algorithm that utilizes data correlations to populate the gaps. The product was an imputation model that can be used in real-time to impute new data.

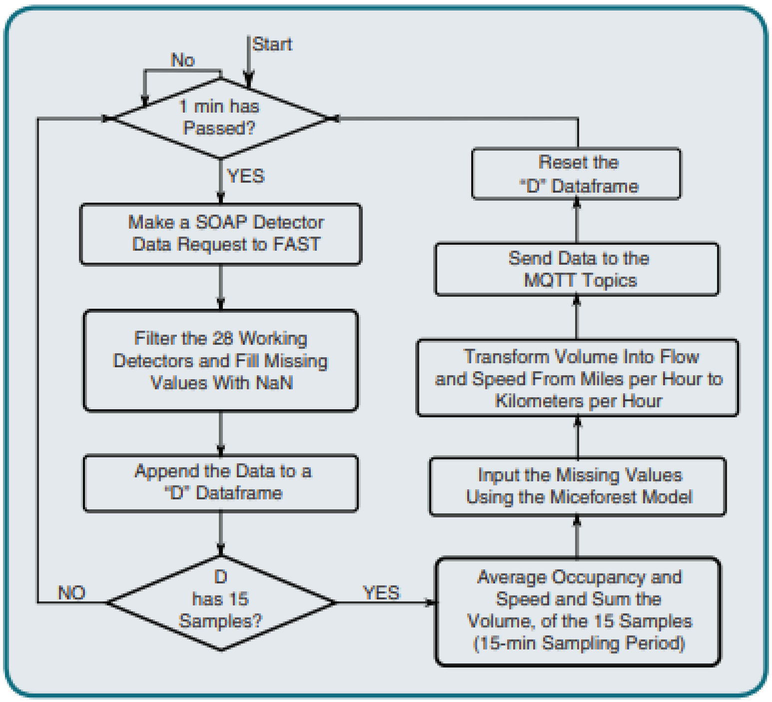
VII.VALIDATED DATA:

Once the four primary issues were resolved, a collection of 28 detectors provided two years' worth of verified data. This data was subsequently utilized to produce three distinct sets of information: Train2018, Val2019, and Test2019.

VIII.RETRAINING:

The occurrence of two significant disruptive events - Project Neon and the COVID-19 outbreak - in 2019 led to the anticipation of poor outcomes from the direct online application of predictors trained with Train2018. As a result, a new dataset called Retrain2021 was generated to progressively retrain the predictors before their online testing on the developed CPS.

The diagram below illustrates the sequence of steps performed by the connector service for processing real-time data.



So, the important takeaways from the data are:

- Good data is important for prediction models to work well.
- Three traffic datasets were created using real-world data from 2018-2019.
- The datasets measure vehicle flow, road occupancy, and average vehicle speed every 15 minutes.
- The data was checked to make sure it was accurate and usable for training, validation, and testing.
- An additional dataset called Retrain2021 was created for updating the predictors before testing them online.
- This was necessary because there have been some changes since the original data was collected in 2019.

The data set description is listed in the following table.

Dataset	Size (Samples)	Collection Dates
Train2018	35,040	1 January–31 December 2018
Val2019	17,520	1 January–31 June 2019
Test2019	17,520	1 July–31 December 2019
Retrain2021	4,504	16 April–2 June 2021

METHODOLOGY:

In transportation, predicting traffic flow is essential for effective traffic management. For this need, data-driven prediction models such as eRCNNLin, eRCNNIter, eRED, and GWNet come into play. Models are designed, in such a fashion, that they forecast the speed of vehicles at different intervals of time on the sensor of each detector, using vehicle speed as a traffic level indicator. The input structures for these models vary, with eRED and two types of eRCNN utilizing a spatiotemporal image. The prediction horizon is 60 minutes, with a sampling period of 15 minutes, and a set of 28 detectors is considered. By employing the data from the previous 6 hours, the model has the capability of predicting a 1-hour timeframe ahead. Overall, traffic prediction models aid transportation management and can improve traffic flow in urban areas. The description below shows the input/output spatiotemporal images for error-recurrent models, namely the GWNet model and the modules used to construct these models. Moreover, descriptions and applications, in our context of research, are also listed as part of the experimental setup.

Tool	Description	Use Case	URL
Supervisor	A client/server system that allows users to monitor and control a number of processes	Services startup, error management, and log generation	http://supervisord.org/
Zeep	A fast and modern Python SOAP client	SOAP communication at the connector service	https://docs.python-zeep.org/en/master/
Mosquitto	An open source message broker that implements the MQTT protocol	Runs the MQTT broker for the internal communication with secured authentication	https://mosquitto.org/
Paho MQTT	A Python MQTT client library	Runs the MQTT client in the services	https://www.eclipse.org/paho/
PyTorch	An optimized Python tensor library for deep learning using GPUs and CPUs	Implementation of prediction models	https://pytorch.org/
GeoPandas	An open source Python library that extends the data types used by pandas to allow spatial operations on geometric types	Construction and formatting of the MQTT messages and handling of the ".geojson" files	https://geopandas.org/
Flask	A lightweight Web Server Gateway Interface web application framework	Implementation of the web interface	https://palletsprojects.com/p/flask/
MapTiler	A mapping software platform that allows using global mapping information	Access to I-15 freeway map information	https://www.maptiler.com/

I.ERCNN:

The eRCNN model uses error feedback to improve future predictions. It consists of a 2D CNN followed by fully connected layers. Two variations, eRCNNIter, and eRCNNLin were proposed for spatiotemporal prediction. eRCNNIter iteratively generates predictions and updates the error vector, while eRCNNLin generates a concatenated prediction vector for the entire prediction window.

1.ERCNNiter:

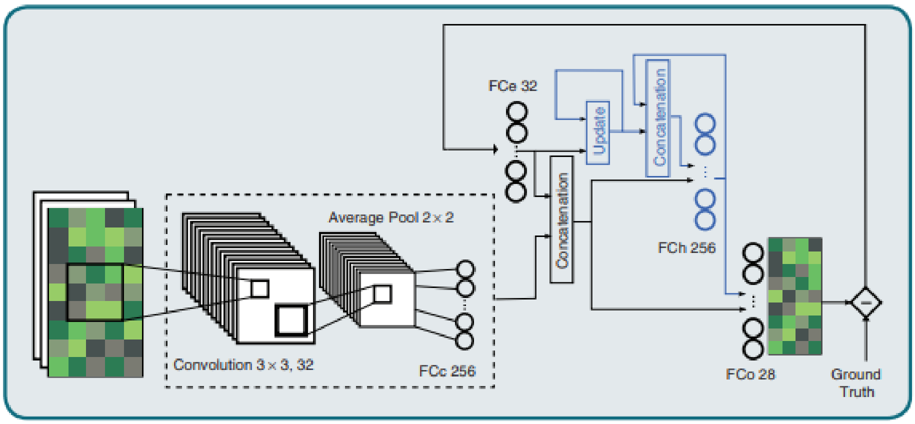
- Create a matrix called Io1 with dimensions W x N, where W is the prediction horizon and N is the number of detectors.
- Apply the eRCNN architecture to the spatiotemporal image I.e. to generate an intermediate output called yconv.

- Use the fully connected layer FCc to produce a hidden state vector hC from the intermediate output yconv.
- Apply the fully connected layer FCe to produce a feedback hidden state vector hE from the error vector Ve.
- Use the auxiliary fully connected layer FCh to produce a hidden state vector hCl of the same size as hC from the Io1 matrix.
- Concatenate the vectors hC, hE, and hCl, and use the last layer FCo to generate the next prediction Io2.
- Update the error vector Ve1 by removing the oldest element and appending a copy of the last element.
- Repeat steps 5-7 for W-1 times to generate the remaining columns of the prediction image.
- Merging of all W columns of the predicted image to generate the final output image.

$$Io = FC_o(\mathrm{concat}(FC_c(y_{\mathrm{conv}}), FC_e(Ve)).$$

$$Io_1 = FC_o(\mathrm{concat}(FC_c(y_{\mathrm{conv}}), FC_e(Ve_1))$$
$$Ve_{j+1} = \mathrm{concat}(Ve_j[1:M], Ve_j[M])$$
$$Io_{j+1} = FC_o(\mathrm{concat}(FC_h(Io_j), FC_e(Ve_{j+1})).$$

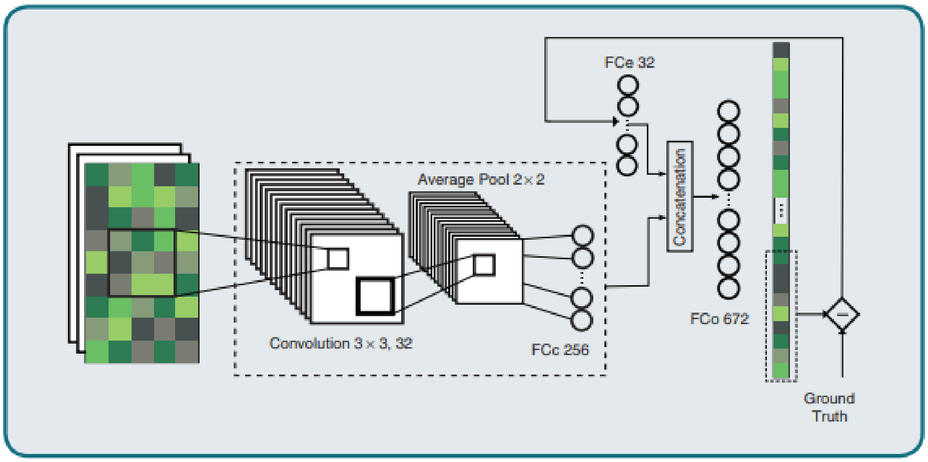
The following figure shows the eRCNNiter. The blue section represents the part of the model that iterates W-1 times to generate the final output.



2.ERCNNin:

- Generate the intermediate output yconv from the spatiotemporal image I using the original eRCNN architecture.
- Utilize the fully connected layer FCc to generate a hidden state vector hC from yconv.
- Generate a feedback hidden state vector hE by passing the error vector Ve through the fully connected layer FCe.
- Concatenate hC and hE to form a hidden vector and pass it through the last layer FCo to produce the concatenated prediction vector Io.
- Resize the concatenated prediction vector Io to a W x N matrix to create the final output image.

The eRCNN depicted in the following figure generates predictions in a single pass.



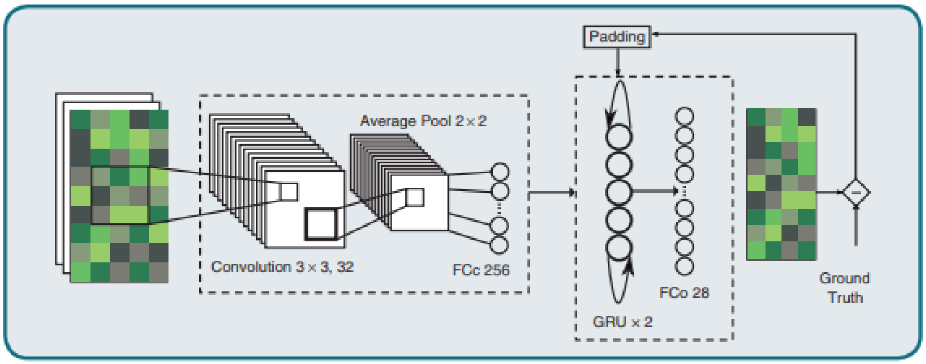
3.ERED:

The eRED model uses error recurrence on an encoder-decoder network that originally used a 1D CNN as an encoder. In the modified version, the encoder is replaced with a 2D CNN to extract temporal and spatial information. The decoder uses the error vector as input to generate the output.

- The original model predicted time series data from industrial applications using a 1D CNN encoder and a gated recurrent unit-based decoder.
- To extract both spatial and temporal information from the input data matrix X, the original model's encoder was replaced with a 2D CNN.
- The 2D CNN generates an encoded vector that serves as the initially hidden state hD0 for the recurrent decoder.
- In addition to its original input, the recurrent decoder now receives an error vector Ve that is created using the last M error values.
- The final hidden state hM r, a vector of size LR x QLR, is available after M iterations.
- A linear layer is used to obtain the final output Io by passing hM r through it.
- The eRED model employs error recurrence to enhance the performance of future predictions.

$$h_D0 = FC_c(y_{\mathrm{conv}}).$$

The eRED model is depicted in the figure below.

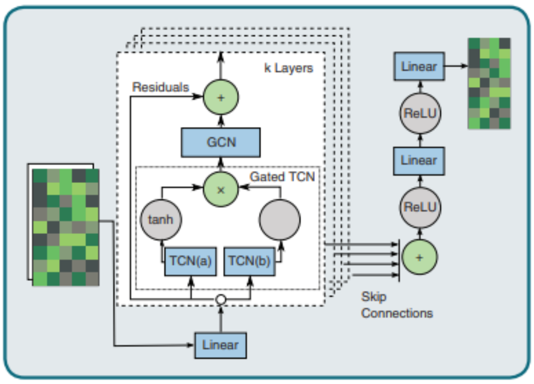


4.GWNet (GCNN):

GWNet is a high-performing GCNN model that predicts traffic using spatiotemporal layers. Each layer contains a GCNN and gated temporal convolution layer to capture various temporal levels. It surpasses other DNN models.

- The model uses graphs as its data structure, which consists of nodes and edges. Nodes represent objects while edges connect related nodes.
- The features of each node and edge in the graph change dynamically over time, resulting in a feature matrix at each time step.
- The objective of the model is to learn a function that can predict the graph features T steps into the future using a sequence of s historical feature matrices.
- GCNNs are a type of Graph Neural Network (GNN) that are intended to extend convolutions to the graph domain. GWNet utilizes these models in its design.
- GWNet is comprised of multiple spatiotemporal layers, each consisting of a GCNN that extracts spatial information and a gated temporal convolution layer (TCN) consisting of two parallel TCNs (TCN(a) and TCN(b)) that extract temporal information.
- Each layer can capture varying temporal levels, and the output of each layer is combined and passed through a group of linear layers to produce the final prediction.

GWNet models working is depicted below in the figure:



EXPERIMENTAL SETUP AND IMPLEMENTATION:

In this section, we will discuss the implementation of the CPS for traffic prediction. We will discuss the hardware and software used to implement the CPS, as well as the tools and libraries used for implementation.

I.TEST2019 RESULTS AND ANALYSIS:

In this section, we will discuss the results of offline testing of the CPS using the Test2019 dataset. Mean Square Error (MSE) and the Mean Absolute Error (MSE) metrics evaluation has been performed by different predictors and compared. The prediction results for the Test2019 dataset are shown in the following table.

Model	MSE	MAE
eRCNN (iterative)	31.61	3.43
eRCNN (linear)	19.74	2.47
eRED	24.92	2.89
GWNet	23.79	2.17

II.ONLINE RESULTS AND ANALYSIS:

In this section, we will discuss the results of online testing of the CPS. We will present the results of the predictors trained with the Retrain2021 dataset and compare them to the predictors trained with Train2018. We will also discuss the performance differences among the predictors and the benefits of conducting periodic incremental retraining.

The following table illustrates the results of online predictions on different models with different levels of retraining.

	No Retraining		Retraining 25%		Retraining 50%		Retraining 45%		Retraining 100%	
Model	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
eRCNN (iterative)	100.8	7.47	61.94	5.54	56.72	5	59.62	5.12	65.57	5.09
eRCNN (linear)	58.22	5.45	39.77	4.01	37.17	3.78	34.8	3.59	34.03	3.53
eRED	133.12	8.62	54.13	4.92	49.96	4.61	47.4	4.35	46.79	4.21
GWNet	44.74	4.2	27.77	2.75	28.17	2.71	27.3	2.71	26.43	2.62

The following table presents the MAE (Mean Absolute Error) results of online prediction.

	No Retraining				Retraining 100%			
Model	15 min	30 min	45 min	60 min	15 min	30 min	45 min	60 min
eRCNN (iterative)	7.76	6.86	7.15	8.11	5.39	4.37	4.98	5.62
eRCNN (linear)	4.93	5.07	5.66	6.13	3.31	3.43	3.61	3.77
eRED	8.69	8.59	8.58	8.63	4.08	4.15	4.25	4.39
GWNet	3.4	4.07	4.52	4.84	2.23	2.58	2.76	2.9

FUTURE IMPLEMENTATION:

In the future, the Cyber-Physical System for data-driven real-time traffic prediction could potentially be implemented in other urban areas with high traffic volume. The system could also be enhanced to incorporate additional data sources such as weather and events, which could have an impact on traffic flow. Furthermore, the system will work with smart traffic management systems to revamp optimization of the traffic flow and eliminate congestion in real time. Ongoing research and development in the field of artificial intelligence and machine learning could lead to the creation of more accurate and efficient prediction models amalgamated with the CPS. Overall, there are many exciting possibilities for the future implementation of this technology in the transportation sector.

CONCLUSION:

In conclusion, traffic congestion is a growing concern that requires efficient traffic management. This report proposed a data-driven solution using CPS and DNN models for real-time traffic prediction with periodic training for accurate results. This study contributed to practical traffic prediction models that can address real-world transportation challenges.