Smart Traffic Management System Solution: Based on Machine Learning and IoT

1st Iftekhirul
ID: 011172006
Department of computer science and engineering
United International University
iftekhirul172006@bscse.uiu.ac.bd

2nd Sharmin Akter
ID: 011171071
Department of computer science and engineering
United International University
sakter171071@bscse.uiu.ac.bd

3rd Mehnaz Hossain Antora
ID: 011162148
Department of computer science and engineering
United International University
mantora162148@bscse.uiu.ac.bd

4th Ishmam Mahmud
ID: 011171083
Department of computer science and engineering
United International University
imahmud171083@bscse.uiu.ac.bd

5th A.K.M. Muzahidul Islam
Computer Science and Engineering
United International University Dhaka, Bangladesh
muzahid@cse.uiu.ac.bd

***Abstract*—Every day, the population grows. For numerous causes, people are migrating from the countryside to the city. As a result, the number of automobiles on the road in the city is increasing, causing major traffic congestion on a daily basis. People waste valuable time in traffic jams in every city. Traffic congestion cause people their valuable money. In this research, we propose a machine learning approach for analyzing and forecasting traffic congestion. Traffic data is an indispensable element of this project. Machine learning techniques are very useful predicting traffic congestion if real-time data is available. Real-time traffic cameras and sensors can be used for collecting data about traffic congestion. We have used a dataset from Kaggle. To study traffic congestion, we have employed several machine learning techniques such as KNN, Linear regression, MLP, and Random Forest regression etc. Random forest regression outperforms all of the other algorithms. We suggest a cloud-based technique for deploying this machine learning model at the end of the study, where it can be used for a variety of purposes. This research may help to reduce our country's traffic congestion if it is implemented properly in our country. All resources can be found here.**

Keywords— Traffic Congestion, IoT, Machine Learning, Cloud

I. Introduction

An effective traffic management system is required for a country to become developed. According to research, traffic congestion costs roughly $3868 million dollar [11] every year in Dhaka, Bangladesh making it one of the most important threats of our economy and daily life. To deal with it, we urgently require a system that reduces traffic congestion, saves our time and money, and is cost-effective for our economy. The goal of this traffic management system is to predict traffic congestion, which are aid traffic management authorities in reducing traffic congestion and providing city residents with real-time traffic information on a daily basis.

One of the most significant challenges is that our city lacks any sensors or surveillance cameras at traffic intersections to collect real-time data, which is a critical component of this project. So, our city must be equipped with IoT device that can collect data and monitor traffic congestion and to create an IoT-based effective traffic management system [1][4], because a city without IoT equipment such as sensors and surveillance cameras cannot be considered a smart city.

There are multiple ways to collect traffic data. Most of the traffic congestion algorithm uses sensor to collect data which is costly [2][3]. Also, it is not possible to collect accurate data through sensor all the time. We recommend employing traffic cameras to collect data which is less costly. The collected images are further processed to identify density of traffic. The patterns in traffic data are diverse. For example, traffic congestion or density is lower on weekends than on weekdays. Another fact is that different times of the day have varied levels of congestion. As a result, we can conclude that the level of density varies depending on the day and hour. Traffic density can be calculated in a variety of ways. The calculation of density depends on which IOT system we use. We provide a method for calculating traffic congestion or density in the data description section.

Deep learning [13] and machine learning [14] have become critical components of every real-world problem solution. We have opted to use a machine learning technique to implement this traffic congestion project. There are numerous machine learning algorithms available. Different machine learning algorithms are utilized depending on the data structure. Classification and regression are two different types of algorithms. We mostly employed regression type algorithms because our obtained data is of the regression type.

We have divided our research into five sections in this paper. The first is an introduction, in which we show an overview of our activities. In the second half, we have read some literature papers and demonstrated all studies that are relevant to our work briefly. The third section is methodology which includes data description, preprocessing, model creation, and cloud deployment. The experimental result is in fourth part. Finally, there's the conclusion and future work.

In this paper, we have worked on these issues

- Studying and analyzing literature related to traffic congestion
- Collect and analyze data for correct precision
- Applying Machine Learning algorithms to collected datasets and predicting the results
- Propose a method to deploy ML model to cloud

II. Background Study and Related Works (Literature Review Goes Here)

In this section we have discussed background study related to our project and reviewed different thesis papers.

A. Background Study

We have read different journals and thesis papers based on traffic congestion. Most of them have used sensors for collecting data. Some of the papers have used real time data analysis technique which is very low. Also, we have reviewed different machine learning techniques [3]. Also, we have read books based on machine learning techniques [9]. We have reviewed the market place if there is any application for solving traffic congestion. We didn’t find any application related to it. We have collected datasets from authentic and reliable websites for our data analysis.

B. Literature Review

IoT-based street lighting and traffic management system [1]: This research paper includes energy-saving methods and smart traffic control systems. They have proposed a model which includes LDR sensors, servo motor, IR motion sensor, LED, solar panel, barrier Fig. 1.

They may control the traffic by installing cameras and monitoring them using the internet. Also, if there is no traffic on the road the lights can remain off. If there is any traffic on the road the lights automatically turn on. They have used a barrier system. At the time of red light, a barrier must appear and if the traffic signal is green the barrier must disappear if there are any accidents on the road the barrier must appear to prevent vehicles from going into the danger zone and it must show yellow light. The barrier must use a servo motor.

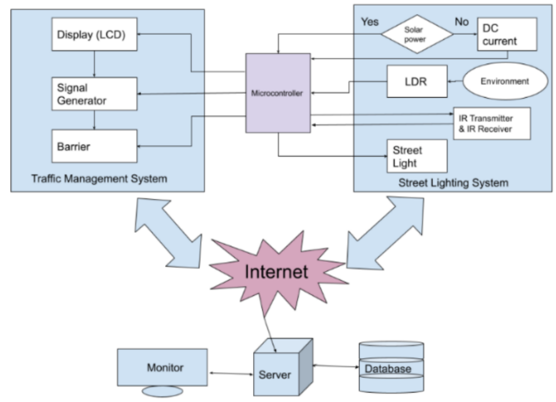

Fig. 1. The System Architecture

Traffic Management System Using IoT Technology [2]: To reduce the ever-increasing traffic congestion, a number of methods have been designated as improved lanes for emergency vehicles in urban areas. This paper has been compared to different methods for managing traffic.
Implementing a proper Intelligent Transportation System here used two parameters for updating the database dynamically: categories, and priority of vehicles. With these two parameters categories and the priority of vehicles, all the vehicle’s information is stored in the database. Here selecting some methods that are required the objective.

- Traffic Management System (TMS)
- Green wave system
- RFID tags
- Wireless Sensor Network (WSN)
- Global System for Mobile Communication (GSM)
- Infrared Sensor (IR)

After comparing these methods, the confirmed suitable method is the mobile application AKA user interface (UI) and this way traffic data is quickly retrieved and sent to Big Data for processing and estimating traffic density. Then suggest alternative ways to reduce traffic congestion. So, on a priority basis, emergency vehicles can be allocated through the intelligent traffic system.

Machine Learning-based traffic congestion prediction in an IoT-based Smart City [3]: The authors proposed a machine learning approach over deep learning for traffic congestion analysis and prediction on a specific path. Here in this paper, the authors talked about a smart city where all roads are equipped with sensors that is essential for traffic monitoring and congestion data collection.

The collected data must be real-time data which is collected by different sensors from different junctions. They have divided the data into two sections T1 and T2. T1 is for training and T2 is for testing the accuracy of the model. They leveled the data using the CONGESTION ALGORITHM which is proposed by Suguna Devi et.al.

They prepared threshold "t" conditions. If the average speed of present vehicles is less than "t", they assume it as congestion otherwise not. After that, they prepared their dataset which is T1. By using that dataset, the models are trained. Then they used the T2 dataset for prediction.

The models are operated to use five supervised machine learning algorithms: Decision tree, Random Forest, SVM, MLP, and logistic regression, with logistic regression outperforming them all TABLE I.

TABLE I. TABLE OF MACHINE LEARNING ALGORITHM (FROM PAPER NO. 3)

ML	Precision	Recall	Accuracy
Decision Tree	100	99.4	99.88
Random Tree	100	99.4	99.88
SVM	100	99.4	99.88
MLP	99.9	99.4	99.86
Logistic regression	100	99.5	99.9

IoT Based Dynamic Road Traffic Management for Smart Cities [4]: The problem of traffic jams and various accidents is increasing day by day in the metropolitan cities. The idea of smart city has been thought to solve various traffic congestion problems which may further improve the quality of life in the city. In this paper the authors have tried to find the possibility of how in smart cities they are able to use embedded PCs in dynamic traffic light optimization and Control using WebIOPI REST API Fig. 2.

The plan outlined in this paper can improve our traffic management if the information about the amount of traffic congestion on a road automatically goes to the RPi unit controlling light intersects. This allows the authorities to make better decisions quickly. If this arrangement can be adopted in front of emergency roads such as hospitals, schools, children's playgrounds, etc. then there will be some relief from traffic congestion.

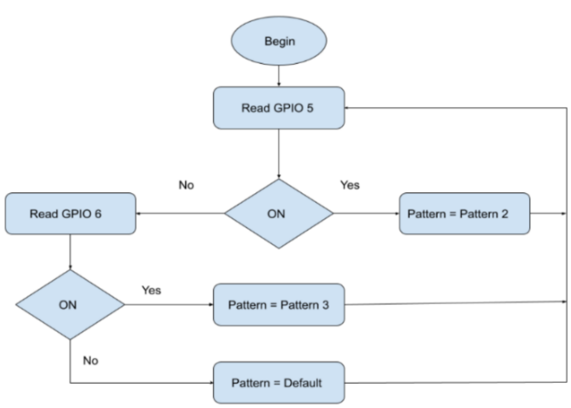

Fig. 2. Dynamic Traffic Light controlling Algorithm

Road Traffic Outlier Detection Technique based on Linear Regression [5]: This paper is based on a statistical model which identifies the road traffic temporal outliers. Used two statistical models Z-score and linear regression technique combination to detect outliers in traffic data. Z-score represents the number of the standard deviation a data point is from the mean value of the observed data set and calculated that observed data at any given time.

After calculating the Z-score, if the standard deviation of the observed vehicle count number is between 3 to -3 then the linear regression technique is applied to infer whether any unusual traffic incident happened or not. The system considers two traffic parameters: average speed and the number of vehicles.

The regression line equation is: y = c + mx

RMSE equation is: RMSE = $\sqrt{\frac{\sum_{i=1}^{n}(P_i - A_i)}{N}}$

Once the regression line is drawn, the linear regression model which is able to predict the average speed of the cars based on the observed number of cars. Then it calculates the error from the predicted average speed value.

In the second phase of this algorithm, it calculates RMSE of the linear regression model created based on traffic data set as per Equation and 3.0416700390674594 is used as a reference to identify the abnormal data points in TABLE II.

TABLE II. RESULT OF TEST TRAFFIC DATABASE WITH Z-SCORE VALUE (FROM PAPER NO. 5)

RoadLinkID (rid)	*Date* (d)	*No.ofVehicles* (v_{count})	*Average speed* (s_{avg})	$z(n_i^j)$	$z(s_i^j)$
RL1	16/6/20	49	66	-0.665815	0.842077
RL1	17/6/20	52	65	-0.595245	0.754042
…	…	…	…	…	…
RL1	28/6/20	145	23	1.534548	2.675766

Machine Learning Approach on Traffic Congestion Monitoring System in Internet of Vehicles [6]: Predicting traffic congestion data only using sensors is a costly method as there is not enough information available. So, in this paper they have used GPS system along with a gaussian process in ML for prediction of traffic data. They have used following equations:

$$q(A) = \frac{d(A)}{|A|}$$

$$k(A) = \frac{t(A)}{|A|}$$

$$V(A) = \frac{d(A)}{t(A)}$$

Where q(A) means flow of vehicles on road segment, k(A) means vehicle density and V(A) means speed of vehicles, d(A) is total distance travelled by the vehicles and t(A) represents total time spent by vehicles in that particular road sector, |A| represents the area covered by the vehicle. They have used training set, prediction set, road sector data as their predication parameter.

We have categorized the above six papers by IoT and machine learning. Papers [1][2][4] give us knowledge about IoT and papers [3][5][6] give us a clear idea about machine learning.
Our paper has detail and accurate machine learning algorithm which shows how the data can be calculated that the other papers does not have.

III. METHODOLOGY

The methodology shows our strategy for proposed technique and how our process is going on.

3.1 Data Description:

The dataset for this project was obtained from Kaggle. [https://www.kaggle.com/fedesoriano/traffic-prediction-dataset]. Since we don't have any traffic congestion statistics for our country. This dataset can be used as an idle dataset for this research TABLE III.

TABLE III. TABLE OF EXAMPLE OF COLLECTED DATASET

Data Time	Junction	Vehicles	ID
11/1/2015 0:00	1	15	20151101001
11/1/2015 1:00	1	13	20151101011
11/1/2015 2:00	1	10	20151101021

In our collected dataset, we have 4 columns where ID is not necessarily important for our model. From Data time column, we have to extract information which may be crucial for model. The Vehicles column is our target column. It contains data on traffic density. Here's an example of how an IoT system can be used to track traffic Fig. 3. The density calculation is dependent on the IOT technology we are using. We offer this approach for surveillance camera monitoring system that we discovered in a study [7].

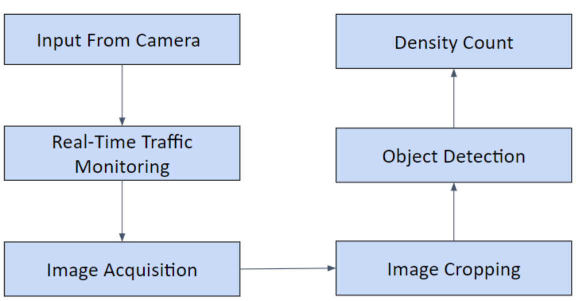

Fig. 3. Process of density measurement

This picture describes the general system model of video processing. First the system takes video signal and image acquisition from fixed cameras. The second part is image cropping where the system crops the image to their convenience. The third part enhances the image quality and in the last part the system counts the density (number of vehicles).

3.2 Preprocessing:

The first and most critical step in preparing data for a model is to preprocess it. We must examine our data to see if there are any missing values or empty cells. Fortunately, the dataset is clean. The ID column is removed from the dataset, and certain new columns such as year, month, day, weekday, and hour which are extracted from datetime column are included with the dataset. The minmax scaler was used to modify features by scaling each one to a specific range (0-1).

3.3 Model Generation:

We have picked Visual Studio Code as our development environment. VS Code allows for real-time collaboration and includes a variety of extensions to aid you in your development. Many machine learning algorithms, such as linear regression, KNN, K-nearest neighbor, Random Forest, and others, are included in the sklearn library [10]. We utilized matplotlib to visualize data patterns and comprehend data patterns [9]. In the train-test set, the dataset has been split 70-30. The Train dataset is used to train the machine learning model Fig. 4., whereas the Test set is used to verify the trained model's accuracy.

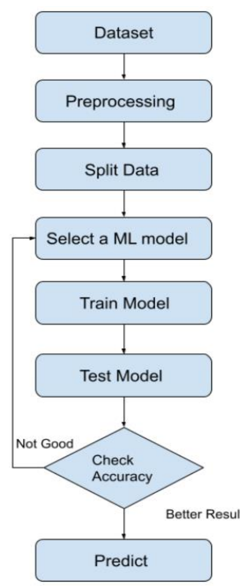

Fig. 4. Approach of Machine learning model

After preprocessing and using standard scaling, we have applied several algorithms to our model. Most of them are Regression algorithm. The applied algorithms are KNN, Multiple Linear Regression, Random Forest, Multi-layer perceptron, Support Vector Regression and Logistic Regression.

3.4 Deploy in Cloud:

There are many of cloud computing platforms such as Amazon web service, Google Cloud Platform, Microsoft Azure etc [12]. for deploying machine learning model. Google App Engine, which is a Platform as a Service on the Google Cloud Platform, is one of the ways to deploy models. There are also cost-free alternative platforms. Heroku [8] is one of them. Heroku provides cloud Platform as a Service.

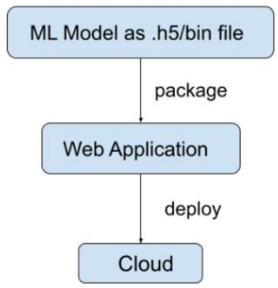

Fig. 5. Steps of deploying ML model

There are a few things to take before deploying machine learning model Fig. 5. The first step is to save the machine learning model as a.h5 or bin file. Then we must integrate this file into a web application, such as Flask as framework. The final step is to deploy to the cloud.

IV. Experimental Results

In this section we have written our experimental results.

4.1 Evaluation:

In order to find accuracy of the model, we used some evaluation metrics in our model.

$$R^2 = 1 - \frac{RSS}{TSS} \quad (1)$$

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - y'_i)^2 \quad (2)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} (P_i - A_i)}{N}} \quad (3)$$

R2 score is a statistical metric that gives us information about our regression model performance. The ideal score is one, which means that the closer the score is to one, the better the model predicts. A popular loss function is mean square error. The lesser the mean squared error, the better the result. The square root of MSE is RMSE. Its purpose is to determine the model's inaccuracy.

4.2 Result:

The results of our project are shown in the table below. As we can see, random forest performs best among them. K-nearest Neighbor and MLP also provide good results TABLE IV.

TABLE IV. The Result of Collected Dataset

ML Algorithm	R2 Score	MSE	RMSE
KNN Regression	0.939	24.947	4.994
Multiple Linear Regression	0.601	164.795	12.837
Random Forest Regression	0.967	13.449	3.667
MLP Regression	0.904	39.658	6.297
Support Vector Regression	0.864	56.091	7.489
Logistic Regression	0.079	151.726	12.317

Here is a visualization of our results in different regressions:

1. R^2 score

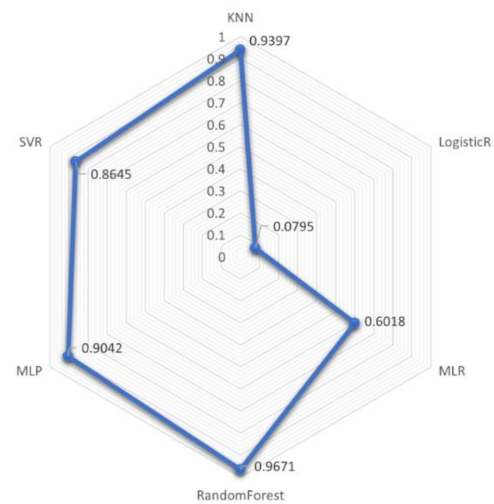

Fig. 6. R-squared score

2. MSE

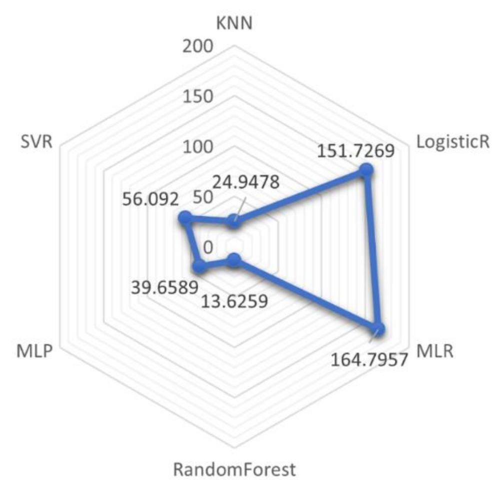

Fig. 7. Mean squared error

3. RMSE

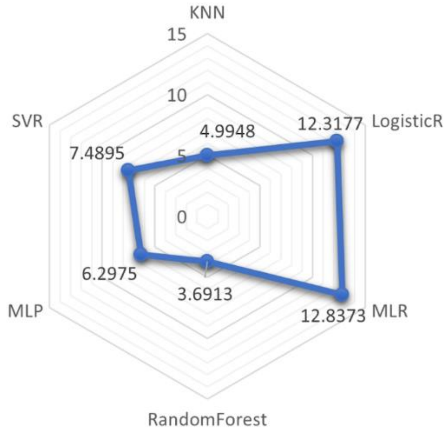

Fig. 7. Root mean squared error

V. Conclusion and Future Work

The indispensable part of our project is traffic congestion data. A machine learning model accuracy depends on its available data. We did not find any latest traffic dataset. If latest dataset can be found, the analysis and prediction accuracy can be better than this. In cloud part, we propose how to deploy machine learning model in cloud. In the future, a cloud-based program that can be utilized for a variety of purposes, including government projects, can be built.

References

[1] Saifuzzaman, Mohd, Nazmun Nessa Moon, and Fernaz Narin Nur. "IoT-based street lighting and traffic management system." 2017 IEEE region 10 humanitarian technology conference (R10-HTC). IEEE, 2017.

[2] Avatefipour, Omid, and Froogh Sadry. "Traffic management system using IoT technology-A comparative review." 2018 IEEE International Conference on Electro/Information Technology (EIT). IEEE, 2018.

[3] Devi, Suguna, and T. Neetha. "Machine Learning based traffic congestion prediction in a IoT based Smart City." Int. Res. J. Eng. Technol 4.5 (2017): 3442-3445.

[4] Misbahuddin, Syed, et al. "IoT based dynamic road traffic management for smart cities." 2015 12th International enabling/emerging technologies (HONET). IEEE, 2015.

[5] Mondal, Md Ashifuddin, and Zeenat Rehena. "Road Traffic Outlier Detection Technique based on Linear Regression." Procedia Computer Science 171 (2020): 2547-2555.

[6] Kamble, Shridevi Jeevan, and Manjunath R. Kounte. "Machine learning approach on traffic congestion monitoring system in internet of vehicles." Procedia Computer Science 171 (2020): 2235-2241.

[7] Çiğdem Çavdaroğlu "Methods & Algorithms for Traffic Density Measurement." KoçSistem (2016).

[8] Middleton, Neil, and Richard Schneeman. Heroku: up and running: effortless application deployment and scaling. " O'Reilly Media, Inc.", 2013.

[9] Wes McKinney. Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. " O'Reilly Media, Inc.", 2012.

[10] Luca Pappalardo, Filippo Simini, Gianni Barlacchi, and Roberto Pellungrini.scikit-mobility: a python library for the analysis, generation and risk assessment of mobility data, 2021.

[11] Islam, Mohammad Mushfiqul, et al. "Congestion Pricing of Urban Street Segments along with MRT Route—The Case Study of Dhaka, Bangladesh." Journal of Transportation Technologies 11.2 (2021): 168-178.

[12] Diaby, Tinankoria, and Babak Bashari Rad. "Cloud computing: a review of the concepts and deployment models." International Journal of Information Technology and Computer Science 9.6 (2017): 50-58.

[13] Ata, A., et al. "Modelling smart road traffic congestion control system using machine learning techniques." Neural Network World 29.2 (2019): 99-110.

[14] Othman, Muhammad Shalihin Bin, Sye Loong Keoh, and Gary Tan. "Efficient journey planning and congestion prediction through deep learning." 2017 International Smart Cities Conference (ISC2). IEEE, 2017.