Computing and Informatics, Vol. 41, 2022, 571–589, doi: 10.31577/cai_2022_2_571

ANALYZING TERRIFIC TRAFFIC IN URBAN AREAS: A SMALL STEP TOWARDS BRINGING ORDER INTO CITY ROADS

Sakhawat Hossain, Mosaddekul Islam, Ashikur Rahman

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology (BUET)
Dhaka-1000, Bangladesh
e-mail: {mdsakhawathossain05, mosaddek2020}@gmail.com,
ashikur@cse.buet.ac.bd

Abstract. Accurate travel time information enables travellers to plan their journey more wisely and efficiently. This in turn, lessens traffic congestion and improves people's travel experience, particularly in urban areas. Open-source traffic data available from different sources and Google Map API have raised opportunities for analyzing and predicting the traffic more accurately. The purpose of this research work is to analyze bus or car travel time data and showcase different insight and aspect of a society from its traffic pattern. Google Distance Matrix API, Python programming language and machine learning algorithms have been applied in this study to automatically extract, analyze, and visualize traffic data and showcase analysis methodology to improve people's travel experience in Dhaka City and the City of New York. In particular, we apply data analytics to develop an oracle that will give answers to different queries about traffic, such as least congested period and/or least congested route within a day/week/month etc., which in turn would enable people to make informed decisions for travel arrangements. The experimental results and detailed analyses show that there exists a wide fluctuation of travel time during the day in both cities. Furthermore, unlike other works, we accomplish various socio-cultural aspects and behaviour from traffic patterns in those two cities, perform the accessibility analysis and provide recommendations for further research.

Keywords: Traveling time prediction, crowdsourced data, Google distance matrix API, polynomial regression, scikit-learn

1 INTRODUCTION

Transportation in an urban area influences urban life to a great extent. Predicting travel time is an important area of research that can improve travel experiences in urban areas. In Intelligent Transportation Systems (ITS), the travel time prediction or traffic pattern prediction is a daunting task. It can be simplified using the regression technique because an approximation of the current travel time can be determined based on the previous travel time. However, collecting reliable data about traffic condition imposes an additional challenge. Many techniques exist for collecting travel time data such as global positioning system (GPS), cellular Geo-location, crowd-sourced travelling data, etc. Thus predicting travel time is an interesting blend of technology and machine learning. The technology part is responsible for accurately capturing and feeding the traffic data to a machine learning model that can analyze and predict the required travel time.

In this work, we use *polynomial regression* to derive machine learning models that can provide an estimation of the travel time based on past travel behaviours in Dhaka City and New York City. We have chosen these two cities as our testbeds because these are the heavily crowded zones in the world. To this extent, crowd sourced Google Distance Matrix API has been used for collecting traffic data in Dhaka City as we found no freely available traffic data for Dhaka City. On the other hand, publicly available taxi trip data from *NYC Taxi and Limousine Commission (TLC)* has been used for New York City. Machine Learning has been used to establish the initial framework as well as to extract, analyze, and visualize the data.

Experimental results show that our model can provide an aggregate view of the traffic with summary statistics based on previous observations. Given a particular "source-destination" pair, our derived models can also provide several useful results such as a graph of weekdays' relative travel time, expected travel time from a specific time in the day, least congested path within a day/week/month, least congested moment within a time interval etc. By using the built prediction models, it is also possible to infer social behavior from traffic patterns such as weekend behavior of the community, office hours, rush hours, holidays, night time travel habits etc. After successful implementation of the model it has been identified that the data provided by Google Distance Matrix API is highly reliable. The public authorities and transportation agencies may use this model to control traffic and effectively reduce the congestion.

The major contributions of this research work are summarized below:

- We utilize available traffic data sources and showcase analysis methodology to improve people's travel experience in Dhaka City and the City of New York.
- We devise methodologies to access those data and apply data analytics to develop a model. The built model can provide traffic forecast ahead of time, for example, "what will be the right time or date to visit a place within a given day/week/month?".

- Our built model can assist travellers/visitors to prepare a schedule considering the traffic so that they can visit a maximum (fixed) number of places within a given period.
- Our traffic model has the ability to assist government for city-wise traffic comparisons for making policies. Besides, it can assist government to take necessary measures to mitigate challenges related to traffic jams. For example: Suppose, the government wants to build:
  - A flyover or other infrastructure to reduce traffic jams, then what will be the appropriate place to utilize the investment to the maximum.
  - Office or other infrastructure, then what will be the appropriate place to build such premises so that the traffic density is equally sparse and it does not affect the surrounding traffic.
- We develop an oracle that will answer different queries about the traffic, such as:
  - Traffic pattern,
  - Least travel time,
  - Least congested path.
- Unlike other works we infer different social behavior from traffic patterns and compare two societies based on their people's travel habit such as:
  - Weekend community behaviour,
  - Rush/office hours,
  - Holidays,
  - Night time travel habits.

2 LITERATURE REVIEW

A number of research works have been conducted for travel time prediction [1]. Siripanpornchan et al. [2] describe the concept and application of Deep Belief Networks (DBN) to predict travel times. In their model they use DBN as a stack of Restricted Boltzmann Machines (RBM) to automatically learn generic traffic features in an unsupervised mode, and then a sigmoid regression has been used to predict travel time in a supervised mode.

In [3], the authors explore a deep learning model named the LSTM neural network model. They used real travel time dataset of 66 routes provided by Highways of England which was split into three parts: training set (80 %), test set (10 %) and the validation set (10 %). In this dataset, the travel times are 15-minute interval average travel times for each route. Gal et al. [4] investigate methods from Queueing Theory and Machine Learning in the prediction process. They implemented model based on segmentation of the travel time into stop-based segments using bus data of Dublin city. In [5], the researchers use Google Map's Distance Matrix API [6] to collect travel distances and time for an origin-destination (OD) pair at a specified departure time. In their analysis, there are number of OD pairs, where $n^2 - n$ is the number of transport analysis zones. Additionally, they extracted each OD pairs' travel time at thirty-minute intervals during the day (from 3:00 to 21:30), which resulted in slightly more than 1.6 million requests [7]. Chiu [8] also uses crowd-sourced data to present a model that is used for identification and evaluation of opportunities to apply Transportation Demand Management strategies.

Deri and Moura [9] present a spectral analysis of taxi movement based on the graph Fourier transform on "NYC Taxi Data Set", a historical repository of 750 million rides of taxi medallions over four years (2010-2013). They present the spectral decomposition of a large directed, sparse matrix. Important considerations toward handling this matrix are explained. Their Preliminary results show that their method can pinpoint locations of co-behavior for traffic in the Manhattan roads.

3 DATA COLLECTION AND PREPARATION

We have worked on two datasets: one is the New York City yellow taxi trip data freely available online, and the other is the travel time data of the Dhaka City which we collected from Google Distance Matrix API [6]. Both are discussed in details below in Section 3.1 and Section 3.2, respectively.

3.1 The TLC and Data

NYC Taxi Trip dataset is a publicly available dataset. *NYC Taxi and Limousine Commission* (TLC) has provided this public dataset which collects trips information including pickup and drop-off timestamp, pickup and drop-off locations, number of passengers travelled, payment method, distances of trips, and others. All this data are available at
`https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page`.

3.1.1 Data Description

The Yellow Taxi Trip Records contains the data regarding several taxi trips in New York City. In each trip record, one row represents a single trip made by a TLC-licensed yellow taxi that has different fields, such as:

Pickup Location ID: This field contains pickup location ID of a trip. It is the TLC Taxi Zone in which the taximeter was started.

Drop off Location ID: This field contains dropping location ID of a trip. It is the TLC Taxi Zone in which the taximeter was stopped.

Pickup Datetime: This field contains pickup date-time of a trip. It is the date and time when the meter was initiated.

Drop off Datetime: This field contains dropping date-time of a trip. It is the date and time when the meter was stopped.

Trip Distance: The elapsed trip distance in miles was kept in this field which was reported by the taximeter.

3.1.2 Data Extraction

First, data was extracted from NY yellow taxi travel dataset taking "Starting_locationID", "Dropping_locationID", "Pickup_datetime", "Dropoff_datetime" and "Distance" fields only. Three new fields – "Duration", "Weekday" and "Pickup_time_min" were added. "Duration" was calculated from pickup and drop-off date-time which is the difference between drop-off and pickup date-time. "Weekday" is the day of the pickup date-time. Only time from pickup date-time was considered to keep input range between 00:00:00 to 23:59:59. The time was mapped to numerical value for polynomial regression and stored in the "Pickup_time_min" field.

For example, suppose the pickup date-time is '2020-09-20 12:10:30' and drop-off date-time is '2020-09-20 12:30:00' So, 'Duration' = 19 minutes and "Pickup_time_min" $= 12 * 60 + 10 = 610$. 'Weekday' of the pickup date-time is Tuesday.

3.2 Travel Dataset for Dhaka City

We found no freely available traffic data for Dhaka City. First we built the dataset by collecting traffic data using *Google Distance Matrix API* [6]. For convenience Dhaka City is divided into different zones and each zone is assigned with a unique ID. A lookup table is used to map source and destination name to their corresponding IDs (Table 1).

Location	Location_ID	Area
Azimpur Bus Stand	1	Azimpur
Asad Gate	2	Mohammadpur
Mazar Road Bus Stand	3	Gabtoli
Uttara Sector 1	4	Uttara
Mohakhali Flyover	5	Mohakhali
Nikunja 1	6	Nikunja
Shahbagh Square	7	Shahbagh
...	...	...

We considered only some locations and their corresponding routes.

Table 1. Look up table for different zones

Google Distance Matrix API [6] is a service that returns travel distance and travel time for a $\langle origin, destination\rangle$ (OD) pair at a specified departure time. The information received is based on the recommended route between the origin and destination, as calculated by the *Google Distance Matrix API*. In our analysis, there were a number of OD (Origin-Destination) pairs. We extracted each OD pairs' travel time at thirty-minute intervals during the day (from 6:00 to 23:30), Every request was sent by an automated script, which constructed the *Uniform Resource Locator (URL)* by using the following parameters [6]:

- Longitude of origin (required);
- Latitude of origin (required);
- Longitude of destination (required);
- Latitude of destination (required);
- Mode of travel (optional);
- Date and time of departure (optional).

Results of the *Google Distance Matrix API* were provided as a *JavaScript Object Notation (JSON)* objects which include

- Duration,
- Duration_in_traffic.

3.2.1 Data Description

The travel time dataset collected using *Google Distance Matrix API* contains the data regarding expected travel time in Dhaka City. Each row in the dataset represents a single trip that has different fields such as:

Starting_point_id: The area mapped by an integer from which the trip started.

Dropping_point_id: The area mapped by an integer from which the trip ended.

Pickup_datetime: The date and time when the trip started.

Duration_in_traffic_value: The length of time it takes to travel this route in minutes stored as a numerical value. This time is calculated based on current and historical traffic conditions.

Weekday: The weekday on which the trip is made.

3.2.2 Data Extraction

Required data was extracted from the travel dataset of Dhaka City taking "Starting_locationID", "Dropping_locationID", "Pickup_datetime", "Dropoff_datetime", "Distance", "Duration" and "Weekday" fields only. Only numerical values were taken for distance and travel duration where 'Duration_in_traffic' were taken as duration. The new field "Pickup_time_min" was also added where the only time from pickup date-time was considered to keep input range between 00:00:00 to 23:59:59 and the times were mapped to numerical values.

4 TRAINING AND TUNING MODEL

After removing unexpected data, the dataset became ready and was split into three subsets:

- Training dataset,
- Validation dataset,
- Testing dataset.

The training dataset is used to train a machine learning model. A validation dataset has been used to validate the model and to overcome the overfitting and underfitting phenomenon. Finally, the testing dataset is used to test the final model.

Polynomial regression has been used to train the model. *PolynomialFeatures* class provided by *scikit-learn* is used for training purpose. Then, for every pair of locations, seven models for seven days of a week have been implemented to predict travel time for a given source, destination and pickup/drop-off date and time.

For each model, *Mean Square Errors* are calculated and compared for different degrees, and the degree for which the *Mean Square Error* is minimum has been selected. Figure 1 shows one such instance of a degree of polynomial selection for New York City for the model used for Saturday traffics. After tuning the model, *Intercept* and *Coefficients* for each model are saved in a *csv* file including starting and drop-off locations, weekday and degree of regression.

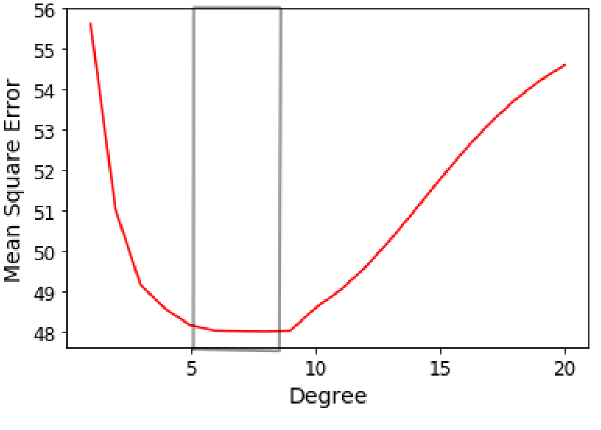


Figure 1. *Mean Square Errors* for a model corresponding to the location pair (*LaGuardia Airport, Midtown East*) on Saturday shown for different degrees of polynomial regression. Here, MSE is lower for the degrees of polynomial ranging from 6 to 9, thus any degree from this range could be selected. In our model, we have chosen 8 as the degree of regression which generates minimal MSE.

5 ANALYSIS

Once we fit the data into a regression model, we can proceed with the data analysis. The following sections provide our results, namely:

1. A graph of weekdays relative travel times,
2. A matrix of mean weekday travel times,
3. Expected travel time from a specific time, and
4. Seven models for seven days of a week.

5.1 Right Time to Travel

Once built, our model is ready to answer different questions such as "What will be the right time or date to visit a place within a given day/week/month"? A pickup time with a minimum required travel time will be recommended. This pickup time can be a time within a day or a week or even it can be a time interval in a day. Different time intervals, as shown in Table 2, have been defined for generating outputs. Four intervals within the day time are considered as during night time the travel time is always minimum.

Portion of the Day	Time Range
Morning	6:00–10:00
Midday	10:01–14:00
Afternoon	14:01–18:00
Evening	18:01–22:00
Night	22:01–rest

Table 2. Time intervals of a day

5.1.1 New York City

We select a pair of locations (*LaGuardia Airport*, *Midtown East*) of New York City to analyze different queries. Some of the possible queries and the corresponding results are shown below.

a) **Day-wise query:** Suppose, the user wants to know, for a given day, what is the minimum travel time and corresponding departure time within different segments of the day? The answer from the model that includes expected departure time and required travel time from *LaGuardia Airport* to *Midtown East* on Saturday is shown in Table 3.

b) **Minimum travel time and departure time for seven days of a week:** Suppose, the user wants to know what is the minimum travel time and corresponding departure time for all seven days of a week so that the user can plan

Portions of a Day	Departure Time	Travel Time (minutes)
Morning	06:01	18
Midday	10:01	23
Afternoon	17:41	25
Evening	21:41	20

Table 3. Expected departure time and travel time

ahead. The output generated by the model that includes expected departure time and required travel time from *LaGuardia Airport* to *Midtown East* for seven days is shown in Table 4 as a sample.

Days of Week	Departure Time	Travel Time (minutes)	Part of Day
Sunday	07:01	16	Morning
Monday	18:41	26	Evening
Tuesday	18:41	27	Evening
Wednesday	18:41	29	Evening
Thursday	12:21	33	Midday
Friday	11:41	31	Midday
Saturday	07:01	19	Morning

Table 4. Expected departure time and travel time for seven days

c) **Within given range, what is the minimum travel time and departure time for seven days of week:** For 08:00 to 14:00, the expected departure time and required travel time from *LaGuardia Airport* to *Midtown East* for seven days, again generated by the built model, is shown in Table 5.

Days of Week	Departure Time	Travel Time (minutes)
Sunday	08:00	18
Monday	14:00	29
Tuesday	14:00	31
Wednesday	13:00	31
Thursday	12:20	33
Friday	11:40	31
Saturday	08:00	20

Table 5. Expected departure time and travel time for seven days (8:00–14:00)

5.1.2 Dhaka City

Similar queries for Dhaka city can be performed using the built models which are described below.

For demonstration purpose, we select *Azimpur Bus Stand* as starting point and *Uttara Sector 1* as destination point because these two locations are two extreme opposite corners of the Dhaka city, *Azimpur Bus Stand* being on the extreme south side and *Uttara Sector 1* being on the extreme north side. Thus, anyone travelling from *Azimpur Bus Stand* to *Uttara Sector 1* needs to bisect the city by using a backbone road and overcome extreme traffic conditions of the city.

a) The expected departure time and required travel time for this ⟨*Azimpur*, *Uttara*⟩ source-destination pair for the seven days of a week are shown in Table 6.

Days of Week	Departure Time	Travel Time (minutes)
Sunday	13:01	43
Monday	13:01	43
Tuesday	13:41	44
Wednesday	8:01	47
Thursday	12:21	45
Friday	8:01	35
Saturday	8:01	37

Table 6. Expected departure time and travel time for seven days (*Azimpur Bus Stand* to *Uttara Sector 1*)

b) For a pair of source-destination, minimum travel time and departure time during the different segments of a given day can be queried from the model. The expected departure time and required travel time from *Azimpur Bus Stand* to *Uttara Sector 1* on Saturday are shown in Table 7.

Parts of Day	Departure Time	Travel Time (minutes)
Morning	07:01	40
Midday	13:01	46
Afternoon	14:01	46
Evening	19:41	54

Table 7. Expected departure time and travel time for *Azimpur Bus Stand* to *Uttara Sector 1*

c) Within a given range, minimum travel time and departure time for which travel time is minimum can be shown for seven days of the week. For 08:00 to 14:00, expected departure time and required travel time from *Azimpur Bus Stand* to *Uttara Sector 1* for seven days are shown in Table 8.

5.2 Least Congested Route

We define the route as having the least travel time among all the routes from a source to a destination for a specific time as Least Congested Route.

Days of Week	Departure Time	Travel Time (minutes)	Part of the Day
Sunday	13:01	43	Midday
Monday	14:00	29	Midday
Tuesday	14:00	31	Midday
Wednesday	13:00	31	Morning
Thursday	12:20	33	Midday
Friday	11:40	31	Morning
Saturday	08:00	20	Morning

Table 8. Expected departure time and travel time from *Azimpur Bus Stand* to *Uttara Sector 1* (8:00–14:00)

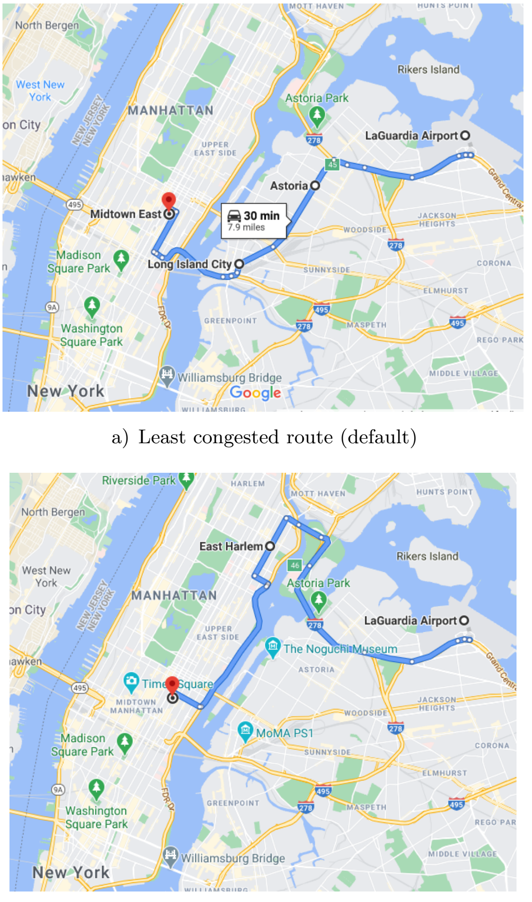
5.2.1 New York City

1. When a preferred travel time and a source-destination pair are given, the model shows the route which requires minimal travel time among all the routes from the same source to the same destination. A least congested route from LaGuardia Airport to Midtown East on Wednesday, November 4, 2020, at 10:00 AM generated by the model is shown in Figure 2 a). In this route, the model predicts the 33 minutes required to complete the journey.
2. When source-destination pair with an intermediate location is given, for a given time, the model shows a route among all the routes from source to destination that goes through the given intermediate location for which the expected travel time is minimum. A least congested path from *LaGuardia Airport* to *Midtown East* on Wednesday, November 4, 2020, at 10:00 AM that goes through *East Harlem* is shown in Figure 2 b). The model predicts 44 minutes to complete the journey.

5.2.2 Dhaka City

Similar queries for Dhaka city can be performed using the built models which are described below.

1. When a preferred travel time and a source-destination pair are given, the model shows the route which requires minimal travel time among all the routes from the same source to the same destination. A least congested route from *Azimpur Bus Stand* to *Uttara Sector 1* on Saturday, February, 2021 at 8:30 PM generated by the model is shown in Figure 3 a). In this route, the model predicts 50 minutes required to complete the journey which is close to the prediction by Google Map (shown Figure 3 a).
2. When a source-destination pair with an intermediate location, in this case *Mazar Road*, is given, the model shows a route among all the routes from source to destination going through that intermediate location for which the expected travel time is minimum. A least congested route from *Azimpur Bus Stand* to



a) Least congested route (default)

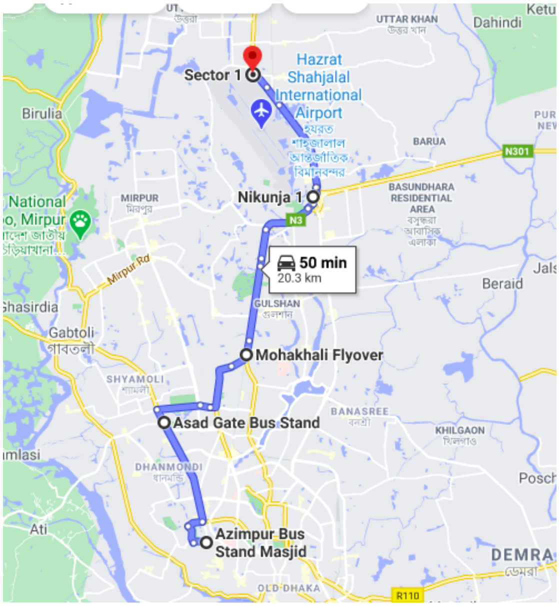
b) Least congested route with a given mid location

Figure 2. Least Congested Route

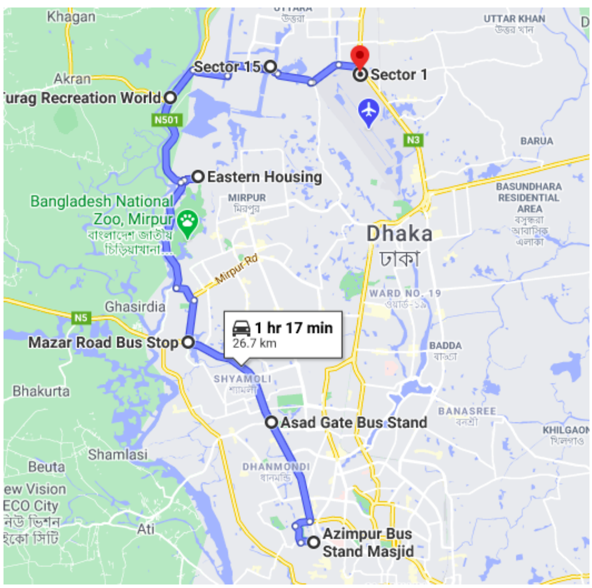
Uttara Sector 1 on Saturday, February, 2021 at 8:30 PM going through *Mazar Road* is shown in Figure 3 b). The model predicts 65 minutes to complete the journey which is very close to the prediction generated by the Google Map in Figure 3 b).

5.3 Least Time to Travel

By *least time*, we mean the pickup time for which travel time is minimum. It may be within a given range of time or default range of time (like 6:00 to 22:00).



a) Least congested route (default)

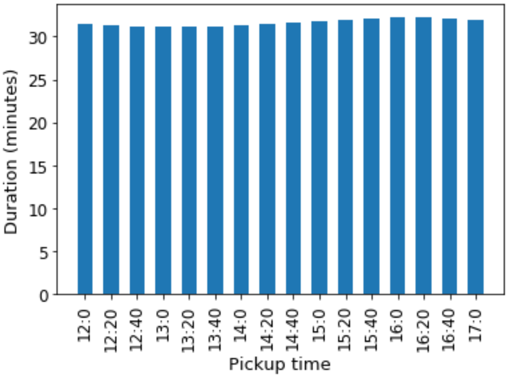


b) Least congested route with a given mid point as *Mazar Road*

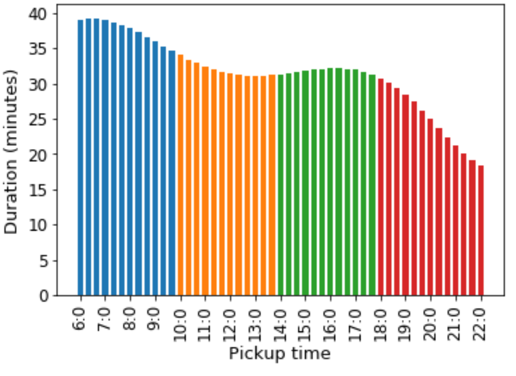
Figure 3. Least Congested Route For *Azimpur Bus Stand* to *Uttara Sector 1*

5.3.1 New York City

The required travel times from *LaGuardia Airport* to *Midtown East* at different pickup times are shown in Figure 4. Travel time is minimum during the night 16:00–22:00 (shown in red in Figure 4 b)).



a) Travel time within 12:00 to 17:00

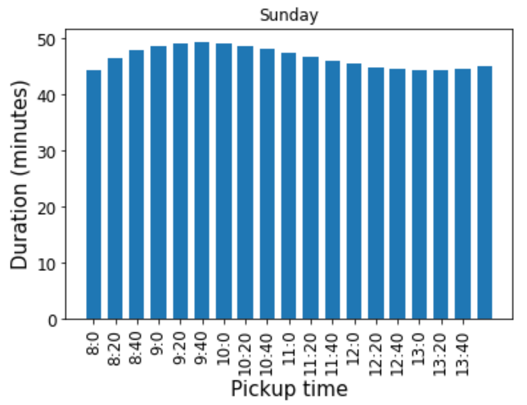


b) Travel time within 6:00 to 22:00 (Default)

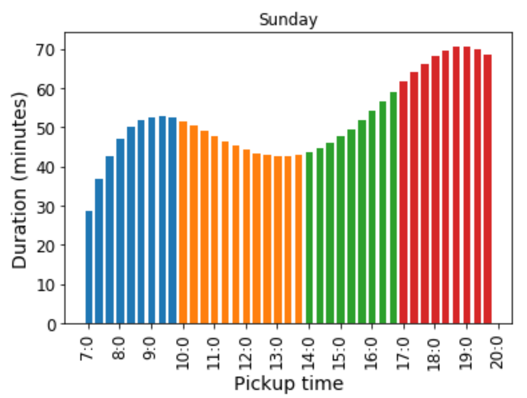
Figure 4. Travel time comparison within a range of period for New York City

5.3.2 Dhaka City

The required travel times from *Azimpur Bus Stand* to *Uttara Sector 1* at different pickup times are shown in Figure 5. Travel time is minimum during the early morning (shown in blue in Figure 5 b)).



a) Travel time within 8:00 to 14:00



b) Travel time within 7:00 to 20:00 (Default)

Figure 5. Travel time comparison within a range of period for Dhaka City

5.4 Socio-Cultural Aspects

Interestingly, different socio-cultural behaviours of a society such as weekend behaviour, rush hours, nighttime travel habits etc. can be learned from its traffic pattern. Different social behaviours are described in this section for New York City and Dhaka City, respectively.

Traffic Pattern: As we can see in Figure 6 the traffic pattern is different from city to city. In Figure 6 a) we show the traffic pattern of New York City while in Figure 6 b) we show the traffic pattern of Dhaka city. Observe that there are two peak-hours in Dhaka's traffic pattern as well as in the New York City's traffic pattern.

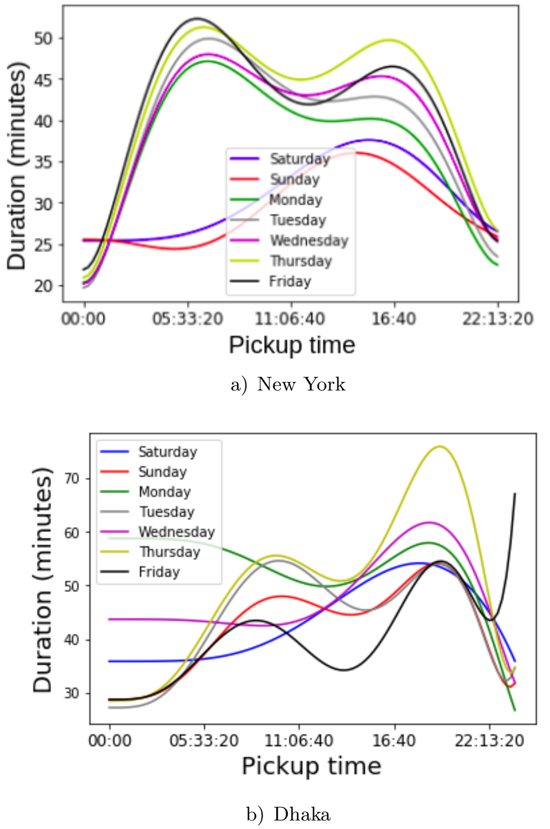
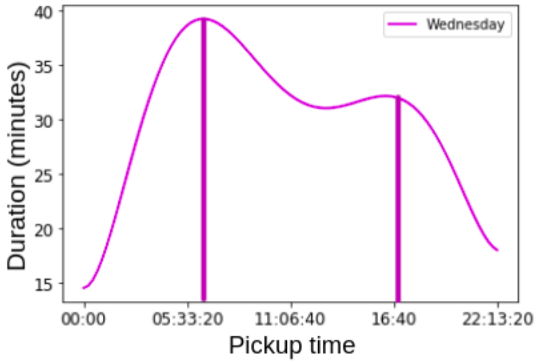


Figure 6. Traffic pattern on different weekdays

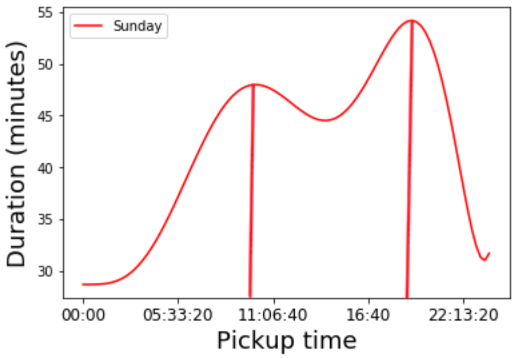
Rush Hours: As observed in Figure 7 during the morning and the late evening the traffic density increases. This is as expected because during these two times offices open and close respectively and trigger heavy traffic in the city.

Weekend: We plotted a graph of weekdays relative travel times in Figure 6. We can compare the average traffic density between the different days of the week. As we can see in Figure 6 a) the traffic density is relatively less on Saturday and Sunday as expected because these two days are weekend in New York City. On the other hand, one can easily see that on Friday traffic density is less in Dhaka city as it is the weekend for this part of the world.

Night Time Travel Habit: One can also observe the nighttime travel habit analysing Figure 6. We found one common pattern between the two figures. The traffic increases at the night right before the weekend. For example, in the case of New York City, it is Friday night while it is Thursday night for Dhaka city.



a) New York



b) Dhaka

Figure 7. Traffic pattern of a day

6 CONCLUSIONS AND FUTURE WORKS

In this work, we showcase a machine learning technique to model road traffics of two populous cities of the world, one chosen from a developed country (the New York City) and the other chosen from a developing country (the Dhaka city). We provide some interesting insights including an impression of their socio-cultural aspects derived from road traffic patterns. In future, we plan to perform some rigorous analysis to assist the government in long term planning and designing flyovers, bridges and roads to improve their experience on road networks.

REFERENCES

[1] Agrawal, V.—Singh, J. N.—Negi, A.: Travel-Time Prediction: A Short Survey. Recent Trends in Parallel Computing, Vol. 4, 2017, No. 3, pp. 13–19.

[2] Siripanpornchana, C.—Panichpapiboon, S.—Chaovalit, P.: Travel-Time Prediction with Deep Learning. 2016 IEEE Region 10 Conference (TENCON), 2016, pp. 1859–1862, doi: 10.1109/TENCON.2016.7848343.

[3] Duan, Y.—Lv, Y.—Wang, F. Y.: Travel Time Prediction with LSTM Neural Network. 2016 IEEE 19th Internation Conference on Intelligent Transportation Systems (ITSC), 2016, pp. 1053–1058, doi: 10.1109/ITSC.2016.7795686.

[4] Gal, A.—Mandelbaum, A.—Schnitzler, F.—Senderovich, A.—Weidlich, M.: Traveling Time Prediction in Scheduled Transportation with Journey Segments. Information Systems, Vol. 64, 2017, pp. 266–280, doi: 10.1016/j.is.2015.12.001.

[5] Dumbliauskas, V.—Grigonis, V.—Barauskas, A.: Application of Google-Based Data for Travel Time Analysis: Kaunas City Case Study. Promet – Traffic and Transportation, Vol. 29, 2017, No. 6, pp. 613–621, doi: 10.7307/ptt.v29i6.2369.

[6] Google: Google Distance Matrix API. 2020, `https://developers.google.com/maps/documentation/distance-matrix/intro` [accessed 10/03/2020].

[7] Kumarage, S. P.: Use of Crowdsourced Travel Time Data in Traffic Engineering Applications. Master Thesis. University of Moratuwa, Sri Lanka, 2018.

[8] Chiu, Y. H.: Application of Crowdsourced Travel Data in Identifying Potential Opportunities for Transportation Demand Management. Master Thesis. University of Waterloo, ON, Canada, 2017.

[9] Deri, J. A.—Moura, J. M. F.: Taxi Data in New York City: A Network Perspective. 2015 49th Asilomar Conference on Signals, Systems and Computers, IEEE, 2015, pp. 1829–1833, doi: 10.1109/ACSSC.2015.7421468.

Sakhawat Hossain completed his B.Sc. degree in computer science and engineering at the Department of CSE, Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh in 2021. Currently, he is studying for M.Sc. in the same department. His research interests include machine learning and artificial intelligence, concurrency and distributed computing, cloud computing, cybersecurity, data mining and big data.

Mosaddekul Islam completed his B.Sc. degree in computer science and engineering at the Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh in 2021. His research interests include artificial intelligence, machine learning, deep learning, networking, cyber security, data mining, big data and 4IR. Now he is working in a company that works with technologies related to 4IR and service automation.

Ashikur Rahman completed his B.Sc. and M.Sc. degrees in computer science and engineering at the Department of CSE, Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh in 1998 and 2001, respectively. He earned his Ph.D. in 2006 from the Department of Computing Science, University of Alberta, Canada. After finishing his Ph.D., he worked as a postdoctoral researcher at Simon Fraser University (2007), University of Calgary (2010-2011), McGill University (2011), Canada, and at Binghamton University (2012), USA. His research interests include cyber-physical systems, wireless ad-hoc and sensor networks, machine learning, swarm intelligence, data center networks, and neural networks. At present, he is working as Professor in the Department of CSE, BUET.