Journal of Engineering Sciences
Vol 11, Issue 4, APRIL /2020
ISSN NO:0377-9254

PREDICTING THE DEMANDS OF CITYWIDE PASSENGERS USING MULTI-LEVEL ATTENTION NETWORKS

Pratyush Thota [1], K. Nehith [2], A. Ram Swaroop Reddy [3], Y.V.V.S Sai Ram [4],

[1,2,3,4] *Department of Computer Science And Engineering, GITAM University, Hyderabad, India.*

***Abstract*— For the emerging mobility-on-demand services, it is of great significance for predicting passenger demands based on historical mobility trips to achieve better vehicle distribution. Prior works have focused on predicting next-step passenger demands at selected locations or hot spots. However, citywide passenger demands encapsulate both time-varying demand trends and global statuses, and hence are more beneficial to avoiding demand-service mismatching and developing effective vehicle distribution/scheduling strategies. We find that adaptations of single-step methods are unable to achieve robust prediction with high accuracy for further steps. In this project, we propose an end-to-end deep neural network model to the prediction task. We introduce a multi-level attention model (global attention and temporal attention) to emphasize the effects of latent citywide mobility regularities and capture relevant temporal dependencies. We evaluated our proposed method using real-world mobility trips (taxis and bikes) and the experimental results show that our method achieves higher prediction accuracy than the state-of-the-art approaches.**

***Keywords*—Demands,Passengers, Multi-Level Attention Network, Encoder, Decoder.**

1. INTRODUCTION

In order to meet various passenger mobility demands, there has been a proliferation of mobility-on-demand (MOD) services such as Uber, Didi and CitiBike. These kinds of services enable the evolution of passenger mobility demands towards more flexible and public vehicles, e.g taxis, shared use cars and bicycles. It is vital that the mobility-on-demand systems can be further developed to predict citywide passenger mobility demands (i.e., pickup and drop off frequencies) with high accuracy, which assists greatly in customizing effective vehicle distribution and scheduling strategies to achieve demand resource balance.

However, we argue that multi-step citywide passenger demand prediction (as illustrated in Figure 1) is considerably more beneficial to nowadays MOD services due to the following two reasons. First, multi-user passenger demands indicate the varying demand trend, which is useful to avoid impulsive vehicle scheduling responses in the presence of temporary demand fluctuation. In contrast, short-term passenger demands prediction results are typically shortsighted and more likely to cause unnecessary vehicle scheduling back and forth. Second, the large volume of vehicles are widespread in the whole city. Citywide passenger demands are expected to encapsulate global statuses and hence are more informative in terms of achieving better vehicle distribution.

2. LITERATURE SURVEY

A. Predicting Taxi–Passenger Demand Using Streaming Data.

: Luis Moreira-Matias, João Gama, Michel Ferreira, João Mendes-Moreira, Luis Damas This paper introduces a novel methodology for predicting the spatial distribution of taxi–passengers for a short-term time horizon using streaming data. First, the information was aggregated into a histogram time series. Then, three time-series forecasting techniques were combined to originate a prediction. Experimental tests were conducted using the online data that are transmitted by 441 vehicles of a fleet running in the city of Porto, Portugal. The results demonstrated that the proposed framework can provide effective insight into the spatio temporal distribution of taxi–passenger demand for a 30-min horizon.

B. Context-aware taxi demand hot spots prediction:
Han-wen Chang, Yu-chin Tai and Jane Yung-jen Hsu This paper proposes mining historical data to predict demand distributions with respect to contexts of time, weather, and taxi location. The four-step process consists of data filtering, clustering, semantic annotation, and hotness calculation. The results of three clustering algorithms are compared and demonstrated in a web mash-up application to show that context-aware demand prediction can help improve the management of taxi fleets.

C. Dynamic Cluster-Based Over-Demand Predictionin Bike Sharing Systems

Longbiao Chen, Daqing Zhang, Leye Wang, Dingqi Yang, Xiaojuan Ma, ShijiaLi, Zhaohui Wu, Gang Pan, Thi-Mai-Trang Nguyen, Jérémie Jakubowicz

This paper proposes a dynamic cluster-based framework for over-demand prediction. Depending on the context, a weighted correlation network is constructed to model the relationship among bike stations, and dynamically group neighboring stations with similar bike usage patterns into clusters. Then Monte Carlo simulation is adopted to predict the over-demand probability of each cluster. Evaluation results using real-world data from New York City and Washington, D.C. shows that the framework proposed accurately predicts over-demand clusters and outperforms the baseline methods significantly.

3. EXISTING SYSTEM

Existing methods for passenger demands prediction have been focused on anticipating next-step passenger demands for a particular set of locations such as taxi stands, bike stations and hot spots. However, we argue that multi-step citywide passenger demands prediction is considerably more beneficial to nowadays MOD services due to the following two reasons. First, multi-step passenger demands indicate the varying demand trend, which is useful to avoid impulsive vehicle scheduling responses in the presence of temporary demand fluctuation. In contrast, short-term passenger demands prediction results are typically shortsighted and more likely to cause unnecessary vehicle scheduling back and forth. Second, the large volume of vehicles are widespread in the whole city. Citywide passenger demands are expected to encapsulate global statuses and hence are more informative in terms of achieving better vehicles distribution. Limitations of proposed is Intuitively, partial passenger demands in sub-areas are insufficient to generate an effective global solution to citywide vehicle distribution or scheduling.

4. PROPOSED SYSTEM

In this, the problem of predicting multistep citywide passenger mobility demands (i.e., pickup and drop-off) accurately. The key technical challenge of this problem is to deal with (1) complex spatio temporal influences on passenger demands, and (2) interactions between pickups and drop-offs. Specifically, passenger demands in a region are typically correlated with the demands in its surrounding region and the demands in a region could be affected by its statuses in previous time intervals. Advantages of the proposed system is To tackle these aforementioned challenges, inspired by the success of deep neural networks to model complex spatio temporal features, propose an end-to end deep neural network solution to the multi-step citywide passenger demand prediction problem.

5. METHODOLOGY

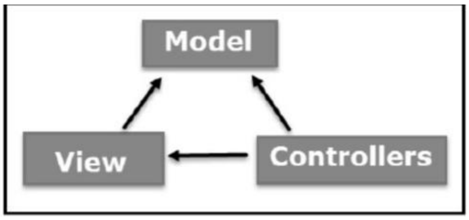


Fig 5.1 : MVC Framework Components

MODEL:

The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data. For example, a Customer object will retrieve the customer information from the database, manipulate it and update it data back to the database or use it to render data.

VIEW:

The View component is used for all the UI logic of the application. For example, the Customer view will include all the UI components such as text boxes, drop downs, etc. that the final user interacts with.

CONTROLLER:

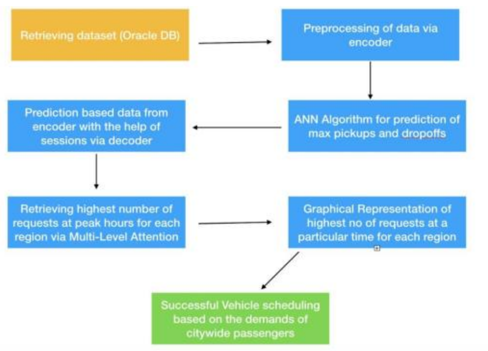
Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output. For example, the Customer controller will handle all the interactions and inputs from the Customer View and update the database using the Customer Model. The same controller will be used to view the Customer data

5.1. MODEL-VIEW-CONTROLLER ARCHITECTURE

The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application. MVC is one of the most frequently used industry-standard web development framework to create scalable and extensible projects.

5.2. ALGORITHM

In this project we implement the Artificial Neural Network Algorithm since we have linear data. The data sets (real world mobility trips: taxis and bikes) we utilize for our project are stored in Oracle DB. Since this data is linear data, we implement the ANN Algorithm to predict the total count of requests (i.e pickups and drop offs) for each region during peak hours.



5.3. WORK FLOW DESCRIPTION

1. First the data sets we utilize are stored in oracle DB.
2. Through Encoder data pre-processing is done using sql constants.

3. In the Decoder Phase, prediction is done with through ANN algorithms and with the help of sessions to predict the max no. of pickups and max no. of drop offs.
4. In the last Multi-Level Attention Phase, the total no of requests (pickups and drop offs) for each region at peak hours is retrieved.
5. Finally, a user-friendly graph is displayed to demonstrate all the regions with the most no of pickups and drop offs during peak hours.
6. Based on the graph, the user can now schedule vehicle pickups and drop offs avoiding unnecessary vehicle scheduling based on the demands of city wide passengers.

5.4. SOFTWARE REQUIREMENTS

Operating System: Windows 7

User Interface : HTML, CSS

Client-side Scripting: JavaScript

Programming Language : Java

Web Applications: JDBC, Servlets, JSP

IDE/Workbench: My Eclipse 8.6

Database: Oracle 11g

Server Deployment: Tomcat 7.0

5.5. HARDWARE REQUIREMENTS

Processor :Intel core i3 or above

Hard Disk :500GB or more

5.6 FUNCTIONALITY

A data access object (DAO) is a pattern that provides an abstract interface to some type of database. This is done by mapping application calls to the database; the DAO provides some specific data operations without exposing details of the database. This isolation supports the single responsibility principle. The UserDao.java separates what data access the application needs, in terms of domain-specific objects and data types (the public interface of the DAO), from how these needs can be satisfied with an oracle database.

5.7 IMPLEMENTATION

Encoder: In this phase, data retrieved from our data sets undergoes data pre-processing. In the DOA module of our project, we create an object which is used to retrieve the pickups data and drop offs data from the database (Oracle DB). We implement a sqlconstant.java (properties page) that enables data pre-processing.

Decoder: In this phase, prediction of the maximum no of pickups and maximum no of drop offs is done via ANN algorithm and with the help of sessions using servlets and JSP Technology.

Multi-Level Attention: In this phase, after the implementation of the encoder-decoder framework, we then retrieve the total count of request i.e pickups and drop offs for each region during peak hours. Then, a graph is plotted representing the data of requests. Finally, user can perform vehicle scheduling efficiently based on the citywide passenger demands.

6. RESULTS

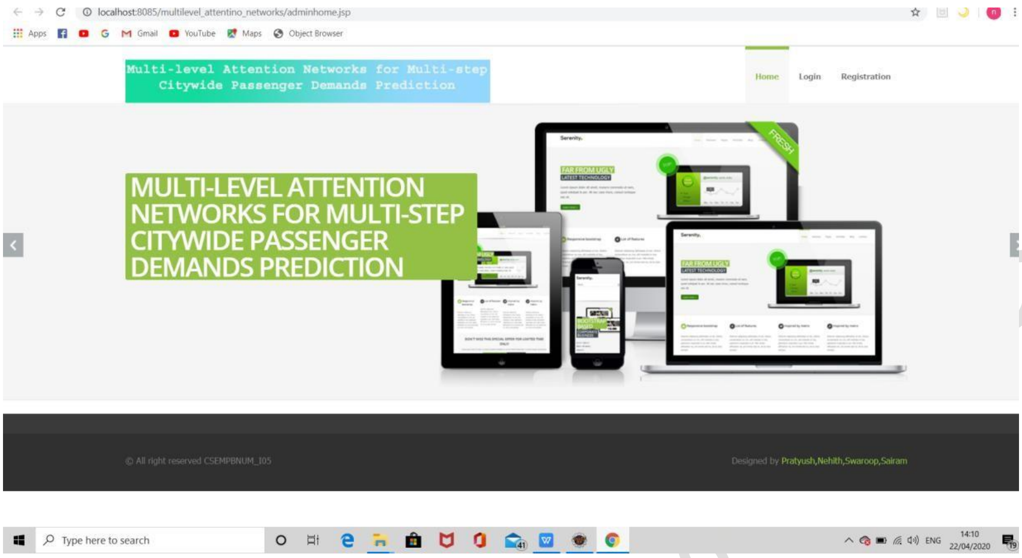


Fig 6.1 : UI of the Web Application. Consists of Home, Login, and Registration.

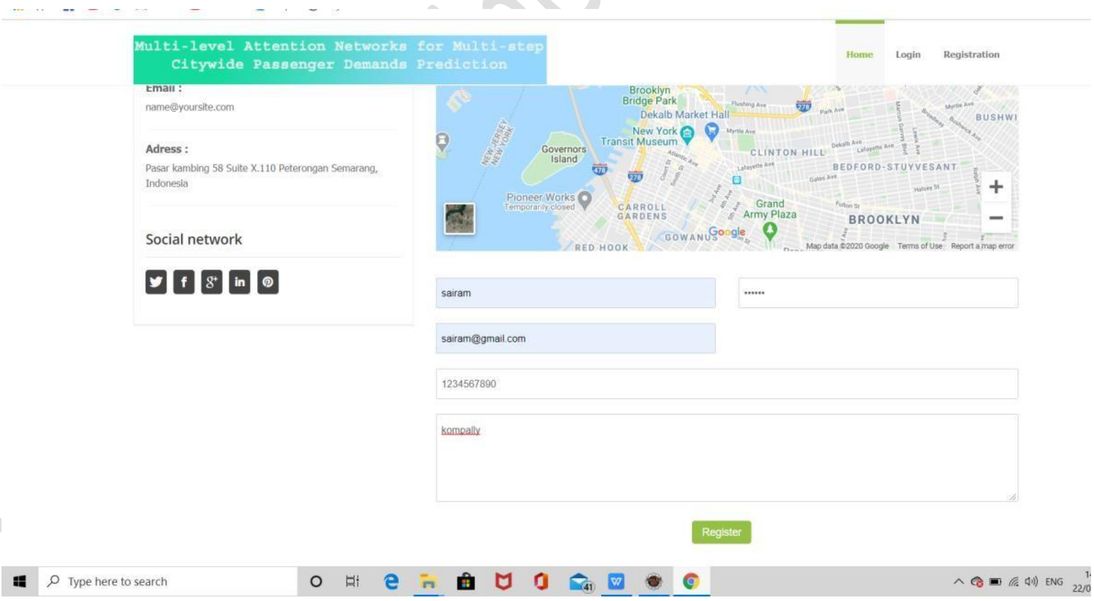
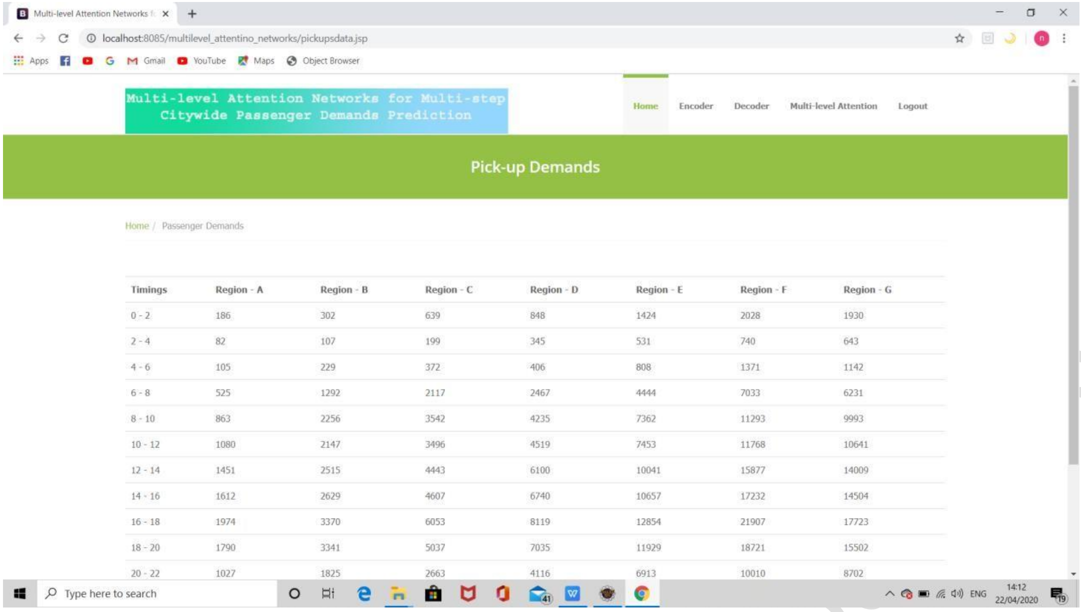


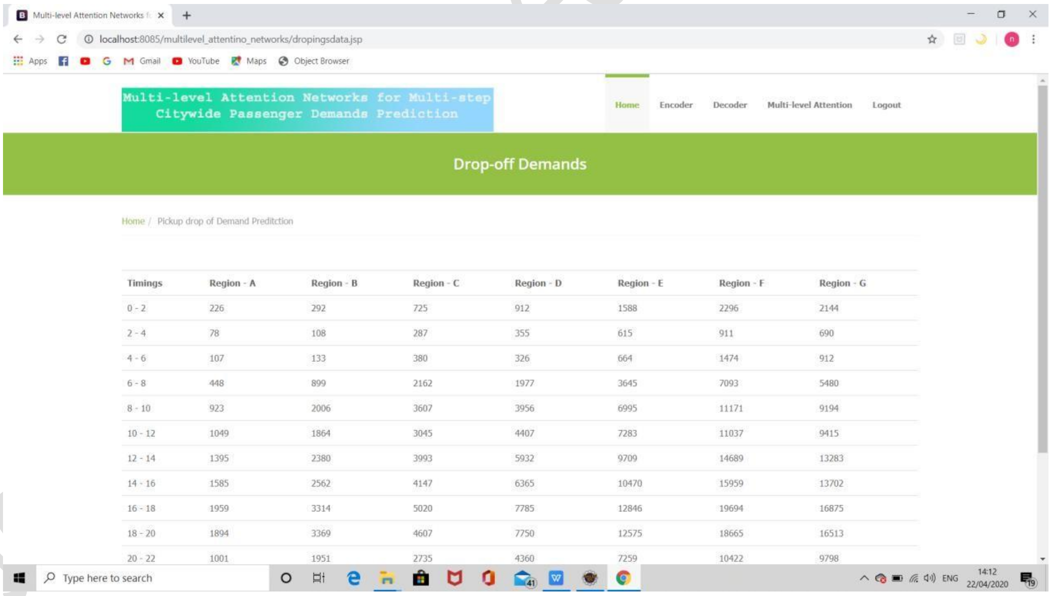
Fig 6.2 : One-time Registration of user when clicks on the registration tab.

JES
Journal of Engineering Sciences
Vol 11, Issue 4, APRIL /2020
ISSN NO:0377-9254



Timings	Region - A	Region - B	Region - C	Region - D	Region - E	Region - F	Region - G
0 - 2	186	302	639	848	1424	2028	1930
2 - 4	82	107	199	345	531	740	643
4 - 6	105	229	372	406	808	1371	1142
6 - 8	525	1292	2117	2467	4444	7033	6231
8 - 10	863	2256	3542	4235	7362	11293	9993
10 - 12	1080	2147	3496	4519	7453	11768	10641
12 - 14	1451	2515	4443	6100	10041	15877	14009
14 - 16	1612	2629	4607	6740	10657	17232	14504
16 - 18	1974	3370	6053	8119	12854	21907	17723
18 - 20	1790	3341	5037	7035	11929	18721	15502
20 - 22	1027	1825	2663	4116	6913	10010	8702

Fig .6.3 : Pick-up Demands of passengers.



Timings	Region - A	Region - B	Region - C	Region - D	Region - E	Region - F	Region - G
0 - 2	226	292	725	912	1588	2296	2144
2 - 4	78	108	287	355	615	911	690
4 - 6	107	133	380	326	664	1474	912
6 - 8	448	899	2162	1977	3645	7093	5480
8 - 10	923	2006	3607	3956	6995	11171	9194
10 - 12	1049	1864	3045	4407	7283	11037	9415
12 - 14	1395	2380	3993	5932	9709	14689	13283
14 - 16	1585	2562	4147	6365	10470	15959	13702
16 - 18	1959	3314	5020	7785	12846	19694	16875
18 - 20	1894	3369	4607	7750	12575	18665	16513
20 - 22	1001	1951	2735	4360	7259	10422	9798

Fig 6.4: Drop-off Demands of passengers.

JES
Journal of Engineering Sciences
Vol 11, Issue 4, APRIL /2020
ISSN NO:0377-9254

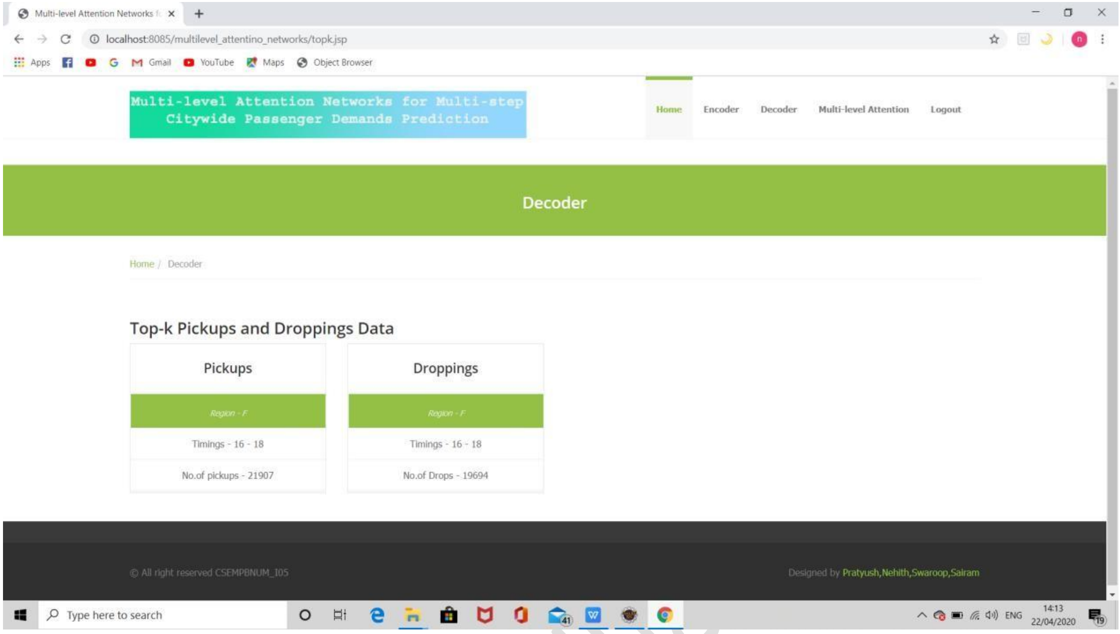


Fig 6.5 : Decoder. It Displays the highest no of pickups and drop-offs at a particular region i.e region F after prediction.

(Navigation: Home-> Decoder)

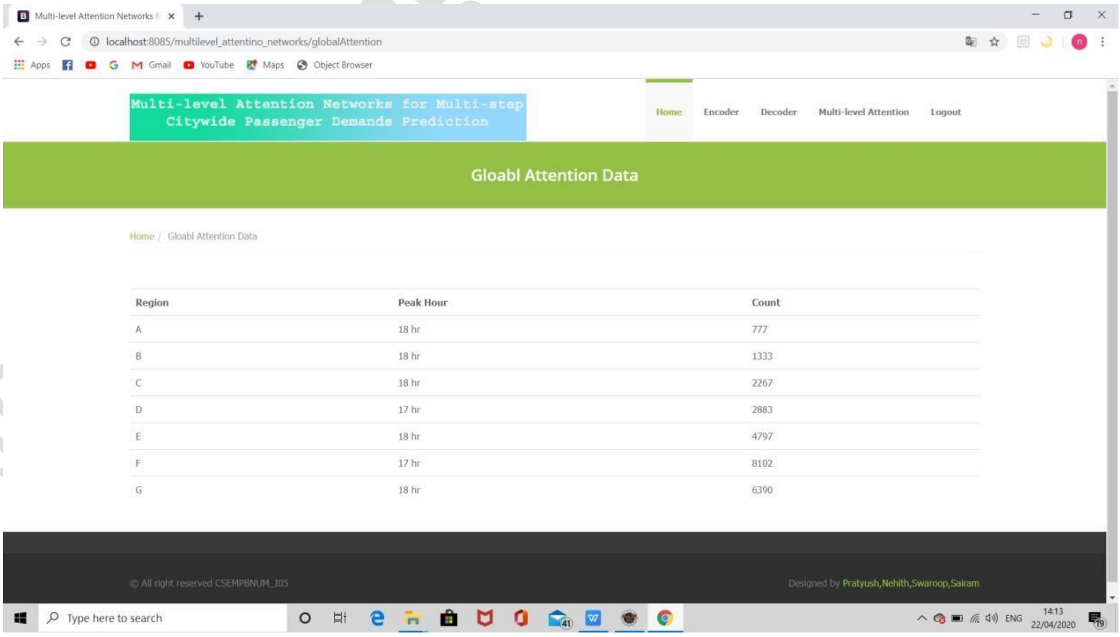


Fig 6.6 : Global Attention Data (Multilevel attention) displaying region wise data including peak hours of traffic, and no of of pickups and drop offs during those peak hours.

Journal of Engineering Sciences
Vol 11, Issue 4, APRIL /2020
ISSN NO:0377-9254

(Navigation: Home-> Multi-Level Attention -> Global Attention)

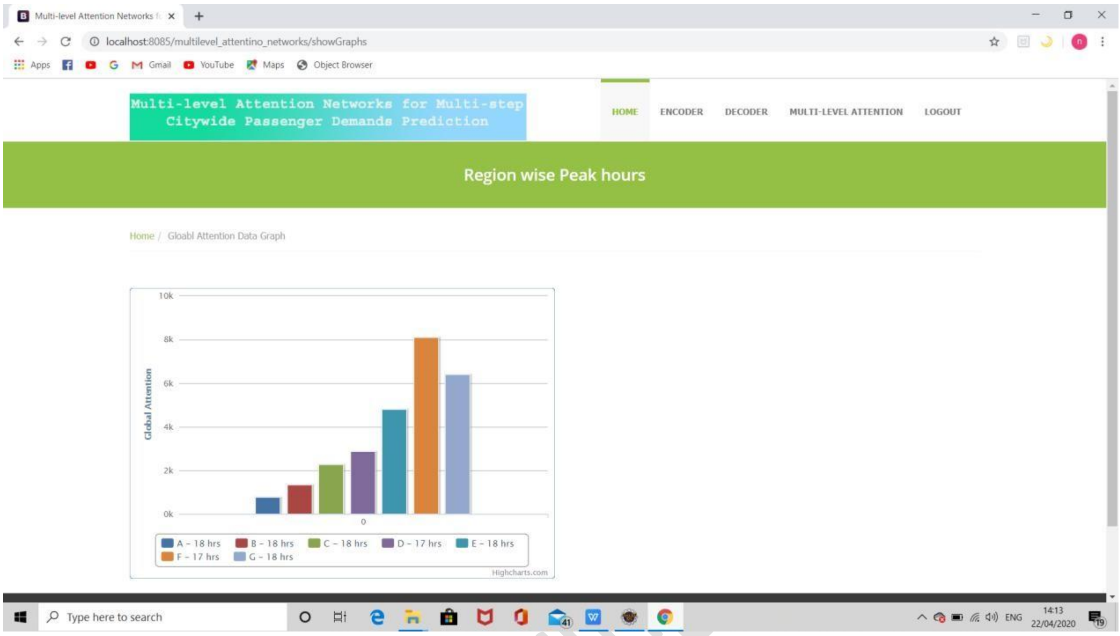


Fig.6.7: Graphical Representation of Region Wise Peak Hours after Prediction.

Analysis: So based on the Global Attention Data and Graphical representation of region wise peak hours, the user can now perform vehicle scheduling as he now has the data which tells him which regions have the highest number of pickups and drop-offs at a specific time (peak hours). So, we can further predict that for the upcoming consecutive days in the future, the respective regions will have high number of pick-ups and drop-offs at their respective peak hours. So, the user can schedule vehicles to those regions, thereby reducing impulsive vehicle scheduling and unnecessary vehicle scheduling back and forth.

7. CONCLUSION

In this paper, we study the problem of predicting multi-step citywide passenger demands. The main technical challenge is to enforce the complex spatio temporal influences, pickup drop off interactions and latent citywide demand regularities into future prediction. We solve the problem by introducing a multi- level attention-based encoder-decoder deep learning approach. We learn global attention and temporal attention to emphasize the effects of representative citywide demand patterns and capture temporal dependencies on each-step prediction during the decoding phase. Our experiments on two real-world data sets (taxi and bike mobility trips) demonstrate that our proposed approach outperforms the state-of-the-art prediction approaches and the multi- level attention model can effectively improve the accuracy for multi- step prediction.

REFERENCES

[1]Luis Moreira-Matias, João Gama, Michel Ferreira, João Mendes-Moreira, and Luis Damas, “Predicting Taxi–Passenger Demand Using Streaming

Data", in IEEE Transactions On Intelligent Transportation Systems, Vol. 14, No. 3, September 2013.
[2]Han-wen Chang, Yu-chin Tai And Jane Yung-jen Hsu, "Context-aware Taxi Demand Hotspots Prediction", In Int. J. Business Intelligence And Data Mining, Vol. 5, No. 1, 2010.
[3]Longbiao Chen, Daqing Zhang, Leye Wang, Dingqi Yang, Xiaojuan Ma, Shijian Li, Zhaohui Wu, Gang Pan, Thi-Mai-Trang Nguyen, Jérémie Jakubowicz, "Dynamic Cluster-Based Over-Demand Prediction in Bike Sharing Systems", in ACM, 2016.
[4]Junbo Zhang, Yu Zheng, Dekang Qi, Ruiyuan Li, Xiuwen Yi, "DNN-Based Prediction Model for Spatial-Temporal Data", in ACM SIGSPACIAL, 2016.
[5]Junbo Zhang, Yu Zheng, Dekang Qi, "Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction", in arXiv:1610.00081v2.