Student's Declaration

We hereby declare that the work presented in the report entitled **"Traffic Analysis through GPS Data"** submitted by us for the partial fulfillment of the requirements for the degree of *Bachelor of Technology* in *Electronics and Communication & Engineering* at Indraprastha Institute of Information Technology, Delhi, is an authentic record of our work carried out under guidance of **Dr. Pravesh Biyani**. Due acknowledgements have been given in the report to all material used. This work has not been submitted anywhere else for the reward of any other degree.

Avdesh Kumar
Saumya Balodi

Place & Date: IIITD, Delhi, 07/01/2019

Certificate

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

.....
Pravesh Biyani

Place & Date: IIIT, Delhi, 07/01/2019

Abstract

Public transport is one of the major forms of transportation in the world. This makes it vital to ensure that public transport is efficient. The paper presents an approach to reduce the waiting time of Delhi buses by analyzing the traffic behavior and proposing a timetable. The algorithm uses a constrained clustering algorithm for classification of trips. This was a tedious task as there wasnt any timetable which was being followed.It further analyses the data statistically to provide a timetable which is efficient in learning the inter and intra month variations. The results have been shown in transit data collected by Delhi Integrated Multi-Modal Transit System Limited (DIMTS) for the months of October and November 2017

Keywords: Timetable Optimization, Bus Scheduling, Data Analytics

Work Distribution

Saumya
I worked on calculating the waiting time for the subsequent bus stops for the October data. For this, I averaged the time difference of all the trips over 15 minutes interval. I also worked on calculating all the pre waiting times for the two bus routes at all the stops. We were also provided the Bangalore bus data which was in SQL format. Since I had some experience with sql before, I worked on extracting the data. Then in December in order to extend the algorithm to Open Transit Data, I learned and worked with pb format. I wrote a script to store data of different buses by matching the appropriate route from a static file.

Avdesh
I made the time table for the first stop by clustering similar trips on the basis of their departure time and then made the complete time table using the data generated by Saumya and post that we calculated the post time tabling results. Along with this work, we also made a tool that uses OTD live data and shows real time location of buses on a map along with their route details. I designed the back-end for that tool on a python based flask server.

Contents

Chapter 1

Introduction

Public transport is one of the most popular means of transportation in various metropolitan cities across the globe. According to the Economic Survey of Delhi 2005-06, buses account for nearly 60 percent of the total demand. While it is well understood that the public transport helps in combating air pollution and congestion caused due to single-occupancy vehicles,the usage of buses in Delhi and other cities in India has seen a nominal decline while the overall travel demand has simultaneously increased.One of the main reasons for this decline in the city of Delhi (and other cities in India) is the lack of re-liability of the bus routes. The timetable is often not made by the transit authorities. Moreover, it is often outdated soon due to the rapid change in infrastructure and the traffic conditions resulting in degradation in the reliability of buses. Finally, this decrease in reliability leads to unknown waiting times at the bus stops for the passengers. Due to the absence of a timetable, most bus trips are operated in an ad-hoc fashion making it extremely difficult for the passengers to trust the public transport network leading to a decrease in passenger trips.Interestingly, the various trips in a given bus-routes till follow a certain pattern in a given day thanks to the pattern in the traffic conditions throughout the day.In other words, even when the transit operators do not follow an explicit time-table, there is an implicittimetable that is followed which is not completely random. The aim of this work is not to uncover this pattern and suggest the implicit time-table that is fol-lowed by the bus-routes in the city of New Delhi. The main goal of this work is to unearth this pattern and develop an operational time-table of the bus routes in the city of Delhi. The efficacy of this suggestedtimetable is measured in terms of the average waiting time a passenger has to endure at various stops in the given route assuming she follows the timetable. This waiting time should ideally be lower than when the passenger does not follow the suggested timetable arrives at the same stop in a random fashion.To arrive at the time-table, we sample two routes operated by the Delhi transport corporation. Out of the two routes, one is frequent, while the other operates at an average frequency of thirty minutes. We use the collected GPS feed of the buses.The paper presents a novel approach which simplifies the problem statement to propose an efficient algorithm for defining the timetable based on GPS bus data collected over a period of two months.

Users

Our target audience are the bus commuters who use buses for their daily mode of communication. The time table created will save their time and energy spent on waiting for buses. It will give them a clear picture of when a bus will arrive at a particular stop and hence, prepare their travel accordingly. This would lead to more commuters preferring buses over other modes of transport thus increasing revenue.

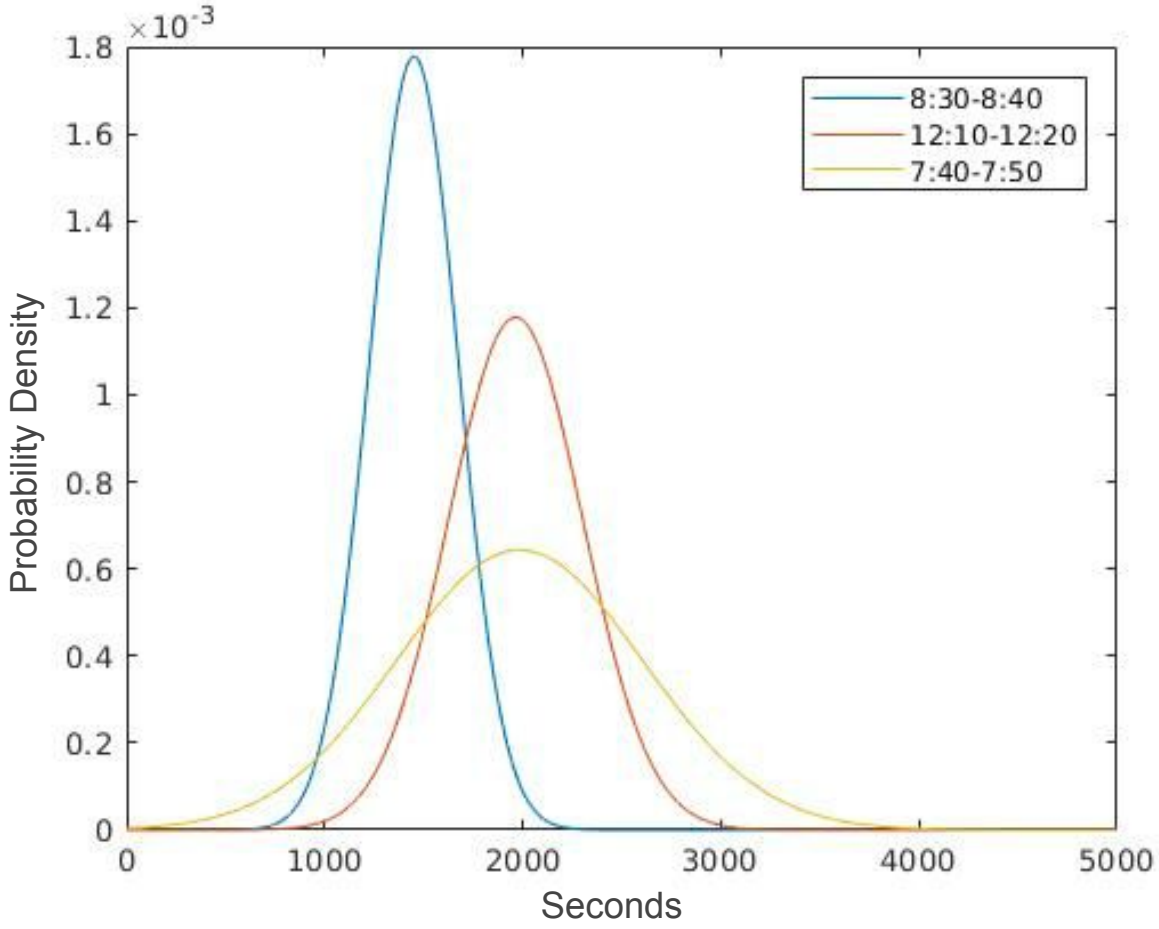

Figure 1.1: Probability distribution of the arrival time at stop 17 for route no. 425 at different times of the day.

Chapter 2

Design, implementation and validation of solution

Current bus timetables are formulated without considering the traffic behavior based on time of the day and the day of the week. It has been observed from the data that these are significant components which affect the arrival time of the bus at a particular stop which needs to be carefully scrutinized to provide a more efficient time-table.

The paper analyses data collected by the Delhi Integrated Multi-Modal Transit System Limited (DIMTS) for two months. The data contains time-based GPS coordinates for all the buses running throughout the day. The timetable was defined for every bus stop by considering it as a node. The data has GPS coordinates sampled every 10 seconds and physical coordinates for every bus stop. To convert this into node based data, any data sample with coordinates within a threshold is considered to a part of that node. This threshold was set to 50 meters for adequate results.

The variables are defined as:

- Total bus stops N are present on the route.
- The data has been taken for d days.
- i represents the trip variable, $\in (0, x^k)$.
- j denotes the stop variable, $\in (0, N)$.
- k represents the day variable $\in (0, d)$.
- x^k represents the number of trips on the k^{th} day.
- $t^k_{i,j}$ represents the arrival time of the bus at the j^{th} stop for the i^{th} trip on the k^{th} day.

To reduce the effect of noise in the data, the algorithm samples the data at every 3^{rd} stop. The algorithm can be summarized in two stages, definition of the starting time at the first stop and calculation of arrival time for the following stops based on the first stop.

The optimization problem finds the most optimal $\widehat{t}^k_{i,j}$ such that,

$$\widehat{t}^k_{i,j} = \min \quad [t^k_{i,j}] \tag{2.1}$$

2.0.1 Defining the starting time

As the time-table depends on the departure time of the bus from the first stop, the algorithm uses K-means clustering for this step. Unlike the standard clustering algorithm, which minimizes the inter-class variance, this approach minimizes the variances while keeping a minimum distance between the clusters. The minimum distance is equivalent to the standard frequency of the buses as suggested by the Transportation Department.

Each of the set of data points in each cluster created using this approach is represented by $C^{(n)}$ at iteration n which contains $M^{(n)}$ data points. The centroid of the l^{th} set of cluster is denoted by $\mu_l^{(n)}$ at iteration n and the total number of clusters are c. Each of these cluster sets can be mathematically represented using equation 2.2, where the new data point being classified is tp.

$$C_l^{(n)} = \{tp : ||tp - \mu_l^{(n)}||^2 < ||tp - \mu_m^{(n)}||^2, \forall m; 1 < m < c\} \tag{2.2}$$

The definition of new clusters is conditioned on equation 2.3, where T_1 is the frequency of the bus.

$$\{||\mu_l^{(n)} - \mu_m^{(n)}|| < T_1, \forall l, m; 1 < l, m < c\} \tag{2.3}$$

The mean is updated using the equation 2.4.

$$\mu_l^{(n+1)} = \frac{\sum_{t_{i,j}^{k,(n)} \in C_l^{(n)}} t_{i,j}^{(n),k}}{|C_l^{(n)}|} = \frac{\mu_l^{(n)} * M^{(n)} + tp^{(n+1)}}{M^{(n)} + 1} \tag{2.4}$$

In the first step, the average departure time for a bus is calculated. While considering each new bus, the nearest mean was searched, and if the difference between the departure time and the nearest mean is less than the threshold T_1, then the mean of that cluster was updated by including this data point as well. Otherwise, this would be considered as a new cluster. Finally, only if the total number of points in a cluster is greater than a threshold T_2, the starting time is considered valid. Otherwise, the data point is considered to be an outlier case and is not considered for further calculations. Due to noise in the data and inconsistency in the starting time of the first bus, the paper grid searches for the best threshold to find the appropriate starting time which reduces the waiting time as well as the bunching of buses. The threshold T_2 found to be 10 days which is one-third of the total number of days for which the data is used.

2.0.2 Adjacent Stops

For every subsequent bus stop, the average time to reach that bus stop is calculated from the first node for every 15-minute interval. The timetable for these nodes is defined as the start time plus the average time taken to reach that stop (in the particular time range). The optimal time for every adjacent stop can be written as,

$$\widehat{t}_{i,j}^{k} = \widehat{t}_{i,1}^{k} + \frac{\sum_1^d (t_{i,j}^k - \widehat{t}_{i,1}^k)}{d} \quad \forall j \in (2, N) \tag{2.5}$$

Figure 1.1 shows that the arrival time of the bus at each stop is in the form of a Gaussian distribution which has higher variance as the day progresses. By taking the mean in the above

approach, the algorithm minimizes the waiting time, which is defined as the variance of this distribution. Thus giving the most optimal timetable.

The data is first divided into 15-minute time slots for more accurate calculation and estimation of traffic behavior. This dynamic time-based timetable takes into consideration the regular changes which come in the traffic behavior every 15 minutes.

2.0.3 Calculating Waiting Time

The pre-timetable waiting time (pwt) is defined as the expected time a person will wait if he/she arrives at a bus stop at any random time. If the next bus arrives after N_0 minutes from the previous bus then the expected waiting time for the next bus at that stop will be:

$$pwt = \sum_{n=1}^{N_0} n/N_0 = (N_0 + 1)/2 \tag{2.6}$$

The algorithm averages this over all the trips for a bus route to get the mean waiting time for different stops.

For calculating the post-timetable waiting time, random points have been selected throughout the day. Waiting time at that instance is equal to the difference of this from the time of arrival of the next bus. This is averaged for all such randomly selected instances throughout the period for which it is considered.

Chapter 3

Results

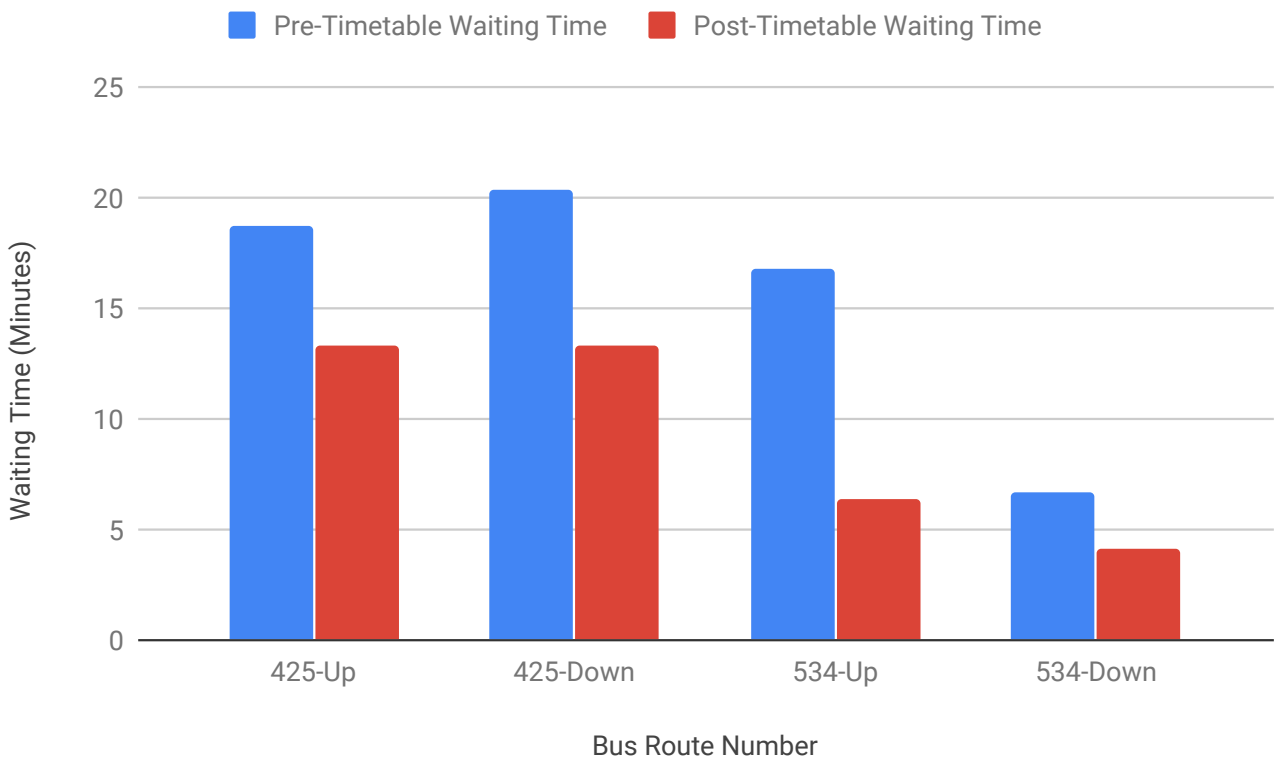

Figure 3.1: Comparison of Waiting Time at the 1st Node for all the Buses.

The paper proposes an approach to create a timetable for the two buses using a relatively direct procedure which resulted in a significant decrease in the waiting time which has been applied to the two bus routes - 425 and 534. It has been applied separately to the up and down routes of the buses.

To show the efficiency of the algorithm in reducing the waiting time of buses, the results have been presented in the form of bar graphs representing the difference brought by the introduction of the timetable. The results of using clustering for the initial node, by following the first protocol, has been presented in figure 3.1. This shows that there a significant amount of randomness in the starting time of these buses. The proposed approach has been able to reduce this random-

ness of the starting time. If this is strictly followed it would yield much more promising results for the subsequent stops as well. As the resultant increase in waiting time of the following bus stops is the addition of the randomness in the 1st stop along with the randomness in the traffic behavior. The first can be controlled by proper implementation of the timetable by public/ private transportation departments.

For the second experiment, the paper looks at the formation of a timetable by looking at the inter-month variations in the data. In doing so, the model shows its efficiency in learning the randomness which is present within the constraints of the same month. Figure 3.2 shows a comparison of the waiting time without any timetable with the waiting time post the new timetable for bus no. 534. Figure 3.3 shows the graph for the the up-route while figure 3.2 shows the graph for the down-route. Both the graphs compare the waiting time at different time intervals.

Lastly, the model has been tested on a completely unseen data after being trained on the data for October. The results upon being tested on the November data has been presented in figure 3.5 and 3.4 for the up and down route of bus number 425 respectively.

The impact of a timetable is more in the case of bus route 425 since the bus is more irregular. Due to the high frequency of bus number 534 introducing large changes isn't possible. During the calculation of final waiting time, the randomness in the starting time due to human constraints have been taken into consideration. As the final waiting time is also calculated using the same probabilistic distributing of the starting stop. Thus, showing that even if the timetable is not followed closely, it would still impact the waiting time considerably. These situations make this essential for real-world applications where human constraints are significant.

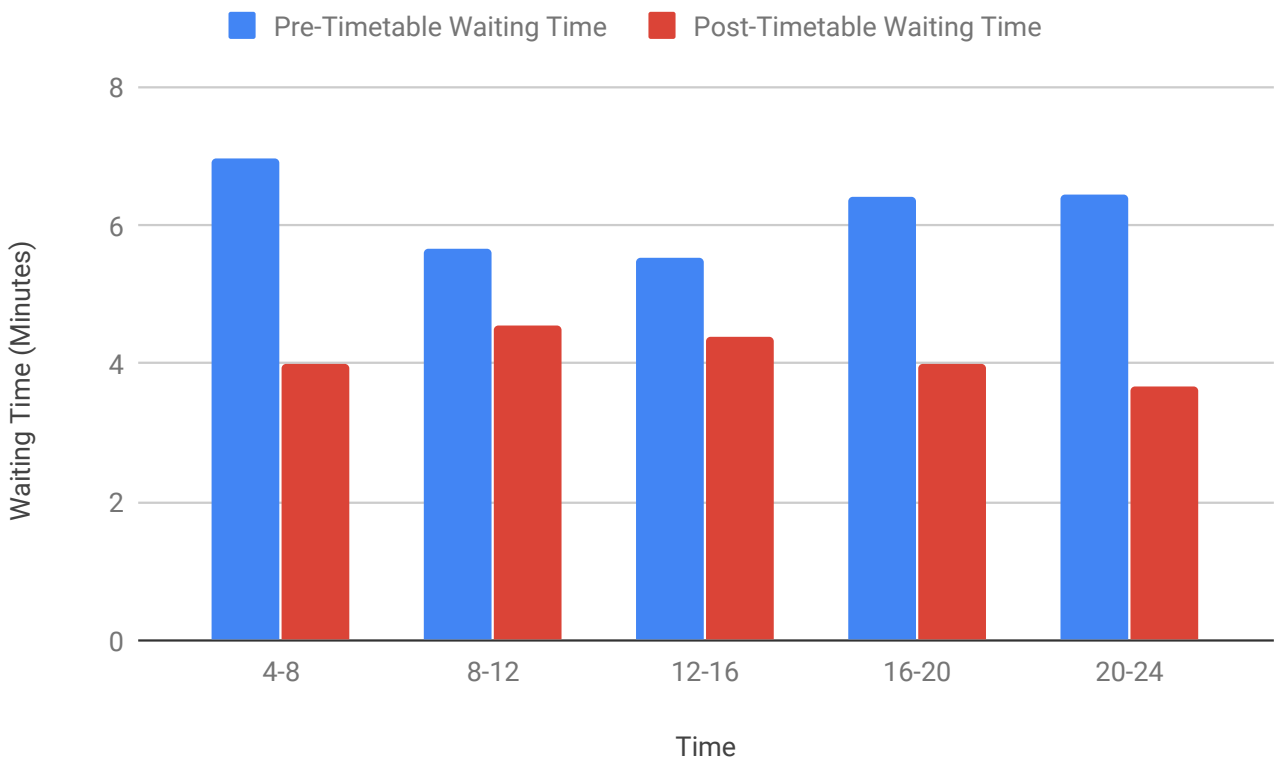

Figure 3.2: Comparison of waiting time for bus no. 534 down route when training and testing on alternate days.

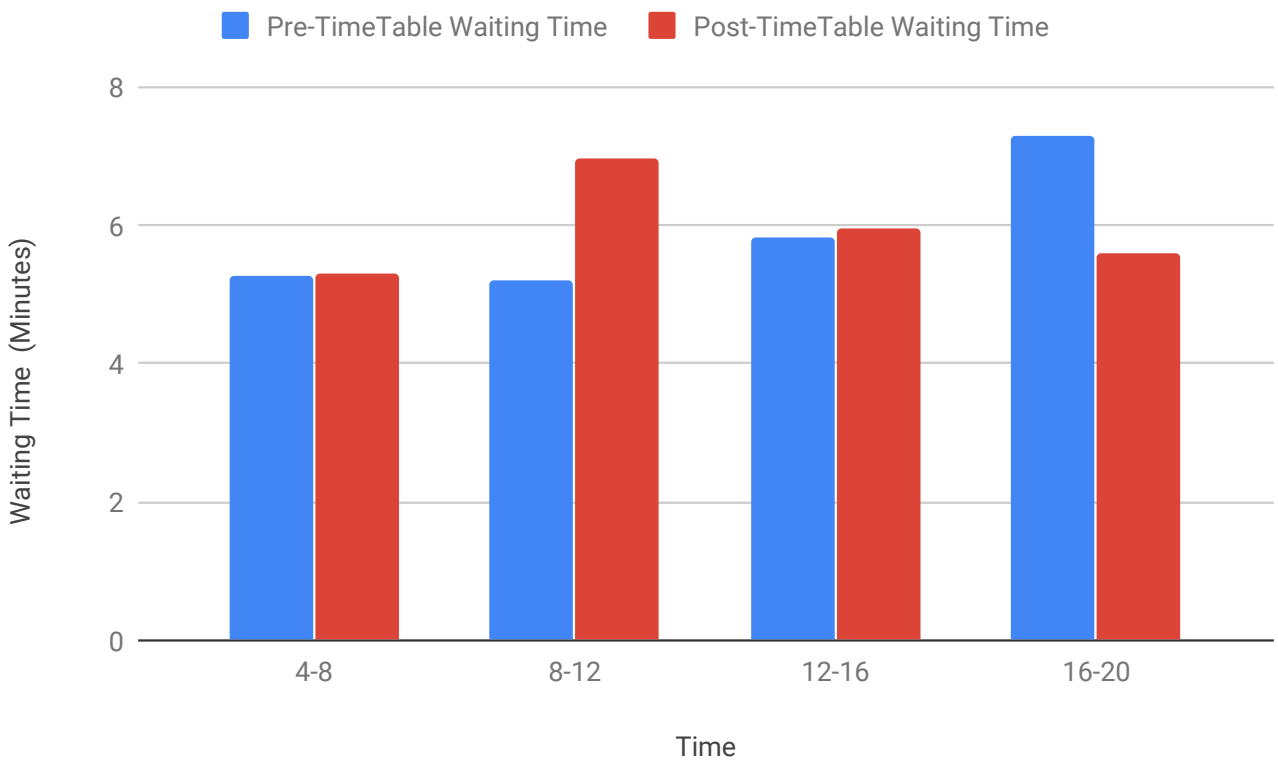

Figure 3.3: Comparison of waiting time for bus no. 534 up route when training and testing on alternate days.

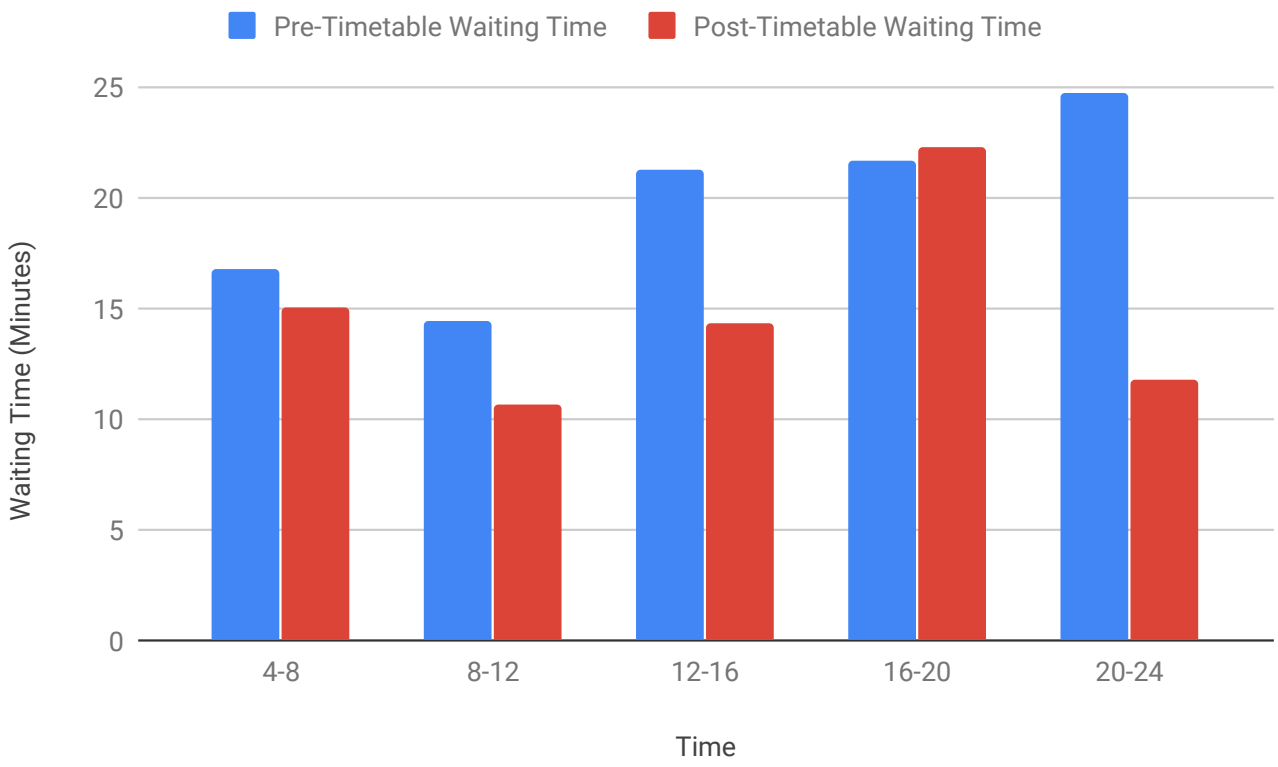

Figure 3.4: Comparison of waiting time for bus no. 425 Down Route before and after the timetable when testing on November.

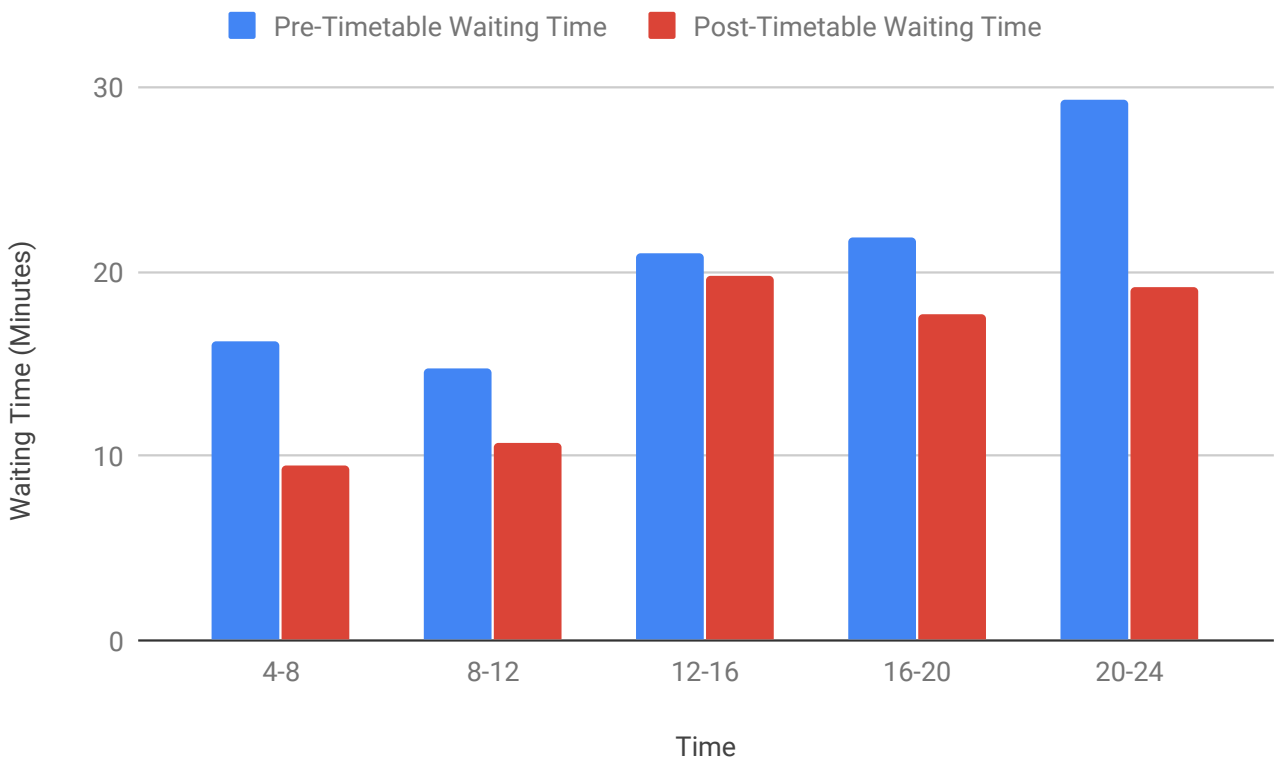

Figure 3.5: Comparison of waiting time for bus no. 425 Up Route before and after the timetable when testing on November.

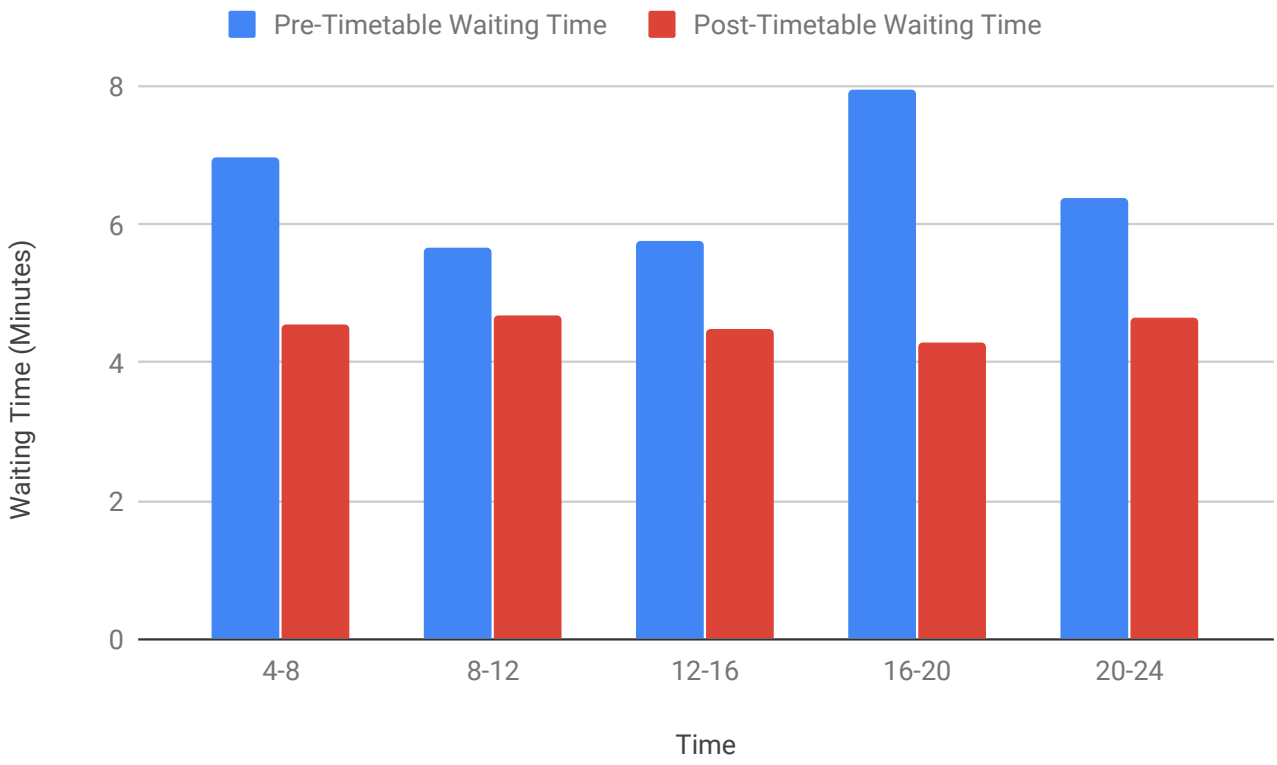

Figure 3.6: Comparison of waiting time for bus no. 534 down route when testing on November.

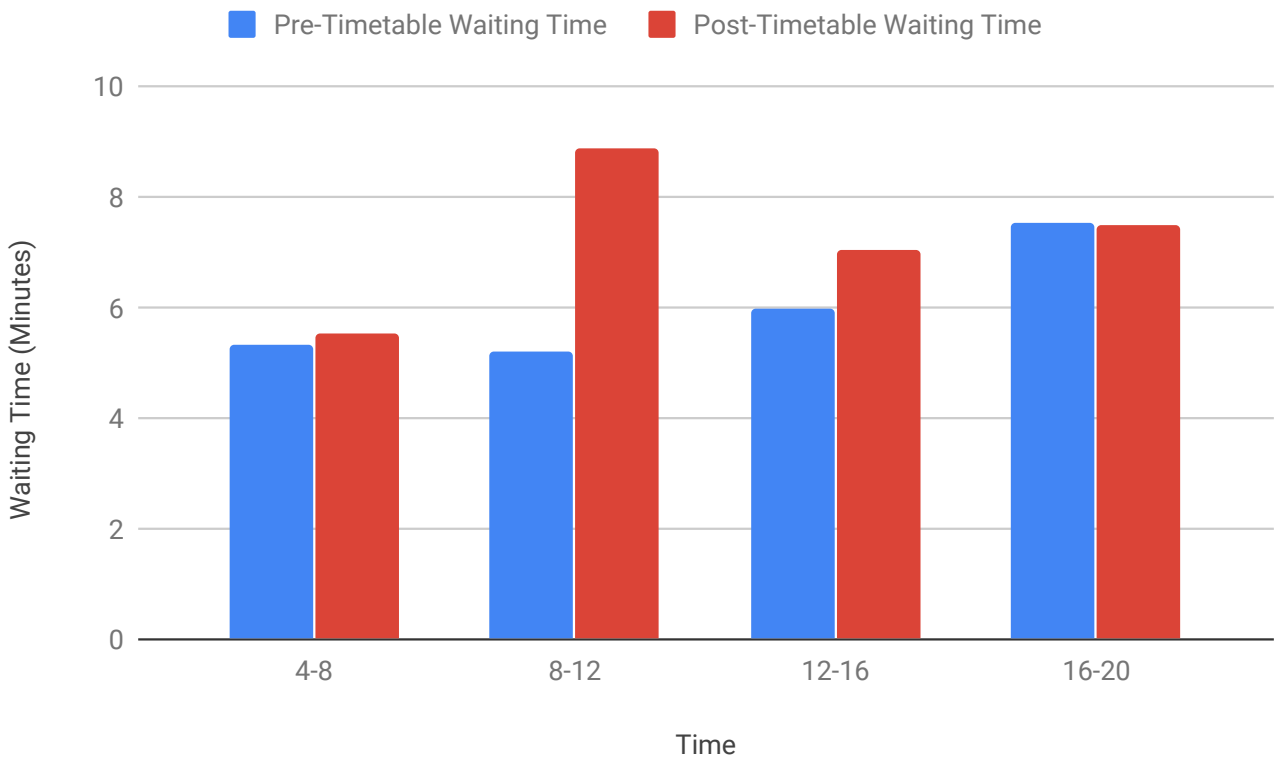

Figure 3.7: Comparison of waiting time for bus no. 534 up route when testing on November.