Imperial College London

Department of Computing

MEng Individual Project Interim Report

Active Delay Warning Transport App

Author:
Yangfan Zhang

Supervisor:
Dr. Peter McBrien

May 13, 2015

Contents

Todo list

Chapter 1

Introduction

The London bus network carries 2.4 billion passengers a year, more than the rest of England combined [1].

The bus arrival times published by Transport for London (TfL) are currently widely available on digital live bus arrivals signs at more than 2,500 bus stops [2]. Passengers can also check this information by sending a text message with the bus stop code, as well as doing a quick search online or on mobile applications.

Passengers rely on the bus arrival times to plan their journey, by factoring in the waiting time when choosing the buses to take. Current London journey planning software takes the journey start time, start location, and destination as input, and recommends routes consisting of a variety of travel mode, with an estimate travel time for each suggested journey. Such popular planners include Google maps [3], Citymapper [4] and Transport for London Journey Planner [5].

However, the accuracy of the bus arrival times published is affected by many external factors. For example, when there is heavy traffic, the buses are likely to be delayed by a difference significant enough for a change in passengers' route picking. Yet, this delay information is not reflected in the arrival time data or the estimated journey time early enough for the passengers to make a decision to choose an alternate route. As a result, passengers waste time waiting for buses that come much later than expected, or chosing to board a bus that takes far longer than the estimated journey time. Although the average bus delay is 1 minute, there was a 16.6 chance of waiting for more than 10 minutes [11].

This can be avoided if passengers are informed of the delays in bus arrival times in advance. Such delays can predicted by analysing the historical delays. This is achieved by collecting data from the TfL live bus arrivals API stream feed, and estimate the average journey time required between two locations on a given time of the day. Next, a bus arrival time table with delays during various time windows over a week can be crafted and fine tuned incrementally. This timetable is the reference point for sending out delay warnings.

Chapter 2

Background

2.1 London Bus Network

The bus network in London is one of the largest and most accessible in the world. It is carrying a staggering number of passengers, with more than 2.4 billion journeys in 2013/14, which was more than any year since 1959 [1].

On an average day between 2005 and 2010, about 14% of the trips made by London residents were by bus [7]. They spent on average 14 minutes per day on these bus trips.

There are currently 19,345 bus stops, and 680 routes served by 8,765 buses daily in London[8].

2.1.1 Bus Network Performance

TfL published the following figures in the second quarter 2014/2015 buses performance data [6].

For the high frequency services, the average scheduled wait was 4.86 minutes, the average excess wait was 0.94 minutes, and the average actual wait was 5.80 minutes. While passengers could expect the buses to come within 10 minutes 83.4% of the time, there was 15.1% chance of waiting for 10-20 minutes, 1.3% chance of waiting for 20-30 minutes, and 0.2% chance of waiting for more than 30 minutes.

For the low frequency services, 87% of the buses services were on time, and 11.4% were 5-15 minutes late.

For the night buses, 84.5% of the services were on time. The average excess wait was 0.68 minutes.

The bus arrivals might be affected by traffic congestion, staff availability, and engineering problems or mechanical breakdown [11].

2.2 Transport for London Open Data

TfL provides free, open data consisting of around 30 feeds and APIs [9].

There are 3 forms of data, static data files which rarely change, feeds that refreshed at regular intervals, and API(Application Programming Interface) that enable a query to receive a bespoke response, depending on the parameters supplied.

Over 5,000 developers have registered for the open data[9], and around 200 travel apps are powered by [1].

2.2.1 Live Bus Arrivals API Stream

The Live Bus Arrivals API Stream provides the predicted time until a bus is expected to arrive at a stop. These predictions are available for the next 30 minutes at any point in time. For example, at 9am, the stream will provide predicted bus arrivals up to 9.30am on the same day. This data is refreshed every 30 seconds [10].

In order to collect bus arrival data for analysis, we supplied the following selected parameters to the API. These parameters specify the fields returned by the API. We then stored the data received to a MySQL database continuously.

Parameters supplied

- *StopID* This is the alphanumeric identifier of a bus stop.
- *LineName* This is the route number that is displayed on the front of the bus on any publicity advertising the route.
- *VehicleID* The unique identifier of the vehicle.
- *DirectionID* This identifies the direction of the trip (either outbound or inbound) that the vehicle is on.
- *TripID* The identifer of the specific trip that the prediction is for.
- *EstimatedTime* This is the predicted time of arrival for the vehicle at a specific stop.
- *ExpireTime* This is the time at which the corresponding prediction is no longer valid and should stop being displayed.

#	Name	Type	Collation	Attributes	Null	Default
1	stop_code_lbsl	varchar(64)	latin1_swedish_ci		No	None
2	route	varchar(64)	latin1_swedish_ci		No	None
3	vehicle_id	varchar(64)	latin1_swedish_ci		No	None
4	trip_id	varchar(64)	latin1_swedish_ci		No	None
5	arrival_time	timestamp			Yes	NULL
6	expire_time	timestamp			Yes	NULL
7	recorded_time	timestamp		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP

Figure 2.1: Arrivals table schema

stop_code_lbsl	route	vehicle_id	trip_id	arrival_time	expire_time	recorded_time
1075	45	9312	134291	2015-01-23 16:01:57	2015-01-23 16:01:57	2015-01-23 15:49:19
11491	95	19423	422987	2015-01-23 15:49:56	2015-01-23 15:49:56	2015-01-23 15:49:19
1164	109	3642	1311824	2015-01-23 16:00:35	2015-01-23 16:00:35	2015-01-23 15:49:19
14799	94	15216	563169	2015-01-23 16:01:45	2015-01-23 16:01:45	2015-01-23 15:49:19
14853	204	11827	1082551	2015-01-23 15:40:09	2015-01-23 15:40:09	2015-01-23 15:49:19
15505	62	5310	151349	2015-01-23 16:04:56	2015-01-23 16:04:56	2015-01-23 15:49:19
16109	186	10118	1098109	2015-01-23 15:46:29	2015-01-23 15:46:29	2015-01-23 15:49:19
17541	163	7814	788546	2015-01-23 15:44:17	2015-01-23 15:44:17	2015-01-23 15:49:19
19280	112	9938	75521	2015-01-23 16:00:29	2015-01-23 16:00:29	2015-01-23 15:49:19

Figure 2.2: Data stored in arrivals table

Each data entry contains an estimated arrival time for each bus journey at a given bus stop. This estimated arrival time is stored in the database via an UPDATE statement, which ensures that only the latest estimated arrival times per journey per bus stop are stored. This has been achieved through running a daemon process written in python on a virtual host. Figure 2.1 shows the schema for arrivals table. See Figure 2.2 for sample data stored.

Assumption

We assume that the actual bus arrival time is the midpoint between the last estimated arrival time, and the system time when the clear signal (Expire-Time $= 0$) is received.

2.2.2 Bus Stop Locations and Routes

This data file contains network information on the location of all bus stops in London, and the sequence of bus stops that every bus route in London stops at.

From this data file, we extracted information on all pairs of neighbouring

#	Name	Type	Collation	Attributes	Null	Default
1	**route**	varchar(4)	utf8_general_ci		Yes	*NULL*
2	**start_stop_code_lbsl**	varchar(6)	utf8_general_ci		Yes	*NULL*
3	**end_stop_code_lbsl**	varchar(6)	utf8_general_ci		Yes	*NULL*

Figure 2.3: Neighbours table schema

route	start_stop_code_lbsl	end_stop_code_lbsl
19	10002	11469
30	10002	11469
4	10002	11469
43	10002	29772
N19	10002	11469
N41	10002	29772

Figure 2.4: Data stored in neighbours table

bus stops and the routes that serve between them. We save this information in the neighbours table in the database. Figure 2.3 shows the neighbours table scheme. Figure 2.4 shows the sample data stored in neighbours table.

Finding the average travel time between neighbouring stops

To experiment with the queries, we selected one pair of the neighbouring stops (10002, 11469), and listed the time required to travel from stop 10002 to stop 11469 by finding the difference in arrival times for each journey. Sample entries of this list is shown in Figure 2.5.

We then calculated the average journey time required to travel from 10002 to 11469 for each hour in each week of the day. This information is stored as a timetable, which would be used for further analysis.

Figure2.6 shows the timetable generated. Each cell indicates the average

start_stop	end_stop	route	trip_id	day	start_time	end_time	length_in_secs
10002	11469	19	346930	Monday	1/12/2015 12:34:16	1/12/2015 12:35:28	72
10002	11469	4	547334	Monday	1/12/2015 12:39:55	1/12/2015 12:41:14	79
10002	11469	19	346961	Monday	1/12/2015 12:40:04	1/12/2015 12:41:53	109
10002	11469	30	226765	Monday	1/12/2015 12:45:07	1/12/2015 12:47:11	124
10002	11469	4	547676	Monday	1/12/2015 12:47:17	1/12/2015 12:49:12	115
10002	11469	19	346976	Monday	1/12/2015 12:48:21	1/12/2015 12:50:09	108
10002	11469	30	276872	Monday	1/12/2015 12:54:05	1/12/2015 12:55:08	63

Figure 2.5: List of journey time from stop 10002 to stop 11469

hour	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
0	NULL	NULL	75.1	56	63.5	62.5	NULL
1	NULL	NULL	68.4	60.6	69.6	54	NULL
2	NULL	NULL	50.5	32.5	58.5	86.5	NULL
3	NULL	NULL	NULL	64	69	51.5	NULL
4	NULL	NULL	NULL	88.5	62	46	NULL
5	NULL	NULL	NULL	61	73	59.3	NULL
6	NULL	NULL	NULL	69	79.4	75.1	NULL
7	NULL	NULL	NULL	88.1	81.7	84.3	NULL
8	NULL	NULL	NULL	90.2	118.8	116.7	NULL
9	NULL	NULL	NULL	82	90.9	101.7	NULL
10	NULL	NULL	NULL	74.4	90.8	145.6	NULL
11	NULL	NULL	NULL	111.2	97.9	124.2	NULL
12	NULL	95.5	267.3	169.9	146.5	114.7	NULL
13	NULL	125.4	245.3	168.2	104.7	NULL	NULL
14	NULL	90.4	NULL	123.8	85.7	NULL	NULL
15	NULL	120.7	113	153.1	87.2	NULL	NULL
16	NULL	168.1	125.1	134.1	100.5	NULL	NULL
17	NULL	113	110.4	124.6	126.5	NULL	NULL
18	NULL	127.1	110.9	179.8	148.6	NULL	NULL
19	NULL	115.5	149.3	243.7	160	NULL	NULL
20	NULL	87.7	96	140.5	85.2	NULL	NULL
21	NULL	80.1	94	94.5	85	NULL	NULL
22	NULL	75	93.3	80.4	100.7	NULL	NULL
23	NULL	54.6	75.6	78	73.7	NULL	NULL

Figure 2.6: Average journey time in seconds from stop 10002 to stop 11469 for each hour of each day of week

journey time required to travel from stop 10002 to stop 11469 at a give hour of a give week of day. The **NULL** values are due to a current databases performance issue. This will be resolved later.

We plan to construct a timetable this way for each pair of the neighbouring bus stop.

2.2.3 Journey Planner Bus Timetables

The official travel times between bus stops at different times are extracted from the Journey Planner Bus Timetables.

Questions to answer in this section

- The first item
- The second item
- The third etc ...

```
xsi:schemaLocation="http://www.transxchange.org.uk/ http://www.transxchange.org.
```

```
The type of information that TransXChange can be used to exchange
```

```
TransXChange schemas can be used to exchange:

bus schedules including stops, routes, departures times, departure frequencies, 
the days on which the services run, including public holidays availability
term times and holidays of schools and other bus serviced organisations
details of the statutory registration including short notice registrations
bus operators information
fare stages, positioning runs, garages, layovers, duty crews and on-board ticket
information about accessibility of stops and services for wheelchair and other u
```

2.2.4 iBus System

The actual arrival time of buses on a route at any given bus stop for the selected day are made available by the London Buses iBus system.

Currently, the routes were selected as the first of the New Routemaster (New Bus for London). Data will be published once a week [11].

This data is stored in comma-separated values (CSV) format. There are two potential uses for this data.

- We can integrate it into the arrivals table to improve the precision of the arrivals data. Since the current entries in the arrivals table contain estimated bus arrival times, the integrated of the iBus data which contains real bus arrival times will likely boost the precision of the data in arrival table.

- We can compare the predicted delays with the iBus data for performance evaluation.

2.3 Additional Background Materials

Here's a list of the potential points to be covered in the final report.

- Geocoding data
- Possible analysis methodologies such as regression
- Literature review on similar research done in other areas of the world

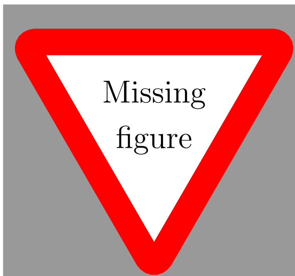
Chapter 3

Technical Details

Missing Testing section

3.1 Overview

- Used Django Framework to build the backend that retrieves data from a MySQL database.
- Used AngularJS with Twitter Bootstrap for frontend development



graph to show the overall structure

3.2 Data Collection

3.3 Backend

3.3.1 MySQL

3.3.2 Django Framework

3.3.3 Deployment

3.4 REST API

3.4.1 Django Rest Framework

3.5 Frontend

3.5.1 AngularJS

3.5.2 Deployment

3.6 Software Project Management

3.6.1 Jenkins

3.6.2 Trello

Chapter 4

Evaluation Plan

4.1 Compare predictions to real data

We can compare the predictions generated to the live bus arrivals data stored in the arrivals table. We can calculate the standard deviation of the difference in the predicted time and actual arrival time. This value will be the direct indicator of the accuracy of the predictions.

4.2 User feedback

We can get students that take buses regularly to use the mobile application to test the accuracy of the delay predictions. The user feedback gathered will be the second input for evaluation.

References

[1] Transport for London Annual Report and Statement of Accounts 2013/14, http://tfl.gov.uk/cdn/static/cms/documents/annual-report-2013-14.pdf [visited on 27/01/2015]

[2] Transport for London Live Bus Arrivals, http://www.tfl.gov.uk/modes/buses/live-bus-arrivals [visited on 27/01/2015]

[3] Google Maps, https://www.google.co.uk/maps [visited on 30/01/2015]

[4] CityMapper, https://citymapper.com/london [visited on 30/01/2015]

[5] TfL Plan A Journey, http://tfl.gov.uk/plan-a-journey/ [visited on 30/01/2015]

[6] Transport for London Buses Network Performance Second Quarter 2014/15, http://tfl.gov.uk/cdn/static/cms/documents/network-performance-latest-quarter.pdf [visited on 27/01/2015]

[7] Travel in London, Supplementary Report: London Travel Demand Survey (LTDS), http://www.tfl.gov.uk/cdn/static/cms/documents/london-travel-demand-survey.pdf [visited on 28/01/2015]

[8] Transport for London Bus Stops Locations and Routes Open Data https://www.tfl.gov.uk/info-for/open-data-users/our-feeds [visited on 28/01/2015]

[9] Transport for London Open Data https://www.tfl.gov.uk/info-for/open-data-users/our-open-data [visited on 28/01/2015]

[10] Transport for London Live Bus & River Bus Arrivals API Interface Documentation https://www.tfl.gov.uk/cdn/static/cms/documents/tfl-live-bus-river-bus-arrivals-api-documentation-v16.pdf [visited on 28/01/2015]

[11] Transport for London Buses Performance Data https://www.tfl.gov.uk/corporate/publications-and-reports/buses-performance-data#on-this-page-5 [visited on 28/01/2015]