How full will my next bus be? A Framework to Predict Bus Crowding Levels

Tahereh Arabghalizi
Department of Computer Science
School of Computing and Information
University of Pittsburgh
tahereh.arabghalizi@pitt.edu

Alexandros Labrinidis
Department of Computer Science
School of Computing and Information
University of Pittsburgh
labrinid@cs.pitt.edu

ABSTRACT

Public transit is one of the first things that come to mind when someone talks about “smart cities.” As a result, many technologies, applications, and infrastructure have been deployed already to bring the promise of the smart city to public transportation. Most of these have focused on answering the question “when will my bus arrive?”; little has been done to answer the question “how full will my next bus be?” which also greatly affects commuters' quality of life. In this paper, we develop a framework to address the fullness question. We formulate the problem as a classification problem, develop a framework to enable predictions using Random Forests, and evaluate our proposed techniques using data from the Pittsburgh region.

CCS CONCEPTS

• **Computing methodologies** → *Modeling and simulation.*

KEYWORDS

smart city, intelligent transportation, urban computing

ACM Reference Format:
Tahereh Arabghalizi and Alexandros Labrinidis. 2019. How full will my next bus be? A Framework to Predict Bus Crowding Levels. In *KDD urbcomp2019, Anchorage, Alaska.* ACM, New York, NY, USA, 8 pages.

1 INTRODUCTION

The rapid growth of urbanization during the past decades is necessitating increased efficiency in city operations. This is manifesting as sensing technologies for data collection, advanced models and algorithms, and relevant data dissemination to city dwellers, whose lives these big data and technologies are ultimately trying to improve [1]. Collectively these techniques are often referred to as “smart city” technologies.

A textbook example domain for a smart city technology is that of *public transportation.* Everybody who lives in a city would wish for public transportation to be “better.” Problems such as bus delays, crowded buses, and general lack of public transportation options especially during rush hours make commuters dissatisfied and unhappy about the city's public services.

A plethora of technologies, applications, and infrastructure have been deployed already to bring the promise of the smart city to public transportation. These include GPS tracking of buses to reliably predict their arrival times, the standardization of transit schedule data [7], and mobile applications (e.g., Transit App [18] and MoovIT [10]) to make such transit information available in real-time to commuters.

Although a lot of work has been done towards figuring out the answer to question “when will my bus arrive?”, little has been done to answer the question “how full will my next bus be?” which also greatly affects the commuters' quality of life. The **Pitt Smart Living** project aims to address exactly that, by considering multimodal transportation in a holistic way. In particular, the project participants will design, develop, deploy, and evaluate a platform that will integrate information from and align the incentives of all involved stakeholders (commuters, transport operators, and local businesses) towards increasing the utilization and quality of public transportation [13]. For example, while waiting at the bus stop, a commuter will receive a push notification alerting them to the next bus being full. In addition, it would offer them a discount towards coffee/tee at the coffee shop around the corner (say $2 off), if they would take a later bus.

1.1 Problem Statement

This work attempts to answer the following question *“how full will my next bus be?”.*

In particular, we would like to be able to predict what would be the crowding level of buses of a certain route [1] arriving to a specific bus stop within a given 15-minute time interval. Such prediction could help passengers make better decisions about which bus to take. We believe this to be a win-win situation in which passengers could trade a few minutes of their time for various incentives (at nearby businesses) and the public transportation system will be more balanced in terms of utilization.

Towards this, we propose a machine learning approach using classification models to predict bus crowding levels. To drive our proposed techniques we will use data for the Pittsburgh area; our techniques are trivially generalizable to other areas.

[1] A bus route is a set of stops and buses with a starting point, an end point, and a direction (inbound or outbound).

KDD urbcomp, 2019, Anchorage, Alaska

Table 1: Statistics about the Pittsburgh area bus data

number of routes (in-/out-bound)	98*2=196
number of stops	6,931
number of records in dataset before cleaning	100,869,765
number of records in dataset after cleaning	89,901,555
number of columns in dataset	189
number of useful columns	18

1.2 State of the Art

The reliability of the public transportation system, in particular with regards to travel time and available space, greatly affects commuters' quality of life in urban travel [15]. Many research works have proposed techniques to predict bus arrival times and improve scheduling [2, 3, 5, 16]. However, only a few previous studies have focused on predicting the space availability as a transit reliability issue. Some works like [19] have studied forecasting bus passenger capacity in the whole urban bus transit system by integrating regression analysis with time-series. Utilizing bus smart card data and GPS data is also another method that has been proposed by [17] to predict the passenger flow in real time by finding the flow pattern that is most similar to the current estimation based on the Extended Kalman Filter model.

Among the works about forecasting bus passenger occupancy, Gayah et al. [8] has the most resemblance to our research. They have developed regression models to predict the real-time passenger occupancy for each bus-stop. However, their work is limited to only one bus route with 15 stops serving the Pennsylvania State University (PSU) University Park campus. We believe that the characteristics of each bus route and stop can be very different from other bus routes and stops. Therefore, one predictive model cannot be applicable for all the routes at all stops.

To the best of our knowledge there is no work that aims to predict bus crowding levels that are more understandable for commuters rather than the passenger occupancy. The accurate forecast of the crowding levels could improve the reliability of the transit system that helps riders make better decisions.

1.3 Our Contributions

This work makes the following contributions:

(1) We explain the extensive data preparation strategies employed over the real-world data received from the Port Authority of Allegheny County (Section 2).
(2) We formulate the bus crowding level prediction problem as an intuitive classification problem and develop appropriate models for prediction (Section 3).
(3) We perform experimental evaluation using real-world data and compare our proposed classification models to a baseline model (Section 4).

2 DATA PREPARATION

We have received two types of Pittsburgh-area bus data from the *Port Authority of Allegheny County*:

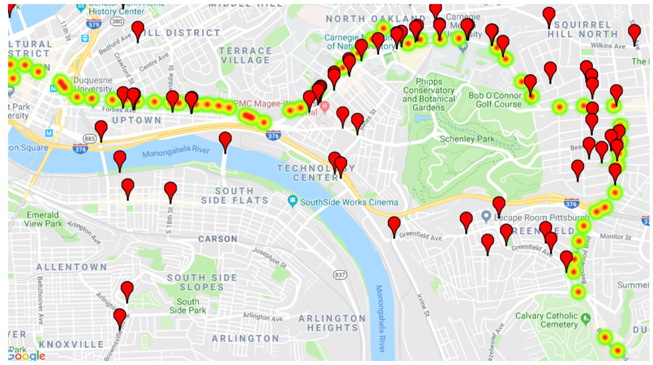


Figure 1: Missing stops for 61C inbound in April 2017 - The yellow-green path with orange dots represents the actual route for 61C inbound and the red markers show the locations where the latitude and longitude of the missing stops in data specify. As you can see many of the markers do not match the true locations of stops in this route during this specific time period.

- **Schedule Data** are given in GTFS format [7]; these contain the published bus schedules (i.e., are equivalent to printed bus schedules).
- **Historical Data** are given in a STEP [2] file format; these contain data about the exact time each bus arrives at a bus stop, along with how many people board or alight the bus. We convert the STEP file to a text standard format like CSV [3].

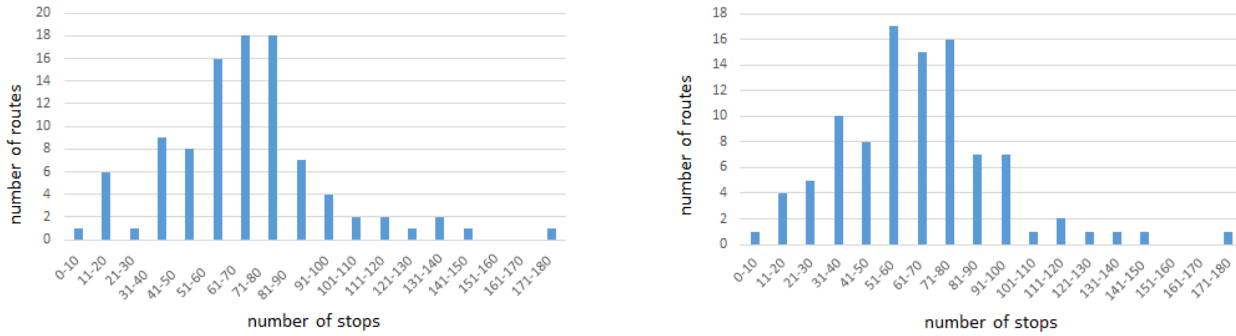
2.1 Data Selection

We only used one year worth of historical data for this study: March 2017 to March 2018. Each data record relates to a specific bus's boarding and alighting history at a specific stop. Additionally, this dataset has about 200 columns, but only a few were considered useful and included in the classification models. Table 1 shows a number of basic statistics about the data. In addition, the relationship between number of stops and routes in Pittsburgh, for both inbound and outbound directions, is indicated in Figure 2. As we can see in Figure 2a, about 55% of inbound routes have between 50 and 80 stops whereas only about 20% have more than 80 stops and nearly 25% have less than 50 stops. Almost the same pattern is observed in Figure 2b for outbound routes. More than half of the outbound routes have 50-80 stops while the other half have more than 80 or less than 50 stops in total. Routes 59 and O1 are two examples that have the highest and the lowest number of stops respectively.

2.2 Data Preprocessing

We first converted the selected data into a form that we could work with. That meant converting the STEP file into a text standard format like CSV. The next step is to detect data anomalies and correct or entirely remove them from the data. The following is

[2] STandard for the Exchange of Product
[3] Comma Separated Values



(a) inbound

(b) outbound

Figure 2: The relationship between number of stops and number of routes

the list of data inconsistencies we identified and removed before starting any data analysis:

- ***Invalid values***: we found out that there were some invalid characters such as a star(*) in some of the data which make their type incorrect. Theses values were removed or replaced by the correct ones after they were discovered.
- ***Missing records***: After comparing the available records in historical data and schedule data, we found out that the data coverage is about 80 percent; we decided to use the existing data for the next phases without imputing the missing records.
- ***Missing values***: During data analysis we found out that bus stop information (Stop ID and Stop Name) was unidentified in a number of data records which means they would be useless for the subsequent analysis and modeling. Therefore, such data records were evicted from the data (Figure 1).
- ***Duplicate records***: we observed that for some of the records, there was at least another copy which contained the same features such as date, bus, trip and stop just like that record, but the copy(s) was different in other features. Our hypothesis was that when a bus driver dwells at a stop behind a red light, he/she probably opens bus' doors to board and/or alight passengers more than once which leads to creating such duplicates in data. Tracking a few example of this scenario proved that our assumption is true up to a certain level. Such records were also eliminated to make more consistency for the following analysis.

2.3 Data Transformation

To prepare the preprocessed data for the machine learning models that we will apply in the next section, we need to perform the following transformations:

- ***Attribute Decomposition***: The date and time features need to be split into their constituent parts before they can be used by the machine learning models. We decomposed date and time from each data instance into month of the year, hour, and minute respectively.
- ***Encoding Categorical Attributes***: One task of data transformation is converting categorical data into numeric data. One of the methods for this conversion is to create dummy variables for all categorical attributes which in our case include month of the year, day of the week and time of the day.
- ***Adding new features***: Because of our modeling needs, we had to add two different kinds of features to the preprocessed data:
  - Features obtained from a secondary data source: Weather is one of the important features that can affect the crowding level in public transportation. We used weather data including average temperature, rainfall and snowfall per hour from Pennsylvania State Climatologist [6] and National Weather Service Climate of Pittsburgh [14] and integrated these features into our data.
  - Features obtained from original data: Some of the features we need for the modeling such as type of a bus, load of a bus at previous stops and the current crowding level were constructed from other features and/or other data instances and then were added to the preprocessed data.

3 MODELING FRAMEWORK

3.1 Required Features

As mentioned before, the main goal of this research is to predict bus crowding levels. Crowding levels can be defined based on the Load Factor which is the ratio of the current number of passengers on bus i and its maximum seating capacity (Equation 1).

$$LoadFactor_i = \frac{number\ of\ current\ passengers\ on\ bus_i}{maximum\ seating\ capacity\ of\ bus_i} \tag{1}$$

However, since the crowding level is going to be considered as the target feature (dependent variable) in the classification models, we need to assign appropriate levels for the obtained values of the load factor. Towards this, we defined five different crowding levels, after consultation with the Port Authority, as follows:

- **CL1:** many seats available (if $loadfactor < 0.5$)
- **CL2:** a few seats available (if $0.5 <= loadfactor < 0.8$)
- **CL3:** a few people standing (if $0.8 <= loadfactor < 1.1$)
- **CL4:** many people standing (if $1.1 <= loadfactor < 1.4$)

Table 2: Feature Sets to be used in classification models

Features	FS1	FS2	FS3	FS4	FS5	FS6	FS7	FS8	FS9
TOD2 - TOD96	✓	✓	✓	✓	✓	✓	✓		
DOW2 - DOW5		✓	✓	✓	✓	✓	✓		
MOY2 - MOY12		✓	✓	✓	✓	✓	✓		
BusType		✓	✓	✓	✓	✓	✓		
Temperature, Rainfall, Snowfall		✓	✓	✓	✓	✓	✓		
PLoad1 - Pload5			✓					✓	
PLoad1 - Pload10				✓					✓
PLoad5					✓				
PLoad10						✓			
PLoad5 - Pload10							✓		

Table 3: Independent variables descriptions

Variable	Description
TOD	96 variables for time of the day (each 15-min time-step)
DOW	5 variables for day of the week (only weekdays)
MOY	12 variables for month of the year
BusType	one variable (if the bus is single or double)
Temperature	one variable for average temperature per hour
Rainfall	one variable for average rainfall per hour
Snowfall	one variable for average snowfall per hour
PLoad1 - 10	10 variables for bus loads in the 10 previous stops. PLoad1 is the bus stop immediately before the one we are predicting for.

Table 4: Selected route-stop pairs for modeling

Cluster	Route	Stop Name
CL1	61B	FIFTH AVE AT BIGELOW BLVD
	71D	FIFTH AVE AT THACKERAY AVE
CL2	12	ANDERSON ST AT GENERAL ROBINSON
	56	GREENFIELD AVE AT IRVINE ST
CL3	Y1	E CARSON ST OPP STATION SQUARE STATION
	28X	LIBERTY AVE AT GATEWAY 4
CL4	P1	EAST BUSWAY AT NEGLEY STATION A
	G31	WEST BUSWAY AT INGRAM STATION C
CL5	P1	EAST BUSWAY AT NEGLEY STATION C
	61C	FORBES AVE AT BEELER ST

- **CL5:** crushed (if $load factor >= 1.4$)

Furthermore, the input features (independent variables) that can impact bus crowding are selected carefully and included in the models. Some of the features are categorical which were converted to dummy variables and some are numerical. Descriptions of the independent variables [4] are provided in Table 3.

3.2 Feature Sets

To have a broader perspective about how classification models perform with different sets of features, we defined 9 sets from the input features mentioned in the previous section. These sets and their selected features are represented in Table 2.

3.3 Clustering Routes-Stops

There are around 100 different routes on two different directions (inbound and outbound) per route and about 7000 stops in Pittsburgh's bus transit system. Since each route at each stop has usually different specifications such as number of buses, number of stops, time schedule, etc., having one separate model for each route-stop pair will result in more accurate prediction rather than having only one model for one route at all of its stops. Therefore, we need to filter out the data for each route-stop pair from the original one-year-long data. The number of such pairs is about 12,000 which means we will end up with about 12,000 datasets and therefore about 12,000 models. Although creating all models is essential for the final PittSmartLiving application which will be used by commuters, it is still ongoing work; for this paper we only select a few route-stop pairs as representative route-stop pairs.

In order to identify representative route-stop pairs, we partitioned routes-stops in the dataset using their "most common" crowding levels. That gave us five different clusters of routes-stops that can be defined as below (using the same names as each corresponding crowding level):

$$CL_i : route\text{−}stop\ pairs\ in\ cluster_i\ whose\ most\ common\ crowding\ level\ is\ CL_i\ ,\ where\ i \in \{1, 2, 3, 4, 5\}$$

For each of the five CL_i clusters, we selected the top two route-stop pairs, that have the highest number of records normalized by the number of their stops. We picked these from each cluster as representatives of all the routes-stops in the same cluster. After that, we filtered the relevant records for each pair out of the one-year-long dataset and stored them in separate CSV files. You can see the list of selected routes-stops in Table 4.

3.4 Train-Test Split

We randomly selected 80% of each preprocessed and transformed dataset for each route-stop pair to be used as our training data for our models. The remaining 20% of each dataset was used as test data for evaluation.

[4]The number of dummy variables for each categorical feature is reduced by one in order to be used in a model

3.5 Classification Models

As described earlier, we view predicting bus crowding levels to be a multinomial/multiclass classification problem. Given the set of independent variables we have, we employed a ***Multinomial Logistic Regression Model*** and a ***Random Forest Classifier***. Thus, for each pair of route-stop we fitted a separate model using the relevant training set and then predicted the crowding level using the test set.

The LogisticRegression class in Python offers two regularization schemes (L1 and L2) and four optimizers: newton-cg, lbfgs, liblinear, and sag [11]. Among these, newton-cg with L2 regularization produces models with higher prediction accuracy. On the other hand, we used RandomForestClassifier with 500 trees and maximum depth 100 which were selected after tuning the parameters. In spite of the fact that Logistic Regression performed almost as well as Random Forest on our data, we only report the outcomes from Random Forest which usually produces highly accurate predictions, limits over-fitting and therefore yields more useful results [4].

3.6 Baseline

The simplest model that we can propose as a baseline is a model with average loads. In this baseline, we compute the average load for each route-stop, for every 15-minute interval of a day, over the one-year-long data. Figure 3 illustrates the average load, obtained from the baseline, for 61C which is one of the busiest routes in Pittsburgh. In this heatmap, the x-axis represents 61C's stops in geographical order only in one direction (inbound, i.e., to the downtown), the y-axis shows 15-minute time intervals of a day and the color scale indicates the measured expression value of the average load. As one can see, the average load dramatically increases during the rush hours in the morning between 7 and 10 at some specific stops in Oakland where the University of Pittsburgh campus is located. It is not surprising because many University of Pittsburgh students, faculty, and staff take this route and the similar ones to get to campus in the morning.

Besides the average load, we should also assign a crowding level to each record in baseline. This can be done using the number of times when load factor is within a specific interval (as explained in the previous section). For example, if we count the number of times when the load factor is greater than 1.4 for a record, and this number is higher than the number of times when the load factor is less than 1.4, then we assign *CL5* as the crowding level for that record. Having a baseline that is created this way, we can evaluate our classification models by comparing their performance with the baseline's performance.

4 EVALUATION

Our goal in our evaluation was two-fold:

- Determine the usefulness of the different feature sets in predicting bus crowding levels, and
- Evaluate the performance of the proposed classification model compared to the baseline.

We have used 20% of each dataset as test data, for model evaluation. In particular, we fed the models with the test data and let them predict the corresponding crowding levels and their uncertainties. To qualify the performance of the classifiers and the baseline, we used two metrics including Log Loss and F1 Score, which we explain next.

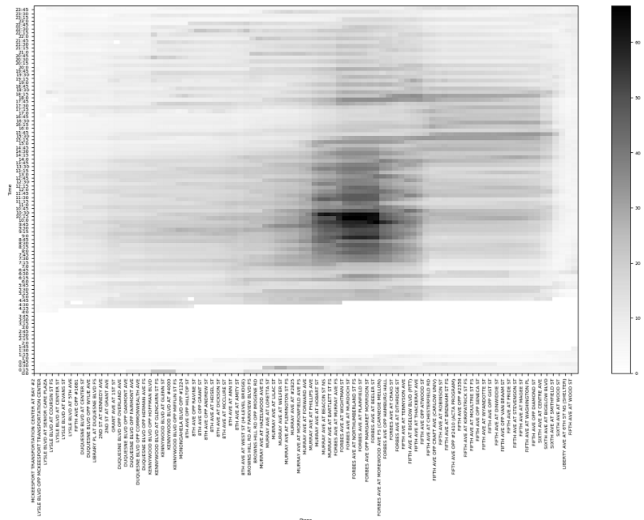


Figure 3: The Spatiotemporal average load for 61C-inbound using the baseline data.

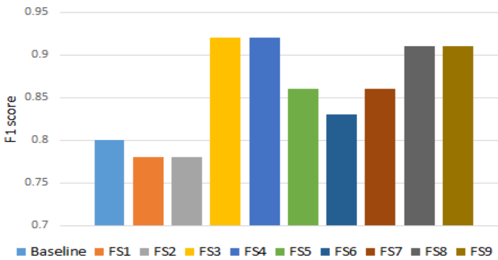
4.1 Metrics

We have chosen two performance metrics namely Log Loss and F1 score to evaluate the predictions coming from the baseline and the Random Forest models. Log Loss is a measure of how good probability estimates are (also known as cross entropy) [9]. The F1 score is defined as the harmonic mean of precision and recall and is known to be more useful than accuracy if there is class imbalance in classification [20]. Since predicting the probabilities of crowding level will be as useful as predicting the crowding level itself, we used log loss as one of the performance metrics. Furthermore, due to the phenomenon of class imbalance in crowding levels, we decided to use the F1 score with micro-averaging that aggregates the contributions of all classes to compute the average metric [12].

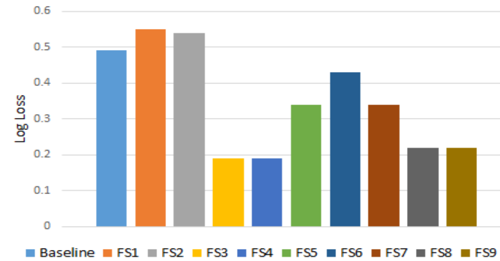
4.2 Results (Fig 4-6)

We summarize the models' evaluation by representing the log-loss and F1 score values for baseline and Random Forest regression models. Figures 4a to 4j illustrate F1 score and log loss for Baseline versus Random Forest models with all 9 different feature sets for 61B, 12, Y1, P1 in CL4 and P1 in CL5. Figures 5 and 6 demonstrate F1 score and log loss for Baseline versus only the best Random Forest models with specific feature sets including FS3, FS4, FS8 and FS9 for all selected route-stop pairs.

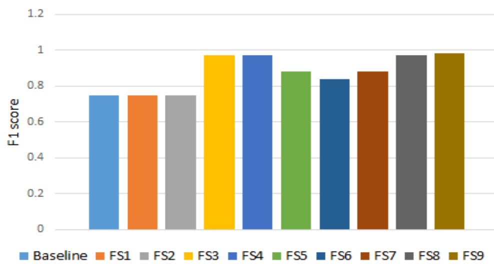
According to the histograms in Figure 4, models with FS3 or FS4 feature sets perform better than the baseline and the other models, in terms of both log loss and F1 score. Models with FS8 and FS9 also perform well and their metric values are very close to models with FS3 and FS4. For instance, Figure 4a shows that F1 score has increased by 15% from baseline to models with FS3 or FS4 feature sets. As it can be seen in Figure 4b this improvement is even more noticeable in log loss which is about 61% decrease from baseline to



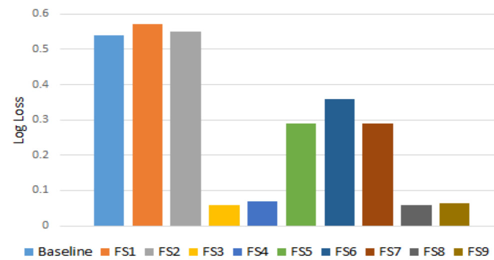
(a) F1 Score histogram for 61B



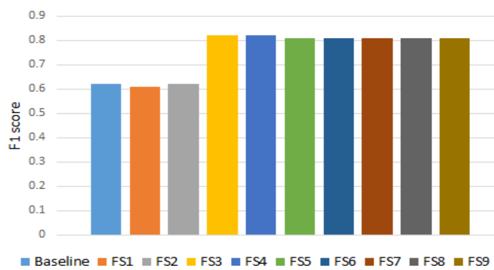
(b) Log Loss histogram for 61B



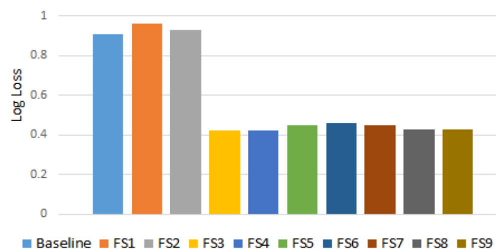
(c) F1 Score histogram for 12



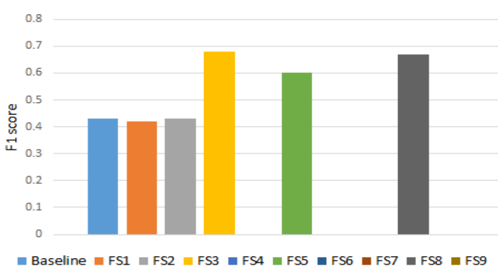
(d) Log Loss histogram for 12



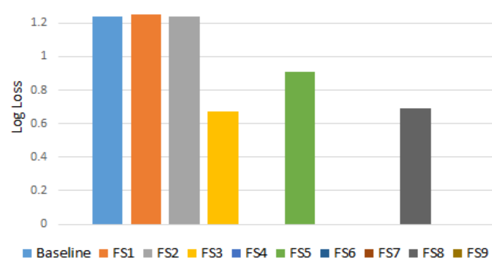
(e) F1 Score histogram for Y1



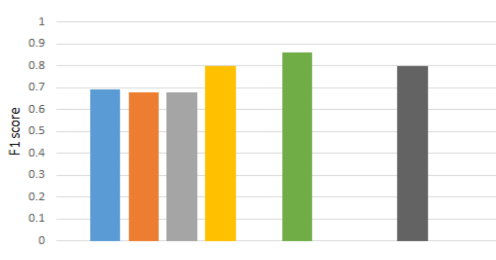
(f) Log Loss histogram for Y1



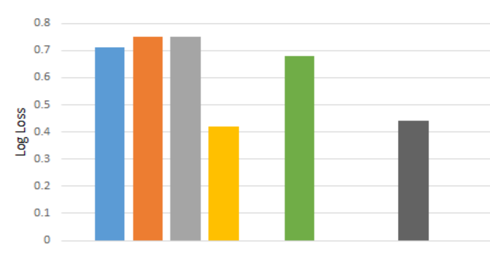
(g) F1 Score histogram for P1 (CL4)



(h) Log Loss histogram for P1 (CL4)



(i) F1 Score histogram for P1 (CL5)



(j) Log Loss histogram for P1 (CL5)

Figure 4: F1 Score and Log Loss histograms

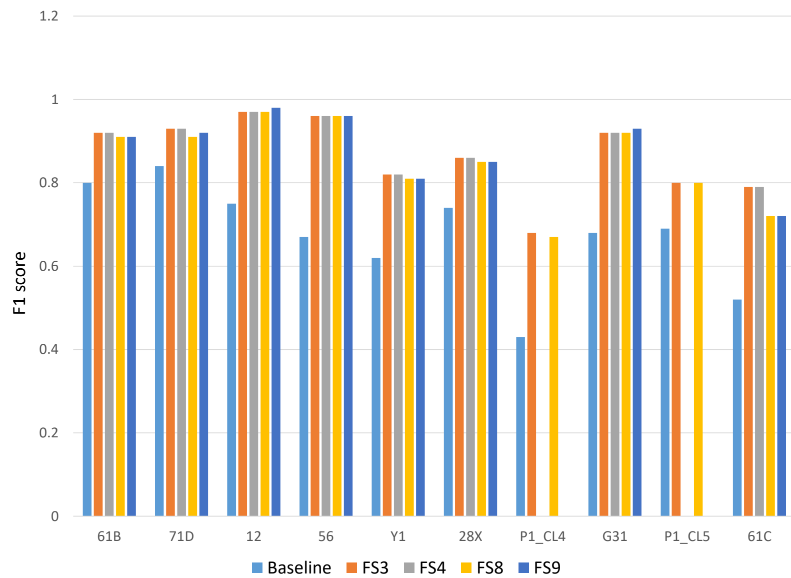


Figure 5: F1 Score histogram for all selected routes - best models only

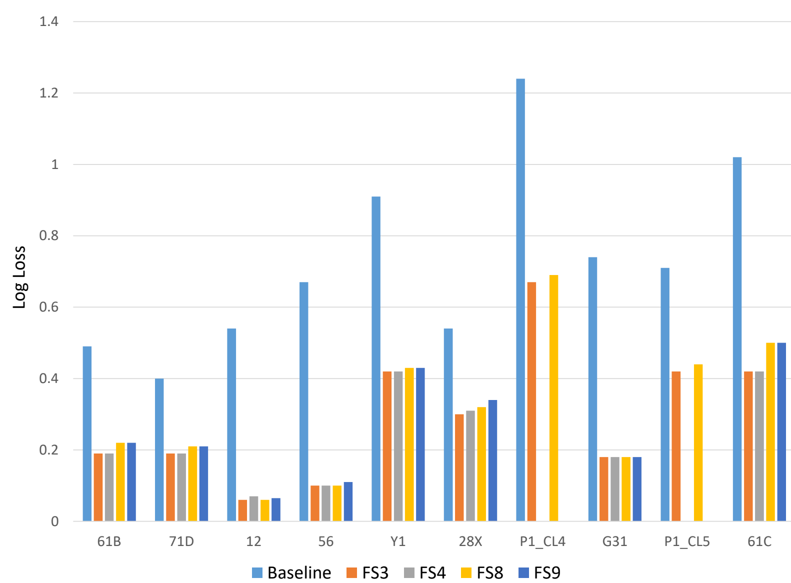


Figure 6: Log Loss histogram for all selected routes - best models only

the models with FS3 or FS4. There is only one exception here that can be observed in Figure 4i for route P1 in CL5. Although F1 score for this route at EAST BUSWAY AT NEGLEY STATION C, with FS3 feature sets has grown by 16% compared to its baseline model, it is about 7% lower than the F1 score for model with FS5. However, this exception can be overlooked because as one can see in Figure 4j, log loss value for this route-stop in model with FS3 is 62% lower than model with FS5.

The same pattern can also be seen in Figures 5 and 6 which show F1 score and log loss for a few models including baseline and the best-performing Random Forests for all selected route-stop pairs. In both figures, models with FS3 and FS4 outperform models with FS8 and FS9 in terms of F1 score and log loss. Our other observation from these histograms is that route-stop pairs that are in clusters CL1 and CL2 , have lower log loss and higher F1 score compared to the other pairs in other clusters. One interpretation is that more crowded routes-stops have “messier” data which leads to less accurate models.

As you may also have noticed, histograms FS4 and FS9 are blank for both P1s. The reason is that this route, which is the busiest route in Pittsburgh, has only 15 stops in each direction which means there are not 10 stops before these specific selected stops to be considered in modeling.

Takeaway: Random Forest Classifiers with feature set FS3/FS4 (i.e., including time of day, day of week, month of year, bus type, temperature, rainfall, snowfall, plus the bus loads for the 5 or 10 prior bus stops) produce the best predictions for bus crowding levels.

4.3 Discussion

In our evaluation, we identified feature sets FS3/FS4 to be the best choice among all other feature sets. However, these sets have more than 100 features. In real-life, it may be preferable to consider the trade-off between model quality and model complexity. As such, another good option may be FS8 which only contains that last 5 prior loads and performs almost as good as the more complicated FS3/FS4 sets.

Taking a step back, one may wonder: *if live bus loads are available, does it still make sense to do modeling?* The answer is yes: ideally you want to combine live data with predictive models so that you are able to more accurately predict bus crowding levels a few stops away. If you just know the bus crowding level from the stop right before you, you will have very little time to do something about it. Ideally you need to be able to plan ahead, for the maximum benefit.

5 CONCLUSION AND FUTURE WORK

In this work, we framed the ”how full will my next bus be?” question as a classification problem and developed a modeling framework to predict bus crowding levels in Pittsburgh. Our results showed that the proposed framework (using Random Forest Classifiers) performs very well when using time of the day, day of the week, month of the year, bus type, weather, and the bus loads from the 5 or 10 prior stops as the selected features. Although we developed our modeling framework using only data from Pittsburgh, we are confident that the same process and the proposed models can be applied on data from other cities.

As part of our future work we intend to evaluate the framework when it is trained to forecast the crowding levels for all existing routes-stops with appropriate accuracy. Furthermore, we aim to deploy these models as part of our PittSmartLiving mobile application, using live real-time data, with the ultimate goal of improving the commuters’ quality of life (through better, actionable information).

ACKNOWLEDGMENTS

This work is part of the PittSmartLiving project which is supported by NSF award CNS-1739413.

REFERENCES

[1] Marco Balduini, Marco Brambilla, Emanuele Della Valle, Christian Marazzi, Tahereh Arabghalizi, Behnam Rahdari, and Michele Vescovi. 2018. Models and Practices in Urban Data Science at Scale. *Big Data Research* (Aug. 2018). https://doi.org/10.1016/j.bdr.2018.04.003

[2] Yu Bin, William HK Lam, and Mei Lam Tam. 2011. Bus arrival time prediction at bus stop with multiple route. *Transportation Research Part C: Emerging Technologies* 19, 6 (Dec. 2011), 1157–1170.

[3] Yu Bin, Yang Zhongzhen, and Yao Baozhen. 2006. Bus arrival time prediction using support vector machines. *Journal of Intelligent Transportation Systems* 10, 4 (Dec. 2006), 151–158.

[4] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (Oct. 2001), 5–32.

[5] Steven I-Jy Chien, Yuqing Ding, and Chienhung Wei. 2002. Dynamic bus arrival time prediction with artificial neural networks. *Journal of Transportation Engineering* 128, 5 (Sept. 2002), 429–438.

[6] Pennsylvania State Climatologist. 2018. PASC IDA Data. http://climate.psu.edu/data/ida/

[7] Google Developers. 2016. General Transit Feed Specification (Static Overview). https://developers.google.com/transit/gtfs/

[8] Vikash V Gayah, Zhengyao Yu, and Jonathan S Wood. 2016. Estimating uncertainty of bus arrival times and passenger occupancies. (2016).

[9] Irving John Good. 1952. Rational Decisions. *Journal of the Royal Statistical Society. Series B (Methodological)* 14, 1 (1952), 107–114.

[10] Moovit Inc. 2012. MoovIT. https://moovitapp.com

[11] Scikit learn Developers. 2007-2018. scikit-learn models. https://scikit-learn.org/

[12] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schutze. 2008. Introduction to Information Retrieval? Cambridge University Press 2008. *Ch* 20 (2008), 405–416.

[13] University of Pittsburgh. 2016. Pitt Smart Living Project. Retrieved March 13, 2019 from https://pittsmartliving.org/

[14] Pittsburgh Weather Forecast Office. 2018. National Weather Service Climate of Pittsburgh. https://w2.weather.gov/climate/xmacis.php?wfo=pbz

[15] Joseph N Prashker. 1979. Direct analysis of the perceived importance of attributes of reliability of travel modes in urban travel. *Transportation* 8, 4 (Dec. 1979), 329–346.

[16] Amer Shalaby and Ali Farhan. 2004. Prediction model of bus arrival and departure times using AVL and APC data. *Journal of Public Transportation* 7, 1 (2004), 3.

[17] Amer Shalaby and Ali Farhan. 2017. A real-time passenger flow estimation and prediction method for urban bus transit systems. *IEEE Transactions on Intelligent Transportation Systems* 18, 11 (Nov. 2017), 3168–3178.

[18] Transit. 2012. TransitApp. https://transitapp.com

[19] Wen xia Sun, Ti Song, and Hai Zhong. 2009. Study on bus passenger capacity forecast based on regression analysis including time series. In *2009 International Conference on Measuring Technology and Mechatronics Automation*, Vol. 2. IEEE, 381–384.

[20] Dell Zhang, Jun Wang, and Xiaoxue Zhao. 2015. Estimating the uncertainty of average F1 scores. (2015).