Dynamic Origin-Destination Travel Time and Traffic Flow Estimation through Heteroskedastic Poisson Processes and Machine Learning

Eric Lin

Abstract

Public transit systems around the world are gaining ridership as a conveniently accessible and ecologically friendly mode of transportation. However, aging infrastructure has led to increasing delays in trip travel times and variances in traffic flow. Accurate forecasts of origin-destination (OD) movement data is exceptionally beneficial for public transit system passengers and authorities. Passengers can avoid traffic congestion and authorities can anticipate and prevent network bottlenecks.

We present a micro-prediction, generative heteroskedastic poisson-distributed approach to predict origin-destination (OD) traffic flow for long term predictions. This resolves the deficiencies current approaches experience in long term prediction. Furthermore, we fuse the traffic flow estimation with machine learning regression models to forecast individual passenger trip times.

Experimental evaluation shows that the proposed model for OD traffic flow provides a robust solution to the long-term deficiencies of baseline and previous Bayesian-based approaches, particularly when predicting further into the future. The model for OD individual trip duration also surpasses baseline forecasting approaches, achieving an average absolute error of 4.5 minutes. This is a vast improvement on the performance of commercial vendors such as Google Maps which achieve an average absolute error of 7.8 minutes. These tests show that the proposed model can be successfully implemented for a wide array of OD movement data for smart city planning.

1 Introduction

In recent years, public transit has gained traction as a cost effective method for transportation. Public systems such as subways or buses offer people who travel in metropolitan areas an easy means of transit. Furthermore, public transit vastly alleviates road traffic congestion as well as the metropolitan area's carbon footprint. In particular for the United States, millions of tourists and about 7.7 million workers produce over 10.8 billion public transit trips [1]. American public transport ridership has increased by 64% over the past two decades [2]. It substantially outpaces the population growth, 21%, during the same time period [3].

However, current public transit infrastructure in the United States lags seriously behind other countries. The increasing stresses imposed by rising public transit demand has resulted in ever larger delays. Nationwide reports have revealed consistent under-investment in infrastructure [4]. This has resulted in overfilled network capacities and breakdown of equipment. Thus, passengers riding public transit experience increasingly larger delay times and are more at risk due to equipment failure such as a subway derailment. Proposed solutions to these eminent infrastructure problems center around the idea of efficiently allocating new resources to the public transit network.

The rise of big data analytics has led to the advent of smart cities. Operating in this technologically advanced and data rich age, city traffic planners are increasingly looking towards traffic flow and trip travel time data analytics as a means to gain insight on optimal allocations of resources. This then necessitates a model that accurately predicts traffic flow and trip travel times for origin-destination spatio-temporal data.

In this paper, origin-destination spatio-temporal data refers to any dataset that consists of information on individual passengers' location (spatial data) at certain times (temporal data). In the case of many modes of public transit such as bike-share and subway networks, this spatio-temporal information is limited to two points for each passenger: 1) the location and place the passenger entered the network (origin) and 2) the location and place the passenger exited the network (destination).

The discrete nature of origin-destination databases is a crucial difference between estimation of public transit networks and private transit. Whereas private transit consists of a set of data-points that is continuous both spatially and temporally, a public transit network has fixed station nodes. Traffic flow for origin-destination databases is then regarded as the large-scale passenger in and out flow at these fixed station nodes. Furthermore, trip travel times intuitively refer to the length of time it takes for an individual passenger to travel from her origin station to her destination station.

This work aims to produce accurate predictions of origin-destination traffic flow as well as accurate trip travel time predictions for individual passengers. To accomplish this, the assumption is made that real-time information of individual passengers is made available to the model. This is a valid assumption since most origin-destination networks, such as the Washington D.C. metro subway system, real-time data is streamed through a web interface [5].

Accurate estimations of traffic flow and trip travel times for origin-destination data leads to a profusion of applications. Using accurate traffic forecasts, people may decide to change their plans. For instance, if the model predicts overcrowded trains and heavy trip time delays, they may expedite their schedule to enter the subway station earlier than usual in order to dodge projected future traffic congestion. The model's estimations are also invaluable for public transit network managers. From the perspective of the network, managers can use traffic flow and travel time estimations to anticipate locations that develop traffic congestion. With this insight, network controllers can then take actions to prevent the traffic congestion. In a public transit system, controllers may move subway trains, buses, and taxis from low-traffic to high-traffic locations. Finally, more accurate origin-destination forecasting may lead to insights in private transit estimation. Each time a subway train stops at a station, the surrounding road traffic may be heavily influenced by the outflow of passengers. Thus, accurate predictions of train arrivals and passenger outflow can lead to improved road traffic estimation.

The arrangement of the remainder of this paper is as follows. Related work is explored in Section 2. The problem of origin-destination trip time and traffic flow estimation is established in Section 3. Our proposed prediction approach uses Bayesian learning traffic flow prediction, Poisson distribution future passenger outflow prediction, and machine learning trip travel time estimation. This is described in detail in Section 4. Section 5 shows the validity of our proposed model through experimental evaluation.

2 Related Work

A plethora of research has been conducted on forecasting origin-destination data. There have been two main subcategories in past research – one is the prediction of traffic flow (how many people enter and exit a station or network) and the second is the estimation of origin-destination travel times for individual users.

Although my work centers around the research of the second kind in that the end goal is to estimate individual origin-destination travel times, the significant portion of my proposed model utilizes a novel solution to the first task of predicting traffic flow. Research on traffic flow is well established for vehicular movement. Past inquiries [6, 7, 8] used GPS data to accurately predict future vehicle movement and thus was able to predict traffic flow. However, their approaches require a continuous set of spatio-temporal data which is seriously lacking in this paper's scenario. By its definition, origin-destination data only consists of two spatio-temporal data points. Therefore, past approaches used for vehicle movement is not feasible for origin-destination transportation such as subways, bike-shares, and other modes of public transportation.

Current attempts for modeling traffic flow within origin-destination systems have conventionally went through a macro-picture approach. [9, 10] propose autoregressive moving average models (ARMA) to utilize historical data to project future traffic flow. A more successful model is presented by [11]. They build a multi-layer machine learning model founded on large-scale patterns. The algorithm consists of using clustering techniques to group stations with similar network flow. While impressive for bike-share data, their model is not directly applicable for subway origin-destination data. Prediction of large scale public transit such as subway networks is intrinsically different from estimations of bike-share stations. Their proposed bipartite clustering performs well because it is assumed that users can relatively easily move from one bike-share station to another. Thus, the clustering on both geographic location and time series characteristics is optimal when users have freedom in choosing which station they return their bike to. On the other hand, it is not feasible for passengers to easily trasition from one station to another. Thus, clustering stations based on geographic location is less valuable and not justifiable.

Traditional methods for the problem of individual origin-destination travel time estimation often utilize machine learning techniques. [12] employs clustering techniques to find similar passengers. They then build a backpropagation neural network (BNN) to predict types of passengers. They cleverly propose a bag-of-words model, drawing on techniques from Natural Language Processing (NLP). Their solution involves treating trips as words and passengers as text. This allows them to classify and cluster similar passengers. From these classifications, a new passenger can be then predicted to follow similarly to a historical passenger's pattern and the algorithm can predict the travel time.

3 Data and Problem Definition

This paper assumes the existence and availability of a database containing *trip information*, as it is commonly recorded in subway and bike-sharing networks. A trip only consists of spatio-temporal origin and destination data-points but lacks any data regarding the trip path taken in-between.

Definition 3.1 (Historical Origin-Destination Trip Database). *Let $\mathcal{S}$ be a finite database of stations where passengers can enter and leave an origin-destination transportation network and $\mathcal{T} \subseteq N$ be a discrete time space. A* historical origin-destination trip *is a quadruple $(S_o \in \mathcal{S}, S_d \in \mathcal{S}, t_o \in \mathcal{T}, t_d \in \mathcal{T})$. The two stations S_o and S_d are respectively denoted as origin and destination stations of a passenger. The discrete times $t_o \leqslant t_d$ denote when the passenger enters S_o and exits S_d, respectively. We assume a database, i.e., a collection of historical origin-destination trips $\mathcal{D}$.*

In most origin-destination public transit networks, the data does not record a passenger's destination until the passenger exits the system. Let the database of trips with no destination at any given point in time be denoted as *real-time origin trips.*

Definition 3.2 (Real-Time Origin Trip Database). *If a passenger has entered S_o but has not yet exited S_d, i.e., if at current time t_i, $t_o \leqslant t_i < t_d$ holds, then destination s_o and arrival time t_o are unknown. We let $\mathcal{D}_i$ denote a set of current trips. Note that for each trip $tr \in \mathcal{D}_i$, S_d and t_d are unknown. That is, $S_d = t_d = \perp$.*

In the operation of a public transit network, it is very likely for passenger traffic flow and individual trip travel time to be influenced by weather factors. Let this set of meteorological information be denoted as the *Meteorology Database.*

Definition 3.3 (Meteorology Database). *Let $M_\tau = (p_\tau, h_\tau, d_\tau, v_\tau, r_\tau, e_\tau)$ denote the meteorological features of a time period τ. $p_\tau, h_\tau, d_\tau, v_\tau, r_\tau, e_\tau$ respectively denotes the average temperature, average humidity, visibility distance, wind speed, precipitation, and meteorological events such as Thunderstorm or Snow.*

The previously described three databases make up the input data of the estimation models. The following definitions establish the traffic flow and trip time estimations.

Definition 3.4 (Outflow). *For an individual station $S \in \mathcal{S}$ and a single discrete time $t \in \mathcal{T}$, we let $outflow(S,t)$ denote the outflow of passengers at station S at time t, that is, the number of passenger leaving S at time t. Formally:*

$$outflow(S,t) := |\{tr \in \mathcal{D} | tr.S_d = S \wedge tr.t_d = t\}|.$$

Definition 3.5 (Outflow Prediction). *Given a set of real-time origin trips $\mathcal{D}_i$, a station $S \in \mathcal{S}$, and a future time $t > t_i$, let $outflow_i(S,t)$ denote the predicted outflow of passengers in $\mathcal{D}_i$ at station S at time t.*

Definition 3.6 (Trip Time). *For an individual trip $tr \in \mathcal{D}$ at time $t \in \mathcal{T}$, let $duration(tr, t)$ denote the trip time duration of an individual passenger at time t, that is, the number of minutes it takes for the passenger to travel from the origin station to the destination station at time t.*

Definition 3.7 (Trip Time Prediction). *Given an individual real-time origin trip $tr \in \mathcal{D}_i$ at time $t \in \mathcal{T}$, let $duration_i(tr, t)$ denote the predicted outflow of passengers in $\mathcal{D}_i$ at time t.*

Up to this point, the model is accurate in short time-span projections. However, the estimations fall off in accuracy as the model predicts further into the future, and all passengers in the real-time origin trip database are projected to exit the network. This deficiency is accounted for with the introduction of inflow estimation.

Definition 3.8 (Inflow). *For an individual station $S \in \mathcal{S}$ and a single discrete time $t \in \mathcal{T}$, we let $inflow(S, t)$ denote the inflow of passengers at station S at time t, that is, the number of passenger leaving S at time t. Formally:*

$$inflow(S, t) := |\{tr \in \mathcal{D} | tr.S_d = S \wedge tr.t_d = t\}|.$$

Definition 3.9 (Inflow Prediction). *Given a set of real-time origin trips $\mathcal{D}_i$, a station $S \in \mathcal{S}$, and a future time $t > t_i$, let $inflow_i(S, t)$ denote the predicted inflow of passengers in $\mathcal{D}_i$ at station S at time t.*

The following section presents our approach to compute $flow_i(S, t)$ and $duration_i(tr, t)$. In essence, we use the completed trip database $\mathcal{D}$ to train a Bayesian model to learn the distribution of trip destinations and, given a destination, the trip duration. We aggregate these distributions into expected outflow which we use for prediction. Then, we augment the outflow prediction with the inflow prediction. Finally, we use outflow prediction, historical data, and meteorological factors to forecast individual trip duration.

4 Methodology

In this section, we present our model for predicting traffic outflow of current passengers and trip time estimation.

For traffic flow, the approach is: For a given current passenger, we know the origin station and can use the set of completed trips to estimate the distribution of destination stations. If a passenger has been in the network for a long time, we intuitively know that short trips leading to spatially close destination stations are more unlikely. We employ Bayesian learning to adapt the probability distribution of destinations given the passenger's time in the network. To build this model, we assume a database of completed trips $\mathcal{D}$, which we use to empirically learn the distribution of destination stations given an origin station. For a given passenger, we use this distribution as an apriori distribution and further condition it to the passengers current time in the network.

4.1 Apriori Model

First, we define the apriori distribution of stations without any knowledge of travel time of an individual passenger.

Definition 4.1 (Apriori Station Distribution)**.** *Let $\mathcal{D}$ be a set of completed trips. Let $(M) \in R^{|\mathcal{S}|} \times R^{|\mathcal{S}|}$ be a stochastic $|\mathcal{S}| \times |\mathcal{S}|$ matrix such that any value $M_{i,j}$ corresponds to the empirical probability $P(tr.S_d | tr.S_o)$ of a random passenger trip tr having station j as their destination, given that their origin is station i. Formally,*

$$M_{i,j} := \frac{\{tr \in \mathcal{D} | tr.S_o = S_i \wedge tr.S_d = S_j\}}{\{tr \in \mathcal{D} | tr.S_o = S_i\}}.$$

Given an origin and destination pair, we can further learn the corresponding distribution of travel times from all observed travel times between these stations in $\mathcal{D}$.

Definition 4.2 (Apriori Travel Time Distribution)**.** *Let $\mathcal{D}^{time}_{i,j} = \{(tr.t_d - tr.t_o) | tr \in \mathcal{D} \wedge tr.S_o = S_i \wedge tr.S_d = S_j\}$ denote the multi-set of travel times of trips between stations S_i and S_j. We model these by a Gaussian distribution $time_{i,j} = N(\mu_{i,j}, \sigma^2_{i,j})$, where $\mu_{i,j} = \frac{\sum_{d \in \mathcal{D}^{time}_{i,j}} d}{|\mathcal{D}^{time}_{i,j}|}$ and $\sigma^2_{i,j} = \frac{\sum_{d \in \mathcal{D}^{time}_{i,j}} (d - \mu_{i,j})^2}{|\mathcal{D}^{time}_{i,j}| - 1}$.*

As an example, consider two stations A and B for which we have only obtained four trips in $\mathcal{D}$ which had the trip times (in minutes) of $\{2, 2, 5, 7\}$. In this case, we would model the trip time by a $N(4, 18)$ distribution, i.e., a normal distribution having a mean of 4 and a standard deviation of $\sqrt{18}$. Note that in this example, a normal approximation is extremely bad, since the number of samples 4 is as low as $n = 4$. However, in the dataset used in our experimental evaluation, we have thousands of trips between most station pairs. Using a single gaussian distribution, and following the Central Limit Theorem assumes that travel times are stationary, thus do not change over time. However, that assumption may not hold in general. For instance, travel times may follow one gaussian during rush-hour, when trains are running frequently, and may follow another gaussian otherwise, when trains are less frequent.

Using the apriori station distribution M (c.f. Definition 4.1), time distribution, and travel time distribution $time_{i,j}$ between two stations (c.f. Definition 4.2), we can describe the probability that a current passenger $u \in \mathcal{D}_i$ who entered the network at station S_i will exit at station S_j at time t. Aggregating these probabilities for all current passengers yields an apriori passenger flow.

Definition 4.3 (Apriori Passenger Flow)**.** *Let $\mathcal{D}$ and $\mathcal{D}_i$ respectively be a set of completed and current trips. We can predict the expected passenger outflow $flow^t(S_j)$ at station S_j at time interval t as:*

$$flow^t_{S_j} = \sum_{tr \in \mathcal{D}_i} M(tr.S_o, S_j) \cdot P(time_{i,j} \in t).$$

For this passenger flow to be unbiased, all passengers would have to have just entered the metro network. However, we know that some passengers may have been in the network for a while already. This information not only affects the arrival time but also affects the distribution of potential destination stations: If a passenger has already been in the metro for 30 minutes, then it becomes very unlikely that his destination is a station that only takes five minutes to get to from the passengers origin. The next subsection describes how we adapt the probability distribution individually for each passenger depending on their current time in the network.

4.2 Traffic Flow Model Adaption

For a current time t_i and individual current trip $tr \in \mathcal{D}_i$, we adapt the distribution of destination stations of tr based on the time the passenger has spent in the system. This observation is equivalent to stating the arrival time is greater than the current time t_i. The destination probabilities are adapted using the Theorem of Bayes:

$$P(tr.S_d = S_j | tr.S_o, tr.t_d > t_i) = \frac{P(tr.t_d > t_i | tr.S_d = S_j \wedge tr.S_o) \cdot P(tr.S_d = S_j | tr.S_o)}{P(tr.t_d > t_i | tr.S_o)}. \quad (1)$$

The condition of knowing the destination $tr.S_o$ is global, thus appearing in all conditions. On the right-hand-side of Equation 1, the probability $P(tr.t_d > t_i | tr.S_d = S_j \wedge tr.S_o)$ is the fraction of trips in $\mathcal{D}$ from S_o to S_j which take more than $t_i - t_o$ time. The prior $P(tr.S_d = S_j | tr.S_o)$ is the apriori probability of going to station S_j from S_o, found in matrix M (see Definition 4.1). Finally, the prior $P(tr.t_d > t_i | tr.S_o)$ is the probability of any trip starting at $tr.S_o$ to take at least $t_i - tr.t_o$ time. Using the Law of Total Probability [13], this term can be rewritten as $P(tr.t_d > t_i | tr.S_o) = \sum_{S \in \mathcal{S}} P(tr.S_d = S | tr.S_o) \cdot P(tr.t_d > tr.t_i | tr.S_o \wedge tr.S_d = S)$. Here, each summand can be obtained analogously to the first prior $P(tr.S_d = S_j | tr.S_o)$.

Given the adapted destination probabilities obtained from Equation 1, we can adapt the time distribution of a given destination in a similar way using Bayes. We adapt the probability $P(tr.t_d = x)$ that the trip takes exactly time x as follows.

$$P(tr.t_d = x | tr.S_o, tr.S_d, tr.t_d > t_i) = \frac{P(tr.t_d > t_i | tr.S_o, tr.S_d, tr.t_d = x) \cdot P(tr.t_d = x | tr.S_o, tr.S_d)}{P(tr.t_d > t_i | tr.S_o, tr.S_d)}. \quad (2)$$

Again, the conditionals $tr.S_o$ and $tr.S_d$ are global. The probability $P(tr.t_d > t_i | tr.S_o, tr.S_d, tr.t_d = x)$ is 1 if $x > t_i$ and equals 0 otherwise. The prior $P(tr.t_d = x | tr.S_o, tr.S_d)$ is taken from the apriori time distribution for trips from station S_o to station S_d as computed in Definition 4.2. Finally, the prior $P(tr.t_d > t_i | tr.S_o, tr.S_d)$ always equals 1 since the destination time must be larger than the current time (otherwise the passenger would have arrived already).

Using the adapted station distribution as computed in Equation 1 and using the adapted arrival time distribution computed in Equation 2, we can describe the probability that a current passenger $u \in \mathcal{D}_i$ who entered the network at station S_i will exit at station S_j at time t, given that u has already been in the station since $tr.d_o$. Aggregating these probabilities for all current passengers yields the adapted aposteriori passenger flow.

Definition 4.4 (Aposteriori Passenger Flow)**.** *Let $\mathcal{D}$ be a set of completed trips and let $\mathcal{D}_i$ be a set of current trips. Further, let $P'(tr.S_d) := P(tr.S_d = S_j | tr.S_o, tr.t_d > t_i)$ denote the* *adapted probability of trip tr ending at destination S_j as computed using Equation 1, and let $P'(tr.t_d \in t) \coloneqq P(tr.t_d \in t | tr.S_o, tr.S_d, tr.t_d > t_i)$ denote the adapted probability of trip tr ending in time interval t, assuming destination station S_j as computed in Equation 2. We can predict the adapted expected passenger outflow $flow^t(S_j)$ at station S_j at time interval t as follows:*

$$aflow^t_{S_j} = \sum_{tr \in \mathcal{D}_i} P'(tr.S_d) \cdot P'(tr.t_d \in t).$$

4.3 Extension of Passenger Prediction

Up until this point, the previously discussed Bayesian model produces a model which retains relatively high prediction accuracy for thirty minutes into the future. For practical use, the model requires constant live streaming of real-time data for passengers entering into the network. Thus, it's usefulness becomes negligible for any long term predictions. Here we propose a few possible solutions to extend the timeframe for high accuracy predictions.

One method is through the use of a heteroskedastic Poisson process. Using this, estimations of the inflow of passengers at each station at a certain point in time is fed into our Bayesian model. It can be seen that the derivation of the heteroskedastic Poisson process in fact points towards the historical inflow average as a theoretically solid source for estimation.

Another interesting avenue for research is the utilization of Markov chains to provide inflow estimation. Building off of predictions for previous time periods, Markov chains can be established to more accurately estimate traffic further into the future.

4.4 Trip Time Model Adaption

Accurate forecasts for Traffic flow allows us to develop a model for the estimation of individual passenger trip times. For a current time t_i and individual current trip $tr \in \mathcal{D}_i$, we input spatial and temporal features into machine learning regression models. We used three sets of features:

Definition 4.5 (S-T Feature Set). *For a given time $t_i \in \mathcal{T}$ and given trip starting at t_i, $tr \in \mathcal{D}_i$, let $\boldsymbol{ST} = (D_i, S_o, t_i, S_d, H_{tr})$ respectively denote the input features day type (day of the week, whether or not it is a holiday), origin station, trip start time, destination station, and historical trip duration.*

We then utilized the Meteorology Database to account for trip time variances due to weather.

Definition 4.6 (S-T + All Weather Feature Set). *For a given time $t_i \in \mathcal{T}$ and given trip starting at t_i, $tr \in \mathcal{D}_i$, let $\boldsymbol{STAW} = (D_i, S_o, t_i, S_d, H_{tr}, p_\tau, h_\tau, d_\tau, v_\tau, r_\tau, e_\tau)$ respectively denote the input features day type (day of the week, whether or not it is a holiday), origin station, trip start time, destination station, historical trip duration, average temperature, average humidity, visibility distance, wind speed, precipitation, and meteorological events such as Thunderstorm or Snow.*

Since some meteorological factors do not have pronounced impact on trip travel time and unnecessary features adds noise that cause certain machine learning models to underperform, We selected the most influential weather features and disregarded the rest of the weather features.

Definition 4.7 (S-T + Influential Weather Feature Set)**.** *For a given time $t_i \in \mathcal{T}$ and given trip starting at t_i, $tr \in \mathcal{D}_i$, let $\boldsymbol{STIW} = (D_i, S_o, t_i, S_d, H_{tr}, p_\tau, d_\tau, r_\tau, e_\tau)$ respectively denote the input features day type (day of the week, whether or not it is a holiday), origin station, trip start time, destination station, historical trip duration, average temperature, visibility distance, precipitation, and meteorological events such as Thunderstorm or Snow.*

Finally and crucially, we added two Flow Estimation features to each S-T set: historical average outflow for the destination station and the FPOP for the destination station. This allows the machine learning model to use Flow Estimation to anticipate whether any given trip will experience delays because of unprecedented traffic levels.

5 Experiments

We assessed our proposed traffic outflow and trip duration models on a dataset retrieved from the Washington D.C. metro network, Washington Metro Transport Authority [5]. The dataset consisted of passenger farecard data recorded in April, 2017. We utilized data from April 1, 2017 to April 24, 2017 as training data and April 25 at 7:20 AM, 2017 as testing data. The selection of April 25 at 7:20AM was achieved through a process of considering the want to test the model on a typical workday. As the last Tuesday in April, April 25 is devoid of holidays, is right before colleges' summer breaks, and gives an ample number of days for training. Since the morning rush hour for metro subway traffic is around 8:00AM, 7:20AM was selected as the starting time for our testing so our model can predict the peak traffic flow and travel times. The training data, consisting from April 1 to April 24, consists of 12,206,721 recorded farecard origin-destination trips from the 91 stations of the Washington DC metro.

Algorithms

We implemented a number of baseline traffic flow approaches for comparison. Using information on empirical station and time distributions, current times an individual passenger has been in the network so far, and adapting the apriori model using Bayesian learning as described in Section 4.2, our main baseline approach is denoted as *Bayesian Model (BM)*. As an adaption, we propose to preprocess the data by removing outliers. The reason is some trips which usually only takes a few minutes may take many hours for some passengers. These trips may correspond to people working inside the metro. A single such outlier drastically changes the fitted normal distribution, thus creating a bias towards extremely long, but also (due to symmetry of gaussians) extremely short trips. We denote this approach that removes outliers as *BM+O*. Note that BM and $BM + O$ estimate the same destination stations and only differ in how they model arrival time.

To compare different flow prediction models, we utilize two comparisons. The first is network outflow estimation. The second comparison is an aggregation of the prediction error, i.e., the absolute difference between prediction and observed ground-truth, for all stations and for all five-minute intervals. We combine the error of all stations by taking the root mean squared error (RMSE):

$$RMSE = \sqrt{\sum_{S \epsilon \mathcal{S}} \left(\sum_{t \epsilon \mathcal{T}} pflow_S^t - GT_S^t \right)^2},$$

where $pflow_S^t$ is predicted flow at t and S, and GT_S^t is corresponding ground truth flow.

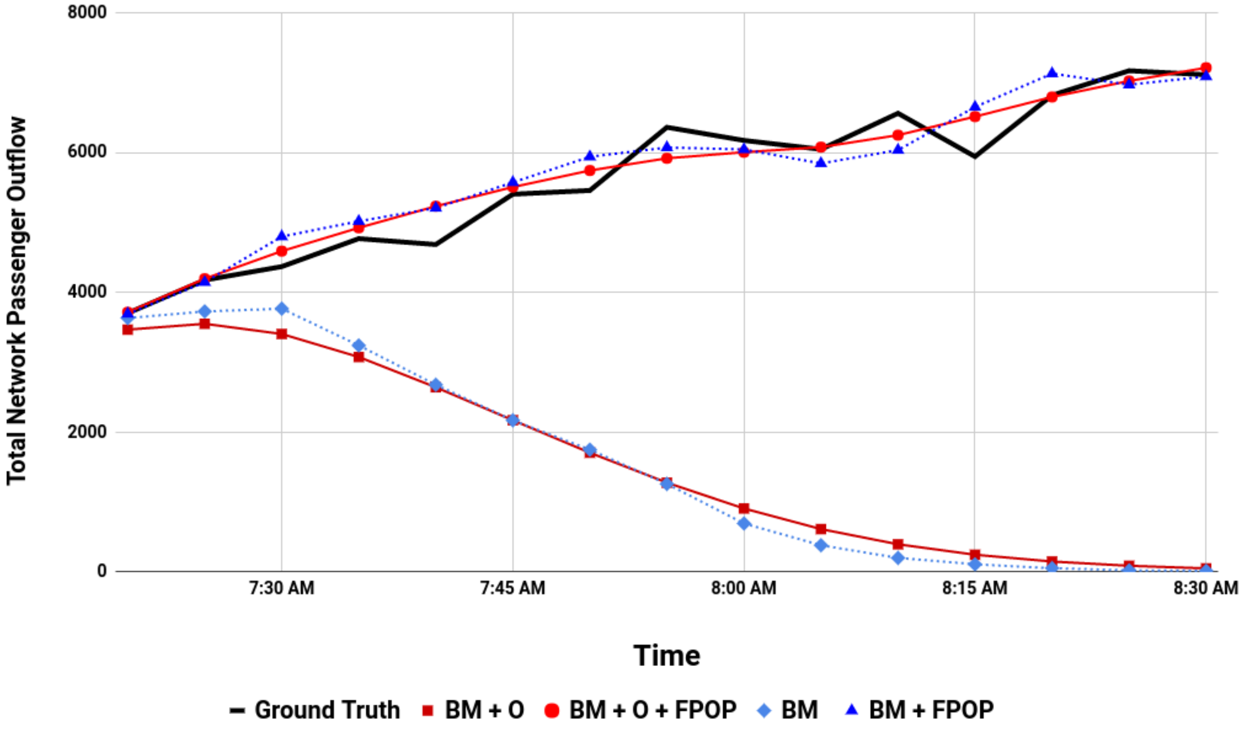

Figure 1: Estimation of Outflow for Entire Network

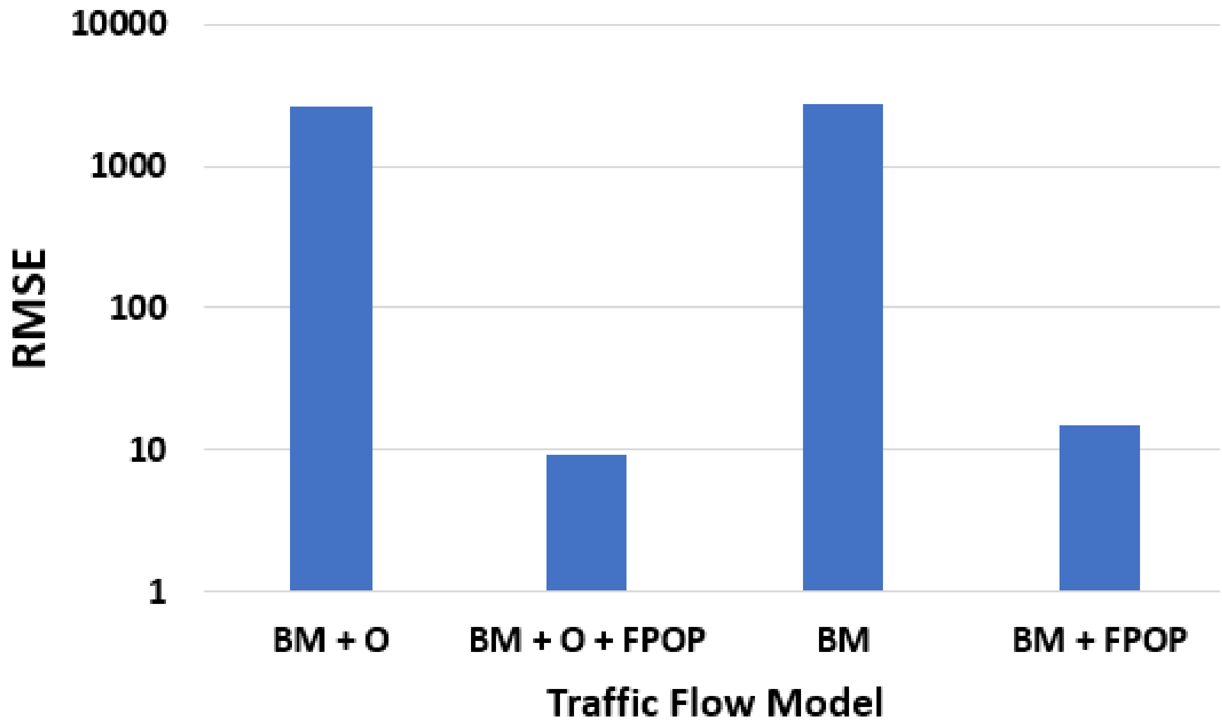

Figure 2: Passenger Flow Prediction Errors

Figure 1 shows the estimating outflow for the entire D.C. metro subway network on the morning of April 25th. It is readily apparent that our proposed Future Passenger Outflow Prediction (FPOP) model solves the long term deficiencies of the Bayesian models BM and $BM+O$. Whereas the previous BM and $BM+O$ models fall off in their prediction accuracy immediately, the FPOP models closely align with the ground truth over one hour into the future and show no signs of fragility. Figure 2 shows the resulting RMSE of our proposed approaches. Again, it is obvious that our proposed FPOP model allows accurate predictions into the long term, as the FPOP RMSE values are several magnitudes better than the RMSE values for Bayesian models.

To compare different trip travel time estimation models, we run two statistical tests. The first is the R-squared goodness of fit test for regression. The second is the mean absolute error for all predicted trip times:

$$MAE = \sum_{tr \in \mathcal{D}} pduration^{t} r - GT^{tr},$$

where $pduration^{tr}$ is the predicted travel time for trip tr, and GT^{tr} is the corresponding ground truth flow.

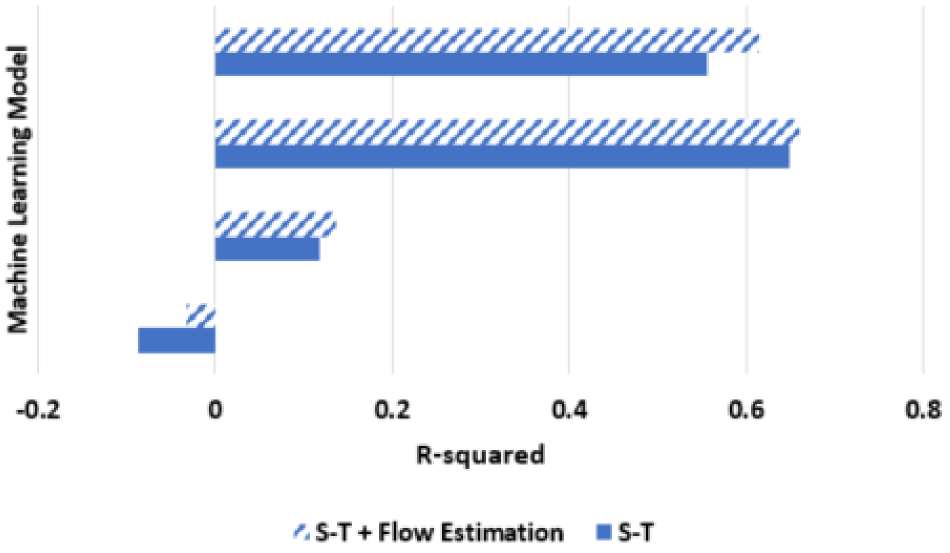

Figure 3: R-squared values for S-T

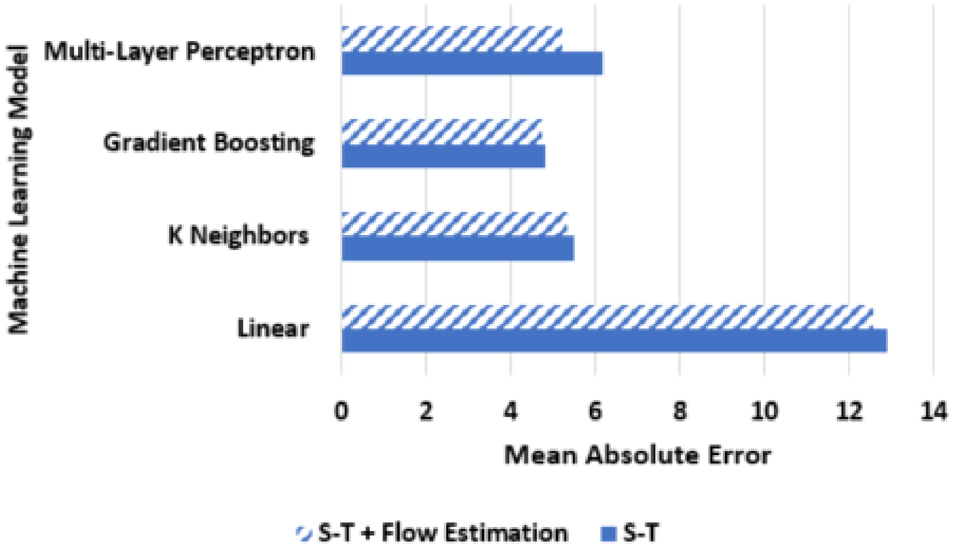

Figure 4: Mean Absolute Error for S-T

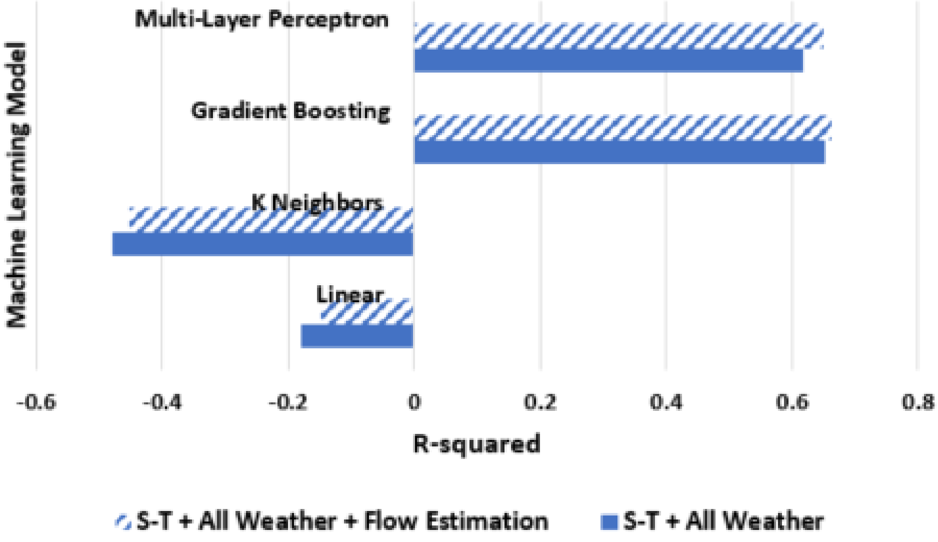

Figure 5: R-squared values for STAW

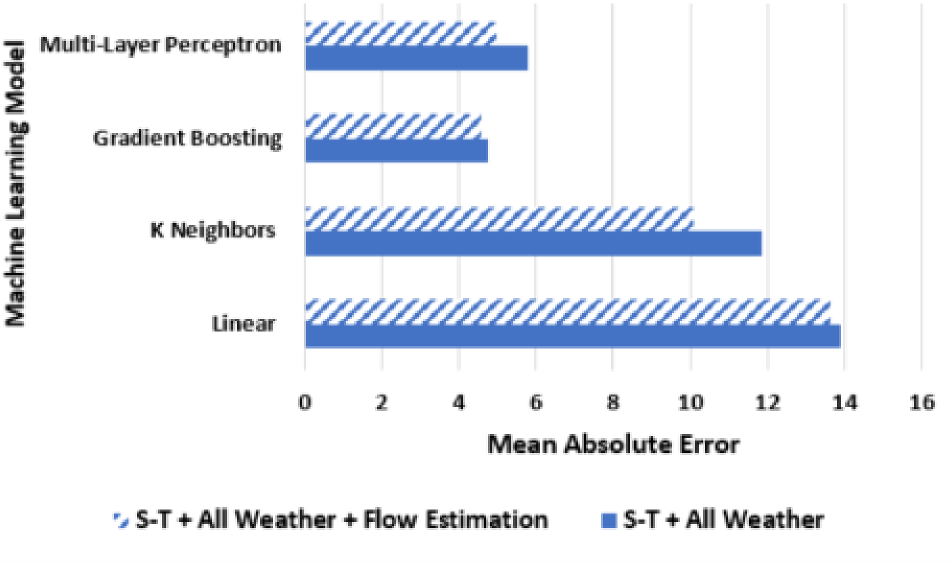

Figure 6: Mean Absolute Error for STAW

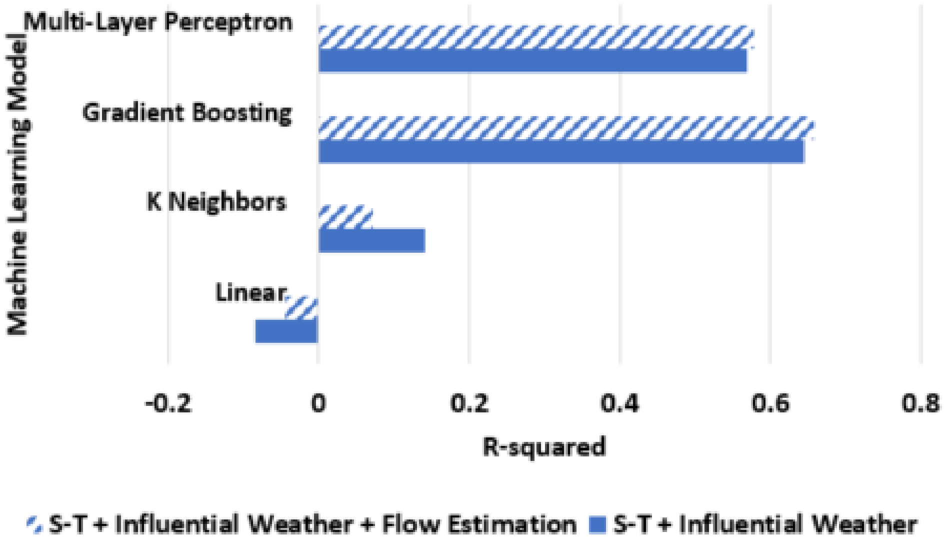

Figure 7: R-squared values for STIW

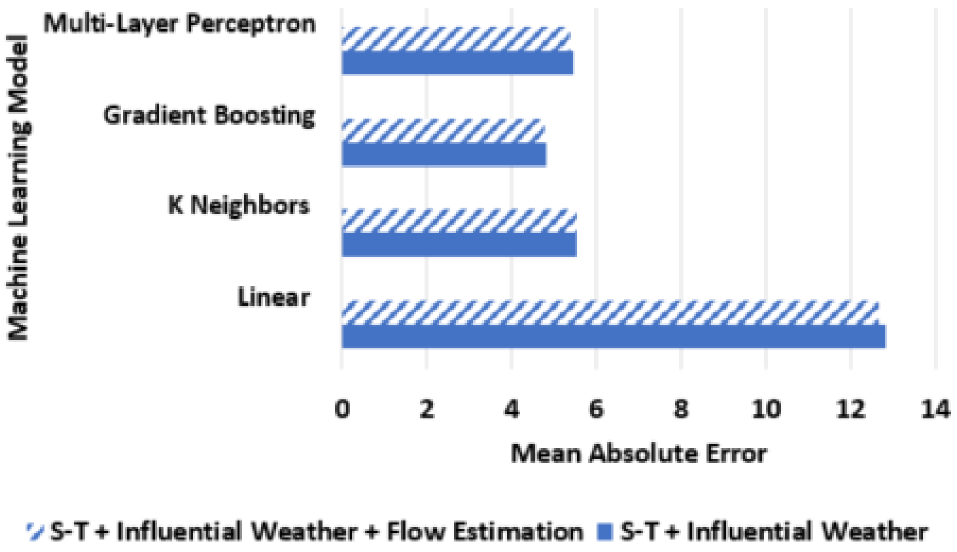

Figure 8: Mean Absolute Error for STIW

In the above figures, we see the difference in performances of the machine learning regression models and the effectiveness of the Traffic Flow forecast to improve travel time estimation. Our proposed model for OD individual trip duration surpasses baseline forecasting approaches, achieving an average absolute error of 4.5 minutes. This is a vast improvement on the performance of commercial vendors such as Google Maps, which use historical average techniques and achieve an average absolute error of 7.8 minutes.

Figures 3, 5, and 7 show the r-squared goodness of fit tests and Figures 4, 6, and 8 show the mean absolute errors (in minutes) for the machine learning models. Both metrics show that the addition of traffic flow estimation improves machine learning models' accuracy in predicting travel time. It is seen that the ensemble model, Gradient Boosting Regression, generally performs the best in terms of high goodness of fit and low mean absolute errors. However, we note that r-squared values may be deceiving as a metric since higher values do not always correspond to lower mean absolute errors. In fact, although K Neighbors Regression posted an extremely negative r-squared value in Figure 5, it performed better than linear regression in the corresponding mean absolute error Figure 6.

It is also important to note the effect of adding meteorology features. The addition of any weather features was enough noise to diminish linear regression's predictive abilities. K Neighbors regression acts similarly in its dismal performance in the STAW model. This is most likely due to the nature of K Neighbors – in its process of searching for similar trips, the model may be completely thrown off by the seemingly random noise generated by the weather features.

6 Conclusions

Past research has mainly focused on traffic flow estimations in the short term. We presented novel solutions for long term predictions for the outflow of passengers currently in a traffic network. For this purpose, we built a Bayesian model which captures empirical distributions but allows to condition this model to individual travel times of individual passengers. Next, we extended this to long term predictions through the use of generative heteroskedastic Poisson models. We then used this traffic flow estimation to accurately predict individual passenger trip times. Our experimental tests has shown that our approach is able to predict public transit outflow accurately for significantly longer times than baselines. Furthermore, our trip time estimation models achieve an average absolute error of 4 minutes and thirty seconds, consistently beating commercial approaches that average around 7 minutes of absolute error.

There are still open challenges. Our prediction models for travel time distribution between two stations are not optimal. The machine learning regression models were not the focus of our research. In the future, we can develop more extensive neural networks to achieve better regression results.

Acknowledgements

This research has been supported by National Science Foundation AitF grants CCF-1637576 and CCF-1637541.

References

[1] American public transport association. http://www.apta.com/mediacenter/ptbenefits/Pages/FactSheet.aspx.

[2] United states department of transportation. bureau of transportation statistics. https://www.bts.gov/topics/passenger-travel.

[3] United states census bureau. population overview. https://www.census.gov/topics/population.html.

[4] Asce infrastructure report card gives public transportation a d- grade.

[5] Washington metropolitan area transit authority. https://www.wmata.com/.

[6] Hans-Peter Kriegel, Matthias Renz, Matthias Schubert, and Andreas Züfle. Statistical density prediction in traffic networks. In *SDM*, volume 8, pages 200–211. SIAM, 2008.

[7] Abdeltawab M Hendawi, Jie Bao, Mohamed F Mokbel, and Mohamed Ali. Predictive tree: An efficient index for predictive queries on road networks. In *Data Engineering (ICDE), 2015 IEEE 31st International Conference on*, pages 1215–1226. IEEE, 2015.

[8] Itzhak Geva, Ezra Hauer, and U Landau. Maximum-likelihood and bayesian methods for the estimation of origin-destination flows.

[9] Jon Froehlich, Joachim Neumann, Nuria Oliver, et al. Sensing and predicting the pulse of the city through shared bicycling. In *IJCAI*, volume 9, pages 1420–1426, 2009.

[10] Andreas Kaltenbrunner, Rodrigo Meza, Jens Grivolla, Joan Codina, and Rafael Banchs. Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport system. *Pervasive and Mobile Computing*, 6(4):455–466, 2010.

[11] Yexin Li, Yu Zheng, Huichu Zhang, and Lei Chen. Traffic prediction in a bike-sharing system. In *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, page 33. ACM, 2015.

[12] Mengyu Dou, Tieke He, Hongzhi Yin, Xiaofang Zhou, Zhenyu Chen, and Bin Luo. Predicting passengers in public transportation using smart card data. In *Australasian Database Conference*, pages 28–40. Springer, 2015.

[13] William Mendenhall, Robert J Beaver, and Barbara M Beaver. *Introduction to probability and statistics*. Cengage Learning, 2012.