BuSCOPE : Fusing Individual & Aggregated Mobility Behavior for "Live" Smart City Services

Lakmal Meegahapola
Singapore Management University
lakmalm@smu.edu.sg

Thivya Kandappu
Singapore Management University
thivyak@smu.edu.sg

Kasthuri Jayarajah
Singapore Management University
kasthurij.2014@phdis.smu.edu.sg

Leman Akoglu
Carnegie Mellon University
lakoglu@andrew.cmu.edu

Shili Xiang
Institute for Infocomm Research
sxiang@i2r.a-star.edu.sg

Archan Misra
Singapore Management University
archanm@smu.edu.sg

ABSTRACT

While analysis of urban commuting data has a long and demonstrated history of providing useful insights into human mobility behavior, such analysis has been performed largely in offline fashion and to aid medium-to-long term urban planning. In this work, we demonstrate the power of applying predictive analytics on real-time mobility data, specifically the smart-card generated trip data of millions of public bus commuters in Singapore, to create two novel and "live" smart city services. The key analytical novelty in our work lies in combining two aspects of urban mobility: (a) conformity: which reflects the predictability in the aggregated flow of commuters along bus routes, and (b) regularity: which captures the repeated trip patterns of each individual commuter. We demonstrate that the fusion of these two measures of behavior can be performed at city-scale using our *BuScope* platform, and can be used to create two innovative smart city applications. The Last-Mile Demand Generator provides $O(mins)$ lookahead into the number of disembarking passengers at neighborhood bus stops; it achieves over 85% accuracy in predicting such disembarkations by an ingenious combination of individual-level regularity with aggregate-level conformity. By moving driverless vehicles proactively to match this predicted demand, we can reduce wait times for disembarking passengers by over 75%. Independently, the Neighborhood Event Detector uses outlier measures of currently operating buses to detect and spatiotemporally localize dynamic urban events, as much as 1.5 hours in advance, with a localization error of 450 meters.

CCS CONCEPTS

• **Computer systems organization** → **Real-time systems**; • **Human-centered computing** → **Ubiquitous and mobile computing**; • **Information systems** → **Information systems applications**;

KEYWORDS

Mobility Behavior; Regularity; Conformity; Live Smart City Services

1 INTRODUCTION

Analysis of digitized urban transportation data, such as taxi location traces or bus commute records, has long been used for a variety of urban applications, such as building mobility models [21], predicting likely future congestion hotspots [3] or classifying land use [25]. In general, these applications operate in *offline* fashion, analyzing historical data traces to generate policy-level outputs. In this paper, we instead focus on the opportunities of performing *live* & predictive analysis of such commuting data streams, to support soft-real time smart city operations.

We specifically focus on smart-card generated trip data for public buses in Singapore, where the vast majority of users tap-in and tap-out when boarding and disembarking from a bus, respectively, thereby providing $(origin, destination)$ records for individual trips. Through careful analysis of a month's worth of anonymity-preserving smart-card generated bus trip data (a total of 108 million trips, taken by ≈ 5 million commuters), we show that **the vast majority of public bus trips are predictable, and driven by routine commuting patterns**. We shall show that such predictability manifests in two aspects: (a) *individual-level regularity*, which allows us to predict an individual's point of disembarkation, as soon as she boards a bus, and (b) *aggregate-level conformity*, which allows us to use historical commuting flows to identify a relatively small set of likely disembarkation points, even for commuters with no relevant prior travel history.

We emphasize the notion of a *live* mobility analytics platform, which enables making operational decisions or generating neighborhood-level insights on streaming mobility data, with $O(mins)$ responsiveness. To support such soft-real time processing of the tens of thousands of passenger boarding and disembarkation events that occur city-wide per minute during peak commuting times, we shall develop *BuScope*, our server-based multi-threaded

platform that continually generates updated per-passenger and per-bus insights. In particular, we shall use such insights for two novel applications:

- *Last Mile Demand Generator (LM-Demand),* which provides $O(mins)$ look-ahead into the number of passengers projected to disembark at different bus-stops. By using this demand projection to dynamically redirect the placement of unmanned Mobility-on-Demand (MoD) vehicles, we can tackle the important problem of improving the *last-mile* commuting experience [1].
- *Neighborhood Event Predictor (NE-Pred),* which uses observed anomalous characteristics of 'live' commuting flows to both predict and *spatiotemporally localize* neighborhood-scale events. Identifying such events even before they start allow city authorities to intervene dynamically, such as dispatch traffic cops or adjust traffic light schedules.

Both of these exemplar applications are based on novel ways to harness this underlying predictability; while *LM-Demand* combines individual-specific regularity and aggregate-level conformity to accurately predict disembarkation volumes at future downstream bus-stops, *NE-Pred* uses bus-level outlier scores derived from the presence of *irregular* commuters to derive the spatiotemporal coordinates of likely urban events.

Key Contributions: Our key contributions in this paper are:

- *Establishing the Predictability of Bus Commuting Patterns:* We first show that most trips have high predictability: this allows us to predict an individual's destination, given the originating bus stop, with high confidence, even when the specific trip has low support in past data. We shall subsequently introduce a hybrid model for disembarkation prediction combining both the individual-level regularity and aggregate flow-based conformity and show that this has high accuracy: we can predict the exact disembarkation with an accuracy of >85% on both weekdays and weekends, with the mean location error in such prediction being 480 meters (approx. 1-2 bus-stops) on weekdays. This hybrid technique is shown to achieve $\sim 30\%$ improvement in prediction accuracy over 2 alternative baseline methods that simply utilize aggregated historical data.
- *Demonstrating the Utility of Disembarkation Prediction for Last-Mile MoD Positioning:* Through our analysis, we show that we can additionally predict disembarkation bus-stop accurately an average of 9 bus-stops (2.89 kms) in advance. By feeding such predictions through a simulation model of neighborhood-level mobility, we show that predictive pre-placement of MoD vehicles (with capacities varying from C =1-3 passengers/vehicle) can reduce the waiting time experienced by disembarking commuters by over 75%, to an average of less than 30 seconds, compared to 2 mins for a reactive baseline ($C = 1$).
- *Developing a Predictability-Driven Model for Event Detection & Localization:* We develop a novel method for event/anomaly detection, which first computes a continually-updated outlier score for each operating bus based on the inherent predictability of its on-board commuters. The method then extrapolates this outlier score to downstream bus-stops, reflecting our hypothesis that events often attract commuters making non-regular trips. By

then aggregating and spatiotemporally clustering such scores across (bus, bus-stop) combinations, we show that we can detect all 3 representative events with low average spatial error (463.8 meters), but also, on average, 100 minutes in advance of an event's start time. Moreover, we show that this approach of downstream extrapolation is superior to an alternative "spot anomaly" technique that is based on changes in the disembarkation volume at individual bus stops: our approach typically identifies and localizes both macro and micro events 40-80 mins in advance of the 'spot anomaly' method.

- *Operationalizing the Analytics through* BuScope*:* We present the design and implementation of *BuScope*, a platform that allows us to perform the predictive analytics outlined above in soft-real time, on underlying streaming data. We show that *BuScope* is flexible enough to recompute the analytical insights, at both individual and bus-level specificity, very frequently for peak city-scale workloads—e.g., it incurs 17.33 msecs average latency to process each of $\approx$ 270,000 boarding and alighting transactions generated by 221,217 commuters on 3777 buses, during a typical weekday, 30 minute peak period.

While *LM-Demand* and *NE-Pred* are novel and innovative smart city applications, we believe that our broader contribution is in demonstrating the power of "live" analytics on such underlying transportation transactional data, thereby potentially paving the way for public transport companies worldwide to make such pseudonymized data available in real-time.

2 DATASET & APPLICATIONS

To provide a clearer understanding of the predictive analysis and the new applications that form the core of this paper, we first detail both our dataset as well as outline the high-level operation of *LM-Demand* and *NE-Pred*.

2.1 Dataset Description

Our analysis and test of the developed analytics platform is based on the public transport smart-card data [2] of 5.1 million commuters of Singapore. Because fares on the Singapore public transit system are distance-proportional and because smart-card fares are significantly lower than paying cash, the overwhelming majority of commuters utilize the smart-card to 'tap-in' (while boarding) and 'tap-out' (while disembarking), thus enabling the capture of the origin & destination locations and timestamps of each journey. The dataset available to us consists of comprehensive records of the tap-in and tap-out details of approximately 180 million trips made by commuters during the month of August, 2013. The data spans across 4913 bus stops and 153 MRT stations—for this paper, we focus solely on the 108 million bus journeys, i.e., those that start and end at bus-stops. The dataset is *pseudonymized* and contains no explicit personally identifiable information (PII): each journey results in a unique commuter-specific entry with the fields described in Table 1, where the identifier *cid* is unique for each smart-card.

Common Definitions: In anticipation of the analysis in Section 3, we define the following terms:

Regularity/Support: Similar to the traditional definition in data mining literature, the *support* of a trip *jid* by commuter *cid* is defined

---

Attribute	Description
jid	captures the unique ID of a journey (e.g., boarding at station X and alighting at station Y).
cid	represents the unique, randomly generated card ID of the commuter.
$tmode$	captures the mode of the transport – (a) bus, (b) train and (c) light rail.
$sernum$	denotes the service number of the bus
dir	direction of the journey (to/from the origin hub)
$regnum$	bus instance ID
$bstop$, $astop$	ID of the boarding and alighting stops
$timestamp$	time of boarding
dis, $time$	total distance and sojourn time of the journey

Table 1: Dataset Description.

as the fraction of total trips (by cid) with the same $bstop$ values (i.e., identical boarding stops), normalized by the total trips (in the entire dataset) involving cid. Note that the support definition may be time-interval specific, with a commuter's support defined separately, for example, for *weekday peak period* vs. *weekend off-peak period.*

Confidence: The *confidence* of a trip jid with a specific ($bstop$, $astop$) tuple is defined as the probability of the user cid's disembarking at bus-stop $astop$, *given* that she has boarded at $bstop$– i.e., the ratio obtained by dividing the number of user cid's trips with ($bstop$, $astop$) as the source-destination pair, divided by the total number of user cid's trips originating at $bstop$.

2.2 The Last-Mile Demand Prediction Application

While Singapore has an ambitious *car-lite* vision that promotes extensive use of its excellent public transportation system, studies[3] show that the overhead of the last-mile commute (the journey from/to the commuter's residence to/from the nearest bus stop) plays a big role in commuter reluctance to switch from a private car[31]. Accordingly, Singapore is pursuing a vision of driverless MoD, where robo-taxis would ferry commuters to/from their doorstep to the nearest public transport node.

A natural operational challenge in this setting is to maximize the utilization of such unmanned resources, and consequently *minimize the waiting time of commuters*. Given a finite set of such MoD resources, the key to minimizing the waiting time (at least for the return commute) is to *pre-position* the robo-taxis at the disembarkation points by anticipating the demand (the number of passengers disembarking at a bus stop at a future time instant). Figure 1 illustrates this concept, at a neighborhood level, with 3 different bus-stops and a 2 robo-taxis. Rather than allocate such MoD vehicles reactively (after passengers have disembarked and requested a ride) and cause passengers to wait, a smarter strategy would have proactively dispatched the robo-taxis to different bus-stops so that commuters find them "magically" waiting as soon as they disembark–e.g., Robo-taxi A picks up the passenger at time $t = 1$, while robo-taxi B moves to the third bus-stop to pick up the disembarking passenger at time $t = 2$. A successful realization of this vision requires us to: (a) *predict* the demand at each bus stop accurately, and (b) perform smart *decision optimization* and

proactively direct the robo-taxis to such predicted demand. In this work, we focus almost exclusively on the demand prediction aspect, and will show how *LM-Demand*'s predictive analytics on such smart-card transactions can provide highly accurate estimates of the number of disembarking passengers, *sufficiently in advance*. Of course, to illustrate the likely benefits of such prediction, we shall provide a comparative performance analysis of a straightforward MoD dispatch strategy, deferring the problem of algorithm design for predictive dispatch to future work.

2.3 The Neighborhood Anomaly/Event Predictor

Large cities are highly dynamic, with potentially dozens of events (such as festivals, concerts and fairs) taking place in different city neighborhoods daily. City planners and urban agencies are very interested in detecting and tracking such events, to gain a better understanding of neighborhood dynamics, ascertain its *livability* and also respond with timely interventions, such as dynamically adjusting transport network parameters (e.g, directionality of traffic lanes or duration of traffic lights) or deploying human resources (e.g., traffic officers) for better event management.

A variety of approaches (e.g., using social media data [28] or bike trip records [39]) have been proposed for such event detection. The challenge, of course, is to reliably isolate the contributory component of an event to such large-scale transactional data, from the daily dynamics of "normal" mobility patterns. Our belief is that many such events cause residents to exhibit anomalous commuting patterns, and that some measure of anomaly aggregation, across the hundreds of geographically dispersed bus instances operational at any instance in a city, will provide a clear and reliable signal about the time and place of such underlying events. Bus usage data seems particularly appropriate for such prediction, as commuters heading to an event location are likely to board buses well in advance–e.g., 30 mins-1 hour before the start of an event. Figure 2 illustrates the high-level idea. We see an event in a city location, with 'red' commuters denoting those exhibiting unusual travel patterns (e.g., travelling on routes that they don't normally use, or at hours not usually seen). Such 'red' commuters are disproportionately present on buses heading *towards* the event location, allowing the use of appropriate spatiotemporal clustering techniques to predictively localize the event. In this work, we shall focus on three aspects of this idea: (a) event detection: correctly declare the occurrence of such neighborhood-scale events; (b) event localization: accurately identify *when* and *where* such an event is happening; and (c) most innovatively, event prediction: forecast the start time of an event.

3 EMPIRICAL INSIGHTS FROM CITY-SCALE COMMUTER PATTERNS

Our exemplary applications and the overall design of *BuScope* are driven by a fundamental observation: *the vast majority of bus trips undertaken by commuters, whether on weekdays or weekends, are in fact predictable.* Such predictability will enable us to predict (a) the number of disembarking passengers at downstream bus-stops (Section 5) or (b) the time & location where an event will be held (Section 6). In this section, we empirically demonstrate two key aspects of such predictability:

---

[3]https://www.todayonline.com/singapore/looking-ahead-2018-restoring-public-confidence-mrt-service-vital-steer-sporeans-away-cars
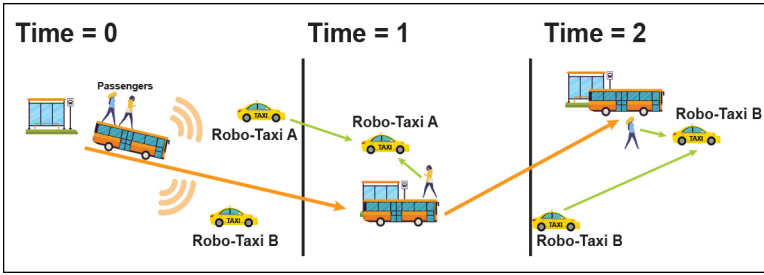
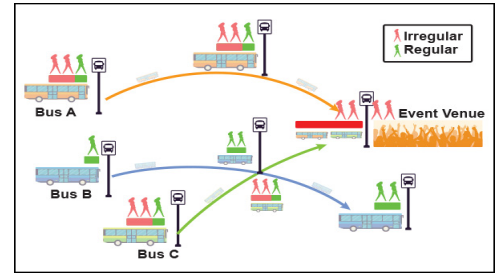Figure 1: Disembarkation Prediction & Last-Mile MoD Pre-placement

Figure 2: Mobility-Driven Event Detection & Localization

- Predictability of a journey's destination–i.e., being able to infer where a passenger will disembark, given his embarkation context (such as the bus stop, bus service and time of boarding).
- Ridership mix in a bus–i.e., characterizing the mix of passengers on the bus who are exhibiting "normal" vs. "abnormal" commuting patterns.

In addition, we also aim to understand the look-ahead time of such predictions. We first look at the typical *regularity* of commuting patterns, and uncover both individual-specific and aggregate-level properties that aid such predictions.

3.1 Person-Specific Commuting Regularity

We first study the inherent *regularity* of individual commuting patterns, viewing each trip as a ($origin$, $destination$) tuple. Predicting the disembarkation location of a commuter is then driven by the conditional probability of alighting at bus-stop 'Y', given the boarding bus-stop 'X', and can be mined as an *association rule* with a corresponding *support* and *confidence*. For example, assume that commuter *c* has 100 trip records *for a specific Context*, out of which 50 originate from bus stop 'X', with 40 of those terminating at bus stop 'Y'. Hence the rule {$Boarding = X$} $\Rightarrow$ {$Alighting = Y$} (interpreted as: if boarding stop='X', then alighting stop='Y') has a support of 50% (=50/100) and confidence of 80%(=40/50). More specifically, we define 4 different diurnal time windows[4]: {*AM peak; AM off-peak; PM peak; PM off-peak*}, corresponding to the four distinct service frequencies defined by the public bus services, each for two different day-of-week categories {*weekends*; *weekdays*}, resulting in 8 distinct contexts. Further, we consider two geographical areas in Singapore: one in the *Central Business District* (CBD), and one in the more residential, *Non- Central Business District* (NCBD) to capture varying dynamics of bus usage behavior.

In Figure 3, we provide a scatterplot of the top-3 *confidence* values, along with the corresponding *support* for each trip (O-D pair) observed across all users. We employ a *Common Path Re-identification* technique, whereby *support* is defined based on (origin, destination) end-points and not just bus routes, as one O-D pair of bus-stops may often lie on the common route of multiple bus services. By aggregating trips made on such different services into a common (O,D) pair, we can improve both the *support* and *confidence* of individual journeys.
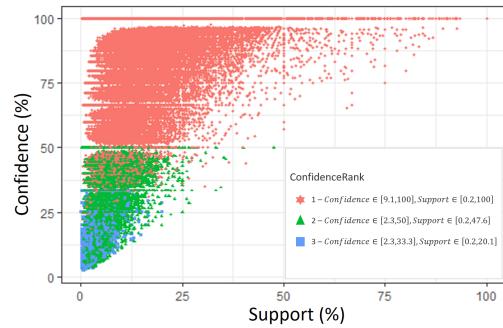

Figure 3: Spread of Support and Confidence for various Confidence Ranks

We observe that the *confidence* increases as support grows progressively larger–i.e., if a commuter has a past history of undertaking a particular journey repeatedly (e.g., home→work), then we are more certain that he/she will alight at her workplace when starting a future journey from home. However, even for low support values (support<5%), the confidence values are quite high (often exceeding 85%). This result would suggest that **predicting disembarkation using such individualized travel history would suffice for users for whom there is even a modest past history of similar trips**. In Figure 4, we plot the CDF of confidence for the top-3 (x-axis) most likely alighting destinations for different (O,D) pairs and space (CBD vs. NCBD) and time (weekdays vs. weekends) bins. We see that individual-level behavior is highly deterministic–in the vast majority of cases, top-3 disembarkation predictions achieve ≈100% confidence indicating that, for most originating destinations, a commuter disembarks at one of most 2-3 bus stops.

3.2 Generic Commuting Trend and Flow Based Patterns

To additionally capture the fraction of bus passengers who do not have enough "support" from past trips to make an individualized prediction, we next examine the overall aggregated *flow-level* behavior of commuters. A significant amount of past literature on urban mobility has utilized such flow-level statistics. Our hypothesis is that some degree of prediction about an individual's likely disembarkation bus-stop may be gleaned by observing aggregate flow-level transition probabilities–i.e., by asking, what fraction, of the total number of individuals boarding at bus-stop 'X', are observed to disembark at bus-stop 'Y'? In Figure 5, we plot the CDF

---

[4]AM peak= (6-10:29am); AM off peak (10.30am-3:59pm), (c) PM peak (4pm-7.59pm), and (d) PM off peak (8pm-5:59am)
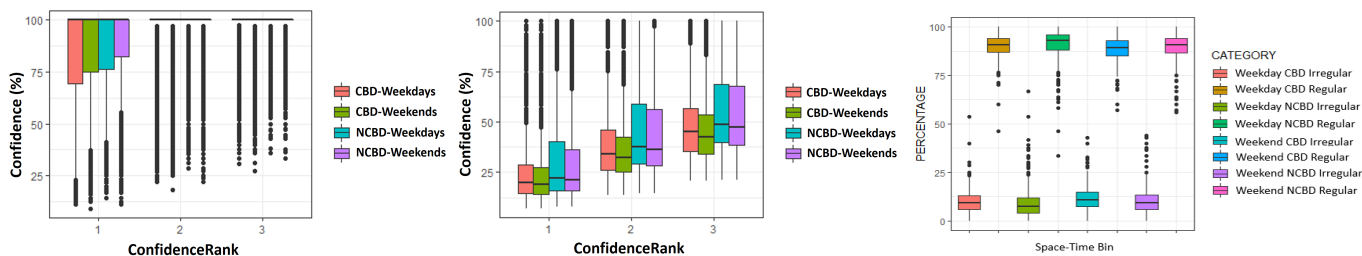
Figure 4: Spatiotemporal variability of confidence of personalized predictions

Figure 5: Spatiotemporal variability of confidence of flow-based predictions

Figure 6: Fraction of *Irregular* **Passengers**

of confidence (per space-time bin) for the top-3 highly probable disembarkation at bus service level.

In general, we observe that, on average, given a source bus stop 'X', the *confidence* that embarking passengers will disembark at one of the top-3 probable locations is close to 50%, even though the average number of stops along any service route is relatively larger, i.e., ≈ 49.77. Through further fine-grained analysis (details omitted due to space limits), we found that, for the vast majority of bus routes, these 3 stops are often common across different values of 'X'. In other words, most bus routes had a few (usually 3-4) *sink nodes*, which see a high volume of disembarking passengers, even though the passengers board the bus at a variety of bus-stops. As an illustration of this, Figure 7 uses a "chord diagram" (see [7]) of both directions of the bus route "139". We can see the existence of a clear set of *sink* nodes (e.g., bus stops 13099– major residential estate and 7517 – shopping district) that witness a disproportionately large volume of disembarking flows.

3.3 Typical Per-Bus Passenger Profiles

Figure 6 plots the fraction of commuters (daily, over the observation period of 30 days) whose trip can be characterized as "irregular"– i.e., for which, there are no records of similar past trips (for this plot, we use the first 3 weeks as training data, and the last week as the test period.) We see that, across the entire city, the fraction of such irregular trips is low, but not negligible (about 15%), implying that ignoring the impact of such irregular commuters may lead to misleading predictions. Moreover, the fraction of such irregular users on any bus is observed to be high only rarely (in less than 5% of the bus instances captured in our 1-month data), suggesting that **this may be a feature potentially indicative of unusual events occurring along or near the corresponding bus route**.

4 THE BUSCOPE SYSTEM

The results in the previous section were focused on empirically establishing key commuting properties of bus users, thereby motivating the smart city applications that we shall describe later. To support the *live* services that we envision, we now describe our design and implementation of *BuScope*, which provides the soft-real time analytics components needed to support multiple smart-city services. As we shall show, the overall number of events of interest across the bus network may appear large (an average of ≈ 3.142 million bus commutes each day) but can be supported by a relatively straightforward, multi-threaded, in-memory implementation on a single production-grade server.

Figure 8 shows the *BuScope* middleware architecture, consisting of the following components:

- *Bus Event Generator (BEG):* This component resides on the bus and is logically part of its telematics unit. It effectively generates a stream of events, aggregating multiple boarding and disembarkation events into a single payload at each bus-stop, generating an average of 743.40 events/minute (across all buses) during peak hours.
- *Passenger Instance Monitor (PIM):* This component logically maintains the state of every passenger currently in transit on any bus in the public transportation network. The incoming data streams from BEG units are de-multiplexed and dispatched to one of multiple PIM threads. The threads operate on a common *In-Memory Passenger Table* (implemented as a hashmap), which maintains a collection of passenger records, indexed by the passenger ID (the *cid*) and the service route. Each record stores, among other fields, a boolean flag indicating whether this is a *regular* passenger or not, and a disembarkation list (with the disembarkation probability for each downstream bus-stop).
- *Bus Instance Monitor (BIM):* Analogous to the PIM, this component logically maintains the state of each bus instance that is currently operational. In particular, the incoming data streams from each bus instance is assigned to one of multiple BIM threads, which share a common hashmap-based *In-Memory Bus Table*. Each record in this table maintains the following bus instance-specific fields: bus location, bus service number, number of on-board passengers, list of on-board passengers (pointers to entries in the In-Memory Passenger Table) and the fraction of on-board passengers classified as *regular*.
- *Profile Repository:* This component stores the results of the off-line analytics that are periodically performed across the entire bus network's transportation data. It computes and stores (a) a passenger-centric profile, which includes a per-passenger, per-service (O, D) matrix (one for each of the 8 day type-time bins described previously) storing the number of past trips for that (O, D) pair; and (b) a bus-service specific matrix that similarly stores the number of observed (O, D) flows between all bus-stops on the route, aggregated over all passengers.

As illustrated in Figure 8, the *BuScope* system exposes a set of service APIs that are used by the *LM-Demand* and *NE-Pred* applications.
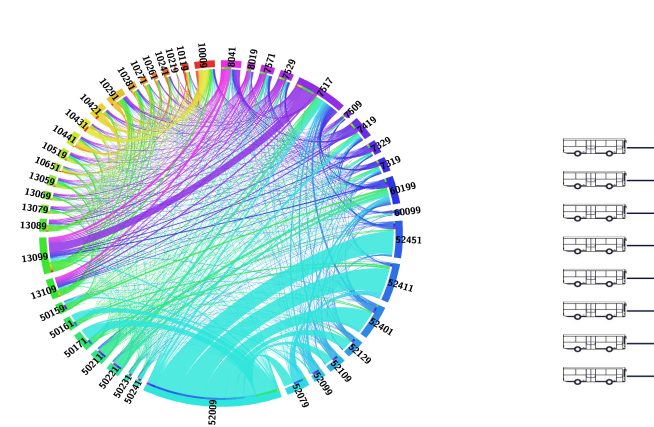
Figure 7: Visualizing Disembarkation Probabilities for Bus Service "139" (Direction 1)

Figure 8: Functional Components of *BuScope*

4.1 Performance Considerations

To understand the workload characteristics of the *BuScope* system and its impact on the system complexity requirements, we first analyzed the historical data to understand (a) the event intensity of embarkation & disembarkation events across all buses, and (b) how often we generate a bus-stop crossing event, across the entire city. Figure 9 plots the average of the transaction events/min–i.e., the sum of the embarkation and disembarkation events across the entire Singaporean bus network, over different hours of the day, at minute-level granularity. We see that, at peak periods, there are approx. 6000 boarding and alighting events/min on average, with the total reaching approx. 12000 such events–each such event will correspond to an update (either creation or removal) in the corresponding PIM entity. Similarly, Figure 10 plots the number of bus-stop crossings per minute. This metric is relevant as the BIM-specific information needs to be potentially updated only after each crossing (as a bus's state will not change in between bus-stops). We see that, during the rush hour peaks, the maximum rate of generation of such crossing is $\tilde{9}00$ crossings/minute, implying that the *BuScope* should be able to update one BIM record with an average latency lower than ~ 60 msecs.

4.2 *BuScope* System Performance

To satisfy the above performance bounds, the *BuScope* implementation, hosted on an Intel Xeon server with 128 GB memory and up to 14 processor, effectively utilizes multiple BIM and PIM threads. Figure 11 plots the relationship between the number of PIM components (N_p), and the average processing latency, when processing incoming events in per-minute chunks. The experiment is performed using events generated during a 30-minute evening peak period (6.45 PM to 7.15 PM) from different weekdays. We observed that the memory footprint (consisting primarily of the records of currently on-board passengers throughout the bus network) remains almost invariant at $\approx$ **10.05 MB**. We can conclude that the use of a modest number (3-4) threads allows *BuScope* to comfortably handle even peak event workloads–even a single-threaded implementation takes $\approx$59 seconds to process an entire minute's worth of events, with each individual event incurring an average of
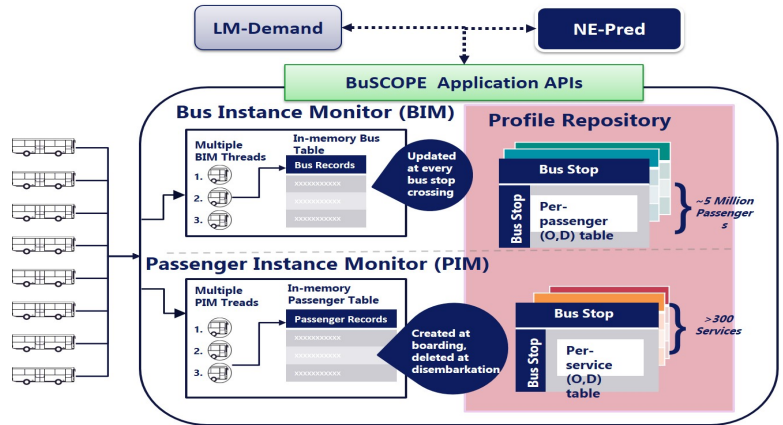
17.33msec processing latency. Similar results are obtained for the BIM components (details omitted due to space constraints): a modest number of threads allows *BuScope* to update each bus-specific record with the details of multiple boarding or alighting passengers at each bus-stop.

5 LM-DEMAND: PREDICTIVE MOBILITY-ON-DEMAND

We now tackle the *LM-Demand* application. At a high-level, this application has two components: (a) an analytics component that looks to predict the number of disembarkations at a location (either an individual bus-stops or a collection of nearby bus-stops) sufficiently in advance; and (b) a resource optimization component that uses such prediction values to smartly dispatch and pre-position the MoD vehicles.

Given the commuter dynamics investigated in the previous section, we are now aimed to forecast the disembarkations in a given bus stop by leveraging the personalised and aggregate level commuter traits. More specifically, our tasks are to have the capabilities of (a) user-level disembarkation prediction given that he boarded a bus service from a particular bus stop, and (b) ability to make such predictions with a certain look-ahead time.

5.1 Hybrid Approach for Disembarkation Prediction

As we have seen previously, the disembarkation point of *regular* users can be predicted quite accurately at the time of boarding; similarly, the disembarkation point of irregular users can also often be assigned to a limited set of *sink* locations along the route. We propose to build a hybrid model that synthesizes both these approaches, taking advantage of *regularity* (the predictability of travel patterns at an individual level) and *conformity* (the tendency for people to follow flow-based statistics at an aggregate level) to obtain a more precise prediction.

Our approach works as follows. Based on the entries in the *BuScope* repository, each boarding commuter is declared as a *regular* vs. *irregular* passenger for this specific journey. More specifically, consider a commuter c boarding a bus b at bus stop s in a give
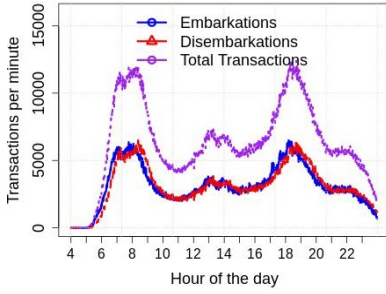
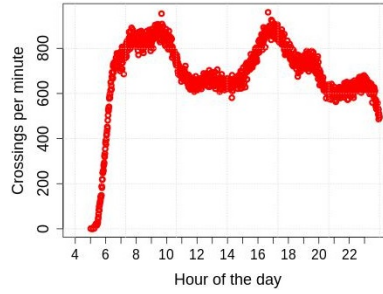**Figure 9: Number of Embarkation/Disembarkation Events**

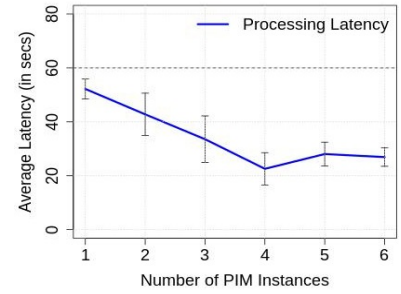**Figure 10: Intensity of Bus-Stop Crossings**

Figure 11: Processing Latency vs. Number of PIM instances

time-bin (as before, for concreteness, we consider the 4 time-bins corresponding to the tuple (off-peak|peak, weekend|weekday)). If this boarding pattern for c has *high support* (we'll quantify 'high' shortly), then this user is marked as *regular* and cid is projected to disembark at the highest-confidence bus-stop (among the bus-stops remaining in the journey) for this particular boarding pattern. (As mentioned before, the boarding pattern uses the common path re-identification strategy to incorporate the possibility that multiple bus services may provide an equivalent journey for this segment). However, if the user support is low, then c is declared an irregular user, in which case we use the aggregate flow information to declare that cid will disembark at the bus-stop that has the highest flow-based conditional probability from s.

To formalize this approach, we use $N_{threshold}$ as a tunable system parameter to determine if a user is regular or not for a specific boarding bus stop. $N_{threshold}$ represents the fraction of historical trips (relative to the number of trips taken by the most regular user of any single bus stop in the data set) needed to classify a user as a regular:

$$N_{threshold} = \left\lfloor \left(\frac{(x_{threshold} - x_{min})}{(x_{max} - x_{min})} 100\% \right) \right\rfloor \quad (1)$$

where x_{max} & x_{min} ($x_{max} \geq x_{min} \geq 0$) respectively denote the maximum & minimum number of records (across all users) that originates from *any* single bus stop. (The expression $N_{threshold}$ simply provides a common way to define regularity across different data sets: given $N_{threshold}$, x_{min} and x_{max}, we can then compute $x_{threshold}$ ($\in [x_{min}, x_{max}]$) denoting the actual minimum number of historical trip records which should be present for a traveler to be considered regular). For a given user, boarding with a given (embarkation bus stop, time-window) context, we then compute the user-specific value x_{user}, the number of prior embarkations in the data set with the given context, and declare the user's current trip to be *regular* iff $x_{user} \geq x_{threshold}$.

5.2 Evaluation of Disembarkation Prediction

To present detailed results, we consider 20 bus services belonging to two distinct classes: 10 buses that pass through the CBD from other areas of Singapore and 10 feeder buses that traverse primarily through NCBD areas. We consider the data of first 3 weeks (comprising 14 weekdays) as a training set and last week of August 2013 as the test set.

Evaluation metrics: We study two key metrics: (a) *Prediction localization error* – this measure computes the spatial distance between
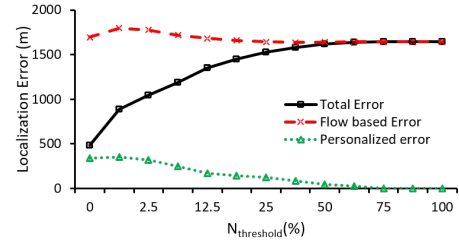

Figure 12: CBD Weekday

the actual and predicted location of disembarkation; and (b) *Bus ridership estimation* – this measure computes the error in predicting the number of passengers remaining on-board a bus, and is computed as the difference between the total passengers actually on the bus at a future stop and the number predicted to remain. In addition, we compare our proposed Hybrid approach against a *Flow-based baseline* [39], where the number of passengers predicted to disembark at bus-stop d (out of the set of passengers N_b who boarded at b) is computed as $N_b * P_{bd}$, where P_{bd} is the flow (transition) probability from bus-stop b to d.

Figure 12 plots the localization error, averaged across all bus instances and all of the 10 routes, separately for buses that travel through CBD on weekdays. The average localization error is plotted as a function of $N_{threshold}$. As $N_{threshold}$ increases, the proportion of trips deemed to be regular diminishes and that of irregular trips increases, as a commuter must have undertaken many more rides to be considered as a candidate for individualized prediction). In particular, for $x_{threshold} = 1$, a trip is considered to be regular if there is a history of even one past embarkation by the commuter within that time-bin (i.e., $x_{user} \geq 1$), and along the current route. We plot both the total average error, as well as the errors for the regular and irregular trips separately. We see that at the left-most extreme point (i.e., all trips with non-zero support value classified as regular) provides the least localization error, of approx. 500 meters (1-2 bus stops). In contrast, the Flow-based baseline conforms to the right most extreme point (i.e., all users classified as non-regular) and incurs an error of >1.5 km (>6 bus stops). Note that, as expected, as $N_{threshold}$ increases, the average personalized error decreases as the set of regular trips are now restricted to only those that are observed even more dominantly and thus represent highly predictable commutes (e.g., home-to-office). However, the fraction of trips deemed regular also decreases, and the higher contribution
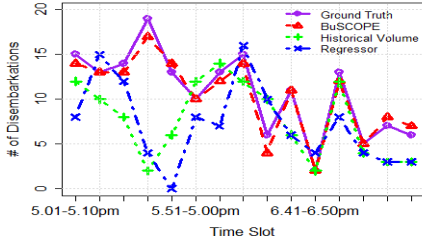
Figure 13: BuScope vs. Baseline (historical) predictors

of flow-based errors increases the overall error rate; hence, we use $x_{threshold} = 1$ for our subsequent analyses.

	CBD (in meters)	NCBD (in meters)
Weekday	480.3	342.8
Weekend	548.4	558.1

Table 2: Localization error of Disembarkation Prediction

In addition to the error, we also studied the accuracy of disembarkation prediction–i.e., the fraction of trips an exact prediction of the alighting stop was made. From Table 2, which tabulates this accuracy for all 4 spatiotemporal bins, we see that the accuracy is generally above 85%.

Furthermore, we compare our hybrid approach *BuScope* with 2 new baseline strategies that are both based on *aggregated* analysis of historical commuting data:

(a) *Historical Volume:* This approach computes (and uses as the predicted value), for each bus stop, the average number of aggregate disembarkations observed historically within a specific temporal (e.g., time-of-day, day-of-week) window.

(b) *Regressor:* This approach constructs a linear regression model (per bus stop) with the following covariates: time-of-day, day-of-week and the number of buses seen to historically transit through that bus stop within that time window. This regression model is then used to estimate the disembarkation; note that this model is not predictive as it needs the retrospectively reconstructed ground-truth of the number of transiting buses.

In Figure 13, we plot the number of predicted disembarkations for a 10-minute time window (depicted in X-axis over a 2.5-hour period) at a city-hub bus stop that serves an urban campus, local museums, and various businesses. We observe that *BuScope* tallies with the ground truth (actual disembarkations) much better, achieving 92.59% accuracy; in contrast the *Historical Volume* and *Regressor* strategies achieve only 55.56% and 62.96% accuracy, respectively. By performing similar analysis over the entire set of bus stops and bus routes, we find that *BuScope* achieves a significant (over 30%) accuracy improvement over both baselines.

Dynamic Disembarkation Prediction and Lookahead Time: The results above require the prediction of a trip's disembarkation bus-stop right at the point of boarding. In a slightly more sophisticated, dynamic version, the disembarkation predicted is updated dynamically, as the journey progresses. In particular, if the passenger remains on-board when the bus passes the currently

predicted disembarkation stop, the prediction is updated to the downstream stop with the highest conditional probability. Accordingly, one can anticipate that the prediction accuracy increases as the journey progresses and the bus gets nearer to the true stop. We empirically found that this dynamic prediction accuracy was significantly higher (89% accuracy) when the prediction was made 13 mins (corresponding to 9 bus stops) in advance of the actual disembarkation[5].

Impact of Spatial Granularity on Prediction Accuracy: We also investigated the impact of the spatial granularity on the prediction accuracy level. We first mapped the individual bus stops island-wide to grids of size $N_g = (200, 400, 600, 800, 1000)$ meters, and re-ran the predictions. Figure 14 plots the accuracy for different grid sizes, across **all** bus services (not just the 20 previously mentioned), for a typical weekday, AM peak period. As anticipated, the accuracy of prediction improves in all three cases (i.e., personalized, flow-based and hybrid), reaching 80% for $N_g = 800$ meters.

5.3 Predictive MoD Performance

Having established the ability to predict disembarkation locations with high accuracy, we now show that such predictions can be used to improve the performance of last-mile MoD systems. We consider the case of a specific Singapore neighborhood[6] (Toa Payoh, one of Singapore's central residential estates) and focus on the disembarkation behavior of passengers across a representative region consisting of 16 bus-stops. We specifically consider a two hour window around the PM-Peak period (when a large number of commuters may be expected to return home)–this region sees, on average, ≈ 300 disembarkations during this period.

Given that a last-mile MoD system does not currently exist, we utilize a simulation framework to model the MoD system. We make a few simplifying assumptions: (a) each vehicle's capacity is C passengers and passengers are serviced in First-come-First-served (FCFS) fashion; (b) the final destination of a last-mile passenger is randomly distributed within the region, and is modeled by a constant travel time (from the bus-stop) of T_D mins; (c) similarly, unless a vehicle is at the disembarking bus-stop, it will take T_D mins to arrive there from its current location. In addition, we assume the availability of accurate travel time estimates (now widely available via various applications in cities such as Singapore) and thus assume that the arrival time at a bus-stop is known via external means.

We study two different strategies:

- *Reactive MoD Strategy (S1)*: Under this strategy, we assume that a vehicle remains stationary after dropping its current complement of passengers, and moves to the next bus-stop whenever there is a waiting passenger there (thereby causing the passenger to experience a wait time of *at least* T_D mins). Of course, if the vehicle is currently busy, it must first complete its current set of dropoffs, before heading back for the next passenger.

- *Proactive MoD Strategy (S2)*: Under this strategy, if a vehicle is free, and a set of disembarkations are predicted to happen in the future, the vehicle proactively moves to the bus-stop with the

---

[5]This lookahead time vs. accuracy tradeoff will be used in our analysis of predictive MoD placement.
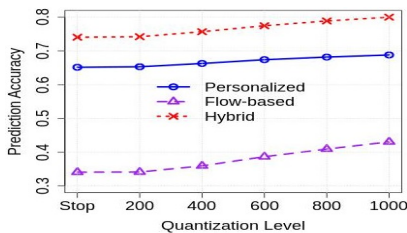[6]https://data.gov.sg/dataset/master-plan-2014-subzone-boundary-web

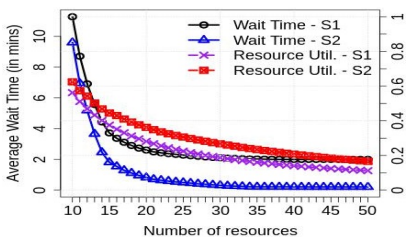**Figure 14: Impact of Spatial Quantization on Prediction Accuracy**

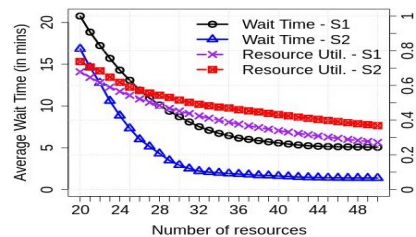**Figure 15: Waiting Time vs. Resource Utilization with T_D=2mins (C=1)**

Figure 16: Waiting Time vs. Resource Utilization with T_D=5mins (C=1)
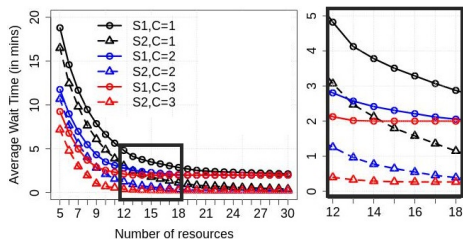

Figure 17: Impact on Waiting Times with Varying Capacity, C, of MoD Vehicles (left), with zoomed-in view for 12 to 18 vehicles (right).

earliest such predicted disembarkation. If the next disembarkation actually occurs more than T_D mins after the vehicle became free, then the passenger will experience zero wait; else, his wait time will be the difference between the vehicle's arrival time and the true disembarkation time. Moreover, because a vehicle takes at most $2T_D$ minutes to respond, we perform disembarkation prediction of bus passengers with a look-ahead time, $T_l = 2T_D$ mins. To handle possible incorrect predictions where a vehicle being allocated to *anticipated* passengers who don't show up eventually, we set an appropriate *expiry* parameter to free up the resource.

Figures 15 and 16 plot the average wait time (across all disembarking passengers) for the two strategies as a function of the number of MoD vehicles, for the simplest case $C = 1$. We simulated two cases: (a) with $T_D = 5$mins and (b) $T_D = 2$mins (the latter one corresponds to a realistic last-mile travel distance of 300 meters, assuming an MoD speed of 10km/h). We see that our proactive approach (S2) results in short wait times—an average of < 30secs for $T_D = 2$, and < 2mins for T_D=5, when the number of MoD vehicles is sufficient (≥ 30). More importantly, this wait time is around 75% lower than that experienced by the baseline Reactive approach (S1).

In practice, we expect MoD vehicles to be *shared* by a number of passengers, and not be allocated solely for a single passenger per trip. In Figure 17, we plot the average wait times for both strategies with $T_D = 2$mins, and the capacity (C) of the vehicle being varied from 1 to 3. We simplify the assignment task by clumping together the $C-$closest passengers arriving, or expected to arrive, in a greedy manner. As observed previously, the proactive approach provides a significant reduction in wait times across all values of C; as expected, higher values of C achieve comparable wait times with fewer vehicles. For instance, with $C = 3$, the number of resources required reduces by two-thirds (from 30 to 10) for a comparable 75%

reduction in wait time. As the assignments happen every epoch (i.e., a minute in our case), it is possible that the last vehicle to be assigned during an epoch to be not filled to its maximum capacity – in other words, if the number of remaining unassigned passengers is less than C, the passengers are assigned an available vehicle right away, instead of being delayed till the next epoch.

6 URBAN EVENT ANOMALY DETECTION

We now detail *NE-Pred*, which uses the *BuScope* system (which provides live tracking of regular vs. irregular passengers on each operating bus) to enable detection of urban events. At a high level, our approach is to first identify commuter-level anomalies (irregular trip patterns–i.e., those with zero prior support) for each bus, assign an anomaly score to each bus & bus-stop based on such anomalies, and finally perform spatial aggregation across all the buses & multiple bus-stops to identify and localize such events. Our proposed approach has three components:

- **Event Detection:** We show that bus-stops near an event venue see a sharp spike in the volume of either boarding or disembarkation by irregular users, *across multiple bus services that transit through those bus-stops*, and thus derive an anomaly threshold-based strategy to detect such events.
- **Event Localization:** We then derive a 2-D agglomerative clustering approach (that computes a cluster representative in space domain) to identify the location of such detected events. By using the *cascade anomaly* we detect the onset of anomaly by propagating the bus-stop and epoch specific anomaly scores to the downstream bus stops. The goal here is to minimize the spatial error and detect the events well in advance.
- **Event Prediction:** Finally, by using a novel *time-shifted anomaly extrapolation* technique, a variant of cascade anomaly described earlier, where bus-specific anomalies are *temporally* extrapolated to future downstream bus-stops, we show that we can identify the start-time and location of such events well in advance (well before visitors begin to show anomalous disembarkation patterns *at* the event venue).

6.1 Computing Anomaly Scores

Both event detection and localization rely on a fundamental technique: computing the anomaly score/contribution from a bus that has just visited a specific bus-stop. Intuitively, at a given bus-stop, if the bus sees either passenger disembarkations (visitors heading to an event) or boardings (perhaps residents avoiding an event) that are *irregular*, its anomaly score will be higher and indicative
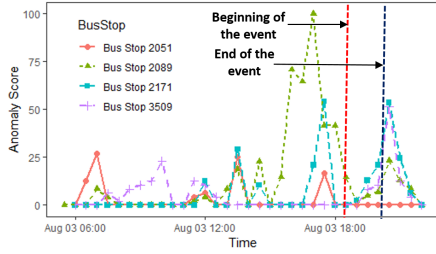
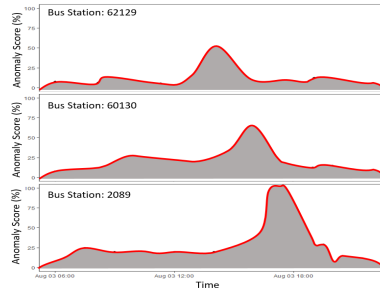Figure 18: Temporal variation of $Anom(.)$ at nearby bus stops (NDR)

Figure 19: Spatial Evolution of Anomaly Score (NDR)

Figure 20: Predictive forecasting (bus 36, bus stop 2051) on 08/03/13

of an event in the vicinity of that bus-stop. This anomaly score is updated for each bus and assigned to the corresponding bus-stop after each such bus-stop crossing. To mathematically express the computed anomaly, for a given bus b, at stop s, let us denote by o_i the interaction of an *irregular individual* (i.e., an individual with zero support) i. (For simplicity we will drop indices b and s; the score is computed in the same way for each bus at each bus-stop.) The interaction value $o_i = 1$ if the irregular individual exhibits one of two possible interactions with b at s, = {embark,disembark} (action set denoted by = {e, d} for short). On the other hand, $o_i = 0$ if i does not interact with the bus at the stop, i.e. i is not observed at the bus-stop or i simply remains on the bus, having boarded earlier and heading to a different stop.

The definition of an irregular interaction is driven by the support of the corresponding events (embarkation or disembarkation) at s for the corresponding context (i.e., the AM/PM & peak/off-peak time-bins). Of course, additional contextual states (e.g., the weather) may affect such commuting patterns and should ideally be incorporated for our specific dataset, note that climatic conditions are relatively stable across Singapore in August (a fairly dry month).

The anomaly count at a given bus stop s for a bus instance b is simply given as number of non-regular commuters who interact with b (i.e., board or alight) at s. The nominal anomaly score $A(s, t)$, for the bus-stop s is then computed, in units of time Δ (=30 mins), by aggregating the anomaly count of all buses that traverse bus stop s during the time $[t - \Delta, t + \Delta]$. The final anomaly score of a stop s at time t, denoted by $Anom(s, t)$ is obtained as the bus-stop specific, *normalized* deviation of the current score, i.e., $Anom(s, t) = \frac{A(s,t) - min_{\forall \tau} \{A(s, \tau)\}}{max_{\forall \tau} \{A(\tau)\} - min_{\forall \tau} \{A(\tau)\}}$.

6.2 Experimental Results

To quantitatively evaluate our event anomaly detection algorithm, we consider a relatively small set of events (tabulated in Table 3) whose occurrence during August 2013 were well documented[7].

6.2.1 Detecting Urban Event Anomalies. Figures 18 plots the temporal variation of anomaly scores of neighboring bus-stops for the NDR event day. We see that $Anom(s)$ is appreciably higher at those stops, close to the event start (as well as end) times. (In fact, while further discrimination between event start vs. end is

Event	Location	Date & Time	Spatial Error (m)	Look-ahead time (mins)	
				Cascade	Spot
National day rehearsal (NDR)	Float @ Marina bay	3-Aug 06:30-08.30PM	345.21	60	20
National day parade (NDP)	Float @ Marina bay	9-Aug 06:30-08.30PM	376.51	210	130
Franz Schubert Piano sonatas (SCH)	SOTA Concert Hall	24-Aug 07:30-08.30PM	669.71	30	NA

Table 3: Summary of Events and Localization Results

possible by differentiating between e and d interactions, we omit this discussion for space reasons.) In general, we observe that a rule "$Anom(s) \geq 50\%$ for two consecutive intervals t_i, t_{i+1}" helps us to accurately identify all 3 representative events.

6.2.2 Event Anomaly Localization. We now describe our clustering-based strategy for spatial event localization. For each bus-stop s, let $t_p(s)$ denote the time at which $Anom(s)$ peaks, while l_s represents the 2-D location of s. We employ a greedy hierarchical agglomerative technique for spatial event localization. Intuitively, we start by *merging* the two bus-stops with the larger $Anom(s)$ values into a single cluster, after which we iteratively pick the bus-stop (among the set of bus-stops remaining to be clustered) with the highest value of $Anom(s)$ and merge it with our cluster. The merging operation involves computing the weighted centroid of the location (l_s) and the current cluster.

To identify the start time of an anomaly, we adopt a cascading technique where $Anom(s, t)$ of a specific bus stop s and bus b at time t is propagated to all its downstream bus stops and re-assigned at each bus stop-crossing. We continually update this anomaly score (for each bus stop) at each successive epoch (with Δ = 30 mins); an anomaly is then declared to have "started", when a bus stop's score, $Anom(s, t)$ exceeds the threshold for 2 consecutive epochs.

Table 3 shows the resulting spatial error and the look-ahead time for all 3 events. We see that the spatial error is around 350-400 meters (roughly $\approx$ 1.5 bus-stops) for the larger-scale National Day events. The slightly higher error for *SCH* may be explained by noting that the event was approx. 200 meters distant from a major station, where many visitors probably disembarked and then walked to the venue. We also note that the cascading technique yields greater look-ahead time (varying between 60 and 210 minutes) for macro events (NDR and NDP) as compared to the micro events (SCH), most likely because, at large events, visitor arrivals peak well before the event start–e.g., on the NDP day, hordes of visitors arrive at least 3 hours ahead to obtain favorable viewing spots.

To further demonstrate the advantages of the cascading approach, we also introduce a naive baseline strategy called *Spot Anomaly*, where an anomaly is defined for each bus stop in isolation (based on changes solely in that bus stop's disembarkation volume. In Table 3, we report the look-ahead time of this baseline – clearly, our *Cascade* method detects the macro events well in advance (40-80 minutes before) of the baseline. Note also that, unlike *Spot*, *Cascade* was able to detect the micro event (SCH). As explained earlier, this limitation may be attributed to the fact that *SCH* was located near a major transit hub. In such a case, the overall change in disembarkation volume at a busy 'sink' node is likely to be insignificant, causing *Spot* to fail. However, *Cascade's* technique of using abnormal occupancy on multiple bus routes can isolate such low-intensity events.

6.2.3 Event Prediction. The results above show that we can in fact use the implicit signals from bus commuting patterns to detect an event's location and look-ahead time with high accuracy. However, we now show that we can achieve something even more powerful: we can predict the start time of an unknown event *well in advance.* The key idea is as follows: bus passengers travelling to participate in an event will often board the buses well in advance (e.g., the average commute from Singapore's residential heartland to the downtown area is over 45 minutes). By effectively propagating such anomalous boarding signals to downstream bus-stops, we can identify the possible future start-time and location of such events. More specifically, our algorithm operates as follows:

- Compute the anomaly score $Anom(b, s, t)$ for a given bus b that traverses bus-stop s at current time t.
- Based on the estimated travel time (denoted as $T(s, \hat{s})$ of bus b to a downstream stop $\hat{s}$), propagate this anomaly score to $\hat{s}$ for future time-instant–i.e., let
  $PredAnom(b, \hat{s}, t + T(s, \hat{s}))= Anom(b, s, t).$
- For each downstream bus-stop $\hat{s}$, aggregate anomaly scores across all buses that will travel to $\hat{s}$.
- If the predicted anomaly score at any bus-stop $\hat{s}$ exceeds the threshold at any future time $t + T$, then declare "event likely at $\hat{s}$ at time $t + T$".

Figure 20 illustrates this concept of predictive anomaly score propagation, using a specific bus-service (No. 36) on the day of the NDR event. We can see that the predicted anomaly score exceeds our threshold (50%) for two consecutive periods at 5pm, and identifies the event start-time as 5.30pm. In other words, we are able to *correctly predict the occurrence of the event 1.5 hours in advance.* Similar results hold for the other events, demonstrating the promise of our proposed method.

7 DISCUSSION

There are several aspects of live bus ridership analytics that need additional investigations.

Threats to Validity: As mentioned previously, it is possible for passengers to pay their fare by cash to the driver, in which case they are essentially invisible to our analysis. While relatively rare (only 4% of trips[8] involve cash), certain groups of commuters (e.g.,

overseas tourists on short trips) may favor such transactions. Our analytical results may consequently be less accurate for locations disproportionately favored by tourists. Also, the predictions on commuter demand patterns should ideally made more holistically, factoring in other modes of transport (e.g., the train network, private on-demand buses, etc.).

Other Application Scenarios: Live, predictive disembarkation prediction can enable other types of smart transportation services. For example, commuters often use transportation Apps[9] that provide "live" feeds of bus arrival times and crowdedness. By using disembarkation prediction, such Apps can provide a commuter, waiting at a particular bus-stop, a more accurate, *anticipated* crowdedness of an en-route bus, as opposed to simply displaying the current crowd levels. Figure 21 plots the average error in occupancy prediction using our Hybrid prediction technique–we see that the average error in predicting ridership at downstream bus-stops is almost always quite low (<2 persons), and may be thus used to enhance such transportation Apps.
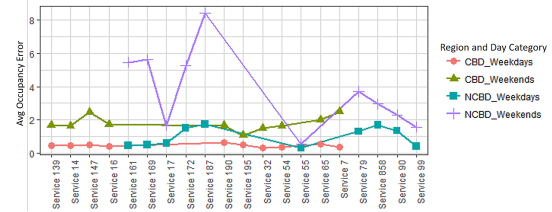
Figure 21: Accuracy of ridership estimation

Smarter MoD Allocation Strategies: We must emphasize that the benefit of lower last-mile wait times illustrated here has been done using a fairly straightforward MoD simulator model. Significant opportunities for optimizing the MoD resource allocation exist–for example, the vehicle assignment may be dynamically updated based on the real last-mile travel distances. Our goal here was not to present a preferred strategy, but simply to empirically demonstrate that disembarkation predication can significantly improve the last-mile commuting experience.

Data Privacy: The use of even pseudonymized data (as we do) can raise possible privacy concerns, such as the possible recovery of a user's identity from detailed individual level mobility traces–e.g., a daily pattern of early morning embarkations and evening alightings would identify the "home" bus stop for a pseudo-identifier. Clearly, there is a risk of privacy compromise by possibly cross-linking such inferences to publicly side available information (e.g., [37]). To get an initial sense of this problem, we conducted a preliminary assessment of the $k - anonymity$ of a typical 'terminal' bus stop–i.e., we ask: "on average, how many unique customers would have the same bus stop as their *'home'*"? For a specific neighborhood, we first extracted the locations of residential blocks and used a *Nearest Neighbor classifier* to assign them to a set of predefined clusters (bus stops are the centroids). By then estimating the total population within each cluster and multiplying it by the bus ridership ratio (≈ 0.32)[10], we find that the k-anonymity values can range from 24

---

[8]https://www.researchgate.net/publication/266878969_use_of_public_transport_smart_card_fare_payment_data_for_travel_behavior_analysis_in_singapore

[9]https://busleh.originallyus.sg/;
https://play.google.com/store/apps/details?id=com.iridianstudio.sgbuses&hl=enSG
[10]https://data.gov.sg/group/transport

– 392 (for low housing density areas such as Marine Parade) to around 79 – 1500 (for more mature residential neighborhoods, such as Toa Payoh). These results suggest that bus stop-level data may typically not be trivial to de-anonymize–however, more careful assessment of privacy vs. data granularity (and its impact on our analytics) remains an area for further research.

8 RELATED WORK

The widespread availability of city-scale mobility data (obtained via GPS/WiFi [23, 41] traces, taxi ridership [6] or bike trip [38] records, public transport data [15, 36]) have driven significant prior research on urban mobility analytics.

Human Mobility Prediction Prior work has shown that human macro-scale mobility is regular and predictable in both spatial and temporal instances [16, 24, 32, 33]. Regular and frequent visiting patterns (e.g., home, work and supermarket) enables the ability to predict human mobility with high accuracy [10, 20]. In particular, in [13] the authors show the existence of individual-level frequent and routine visits to a few locations. In [29], it is shown that the human mobility is accurately predictable on college campuses (93% accuracy). Works such as those of Becker et al.[2] utilize metropolitan-scale mobility data (from CDR records) to characterize population movement for use cases such as commute time predictions and disease spreading. Besides characterizing such predictability, researchers have also worked on predicting future locations based on historical mobility traces. In recent past Markov models are proposed to predict the future location [9, 18, 30, 34, 35] in both indoors and outdoors [11] by utilising historical transition between places. The authors in [12] complemented the historical traces by leveraging various contextual information inferred by exploiting various sensors (bluetooth, accelerometer etc). In this work, we leverage the existence of 'regularity' in human mobility patterns (observed only sporadically through public transport usage) at both individual and collective scales.

Demand Prediction of Urban Transport Networks Prior works focused on analyzing the demand of public transport by modelling spatiotemporal historical demand and availability of vehicles [4, 6, 19]. Similarly, demand estimation in ride sharing/ride hailing (i.e., MOD) has garnered significant attention, specially after the emergence of services such as Uber. In such environments, the focus was often on predicting the arrival rate of the passengers at a given location to re-position the vehicles to cater future demand [8, 27, 40]. Closest to our work, Balan et al.[1] provide real-time trip information services based on historic trips (e.g., fare and distance estimates of similar trips in the past), and discuss use cases such as anomaly detection – our work is different in that we focus on soft-real time guarantees but operate on live, city-scale streaming mobility data.

Urban Event Detection Work here can be classified in sub-domains of urban event detection and prediction and anomaly detection in transport (particularly in road networks). For event detection, works such as CitySense [23] utilise aggregated GPS traces collected using a mobile application to detect hot-spots and anomalies/outbreaks. This approach is similar to our *Spot* anomaly baseline which looks at aggregate disembarkations to detect outliers. Konishi et al.[17] have recently proposed an approach to

predict irregularities (e.g., large scale events) ahead of time using a two-step modeling process. By querying route information using a mobile transit App, the authors model short and long term population models using auto-regression and bi-linear Poisson regression, respectively. Similarly, social media data has been used to detect and track earthquakes from user posted information on Twitter [28] and to detect and characterize urban events from text, images and metadata [14].

Previous works on transportation anomaly detection have looked at varied aspects such as detection of anomalies, understanding the spatiotemporal ordering and finding root causes. Pang et al. [26] detect contiguous, spatiotemporal cells as anomalous regions using Likelihood Ratio Tests. Further, Liu et al. [22] proposed a formulation for "causal outlier detection" for detecting the emergence, propagation and disappearance of outliers (e.g., traffic jam). Subsequently, Chawla et al. [5] identified routes in a road network with anomalous traffic using a 2-step approach: (1) first they detect anomalous links using Principal Component Analysis (as seen in many works on network traffic anomaly detection) and (2) using a link-route matrix, they detect which routes were root causes for the detected anomalies using L1 machinery. In contrary, our work aims to detect and localize events by specifically exploiting the inherent 'regularity' of *individual-level* human mobility.

9 CONCLUSIONS

We have described *BuScope*, a system that supports soft-real time processing of public bus commuting data. From the analysis of such individualized bus trip records, we have shown that the destination for most trips has high predictability, even on routes where the commuter has made only one past journey. Subsequently, by combining individualized and flow-based predictions, we show that we can predict a commuter's disembarkation bus-stop with an accuracy of over 85% and a mean error of less than 1-2 bus-stops. We have then shown how such collective predictions can be used in a last-mile MoD system, where unmanned vehicles are pre-positioned to respond to anticipate disembarkation demand, resulting in an ≈75% decrease in commuter wait times. We have also shown how the real-time detection of irregular commuters, along multiple bus routes, can be used to detect urban events with high spatial accuracy (≈450 meter error), well in advance (100 mins) of the event start time. We anticipate that this work will motivate public agencies to view mobility data as not just a policy planning resource, but as an enabler of a new class of *live* smart city services.

10 ACKNOWLEDGMENT

REFERENCES

[1] Rajesh Krishna Balan, Khoa Xuan Nguyen, and Lingxiao Jiang. 2011. Real-time trip information service for a large taxi fleet. In *Proceedings of the 9th international conference on Mobile systems, applications, and services.* ACM, 99–112.

[2] Richard Becker, Ramón Cáceres, Karrie Hanson, Sibren Isaacman, Ji Meng Loh, Margaret Martonosi, James Rowland, Simon Urbanek, Alexander Varshavsky, and Chris Volinsky. 2013. Human mobility characterization from cellular network data. *Commun. ACM* 56, 1 (2013), 74–82.

[3] Pablo Samuel Castro, Daqing Zhang, and Shijian Li. 2012. Urban traffic modelling and prediction using large scale taxi GPS traces. In *International Conference on Pervasive Computing.* Springer, 57–72.

[4] H. Chang, Y. Tai, and J. Y. Hsu. 2010. Context-aware Taxi Demand Hotspots Prediction. *Business Intelligence and Data Mining* (2010).

[5] Sanjay Chawla, Yu Zheng, and Jiafeng Hu. [n. d.]. Inferring the Root Cause in Road Traffic Anomalies. In *Proceedings of the 2012 IEEE 12th International Conference on Data Mining (ICDM '12).*

[6] M. F. Chiang, T. A. Hoang, and E. P. Lim. 2015. Where are the Passengers? A Grid-based Gaussian Mixture Model for Taxi Bookings. In *ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL).*

[7] Pieter Colpaert, Alvin Chua, Ruben Verborgh, Erik Mannens, Rik Van de Walle, and Andrew Vande Moere. 2016. What public transit API logs tell us about travel flows. In *Proceedings of the 25th International Conference Companion on World Wide Web.* International World Wide Web Conferences Steering Committee, 873–878.

[8] N. Davis, G. Raina, and K. Jagannathan. 2016. A Multi-level Clustering Approach for Forecasting Taxi travel Demand. In *Intelligent Transportation Systems (ITSC).*

[9] Nathan Eagle and Alex Pentland. 2006. Reality mining: sensing complex social systems. *Personal and ubiquitous computing* 10, 4 (2006), 255–268.

[10] N. Eagle and A. S. Pentland. 2009. EigenBehaviors: Identifying Structure in Routine. *Behavioral Ecology and Socio-biology* (2009).

[11] Sébastien Gambs, Marc-Olivier Killijian, and Miguel Núñez del Prado Cortez. 2012. Next place prediction using mobility markov chains. In *Proceedings of the First Workshop on Measurement, Privacy, and Mobility.* ACM, 3.

[12] João Bártolo Gomes, Clifton Phua, and Shonali Krishnaswamy. 2013. Where will you go? mobile data mining for next place prediction. In *International Conference on Data Warehousing and Knowledge Discovery.* Springer, 146–158.

[13] M. C. Gonzalez, C. A. Hidalgo, and A. L. Barbasi. 2008. Understanding Individual Human Mobility Patterns. *Nature* (2008).

[14] Kasthuri Jayarajah and Archan Misra. 2016. Can Instagram posts help characterize urban micro-events?. In *Information Fusion (FUSION), 2016 19th International Conference on.* IEEE, 130–137.

[15] Kasthuri Jayarajah, Vigneshwaran Subbaraju, Noel Athaide, Lakmal Meegahapola, Andrew Tan, and Archan Misra. 2018. Can Multimodal Sensing Detect and Localize Transient Events?. In *Proceedings Volume 10635, Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR IX (SPIE Defense + Security '18).*

[16] M. Kim and D. Kotz. 2007. Periodic Properties of User Mobility and Access-point Popularity. *Pervasive and Mobile Computing* (2007).

[17] Tatsuya Konishi, Mikiya Maruyama, Kota Tsubouchi, and Masamichi Shimosaka. 2016. CityProphet: City-scale Irregularity Prediction Using Transit App Logs. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16).*

[18] J. Krumm and E. Horovitz. 2006. Predestination: Infferring destinations from partial trajectories. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp).*

[19] J. Li, I. Shin, and G. L. Park. 2008. Analysis of Passenger Pick-up Pattern for Taxi Location Recommendation. In *NCM.*

[20] Z. Li, B. Ding, J. Han, R. Kays, and P. Nye. 2010. Mining Periodic Behaviors for Moving Objects. In *ACM SIGKDD.*

[21] Liang Liu, Anyang Hou, Assaf Biderman, Carlo Ratti, and Jun Chen. 2009. Understanding individual and collective mobility patterns from smart card records: A case study in Shenzhen. *12th International IEEE Conference on Intelligent Transportation Systems (ITS)* (2009), 1–6.

[22] Wei Liu, Yu Zheng, Sanjay Chawla, Jing Yuan, and Xie Xing. [n. d.]. Discovering Spatio-temporal Causal Interactions in Traffic Data Streams. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '11).*

[23] Markus Loecher and Tony Jebara. 2009. CitySense: Multiscale space time clustering of gps points and trajectories. In *Proceedings of the Joint Statistical Meeting.*

[24] E. M. R. Oliveira, A. C. Viana, C. Sarraute, J. Brea, and I. A. Hamelin. 2015. On The Regularity of Human Mobility. *Pervasive and Mobile Computing* (2015).

[25] Gang Pan, Guande Qi, Zhaohui Wu, Daqing Zhang, and Shijian Li. 2013. Land-Use Classification Using Taxi GPS Traces. *IEEE Transactions on Intelligent Transportation Systems* 14 (2013), 113–123.

[26] Linsey Xiaolin Pang, Sanjay Chawla, Wei Liu, and Yu Zheng. [n. d.]. On Mining Anomalous Patterns in Road Traffic Streams. In *Proceedings of the 7th International Conference on Advanced Data Mining and Applications - Volume Part II (ADMA'11).*

[27] M. Pavone, S. L. Smith, E. Frazzoli, and D. Rus. 2012. Robotic Load balancing for Mobility On-demand Systems. *Robotics Research* (2012).

[28] T. Sakaki, M. Okazaki, and Y. Matsuo. [n. d.]. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors. In *Proceedings of the 19th International Conference on World Wide Web (WWW '10).*

[29] C. Song, Z. Qu, N. Blumm, and A. L. Barbasi. 2010. Limits of Predictability in Human Mobility. *Science* (2010).

[30] Libo Song, David Kotz, Ravi Jain, and Xiaoning He. 2003. Evaluating Location Predictors with Extensive Wi-Fi Mobility Data. *SIGMOBILE Mob. Comput. Commun. Rev.* 7, 4 (Oct. 2003), 64–65.

[31] Krygsman Stephan, Dijst Martin, and Arentze Theo. 2004. Multimodal public transport: an analysis of travel time elements and the interconnectivity ratio. In *Transport Policy.* Elsevier.

[32] D. Wang, D. Pedreschi, C. Song, F. Giannotti, and A. L. Barbasi. 2011. Human Mobility, Social Ties and Link Prediction. In *ACM SIGKDD.*

[33] Y. Wang, N. J. Yuan, D. Lian, L. Xu, X. Xie, E. Chen, and Y. Rui. 2015. Regularity and Conformity: Location Prediction Using Heterogeneous Mobility Data. In *ACM SIGKDD.*

[34] A. Y. Xue, J. Qi, X. Xie, R. Zhang, J. Huang, and Y. Li. 2015. Solving the data sparsity problem in destination prediction. *Very Large DataBase (VLDB)* (2015).

[35] A. Y. Xue, R. Zhang, Y. Zheng, X. Xie, J. Huang, and Z. Xu. 2013. Destination prediction by sub-trajectory synthesis and privacy protection against such prediction. In *ACM International Conference on Data Engineering (ICDE).*

[36] N. J. Yuan, Y. Wang, F. Zhang, X. Xie, and G. Sun. 2013. Reconstructing Individual Mobility from Smart Card Transactions. In *IEEE International Conference on Data Mining (ICDM).*

[37] Hui Zang and Jean Bolot. 2011. Anonymization of Location Data Does Not Work: A Large-scale Measurement Study. In *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (MobiCom '11).* ACM.

[38] H. Zhang, Y. Zheng, and Y. Yu. 2018. Detecting Urban Anomalies Using Multiple Spatio-Temporal Data Sources. *Journal of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* (2018).

[39] Junbo Zhang, Yu Zheng, Dekang Qi, Ruiyuan Li, Xiuwen Yi, and Tianrui Li. 2018. Predicting citywide crowd flows using deep spatio-temporal residual networks. *Artificial Intelligence* (2018).

[40] R. Zhang, F. Rossi, and M. Pavone. 2016. Model Predictive Control of Autonomous Mobility On-demand Systems. In *Robotics and Automation.*

[41] M. Zhou, M. Ma, Y. Zhang, K. Suia, S. Pei, and T. Moscibroda. 2016. EDUM: Classroom Education Measurements via Large-scale WiFi Networks. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp).*