

UrbanCPS: a Cyber-Physical System based on Multi-source Big Infrastructure Data for Heterogeneous Model Integration

Desheng Zhang
zhang@cs.umn.edu
University of Minnesota, USA

Juanjuan Zhao, Fan Zhang
{jj.zhao, zhangfan}@siat.ac.cn
SIAT, Chinese Academy of Sciences

Tian He
tianhe@cs.umn.edu
University of Minnesota, USA

Abstract

Data-driven modeling usually suffers from data sparsity, especially for large-scale modeling for urban phenomena based on single-source urban infrastructure data under fine-grained spatial-temporal contexts. To address this challenge, we motivate, design and implement UrbanCPS, a cyber-physical system with heterogeneous model integration, based on extremely-large multi-source infrastructures in a Chinese city Shenzhen, involving 42 thousand vehicles, 10 million residents, and 16 million smartcards. Based on temporal, spatial and contextual contexts, we formulate an optimization problem about how to optimally integrate models based on highly-diverse datasets, under three practical issues, *i.e.*, heterogeneity of models, input data sparsity or unknown ground truth. Based on an integration of five models, we propose a real-world application called Speedometer, inferring real-time traffic speeds in urban areas. The evaluation results show that compared to a state-of-the-art real-world system, Speedometer increases the inference accuracy by 21% on average.

1 Introduction

The recent advance of urban infrastructures increases our ability to collect, analyze and utilize big infrastructure data to improve urban phenomenon modeling [18]. Numerous data-driven models have been proposed based on these infrastructure data to capture urban dynamics [2] [13] [16]. However, although each infrastructure produces abundant data, almost all resultant models suffer from data sparsity [18]. This is because urban phenomena are typically in an extremely large scale, and so it is almost impossible to collect complete data about a particular phenomenon under fine-grained spatial-temporal contexts. For example, traffic speeds can be modeled by GPS data from taxicabs [2], but under fine-grained spatial-temporal contexts, such a speed model suffers from data sparsity. As shown by our empirical analysis on a Chinese city Shenzhen, given a middle-length time slot of five minutes during 24 hour a day, 57% of its 110-thousand road segments on average do not have any taxicab, which leads to data sparsity.

In this work, we argue that with increasing updates of urban infrastructures, one urban phenomenon can be separately modeled by many *heterogeneous* infrastructure datasets. For example, a traffic speed can be directly modeled by vehicle GPS data and loop detector data [2], or indirectly modeled by cellphone and transportation smart-

card data [10]. Integrating these relevant yet heterogeneous models can provide complementary predictive powers by combining the expertise of heterogeneous infrastructures, which is used to address data sparsity issues about single infrastructures. Although many effective models have been proposed based on infrastructure data, they are typically based on single-source data [2] [10] [3] [11]. Due to various technical and logistical reasons, little work, if any, has been done to integrate single-source heterogeneous models into a unified multi-source model based on large-scale infrastructure data (TB level data) to address practical issues, *e.g.*, sparse data, for real-world applications.

To this end, we motivate and design UrbanCPS, a CPS system with a generic heterogeneous-model integration based on extremely-large infrastructure data. In UrbanCPS, we implement five heterogeneous models based on data from five infrastructures of Shenzhen (the most crowded city in China with 17,150 people per KM^2), including a 14 thousand taxicab network, a 15 thousand truck network, a 13 thousand bus network, a 10 million user cellular network, and an automatic fare-collection system with 17 thousand smartcard readers and 16 million smartcards. Based on these five highly-diverse heterogeneous models, we propose a model-integration technique to address their data sparsity, *e.g.*, integrating traffic-speed models based on vehicles data and urban-density models based on cellphone data. However, we face three challenges as follows.

1. Among all heterogeneous models, some models are only indirectly relevant to a particular phenomenon of interest, *e.g.*, an urban-density model is only indirectly relevant to traffic speeds. Thus, it is challenging to effectively integrate directly-relevant models with indirectly-relevant models due to their heterogeneity.
2. Indirectly-relevant models normally cannot output a measurement about phenomena of interest directly. Thus, even with complementary knowledge from indirectly-relevant models, it is a non-trivial problem to solve data sparsity for directly-relevant models.
3. During a model integration, different models have different weights under different temporal, spatial and contextual conditions, and the optimal weights are usually obtained by regression with the ground truth. But the ground truth of urban scale phenomena is almost impossible or really expensive to be obtained.

A unique combination of the above three challenges makes our work significantly different from the previous model integration, where integrated models are often homogenous and based on complete data with known ground truth. The key contributions of the paper are as follows:

- We propose the first generic CPS system UrbanCPS

with heterogeneous model integration based on metropolitan-scale multi-source infrastructure data. To our knowledge, the integrated models have by far the highest standard for urban modeling in two aspects: (i) modeling based on most complete infrastructure data including cellular, taxicab, bus, subway and truck data for the same city, and (ii) modeling based on the largest residential and spatial coverage (*i.e.*, 95% of 11 million permanent residents and 93% of 110 thousand road segments in Shenzhen). The sample data are given in [1].

- We theoretically formulate an optimization problem to integrate heterogeneous models. We propose a technique to dynamically measure heterogeneous-model similarity on phenomena of interest under different temporal, spatial and contextual conditions to address three practical issues as follows: (i) how to integrate indirectly-relevant heterogeneous models; (ii) how to use an integrated model to address data sparsity; (iii) how to assign weights to different models without a regression process based on the ground truth.
- We design and implement a real-world application called Speedometer, which infers real-time traffic speeds in urban areas based on an integration of five models built upon taxicab, bus, truck, cellphone, and smartcard-reader networks. We test UrbanCPS based on a comprehensive evaluation with 1 TB real-world data in Shenzhen. The evaluation results show that compared to a current system, UrbanCPS increases the inference accuracy by 21% on average.

We organize the paper as follows. Section 2 gives our motivation. Section 3 presents the UrbanCPS. Section 4 describes our model integration. Section 5 validates UrbanCPS with a real-world application, followed by the related work and the conclusion in Sections 6 and 7.

2 Motivation

To show our motivation, we compare two traffic-speed models built upon large-scale empirical data we collected in Shenzhen. The first model is called SZ-Taxi [14], which is a real-world system deployed and maintained by Shenzhen Transport Committee to infer real-time traffic speeds based on taxicab GPS data in Shenzhen. The second model is called TSE [13], which is a state-of-the-art traffic model in the research community based on vehicle GPS data. We feed our bus and truck GPS data to TSE and obtain two models TSE-Bus and TSE-Truck, respectively. The details are given in Section 5.2. As in Figure 1, we compare three models based on taxicab, bus and truck data to the ground truth on a major road segment in Shenzhen called Shahe Road in 5-min slots during a regular Monday.

In general, all three models have data sparsity issues, *i.e.*, among a total of 288 5-min slots, SZ-Taxi, TSE-Bus and TSE-Truck have data on 87, 49 and 39 slots, *i.e.*, 30%, 17% and 14%, restrictively. If the data are all complete for all three models, we should have 24 points for every model, *i.e.*, a total of 72 points, for every red box covering a 2 hour period, but we have much fewer than 72 points as shown in

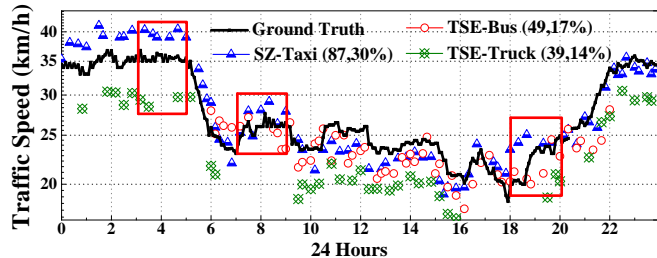


Fig 1. Inferred Traffic Speeds by Three Models

Figure 1. (i) SZ-Taxi has a major data sparsity issue during the early morning when no taxicabs are on this road segment. Further, it typically overestimates the speed in the nighttime since taxicab drivers typically drive much faster than regular drivers in the nighttime when passengers are few, but it underestimates the speed in the daytime due to frequent stopping for pickups and dropoffs as well as long-time waiting for passengers. (ii) TSE-Bus has sparse data in the nighttime when the bus service is not available, and in some regular daytime. Further, it underestimates the speed in the non-rush hour due to frequent stops, but it overestimates the speed in the rush hour because of dedicated fast traffic lines for bus only. (iii) TSE-Truck has sparse data in the morning and evening rush hour, because trucks are forbidden to use several major roads during the rush hour to relief traffic congestion. Even for the time period where trucks are allowed, it still has such an issue. Also, it usually underestimates the speed during other times due to the speed limit of trucks. Note that this road segment was selected as one of ten major road segments in Shenzhen, but we still face major data sparsity issues, which are much worse on other small road segments where taxicabs, buses or trucks are much fewer as shown in Section 3.2.

A seemingly promising solution is to integrate these three models to address data sparsity issues from a *homogenous* complimentary view. However, such a straightforward homogenous-model integration may still face data sparsity issues due to their inherent homogeneity, *e.g.*, all three models have incomplete data in common slots in the red boxes. In this work, we address this challenge by introducing other *heterogeneous* models (*e.g.*, urban-density models) based on different datasets (*e.g.*, cellphone data) under the observation that the traffic speed is correlated with urban density in same spatial-temporal contexts [4]. Based on these heterogeneous models, we propose an integration technique in a reference implementation of an extremely-large CPS system, which presented as follows.

3 Urban Cyber Physical System

Broadly, a CPS can be considered as a system of systems. Therefore, in this work, we consider a set of urban infrastructure systems (*e.g.*, cellular, taxicab, bus, subway and truck networks) as a Urban Cyber Physical System (UrbanCPS) from a broad perspective: any device in urban infrastructures is considered as a pervasive sensor in Urban CPS, if it generates data that can be used to build a model to describe phenomena of interest. Built upon an integration of

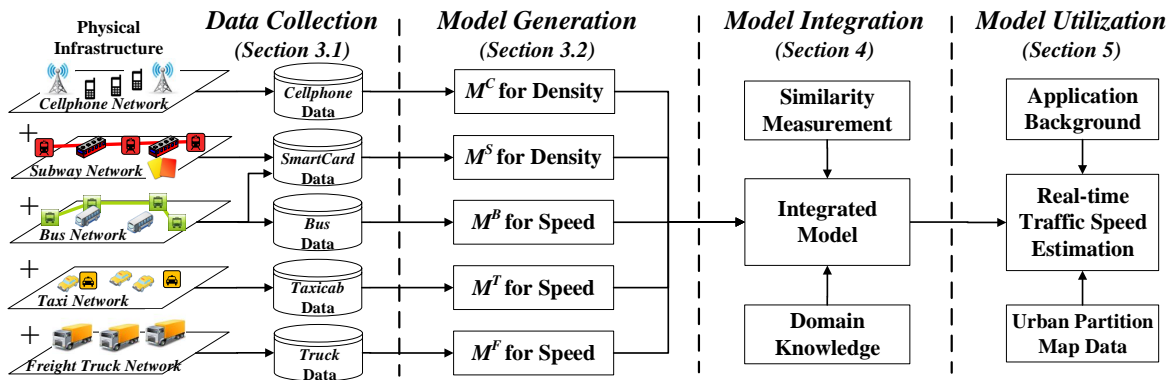


Fig 2. Urban Cyber Physical System

models based on multiple data sources, UrbanCPS provides unseen urban dynamics under extremely fine-grained spatio-temporal resolutions to support real-world applications, which cannot be achieved by any model from single data source in isolation, *e.g.*, a monolithic infrastructure. In Figure 2, we outline UrbanCPS with four components, *i.e.*, Data Collection, Model Generation, Model Integration and Model Utilization. These four components span the whole data-processing chain in UrbanCPS.

As in Figure 2, we provide a road map for the rest of paper as follows. (i) In Section 3.1, we first introduce the data collection where we individually collect multiple-source data from urban infrastructures of Shenzhen. (ii) In Section 3.2, we generate various heterogeneous models based on collected single-source data. (iii) In Section 4, we effectively combine these heterogeneous models by our model integration based on their similarity and domain knowledge. (iv) In Section 5, to close the control loop, we propose an application to estimate real-time traffic speeds based on integrated models and other supporting data, *e.g.*, map data and urban partition data. We envision that urban residents would use this application to find efficient routes, which in turn provides feedback to urban infrastructures. As a result, with the highlights on extremely-large data collection and highly-generic heterogeneous model integration, UrbanCPS builds an architectural bridge between multiple domain-independent urban infrastructures and real-world knowledge output tailored by applications.

3.1 Data Collection

In our project, we have been collaborating with several service providers and the Shenzhen Transport Committee (hereafter STC) for the real-time access of urban infrastructures. In Figure 2, we consider five kinds of devices in this version of implementation, which detects urban dynamics from complimentary perspectives.

- **Cellphones** are used to detect cellphone users' locations at cell tower levels based on call detail records. We utilize cellphone data through two major operators in Shenzhen with more than 10 million users. The cellphone data give 220 million locations per day.
- **Smartcard Readers** are used to detect locations of a total of 16 million smartcards used to pay bus and subway fares. These readers capture more than 10 million rides and 6 million passengers per day. We study read-

er data from STC, which accesses real-time data feeds of a company that operates the smartcard business.

- **Buses** are used to detect real-time traffic and bus passengers' locations by cross-referencing data of onboard smartcard readers for fare payments. We study bus data through STC to which bus companies upload their bus status in real time, accounting for all 13 thousand buses generating 2 GPS records/min.
- **Taxicabs** are used to detect real-time traffic and taxicab passengers' locations based on taxicab status (*i.e.*, GPS and occupancy). We study taxicab data through STC to which taxicab companies upload their taxicab status in real time, accounting for all 14 thousand taxicabs generating 2 GPS records/min.
- **Trucks** are used to detect real-time traffic by logging real-time GPS locations of a fleet of 15-thousand freight trucks, which travel within Shenzhen and around nearby cities. We study this truck network through a freight company that installs GPS devices on all these trunks for daily managements. Every truck uploads its real-time GPS location and driving speed back to the company server every 15s on average, which then are routed to our server.

Since our paper concentrates on system aspects, we briefly introduce our data related issues due to space limitation. We establish a secure and reliable transmission mechanism, which feeds our server the above data collected by STC and service providers with a wired connection. As in Figure 3, we have been storing a large amount of data to generate single-source models. We utilize a 34 TB Hadoop Distributed File System (HDFS) on a cluster consisting of 11 nodes, each of which is equipped with 32 cores and 32 GB RAM. For daily management and processing, we use the MapReduce based Pig and Hive. We have been finding several kinds of errant data, *e.g.*, missing data, duplicated data and data with logical errors, and thus we have been conducting a detailed cleaning process to filter out errant data on a daily basis. We protect the privacy of residents by anonymizing all data and presenting models in aggregation. In short, our endeavor of consolidating the above data enables extremely large-scale fine-grained urban phenomenon rendering based on existing single-source models, which is unprecedented in terms of both quantity and quality shown as follows.

Cellphone Dataset		Smartcard Dataset		Bus Dataset		Taxicab Dataset		Freight Truck Dataset	
Beginning	2013/10/1	Beginning	2011/7/1	Beginning	2013/1/1	Beginning	2012/1/1	Beginning	2013/9/11
# of Users	10,432,246	# of Cards	16,000,000	# of Buses	13,032	# of Taxis	14,453	# of Trucks	15,001
Size	1 TB	Size	600 GB	Size	720 GB	Size	1.7 TB	Size	1.2 TB
# of Records	19 billion	# of Records	6 billion	# of Records	9 billion	# of Records	22 billion	# of Records	16 billion
Format		Format		Format		Format		Format	
SIM ID	Date&Time	Card ID	Date&Time	Plate ID	Date&Time	Plate ID	Date&Time	Plate ID	Date&Time
Cell Tower	Activities	Device ID	Station ID	Stop ID	GPS&Speed	Status	GPS&Speed	Odometer	GPS&Speed

Fig 3. Datasets from Model Generation

3.2 Model Generation

Fellow researchers have proposed many effective single-source models [18], so we restrain ourselves from developing new models. Instead, we directly use our data to generate single-source models based on existing methods.

3.2.1 Model Summary

We implement two kinds of models based on the data collected in UrbanCPS. (i) Speed Models: including M^T , M^B , M^F , which use GPS data from Taxicab, Bus and Freight truck networks individually to estimate real-time traffic speeds. They are implemented similarly according to a state-of-the-art speed model TSE, which uses historical and real-time vehicle data as well as contexts (*e.g.*, physical features of roads) for a collaborative filtering [13]. In addition, we consider all Vehicles as a single fleet and feed its data to TSE to obtain a new model M^V . (ii) Density Models: including M^C and M^S , which use the Cellphone and Smartcard data to estimate real-time urban density (*i.e.*, count of residents). M^C is based on a population density model that predicts future CDR records based on the previous CDR records to indicate the density [10]. M^S is based on a Gaussian process-based predictive model that uses contexts, *e.g.*, time of day and day of week, to infer transit passenger density [3]. We provide a summary of these models in Table 1 based on their results in one day.

Table 1. Heterogeneous Models

Model Name	Spatial Resolution	Temporal Resolution	Resident Coverage
M^T	87% of Roads	30s	N/A
M^B	59% of Roads	30s	N/A
M^F	45% of Roads	15s	N/A
M^V	93% of Roads	7.5s	N/A
M^C	17,859 Towers	Various	95%
M^S	10,442 Stations	Various	55%

During one day, based on the GPS uploading speeds and traveling patterns, M^T , M^B , M^F , and M^V cover 87%, 59%, 45%, and 93% of all 110 thousand road segments in Shenzhen. During one day, M^C covers 95% of 11 million residents and produces their locations as one of 17,859 cell towers when they use their phones. M^S covers 55% of all residents and produces their locations as one of 10,442 transit stations when they use their smartcards.

3.2.2 Data Sparsity in Fine Granularity

Although all these models have comprehensive *daily* data, real-world applications typically require knowledge under fine-grained spatial-temporal contexts [2] [13] [16] where all these models experience data sparsity issues.

We show the percentage of segments where speeds can be captured by speed models in 5-min slots in Figure 4.

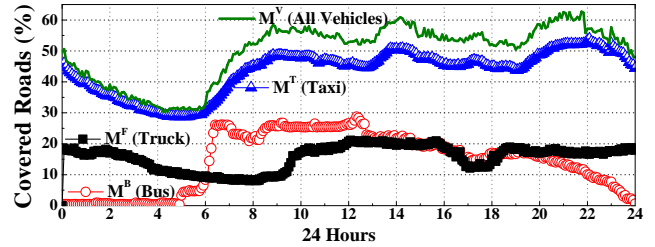


Fig 4. Covered Road Segments

We found that these models capture a low percentage of segments under 5-min slots, *e.g.*, even for M^V based on all vehicle data, we only have 49% of road segments on average with vehicles, which leads to data sparsity.

Similarly, we show the number of residents captured by M^C and M^S in Figure 5 where the result for M^S is shown by a factor of 10 in order to show the fluctuation.

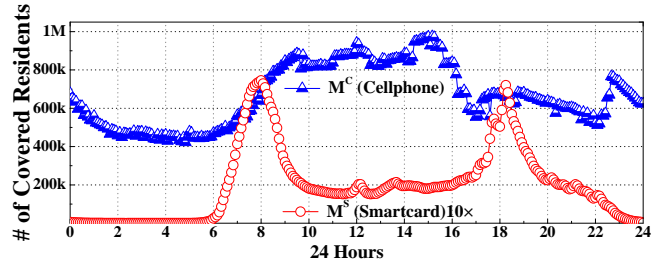


Fig 5. Covered Urban Residents

We found that these two density models also have data sparsity issues due to high total population in Shenzhen, *e.g.*, among 11 million permanent residents, M^C can only capture 1 million of them at most during a 5-min slot around 15:00, accounted for only 9% of all residents. M^C can only capture 80 thousand of them at most during a 5-min slot of the morning rush hour, accounted for only 0.7% of all residents.

3.2.3 Opportunity for Model Integration

In this work, we found that although all these models have data sparsity issues, M^C and M^S have more complete data than others, *e.g.*, for every 5-min slot in both M^C and M^S , we have density data at cell tower and transit station levels. Therefore, by resetting their spatial granularity to road segment levels (*i.e.*, the details are given in Section 5.2), density models M^C and M^S are capable of providing complimentary knowledge for speed models M^T , M^B and M^F , which have severe data sparsity issues on road segment levels, *e.g.*, if a speed model does not have GPS data about a road segment during a time slot, we infer missing GPS data based on GPS data from another road segment with similar urban density, shown by our model integration as follows.

4 Model Integration

We introduce our integration technique by combining models directly or indirectly relevant to phenomena of interest (hereafter direct and indirect models for conciseness). In this work, we simply identify a model as a direct model to an urban phenomenon, if it is based on the data with direct measurements of this phenomenon, *e.g.*, a model based on taxicab data is a direct model for the phenomenon of traffic speeds, because taxicab data have direct measurements of speeds. But a model based on cellphone data is only an indirect model for speeds because it does not have direct measurements on speeds. Note that direct and indirect models are different from classic supervised and unsupervised models in data mining, which are both direct models in our context since they are based on data with direct measurements for phenomena of interest.

4.1 Problem Formulation

Let $x_{t,s}$ be an urban phenomenon we want to characterize associated with a temporal context t and a spatial context s , and let y be a class label, where $x_{t,s}$ and y are selected from a phenomenon space \mathbf{X} and a label space \mathbf{Y} . Based on K different data sources in various urban infrastructures, we have a set of K models, *i.e.*, from M^1 to M^K , and each of them is independently formulated based on a corresponding data source. For example, in our later application, $x_{t,s}$ is a traffic speed on a road segment s during a time period t ; y is a label of 20km/h; M^1 is a model based on taxicab data and assigns a particular label y to $x_{t,s}$.

Formally, based on the Bayesian model averaging approach, we have the probability distribution about y as follows.

$$P(y|x_{t,s}) = \sum_{k=1}^K P(y|M^k, x_{t,s}) \times P(M^k|x_{t,s}), \quad (1)$$

where $P(y|M^k, x_{t,s})$ is the prediction made by M^k regarding to $x_{t,s}$; $P(M^k|x_{t,s})$ is considered as a model weight for a particular model M^k given a particular urban phenomenon $x_{t,s}$ under with a temporal context t and a spatial context s .

To integrate different models in small-scale systems, Eq.(1) can be directly used. In particular, $P(y|M^k, x_{t,s})$ can be accurately obtained by a direct model M^k directly-relevant to the phenomenon of interest $x_{t,s}$, based on the complete data. Further, the ground truth of conditional probability $P(y = y_j|x_{t,s})$ can also be measured and then used by a regression process to obtain the optimal weight $P(M^k|x_{t,s})$ for a model M^k given $x_{t,s}$. However, to integrate models in our UrbanCPS with Eq.(1), we face three challenges to directly obtain the two factors, *i.e.*, $P(y|M^k, x_{t,s})$ and $P(M^k|x_{t,s})$.

First, the models in our UrbanCPS are mostly heterogeneous and based on the data generated by service providers primarily for their own benefits, and thus these models may be only *indirectly* relevant to the phenomenon of interest. As a result, given an indirect model M^k to the phenomenon $x_{t,s}$, $P(y|M^k, x_{t,s})$ in Eq.(1) is unknown.

Second, due to large-scale phenomena of interest, the data in UrbanCPS are typically quite *sparse*. As a result, even for a direct model M^k for the phenomenon $x_{t,s}$, $P(y|M^k, x_{t,s})$ in Eq.(1) may still be unknown.

Third, due to technical issues and high costs for direct measurements on urban phenomena, the ground truth for certain phenomena is typically *unknown*. Without the ground truth, we cannot use a regression process to obtain the optimal weights for all models during integration. Thus, even with known $P(y|M^k, x_{t,s})$ based on a direct model with complete data, $P(M^k|x_{t,s})$ in Eq.(1) may still be unknown.

A combination of these three challenges provides us a unique design space for our model integration compared to the existing work. As follows, we first show how to solve this problem optimally, and then deal with these challenges.

4.2 Optimal Solution

Suppose the label space \mathbf{Y} is mapped into discrete labels $\{y_1, \dots, y_{|\mathbf{Y}|}\}$ where $|\mathbf{Y}|$ is the number of labels. Let $\mathbf{H}_{t,s}$ be a $|\mathbf{Y}| \times K$ matrix where $H_j^k = P(y = y_j|M^k, x_{t,s})$ is the kj entry, and thus it represents all predictions made for $x_{t,s}$ from all K models. Let $\mathbf{w}_{t,s}$ be a $K \times 1$ weight vector where $w_{t,s}^k = P(M^k|x_{t,s})$, and thus it represents weights of all K models. As a result, a $|\mathbf{Y}| \times 1$ vector $\mathbf{H}\mathbf{w}_{t,s}$ is the output of our model integration for $x_{t,s}$, which gives a probability distribution of $x_{t,s}$ on a label space \mathbf{Y} of $\{y_1, \dots, y_{|\mathbf{Y}|}\}$. With this output, we aim to minimize the distance from this output to the true conditional probability (given by the ground truth), which is represented by a $|\mathbf{Y}| \times 1$ vector $\mathbf{f}_{t,s}$ where $f_j = P(y = y_j|x_{t,s})$. Therefore, based on a straightforward squared error loss without regularization, the key objective of our model integration is to find an optimal weight vector $\mathbf{w}_{t,s}^*$ that minimizes the distance between the true $\mathbf{f}_{t,s}$ and our output $\mathbf{H}\mathbf{w}_{t,s}$ as follows.

$$\mathbf{w}_{t,s}^* = \arg \min_{\mathbf{w}_{t,s}} (\mathbf{f}_{t,s} - \mathbf{H}\mathbf{w}_{t,s})^T (\mathbf{f}_{t,s} - \mathbf{H}\mathbf{w}_{t,s}).$$

The optimal solution of this function can be directly obtained by a least-square linear regression.

However, as discussed before, this optimal solution has three impractical assumptions (*i.e.*, all directly-relevant models, complete data and known ground truth), which leads to two issues. First, an element in $\mathbf{H}_{t,s}$, *e.g.*, $H_j^k = P(y = y_j|M^k, x_{t,s})$, is not always available for an *indirect* model M^k or a direct model M^k based on *sparse* data. Second, the true conditional distribution $\mathbf{f}_{t,s}$ is mostly unknown due to the *unknown ground truth*. As in following three subsections, we relax these three assumptions one by one and discuss the issues of (i) how to obtain $P(y|M^k, x_{t,s})$ for an indirect model, (ii) how to obtain $P(y|M^k, x_{t,s})$ for a direct model based on sparse data, and (iii) how to infer the weights without the ground truth, respectively.

4.3 Indirect Models

In our UrbanCPS, various models are built based on the collected data, and some of these may not be directly relevant to the urban phenomenon we try to characterize. Suppose we have a set of urban phenomena associated with different real-world temporal and spatial contexts $\mathbf{X} = \{x_{t_1 \cdot s_1}, x_{t_1 \cdot s_2}, x_{t_2 \cdot s_1}, x_{t_2 \cdot s_2}\}$, and aim to characterize them into a label space of $\mathbf{Y} = \{y_1, y_2, y_3\}$. Suppose among all K models, the models from M^1 to M^d are direct models, and the models from M^{d+1} to M^K are indirect models. For a direct model $M^p \in (M^1, \dots, M^d)$, $P(y|M^p, x_{t,s})$ is directly ob-

tained; but for an indirect model $M^q \in (M^{d+1}, \dots, M^K)$, $P(y|M^q, x_{t,s})$ is typically unknown. The main objective of the following is to infer $P(y|M^q, x_{t,s})$ for an indirect model M^q . The key idea of our method is to use the *internal similarity* between an indirect model M^q and all direct models to infer $P(y|M^q, x_{t,s})$ for M^q for a particular temporal spatial combination. However, the internal similarity between models is difficult to be directly quantified, so we introduce a process of categorizing all elements in the phenomenon space \mathbf{X} by individual models as follows.

4.3.1 Categorizing

Based on a direct model M^p , we directly categorize all elements in $\mathbf{X} = \{x_{t_1 \cdot s_1}, x_{t_1 \cdot s_2}, x_{t_2 \cdot s_1}, x_{t_2 \cdot s_2}\}$ into $|M^p|$ categories, and each of category is associated with a unique label in \mathbf{Y} . Thus, for a direct model M^p , $|M^p| = |\mathbf{Y}|$. Similarly, based on an indirect model M^q , we also categorize all elements in \mathbf{X} into $|M^q|$ categories by a given clustering algorithm (the metric for clustering could be the direct measurement of data used to build M^q). Normally, an indirect model M^q cannot directly characterize the elements in \mathbf{X} because M^q has a different phenomenon space \mathbf{Z} . But we use a temporal-spatial context $t \cdot s$ to perform one to one mapping from elements in \mathbf{Z} to elements in \mathbf{X} in order to let M^p categorize \mathbf{X} . For example, if an indirect model M^q clusters elements in its own phenomenon space $\mathbf{Z} = \{z_{t_1 \cdot s_1}, z_{t_1 \cdot s_2}, z_{t_2 \cdot s_1}, z_{t_2 \cdot s_2}\}$ into two categories $\{z_{t_1 \cdot s_1}, z_{t_1 \cdot s_2}\}$ and $\{z_{t_2 \cdot s_1}, z_{t_2 \cdot s_2}\}$, then it also categorizes \mathbf{X} into two categories $\{x_{t_1 \cdot s_1}, x_{t_1 \cdot s_2}\}$ and $\{x_{t_2 \cdot s_1}, x_{t_2 \cdot s_2}\}$ under an observation of similarity between elements in \mathbf{X} and \mathbf{Z} with the same spatial and temporal conditions. Note that for an indirect model M^q , $|M^q|$ is based on a given clustering algorithm, and thus is not necessarily equal to $|\mathbf{Y}|$.

Table 2. Categorizing Example

	Label ID			Similarity Vectors							
	M^1	M^2	M^3	M^1		M^2			M^3		
	D	D	I	c_1^1	c_2^1	c_3^1	c_1^2	c_2^2	c_3^2	c_a^3	c_b^3
$x_{t_1 \cdot s_1}$	y_1	y_2	a	1	0	0	0	1	0	1	0
$x_{t_1 \cdot s_2}$	y_1	y_1	a	1	0	0	1	0	0	1	0
$x_{t_2 \cdot s_1}$	y_2	y_1	b	0	1	0	1	0	0	0	1
$x_{t_2 \cdot s_2}$	y_3	y_3	b	0	0	1	0	0	1	0	1

For example, as in Table 2, we have $K = 3$ models among which M^1 and M^2 are Direct models, and M^3 is an Indirect model. Thus, M^1 categorizes all elements in $\mathbf{X} = \{x_{t_1 \cdot s_1}, x_{t_1 \cdot s_2}, x_{t_2 \cdot s_1}, x_{t_2 \cdot s_2}\}$ into $|\mathbf{Y}| = \{y_1, y_2, y_3\} = 3$ categories, *i.e.*, c_1^1, c_2^1, c_3^1 , where the elements of \mathbf{X} in c_i^1 are with the label y_i . As in Table 2, suppose the model M^1 (i) assigns a label of y_1 to $x_{t_1 \cdot s_1}$ and $x_{t_1 \cdot s_2}$, leading to its first category $c_1^1 = \{x_{t_1 \cdot s_1}, x_{t_1 \cdot s_2}\}$, (ii) assigns a label of y_2 to $x_{t_2 \cdot s_1}$, leading to its second category $c_2^1 = \{x_{t_2 \cdot s_1}\}$, and (iii) assigns a label of y_3 to $x_{t_2 \cdot s_2}$, leading to its third category $c_3^1 = \{x_{t_2 \cdot s_2}\}$. Similarly, we have three categories for the direct model M^2 as well, and each of categories is also associated to a label in \mathbf{Y} . But for the indirect model M^3 , we only have two categories c_a^3 and c_b^3 , which are not directly associated to any label in \mathbf{Y} . Continuing with the previous real-world application where we try to characterize $x_{t_1 \cdot s_1}$, *i.e.*, the traffic speed for a road segment s_1 during a time pe-

riod t_1 . M^1 is the speed model M^T based on taxicab data, M^2 is the speed model M^B based on bus data, and M^3 is the urban density model M^C based on cellphone data. Based on M^1 , we assign a label $y_1 = 10$ km/h to $x_{t_1 \cdot s_1}$; but based on M^2 , we assign a label $y_2 = 20$ km/h to $x_{t_1 \cdot s_1}$. Further, the indirect model M^3 can only tell us that $x_{t_1 \cdot s_1}$ may be similar to $x_{t_1 \cdot s_2}$, because according to M^3 , the urban densities for road segments s_1 and s_2 are similar during a time period t_1 .

Based on categorizing, given $x_{t_1 \cdot s_1}$, we have a unified formula for either a direct or indirect model M^k as follows.

$$P(y|M^k, x_{t,s}) = \sum_{l=1}^{|M^k|} P(y|c_l^k, M^k, x_{t,s}) \cdot P(c_l^k|M^k, x_{t,s}),$$

where c_l^k is the l -th category of M^k ; $P(c_l^k|M^k, x_{t,s}) = 1$ if $x_{t,s} \in c_l^k$; $P(c_l^k|M^k, x_{t,s}) = 0$ if otherwise. Thus, given $x_{t,s} \in c_l^k$,

$$P(y|M^k, x_{t,s}) = P(y|c_l^k, M^k, x_{t,s}) = P(y|c_l^k, x_{t,s}). \quad (2)$$

Therefore, we transfer the problem from the model level $P(y|M^k, x_{t,s})$ to the category level $P(y|c_l^k, x_{t,s})$, because the comparison between categories is earlier to quantify.

Given $x_{t,s} \in c_l^p$ where c_l^p belongs to a direct model M^p ,

$$P(y = y_i|c_l^p, x_{t,s}) = \begin{cases} 1 & \text{if } l = i \\ 0 & \text{if } l \neq i \end{cases}. \quad (3)$$

Note that for simplicity we assume that there are no errors during categorizing, *i.e.*, given $x_{t,s} \in c_l^p$, it is always assigned to y_l . But if $P(y = y_i|c_l^p, x_{t,s})$ follows an empirical distribution instead of as in Eq.(3), our method still works with a straightforward probabilistic method.

Given $x_{t,s} \in c_l^q$ where c_l^q belongs to an indirect model M^q , however, $P(y = y_i|c_l^q, x_{t,s})$ is unknown. Thus, the key question we have now is how to infer $P(y|c_l^q, x_{t,s})$ for a category c_l^q belonging an indirect model M^q . As follows, we solve this issue by exploring similarity between categories from direct and indirect models.

4.3.2 Similarity Measurement

Basically, the rationale behind the similarity measurement is that given a category c_l^p from a direct model M^p and a category c_l^q from an indirect model M^q , the closer c_l^q is to c_l^p , the more likely that the members in c_l^q have the same label with the members in c_l^p . Essentially, we transfer the expertise from direct models to indirect models by comparing their similarities on category levels.

Formally, for $P(y|c_l^q, x_{t,s})$ where the category c_l^q belonging an indirect model M^q , we have

$$P(y = y_i|c_l^q, x_{t,s}) = \frac{\sum_{j=1}^d \mathbf{S}(c_l^q, c_j^i)}{\sum_{i=1}^{|\mathbf{Y}|} \sum_{j=1}^d \mathbf{S}(c_l^q, c_j^i)}, \quad (4)$$

where $\mathbf{S}(c_l^q, c_j^i)$ is the similarity between two categories c_l^q and c_j^i . Therefore, the numerator is the sum of similarity between a category c_l^q and all categories with a *particular* label y_i from all direct models (*i.e.*, from M^1 to M^d); the denominator is the sum of similarity between a category c_l^q

and all categories with *all labels* (i.e., from y_1 to $y_{|\mathbf{Y}|}$) from all direct models (i.e., from M^1 to M^d).

To quantify similarity between two categories, we use a similarity vector \mathbf{c}_i^k to represent the membership of elements in \mathbf{X} for a category c_i^k . For example, as in Table 2, we have $\mathbf{c}_1^1 = \{1, 1, 0, 0\}$ indicating the first and second elements in \mathbf{X} , i.e., $x_{t_1 \cdot s_1}$ and $x_{t_1 \cdot s_2}$, belong to c_1^1 . With similarity vectors, we calculate $\mathbf{S}(c_i^q, c_i^j)$ by Jaccard index.

$$\mathbf{S}(c_i^q, c_i^j) = \frac{|\mathbf{c}_i^q \cap \mathbf{c}_i^j|}{|\mathbf{c}_i^q \cup \mathbf{c}_i^j|}.$$

For example, in Table 2, $\mathbf{S}(c_1^1, c_2^1) = \frac{0}{3}$, and $\mathbf{S}(c_1^1, c_1^2) = \frac{1}{3}$. By changing y_i from y_1 to $y_{|\mathbf{Y}|}$ in Eq.(4), we have the distribution of $P(y_i | c_i^q, x_{t \cdot s})$.

4.3.3 Summary

In short, based on $P(y_i | c_i^p, x_{t \cdot s})$ in Eq.(3) for a category c_i^p from a direct model M^p where $p \in [1, d]$ and $P(y_i | c_i^q, x_{t \cdot s})$ in Eq.(4) for a category c_i^q from an indirect model M^q where $q \in [d+1, K]$, we have $P(y_i | c_i^k, x_{t \cdot s})$ for any category from both either a direct model M^p or an indirect model M^q . As a result, we have $P(y_i | M^k, x_{t \cdot s})$ for all models where $k \in [1, K]$ in Eq.(2), which addressed the challenge of integrating heterogeneous direct models and indirect models.

4.4 Models based on Sparse Data

In this subsection, we address the issue related to models based on sparse data, according to the proposed similarity measurement. For example, in Table 2, we consider an extreme example where all direct models, both M^1 and M^2 , cannot infer the phenomenon $x_{t_1 \cdot s_2}$ due to the sparse data issue in the temporal context t_1 and the spatial context s_2 , i.e., $M^1(x_{t_1 \cdot s_2}) = \emptyset$, and $M^2(x_{t_1 \cdot s_2}) = \emptyset$. But based on the indirect model M^3 , we found that $x_{t_1 \cdot s_2}$ may be similar to $x_{t_1 \cdot s_1}$, because in M^3 , $x_{t_1 \cdot s_2}$ and $x_{t_1 \cdot s_1}$ are in the same category. Note this categorizing result is based on knowledge of M^3 itself, and is not affected by the sparse data issue in M^1 and M^2 . Because $x_{t_1 \cdot s_2}$ is similar to $x_{t_1 \cdot s_1}$, (i) $x_{t_1 \cdot s_2}$ may belong to c_1^1 in M^1 with label y_1 because $x_{t_1 \cdot s_1}$ belongs to c_1^1 in M^1 ; (ii) $x_{t_1 \cdot s_2}$ may also belong to c_2^2 in M^2 with label y_2 because $x_{t_1 \cdot s_1}$ belongs to c_2^2 in M^2 . Intuitively, we use knowledge of indirect models to address the sparse data issue of direct models.

Formally, for a model M^k where $M^k(x_{t \cdot s}) = \emptyset$, we have

$$P(y_i | M^k, x_{t \cdot s}) = \frac{\sum_{j=1, j \neq k}^K \mathbf{S}(M^k, M^j) \cdot \mathbf{D}(c_i^j, x_{t \cdot s})}{\sum_{i=1}^{|\mathbf{Y}|} \sum_{j=1, j \neq i}^K \mathbf{S}(M^k, M^j) \cdot \mathbf{D}(c_i^j, x_{t \cdot s})}, \quad (5)$$

where $\mathbf{D}(c_i^j, x_{t \cdot s}) = 1$ if $x_{t \cdot s} \in c_i^j$; $\mathbf{D}(c_i^j, x_{t \cdot s}) = 0$ if otherwise. As a result, the numerator is the weighted number of the categories where $x_{t \cdot s}$ is labeled with y_i ; and the denominator is the weighted number of the categories where $x_{t \cdot s}$ is labeled from y_1 to $y_{|\mathbf{Y}|}$. Essentially, we assign a label y_i to $x_{t \cdot s}$ in a model M^k where $M^k(x_{t \cdot s}) = \emptyset$, by checking the labels assigned to $x_{t \cdot s}$ in other models and the similarity between these models and M^k . Instead of a simple majority voting, we use the similarity $\mathbf{S}(M^k, M^j)$ between two models M^k

and M^j as a weight to improve the accuracy.

$$\mathbf{S}(M^k, M^j) = \frac{\sum_{u=1}^{|M^k|} \sum_{v=1}^{|M^j|} \mathbf{S}(c_u^k, c_v^j)}{|M^k| \cdot |M^j|},$$

where we use the similarity at category levels to indicate the similarity at model levels. Therefore, based on Eq.(5), we solved the issue about models based on sparse data.

Note that this method addresses data sparsity for direct models by assuming the data are complete for at least one indirect model. If we have missing data for all models, we have to use traditional methods, e.g., weighted averaging, to infer missing data based on historical data.

4.5 Weighting Models without Ground Truth

In this subsection, we address the issues of assigning a weight to a model for the integration without ground truth. Normally, the closer a model M^k is to the majority of all models, the higher weight it should be assigned with. Therefore, based on the similarity we proposed in the previous subsection, we assign the weight of a model M^k for a particular combination of a temporal context t and a spatial context s as follows.

$$P(M^k | x_{t \cdot s}) = w_{t \cdot s}^k = \frac{\sum_{j=1, j \neq k}^K \mathbf{S}(M^k, M^j)}{\sum_{i=1}^K \sum_{j=1, j \neq i}^K \mathbf{S}(M^i, M^j)},$$

where the numerator is the sum of similarity between M^k and all models; the denominator is the sum of similarity among all models.

Note that existing work usually weights each model globally, but our method assigns weights to each model according to a unique phenomenon x under a unique temporal-spatial combination $t \cdot s$, which is used to identify variations in the model performance for different real-world contexts. There usually do not exist one weighting scheme that is globally optimal for any phenomenon under all temporal-spatial contexts. Usually, the urban phenomenon under different temporal and spatial contexts may favor different models. Thus, the weighting scheme based on temporal-spatial contexts is better than the global weighting scheme in terms of prediction accuracy.

4.6 Summary

Based on the problem formulation in the first subsection, we obtain the optimal solution for model weights, which minimizes the distance between the true conditional distribution and the output of our integration. Then in the following three subsections, we relax the three key assumptions in the optimal solution one by one towards a practical model integration. Essentially, the key idea we have been using is to compare internal similarity of effects of different models on a set of given urban phenomena. Then, we transfer predictive powers of indirect models with complete data to direct models with sparse data. The rationale is that the more similar two models, the more likely they would make the same prediction about an urban phenomenon. Finally, the similarity is used as an indication of a model's weight, by assuming the majority of the models are correct, and thus the closer a model is to other models, the higher weight it carries.

5 Application: Speedometer

In this section, we present an application called Speedometer to test the performance of our model integration based on the data we collected in Shenzhen.

5.1 Application Background

The real-time traffic speed in urban regions is an important phenomenon for both residents and transportation authority. An accurate inference about traffic speeds on road segment levels under fine-grained time slots improves many urban applications, *e.g.*, more efficient automobile navigation. A direct yet trivial solution is to install speed detectors, such as loop detectors, in every road segment. However, this solution would involve tremendous costs, so these detectors are only installed in major segments for most cities. To achieve a speed inference for all segments, vehicle GPS data from commercial vehicles, such as taxicab, are utilized to produce several models to infer traffic speeds [2]. Also, several systems also infer traffic speeds based on participatory sensing [18]. But these models typically are based on single-source homogenous data and are ineffective when data are sparse in fine-grained contexts.

To address this issue, we propose Speedometer, which infers real-time traffic speeds on segment levels based on an integration of five models, *i.e.*, M^T , M^B , M^F , M^C and M^S , as proposed in Section 3.2. M^T , M^B and M^F are speed models based on Taxicab, Bus and Freight truck data; whereas M^C and M^S are density models based on cellphone and smart-card data. Thus, M^T , M^B and M^F individually map a traffic speed $x_{t,s}$ on a segment s during a period t into a label space \mathbf{Y} to indicate a traffic speed. M^C and M^S individually infer an urban density into another label space to indicate a density under the same contexts. Based on domain knowledge, M^T , M^B and M^F are direct models to speeds, and M^C and M^S are indirect models. Thus, Speedometer effectively integrates them to produce accurate speed inferences based on our integration. For different applications, Speedometer infers traffic speeds on both segment and region levels by aggregating segments with the minimum time slot of 5 mins. Figure 6 gives a visualization on average speeds inferred by Speedometer from 6PM to 7PM in 496 Shenzhen regions where a warmer color indicates a slower speed.

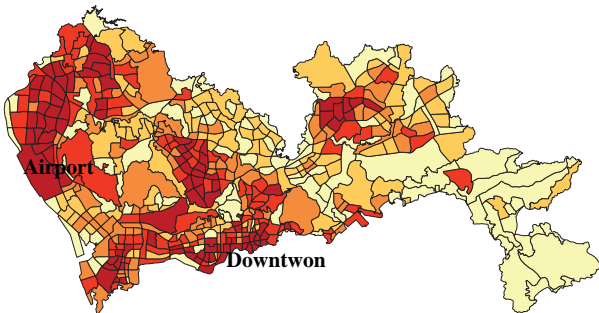


Fig 6. Traffic Speeds across Urban Regions

5.2 Application Evaluation

We compare Speedometer with one real-world system and one state-of-the-art model. The **SZ-Taxi System**: Shenzhen government has a pilot program called TravelIndex to

infer congestion levels on road segments for the convenience of its residents, which shows inferred traffic speeds in real time based on GPS data from all taxicabs in Shenzhen [14]. SZ-Taxi serves as a single-source model suitable for the situation where the multi-source data are not available. The **TSE Model**: TSE uses real-time and historical vehicle GPS data and contexts (*e.g.*, physical features of roads) to infer traffic speed with a collaborative filtering [13]. For a fair comparison, we aggregate GPS data from taxicabs, buses, and trucks to feed TSE. TSE serves as a naive multi-source approach for the situation where multiple heterogeneous data sources are available, but the integration is at data levels. Differently, Speedometer uses five models, *i.e.*, M^T , M^B , M^F , M^C and M^S for an integration at model levels. We reset M^C and M^S to the same spatial granularity with M^T , M^B and M^F . In particular, M^C and M^S give the urban density at cell towers and transit station levels, which can be redistributed to road segment levels based on coverage areas of particular cell towers or transit stations. We assign numbers of residents inferred by M^C and M^S within a coverage area to all segments in this area. The number of residents assigned to a segment is proportional to the segment length. Further, we use DBSCAN to obtain categories for the similarity measurement.

We utilize 91 days of datasets from all infrastructures in Figure 3. We use a cross-validation approach to divide the data into two subsets: the *testing set* as streaming data, including the data for one particular day; and the *historical set* as historical data, including the data for the remaining of 90 days. For a particular day, if we use 10-min slots, at the end of the first slot, *i.e.*, 12:10AM, we use models to infer the speed for the slot from 12:00AM to 12:10AM, based on both the “real-time” data from 12:00AM to 12:10AM in the testing set and all historical data in the historical set. We move the data in the testing set forward for 90 days, leading to 91 experiments. The average results were reported.

We test the models with **Mean Average Percent Error (MAPE)** as $MAPE = \frac{100}{n} \sum_{i=1}^n \frac{|\bar{T}_i - T_i|}{T_i}$, where n is the total number of temporal-spatial combinations we tested. We test all models on 18 road segments under 10 min slots, which leads to $24 \times \frac{60}{10} \times 18 = 2592$ combinations for a one-day evaluation. T_i is the traffic speed inferred by a model under a temporal-spatial combination i ; \bar{T}_i is the ground truth of the traffic speed under a temporal-spatial combination i . An accurate model yields a small MAPE, and *vice versa*. We test models on these specific road segments because we have access to the ground truth of traffic speeds on these road segments. This ground truth is obtained by loop detectors in Shenzhen road networks, which are inductive loops installed in selected major road segments, and can detect metal and thus accurately detect vehicle speeds.

We first compare all models to show results on four particular road segments and the average result on all road segments. Then, we study impacts of inference slot lengths. Further, we investigate the impact of historical data sizes on the running time and the accuracy of Speedometer to show its feasibility and robustness for the real-time inferences. Finally, we present an evaluation summary.

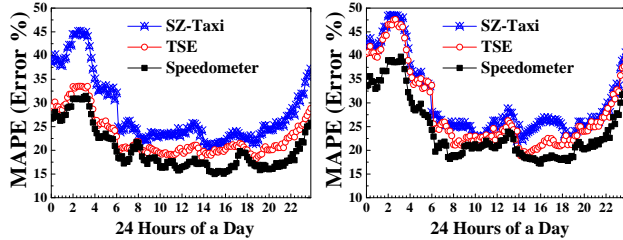


Fig 7. Nantou MAPE

Fig 8. Tongle MAPE

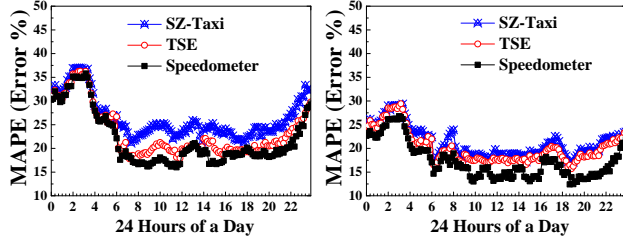


Fig 9. Fulong MAPE

Fig 10. Shennan MAPE

5.2.1 Accuracy on Road Segments

Figures 7, 8, 9 and 10 plot the MAPE under 10-min slots for four major road segments (*i.e.*, Nantou, Tongle, Fulong and Shennan) in Shenzhen urban area. The first three road segments are in uptown, and the last road segment is in downtown. In general, Speedometer outperforms TSE, which outperforms SZ-Taxi. This is because SZ-Taxi only considers taxicabs to infer the speeds, which leads to high MAPE, *e.g.*, the early morning in Nantou as in Figure 7. Though TSE uses all data from commercial vehicles, it does not consider other indirect density models. Thus, when the GPS data are not available during certain temporal-spatial combinations, its MAPE is high, *e.g.*, the early morning in Tongle as in Figure 8. For road segments where vehicles are abundant, these three models have the similar MAPE, *e.g.*, the early morning in Fulong as in Figure 9. In general, the performance gain between Speedometer and others is lower during the daytime and the road segments in downtown, *e.g.*, Shennan in Figure 10. This is because taxicabs and other commercial vehicles are abundant and thus quite representative in the downtown during the daytime, so all models have better performance.

Figure 11 gives the average MAPE for all road segments under 10-min slots during 24 hours. The MAPE of all three models are typically higher than the MAPE we found in Figures 7, 8, 9 and 10. This is because the traffic speed may change dramatically between road segments, and some remote road segments with few vehicles uploading GPS data lead to higher MAPE. But the relative performance between the three models is similar. Speedometer outperforms TSE by 13% on average, and the performance gains are more obvious in the regular daytime, which may result from the consideration of density models. Speedometer outperforms SZ-Taxi by 21%, resulting from its integration of the multiple models.

5.2.2 Impact of Slot Lengths

Figure 12 plots the MAPE of all models with different slot lengths with a default value of 10 mins. The MAPE of all models reduces with an increase in the lengths of the

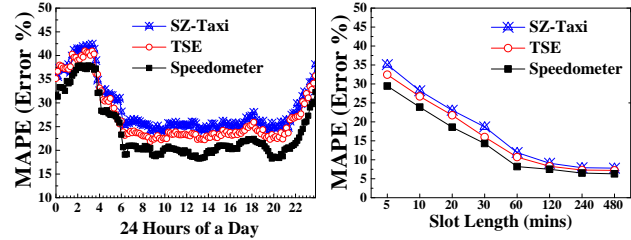


Fig 11. Hourly MAPE

Fig 12. Effects of Lengths

time slots, because in a longer slot we accumulate more data about vehicles, and the traffic speed becomes more stable. Speedometer outperforms TSE and SZ-Taxi significantly if the slot is shorter than 30 mins, which results from the consideration of density models. But when the slot becomes longer than one hour, all models have similar performance, because in such a long slot, all models have enough data for an accurate inference about relatively stable speeds.

5.2.3 Impact of Historical Data

In this subsection, we study the impact of historical data on model accuracies and running times by comparing Speedometer to TSE with a default value of 13 weeks. Normally, the more the historical data, the more accurate the models, the lower the MAPE error and the longer the running time. Figures 13 and 14 plot running times and MAPE on different lengths of historical data in terms of weeks. Speedometer has a 17% longer running time, which in turn leads to an 11% lower MAPE. This is because Speedometer has to perform its integration involving heterogeneous models, which takes time to calculate the model similarity.

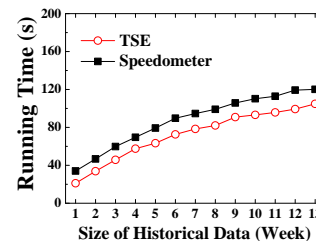


Fig 13. Data vs. Time

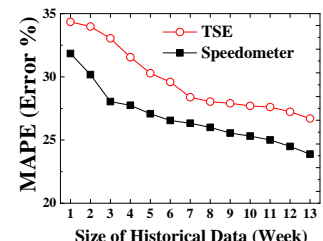


Fig 14. Data vs. MAPE

5.2.4 Evaluation Summary

We have the following observations: (i) The inference accuracy is highly dependent on both locations and times as shown by Figures from 7 and 11. On average, all models have better performance in more dense area during the daytime, due to the abundance of the data to feed models. (ii) The length of slots has a significant impact on the performance of all models as in Figure 12. It is intuitive that a longer slot has lower error rates, yet it also reduces the practicality for real-time applications. (iii) As in Figures 13 and 14, the model integration takes a longer running time especially when the historical dataset is big, but it increases the accuracy. A good tradeoff between accuracy and running times has to be designed based on domain knowledge and user preferences. (iv) Looking across different factors, we found that slot lengths have the largest impact, and then locations and times, and finally historical data sizes.

6 Related Work

Two types of work are related to our UrbanCPS: (i) models based on single-source urban infrastructure data and (ii) theoretical ensemble of multiple models.

6.1 Models based on Single-Source Data

Numerous novel models and systems have been proposed based on various urban infrastructure data to improve urban efficiency. We focus on the work closely related to models based on vehicular GPS, cellphone and transit data. Based on GPS data, many models and systems are proposed to benefit various urban residents: estimating traffic volumes or speeds for regular drivers [2]; assisting regular drivers to improve their driving performance [16]; detecting anomalous taxicab trips to discover driver fraud for taxicab operators [17]. Further, many methods have been proposed for the study of human density and mobility based on cellphone CDR data, *e.g.*, identifying cellphone users' important locations [9]; modeling how cellphone users move [10]; predicting where cellphone users will travel next [5]. Finally, transit GPS data are another important source for research in human density and mobility, *e.g.*, identifying passenger locations based on data from taxicabs [6], buses [3], and subways [11].

To our knowledge, we are the first to store such a large multi-source dataset, and then build models based on single-source sparse data, and finally systemically integrate these models from a complimentary standpoint. Obviously, the key difference of our work is that our model integration is built upon these models based on single-source data, and then effectively integrates them for better performance.

6.2 Theoretical Ensemble of Multiple Models

Our integration approach is inspired by several studies in the data mining community proposed to theoretically combine different models to improve their performance [12] [15] [8] [7]. However, these studies are mostly under perfect conditions, *e.g.*, the models are based on the complete data and directly-relevant data [15]. Differently, our work is focused on models based on the imperfect data, *e.g.*, sparse and indirectly-relevant data. Further, semi-supervised learning also address issues related to imperfect data, *e.g.*, the unlabeled data, but the models in these work are mostly based on same domain knowledge, *e.g.*, similar weather data from different websites [12] or similar email data from different users [8]. In contrast, our approach is to combine much more diverse models, *i.e.*, speed models and density models, based on various urban infrastructure data. In addition, most studies on model integration in the data mining community are based on small-scale data, so their computation is often complex for better performance [15], *e.g.*, computing inverse matrices and conducting non-linear programming, which is undesirable for real-time applications based on large-scale urban infrastructure data. Differently, the similarity measurement in our model integration is optimized for computation efficiency, which makes our work suitable for real-time applications.

7 Conclusion

In this work, we design and implement UrbanCPS to effectively integrate heterogeneous models based on

multi-source infrastructure data. Our endeavors offer a few valuable insights which we hope will allow fellow researchers to utilize our system for not only model integration but also real-world applications: (i) heterogeneous models based on different urban infrastructure data provide different yet complimentary view for the same urban phenomenon, and thus an effective integration among them would boost the model performance; (ii) for many urban phenomena, indirectly-relevant models are often powerful to address the issue of directly-relevant models, *e.g.*, sparse data, but we need an effective method to integrate them with direct-relevant models; (iii) though difficult to be obtained, the ground truth data about urban phenomena are vital for both model designs and evaluations. (iv) while it is challenging to integrate heterogeneous models, it is more challenging to negotiate with service providers for large-scale infrastructure data to feed models.

8 Acknowledgements

The authors thank Ling Yin in SIAT for the data support. This work was supported in part by the US NSF Grants CNS-1239226 and China 973 Program No. 2015CB352400.

9 References

- [1] Sample data. <http://www.cs.umn.edu/~zhang/ICCPs>.
- [2] ASLAM, J., LIM, S., PAN, X., AND RUS, D. City-scale traffic estimation from a roving sensor network. In *Proceedings of 10th ACM Conference on Embedded Network Sensor Systems*, SenSys '12.
- [3] BHATTACHARYA, S., PHITHAKITNUKON, S., NURMI, P., KLAMI, A., VELOSO, M., AND BENTO, C. Gaussian process-based predictive modeling for bus ridership. UbiComp '13.
- [4] COX, W. How urban density intensifies traffic congestion and air pollution. <http://americandreamcoalition.org/landuse/denseair.pdf>.
- [5] DUFKOVÁ, K., LE BOUDEC, J.-Y., KENCL, L., AND BJELICA, M. Predicting user-cell association in cellular networks. MELT'09.
- [6] GANTI, R., SRIVATSA, M., RANGANATHAN, A., AND HAN, J. Inferring human mobility patterns from taxicab traces. UbiComp '13.
- [7] GAO, J., FAN, W., JIANG, J., AND HAN, J. Knowledge transfer via multiple model local structure mapping.
- [8] GAO, J., FAN, W., SUN, Y., AND HAN, J. Heterogeneous source consensus learning via decision propagation and negotiation.
- [9] ISAACMAN, S., BECKER, R., CÁCERES, R., KOBOUROV, S., AND ROWLAND, J. A tale of two cities. In *HotMobile '10*.
- [10] ISAACMAN, S., BECKER, R., CÁCERES, R., MARTONOSI, M., ROWLAND, J., VARSHAVSKY, A., AND WILLINGER, W. Human mobility modeling at metropolitan scales. MobiSys '12.
- [11] LATHIA, N., AND CAPRA, L. How smart is your smartcard?: Measuring travel behaviours, perceptions, and incentives. UbiComp '11.
- [12] LI, Q., LI, Y., GAO, J., ZHAO, B., FAN, W., AND HAN, J. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In *ACM SIGMOD'14*.
- [13] SHANG, J., ZHENG, Y., TONG, W., CHANG, E., AND YU, Y. Inferring gas consumption and pollution emission of vehicles throughout a city. In *KDD 2014 (August 2014)*, ACM.
- [14] TRANSPORT COMMISSION OF SHENZHEN MUNICIPALITY. Shenzhen Travel Index. In <http://szmap.sutpc.com/>.
- [15] XIE, S., GAO, J., FAN, W., TURAGA, D., AND YU, P. S. Class-distribution regularized consensus maximization for alleviating overfitting in model combination. In *ACM KDD'14*.
- [16] YUAN, J., ZHENG, Y., XIE, X., AND SUN, G. Driving with knowledge from the physical world. KDD '11.
- [17] ZHANG, D., LI, N., ZHOU, Z.-H., CHEN, C., SUN, L., AND LI, S. ibat: detecting anomalous taxi trajectories from gps traces. In *13th conference on ubiquitous computing*, UbiComp '11.
- [18] ZHENG, Y., CAPRA, L., WOLFSON, O., AND YANG, H. Urban computing: concepts, methodologies, and applications. *ACM TIST*, 5(3), 2014.