NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

# MACHINE LEARNING MODELS IN URBAN TRANSPORT APPLICATIONS: UNDERSTANDING AND ACTUATION

**MINGQIAN LI**

**INTERDISCIPLINARY GRADUATE PROGRAMME**
**ALIBABA-NTU JOINT RESEARCH INSTITUTE**

**2023**

# MACHINE LEARNING MODELS IN URBAN TRANSPORT APPLICATIONS: UNDERSTANDING AND ACTUATION

**MINGQIAN LI**

Interdisciplinary Graduate Programme
Alibaba-NTU Joint Research Institute

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy

**2023**

# Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

13 Oct 2023

. . . . . .

Date

. . . . . .

Mingqian Li

# Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

13 Oct 2023

.....

Date

.....

Li Mo

# Authorship Attribution Statement

This thesis contains material from TWO papers published in the following peer-reviewed journal(s) / from papers accepted at conferences in which I am listed as an author.

Chapter 3 is published as Li, M., Tong, P., Li, M., Jin, Z., Huang, J., & Hua, X.-S. (2021). Traffic Flow Prediction with Vehicle Trajectories. Proceedings of the AAAI Conference on Artificial Intelligence, 35(1), 294-302. https://doi.org/10.1609/aaai.v35i1.16104

The contributions of the co-authors are as follows:

- Dr. Mo Li provided the project direction and technical direction. He discussed with me at each stage of this paper.
- I designed the study, proposed the solutions and experiment design, and performed all the laboratory work at IGS, NTU, Singapore, with the collaboration of CityBrain, Alibaba Cloud. The final design of methods and experiments were determined after several rounds of meeting and discussion with Dr. Mo Li, Dr. Panrong Tong, Dr. Zhongming Jin, Jianqiang Huang and Dr. Xian-Sheng Hua.
- I prepared the manuscript drafts. Dr. Mo Li and Dr. Panrong Tong improved the writing quality of the paper.

Chapter 4 is a paper in submission, Towards sample re-weighting in uplift modeling, by Mingqian Li, Mo Li, Panrong Tong, Zhongming Jin, and Jieping Ye.

The contributions of the co-authors are as follows:

- Dr. Mo Li provided the project direction and technical direction. He discussed with me at each stage of this paper.
- I designed the study, proposed the solutions and experiment design, and performed all the laboratory work at IGS, NTU, Singapore, with the collaboration of CityBrain, Alibaba

Cloud. Dr. Panrong Tong helped in data acquisition for experiments. The discussion with Dr. Mo Li helped me refine the solutions and improve the experiments. The final design of methods and experiments were determined after several rounds of meeting and discussion with Dr. Mo Li, Dr. Panrong Tong, Dr. Zhongming Jin, and Dr. Jieping Ye.

- I prepared the manuscript drafts. Dr. Mo Li improved the writing quality of the manuscript.

13 Oct 2023

. . . . . .
Date

. . . . . .
Mingqian Li

# Acknowledgement

First and foremost, I would like to express my sincere gratitude to Professor Li Mo from the School of Computer Science and Engineering, for being my PhD supervisor in NTU and for his continuous support and guidance for my research projects. I would always appreciate his patience in guiding me through my PhD studies from brainstorming research topics to drafting papers, from rebuttal to presentation, from lab experiments to industrial collaborations, and from research to life, without which I would not make this thesis possible. I would also remember the times we partied at Prof. Li's house, with delicious food and fun activities. I am also grateful to my co-supervisor, Professor Cyril Leung, and my mentor, Dr. Wang Di, for being the members of my Thesis Advisory Committee and providing constructive feedback and insightful comments on my work via emails, talks and catch-ups.

I would also like to thank my group mates in WANDS for their great support and encouragement. I would always remember the extensive discussion I had with Dr. Tong Panrong and Dr. Mo Xiaoyun on our collaborative work, the generous help from Dr. Cao Chu, Liu Kaiqi and Ji Sijie on every detail in research and in life. I would like to extend my thanks to my peers in Alibaba-NTU Talent Program for always being encouraging, positive and enthusiatic; and to the work mates in Alibaba for the valuable support and research facilities.

Last but not least, I would like to thank my husband, Han Qiao, for being the greatest companion in my life, giving support for every decision I made and giving help for every struggle I had during my research and my life. I would like to thank my baby, Han Ziqian, for the happy time I had with him. I would also like to thank my parents, family, peers and friends for their love, support and encouragement.

# Contents

# List of Acronyms

| | |
|---|---|
| **ACIC** | Atlantic Causal Inference Conference |
| **AI** | Artificial intelligence |
| **ARIMA** | Autoregressive Integrated Moving Average |
| **ASTGCN** | Attention Based Spatial-Temporal Graph Convolutional Networks |
| **ATE** | Average treatment effect |
| **ATT** | Average treatment effect on treated |
| **AUUC** | Area Under the Uplift Curve |
| **AUUUC** | Area Under the Unconfounded Uplift Curve |
| **BN** | Bayesian Network |
| **CATE** | Conditional Average Treatment Effect |
| **CEVAE** | Causal Effect Variational Autoencoder |
| **Chi** | $\chi^2$-divergence |
| **CIA** | Conditional Independence Assumption |
| **ClsTrans** | Class Transformation |
| **ClsTrans (RF)** | Class Transformation with Random Forest |
| **ClsTrans (RF_IPS)** | Class Transformation (Random Forest with Inverse Propensity Scoring) |
| **ClsTrans_IPS (RF)** | Class Transformation with Inverse Propensity Scoring (Random Forest) |
| **CNN** | Convolutional Neural Network |
| **CTS** | Contextual Treatment Selection |
| **DBN** | Deep Belief Network |
| **DCRNN** | Diffusion Convolutional Recurrent Neural Network |

| | |
|---|---|
| **DDP** | Delta-delta-p |
| **DS** | Dataset Sampling |
| **ED** | Euclidean Distance |
| **EduAcc** | Education-accident |
| **ETC** | Eletronic Toll Collection |
| **GCN** | Graph Convolutional Network |
| **GMAN** | Graph Multi-Attention Network |
| **GNN** | Graph Neural Network |
| **GPS** | Global Positioning System |
| **GPU** | Graphics Processing Unit |
| **GRU** | Gated Recurrent Unit |
| **HA** | Historical Average |
| **HMM** | Hidden Markov Model |
| **IHDP** | Infant Health and Development Program |
| **IPS** | Inverse propensity scoring |
| **KL** | Kullback-Leibler divergence |
| **k-NN** | k-Nearest Neighbors |
| **LGBM** | Light Gradient Boosting Machine |
| **LR** | Logistic Regression |
| **LSTM** | Long Short-Term Memory |
| **MA** | Moving Average |
| **MAE** | Mean Absolute Rrror |
| **MAPE** | Mean Absolute Percentage Error |
| **ML** | Machine learning |
| **MLE** | Maximum Likelihood Estimator |
| **MMoE** | Multi-gate Mixture-of-Expert |
| **MRT** | Mass Rapid Transit |
| **MSE** | Mean-Squared Error |
| **MTUC** | Maximum of the True Uplift Curve |
| **MRes-RGNN** | Gated Residual Recurrent Graph Neural Network |
| **NLP** | Natural Language Processing |
| **NN** | Neural Network |
| **O-D** | Origin-Destination |

| | |
|---|---|
| **POI** | Point of interest |
| **QR** | Quick response |
| **RAM** | Random access memory |
| **RBM** | Restricted Boltzmann Machine |
| **RCT** | Randomized Control Trial |
| **RF** | Random Forest |
| **RMSE** | Root Mean Squared Error |
| **RNN** | Recurrent Neural Network |
| **SAE** | Stacked Auto-encoder |
| **SARIMA** | Seasonal Autoregressive Integrated Moving Average |
| **SLC** | Structure Learning Convolution |
| **STGCN** | Spatio-Temporal Graph Convolutinoal Network |
| **SUTVA** | Stable Unit Treatment Value Assumption |
| **SVR** | Support Vector Machine |
| **T-GCN** | Temporal Graph Convolutional Network |
| **TrGNN** | Trajectory-based Graph Neural Network |
| **TUC** | True Uplift Curve |
| **VAR** | Vector auto=regression |
| **XGB** | eXtreme Gradient Boosting |

# List of Notations

| | |
|---|---|
| $A$ | The weighted road adjacency matrix. |
| $\tilde{A}$ | The normalized variant of the weighted road adjacency matrix $A$. |
| $D$ | The traffic demand tensor. |
| $H_t$ | The initial prediction of traffic flows upon time $t$. |
| $K$ | The number of hops in a graph convolutional layer; |
| | The key matrix in the attention mechanism; |
| | The threshold for quota in the driver recipient selection problem. |
| $L$ | The graph Laplacian, a variant of $A$. |
| $M$ | The number of nodes in the road graph; |
| | the number of road segments in the road network; |
| | The output feature dimension in the graph convolutional layer; |
| | The number of features. |
| $N$ | The number of features; |
| | The number of samples. |
| $N_q$ | The number of query vectors in the attention mechanism. |
| $N_k$ | The number of key vectors in the attention mechanism; |
| | the number of value vectors in the attention mechanism. |
| $Q$ | The query matrix in the attention mechanism. |
| $S$ | The traffic status tensor. |
| $T$ | The number of time intervals; |
| | The treatment variable: education. |
| $T_d$ | The number of time intervals of a day. |
| $T_{in}$ | The number of input time intervals. |
| $U$ | The confounder: driver's riskiness. |
| $V$ | The value matrix in the attention mechanism. |

| | |
|---|---|
| $W$ | The parameters in the graph convolutional layer that controls the fusion of different features. |
| $X$ | Traffic flows;<br>Features. |
| $\hat{X}$ | Predicted traffic flows. |
| $X11$ | Bad records in history. |
| $X^{(S)}$ | The set of strong confounders. |
| $Y$ | The outcome variable: accident. |
| $Y(\cdot)$ | The potential outcome. |
| $Y_i^*$ | The transformed outcome. |
| $Z_i$ | The transformed outcome. |
| $d$ | The demand hop. |
| $d_k$ | The dimension of each query vector in the attention mechanism;<br>the dimension of each key vector in the attention mechanism. |
| $d_v$ | The dimension of each value vector in the attention mechanism. |
| $do(\cdot)$ | The do-operator. |
| $e_{ij}$ | Road incidence between road segment $i$ and road segment $j$. |
| $\hat{f}$ | The propensity score estimator. |
| $\hat{f}_\Theta$ | The prediction function $\hat{f}$ with parameter $\Theta$. |
| $i$ | The sample index. |
| $l(\cdot)$ | The function that re-indexes samples. |
| $p(\cdot)$ | The frequency. |
| $p_i$ | The propensity score for sample $i$. |
| $\hat{p}_i$ | The estimated propensity score for sample $i$. |
| $s$ | The status hop. |
| $t$ | The time interval. |
| $v_i$ | Road segment $i$. |
| $w(\cdot)$ | The sample weighting function. |
| $x^{(S)}$ | A particular value of the set of strong confounders. |

$\mathbf{N}(v_i)$ The set of downstream neighbors of road segment $v_i$.

$\mathbb{S}$ The traffic flow dataset.

$\mathbb{D}$ The uplift modeling dataset.

$\mathscr{E}$ The set of edges in the road graph.

$\mathscr{G}$ The road graph.

$\mathscr{P}$ The trajectory transition tensor.

$\mathscr{T}$ The trajectory of a vehicle.

$\mathscr{V}$ The set of nodes in the road graph;
the set of road segments in the road network.

$\alpha$ The attention coefficients.

$\theta$ The parameters in the graph convolutional layer that controls the fusion of different hops.

$\sigma$ The activation function.

$\tau_i$ The uplift for sample $i$.

$\hat{\tau}_i$ The uplift estimate for sample $i$.

# List of Figures

# List of Tables

# Abstract

The prosperity of big data has boosted the development of AI techniques in smart city. In this thesis, we study the adaptation of newly-emerging machine learning techniques to urban transport applications. We review existing tasks and techniques in both urban transport understanding and urban transport actuation, based on which we identify remaining challenges, which serve as the motivation of my PhD studies. My research during PhD studies primarily tackles two major challenges (in particular contexts), namely, the presence of non-recurrent mobility patterns and the experimental limitations in actuation tasks.

In the first work, we tackle the challenge of non-recurrent mobility patterns in the context of highway vehicular traffic flow prediction. We adopt spatiotemporal modeling techniques and propose to mine the transition patterns from historical trajectories to complement non-recurrent mobility patterns. Specifically, we devise a model, Trajectory-based Graph Neural Networks (TrGNN), that incorporates trajectory transition patterns into the spatiotemporal deep learning framework based on graph to improve the accuracy of traffic flow prediction. Experiments with our approach on a real-world dataset achieves significant improvement on prediction accuracy, especially in non-recurrent scenarios.

Our second work is motivated from the challenge of an experimental limitation encountered in the task of driver recipient selection for traffic safety education to reduce traffic accidents. We formulate the task into an uplift modeling problem, a typical problem in causal inference. We identify the challenge that due to the infeasibility of proper experiments (i.e. Randomized Control Trials) in such a real-world scenario, datasets often come with bias, which deteriorates the estimation of uplifts by existing models and evaluation metrics. This is a common challenge in uplift modeling across various domains (even beyond urban transport). We systematically study uplift modeling approaches and propose organic integration of sample re-weighting in existing uplift models and evaluation metrics. Extensive experiments with three real-world datasets as well as the case study on traffic safety education show significant performance gain

from our proposed approach.

A large portion of our research is found to be easily applicable to domains beyond urban transport. For example, we realize that the techniques for spatiotemporal modeling studied in our first work could be easily transferred to domains such as weather forecasting. Besides, the advance of sample re-weighting proposed in our second work can be applied to medical study with satisfactory results, as already demonstrated by our experiments with the infant dataset and the twins dataset, and can be further applied to domains like e-commerce marketing. Moreover, the major challenges we have identified in urban transport understanding and actuation, such as multi-source data fusion or knowledge transferability, can in fact be seen in various smart city applications beyond urban transport (e.g. surveillance, smart lighting, air quality management), and even domains beyond smart city. Hence, effective solutions to these challenges would bring benefits more than we expected.

# Chapter 1

# Introduction

With the increasing population and vehicles in cities, it is of greater importance to study their mobility and solve related problems, referred to as urban transport applications. Urban transport applications range from understanding to actuation. Typical problems in urban transport understanding vary in their contexts (such as vehicular mobility or human mobility), in their objectives (such as prediction or estimation), and in their degrees of sparsity (e.g. anomaly such as accidents or system breakdown). Accurate understanding of urban transport lays the foundation for its actuation. Urban transport actuation refers to the intervention with the transportation environment for the sake of certain improvement, such as traffic signal control and traffic safety education. Problems in urban transport understanding and actuation, from a modeling perspective, are problems of prediction and optimization, and the interplay between the two is often non-negligible.

The prosperity of big data has brought unprecedented opportunities to the development of AI techniques, and in particular, machine learning, as well as to its application in urban transport. Big data and AI techniques have been leveraged in urban transport applications for long. For prediction tasks, a variety of approaches have been adopted, such as naive approaches (e.g. HA [83]), traditional statistical approaches (e.g. time series analysis [45]), traditional machine learning (e.g. SVR [2], RF [13]), and its recently advanced branch, deep learning (e.g. CNN [69], RNN [77], GNN [160]). For optimization tasks, besides simulation (e.g. Sidra [3]) and operations research (e.g. mathematical programming [82]), machine learning models also come into play. For example, a recent trend explores the application of deep reinforcement learning to traffic signal control [147, 131].

However, challenges remain when machine learning models are adapted to urban transport applications. These challenges include, but are not limited to, the presence of non-recurrent mobility patterns (especially in understanding), and the limitations of proper experiments for model testing (especially in actuation).

My PhD study aims at resolving the challenges that remain in urban transport applications, including both understanding and actuation, via adapting the emerging modeling techniques from machine learning.

Our first work examines a prediction problem, namely, vehicular traffic flow prediction. When applying machine learning models in vehicular traffic flow prediction, we identify the challenge that the presence of non-recurrent traffic flow patterns limits the prediction performance of existing approaches. Therefore, we propose a novel spatiotemporal deep learning framework to mine the underlying causality of flows leveraging historical trajectories, so as to reach more accurate prediction results.

Our second work is motivated from an actuation task: to select the best set of drivers as recipients of traffic safety education to reduce traffic accidents. We discover the task as a causal inference problem, named uplift modeling, or individual treatment effect estimation. When applying existing uplift models in traffic safety education, we identify the challenge that due to the infeasibility of proper experiments (i.e. Randomized Control Trials) in such a real-world scenario, the dataset collected from historical observations comes with selection bias, which deteriorates the estimation of uplifts by existing models and also deteriorates the validity of existing evaluation metrics. Therefore, we systematically examine this challenge and propose and analyze sample re-weighting methods to correct the bias, in the hope of finding the most 'persuadable' drivers for education to reduce accidents despite the inaccessibility to their ground truths.

Section 1.1 and 1.2 briefly summarize our contributions in the two works.

## 1.1 Traffic Flow Prediction with Vehicle Trajectories

This work proposes a spatiotemporal deep learning framework, Trajectory-based Graph Neural Network (TrGNN), that mines the underlying causality of flows from historical vehicle trajectories and incorporates that into road traffic prediction. The vehicle trajectory transition patterns are studied to explicitly model the spatial traffic demand via graph propagation along

the road network; an attention mechanism is designed to learn the temporal dependencies based on neighborhood traffic status; and finally, a fusion of multi-step prediction is integrated into the graph neural network design. The proposed approach is evaluated with a real-world trajectory dataset. Experiment results show that the proposed TrGNN model achieves over 5% error reduction when compared with the state-of-the-art approaches across all metrics for normal traffic, and up to 14% for atypical traffic during peak hours or abnormal events. The advantage of trajectory transitions especially manifest itself in inferring high fluctuation of flows as well as non-recurrent flow patterns.

## 1.2 Towards Sample Re-Weighting in Uplift Modeling

Uplift models often rely on Randomized Control Trials (RCTs) and assert strong assumptions on data such as unconfoundedness, while in real-world applications data are often collected as observables at the presence of confounding bias. This work analyzes methods that bridge the gap between existing uplift models and real-world applications due to confounding bias. First, we systematically summarize the assumptions made by existing uplift models either explicitly or implicitly, with attention to the unconfoundedness assumption. Second, we emphasize the role of sample re-weighting in eliminating confounding bias: for uplift modeling, we analyze the pros and cons of direct dataset sampling with four modes of implementation proposed, and we propose two novel approaches to organically integrate inverse propensity scoring into the structural design of the model; for uplift evaluation, we propose two adapted metrics, namely, AUUUC and MTUC. Extensive experiments are conducted on three datasets with a variety of uplift frameworks and models, and results show that an organic integration of inverse propensity scoring into the structural design of the model has great potential for performance gain. Specifically, our proposed model Class Transformation (Random Forest with Inverse Propensity Scoring) increases AUUUC by up to 46%, and in a case study of education recipient selection, it would help traffic police further reduce monthly accident rate by 3.4%.

## 1.3 Organization of This Thesis

This thesis consists of five chapters. We introduce the background and related work, and elaborate the remaining challenges in Chapter 2. In Chapter 3, we introduced our first work with

TrGNN, a spatiotemporal deep learning framework for vehicular traffic flow prediction. We introduce our second work on sample re-weighting in uplift modeling in Chapter 4. Finally, we conclude the thesis and imagine future work in Chapter 5.

# Chapter 2

# Background and Related Work

Our research work throughout the PhD program and in this thesis is around solving problems in the urban transport applications. The contexts would involve vehicular mobility (such as vehicles moving on a road network), or human mobility (such as passengers in a public transit system like MRT, or pedestrian on the streets). These different contexts give rise to a variety of tasks to serve different purposes. For example, some tasks aim at understanding the mobility patterns better, such as traffic prediction. In particular, many of these understanding tasks care about anomalies (for example, accidents), because usually these anomalies are the ones that really have strong impacts on the urban mobility patterns. Another type of tasks focuses on actuation, where we really intervene with the environment for the sake of certain improvement, and one example is traffic safety management.

If we summarize all the tasks from a technical perspective, we are actually dealing with an interplay between two problems: prediction (for urban transport understanding) and optimization (for urban transport actuation). With prediction, we understand things that are not explicitly known, and then with optimization, we make a difference, and see what happens, and based on that impact modify our action, and so on and so forth.

Therefore, we divide the review of background and related work into two parts: urban transport understanding (Section 2.1) and urban transport actuation (Section 2.2). We discuss common applications and techniques required in each domain. Based on the review, we identify challenges that remain in the application of machine learning models to urban transport (Section 2.3), which serve as the motivation of my PhD studies in this direction.

## 2.1 Urban Transport Understanding

Urban transport understanding involves understanding of current or future traffic patterns in urban cities. Typical problems in urban transport understanding (Section 2.1.1) vary in their contexts (such as vehicular mobility or human mobility), in their objectives (such as prediction or estimation), and in their degrees of sparsity (e.g. anomaly such as accidents or system breakdown). These problems often require AI techniques for spatiotemporal modeling (Section 2.1.2), as well as the availability of sensing data (Section 2.1.3).

### 2.1.1 Applications

Tasks in urban transport understanding can be categorized by their contexts. We provide a summary of existing tasks of our interest for vehicular mobility, human mobility, on-demand services and others.

- Vehicular mobility. Vehicular mobility understanding tasks are usually associated with a road network. Macroscopic tasks include traffic flow/speed/density estimation and prediction, detection and prediction of accidents and its impact; microscopic tasks include travel time estimation, trajectory recovery, and trajectory prediction. Modeling vehicular mobility plays an important role in actuation (to be discussed in Section 2.2) such as to improve efficiency in traffic control and planning, to avoid congestion and reduce accidents, and to save environmental resources. Our studies later in Chapter 3 will focus on the context of vehicular mobility.

- Human mobility. Human mobility can occur in public transit systems such as railway or bus, or in a 2D subspace such as streets and grounds, or in a 3D subspace such as shopping mall. Macroscopic understanding tasks include crowd flow estimation and prediction, detection and prediction of system breakdown and its impact; and microscopic understanding tasks include travel time estimation, trajectory recovery, trajectory prediction, and transportation mode inference. Understanding human mobility plays an important role in public resource allocation (e.g. public transport, policing), as well as marketing.

- On-demand services. On-demand services include taxi, ride-hailing, bike-sharing, and so on. Understanding tasks include prediction of demand and supply of these services.

- Others. Other urban transport understanding tasks include analysis of city functionalities, such as region-wise Origin-Destination (O-D) analysis and Point of Interest (POI) tagging.

### 2.1.2 Spatiotemporal Modeling

As urban transport often demonstrates strong spatiotemporal properties, spatiotemporal modeling plays a crucial role in most of the tasks in urban transport understanding. Hence, it is of our interest to explore existing solutions in spatiotemporal modeling and their adaptations to different tasks in the domain of urban transport understanding. In this subsection, we provide a review of existing spatiotemporal tasks in urban transport understanding, and existing approaches in spatiotemporal modeling, most of which have been applied to urban transport understanding. We provide a guideline for judging the suitability of applying a spatiotemporal modeling approach to a certain task in urban transport understanding. Finally, we identify potential gaps for future study.

#### Tasks

We identify common spatiotemporal tasks related to urban transport understanding. Prediction is the most common type of tasks in urban transport understanding, such as predicting an event (e.g. traffic accident), a trajectory (vehicular trajectory), or a macroscopic metric (traffic flow). To name a few other types of tasks: representation learning which extracts features from spatiotemporal data (e.g. trajectories) to facilitate more specific downstream tasks; estimation (e.g. traffic estimation, trajectory recovery) which infers fine-grained knowledge from coarse data; anomaly detection (e.g. accident detection) which identifies abnormal patterns from the data to raise alarms; multi-source data fusion which integrates spatiotemporal data with other data sources to improve the performance of various tasks. In general, tasks vary in their contexts, objectives, and corresponding data types. One particular spatiotemporl task is vehicular traffic flow prediction to be detailed in Chapter 3.

Table 2.1: An overview of spatiotemporal modeling approaches and their applications.

| Year | Model | Spatial | Temporal | Remarks | Applications |
|---|---|---|---|---|---|
| - | HA[83] | - | Historical Average | periodicity | traffic prediction |
| - | MA | - | Moving Average | stationarity | traffic prediction |
| 1996-2014 | time series | - | ARIMA[11],SARIMA[132] | linear | traffic flow/speed prediction; travel time estimation |
| | | VARMA[53],STARIMA[79] | | | |
| 2006 | BN[116] | Bayesian Network | | probabilistic | traffic flow prediction |
| 2015 | SVR[2] | Support Vector Regression | | non-linear | traffic flow prediction |
| 2015 | SAEs[74] | Stacked Auto-encoders | | non-linear | traffic flow prediction |
| 2014-2016 | Boltzmann machines | RBM[90];DBN[50] | | probabilistic | traffic speed prediction |
| 2016 | kNN[134] | k-Nearest Neighbors | | - | traffic flow prediction |
| 2016 | DeepST[151] | CNN | FullyConnected | region-based | crowd flow prediction |
| 2016 | ConvLSTM[136] | CNN | LSTM | region-based | precipitation forecasting |
| 2017 | CNN[76] | CNN | | time-space | traffic speed prediction |
| 2017 | Deep LSTM[146] | - | LSTM | accident | traffic speed prediction |
| 2017 | ST-ResNet[150] | ResNet | ResNet | region-based | crowd flow prediction |
| 2018 | DMVST-Net[143] | CNN | LSTM | region-based | taxi demand prediction |
| 2018 | MURAT[66] | Graph Embedding | ResNet | multi-task | travel time estimation |
| 2018 | T-GCN[156] | GCN | GRU | graph-based | traffic speed prediction |
| 2018 | DST-GCNN[127] | GCN | CNN | dynamic graph | traffic flow prediction |
| 2018 | DCRNN[67] | Diffusion Conv | Seq-to-seq with GRU | graph-based | traffic speed prediction |
| 2018 | STGCN[145] | CNN | CNN | parallel run | traffic speed prediction |
| 2018 | Q-Traffic[70] | CNN | Seq-to-seq with LSTM | map queries | traffic speed prediction |
| 2018 | DeepMove[35] | - | GRU + Attention | attention | human trajectory prediction |
| 2019 | DMPP[92] | Deep Mixture Point Process | | probabilistic | accident prediction |
| 2019 | LightNet[39] | Seq-to-seq with ConvLSTM | | - | lightning prediction |
| 2019 | ST-MGCN[38] | GCN | Contextual Gated RNN | multi-graph | ride-hailing demand forecast |
| 2019 | MRes-RGNN[16] | Diffusion Conv | GRU | residual unit | traffic speed prediction |
| 2019 | ST-MetaNet[94] | GAT | GRU | attention | traffic flow/speed prediction |
| 2019 | STG2Vec[68] | Graph Embed+Att | LSTM | attention | bike-sharing demand forecast |
| 2019 | STDN[142] | CNN | LSTM+Attention | attention; region-based | taxi demand forecast; ride-hailing demand forecast |
| 2020 | GMAN[158] | Graph Attention | Attention | multi-head att | traffic flow/speed prediction |
| 2016-2018 | [124] [51] | Reinforcement Learning | | peak hours | traffic/passenger flow control |
| 2018 | TT-DL[125] | CNN + Transfer Learning | | - | traffic flow prediction |
| 2018 | RegionTrans[126] | ConvLSTM + Transfer Learning | | - | crowd flow prediction |

### Approaches

In Table 2.1, we provide a summary of spatiotemporal modeling approaches that have been explored as well as their applications in urban transport understanding (or else) up to the year 2019. We scanned through works primarily retrieved from the Google Scholar website with keywords such as 'spatiotemporal modeling' and 'traffic prediction', followed by scanning through the papers cited in these works, with special attention to papers published in top-tier conferences and journals in the domains of machine learning (e.g. NIPS, ICLR, ICML, AAAI) and intelligent transportation systems (e.g. Transportation Research, T-ITS). We analyze the works in terms the publication time, the model applied (and in particular, the spatial and temporal elements involved in the design of the model architecture), and the specific task in urban transport understanding to solve.

The most naive approaches are HA and MA, which simply takes the average of historical readings as future measurement. These approaches are straightforward, but they make strong assumptions on the periodicity or stationarity of traffic, which seldom hold for traffic data. For decades, time series approaches (e.g. ARIMA and its variants) have been thoroughly exploited to capture the temporal correlations of traffic. However, these approaches rely on strong assumptions about linearity and stationarity of traffic and often ignore the spatial impact from neighboring traffic. Later on, traditional machine learning models come into play, either discriminative (e.g. kNN, SVR) or generative (e.g. RBM, BN), which are able to model nonlinear or complex spatiotemporal correlations. With recent advances in AI, a spatiotemporal deep learning framework becomes the favorite due to its high learning capability from big data: it models the spatial correlations via either CNN (as for images) or GCN (as for graphs) to capture the impact from neighborhood traffic; it models the evolution of traffic via either RNN (e.g. LSTM, GRU) or sequence-to-sequence RNN to capture the temporal correlations; and it makes it flexible to embed auxiliary information (e.g. periodicity, weather, event) as a complement. Attention mechanism has also become popular since 2019, to replace the spatial and/or temporal component in the framework. Many instances of these approaches are applied or adapted to various tasks in urban transport understanding. Meanwhile, reinforcement learning and transfer learning have also been applied to solve tasks like traffic control and inter-context traffic prediction respectively. We refer interested readers to surveys [128, 83, 144, 135] for detailed reviews of spatiotemporal modeling. We will propose a new spatiotemporal modeling approach in Chapter 3, and compare it with some of the approaches discussed above.

### Guideline for new applications

We raise up a few perspectives to consider when applying a spatiotemporal modeling approach to a certain task in urban transport understanding.

Spatially:

- What is the spatial space (e.g. grid-like, graph, 2D, 3D)?
- What is the granularity level (e.g. point, road segment, region)?
- Are exact locations or approximate locations needed?
- Do I need microscopic (e.g. trajectory) or macroscopic (e.g. flow) data?
- What should be the scope of the neighborhood, and how should neighbors be selected?

Temporally:

- What is the granularity level, in minute (e.g. travel time estimation) or in hour (e.g. traffic prediction) or else?
- Is the sampling frequency of the data sufficient to support our task?
- Does the dataset come with stationary property?
- Does the task require real-time response?

A systematic preliminary analysis of the questions above would help researchers to decide on representative datasets for experiments, to set the correct data preprocessing parameters as well as to select the most appropriate modeling approaches.

### Research gaps

We identify a few research gaps in the application of spatiotemporal modeling approaches to urban transport understanding. Namely,

- The need for generic solutions. The performance of a model strongly depends on tasks, and even strongly depends on datasets - there is not a single 'best' model. Hence, there is a need to design more generic models: models that are transferable from one city to another (e.g. in POI mining, traffic prediction), from one context to another (e.g.

expressway network vs downtown), from microscopic to macroscopic (e.g. trajectory vs flow), and across different measures (e.g. flow vs speed).

- The complexity of pedestrian movement. Pedestrian movement is complex in that pedestrian move flexibly in a 2D (e.g. ground) or 3D (e.g. shopping mall) space, unlike vehicular mobility which is constraint by a road network. There is limited work on the analysis of pedestrian movement, due to limited data sources. However, with the increasing availability of surveillance camera data, the analysis of pedestrian movement can be a promising direction. One interesting problem is to analyse pedestrian flow based on approximate, incomplete pedestrian trajectories, which could involve geometric inference.

- The utilization of a transformer [122]. Transformer is an attention-based encoder-decoder architecture introduced in Natural Language Processing (NLP) that can be trained in parallel and becomes the new basic building block in NLP. Being a newly emerging and increasingly popular deep learning model in recent years, we see huge potential for adapting it to urban transport understanding, with the ability to model urban transport data booth spatially and temporally. Works on this adaptation exist but are limited, and we believe this is a promising direction in solving urban transport understanding problems.

### 2.1.3 Sensing Data

With the development of sensing devices and sensing techniques, there have been increasingly available data for us to study urban transport, and they come on a larger scale and in a deeper level of granularity. However, limitations exist in the sensing devices as well as in the data collection process, and different types of sensors have different impacts on the quality of the datasets. In this subsection, we provide a review of different types of sensors and their impacts on the data quality. To complete the picture, we also introduce useful data sources other than sensing. We provide a guideline for modeling a certain task in urban transport understanding with sensing data. Finally, we introduce related works and point out a few research gaps for dealing with sensing data.

#### Sensors and their impact on data quality

The most common sensors for urban transport data collection include: loop detectors, GPS devices, toll collection devices and cameras.

- Loop detectors. Loop detectors lie on fixed locations in the road network, and collect macroscopic data about vehicles passing by, such as traffic flow, speed or density. The deployment of loop detectors are often sparse (e.g. on arterial roads only), resulting in spatially sparse data. Trajectory information (i.e. a sequence of locations stamped by timing for an individual vehicle) is not available from loop detectors.

- GPS devices. GPS devices move with vehicles or pedestrians to collect their location readings at a certain frequency (e.g. one reading per 30 seconds). The location readings are not exact, on an error range of 1 - 5 meters by radius, and map matching is often required for vehicles to map a reading to the road network. Trajectories can be extracted from GPS data via matching of vehicle or pedestrian identities between readings. The dataset is often partial, not able to cover all vehicles or pedestrian (e.g. trajectory data from a taxi company, via GPS devices deployed on taxis only). Our experiments in Section 3.5 leverage trajectory data that are extracted from GPS readings.

- Toll collection devices. Toll collection devices (e.g. fare card readers, ETC, QR code) are often deployed in a public or private transit system (e.g. highway, subway, bus) on fixed locations (except for bike-sharing), and collect data (including id, time, location, fare price) of vehicles or pedestrians on entries, exits, or checkpoints. Explicit O-D information and identity information are available, but intermediate trajectory is largely uncertain due to the sparseness in deployment.

- Cameras. Cameras have become the most promising sensors for future research on urban transport understanding, due to its increasing coverage and the newly emerged computer vision techniques in recent years. Camera data enjoy the advantageous properties of both loop detectors data (e.g. exact location) and GPS data (e.g. trajectories via identification), plus a rich array of attributes (e.g. vehicle type, pedestrian appearance). Nevertheless, camera data can be sparse/missing/wrong depending on computer vision techniques, camera deployment and functionality, and environmental factors.

Figure 2.1 illustrates the layout, pros and cons of these common sensors in the scenario of vehicular mobility. Other types of sensors are available for urban transport data collection. Table 2.2 lists out a variety of sensors and their impacts on the data acquired. We refer interested readers to a handbook [58] for a more detailed comparison of different sensors.

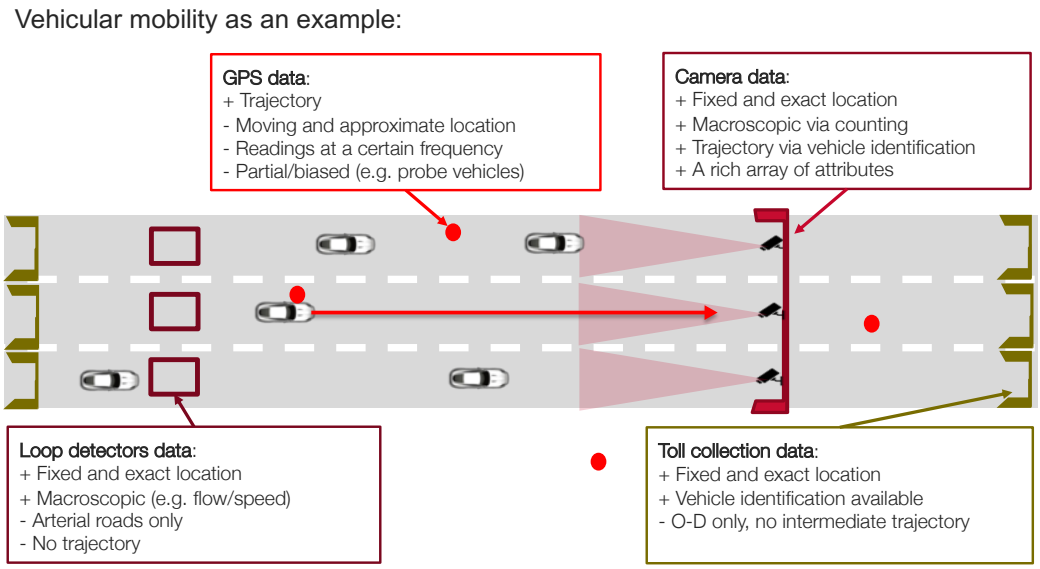



Figure 2.1: The most common sensors in vehicular mobility. The '+' and '-' signs denote pros and cons.

### Alternative data sources

It is worth mentioning that apart from sensing data, other types of data can also have strong impacts or implications on urban transport. These data include but are not limited to periodicity (e.g. time of day, day of week, public holidays, seasons), special events (e.g. concerts, sports games), anomalies (e.g. accident records), road conditions (e.g. road closure, road construction), environment (e.g. weather, air quality, lighting conditions), functional areas (e.g. POIs), and map queries. Incorporating these miscellaneous data could significantly benefit the performance of machine learning models in urban transport understanding. Our work in Section 3.4 leverages periodicity information, and design for periodic features to improve the prediction accuracy.

### Guideline for leveraging sensing data

We raise up a few perspectives to consider when modeling a certain task in urban transport understanding based on sensing data.

To leverage sensing data,

Table 2.2: An overview of sensing techniques and their impact on data quality.

| **Sensors** | **Data entry** | **Data property** |
|---|---|---|
| Inductive loop | flow/speed/density | exact location, macroscopic data, trajectory not available |
| Magnetic | | |
| Microwave radar | | |
| Infrared | | |
| Laser radar | | |
| Audio | | |
| Infrared | | |
| Automated Fare Collection | entry/exit/checkpoint | exact location, O-D |
| Call Detail Record | pedestrian trajectory | approx location, large-scale, O-D |
| GPS (smartphone) | | approx location (map matching required for vehicles) |
| GPS (vehicles) | vehicular trajectory | |
| Camera | videos | exact location, rich array of data |

- Which data source(s) are available, i.e. granted permit of usage by owners, easy to obtain and legally allowed?
- Should we use one single data source or fuse multiple data sources?
- Which data source(s) is(are) the most appropriate?
- Is data preprocessing required, and is any relevant preprocessing technique required?
- Need data be collected in real-time, and what is the tolerance of latency in data acquisition?

How to utilize sensing data in a proper way in order to solve a specific task in urban transport understanding is a practical and interesting problem.

### Research gaps

Depending on the nature of the datasets and the task, we may need to address several issues when using sensing data for tasks in urban transport understanding. We introduce existing work and identify a few research gaps toward this direction. Namely,

- Data conversion. Depending on the task to fulfill, raw data collected from sensors may need to be converted into another data type. For example, [88] designs an HMM-based map matching algorithm to convert GPS readings of probe vehicles into trajectories on

a road network to serve road network-based tasks. Computer vision techniques can be applied to extract vehicle related features (e.g. vehicle plate number, vehicle appearance) from camera snapshots [120]. We would expect more work on data conversion when more data are collected in urban transport understanding for the sake of future research.

- Imperfect data quality (e.g. sparse/missing/wrong data). Data collected from sensors are often sparse in space and/or time as discussed above; and data can be missing or wrong due to various reasons in the collection process (e.g. faulty devices, vague videos from cameras and wrong vehicle identity recognition, and noise from the environment). As an attempt to improve data quality, [71] proposes to estimate traffic speeds of all road segments from sparse trajectories via compressive sensing techniques. However, there is a lack of work on a variety of alternative scenarios, such as to recover fine-grained data from coarse data, to remove noise and uncertainties from raw data, and to extract a cleaned version of data with higher confidence. We believe that some of these issues can be resolved via the incorporation of spatiotemporal reasoning.

- Multi-source data fusion. Data come from different types of sources (e.g. flow, speed, periodicity, weather). A natural way to overcome the weakness of data quality is to combine multiple data types that could complement each other. For example, [120] recovers trajectory data from camera sensing data by fusing information from multiple data sources including vehicle identity, vehicle appearance features as well as spatiotemporal constraints; as discussed in Section 2.1.2, a deep learning framework would make it flexible to embed auxiliary information [70] into the model, or combine multi-model data by a simple concatenation [32]; [5] attempts to estimate traffic density fusing traffic flow data from videos and travel time data from GPS readings of probe vehicles; and [118] fuses license plate recognition data from videos with GPS trajectory data of probe vehicles to estimate queue length at intersections. We believe that more work can be done in this direction.

## 2.2 Urban Transport Actuation

Urban transport actuation refers to intervening with the environment for the sake of certain improvement in urban transport. Section 2.2.1 lists out typical applications in urban transport

actuation. We usually regard a task in urban transport actuation as more challenging to solve than a task in urban transport understanding. To elaborate, urban transport actuation not only requires urban transport understanding in the first place, but also requires a decision strategy for intervention, which is by nature an optimization problem, and thus techniques for solving optimization problems (Section 2.2.2) are usually required. Moreover, urban transport actuation involves intervention that modifies the underlying distribution of urban transport data, making it harder to understand - this is where causal inference (Section 2.2.3) comes into play.

### 2.2.1 Applications

Tasks in urban transport actuation vary in their contexts and their objectives. Below we list out a few typical applications in urban transport where actuation is involved.

- Traffic signal control. Traffic signals control the order of traffic flows when multiple flows of vehicles meet at an intersection. A good traffic signal control strategy can improve traffic efficiency, reduce traffic accidents, and reduce air pollution. Traffic signals nowadays are mostly operated automatically (instead of manually). They can be operated according to pre-decided rules, or dynamically adjusted based on real-time traffic information, with or without machine learning models.

- Traffic safety education. Traffic safety education could raise public awareness of traffic safety and reduce traffic accidents. A good allocation of traffic police resources over education recipients, contents, or locations could achieve effectiveness on accident prevention and reduction. Our work in Chapter 4 discusses a particular problem in the context of traffic safety education.

- Traffic police scheduling. Traffic police scheduling allocates traffic police officers over traffic control duties. Traffic police scheduling is essential for a variety of objectives such as reducing traffic jams, preventing or reducing traffic accidents, and emergency vehicle routing [84].

- Railway/bus scheduling. A good schedule would achieve a proper allocation of trains or buses over time and space, to best meet the spatiotemporal-varying travel demand of passengers. It involves more specific tasks such as timetable scheduling and bus route design.

- Ride-hailing order dispatch. Ride-hailing order dispatch refers to a real-time matching from the riding demand (i.e. a pair of origin and destination) of a rider to an available driver nearby. To achieve effective matching on a large scale, the ride-hailing platform would take into consideration the distance between driver and rider, alternative driver/rider nearby, estimated time of arrival, pricing, safety factors, as well as rider/driver preferences.

- Vehicle/pedestrian/public transit route planning. Route planning advises the fastest or shortest route between an origin and a destination for the traveling users, subject to real-time traffic conditions, locations, transportation resources, and user preferences and environmental factors. Route planning is a crucial application for individual travellers. An efficient route choice would save travellers' time; and for a public application with users on a large scale, such as Google Maps, a good coordination of multiple drivers on a road network would further improve transportation efficiency and maximize the utilization of transportation resources.

On the one hand, some of these applications are popular for machine learning researchers. For example, we see recent works that apply deep reinforcement learning to traffic signal control or order dispatch. On the other hand, some applications, such as bus scheduling or route planning, have been in place for long, but their solutions still remain classical without much recent advance.

### 2.2.2 Optimization

Optimization, in the domain of operations research, refers to the set of problems which aim at selecting a strategy (i.e. an solution) that optimizes a certain goal (i.e. an objective function) given a set of constraints. In urban transport actuation, for example, an optimization problem may aim at improving traffic efficiency, ensuring public safety or ensuring fairness given limited transportation resources via an optimal resource allocation strategy.

Mathematically, a general optimization problem [31] can be formulated as

$$\begin{aligned} &\underset{x}{\text{Optimize}}\ f(x) \\ &\text{subject to } h(x) = 0, \\ &\qquad\qquad g(x) \leq 0, \end{aligned} \tag{Eq. 2.1}$$

where $f(x)$ represents the objective function; $h(x)$ represents the set of functions in equality constraints; and $g(x)$ represents the set of functions in inequality constraints.

Tasks in urban transport actuation can largely be transformed into optimization problems, ranging from the simplest linear programming to more complex problems such as combinatorial optimization. One particular task, driver recipient selection for traffic safety education, has been formulated into an optimization problem in Chapter 4. By formulating an actuation task into a typical optimization problem mathematically, one would find algorithmic solutions readily available. Alternatively, the optimization problems in urban transport actuation can also be defined arbitrarily, i.e. not falling into a conventional category of optimization problems in the domain of operations research, but designed considering the specific task together with its contexts, and then solved by more flexible approaches such as machine learning (e.g. deep reinforcement learning).

In Table 2.3, we illustrate typical works in recent years that convert an urban transport actuation task into an optimization problem, the solution of which may leverage machine learning techniques. In particular, our review finds deep reinforcement learning as the most welcome machine learning technique applied in urban transport actuation.

Table 2.3: Selected works on urban transport actuation: the applications, their formulation into optimization problems and the machine learning techniques used.

| **Application** | **Optimization** | **Machine learning techniques** |
|---|---|---|
| Traffic signal control [147] | Collaborative | Reinforcement learning |
| Traffic signal control [131] | Self-defined | Reinforcement learning |
| Ride-hailing order dispatch [137] | Combinatorial | Reinforcement learning |
| Taxi order dispatch [152] | Combinatorial | Bayesian framework |
| Route planning [10] | Shortest path | N.A. |
| Bus schedule optimization [130] | Mixed integer | k-NN |
| Vehicle routing problem [86] | Combinatorial | Reinforcement learning |
| Electric vehicle charging scheduling [34] | Multi-objective | N.A. |
| Electric vehicle charging scheduling [36] | Game problem | Reinforcement learning |
| Bike-sharing rebalancing [64] | Multi-objective | Reinforcement learning |
| Car-sharing rebalancing [102] | Markov decision process | Reinforcement learning |

### 2.2.3 Causal Inference

Different from urban transport understanding, urban transport actuation involves intervention that modifies the underlying distribution of traffic data. This often renders the original understanding inaccurate or biased, and the optimization problem has to be formulated based on the newly-modified distribution, of which we have little knowledge purely based on historically observed data. This is where causal inference comes into play.

Causal inference studies the causal effect of intervening one variable (called treatment variable) on the expected value of another variable (called outcome variable) [96]. In traffic safety education, for example, we may be interested at the causal effect of the traffic police's education on traffic accident rate. Causal inference aims at quantifying this amount of change from sufficient observations. However, a fundamental issue arising from the definition of causal effect is that for any sample observed in the dataset, the treatment can only be done to the subject once - either treat or not treat in the case of binary treatment settings. Thus only one part of the effect on the outcome variable can be observed, and the other part is never observed and is called counterfactual. The existence of counterfactuals is known as the fundamental challenge in causal inference.

In practice, two major types of approaches are available to solve this fundamental challenge. One is to conduct Randomized Control Trials (RCTs) [112], for example, A/B Tests. In RCTs, a representative group of the analyzed objects is usually selected for intervention experiments, and treatment is randomly assigned to each object in this group so that the treated subgroup and untreated subgroup (in the case of binary treatment) are similar (in terms of feature distribution). Under these conditions, the causal inference problem can be directly solved by comparing the outcomes of the two subgroups, and the estimation is theoretically unbiased.

However, experiments like RCTs are not always feasible in real-world applications, and in fact, often infeasible or prohibited in urban transport actuation due to legal or political constraints, data sensitivity issues, or safety concerns. In most real-world tasks in urban transport actuation, such as traffic safety education or traffic signal control, datasets often come from historical observations with bias - the treated subgroup and the untreated subgroup can follow significantly different feature distributions due to targeted selection. Section 4.1 will exemplify such bias in traffic safety education.

To alleviate the confounding effect in observed data, another type of causal inference approaches 'tune' the distribution of observed data to mimic an RCT. The tuning may be facil-

itated with the incorporation of prior knowledge or domain expertise, such as the conceptual building of a causal graph [96] to indicate the causal directions between any pair of variables in scope. One effective way to tune the observed distribution is sample re-weighting, provided that certain assumptions hold - this will be discussed in details in Chapter 4.

Conceptually, two major causal inference frameworks are proposed and widely accepted: the potential outcome framework [111, 105] and the Stuctural Causal Model [96]. Materials introducing the elements of causal inference are widely available, such as in an introductory book [97] and in a survey [149].

Research exists that applies causal inference and machine learning in urban transport actuation, but mostly within transportation safety studies, such as analyzing factors of traffic accidents [54, 24]. However, we notice some very recent trend that explores the application of causal inference in traffic signal control [139] as well. We do foresee a trend for exploring more tasks in urban transport actuation as problems in causal inference, and applying causal inference-based machine learning to urban transport actuation. In addition, we spot works that study causality in urban transport understanding, such as evaluating the causal interaction of traffic states among different road segments [138]. However, these studies focus on causal discovery instead of causal inference.

It is worth commenting on the relationship between causal inference and optimization mentioned in previously. Results of causal inference can be incorporated into an optimization problem. A typical solution to an actuation task could be to formulate the task into an optimization problem, whose objective function or constraints upon an intervention policy are estimated via causal inference, and then solve it. Such practice is not only demonstrated by our work on recipient selection for traffic safety education (Section 4.6.4), but also demonstrated by applications in other domains, such as recommendation [81] and marketing [155].

## 2.3 Challenges

Based on our review of common applications and techniques in urban transport understanding in Section 2.1 and in urban transport actuation in Section 2.2, we identify a few challenges that remain in the application of machine learning models to urban transport, which points out directions of my PhD studies.

**Challenge 1. Non-recurrent mobility patterns.**

Urban transport often demonstrates non-recurrent patterns, due to the fluctuation of traffic demand and supply in time and space, the complex nature of interactions among vehicles/pedestrians and the transportation network, as well as external factors from the environment. Examples include traffic accidents, railway system breakdown, road closure, or stampede. On the other hand, existing works are good at modeling recurrent mobility patterns, but they are less effective in modeling non-recurrent mobility patterns due to their rare occurrences and insufficient data support. This gap remains as a major challenge in traffic estimation and prediction, though many approaches have been proposed to alleviate or resolve it. We introduce an example of the non-recurrent patterns in the context of vehicular traffic in Chapter 3.

**Challenge 2. Experimental limitations in actuation.**

Actuation requires intervention in a transport system, but the decisions to be made on an intervention policy often rely on preliminary analysis without sufficient access to intervening the transport system in real-world experiments. This is a typical chicken and egg problem. A Semi-Simpson's paradox is introduced in Section 4.1 to demonstrate the deterioration of modeling results due to experimental limitations.

**Challenge 3. Multi-source data fusion.**

In many contexts, data come from multiple sources and can be fused to complement each other. As discussed in Section 2.1.3, a variety of work involves multi-source data fusion. However, challenges remain to integrate data in a proper way so that we can take good advantage of the nature of each source. We believe that more work can be done in this direction, considering the variety of tasks and data sources.

**Challenge 4. Imperfect data quality.**

As discussed in Section 2.1.3, data used to study urban transport can be sparse, missing or wrong due to the limitation of data collection via sensing devices. When solving certain tasks in unban transport understanding, we have to adapt our approach considering the nature and quality of the data in place.

**Challenge 5. Knowledge transferability.**

As displayed in Section 2.1.2, most of existing work in urban transport understanding are context-specific, strongly dependent on tasks and datasets. However, one cannot reject the potential of a domain transfer from a highway network to a railway network (both in the forms of lines with stations and intersections), or from one city to another (both consisting of downtown and urban areas and displaying similar POI profiles, for example). Nevertheless, very limited attempts exist to mine general knowledge from urban transport and apply it across various contexts.

Our research aims at tackling these challenges under different contexts. In particular, our first work tackles Challenge 1, non-recurrent mobility patterns, in traffic flow prediction, via spatiotemporal modeling with graph neural networks, which will be detailed in Chapter 3; our second work is motivated by Challenge 2, experimental limitations in actuation, in traffic safety education to reduce traffic accidents, to analyze the sample re-weighting techniques in uplift modeling, which will be detailed in Chapter 4. In addition, our collaborative work [80] tackles Challenge 1 in urban rail transit system via the modeling of multivariate point process; and our collaborative work [120] tackles Challenge 3 and 4 in reconstructing large-scale vehicle trajectories from camera sensing data via graph convolution, fusing vehicle identity information with vision-based information and spatiotemporal constraints by the road network. Details of these works can be found in our publications listed in Appendix A.

# Chapter 3

# Traffic Flow Prediction with Vehicle Trajectories[1]

Our first work examines a problem in urban transport understanding, namely, vehicular traffic flow prediction. When applying machine learning models in vehicular traffic flow prediction, we identify the challenge that the presence of non-recurrent traffic flow patterns limits the prediction performance of existing approaches, which corresponds to Challenge 1 in Section 2.3. Therefore, we propose a novel spatiotemporal deep learning framework to mine the underlying causality of flows leveraging historical trajectories, so as to reach more accurate prediction results.

## 3.1 Introduction

Robust and accurate predictions of vehicular traffic conditions (e.g., flow, speed, density), either short-term or long-term, is necessary for transportation services such as traffic control and route planning. The challenge of traffic prediction primarily stems from the complex nature of spatiotemporal interactions among vehicles and the road network.

Data driven approaches have been extensively exploited in predicting vehicular traffic on the road network. Early attempts leverage time series analysis [132], which primarily models the temporal correlations of traffic. Conventional machine learning models [116] are applied to learn the spatiotemporal correlations of traffic from historical data. Latest works apply deep

[1] This chapter is partially published on AAAI 2021 [65], of which I am the first author, with a patent filed (not yet granted) in China.


Regular pattern Non-recurrent pattern

(a) When only flow observations are available

Regular pattern Non-recurrent pattern

(b) When vehicle trajectories are available

Figure 3.1: The challenge in predicting non-recurrent traffic flows and how vehicle trajectory information may help.

learning to traffic prediction, and they typically follow a spatiotemporal framework, e.g., Graph Neural Network (GNN) [67], which demonstrates superior capability in learning complicated spatiotemporal correlations.

According to studies in transportation domain [110], the road traffic contains two parts: recurrent traffic, which often arises from periodic traffic demand such as daily commuters during morning and evening rush hours, and non-recurrent traffic, which is triggered by unexpected causes. There are two major types of non-recurrent traffic patterns as we have observed: unexpected change in travel demand and unexpected change in road capacity. An unexpected change in travel demand can be caused by an public transit system breakdown (e.g. an MRT breakdown, which results in more people from the train station calling taxis as an alternative, and thus unexpected higher travel demand in the road network), or by a major event; while an unexpected change in road capacity can be caused by an accident that affects the traffic conditions (especially those on a highway or in a tunnel where alternative routes are not readily available), or by a temporal road closure due to construction for example.

Existing approaches learn spatiotemporal traffic correlations from patterns that were seen in history, and thus are favourable in predicting recurrent traffic. In predicting non-recurrent traffic, however, existing approaches may fail to achieve the same level of accuracy, mainly due to insufficient observations of similar flow patterns in history. Figure 3.1(a) illustrates such an issue with an example - when only flow observations (i.e., the number of vehicles passing

each road segment) are available, existing approaches may learn spatiotemporal correlations of recurrent flow patterns among different road segments across different time, which cannot effectively reason how a previously unseen part of the traffic is credited to future road traffic.

In this work, we target at addressing the current challenge with non-recurrent traffic flow prediction - the challenge that historical flow data cannot provide insights on how non-recurrent traffic flows correlate in time and space. Thus, complementary to conventional spatiotemporal modeling using aggregated traffic flow observations, we also exploit intact vehicle trajectories to infer short-term traffic dependency. As suggested in Figure 3.1(b), trajectory data provide information of how each portion of a traffic flow transits from one road segment to another, and thus implies the dependency of upstream and downstream flows. The dependency embeds knowledge of how downstream traffic are caused by upstream traffic and thus may help infer traffic flow patterns that have not been seen before.

Incorporating trajectories to traffic flow prediction entails the following challenges: 1) trajectories are only observed from historical data; 2) traffic patterns at trajectory level need be aggregated properly to reflect the traffic patterns at flow level; and 3) future flows may deviate from observed patterns due to influence from the environment. To address those challenges, we propose a novel model, Trajectory-based Graph Neural Network (TrGNN), which learns trajectory transition patterns from historical trajectories and incorporates that into a spatiotemporal graph-based deep learning framework. The main contributions of our work are summarized as follows:

- We identify the challenge of predicting non-recurrent traffic flows, and we propose to incorporate vehicle trajectory data in traffic flow prediction. To the best of our knowledge, this is the first study that attempts to leverage vehicle trajectory data to mine the underlying causality of flows among roads.

- We design an end-to-end spatiotemporal graph-based deep learning model to predict traffic flows of the entire road network. Our model embeds trajectory transition into graph propagation along the road network to model the spatial traffic demand; it learns the temporal dependencies with an attention mechanism based on neighborhood traffic status; and finally it fuses multi-step predictions.

- We conduct extensive experiments [2] with real-world vehicle trajectory data and the re-

[2] Code and dummy data are available at https://github.com/mingqian000/TrGNN.

sults suggest that our model outperforms state-of-the-art approaches in terms of prediction errors across various scenarios (over 5% error reduction), and is especially superior in predicting non-recurrent flow patterns, e.g., during abnormal events (up to 14% error reduction).

The rest of this chapter is organized as follows. Section 3.2 discusses related work on vehicular traffic prediction. Section 3.3 defines the problem and introduces some preliminary knowledge. The proposed model TrGNN is introduced in Section 3.4, and experimentally evaluated in Section 3.5. Section 3.6 concludes this chapter.

## 3.2 Related Work

For decades, data driven approaches have been exploited to predict road traffic conditions, such as flow [74], speed [67], density [101], accident rate [115], and arrival time [161].

Early attempts with time series analysis model the temporal correlations of traffic, such as SARIMA [132] and VAR [15]. Those approaches rely on strong assumptions of linearity and stationarity and often ignore the spatial impact from neighboring traffic. Another line of research focuses on studies of conventional machine learning models, such as k-NN [25], Bayesian network [116], and SVR [121]. In those models, spatiotemporal features are manually designed and extracted, and the models are often shallow in structure with limited learning capability.

Recent advances in deep learning have motivated its application in traffic prediction [72]. Earlier neural network architectures include SAEs [74] and DBN [50]. State-of-the-art approaches typically follow a spatiotemporal framework: it models the spatial correlations by CNNs [145] viewing the map as an image, or by GCNs [67] viewing the road network as a graph; and it models the temporal evolution of traffic as a sequence of signals [156]. The spatiotemporal framework makes it flexible to embed auxiliary information such as weather conditions [143], accident data [146], map query records [70], and POIs [38]. Similar to these works, our model follows a graph-based spatiotemporal deep learning framework; in addition, we incorporate vehicle trajectory data into the design to address the challenge of predicting non-recurrent flows.

Among deep learning approaches, Graph Wavenet [133] and SLC [153] mine latent graph structures to capture long-range dependencies, and MRes-RGNN [16] designs a multiple hop

scheme to capture long-term periodic patterns; the design goals of those methods deviate from our key objective of predicting non-recurrent traffic which is often caused by sudden disruptions locally. Other methods explore temporal building blocks and combine them with graph convolution, e.g., attention in ASTGCN [42] and GMAN [158], gated recurrent unit in T-GCN [156], temporal convolution in STGCN [145]. Most of these methods aim at reducing overall prediction errors without specific focus on non-recurrent traffic.

Few existing works leverage trajectory data in traffic flow prediction. [154] leverages trajectories in traffic state estimation, only to calibrate the embedding of road intersections. In comparison, our work utilizes trajectories to mine the traffic flow transitions among road segments. Some work leverages trajectories for other purposes such as map generation [104] which is less relevant to the topic of our study.

## 3.3 Preliminaries

We first define the problem of traffic flow prediction, and then introduce some preliminary knowledge of Graph Convolutional Networks (GCNs) and attention mechanism.

### 3.3.1 Problem Definition

In traffic flow prediction, the target is to predict future traffic flows given historical traffic flows on a static road network.

**Definition 3.1 (Road Graph)** *The road network can be formulated into a directed road graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. $\mathcal{V} = \{v_i\}_{i=1,2,...,M}$ *is a finite set of nodes where each node* $v_i$ *represents a road segment* $i$*, and* $\mathcal{E} = \{e_{ij}\}$ *is a set of directed edges where each edge* $e_{ij} = (v_i, v_j)$ *indicates that road segment* $i$ *is an immediate upstream of road segment* $j$*, and* $A \in [0,1]^{M \times M}$ *represents the weighted road adjacency matrix. Each node* $v_i$ *has a self-loop, i.e.,* $e_{ii} \in \mathcal{E}$.

**Definition 3.2 (Traffic Flows)** *Traffic flow is defined as the number of vehicles passing by a road segment during a specific time interval. Given a road graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$*, we use* $X \in \mathbb{R}^{T \times |\mathcal{V}|}$ *to represent the time series of traffic flows, where for each interval* $t = 1, 2, ..., T$*,* $X_t \in \mathbb{R}^{|\mathcal{V}|}$ *represents the traffic flows of all road segments in the road network during time interval* $t$.

**Definition 3.3 (Trajectory)** *Given a road graph $\mathscr{G} = (\mathscr{V}, \mathscr{E}, A)$, we use $\mathscr{T} = (v_1, v_2, ..., v_I)$ to represent a trajectory of a vehicle, where each $v_i \in \mathscr{V}$ represents a road segment in the trajectory, satisfying $(v_i, v_{i+1}) \in \mathscr{E}, v_i \neq v_{i+1}, \forall i = 1, 2, ..., I-1$.*

**Problem 1 (Traffic Flow Prediction)** *Given a road graph $\mathscr{G} = (\mathscr{V}, \mathscr{E}, A)$, find a prediction function $\hat{f}$ with parameter $\Theta$ such that given traffic flows $X_{(t-T_{in}+1):t}$ within a historical window period $T_{in}$ up to time interval $t$, $\hat{f}$ estimates the most likely traffic flows $\hat{X}_{t+1}$ for the next time interval $t+1$, i.e.,*

$$\begin{aligned} \hat{X}_{t+1} &:= \hat{f}_\Theta(X_{t-T_{in}+1}, ..., X_t) \\ &\approx \arg\max_{X_{t+1}} \log p(X_{t+1} | X_{(t-T_{in}+1):t}), \end{aligned} \tag{Eq. 3.1}$$

We examine Problem 1 from a probabilistic point of view. It can be derived statistically that under the assumption of Gaussian noise, the maximum likelihood estimator (MLE) of prediction function $\hat{f}$ minimizes the mean-squared error (MSE), as proved in Proposition 1. Accordingly, we may model the prediction function $f$ as a graph neural network with parameter $\Theta$, and use MSE as the loss function for training.

**Proposition 1** *Assume that there exists a function $f$ that satisfies*

$$X_{t+1} = f(X_{(t-T_{in}+1):t}) + \mathscr{E}, \quad \mathscr{E} \sim \mathscr{N}(0, \sigma^2 I), \tag{Eq. 3.2}$$

*and assume that for a sample traffic flow dataset $\mathbb{S} = \{(X_{(t_i-T_{in}+1):t_i}, X_{t_i+1})\}_{i=1,2,...,N}$ the $N$ samples are independent and identically distributed (i.i.d.), then for a prediction model $f_\Theta$,*

$$\Theta_{MLE}(\mathbb{S}) = \arg\min_{\Theta} MSE_{\mathbb{S}}(\Theta). \tag{Eq. 3.3}$$

**Proof:** Denote dataset $\mathbb{S}$ by $\{(X_i, \mathbf{y}_i)\}_{i=1,2,...,N}$ for simplicity.

The log likelihood of the model is

$$
\begin{aligned}
& l(\Theta|\mathbb{S}) \\
&= \sum_{i=1}^{N} \log p(\mathbf{y}_i|X_i,\Theta) \\
&= \sum_{i=1}^{N} \log \frac{\exp\{-\frac{1}{2}[\mathbf{y}_i - f_\Theta(X_i)]^\intercal(\sigma^{-2}I)[\mathbf{y}_i - f_\Theta(X_i)]\}}{\sqrt{(2\pi)^{T_{in}}|\sigma^2 I|}} \\
&= \sum_{i=1}^{N} -\frac{1}{2}\left[(\mathbf{y}_i - f_\Theta(X_i))^\intercal(\sigma^{-2}I)(\mathbf{y}_i - f_\Theta(X_i)) + C\right],
\end{aligned}
\tag{Eq. 3.4}
$$

where C is a constant term w.r.t. $\Theta$.

Therefore, the parameters that maximize the log likelihood are

$$
\begin{aligned}
\Theta^* &= \arg\max_{\Theta} l(\Theta|\mathbb{S}) \\
&= \arg\max_{\Theta} \sum_{i=1}^{N} -\frac{1}{2\sigma^2}\|(\mathbf{y}_i - f_\Theta(X_i))\|^2 \\
&= \arg\min_{\Theta} \frac{1}{N|\mathscr{V}|}\sum_{i=1}^{N}\|(\mathbf{y}_i - f_\Theta(X_i))\|^2 \\
&:= \arg\min_{\Theta} MSE_{\mathbb{S}}(\Theta).
\end{aligned}
\tag{Eq. 3.5}
$$

Hence, the maximum likelihood estimator (MLE) of prediction function $\hat{f}$ with parameter $\Theta^*$ should minimize the mean-squared error (MSE).

### 3.3.2 Graph Convolutional Networks (GCNs)

To model the spatial traffic demand, we leverage the idea of graph propagation from GCNs.

A GCN is defined over a graph $\mathscr{G} = (\mathscr{V}, \mathscr{E}, A)$. It applies convolutional operations on graph signals in spectral domain [57, 26]. Given a graph signal $X \in \mathbb{R}^{|\mathscr{V}|\times N}$ where $N$ is the number of features, a typical formulation of a $K$-hop graph convolutional layer is

$$
GCN_{\mathscr{G}}(X;W,\theta) = \sigma(\sum_{k=0}^{K}\theta_k L^k X)W \tag{Eq. 3.6}
$$

where $L \in [0,1]^{|\mathscr{V}|\times|\mathscr{V}|}$ is the graph Laplacian, a variant of $A$, to control the graph propagation

across nodes; $\theta \in \mathbb{R}^K$ controls the fusion of different hops; $W \in \mathbb{R}^{N \times M}$ ($M$ is the output feature dimension) controls the fusion of different features; and $\sigma$ is the activation function.

In this work, to model traffic demand and traffic status, we adopt graph propagation (GraphProp), a simplified variant of GCN, with one single input feature (i.e., traffic flow) and thus ignoring the feature-wise parameters $W$:

$$GraphProp(X, A; K) := [X \,\|\, AX \,\|\, A^2X \,\|\, ... \,\|\, A^KX]. \quad \text{(Eq. 3.7)}$$

Instead of directly learning $\theta$ to fuse different hops of traffic demand, we define an attention mechanism to learn the temporal weights, as illustrated in Section 3.3.3.

### 3.3.3 Attention Mechanism

In learning a weighted sum of values, an attention mechanism [122, 123] replaces the weight parameters with a learning module in which the same set of parameters (called keys) are shared across all values in calculating the weights. A typical formulation of an attention mechanism given queries $Q \in \mathbb{R}^{N_q \times d_k}$, keys $K \in \mathbb{R}^{N_k \times d_k}$ and values $V \in \mathbb{R}^{N_k \times d_v}$ is

$$Attention(Q, K, V) := softmax(QK^\intercal)V. \quad \text{(Eq. 3.8)}$$

In this work, we apply the attention mechanism for a more flexible fusion of traffic demand based on traffic status.

## 3.4 Methodology

In this section, we introduce our proposed model, TrGNN, to address the problem of traffic flow prediction (Problem 1). The model follows a spatiotemporal framework, leveraging trajectory transition patterns. The extraction of trajectory transition is illustrated in Figure 3.2. An overview of the model architecture is illustrated in Figure 3.3. We elaborate each part of the architecture in the subsections below.

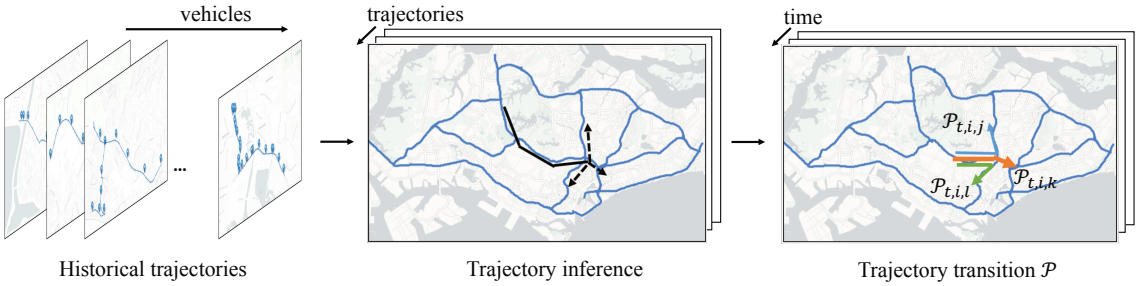



Figure 3.2: Learning trajectory transition from historical trajectories.

### 3.4.1 Trajectory Transition

We have illustrated in Figure 3.1 in the introduction the extra information gain from historical trajectory data when inferring non-recurrent traffic patterns. Compared to flows, trajectories provide essential knowledge about drivers' origin and destination (O-D), and help infer their choices of routes at road intersections. Figure 3.1(b) visualizes a contrast between green and dark red trajectories, which indicates that vehicles coming from different upstream road segments (or origins) may differ in their distributions of downstream road segments (or destinations). Hence, for non-recurrent traffic patterns, we may infer flows on a trajectory basis: first obtain the origins of the existing vehicles, and then based on their origins, infer the distribution of their destinations. In Figure 3.1(b), for example, we may infer that the extra spike of flow is more likely caused by extra vehicle flow of dark red trajectories instead of green trajectories.

We utilize historical trajectories to explicitly model the transition of flows between upstream and downstream road segments. In our work, the trajectory data are extracted from the GPS readings of taxis in Singapore, the source of which is detailed in Section 3.5.1. Figure 3.2 illustrates the extraction of trajectory transition. We view historical trajectories as Markov processes, and by aggregating trajectories of all vehicles, we may infer the transition of flows from one road segment to others. The rest of this subsection elaborates the extraction of trajectory transition.

The trajectory generation of a vehicle can be modeled as a first-order Markov process, assuming that the transition probability from each upstream road segment to each downstream road segment is stationary across days. We define a trajectory transition tensor $\mathscr{P} \in \mathbb{R}^{T_d \times |\mathcal{V}| \times |\mathcal{V}|}$, where each $\mathscr{P}_t \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ represents the trajectory transition probabilities for the $t^{th}$ time in-

terval of the day. The trajectory generation process can be represented as

$$\begin{aligned} P(\mathscr{T}|t) &= P((v_1, v_2, ..., v_I)|t) \\ &= \pi(v_1) \prod_{i=1}^{I-1} P(v_{i+1}|v_i; t) \\ &= \pi(v_1) \prod_{i=1}^{I-1} \mathscr{P}_{t,v_i,v_{i+1}}. \end{aligned} \tag{Eq. 3.9}$$

Alternatively, $\mathscr{P}$ can be derived from a higher-order Markov process, which would require larger sample size and higher computational complexity.

To estimate tensor $\mathscr{P}$, we collect historical trajectories of all vehicles from the training set, and aggregate the cumulative transition probability with respect to time of day, upstream road segment, and downstream road segment:

$$\hat{\mathscr{P}}_{t,v_i,v_j} = \frac{\#vehicles\,(v_i \to v_j|t) + \mathbf{1}[v_j \in \mathbf{N}(v_i)]}{\#vehicles(v_i|t) + \|\mathbf{N}(v_i)\|}, \tag{Eq. 3.10}$$

where $t$ stands for the $t^{th}$ time interval of day, and $\mathbf{N}(v_i)$ denotes the set of downstream neighbors of road segment $v_i$. To mitigate data sparsity, $\hat{\mathscr{P}}$ is smoothed out by adding a constant 1 for any pair of consecutive road segments.

The trajectory transition tensor $\mathscr{P}$ summarizes the probability distribution of drivers' choices of routes. In a macroscopic view, it approximates the transition of flows from upstream to downstream road segments in near future, and serves as a lookup table in the proposed TrGNN model.

### 3.4.2 Spatial Modeling of Traffic Demand

Based on trajectory transition, we design a graph propagation mechanism to infer the traffic demand in the spatial domain. Traffic demand refers to the short-range and long-range destinations of existing vehicles on the road network.

We leverage graph propagation from Graph Convolutional Networks (Section 3.3.2) to simulate the transition of vehicles along the road network. We perform graph propagation in $d$ hops, resulting in a graph of traffic demand for each hop. For each input time interval $t$, we can

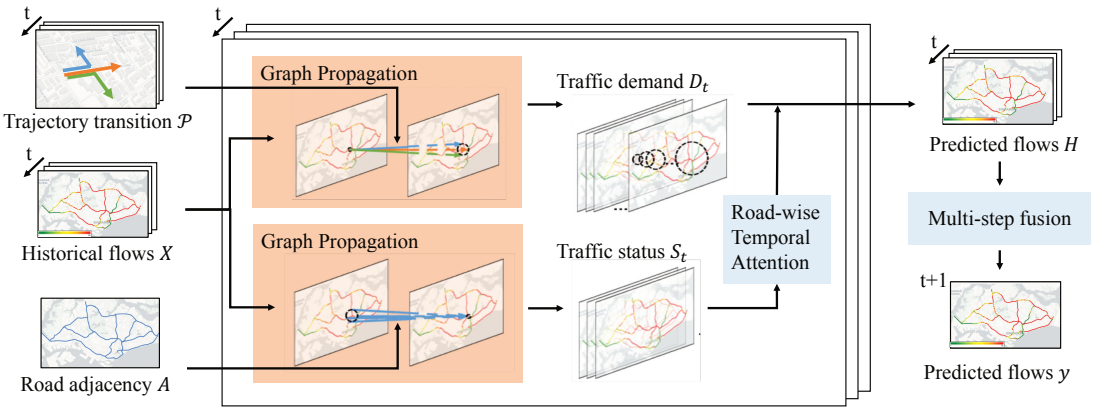



Figure 3.3: Trajectory-based Graph Neural Networks (TrGNN). The framework models spatial traffic demand via graph propagation based on trajectory transition, and models temporal dependencies via attention mechanism based on neighborhood traffic status. The final prediction is a fusion of multi-step prediction.

derive traffic demand $D_t \in \mathbb{R}^{|\mathcal{V}|\times(d+1)}$ via graph propagation:

$$\begin{aligned} D_t &= GraphProp(X_t, \mathscr{P}_t^\intercal; d) \\ &= [X_t \,\|\, \mathscr{P}_t^\intercal X_t \,\|\, (\mathscr{P}_t^\intercal)^2 X_t \,\|\, \ldots \,\|\, (\mathscr{P}_t^\intercal)^d X_t], \end{aligned} \tag{Eq. 3.11}$$

where $\|$ denotes concatenation, $\cdot^\intercal$ denotes matrix transpose, and parameter $d$ stands for demand hop, controlling the farthest possible destination.

The graph propagation simulates the propagation of flows along the road network, and as a result, the traffic demand $D$ is an aggregation of the short-range and long-range destinations (in different hops) of all vehicles in existing flows.

### 3.4.3 Temporal Modeling of Traffic Demand Based on Traffic Status

The modeling of traffic demand in Section 3.4.2, however, does not consider the propagation speed of flows, which should depend on traffic status. Traffic status refers to the overall traffic volume in the neighborhood of each road segment. If the traffic status is congested around a road segment (i.e., high volume of flows in the neighboring road segments), the propagation of flows along that road segment should be slow, and vice versa.

A temporal module is thus designed to infer how each hop of traffic demand, from short range to long range, corresponds to the future traffic flow in the targeted time interval. This is

done by assigning a weight to each hop of traffic demand via an attention mechanism (Section 3.3.3) based on traffic status. Thus, we first obtain traffic status, and then build an attention mechanism based on traffic status.

For each input time interval $t$, we obtain traffic status $S_t \in \mathbb{R}^{|\mathcal{V}| \times (2^{s+1}-1)}$ via graph propagation in $s$ hops from neighboring road segments. The graph propagation is done via dual random walk to incorporate both upstream and downstream traffic:

$$\begin{aligned} S_t &= GraphProp_{dual}(X_t, \tilde{A}; s) \\ &= [X_t \,\|\, \tilde{A}^\intercal X_t \,\|\, \tilde{A} X_t \,\|\, \tilde{A}^\intercal \tilde{A}^\intercal X_t \,\|\, ... \,\|\, \tilde{A}^s X_t], \end{aligned} \tag{Eq. 3.12}$$

where $\tilde{A}$ is a normalized variant of the weighted road adjacency matrix $A$, and parameter $s$ stands for status hop, controlling the radius of the neighborhood.

We apply a road-wise attention mechanism (referring to the dot-product attention in [122]) parameterized by keys $K \in \mathbb{R}^{|\mathcal{V}| \times (2^{s+1}-1) \times (d+1)}$, taking traffic status $S_t$ as queries and traffic demand $D_t$ as values, to assign weights $\alpha \in [0,1]^{|\mathcal{V}| \times (d+1)}$ to different hops in traffic demand $D_t$ and take the weighted sum as an initial prediction of flows $H_t \in \mathbb{R}^{|\mathcal{V}|}$:

$$\begin{aligned} H_t &= Attention(S_t, D_t; K) \\ &= \sum_{i=0}^{d} \alpha_{:,i} \odot D_{t,:,i} \\ &= \sum_{i=0}^{d} [softmax(S_t \circ K)]_{:,i} \odot D_{t,:,i} \end{aligned} \tag{Eq. 3.13}$$

where $softmax(\cdot)$ is applied over the dimension of demand hop, $\circ$ denotes road-wise matrix product, and $\odot$ denotes element-wise (or Hadamard) product.

### 3.4.4 Multi-Step Fusion

From a sequence of input flows $\{X_i\}_{i=t-T_{in}+1,...,t}$, we obtain a sequence of initial predictions $H \in \mathbb{R}^{T_{in} \times |\mathcal{V}|}$. The final layer of the model is a temporal fusion of $H$. We adopt a road-wise

fully connected layer. For each road segment $v$,

$$\begin{aligned} y_v &:= X_{t+1,v} \\ &= FullyConnected(H_{:,v}; \Theta) \\ &= \Theta^{\intercal} H_{:,v}. \end{aligned} \tag{Eq. 3.14}$$

Alternatively, this layer can be replaced by any RNN cell such as LSTM [47] or GRU [22]), or sequence modeling [117], for a longer-term prediction.

As a side note, the conservation of vehicles on the road network does not hold in practice. Future flows not only depend on trajectory transition within the road network, but also depend on new vehicles entering the road network (e.g., entering from the boundary of the region, or entering from a local road to an arterial road) and existing vehicles leaving the road network, which we call boundary flows. Since the boundary flows are strongly associated with drivers' O-D demand which is periodic, we embed some periodic features (e.g., time of day, is working day) into the multi-step fusion module to model the boundary flows.

## 3.5 Experimental Evaluation

### 3.5.1 Dataset Description

We evaluate our model with SG-TAXI, a real-world dataset comprising GPS mobility traces from over 20,000 taxis in Singapore. The dataset is provided by Singapore Land Transport Authority. We collect the GPS readings of all active taxis for a period of 8 weeks ($14^{\text{th}}$ Mar-$8^{\text{th}}$ May 2016). Each GPS reading comprises vehicle id, longitude, latitude, and timestamp. The road network comprises 2,404 road segments, covering all expressways in Singapore. The metadata of SG-TAXI dataset are summarized in Table 3.1.

Alternative datasets could be available for the study of traffic flow prediction with vehicle trajectories as well. For example, traffic flow data and vehicle trajectory data can be extracted from street cameras deployed in downtown regions - computer vision techniques can be applied to count and identify vehicles on the road from the vehicle snapshots [120]. While traffic flow data are relatively easy to collect and are publicly available in various countries and cities worldwide (such as the datasets used in [67, 145]), vehicle trajectory data are in general not easily accessible, as they involve identifying vehicles which are often claimed as confidential

by the data owners.

Table 3.1: Metadata of SG-TAXI dataset.

| Entries | Statistics |
|---|---|
| Map region | 50 km * 27 km |
| Road segments | 2,404 |
| Road lengths | 0 - 200m |
| Vehicles (daily unique) | 16.5k |
| Reading frequency | every 30 seconds |
| Period | 14$^{\text{th}}$ Mar - 8$^{\text{th}}$ May 2016 |
| Total GPS readings | 78 million |
| Avg flow per road | 142 vehicles per hour (vph) |
| Proportion of zero flows | 7.4% |

### 3.5.2 Data Preprocessing

We preprocess the SG-TAXI dataset in 4 steps:

(i) **Road graph formulation.** For the road graph $\mathscr{G} = (\mathscr{V}, \mathscr{E}, A)$, we calculate the weighted road adjacency matrix $A$ as the exponential decay of distance between roads:

$$A_{i,j} = \begin{cases} \lambda \cdot e^{-\lambda \cdot dist(v_i, v_j)} & \text{if } (v_i, v_j) \in \mathscr{E} \\ 0 & \text{o.w.} \end{cases} \tag{Eq. 3.15}$$

where

$$dist(v_i, v_j) = \begin{cases} 0, & \text{if } i = j \\ \frac{len(v_i)+len(v_j)}{2}. & \text{if } i \neq j, (v_i, v_j) \in \mathscr{E} \end{cases} \tag{Eq. 3.16}$$

We implement the road graph via Python NetworkX package, setting $\lambda = 1$.

(ii) **Map matching.** We apply the Hidden Markov map matching algorithm [88] to correct GPS readings to their corresponding road segments.

(iii) **Trajectory cleansing.** Given a sequence of mapped GPS points, we cleanse the vehicle's trajectory as follows: 1) eliminate duplicate records; 2) if GPS reading is off for over 10 minutes (e.g., the driver turns off the sensing device), split the trajectory; 3) if driver stays on the same road segment for over 2 minutes, split the trajectory; 4) if no path exists

between two consecutive GPS points (e.g., driver drives off the road network), split the trajectory; and 5) remove GPS points with extreme speed (i.e., speed derived from two consecutive GPS points exceeds 120 km/hr). Finally, we recover the full trajectory via Dijkstra's algorithm [30].

(iv) **Flow aggregation.** We aggregate trajectories into flows per road segment per 15-minute interval. We calibrate flows to correct the daily fluctuation in taxi arrangement and better represent the overall traffic flows in Singapore.

### 3.5.3 Experiment Settings

The model is trained on the preprocessed SG-TAXI dataset. The train-validate-test split is 5-1-2 week. Each data point consists of input flows for 4 intervals (i.e., 1 hour) and output flows for 1 interval (i.e., 15 minute). Flows are normalized before being input into the model. For hyperparameters, the demand hop $d$ is set to 75, i.e., the maximum number of road segments that a vehicle with a normal speed could traverse within a 15-minute interval, and the status hop $s$ is set to 3.

The model is implemented in PyTorch [95] on a single Tesla P100 GPU and is trained using Adam optimizer [56] to minimize MSE loss. The learning rate is initially set to 0.004 and is halved every 30 epochs. The maximum epochs to train is set to 100. Early stopping is applied on validation MAE. The training takes less than 4GB RAM and less than 1GB GPU memory.

### 3.5.4 Baseline Approaches

Our model TrGNN is compared to representative baseline methods of each type, including naive methods (HA, MA), time series analysis (VAR), conventional machine learning (RF), and deep learning (T-GCN, STGCN, DCRNN). Specifically,

(i) **HA** (Historical Average) is the average flow of the same time on the same day in the past four weeks;

(ii) **MA** (Moving Average) is the average flow of the previous 1 hour;

(iii) **VAR** (Vector Auto-Regression) [45] models the future flow as a linear combination of historical flows in 5-hop neighborhood, implemented in StatsModels [107];

(iv) **RF** (Random Forest) is a decision-tree-based ensemble method that fits a piece-wise function on historical flows in 5-hop neighborhood, implemented in Scikit-learn [98] with 100 trees;

(v) **T-GCN** (Temporal Graph Convolutional Network) [156] is a graph-based neural network that integrates GCN with GRU, implemented in Tensorflow [3];

(vi) **STGCN** (Spatio-Temporal Graph Convolutional Networks) [145] is a graph-based neural network that models both spatial and temporal dependencies via convolution, implemented in Pytorch [4]; and

(vii) **DCRNN** (Diffusion Convolutional Recurrent Neural Network) [67] is a graph-based neural network that integrates diffusion convolution on graph with sequence learning, implemented in PyTorch [5].

Additionally, we build a variant of TrGNN, **TrGNN-**, to analyze the utility of trajectories, by replacing the trajectory transition tensor $\mathscr{P}$ in TrGNN with the road adjacency matrix $A$.

### 3.5.5 Evaluation Metrics

We evaluate prediction results by three error metrics: MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), and RMSE (Root Mean Squared Error), the same as in [67]. For a dataset of N samples, let $X, \hat{X} \in \mathbb{R}^{T \times |\mathscr{V}|}$ denote the ground truth and predicted flows (as defined in Definition 3.2), then

- MAE (Mean Absolute Error)

$$MAE(X,\hat{X}) = \frac{1}{N|\mathscr{V}|} \sum_{i=1}^{N} \sum_{j=1}^{|\mathscr{V}|} |X_{t_i+1,j} - \hat{X}_{t_i+1,j}|, \qquad \text{(Eq. 3.17)}$$

- MAPE (Mean Absolute Percentage Error)

$$MAPE(X,\hat{X}) = \frac{1}{N|\mathscr{V}|} \sum_{i=1}^{N} \sum_{j=1}^{|\mathscr{V}|} \left| \frac{X_{t_i+1,j} - \hat{X}_{t_i+1,j}}{X_{t_i+1,j}} \right|, \qquad \text{(Eq. 3.18)}$$

[3] https://github.com/lehaifeng/T-GCN
[4] https://github.com/FelixOpolka/STGCN-PyTorch
[5] https://github.com/chnsh/DCRNN_PyTorch

Table 3.2: Performance of different approaches for traffic flow prediction on SG-TAXI dataset.

| Method | Overall<br>MAE/MAPE/RMSE | Peak hours<br>MAE/MAPE/RMSE | Non-peak hours<br>MAE/MAPE/RMSE | MRT breakdown<br>MAE/MAPE/RMSE |
|---|---|---|---|---|
| HA | 33.74 / 0.34 / 52.58 | 36.83 / 0.25 / 55.02 | 32.53 / 0.28 / 48.67 | **40.07** / 0.27 / **59.34** |
| MA | 31.55 / 0.35 / 47.69 | 36.14 / 0.26 / 53.18 | 28.18 / 0.27 / 39.41 | 44.85 / 0.30 / 71.43 |
| VAR | 29.27 / 0.33 / 43.22 | 34.23 / 0.24 / 49.71 | 28.10 / 0.26 / 39.28 | 40.68 / **0.27** / 64.41 |
| RF | 29.26 / 0.33 / 43.38 | 34.13 / **0.24** / 49.75 | **27.53** / **0.26** / **38.53** | 42.28 / 0.28 / 66.53 |
| T-GCN | 31.12 / 0.35 / 45.69 | 36.57 / 0.27 / 52.91 | 30.03 / 0.29 / 41.53 | 42.38 / 0.30 / 67.39 |
| STGCN | 29.88 / 0.33 / 44.51 | 34.86 / 0.24 / 50.86 | 27.94 / 0.27 / 39.05 | 42.19 / 0.28 / 66.40 |
| DCRNN | **29.01** / **0.31** / **43.12** | **33.74** / 0.25 / **48.88** | 27.75 / 0.27 / 38.74 | 40.39 / 0.28 / 64.28 |
| TrGNN- | 27.34 / 0.31 / 40.05 | 31.35 / 0.23 / 45.11 | 26.61 / 0.26 / 37.20 | 38.57 / 0.27 / 59.53 |
| **TrGNN** | **26.43** / **0.30** / **38.65** | **29.81** / **0.23** / **42.62** | **25.65** / **0.25** / **35.68** | **34.56** / **0.25** / **54.31** |
| %diff | -9% / -5% / -10% | -12% / -6% / -13% | -7% / -4% / -7% | -14% / -8% / -8% |

Numbers in bold denote the best baseline performance and the best performance.
%diff denotes the error reduction of TrGNN from the best baseline performance.

- RMSE (Root Mean Squared Error)

$$RMSE(X,\hat{X}) = \sqrt{\frac{1}{N|\mathcal{V}|}\sum_{i=1}^{N}\sum_{j=1}^{|\mathcal{V}|}(X_{t_i+1,j}-\hat{X}_{t_i+1,j})^2}. \quad \text{(Eq. 3.19)}$$

All errors are in the unit of vehicles per hour (vph). Lower errors indicate better performance of a model.

### 3.5.6 Results and Analysis

Table 3.2 summarizes the evaluation of different approaches for traffic flow prediction on SG-TAXI dataset. The comparison covers overall testing as well as specific scenarios including peak hours, non-peak hours and MRT breakdown.

**Overall Performance.** According to Table 3.2, the overall prediction errors of our model TrGNN are 26.43/0.30/38.65 vph for MAE/MAPE/RMSE, and TrGNN achieves over 5% error reduction from baselines across all metrics. The naive baselines generally give high errors, as they consider only temporal correlations of flows; MA is more accurate than HA, indicating that near-past flows play a stronger role than periodicity. VAR and RF perform better than the naive baselines, as they incorporate neighborhood flows to model spatial correlations; in

particular, RF performs better than VAR, implying that flows are not linearly correlated. For deep learning, DCRNN achieves the best results out of all baselines, indicating the capability of graph-based deep learning in capturing the spatiotemporal correlations. Finally, TrGNN outperforms all existing baselines in all metrics, which verifies the effectiveness of learning spatiotemporal transition of flows from trajectories.

The line plot in Figure 3.4 visualizes predicted flows of TrGNN and a few representative baselines. HA fits the worst to the ground truths, implying high variation of flows from week to week; while TrGNN and DCRNN are more sensitive to the real-time fluctuations of flows. If we look further into the peak hours indicated in the dashed box, TrGNN captures the fluctuations of flows slightly earlier than DCRNN.

**Peak hours and non-peak hours.** We select two typical periods for experiments: peak hours (8-10pm on working days, when public transport services become limited and the demand for taxis increases, thus with high fluctuation of flows); and non-peak hours (2-4pm on working days when people stay at offices and the demand for taxis stabilizes, thus with low fluctuation of flows). Results are summarized in Table 3.2 (under 'Peak hours' and 'Non-peak hours' column). In peak hours, absolute errors (MAE/RMSE) are consistently higher than in overall testing; while in non-peak hours, the results are the opposite. This meets our expectation, as predicting peak hour flows is more challenging due to higher fluctuation in traffic demand. In both peak hours and non-peak hours, TrGNN outperforms all baselines, and the error reduction of TrGNN is more significant during peak hours (6-13% reductions on the performance metrics).

Figure 3.4 displays the heatmap snapshots of the prediction errors of HA, DCRNN and TrGNN on the entire road network during selected peak hours. The color indicates increasing prediction error from green to red. A comparison of the heatmap snapshots suggests the robustness of TrGNN in capturing the periodic fluctuation of flows in peak hours.

**Abnormal event: MRT breakdown.** We analyze an abnormal event in Singapore, an MRT (Mass Rapid Transit) breakdown, when train services were disrupted due to power fault [21]. The disruption falls on a Monday night lasting for more than one hour, and it affects 52 train stations on 4 train lines, covering the whole area of west Singapore. In Figure 3.5, the heatmap visualizes the abnormal spike of flows of the affected region due to the increase in taxi demand during the MRT breakdown period, and the line plot visualizes the predicted flows on a sample abnormal road segment - TrGNN fits the best to the ground truths.

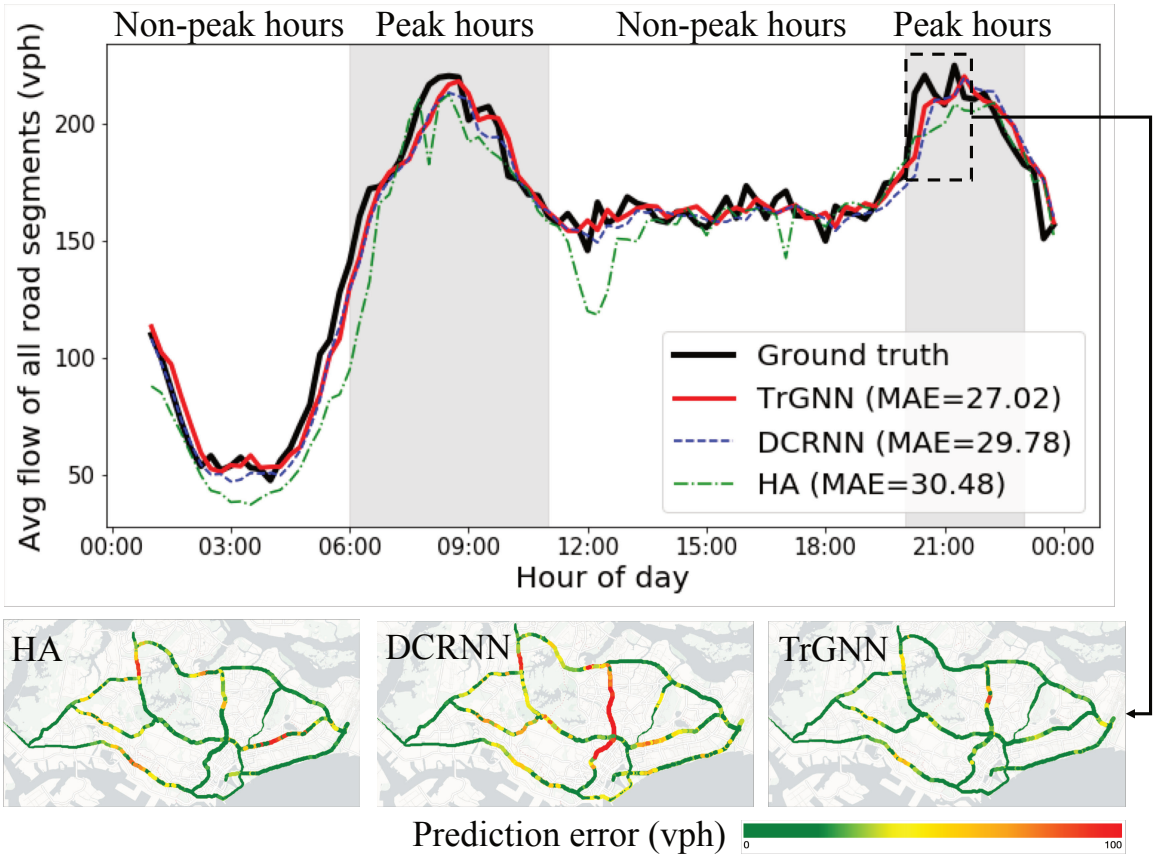



Figure 3.4: Line plot of predicted flows on the road network over a working day, and heatmap snapshots of prediction errors during peak hours.

We select road segments within a 3km neighborhood of any affected train station, and summarize their prediction results during the breakdown period in Table 3.2 (under 'MRT breakdown' column). Compared to 'Overall', we observe a significant increase in MAE for all baselines, ranging from 19% to 44%, which demonstrates the performance drop in predicting abnormal flows. Nevertheless, TrGNN outperforms baselines by a significant error reduction of 14%. The result suggests the capability of TrGNN in capturing the spatiotemporal causality even for non-recurrent flow patterns, instead of simply memorizing the historical flow patterns.

**Component analysis: trajectory transition.** Results in Table 3.2 show that compared to TrGNN-, in which the trajectory transition tensor is not used, TrGNN reduces the prediction errors on all metrics in all scenarios, especially in MRT breakdown where MAE drops from 38.57 vph to 34.56 vph. This verifies the effectiveness of trajectory transition in capturing flow

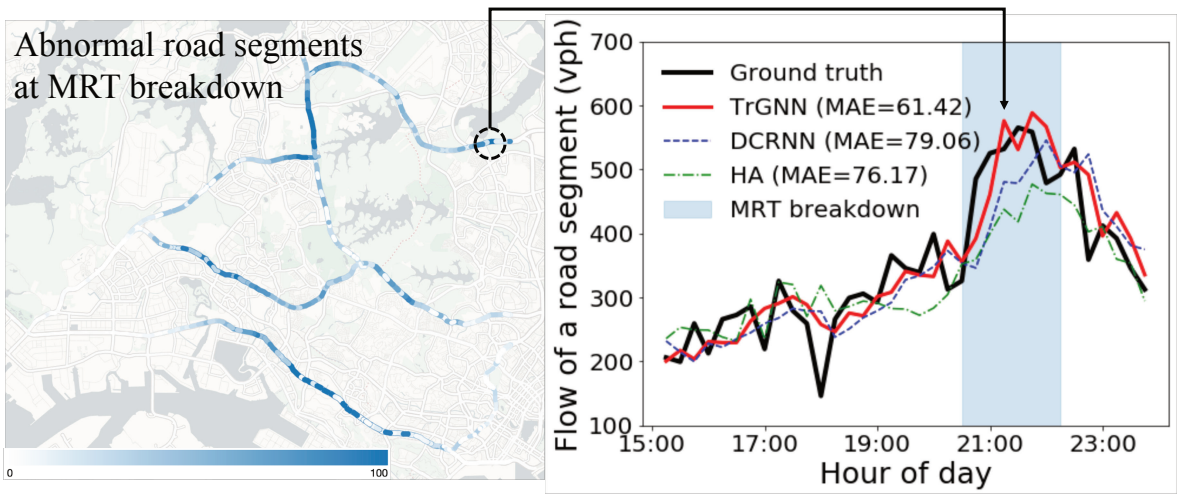



Figure 3.5: Heatmap of abnormal flows in west Singapore due to MRT breakdown, and line plots of abnormal flows and predicted flows on a road segment. In heatmap, the color scale indicates the amount of extra flow compared to that of a normal day.

dependency.

## 3.6 Conclusion and Future Work

This work proposes a spatiotemporal deep learning model, Trajectory-based Graph Neural Network (TrGNN), to solve the traffic flow prediction problem. The architecture leverages historical trajectory transition as an input into the graph-based deep learning framework. TrGNN is evaluated on SG-TAXI dataset. Results show that TrGNN outperforms state-of-the-art approaches, especially being superior in predicting non-recurrent traffic flows such as in MRT breakdown event. Potential future work includes expansion to a higher-order Markov model, longer-term prediction, and optimization of computational complexity in extracting trajectories. Moreover, our work points out a promising direction in incorporating trajectory data into traffic prediction.

# Chapter 4

# Towards Sample Re-Weighting in Uplift Modeling[1]

Our first work in Chapter 3 primarily deals with a prediction problem, vehicluar traffic flow prediction, in the domain of urban transport understanding. Our second work in this chapter will take a step further looking into an optimisation problem, driver recipient selection for traffic safety education, in the domain of urban transport actuation, and discuss some technical advances we have made in this context.

Our second work is motivated from an actuation task: to select the best set of drivers as recipients of traffic safety education to reduce traffic accidents. We discover the task as a causal inference problem, named uplift modeling, or individual treatment effect estimation. When applying existing uplift models in traffic safety education, we identify the challenge that due to the infeasibility of proper experiments (i.e. Randomized Control Trials) in such a real-world scenario, the dataset collected from historical observations comes with selection bias, which deteriorates the estimation of uplifts by existing models and also deteriorates the validity of existing evaluation metrics. It corresponds Challenge 2 in Section 2.3. Therefore, we systematically examine this challenge and propose and analyze sample re-weighting methods to correct the bias, in the hope of finding the most 'persuadable' drivers for education to reduce accidents despite the inaccessibility to their ground truths. Surprisingly, our studies in traffic safety education not only reach decent improvement in reducing traffic accidents, but are also found to be generally applicable to other uplift modeling scenarios such as in understanding

[1] This chapter is partially from a paper in submission, titled 'Towards Sample Re-Weighting in Uplift Modeling', of which I am the first author.

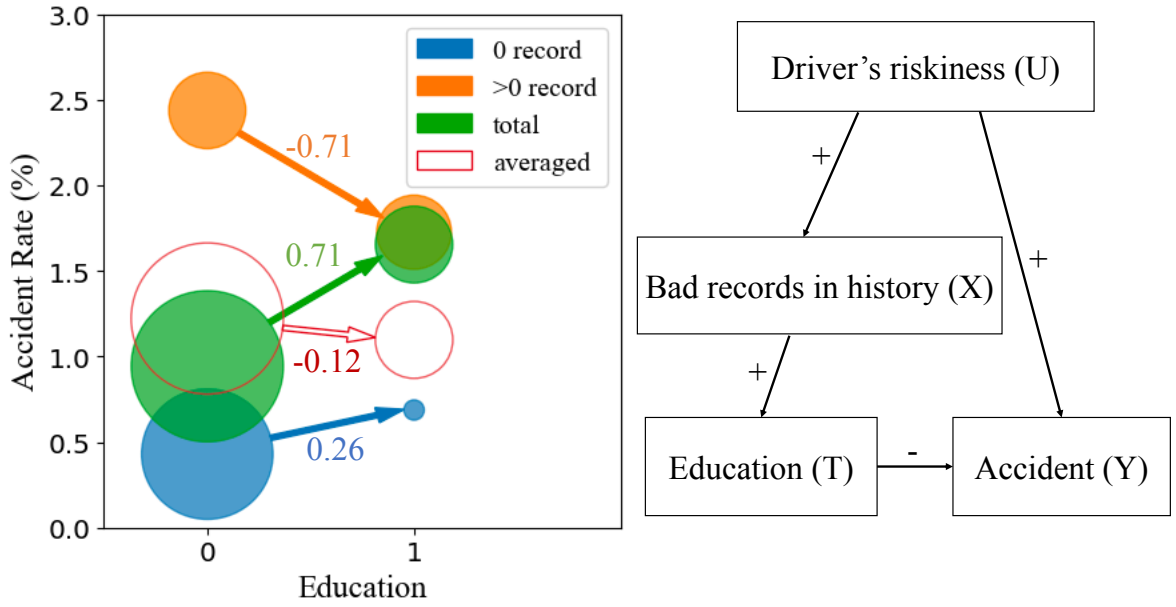
the survival of twin births.

## 4.1 Introduction

The problem of uplift modeling, also known as 'individual treatment effect estimation' in the study of causal inference, has been widely encountered in real-world applications [81, 100, 28, 29, 159, 49, 93]. Consider a particular scenario in urban transportation: when the traffic police educate over 30k drivers in a city for two years in order to raise the public awareness of traffic safety [37], they would be concerned with (i) evaluating the effectiveness of such education practice on reducing traffic accidents and (ii) selecting the best set of target drivers to receive education given limited traffic police resource. The core question of (i) and (ii) is to estimate the 'persuadability' of each individual driver, i.e. the difference in the underlying accident rate if the driver were educated as compared to not educated, and this 'persuadability' is known as 'uplift' [43]. A variety of classical models [61, 61, 49, 99, 46, 157] have been proposed to estimate uplifts and have enjoyed great popularity.

Challenge remains, however, due to the ubiquity of confounding bias in real-world applications, which has drawn increasing attention from causal inference researchers in the past few years [27, 141, 103]. The confounding bias refers to the existence of confounder(s), i.e. variable(s) that can affect both the treatment variable and the outcome variable concurrently. Figure 4.1(a) visualises the presence of confounding bias in the education-accident scenario with real statistics. Considering whether a driver has bad records in history, the figure divides the whole group of drivers (in green) into two sub-groups (in blue and orange). The arrow gradients show that the 'uplift' estimated on the whole group is surprisingly positive (0.71%) - in fact larger than the uplift estimated on any of the two sub-groups - while the 'uplift' averaged over the two sub-groups (weighted by their sizes) is negative (-0.12%). Here, the positive 'uplift' could mislead analysts that education 'causes' more accidents in general, while in fact the observed higher accident rate for the educated drivers is not due to education, but due to the higher 'riskiness' of the educated drivers themselves, as the traffic police tend to educate drivers with bad records in history. Such a paradox, named as semi-Simpson's paradox, results from the presence of confounding bias, and it leads to inaccurate estimation of uplift. Figure 4.1(b) diagrams the confounding path that explains this paradox.

The presence of the confounding bias can deteriorate uplift modeling. Classical uplift mod-




(a) The semi-Simpson's paradox. The circle area represents the group population, and the arrow gradient corresponds to the uplift estimate.

(b) The confounding path. It gives rise to a positive correlation that more than offsets the negative causal effect of T on Y.

Figure 4.1: The semi-Simpson's paradox in the education-accident scenario. The variable of 'bad records in history' appears to lie on a confounding path in estimating the causal effect of education on accident.

els often assert certain assumptions on data like unconfoundedness (i.e. the absence of confounding bias) to facilitate estimation. These models often require Randomized Control Trials (RCTs) [112] that randomly assign treatment to individuals so that the generated data be unconfounded. However, in real-world applications, experiments like RCTs can be infeasible or prohibited, and data are often collected as observables from historical records, where the unconfoundedness assumption can be easily violated. Such violation can diminish the performance of existing uplift models, and even deteriorate the validity of the evaluation metrics such as AUUC (Area Under the Uplift Curve) [103].

To bridge the gap between the RCT-based models and the often confounded real-world data, sample re-weighting becomes a crucial component in uplift modeling. Sample re-weighting modifies the distribution of training data to make the treatment variable independent of potential confounders in the original distribution and thus relaxes the unconfoundedness assumption.

The research attention of uplift modelers to sample re-weighting has increased in recent years, seeing works that incorporate inverse propensity scores in their model design, including some meta-learners [61, 89, 7] and neural network-based models [108, 109]. However, we still see space for performance improvement upon our experiments with these models on real-world datasets, as meta-learners are tied to the choice of base learners and can twist the problem definition, and neural networks rely on extensive training and strong human expertise on hyperparameter tuning. Moreover, we notice a lack of systematic understanding of the role that sample re-weighting plays in uplift modeling, in terms of its different forms of implementation and the actual gain it brings about in different real-world experimental settings.

This work focuses on sample re-weighting in uplift modeling, and aims to provide a systematic discussion on elements related to sample re-weighting, from its motivation, to its realization, to its actual gain in end-to-end real-world tasks. The main contributions of this work are as follows:

- We systematically review the assumptions either explicitly stated or implicitly made by existing uplift models, and identify the gap in applying existing uplift models to real-world applications due to the confounding bias. To further eliminate the confounding bias, we propose novel sample re-weighting approaches that organically integrate inverse propensity scoring (IPS) into the structural design of the model, and compare them with straightforward dataset sampling method implemented in four modes.

- We identify the deficiencies of existing evaluation metrics in real-world dataset due to the confounding bias, and propose two adapted metrics: Area Under the Unconfounded Uplift Curve (AUUUC) and Maximum of the True Uplift Curve (MTUC). AUUUC eliminates the confounding bias with IPS, and MTUC directly leverages the ground truth uplifts.

- We conduct extensive experiments to test our proposed sample re-weighting methods with three real-world datasets. We test the uplift models with two public datasets and further apply our proposed models to a case study of education recipient selection to reduce traffic accidents. Results show that our proposed model, Class Transformation (Random Forest with Inverse Propensity Scoring), outperforms baselines not only in uplift modeling (AUUUC increased by up to 46%) but also on subsequent tasks (further reducing monthly accident rate by 3.4% in education recipient selection).

The rest of the chapter is organized as follows. Section 4.2 discusses related work. Section 4.3 reviews uplift models and their assumptions. Section 4.4 discusses sample re-weighting methods. Section 4.5 proposes adapted evaluation metrics. Section 4.6 details the experiments with three real-world datasets. Section 4.7 concludes this chapter.

## 4.2 Related Work

The problem of uplift modeling has been encountered in various fields such as recommendation (e.g. air shipping recommendation [81], paging request strategy [100]), marketing (e.g. customer retention [28], campaigns [29], voucher distribution [159]), medicine (e.g. clinical trials [49]), and education (e.g. student retention [93]). This work introduces a new application scenario in urban transportation, studying the causal effect of education on reducing accidents.

A variety of models are proposed in the literature to estimate uplifts. To name a few, meta-learners such as S-Learner [61], T-Learner [61], X-Learner [61], R-Learner [89], and DR-Learner [129]; Class Transformation [49] and Transformed Outcome [7]; tree-based models such as Uplift Trees/Forests [99, 46, 157, 41, 78, 6, 8]; neural network-based models such as TARNet [108], DragonNet [109] and CEVAE [73]. A survey of uplift modeling is available in [43]. In the presence of confounders, which is often the case in real-world observed dataset [141, 27], the performance of existing uplift models is often diminished; and existing evaluation metrics such as AUUC are also found deficient upon violation of the unconfounded assumption [103].

Approaches have been proposed to use sample re-weighting to eliminate the confounding bias [113, 9, 43, 60] in uplift modeling. Some uplift models, such as meta-learners [61, 89, 7] or neural network-based models [108, 109], incorporate propensity scores into their model architecture. However, our experiments still suggest space for performance improvement with these models on real-world datasets, possibly due to meta-learners being tied to the choice of base learners and twisting the problem definition, and neural networks relying on extensive training and strong human expertise on hyperparameter tuning.

Recent works have witnessed the increasing popularity of sample re-weighting and deconfounding in machine learning as well. In stable learning [23], sample weights are learnt to make feature variables independent of each other to achieve uniformly small prediction errors across samples, while our work learn sample weights to make treatment variable independent

from feature variables and retain the distribution of feature variables to obtain accurate uplift estimates. Extensive work has focused on eliminating the confounding bias in domain-specific prediction tasks, such as in computer vision [48, 140], long-tail problems [119, 162], event forecasting [27], graph representation [114] and recommendation [148]. However, these works target at prediction tasks, which is fundamentally different from the problem setting of uplift modeling in our work, and thus their methods cannot be directly adopted.

## 4.3 Uplift Modeling: the Problem, the Models and Their Assumptions

In this section, we define the uplift modeling problem, summarize the most popular uplift models, and provide a systematic review of the explicit and implicit assumptions made in these models.

### 4.3.1 The Uplift Modeling Problem

We target at the binary-treatment binary-outcome uplift modeling problem. Let $i = 1, 2, ..., N$ be the indices of $N$ samples. Denote $Y_i \in \{0, 1\}$ as the outcome variable, and $T_i \in \{0, 1\}$ as the treatment variable. Let $\{X_i^{(j)} \in \mathbb{R}\}_{j=1,2,...,M}$ represent $M$ feature variables (i.e. covariates) observed for sample $i$. The uplift modeling problem aims at estimating the causal effect of $T_i$ on $Y_i$ for each sample $i$.

**Problem 2 (Uplift Modeling)** *Given a dataset*

$$\mathbb{D} = \{Y_i, T_i, \{X_i^{(j)}\}_{j=1,2,...,M}\}_{i=1,2,...,N},$$

*estimate uplift*

$$\begin{aligned} \tau_i :&= E[Y_i|do(T_i = 1)] - E[Y_i|do(T_i = 0)] \\ &= P(Y_i = 1|do(T_i = 1)) - P(Y_i = 1|do(T_i = 0)), \end{aligned} \tag{Eq. 4.1}$$

*where the do-operator represents an intervention in the variable following Pearl's Structural Causal Model [97, 96], controlling all other variables that are not affected by the variable $T_i$ causally.*

### 4.3.2 Uplift Models

Popular uplift models can be categorized into 4 groups:

(i) **Meta-learners.** Meta-learners are frameworks that convert the uplift modeling problem into a prediction task, so that classical machine learning models can be applied off the shelf. The most common meta-learners are listed below.

- S-Learner [61] estimates uplifts with a single machine learning model. Mathematically, S-Learner requires a base learner $f : (X,T) \rightarrow E[Y]$ setting $T$ as a feature variable.
- T-Learner [61] estimates uplifts with two machine learning models. Mathematically, T-Learner requires two base learners: $f_0 : X \rightarrow E[Y|T = 0]$ for treatment group and $f_1 : X \rightarrow E[Y|T = 1]$ for control group.
- X-Learner [61] extends T-Learner with two additional base learners to estimate uplifts, and possibly one additional propensity score estimator to fuse the uplift estimates.
- R-Learner [89] requires two base learners to estimate the cross-validation out-of-fold outcomes and propensity scores, followed by an additional base learner to estimate uplifts via minimizing the R-Loss. Effectively, samples with fewer supports are assigned with higher weights.

(ii) **Class transformation.** Class transformation is a special type of meta-learner that is only applicable to binary treatment and binary outcome problem. It transforms the treatment variable $T_i$ and the outcome variable $Y_i$ into a new variable. Specifically,

- Class Transformation ('ClsTrans' for short) [43]. A typical Class Transformation framework learns $f : X \rightarrow P(Z)$, where $Z_i := Y_i * T_i + (1 - Y_i) * (1 - T_i)$.
- Transformed Outcome [7]. A generalized method proposes Transformed Outcome [7] $Y_i^* := Y_i * T_i/\hat{p}_i - Y_i * (1 - T_i)/(1 - \hat{p}_i)$ which incorporates a propensity score estimator to account for unbalanced and confounded treatment assignment.

(iii) **Tree-based models (modeling uplift directly).** Tree-based uplift models are adapted from existing tree-based machine learning models to model uplifts directly. Mathematically, a tree-based model learns $f : X \rightarrow \tau$. Instead of splitting a node based on the purity of label $Y$ in a tree-based machine learning model, a tree-based uplift model splits a node to maximize

the divergence of probability distributions of $Y$ between treatment group ($T = 1$) and control group ($T = 0$). Popular tree-based models include Uplift Trees and Uplift Random Forests ('Uplift RF' for short) with various splitting criteria using Kullback-Leibler divergence (KL) [99], Euclidean Distance (ED) [99], $\chi^2$-divergence (Chi) [99], as well as Contextual Treatment Selection (CTS) [157] and delta-delta-p (DDP) [46].

(iv) **Neural Network (NN)-based models.** The nature of Neural Networks provides high flexibility in designing an uplift model and theoretically strong learning ability with its deep architecture. For example,

- Causal Effect Variational Autoencoder (CEVAE) [73] leverages Variational Autoencoder (VAE) [62] to model the unknown latent space involving confounding effect;
- DragonNet [109] takes the last layer of an NN-based propensity score estimator as input into an NN-based uplift estimator to eliminate the confounding effect.

These models serve as baselines for our analysis.

### 4.3.3 Confounding and Effect Modification

Before we look into the assumptions made in existing uplift models, we highlight and differentiate two concepts that we believe are crucial for understanding uplift modeling, and are nevertheless seldom articulated in the uplift modeling literature.

A covariate variable $X_i^{(j)}$ can affect the estimation of uplifts in two ways: as a confounder, and/or as an effect modifier [40].

**Definition 4.4 (Confounder)** *Consider an uplift modeling problem with generating variables including covariates $\{X_i^{(j)}\}_{j=1,2,\ldots,M}$, treatment variable $T_i$, and outcome variable $Y_i$. $X_i^{(j)}$ is a confounder iff $X_i^{(j)}$ affects $T_i$, $Y_i|T_i = 0$ and $Y_i|T_i = 1$ concurrently.*

**Definition 4.5 (Effect modifier)** *Consider an uplift modeling problem with generating variables including covariates $\{X_i^{(j)}\}_{j=1,2,\ldots,M}$, treatment variable $T_i$, and outcome variable $Y_i$. $X_i^{(j)}$ is an effect modifier iff $X_i^{(j)}$ affects $\tau_i := E[Y_i|do(T_i = 1)] - E[Y_i|do(T_i = 0)]$.*

Here, $Y_i|do(T_i = 0)$ and $Y_i|do(T_i = 1)$ are equivalent to the potential outcomes $Y_i(0)$ and $Y_i(1)$ in the potential outcome framework [111, 105]. The term 'affects' indicates the causal relationship between the two variables. Statistically, 'affects' implies that the two variables are dependent.

A covariate $X_i^{(j)}$ can be an effect modifier, or a confounder, or both. Both an effect modifier and a confounder concurrently impact the observed dependence between $T$ and $Y$. In an uplift modeling problem, effect modification (i.e. the causal relationship from $X_i^{(j)}$ to $\tau_i$) is the 'true' value of interest, while confounding is the 'fake' value that should be eliminated if present.

### 4.3.4 Assumptions

In the uplift modeling literature, we notice a lack of systematic review of the assumptions made by existing models, either explicitly or implicitly, and we target at enumerating these assumptions in this subsection.

We first list out four assumptions related to the identifiability [87] of a causal inference problem in general. Only upon these assumptions would a causal relationship such as an uplift be identifiable.

(a) **No interference**:

$$Y_i(t_1, t_2, ..., t_N) = Y_i(t_i),$$

where $Y_i(\cdot) := (Y_i = 1|do(T = \cdot))$ follows the notation in the potential outcome framework [111, 105]. The potential outcome for one sample is not affected by the treatment for other samples. To illustrate with the education-accident example, whether driver A receives traffic education or not does not affect driver B's accident rate.

(b) **Consistency**:

$$T_i = t \rightarrow Y_i = Y_i(t).$$

For each sample there are no different forms or versions of each treatment level, which lead to different potential outcomes. For example, the education content does not vary across different geographic districts which leads to different impacts on a driver's accident rate. (a) and (b) combined is widely known as the **Stable Unit Treatment Value Assumption (SUTVA)** [105, 87, 106].

(c) **Conditional Independence Assumption (CIA)** [63] (also known as 'no unobserved confounders' or 'ignorability'):

$$T_i \perp \{Y_i(0), Y_i(1)\}|X_i.$$

Given $X_i$, the treatment assignment $T_i$ is independent of sample $i$'s potential outcomes $\{Y_i(0), Y_i(1)\}$.

The assumption is violated when $T_i$ depends on some latent variable (other than $X$) that also affects $\{Y_i(0), Y_i(1)\}$. In other words, CIA assumes no unobserved confounders.

(d) **Positivity** (also known as 'overlapping treatment'):

$$0 < P(T_i = 1|X_i = x) < 1, \ \forall x.$$

The middle term of the inequality is also known as 'propensity score' (see Definition (4.6)). A dataset with data sparsity issue violates positivity. (c) and (d) combined is also known as **strong ignorability** [52].

Next, we list out a few assumptions that are commonly made in existing uplift models and are usually satisfied by RCTs, but are seldom articulated - either stated vaguely or left unmentioned.

(e) **Unconfoundedness**:

$$T \perp X.$$

None of covariates $X$ is a confounder. Some literature uses the term 'unconfoundedness' for assumption (c), referring to the set of latent variables being confounders, leaving (e) assumed but unmentioned. In our work, the term 'unconfoundedness' is reserved for (e), referring to the set of covariates being confounders.

(f) **Balance**:

$$P(T = 1) = P(T = 0) = 1/2.$$

Some literature [20] refer to 'balance' as treatment assignment being balanced across features, i.e., $P(T = 1|X) = P(T = 0|X) = 1/2$, which is equivalent to assuming both (e) and (f) in our terminology. Others may refer to 'balance' as class balance [91], i.e., $P(Y = 1|X) = P(Y = 0|X) = 1/2$.

(g) **Sufficient features** (i.e. no unobserved effect modifiers). The uplift of each individual $\tau_i$ can be characterized by its features $X_i$'s via a fixed function $g$:

$$\exists g, \text{ s.t. } \forall i, \ \tau_i = g(X_i).$$

(h) **Pre-treatment features**:

$$P(X_i) = P(X_i|do(T_i = 1)) = P(X_i|do(T_i = 0)).$$

Table 4.1: A summary of the assumptions made in the uplift modeling literature.

| | **Assumptions** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Method** | (a) | (b) | (c) | (d) | (e) | (f) | (g) | (h) |
| S-Learner | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| T-Learner | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| X-Learner | ✓ | ✓ | ✓ | ✓ | - | - | ✓ | ✓ |
| R-Learner | ✓ | ✓ | ✓ | ✓ | - | - | ✓ | ✓ |
| ClsTrans | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Transformed Outcome | ✓ | ✓ | ✓ | ✓ | - | - | ✓ | ✓ |
| Uplift Tree/RF | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| CEVAE | ✓ | ✓ | - | ✓ | - | - | ✓ | ✓ |
| DragonNet | ✓ | ✓ | ✓ | ✓ | - | - | ✓ | ✓ |

All features in $X$ are pre-treatment variables [149]. In other words, changing $T_i$ will not cause $X_i$ to change. Following the terminology in Pearl's framework [96], $X$ can only be confounders, neither mediators nor colliders.

With assumptions (g) and (h) we may rewrite the uplift in (Eq. 4.1) as

$$\tau_i = g(X_i) = P(Y_i = 1|do(T_i = 1), X_i) - P(Y_i = 1|do(T_i = 0), X_i).$$

The uplifts are now equivalent to the Conditional Average Treatment Effects (CATEs) [87]. Instead of estimating an uplift for each sample $i$, the uplift modeling problem now is to estimate an uplift function over all supports of $X$, which allows extrapolation over new samples.

Table 4.1 summarizes the assumptions made in each of the uplift models in Section 4.3.2. We would like to drive readers' attention to (e) the unconfoundedness assumption. Classical uplift models usually rely on Randomized Control Trials (RCTs) and assume unconfoundedness. In many real-world applications such as the education-accident scenario introduced in Section 4.1, the unconfoundedness assumption, however, is violated, diminishing the performance of these models. Although some models put efforts to relax the unconfoundedness assumptions in their model design (marked with '-' in column (e)), our experiments with these models still suggest space for performance improvement: the performance of meta-learners (X-Learner, R-Learner, Transformed Outcome) could be closely tied to the choice of base learners and twist the problem settings from binary outcome to continuous outcome, and the neural network-based models (CEVAE, DragonNet) could require extensive training and strong human expertise on hyperparameter tuning.

## 4.4 Sample Re-Weighting

To address the issue of confounding bias with real-world observed datasets, we consider applying the idea of sample re-weighting to existing uplift models to relax the unconfoundedness assumption. We first propose and analyze direct dataset sampling approach, implemented in four modes. We further propose two novel approaches that organically incorporates the idea of sample re-weighting in the model design via inverse propensity scoring. The pros and cons of different sample re-weighting methods are discussed.

### 4.4.1 Dataset Sampling (DS)

Direct dataset sampling physically modifies the datasets before fitting the uplift model. We propose four modes of implementation for analysis.

**DS1. Resampling on Strong Confounders Only (Partial)**. If one or more strong confounder(s) (Definition 4.4) are known from prior knowledge or expertise, such as the variable of 'bad records in history' in the education-accident scenario introduced in Section 4.1 and in Figure 4.1, sample re-weighting can be done 'partially' on strong confounders only.

Denote the strong confounders by $X^{(S)} := \{X^{(j)}\}_{j \in S}$. For any possible value of strong confounders $x^{(S)}$, randomly sample the treatment subgroup ($T = 1$]) and the control subgroup ($T = 0$) respectively, s.t. $p(T = 1|X^{(S)} = x^{(S)}) = p(T = 0|X^{(S)} = x^{(S)}) = 1/2$ and $p(X^{(S)} = x^{(S)})$ retains. Here, $p(\cdot)$ refers to the frequency of samples.

In practice, values for continuous confounders need to be divided into intervals before grouping; some confounder groups may encounter data sparsity issues (i.e. its treatment subgroup or control subgroup may not have enough supports for sampling) and are left as they are, and therefore the unconfoundedness assumption may not be fully relaxed; and some samples are left unused after resampling.

**DS2. Resampling**. Similar to DS1, resampling randomly sample treatment and control subgroups for each $X = x$ group, s.t. $\forall x,\ p(T = 1|X = x) = p(T = 0|X = x) = 1/2$ and $p(X = x)$ retains. Here $X$ denotes the set of all covariates. In practice, resampling may encounter issues with value grouping, data sparsity as well as partial usage of data, the same as in DS1.

**DS3. Undersampling**. Undersampling only sample the majority $T$ subgroup (i.e. the subgroup with larger sample size out of treatment and control subgroups) in each large-size $X = x$ group, keeping the minority $T$ subgroups the same, s.t. $p(T = 1|X) = p(T = 0|X) = 1/2$.

Undersampling results in a smaller sample size, which reduces the computational cost for uplift modeling. However, undersampling may encounter issues with value grouping, data sparsity as well as partial usage of data, the same as in DS1. In general, undersampling is only suitable for large-scale datasets. Moreover, as the frequency distribution of $p(X)$ cannot be retained, the calculation of the average treatment effect (ATE) on the undersampled dataset is no more meaningful.

**DS4. Oversampling**. Oversampling sample both treatment and control subgroups with replacement in each $X = x$ group, doubling the total size of each $X = x$ group to retain $p(X)$.

Our implementation of oversampling guarantees full use of all samples. However, due to larger sample size after oversampling, computational costs increases for uplift modeling. Onversampling may also encounter issues with value grouping and data sparsity, the same as in DS1.

The direct dataset sampling approach eliminates the confounding bias in observed data via physically modifying the original dataset. Obviously, their common advantage is that the sample re-weighting step is decoupled from the uplift modeling step and is thus generally applicable to any existing uplift model. However, such implementations in practice are often limited by certain issues, weakening their performance as expected. Table 4.2 summarizes the possible issues encountered by direct dataset sampling.

### 4.4.2 Inverse Propensity Scoring (IPS)

To go beyond the limitation of direct dataset sampling, we look into a more organic way to realize the idea of sample re-weighting using inverse propensity scoring (IPS) [9]. Inverse propensity scoring explicitly estimates the sample weights for the whole dataset, and incorporate that into the uplift model training step to eliminate the confounding bias. As the implementation of IPS requires model-specific design, we use Class Transformation with Random Forest (or 'ClsTrans (RF)' for short as a representative baseline uplift model for illustration. We propose two new models, one that incorporates IPS into the Class Transformation framework, and the other that incorporates IPS into the bootstrapping stage in Random Forest. The implementation of IPS is detailed below.

### Propensity score estimation

IPS first requires a propensity score estimator that assigns a propensity score to each sample in the dataset. The propensity score quantifies the dependency of treatment assignment on covariates.

**Definition 4.6 (Propensity score)** *Given* $\mathbb{D} = \{T_i, X_i\}_{i=1,2,...,N}$ *where* $X_i$ *is short for* $\{X_i^{(j)}\}_{j=1,2,...,M}$, *the propensity score is* $p_i := P(T_i = 1|X_i)$.

The propensity score estimator $\hat{f}$ of $f : X \to p$ is learnt from the original dataset, and in this work, it is set as the elastic net propensity model (i.e. logistic regression with L1 and L2 regularization), following the default implementation in the Causal ML package [17]. The estimated propensity scores $\hat{p}_i$ are then organically incorporated into the uplift model.

### Propensity score incorporation

For propensity score incorporation, we take ClsTrans (RF) as a baseline uplift model, considering RF's strength in eliminating bias via randomness, and ClsTrans's unique design in the binary-outcome problem (see experiments in Section 4.6). We propose two ways to incorporate the estimated propensity scores.

Approach (a). **Class Transformation with Inverse Propensity Scoring (Random Forest)** ('ClsTrans (RF_IPS)'). The idea is motivated from Transformed Outcome [7], defining the adapted outcome variable

$$Y_i^* := Y_i * T_i/\hat{p}_i - Y_i * (1 - T_i)/(1 - \hat{p}_i)$$

to replace the existing outcome variable, and leaving the rest of the model the same.

Approach (b). **Class Transformation (Random Forest with Inverse Propensity Scoring)** ('ClsTrans_IPS (RF)'). The training of Random Forest is based on bootstrapping that repeatedly sample training subsets from the original dataset. Hence, a simple yet effective way to incorporate propensity scores into Random Forest is to set the sample weights in boostrapping as

$$w(i) := T_i/\hat{p}_i + (1 - T_i)/(1 - \hat{p}_i)$$

.

Table 4.2: A summary of sample re-weighting methods and their issues encountered. 'N' refers to the original uplift models in Section 4.3.2 without sample re-weighting.

| Method | N | DS1 | DS2 | DS3 | DS4 | IPS |
|---|---|---|---|---|---|---|
| Modify dataset | - | ✓ | ✓ | ✓ | ✓ | - |
| Model-specific design | - | - | - | - | - | ✓ |
| Cannot retain $p(X)$ | - | - | - | ✓ | - | - |
| Require value grouping | - | ✓ | ✓ | ✓ | ✓ | - |
| Data sparsity issue | - | ✓ | ✓ | ✓ | ✓ | - |
| Partial data usage | - | ✓ | ✓ | ✓ | - | - |
| Higher computational cost | - | - | - | - | ✓ | ✓ |

In IPS, the original dataset not modified. Instead, extra information, i.e. propensity score, is fed into the uplift model. This avoids many practical issues encountered in direct dataset sampling, and theoretically produces more promising results. For example, the extrapolation ability of the propensity score estimator avoids the data sparsity issue, due to which a direct dataset sampling methood would have to group covariate values or twist $p(X)$. Admittedly, IPS requires extra modeling effort to modify uplift models to incorporate the propensity scores. The properties of IPS are also included in Table 4.2.

## 4.5 Adapted Evaluation Metrics

In this section, we analytically demonstrate that the well-adopted evaluation metrics for uplift modeling (i.e. uplift curve and AUUC) become unsound in real-world observed data when the unconfoundedness assumption is violated, and we propose two adapted metrics to relax the unconfoundedness assumption.

### 4.5.1 Area Under the Unconfounded Uplift Curve (AUUUC)

The most popular evaluation metrics for uplift modeling are uplift curve and Area Under the Uplift Curve (AUUC) [85] as defined below.

**Definition 4.7 (Uplift curve)** *Given* $\mathbb{D} = \{Y_i, T_i, \hat{\tau}_i\}_{i=1,..,N}$ *where* $\hat{\tau}_i$*'s refer to the uplift estimates by an uplift model, let* $l(\cdot)$ *re-index i's s.t.* $\hat{\tau}_{l(1)} \geq ... \geq \hat{\tau}_{l(i)} \geq ... \geq \hat{\tau}_{l(N)}$*, then the cumu-*

*lative uplift is*

$$gain(k) := \left( \frac{\sum_{l(i)=1}^{k} Y_i * T_i}{\sum_{l(i)=1}^{k} T_i} - \frac{\sum_{l(i)=1}^{k} Y_i * (1-T_i)}{\sum_{l(i)=1}^{k} (1-T_i)} \right) * k,$$

*and the uplift curve plots the function gain.*

**Definition 4.8 (AUUC)** *Given the cumulative uplift function gain, the Area Under the Uplift Curve is*

$$AUUC := \sum_{k=1}^{N} gain(k).$$

An uplift curve visualizes the uplift ranking ability of an uplift model, and AUUC quantifies it. Variants of uplift curve and AUUC are also adopted in the uplift modeling literature [29].

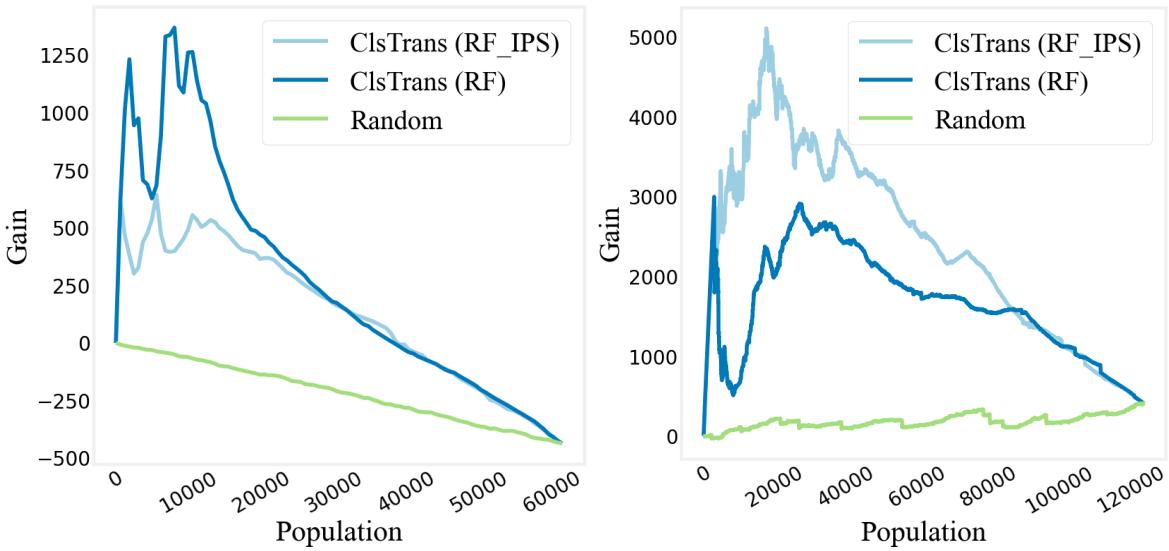
However, the usage of uplift curve and AUUC requires the unconfoundedness assumption [103]. If the unconfoundedness assumption is violated, an uplift curve may not provide meaningful visualization and AUUC may not properly rank a good model versus a bad model. Figure 4.2(a) visualizes the uplift curves in the education-accident scenario with the real-world traffic education and accident dataset EduAcc (see Section 4.6 for details), with $T$ being education and $Y$ being NON-accident. Based on human expertise, traffic education should reduce the number of accidents and thus $T$ positively affects $Y$. However, the downward random curve in Figure 4.2(a) (in green color) misleadingly implies that the correlation between $T$ and $Y$ is negative. The contradiction exemplifies that the validity of uplift curve and AUUC can be deteriorated when the unconfoundedness assumption is violated.

To bridge the gap between existing metrics (uplift curve and AUUC) and real-world scenarios (with confounding bias), we propose the following adapted metrics using IPS (introduced in Section 4.4.2) to relax the unconfoundedness assumption:

**Definition 4.9 (Unconfounded uplift curve)** *Given a dataset $\mathbb{D} = \{Y_i, T_i, \hat{\tau}_i, \hat{p}_i\}_{i=1,..,N}$ where $\hat{\tau}_i$'s refer to the uplift estimates by an uplift model, and $\hat{p}_i$'s refer to the estimated propensity scores, let $l(\cdot)$ re-index $i$'s s.t. $\hat{\tau}_{l(1)} \geq ... \geq \hat{\tau}_{l(i)} \geq ... \geq \hat{\tau}_{l(N)}$, then the unconfounded cumulative uplift is*

$$gain(k) := \left( \frac{\sum_{l(i)=1}^{k} Y_i * \frac{T_i}{\hat{p}_i}}{\sum_{l(i)=1}^{k} \frac{T_i}{\hat{p}_i}} - \frac{\sum_{l(i)=1}^{k} Y_i * \frac{1-T_i}{1-\hat{p}_i}}{\sum_{l(i)=1}^{k} \frac{1-T_i}{1-\hat{p}_i}} \right) * \sum_{l(i)=1}^{k} w(i),$$

*where $w(i) := \frac{T_i}{\hat{p}_i} + \frac{1-T_i}{1-\hat{p}_i}$ is the sample weight by inverse propensity scores. The unconfounded uplift curve plots the unconfounded cumulative uplifts as $\{(w(k), gain(k))\}_{k=1,...,N}$ and with normalization, the curve plots $\{(\frac{\sum_{i=1}^{k} w(i)}{\sum_{i=1}^{N} w(i)}, \frac{gain(k)}{gain(N)})\}_{k=1,...,N}$.*




(a) Uplift curves. (b) Unconfounded uplift curves.

Figure 4.2: Uplift curves versus unconfounded uplift curves for random sorting and selected models on EduAcc dataset.

**Definition 4.10 (AUUUC)** *Given the unconfounded cumulative uplift function gain and sample weighting function w, the Area Under the Unconfounded Uplift Curve is*

$$AUUUC := \sum_{k=1}^{N} gain(k) * w(k).$$

*With normalization,*

$$normalized\_AUUUC := \frac{AUUUC}{gain(N) * \sum_{k=1}^{N} w(k)}.$$

Figure 4.2(b) visualizes the unconfounded uplift curves of the EduAcc data. As the unconfoundedness assumption is relaxed in the new metric, random curve (in green color) now goes upward as expected. Furthermore, comparing the uplift curves for ClsTrans_IPS (RF) and ClsTrans (RF_IPS) between 4.2(a) and 4.2(b), a model with a higher AUUUC score (denoted by the curve in light blue) gets a lower AUUC score instead. Hence, AUUC is no longer a proper metric for ranking uplift models. We will adopt the proposed unconfounded uplift curve and AUUUC for experiments in Section 4.6.

### 4.5.2 Maximum of the True Uplift Curve (MTUC)

If ground truth uplifts are available - though rarely the case in real-world scenarios, as we define the metric mainly for experimental purpose - we propose a new adaptation of the uplift curve using ground truth uplifts:

**Definition 4.11 (True uplift curve)** *Given $\mathbb{D} = \{\tau_i, \hat{\tau}_i\}_{i=1,..,N}$ where $\tau_i$'s refer to the ground truth uplifts available from the dataset and $\hat{\tau}_i$'s refer to the uplift estimates by an uplift model, let $l(\cdot)$ re-index i's s.t. $\hat{\tau}_{l(1)} \geq ... \geq \hat{\tau}_{l(i)} \geq ... \geq \hat{\tau}_{l(N)}$, then the cumulative true uplift is*

$$gain(k) := \sum_{l(i)=1}^{k} \tau_i.$$

*The true uplift curve plots the function gain.*

**Definition 4.12 (MTUC)** *Given the cumulative true uplift gain, the Maximum of the True Uplift Curve is*

$$MTUC := \max_k gain(k),$$

*i.e. the peak value of the true uplift curve. With normalization,*

$$normalized\_MTUC := \frac{MTUC}{gain(N)}.$$

## 4.6 Experiments and Results

We conduct experiments with three real-world datasets. We compare the performance of existing uplift frameworks and models; we compare our proposed sample re-weighting methods with existing approaches, and evaluate the significance of sample re-weighting in both confounded and unconfounded datasets; we discuss the necessity and validity of our proposed evaluation metrics. Finally, we conduct a case study to quantitatively evaluate the role of sample re-weighting and uplift modeling in real-world optimization tasks.

### 4.6.1 Dataset Description

Our experiments are conducted with three datasets: an open-source semi-synthetic infant health dataset **ACIC**, an open-source real-world twin birth and mortality dataset **TWINS**, and a real-

world traffic education and accident dataset **EduAcc**.

### ACIC dataset

The Atlantic Causal Inference Conference (**ACIC**) Data Analysis Challenge 2017 [44] provides 8,000 semi-synthetic datasets with an original focus on the estimation of conditional average treatment effect (CATE) in the presence of targeted selection (and thus strong confounding). We select a representative dataset of 4,302 samples with high selection strength (and i.i.d. errors, low effect magnitude, low noise level, indexed by 1) for experiment. The dataset takes 8 real-world covariates $X$ directly from the Infant Health and Development Program (IHDP)[14]. The 8 features are listed in Table 4.3. The treatment variable $T$ and the outcome variable $Y$ are synthesized to simulate confounding bias. $Y$ and $T$ are generated such that $T$ can be regarded as following a Bernoulli distribution as a function of the zero-treatment potential outcomes, i.e., $P(T=1) = f(Y(0))$. Details about the data generating process can be found in [44] and data files are available at the first author's web page.[1]

For adaptation, we convert $Y_i$'s[2] into binary values to facilitate Class Transformation; we use one-hot encoding for categorical covariates $X21$ and $X24$; we convert continuous covariates $X1$ and $X43$ into ordinal values for DS1-4. The preprocessed feature matrix is in the form of (4,302 samples * 8 features). We conduct transductive learning (i.e. overfitting) on the dataset. For DS1, $X1$ and $X43$ are selected as strong confounders.

### TWINS datasets

The TWINS dataset [4] records twin births in the U.S. from 1989 to 1991. Each twin birth consists of two twins. We set treatment variable $T_i := \mathbb{I}(\text{being born the heavier twin in twin birth } i)$, and outcome variable $Y_i := \mathbb{I}(\text{NON-mortality}^2 \text{ in the first year of life})$. The dataset is available in [73] with 71,345 pairs of twins and can be downloaded from the first author's GitHub project.[3] 50 covariates are used except for dbirwt and bord.[4] Feature descriptions are available in files covar_desc.txt and covar_type.txt in the GitHub project.

[1] https://math.la.asu.edu/ prhahn/

[2] In ACIC dataset, $Y_i$'s are 0-1 flipped to conform with typical situations where uplifts are supposed to be positive in general. This facilitates the visualization of uplift curves. The same is done for TWINS datasets and EduAcc dataset.

[3] https://github.com/AMLab-Amsterdam/CEVAE/tree/master/datasets/TWINS

[4] Both are descendants of $T$ in the causal graph and should not be controlled for.

Table 4.3: ACIC dataset variable description.

| Variable | Data type | Variable description |
|---|---|---|
| $Y$ | Continuous | (Synthetic) |
| $T$ | Binary | (Synthetic) |
| $X1$ | Continuous | Mother's age |
| $X3$ | Binary | Mother's cigarettes per day |
| $X10$ | Binary | Mother's endocrine condition |
| $X14$ | Binary | Mother's nervous system condition |
| $X15$ | Binary | Mother's obstetric complications |
| $X21$ | Categorical | Mother's birth place |
| $X24$ | Categorical | Mother's race |
| $X43$ | Continuous | Mother's bilirubin |

An observational study can be simulated by randomly selecting one of the two twins for each pair of twins, while the outcomes for both twins can be regarded as potential outcomes and thus uplifts can be calculated and regarded as observed. Our experiments conduct simulation in two ways: unbiased and biased.

An unbiased dataset **TWINS-u** simulates RCTs with

$$T_i \sim Bernoulli(0.5).$$

As a result, all covariates satisfy assumptions (e) and (f). For DS1, covariate $mager8$ is selected as a 'strong confounder' as it is weakly correlated with $T$ and $Y$, though there is no significant confounding bias in data.

A biased dataset **TWINS-b** simulates biased treatment assignment with

$$T_i|X_i^-, Z_i \sim Bernoulli(\sigma(w_o * x + w_h * (Z_i/10 - 0.1))),$$

where $Z$ represents covariate gestat10, $X^-$ represents all covariates except gestat10, and $w_o \sim N(0, 0.1 * I)$ and $w_h \sim N(5, 0.1)$ are random noise. This simulation deliberately makes $gestat10$ a strong confounder, similar to [73]. For DS1, $gestat10$ is selected as a strong confounder. In both simulations, we filter top 5 covariates with the highest correlations with $T$ for DS2-4 to reduce computational cost. We conduct transductive learning (i.e. overfitting) on both datasets.

The outcomes for both twins can be regarded as potential outcomes and thus uplifts calculated as the difference of the two can be regarded as observed, and are used to calculate

Table 4.4: EduAcc dataset variable description.

| Variable | Data type | Variable description |
|---|---|---|
| $Y$ | Binary | Driver is NOT involved in accident |
| $T$ | Binary | Driver has been educated |
| $X1$ | Int | (Confidential) |
| $X2$ | Binary | (Confidential) |
| $X3$ | Int | (Confidential) |
| $X5$ | Int | (Confidential) |
| $X6$ | Int | (Confidential) |
| $X8$ | Int | (Confidential) |
| $X9$ | Binary | (Confidential) |
| $X10$ | Binary | (Confidential) |
| $X11$ | Int | Bad records in history |

MTUC.

**EduAcc dataset**

The **EduAcc** dataset is a 3-month collection of city-wide traffic education and accident data of 61,407 drivers. We set $Y_i := \mathbb{I}(\text{driver } i \text{ is NOT}^2 \text{ involved in accident in current month})$, $T_i := \mathbb{I}(\text{driver } i \text{ has been educated in the past three months})$, and select 9 driver-related features as covariates $X$. Covariate data types are summarized in Table 4.4, while the true meaning of the covariates are kept confidential. The dataset is characterised by strong confounding, as demonstrated with the semi-Simpson's paradox in Section 4.1. We conduct transductive learning (i.e. overfitting) on the dataset. For DS1, we select covariate $X11$ (bad records in history) as a strong confounder.

### 4.6.2 Experimental Settings

All code is written in Python 3.8.15 and run on a Linux (CentOS) server. The code for neural network-based models, including CEVAE and DragonNet, is run on a single Tesla P100 GPU.

Uplift frameworks are largely implemented with the Causal ML package version 0.13.0 [17], including S-classifier, T-classifier, X-regressor, and R-regressor, Uplift Tree Classifier, Uplift RF Classifier (KL), Uplift RF Classifier (ED), Uplift RF Classifier (Chi), Uplift RF Classifier (CTS), Uplift RF Classifier (DDP), CEVAE and DragonNet. In particular, CEVAE is implemented in PyTorch 1.13.1 [95], and DragonNet is implemented in TensorFlow 2.11.0

[1]. We implement Class Transformation and Transformed Outcome following the definition formulae in [43].

For base models, Logistic Regression (LR) and Random Forest (RF) are implemented with the Scikit-Learn package [98], XGBoost (XGB) is implemented in https://xgboost.readthedocs.io/, and LightGBM (LGBM) is implemented in https://lightgbm.readthedocs.io/. We selected these models as the base models for experiments, as they are the most popular machine learning models for solve prediction tasks with decent performance in practice.

### 4.6.3 Results and Discussion

#### Uplift modeling without sample re-weighting

Table 4.5 compares the AUUUC of existing uplift frameworks and base models on two confounded datasets.[5] Each number in the table denotes the AUUUC of each uplift model on a particular dataset, which is the area under the unconfounded uplift curve. On a fixed dataset, a model with a higher AUUUC indicates a better uplift ranking ability of the model.

*Finding 1. In general, Class Transformation (ClsTrans) is the best-performing framework, and Random Forest (RF) demonstrates superior results among base learners*. The superiority of RF comes from the efforts it puts in the model design to obtain unbiased estimation via randomness. *Finding 2. Surprisingly, methods that put efforts to relax the unconfoundedness assumption in their model design do not achieve satisfactory results.* X-Learners and R-Learners largely rely on the choice of base learners. CEVAE and DragonNet do not perform well in practice without extensive training and strong human expertise in hyperparameter tuning.

Figure 4.3 plots the unconfounded uplift curves of representative baseline models on ACIC dataset. A higher curve indicates a better uplift ranking ability. Therefore, ClsTrans (RF) significantly outperforms others in terms of uplift score ranking.

#### Uplift modeling with sample re-weighting

To analyze the performance gain via sample re-weighting, we select the best-performing uplift models as representatives according to results in Table 4.5, including ClsTrans, S-classifier and T-classifier with RF as base learner. We compare the four modes of implementation for direct

[5] Models are implemented with the Causal ML package [17].

Table 4.5: AUUUC of different uplift frameworks and base models on two datasets. Specifically, uplift frameworks include S-classifier [61], T-classifier [61], X-regressor [61], and R-regressor [89], Class Transformation (ClsTrans) [43], Uplift Tree / Random Forest (RF) Classifiers [99, 157, 46], CEVAE [73] and DragonNet [109]; and base learners include Logistic Regression (LR) [59], Random Forests (RF) [13], XGBoost (XGB) [19], and LightGBM (LGBM) [55]. No sample re-weighting is applied. Models are sorted by ACIC results.

| **Method** | **ACIC** | **EduAcc** |
|---|---|---|
| ClsTrans (RF) | 3.85 | 3.98 |
| ClsTrans (XGB) | 2.92 | 2.45 |
| ClsTrans (LGBM) | 2.73 | 2.25 |
| S-classifier (RF) | 2.59 | 4.38 |
| T-classifier (RF) | 2.47 | 4.39 |
| R-regressor (RF) | 1.61 | 3.72 |
| R-regressor (XGB) | 1.53 | 3.38 |
| R-regressor (LGBM) | 1.47 | 3.89 |
| Uplift RF Classifier (CTS) | 1.29 | 1.41 |
| Uplift RF Classifier (ED) | 1.22 | 1.61 |
| S-classifier (XGB) | 1.20 | 2.93 |
| Uplift RF Classifier (KL) | 1.19 | 1.65 |
| Uplift RF Classifier (Chi) | 1.18 | 1.48 |
| T-classifier (LGBM) | 1.12 | 4.08 |
| X-regressor (LGBM) | 1.06 | 3.47 |
| T-classifier (XGB) | 1.06 | 4.05 |
| S-classifier (LGBM) | 1.04 | 3.39 |
| X-regressor (XGB) | 0.99 | 3.71 |
| ClsTrans (LR) | 0.98 | 1.44 |
| X-regressor (RF) | 0.78 | 3.87 |
| Uplift RF Classifier (DDP) | 0.75 | 1.76 |
| Random | 0.65 | 0.44 |
| T-classifier (LR) | 0.60 | 1.37 |
| CEVAE | 0.59 | 1.39 |
| X-regressor (LR) | 0.55 | (0.38) |
| Uplift Tree Classifier | 0.52 | 0.59 |
| DragonNet | 0.09 | 1.84 |
| S-classifier (LR) | (0.27) | 1.72 |

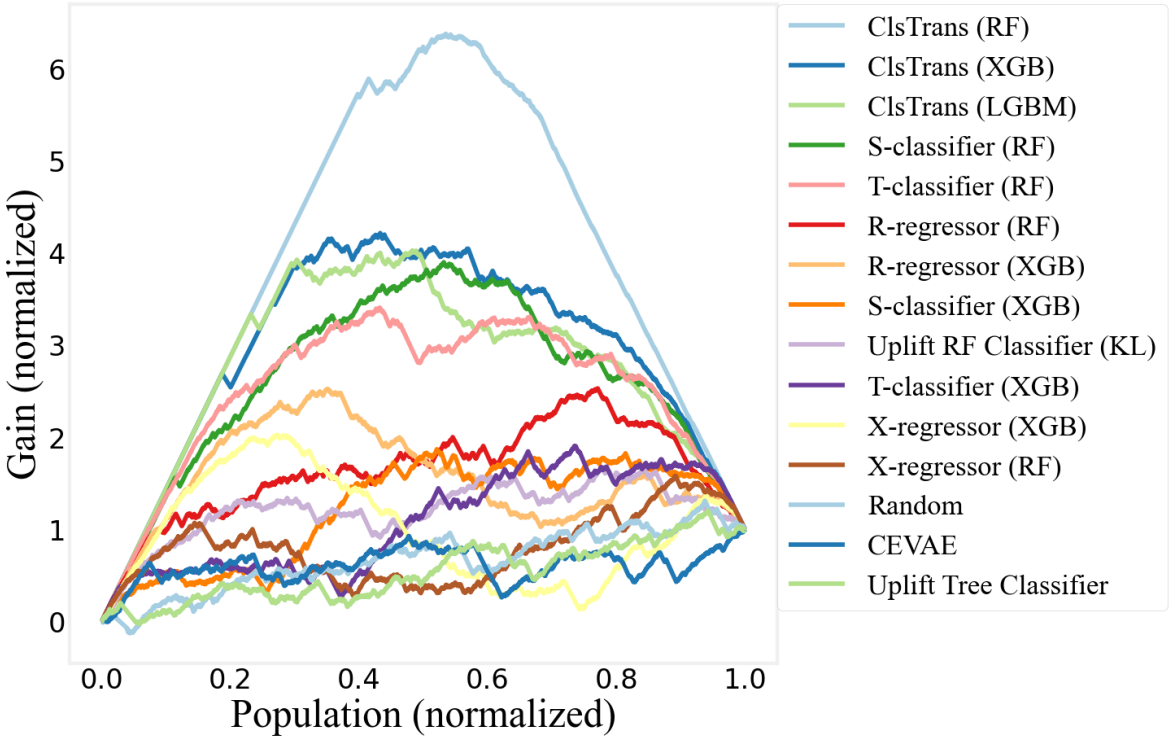



Figure 4.3: The unconfounded uplift curves of selected baseline uplift models on ACIC dataset. The end points are normalized to (1, 1).

dataset sampling and the organic integration of IPS into the model design, and their AUUUC results on two datasets are summarized in Table 4.6. *Finding 3. Direct dataset sampling may or may not improve the performance of uplift models*, considering the sacrifice they have to make to overcome practical issues upon implementations (listed in in Table 4.2). In general, oversampling and partial resampling with known strong confounders could bring performance gain via larger sample size or incorporation of human expertise, while undersampling could worsen the performance due to reduced dataset scale. *Finding 4. Incorporation of inverse propensity scoring (IPS) in base learner brings significant performance gain by up to 46%* as IPS effectively eliminates the confounding bias in dataset. On the other hand, embedding IPS in the ClsTrans framework worsens the performance, as such embedding converts the binary outcome into continuous outcome and renders the base learner as a regressor. To summarize, an organic embedding of IPS in an existing uplift model has great potential for improving the performance, but the performance relies on the model structure itself and the particular design of the embedding.

Table 4.6: AUUUC of different sample re-weighting methods on two datasets. Numbers in bold represent best results.

| Method | ACIC | EduAcc |
|---|---|---|
| **No reweighting** | | |
| S-classifier (RF) | 2.59 | 4.38 |
| T-classifier (RF) | 2.47 | 4.39 |
| ClsTrans (RF) | 3.85 | 3.98 |
| **Partial** | | |
| S-classifier (RF) | 3.05 | 3.93 |
| T-classifier (RF) | 3.12 | 3.87 |
| ClsTrans (RF) | 3.87 | 5.75 |
| **Resampling** | | |
| S-classifier (RF) | 2.54 | 3.85 |
| T-classifier (RF) | 2.41 | 3.89 |
| ClsTrans (RF) | 3.80 | 5.08 |
| **Undersampling** | | |
| S-classifier (RF) | 2.58 | 1.10 |
| T-classifier (RF) | 2.32 | 1.26 |
| ClsTrans (RF) | 3.70 | 0.73 |
| **Oversampling** | | |
| S-classifier (RF) | 2.56 | 3.93 |
| T-classifier (RF) | 2.43 | 3.99 |
| ClsTrans (RF) | 3.85 | 5.37 |
| **IPS** | | |
| ClsTrans_IPS (RF) | 3.02 | 1.22 |
| ClsTrans (RF_IPS) | **3.93** | **5.81** |

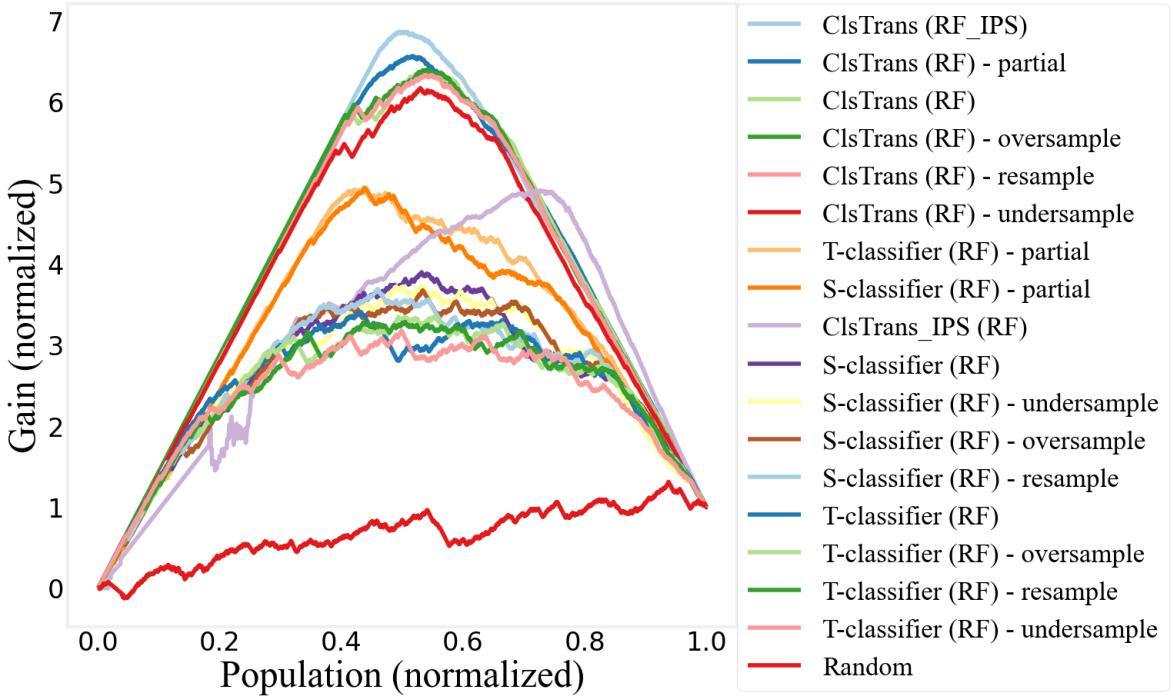



Figure 4.4: The unconfounded uplift curves of selected uplift models with different sample re-weighting methods on ACIC dataset. The end points are normalized to (1, 1).

Figure 4.4 plots the unconfounded uplift curves of different sample re-weighting methods on ACIC dataset. A higher curve indicates a better uplift ranking ability. The top curve for ClsTrans (RF_IPS), as compared to ClsTrans (RF), visualizes the performance gain via embedding IPS in the bootstrapping step of random forest.

### Evaluation metrics

Figure 4.2 visualizes the commonly used uplift curves versus our proposed unconfounded uplift curves on EduAcc dataset. *Finding 5. in the presence of strong confounding bias, adaptation to evaluation metrics such as uplift curve is necessary.* The discussion is detailed in Section 4.5.

Table 4.7 lists out the best-performing models on TWINS datasets based on our proposed metric MTUC. Inverse propensity scoring and oversampling appear to be most effective, which is in line with our findings in Table 4.6. However, MTUC can only be calculated when ground truth uplifts are available.

Table 4.7: MTUCs of different uplift models on TWINS datasets. Only top 20 (out of 132) models together with random sorting are displayed. Y-axis of TUCs is normalized upon calculation.

| **Method** | **TWINS-b** | **TWINS-u** |
|---|---|---|
| ClsTrans (RF_IPS) | 2.52 | 2.52 |
| ClsTrans (RF) - oversample | 2.52 | 2.52 |
| ClsTrans (RF) | 2.52 | 2.52 |
| T-classifier (RF) | 2.39 | 2.16 |
| T-classifier (RF) - oversample | 2.38 | 2.23 |
| S-classifier (RF) - oversample | 2.37 | 2.16 |
| R-regressor (RF) | 2.36 | 2.27 |
| R-regressor (RF) - oversample | 2.35 | 2.27 |
| S-classifier (RF) | 2.33 | 2.04 |
| ClsTrans (RF) - resample | 1.93 | 2.33 |
| ClsTrans_IPS (RF) | 1.92 | 1.79 |
| ClsTrans (RF) - partial | 1.86 | 2.36 |
| X-regressor (RF) - oversample | 1.85 | 1.98 |
| T-classifier (RF) - resample | 1.77 | 2.06 |
| S-classifier (RF) - resample | 1.76 | 1.96 |
| R-regressor (RF) - resample | 1.75 | 2.05 |
| R-regressor (RF) - partial | 1.69 | 2.04 |
| T-classifier (RF) - partial | 1.67 | 2.07 |
| S-classifier (RF) - partial | 1.65 | 2.00 |
| R-regressor (XGB) - oversample | 1.57 | 1.63 |
| Random | 1.00 | 1.00 |

Table 4.8: Average monthly accident rate reduction of the filtered educated group (as compared to the same group of drivers before education) upon reduced traffic police resource allocation based on the estimates of different uplift models. Models are sorted by the 25% column.

| **Traffic police resource reduced to** | **25%** | **50%** | **100%** |
|---|---|---|---|
| *Treatment population* | *3,158* | *6,316* | *12,633* |
| ClsTrans (RF_IPS) | -6.3% | -3.8% | -2.0% |
| ClsTrans (RF) - partial | -4.9% | -3.0% | -2.0% |
| ClsTrans_IPS (RF) | -4.3% | -2.8% | -2.0% |
| ClsTrans (RF) - oversample | -3.7% | -3.1% | -2.0% |
| ClsTrans (RF) - resample | -3.3% | -3.0% | -2.0% |
| ClsTrans (RF) | -3.2% | -3.3% | -2.0% |
| Random | -2.1% | -1.9% | -2.0% |

### 4.6.4 A Case Study: Education Recipient Selection to Reduce Traffic Accidents

We consider a few end-to-end optimization tasks on EduAcc dataset and evaluate the role that sample re-weighting and uplift modeling plays in such practical scenarios.

#### Data replay

We simulate reduced traffic police resource allocation by filtering the educated group in real data with the highest estimated uplift scores. Table 4.8 compares the average monthly accident rate reduction of the filtered educated group upon different filtering strategies (i.e. based on the estimates of different uplift models). The percentage numbers in the table are also known as 'the average treatment effects on treated' (ATTs). Education on the originally selected 12,633 recipients in real data reduces the monthly accident rate by 2.0% absolutely. If the traffic police resource allocated for education were reduced to 50% and even 25%, the uplift model ClsTrans (RF) could have reduced the monthly accident rate of the filtered educated group by up to 3.2%; and if sample re-weighting were applied, the accident rate reduction could have been further enlarged to 6.3% (for the filtered educated group based on ClsTrans (RF_IPS) estimates). *Finding 6. Sample re-weighting for uplift modeling plays a significant role in selecting a better set of education recipients to reduce traffic accidents than the existing strategy.*

**Constrained education recipient selection**

We consider a simple optimization problem to select the best set of education recipients with limited traffic police resource to minimize the overall accident rate:

$$\begin{aligned} &\min_{T_i} \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}[1-Y_i|T_i=1]*T_i+\mathbb{E}[1-Y_i|T_i=0]*(1-T_i) \\ &\text{s.t. } \sum_{i=1}^{N} T_i \leq K,\ T_i \in \{0,1\},\ \forall i \in \{1,2,...,N\}. \end{aligned} \tag{Eq. 4.2}$$

Given the uplift estimates, the problem can be converted into maximizing the average uplift:

$$\max_{T_i} \frac{1}{N}\sum_{i=1}^{N} \hat{\tau}_i * T_i \text{ s.t. } \sum_{i=1}^{N} T_i \leq K,\ T_i \in \{0,1\},\ \forall i \in \{1,2,...,N\}, \tag{Eq. 4.3}$$

which is a binary knapsack problem [33] with possibly negative profit $\hat{\tau}_i$'s and constraint parameters set to 1. Its optimal solution is

$$T_i^* = \mathbb{I}(l(i) \leq K) * \mathbb{I}(\hat{\tau}_i > 0),\ \forall i \in \{1,2,...,N\},$$

where $l(\cdot)$ re-indexes $i$'s s.t. $\hat{\tau}_{l(1)} \geq ... \geq \hat{\tau}_{l(i)} \geq ... \geq \hat{\tau}_{l(N)}$, i.e., select (up to) top-K individuals with the highest (and positive) uplift estimates for treatment. We set $K$ to 12,633 according to real data.

Table 4.9 compares new selection strategies with selected uplift models to the existing strategy in real data. To facilitate the analysis, we use the best-performing uplift model ClsTrans (RF_IPS) to estimate uplifts and evaluate the objective value for (Eq. 4.3). Results show that our proposed strategies can further reduce the overall accident rate of the experiment population, with ClsTrans (RF_IPS) achieving a further reduction of monthly accident rate by 3.4%.[6]

## 4.7 Conclusion and Future Work

This work systematically analyzes methods that relax the unconfoundedness assumption to adapt existing uplift models and evaluation metrics to real-world applications. Four modes of

[6] The accident rate reduction is only an estimate, as ground truth uplifts are not available in the dataset.

Table 4.9: Education recipient selection strategies with different uplift models compared to existing strategy in real data. The 'Delta' column indicates a further monthly accident rate reduction if the new strategy is adopted.

| **Education recipient selection strategy** | **Educated population** | **Average uplift**[6] | **Delta** |
|---|---|---|---|
| Existing strategy | 12,633 | 8.5% | 0% |
| ClsTrans (RF_IPS) | 12,633 | 11.8% | 3.4% |
| ClsTrans_IPS (RF) | 12,633 | 10.5% | 2.0% |
| ClsTrans (RF) - partial | 12,633 | 10.1% | 1.6% |
| ClsTrans (RF) | 11,741 | 9.2% | 0.7% |
| ClsTrans (RF) - oversample | 12,178 | 9.0% | 0.5% |
| ClsTrans (RF) - resample | 12,121 | 8.9% | 0.4% |

implementation for direct dataset sampling and two novel approaches for inverse propensity scoring are proposed and extensively tested with a variety of existing uplift frameworks and base learners with three real-world datasets. Results in our work point out a promising direction towards organic integration of the idea of sample re-weighting into existing model structure. Potential future work includes: new methods to be proposed for sample re-weighting, and in particular, for an organic embedding of sample re-weighting into specific model design; expansion from a binary-treatment binary-outcome problem to a more general uplift modeling problem; further relaxation of the assumptions of existing uplift models besides unconfoundedness.

# Chapter 5

# Conclusion and Future Work

## 5.1 Conclusion

In this thesis, we studied the adaptation of newly-emerging machine learning techniques to urban transport applications. We reviewed existing tasks and techniques in both urban transport understanding and urban transport actuation, based on which we identified challenges that remain for the application of machine learning in urban transport, which serve as the motivation of my PhD studies. These challenges include, in particular, the presence of non-recurrent mobility patterns, and the often encountered experimental limitations in actuation tasks. Our research aimed at tackling the challenges in particular contexts.

In the first work, we tackled the challenge of non-recurrent mobility patterns in the context of highway vehicular traffic flow prediction. We adopted spatiotemporal modeling techniques and proposed to mine the transition patterns from historical trajectories to complement non-recurrent mobility patterns. Specifically, we devised a model, Trajectory-based Graph Neural Networks (TrGNN), that incorporates trajectory transition patterns into the spatiotemporal deep learning framework based on graph to improve the accuracy of traffic flow prediction. Experiments with our approach on a real-world dataset achieves significant improvement, especially in non-recurrent scenarios.

Our second work was motivated from the challenge of an experimental limitation encountered in the task of driver recipient selection for traffic safety education to reduce traffic accidents: the absence of data from Randomized Control Trials and the presence of confounding bias in observed datasets. We identified this as a common challenge in uplift modeling across

various domains (not only in smart city applications), and we systematically studied uplift modeling approaches and proposed organic integration of sample re-weighting in existing uplift models and evaluation metrics. Extensive experiments with three real-world datasets as well as the case study on traffic safety education show significant performance gain from our proposed approach.

## 5.2 Future Work

During our studies on the application of machine learning models in urban transport, we identified a few major challenges that remain, and we proposed novel approaches in our work to tackle some of these challenges under different contexts. Based on our understanding in this direction, we could point out a few directions to consider for future work, including novel extension of our existing work, novel solutions for remaining challenges in urban transport. Moreover, during our second work on uplift modeling we realized that the challenges identified and the approaches proposed in this direction would potentially be extended to domains beyond urban transport, and we name a few alternative domains to conclude the thesis.

### 5.2.1 Extension of Existing Work

We first point out a few directions to extend our works detailed in this thesis.

**Vehicular traffic prediction with vehicle trajectories from camera sensing data.** Our first work focused on highway vehicular traffic network with GPS trajectory data. This work can be extended to similar urban transport contexts like a downtown region with camera sensing data. With vehicle snapshots taken from street cameras, vehicle identities can be recovered with high quality via the well-developed computer vision techniques, and combining vehicle identity information with spatiotemporal information, vehicle trajectories can be recovered on a large scale [120], followed by macroscopic metrics such as traffic flow, speed and occupancy. We believe our work and works in this thread can be easily extended to the new context (i.e. downtown regions in urban cities) with the new data source (i.e. camera sensing data) to further improve our understanding of urban transport.

**Organic integration of sample re-weighting into existing model architecture.** Our second work shows significant benefit of embedding inverse propensity scoring organically into the boosting stage of a random forest model. This points out a promising direction towards

an organic integration of the idea of sample re-weighting into existing model architecture. We suggest future work for new sample re-weighting approaches, or embedding of sample re-weighting in alternative models.

**Relaxation of assumptions in uplift modeling.** Our second work focused on relaxing the unconfoudnedness assumption to adapt to the reduced access to randomized control trials in uplift modeling. Unconfoundedness is only one of the assumptions that could be severely violated in the absence of randomized control trials. Further relaxation of other assumptions (as enumerated in Section 4.3.4) is considered as a promising research direction.

**Adaptation of machine learning model architecture to uplift modeling.** During our study on uplift modeling in our second work, we foresee the potential for adaptation of existing machine learning models into causal inference models, like the adaptation of Trees or Random Forests into Uplift Trees or Uplift Random Forests. Multi-gate Mixture-of-Experts (MMoE) [75] could be another good example to start with.

### 5.2.2 Applications in Urban Transport

Next, we point out a few directions to consider to solve the remaining challenges identified in this thesis (in Section 2.3).

**Non-recurrent mobility patterns.** In addition to vehicular traffic states, non-recurrent mobility patterns exist in alternative contexts such as traffic accidents, railway system breakdown, road closure, and stampede. Further analysis of these non-recurrent patterns (such as prediction of their occurrences or prediction of their impacts) would bring significant benefit for the safety, the efficiency and the sustainability of a smart city.

**Experimental limitations in actuation.** Apart from the reduced access to randomized control trials, experimental limitations also exist in alternative forms such as imperfect data quality due to the limitations of sensing techniques or alternative data collection techniques, or small representative samples due to limited experiment recipients, towards which future research is promising.

### 5.2.3 Applications Beyond Urban Transport

Some of the challenges identified in this thesis, when machine learning is applied, are generally applicable to domains beyond urban transport, and even beyond smart city applications.

Finally, we name a few directions towards these generally encountered challenges, overcoming which would bring significant benefits across domains.

**Multi-source data fusion.** In smart city applications, data often come from multiple sources and come in different forms: locations, time, graph, images, text, and so on. It naturally leads to problems of multi-source data fusion, which is also encountered in domains such as robotics, healthcare, environmental engineering, an so on [12]. Machine learning, with its well-known high capabaility of representing and transforming data, is seen as a powerful and promising tool in solving the multi-source data fusion problem in urban transport applications. Research efforts exist in this direction, but we believe that more work can be done, considering the variety of tasks and data sources in smart city applications.

**Knowledge transferability.** Most of existing work in urban transport understanding are context-specific, strongly dependent on tasks and datasets. However, one cannot reject the potential of a domain shift from a highway network to a railway network (both in the forms of lines with stations and intersections), or from one city to another (both consisting of downtown and urban areas and displaying similar POI profiles, for example). Nevertheless, very limited attempts exist to mine general knowledge from urban transport and apply it to a broader context. The problem of knowledge transferability is also encountered in vision and language tasks [18], for example. We see emerging solutions in machine learning to transfer knowledge across domains, such as transfer learning and meta-learning, and we foresee their potential for urban transport understanding and actuation as well.

# References


[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] J. Y. Ahn, E. Ko, and E. Kim. Predicting spatiotemporal traffic flow based on support vector regression and bayesian classifier. In *2015 IEEE Fifth International Conference on Big Data and Cloud Computing*, pages 125–130. IEEE, 2015.

[3] R. Akçelik and M. Besley. Sidra-2 user guide. 1984.

[4] D. Almond, K. Y. Chay, and D. S. Lee. The Costs of Low Birth Weight. *The Quarterly Journal of Economics*, 120(3):1031–1083, 2005.

[5] A. Anand, G. Ramadurai, and L. Vanajakshi. Data fusion-based traffic density estimation and prediction. *Journal of Intelligent Transportation Systems*, 18(4):367–378, 2014.

[6] S. Athey and G. Imbens. Recursive partitioning for heterogeneous causal effects: Table 1. *Proceedings of the National Academy of Sciences*, 113(27):7353–7360, 2016.

[7] S. Athey and G. W. Imbens. Machine learning methods for estimating heterogeneous causal effects. *stat*, 1050(5):1–26, 2015.

[8] S. Athey, Julie Wager. Generalized random forests. *The Annals of Statistics: An Official Journal of the Institute of Mathematical Statistics*, 47(2), 2019.

[9] P. C. Austin and E. A. Stuart. Moving towards best practice when using inverse probability of treatment weighting (iptw) using the propensity score to estimate causal treatment effects in observational studies. *Statistics in medicine*, 34(28):3661–3679, 2015.

[10] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck. Route planning in transportation networks. *Algorithm engineering: Selected results and surveys*, pages 19–80, 2016.

[11] D. Billings and J.-S. Yang. Application of the arima models to urban roadway travel time prediction-a case study. In *2006 IEEE International Conference on Systems, Man and Cybernetics*, volume 3, pages 2529–2534. IEEE, 2006.

[12] R. Bokade, A. Navato, R. Ouyang, X. Jin, C.-A. Chou, S. Ostadabbas, and A. V. Mueller. A cross-disciplinary comparison of multimodal data fusion approaches and applications: Accelerating learning through trans-disciplinary information sharing. *Expert Systems with Applications*, 165:113885, 2021.

[13] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[14] J. Brooks-Gunn, F. ruey Liaw, and P. K. Klebanov. Effects of early intervention on cognitive function of low birth weight preterm infants. *The Journal of Pediatrics*, 120(3):350–359, 1992.

[15] S. R. Chandra and H. Al-Deek. Predictions of freeway traffic speeds and volumes using vector autoregressive models. *Journal of Intelligent Transportation Systems*, 13(2):53–72, 2009.

[16] C. Chen, K. Li, S. G. Teo, X. Zou, K. Wang, J. Wang, and Z. Zeng. Gated residual recurrent graph neural networks for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 485–492, 2019.

[17] H. Chen, T. Harinen, J.-Y. Lee, M. Yung, and Z. Zhao. Causalml: Python package for causal machine learning, 2020.

[18] T. Chen, N. Garcia, M. Otani, C. Chu, Y. Nakashima, and H. Nagahara. Learning more may not be better: Knowledge transferability in vision and language tasks. *arXiv preprint arXiv:2208.10758*, 2022.

[19] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[20] X. Chen, Z. Liu, L. Yu, L. Yao, W. Zhang, Y. Dong, L. Gu, X. Zeng, Y. Tan, and J. Gu. Imbalance-aware uplift modeling for observational data. 2022.

[21] H. M. Chew. Train service disruptions on three mrt lines and bukit panjang lrt due to power fault: Smrt. *The Straits Times*, 2016.

[22] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[23] P. Cui and S. Athey. Stable learning establishes some common ground between causal inference and machine learning. *Nature Machine Intelligence*, 4(2):110–115, 2022.

[24] G. A. Davis. Accident reduction factors and causal inference in traffic safety studies: a review. *Accident Analysis & Prevention*, 32(1):95–109, 2000.

[25] G. A. Davis and N. L. Nihan. Nonparametric regression and short-term freeway traffic forecasting. *Journal of Transportation Engineering*, 117(2):178–188, 1991.

[26] M. Defferrard, X. Bresson, and P. Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29, pages 3844–3852. Curran Associates, Inc., 2016.

[27] S. Deng, H. Rangwala, and Y. Ning. Robust event forecasting with spatiotemporal confounder learning. KDD '22, page 294–304, New York, NY, USA, 2022. Association for Computing Machinery.

[28] F. Devriendt, J. Berrevoets, and W. Verbeke. Why you should stop predicting customer churn and start using uplift models. *Information Sciences*, 548:497–515, 2021.

[29] F. Devriendt, J. Van Belle, T. Guns, and W. Verbeke. Learning to rank for uplift modeling. *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[30] E. W. Dijkstra et al. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

[31] U. M. Diwekar. *Introduction to applied optimization*, volume 22. Springer Nature, 2020.

[32] S. Du, T. Li, X. Gong, Z. Yu, Y. Huang, and S.-J. Horng. A hybrid method for traffic flow forecasting using multimodal deep learning. *arXiv preprint arXiv:1803.02099*, 2018.

[33] K. Dudziński and S. Walukiewicz. Exact methods for the knapsack problem and its generalizations. *European Journal of Operational Research*, 28(1):3–21, 1987.

[34] V. K. A. Durga Mahato and A. Sinha. Multi-objective optimisation model and hybrid optimization algorithm for electric vehicle charge scheduling. *Journal of Experimental & Theoretical Artificial Intelligence*, 0(0):1–23, 2023.

[35] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, and D. Jin. Deepmove: Predicting human mobility with attentional recurrent networks. In *Proceedings of the 2018 world wide web conference*, pages 1459–1468, 2018.

[36] L. Fu, T. Wang, M. Song, Y. Zhou, and S. Gao. Electric vehicle charging scheduling control strategy for the large-scale scenario with non-cooperative game-based multi-agent reinforcement learning. *International Journal of Electrical Power & Energy Systems*, 153:109348, 2023.

[37] Z. Ganghang. The ongoing 'xian': Hangzhou traffic police carry out education activities., 2022.

[38] X. Geng, Y. Li, L. Wang, L. Zhang, Q. Yang, J. Ye, and Y. Liu. Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3656–3663, 2019.

[39] Y.-a. Geng, Q. Li, T. Lin, L. Jiang, L. Xu, D. Zheng, W. Yao, W. Lyu, and Y. Zhang. Lightnet: A dual spatiotemporal encoder network model for lightning prediction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2439–2447, 2019.

[40] S. Greenland and H. Morgenstern. Ecological bias, confounding, and effect modification. *International journal of epidemiology*, 18(1):269–274, 1989.

[41] Guelman, Guillen, Perez-Marin, and AM. Uplift random forests. *CYBERNET SYST*, 2015.

[42] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 922–929, 2019.

[43] P. Gutierrez and J.-Y. Gérardy. Causal inference and uplift modelling: A review of the literature. In *International Conference on Predictive Applications and APIs*, 2017.

[44] P. R. Hahn, V. Dorie, and J. S. Murray. Atlantic causal inference conference (acic) data analysis challenge 2017. *arXiv preprint arXiv:1905.09515*, 2019.

[45] J. D. Hamilton. *Time Series Analysis*, volume 2. Princeton University Press Princeton, NJ, 1994.

[46] B. Hansotia and B. Rukstales. Incremental value modeling. *Journal of Interactive Marketing*, 2002.

[47] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[48] J. Huang, Y. Qin, J. Qi, Q. Sun, and H. Zhang. Deconfounded visual grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 998–1006, 2022.

[49] M. Jaskowski and S. Jaroszewicz. Uplift modeling for clinical trial data. In *ICML Workshop on Clinical Data Analysis*, volume 46, pages 79–95, 2012.

[50] Y. Jia, J. Wu, and Y. Du. Traffic speed prediction using deep learning method. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1217–1222. IEEE, 2016.

[51] Z. Jiang, W. Fan, W. Liu, B. Zhu, and J. Gu. Reinforcement learning approach for coordinated passenger inflow control of urban rail transit in peak hours. *Transportation Research Part C: Emerging Technologies*, 88:1–16, 2018.

[52] M. M. Joffe and P. R. Rosenbaum. Invited commentary: propensity scores. *American journal of epidemiology*, 150(4):327–333, 1999.

[53] Y. Kamarianakis and P. Prastacos. Forecasting traffic flow conditions in an urban network: Comparison of multivariate and univariate approaches. *Transportation Research Record*, 1857(1):74–84, 2003.

[54] V. Karwa, A. B. Slavković, and E. T. Donnell. Causal inference in transportation safety studies: Comparison of potential outcomes and causal diagrams. *The Annals of Applied Statistics*, 5(2B):1428 – 1455, 2011.

[55] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

[56] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[57] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[58] L. A. Klein, M. K. Mills, D. R. Gibson, et al. Traffic detector handbook: Volume i. Technical report, Turner-Fairbank Highway Research Center, 2006.

[59] D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, and M. Klein. *Logistic regression*. Springer, 2002.

[60] K. Kuang, P. Cui, B. Li, M. Jiang, S. Yang, and F. Wang. Treatment effect estimation with data-driven variable decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[61] S. R. Künzel, J. S. Sekhon, P. J. Bickel, and B. Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the national academy of sciences*, 116(10):4156–4165, 2019.

[62] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato. Grammar variational autoencoder. In *International conference on machine learning*, pages 1945–1954. PMLR, 2017.

[63] M. Lechner. Identification and estimation of causal effects of multiple treatments under the conditional independence assumption. In *Econometric evaluation of labour market policies*, pages 43–58. Springer, 2001.

[64] G. Li, N. Cao, P. Zhu, Y. Zhang, Y. Zhang, L. Li, Q. Li, and Y. Zhang. Towards smart transportation system: A case study on the rebalancing problem of bike sharing system based on reinforcement learning. *Journal of Organizational and End User Computing (JOEUC)*, 33(3):35–49, 2021.

[65] M. Li, P. Tong, M. Li, Z. Jin, J. Huang, and X.-S. Hua. Traffic flow prediction with vehicle trajectories. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 294–302, 2021.

[66] Y. Li, K. Fu, Z. Wang, C. Shahabi, J. Ye, and Y. Liu. Multi-task representation learning for travel time estimation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1695–1704, 2018.

[67] Y. Li, R. Yu, C. Shahabi, and Y. Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations (ICLR)*, 2018.

[68] Y. Li, Z. Zhu, D. Kong, M. Xu, and Y. Zhao. Learning heterogeneous spatial-temporal representation for bike-sharing demand prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1004–1011, 2019.

[69] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 2021.

[70] B. Liao, J. Zhang, C. Wu, D. McIlwraith, T. Chen, S. Yang, Y. Guo, and F. Wu. Deep sequence learning with auxiliary information for traffic prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 537–546, 2018.

[71] Z. Liu, Z. Li, M. Li, W. Xing, and D. Lu. Mining road network correlation for traffic estimation via compressive sensing. *IEEE Transactions on Intelligent Transportation Systems*, 17(7):1880–1893, 2016.

[72] Z. Liu, Z. Li, K. Wu, and M. Li. Urban traffic prediction from mobility data using deep learning. *IEEE Network*, 32(4):40–46, 2018.

[73] C. Louizos, U. Shalit, J. Mooij, D. Sontag, R. Zemel, and M. Welling. Causal effect inference with deep latent-variable models. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6449–6459, Red Hook, NY, USA, 2017. Curran Associates Inc.

[74] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang. Traffic flow prediction with big data: a deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):865–873, 2014.

[75] J. Ma, Z. Zhao, X. Yi, J. Chen, L. Hong, and E. H. Chi. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1930–1939, 2018.

[76] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang. Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction. *Sensors*, 17(4):818, 2017.

[77] L. R. Medsker and L. Jain. Recurrent neural networks. *Design and Applications*, 5:64–67, 2001.

[78] Micha, Sotys, Szymon, Jaroszewicz, Piotr, and Rzepakowski. Ensemble methods for uplift modeling. *Data Mining and Knowledge Discovery*, 29(6):1531–1559, 2014.

[79] W. Min and L. Wynter. Real-time road traffic prediction with spatio-temporal correlations. *Transportation Research Part C: Emerging Technologies*, 19(4):606–616, 2011.

[80] X. Mo, M. Li, and M. Li. Predicting abnormal events in urban rail transit systems with multivariate point process. In J. Gama, T. Li, Y. Yu, E. Chen, Y. Zheng, and F. Teng, editors, *Advances in Knowledge Discovery and Data Mining*, pages 41–53, Cham, 2022. Springer International Publishing.

[81] A. Mondal, A. Majumder, and V. Chaoji. Aspire: Air shipping recommendation for e-commerce products via causal inference framework. In *Proceedings of the 28th ACM*

*SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3584–3592, 2022.

[82] J. Mula, D. Peidro, M. Díaz-Madroñero, and E. Vicens. Mathematical programming models for supply chain production and transport planning. *European Journal of Operational Research*, 204(3):377–390, 2010.

[83] A. M. Nagy and V. Simon. Survey on traffic prediction in smart cities. *Pervasive and Mobile Computing*, 50:148–163, 2018.

[84] M. Nama, A. Nath, N. Bechra, J. Bhatia, S. Tanwar, M. Chaturvedi, and B. Sadoun. Machine learning-based traffic scheduling techniques for intelligent transportation system: Opportunities and challenges. *International Journal of Communication Systems*, 34(9):e4814, 2021.

[85] H. Nassif, F. Kuusisto, E. S. Burnside, and J. W. Shavlik. Uplift modeling with roc: An srl case study. In *ILP (late breaking papers)*, pages 40–45. Citeseer, 2013.

[86] M. Nazari, A. Oroojlooy, L. Snyder, and M. Takac. Reinforcement learning for solving the vehicle routing problem. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[87] B. Neal. Introduction to causal inference, 2020.

[88] P. Newson and J. Krumm. Hidden markov map matching through noise and sparseness. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 336–343, 2009.

[89] X. Nie and S. Wager. Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, 108(2):299–319, 2021.

[90] X. Niu, Y. Zhu, and X. Zhang. Deepsense: A novel learning mechanism for traffic prediction with taxi gps traces. In *2014 IEEE global communications conference*, pages 2745–2750. IEEE, 2014.

[91] O. Nyberg, T. Kuśmierczyk, and A. Klami. Uplift modeling with high class imbalance. In *Asian Conference on Machine Learning*, pages 315–330. PMLR, 2021.

[92] M. Okawa, T. Iwata, T. Kurashima, Y. Tanaka, H. Toda, and N. Ueda. Deep mixture point processes: Spatio-temporal event prediction with rich contextual information. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 373–383, 2019.

[93] D. Olaya, J. Vásquez, S. Maldonado, J. Miranda, and W. Verbeke. Uplift modeling for preventing student dropout in higher education. *Decision Support Systems*, 134:113320, 2020.

[94] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang. Urban traffic prediction from spatio-temporal data using deep meta learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1720–1730, 2019.

[95] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8026–8037, 2019.

[96] J. Pearl. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146, 2009.

[97] J. Pearl and D. Mackenzie. *The book of why: the new science of cause and effect*. Basic books, 2018.

[98] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[99] Piotr, RzepakowskiSzymon, and Jaroszewicz. Decision trees for uplift modeling with single and multiple treatments. *Knowledge & Information Systems*, 2012.

[100] X. Qian, Y. Xu, F. Lv, S. Zhang, Z. Jiang, Q. Liu, X. Zeng, T.-S. Chua, and F. Wu. Intelligent request strategy design in recommender system. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, page 3772–3782, New York, NY, USA, 2022. Association for Computing Machinery.

[101] J. Raj, H. Bahuleyan, and L. D. Vanajakshi. Application of data mining techniques for traffic density estimation and prediction. *Transportation Research Procedia*, 17:321–330, 2016.

[102] C. Ren, L. An, Z. Gu, Y. Wang, and Y. Gao. Rebalancing the car-sharing system with reinforcement learning. *World Wide Web*, 23:2491–2511, 2020.

[103] C. Renaudin and M. Martin. About evaluation metrics for contextual uplift modeling. *arXiv preprint arXiv:2107.00537*, 2021.

[104] S. Ruan, C. Long, J. Bao, C. Li, Z. Yu, R. Li, Y. Liang, T. He, and Y. Zheng. Learning to generate maps from trajectories. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01):890–897, Apr. 2020.

[105] D. B. Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688, 1974.

[106] D. B. Rubin. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

[107] S. Seabold and J. Perktold. Statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

[108] U. Shalit, F. D. Johansson, and D. Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *International Conference on Machine Learning*, pages 3076–3085. PMLR, 2017.

[109] C. Shi, D. Blei, and V. Veitch. Adapting neural networks for the estimation of treatment effects. *Advances in neural information processing systems*, 32, 2019.

[110] A. Skabardonis, P. Varaiya, and K. F. Petty. Measuring recurrent and nonrecurrent traffic congestion. *Transportation Research Record*, 1856(1):118–124, 2003.

[111] J. Splawa-Neyman, D. M. Dabrowska, and T. Speed. On the application of probability theory to agricultural experiments. essay on principles. section 9. *Statistical Science*, pages 465–472, 1990.

[112] K. Stanley. Design of randomized controlled trials. *Circulation*, 115(9):1164–1169, 2007.

[113] E. A. Stuart. Matching methods for causal inference: A review and a look forward. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 25(1):1, 2010.

[114] Y. Sui, X. Wang, J. Wu, M. Lin, X. He, and T.-S. Chua. Causal attention for interpretable and generalizable graph classification. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1696–1705, 2022.

[115] J. Sun and J. Sun. A dynamic bayesian network model for real-time crash prediction using traffic speed conditions data. *Transportation Research Part C: Emerging Technologies*, 54:176–186, 2015.

[116] S. Sun, C. Zhang, and G. Yu. A bayesian network approach to traffic flow forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):124–132, 2006.

[117] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, 2014.

[118] C. Tan, L. Liu, H. Wu, Y. Cao, and K. Tang. Fuzing license plate recognition data and vehicle trajectory data for lane-based queue length estimation at signalized intersections. *Journal of Intelligent Transportation Systems*, pages 1–18, 2020.

[119] K. Tang, J. Huang, and H. Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. *Advances in Neural Information Processing Systems*, 33:1513–1524, 2020.

[120] P. Tong, M. Li, M. Li, J. Huang, and X. Hua. Large-scale vehicle trajectory reconstruction with camera sensing network. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, MobiCom '21, page 188–200, New York, NY, USA, 2021. Association for Computing Machinery.

[121] L. Vanajakshi and L. R. Rilett. A comparison of the performance of aatificial neural networks and support vector machines for the prediction of traffic speed. In *IEEE Intelligent Vehicles Symposium*, pages 194–199. IEEE, 2004.

[122] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30:5998–6008, 2017.

[123] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[124] E. Walraven, M. T. Spaan, and B. Bakker. Traffic flow optimization: A reinforcement learning approach. *Engineering Applications of Artificial Intelligence*, 52:203–212, 2016.

[125] B. Wang, Z. Yan, J. Lu, G. Zhang, and T. Li. Road traffic flow prediction using deep transfer learning. In *Data Science and Knowledge Engineering for Sensing Decision Support: Proceedings of the 13th International FLINS Conference (FLINS 2018)*, volume 11, pages 331–338. World Scientific, 2018.

[126] L. Wang, X. Geng, X. Ma, F. Liu, and Q. Yang. Cross-city transfer learning for deep spatio-temporal prediction. *arXiv preprint arXiv:1802.00386*, 2018.

[127] M. Wang, B. Lai, Z. Jin, Y. Lin, X. Gong, J. Huang, and X. Hua. Dynamic spatio-temporal graph-based cnns for traffic prediction. *arXiv preprint arXiv:1812.02019*, 2018.

[128] S. Wang, J. Cao, and P. S. Yu. Deep learning for spatio-temporal data mining: A survey. *arXiv preprint arXiv:1906.04928*, 2019.

[129] X. Wang, R. Zhang, Y. Sun, and J. Qi. Doubly robust joint learning for recommendation on data missing not at random. In *International Conference on Machine Learning*, pages 6638–6647. PMLR, 2019.

[130] Y. Wang, D. Zhang, L. Hu, Y. Yang, and L. H. Lee. A data-driven and optimal bus scheduling model with time-dependent traffic and demand. *IEEE Transactions on Intelligent Transportation Systems*, 18(9):2443–2452, 2017.

[131] H. Wei, G. Zheng, H. Yao, and Z. Li. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *Proceedings of the 24th ACM SIGKDD International*

*Conference on Knowledge Discovery and Data Mining*, KDD '18, page 2496–2505, New York, NY, USA, 2018. Association for Computing Machinery.

[132] B. M. Williams and L. A. Hoel. Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results. *Journal of Transportation Engineering*, 129(6):664–672, 2003.

[133] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1907–1913, 2019.

[134] D. Xia, H. Li, B. Wang, Y. Li, and Z. Zhang. A map reduce-based nearest neighbor approach for big-data-driven traffic flow prediction. *IEEE access*, 4:2920–2934, 2016.

[135] P. Xie, T. Li, J. Liu, S. Du, X. Yang, and J. Zhang. Urban flow prediction from spatiotemporal data using machine learning: A survey. *Information Fusion*, 2020.

[136] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015.

[137] Z. Xu, Z. Li, Q. Guan, D. Zhang, Q. Li, J. Nan, C. Liu, W. Bian, and J. Ye. Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '18, page 905–913, New York, NY, USA, 2018. Association for Computing Machinery.

[138] S. Yang, L. Ning, X. Cai, and M. Liu. Dynamic spatiotemporal causality analysis for network traffic flow based on transfer entropy and sliding window approach. *Journal of Advanced Transportation*, 2021:1–17, 2021.

[139] S. Yang, B. Yang, Z. Zeng, and Z. Kang. Causal inference multi-agent reinforcement learning for traffic signal control. *Information Fusion*, 94:243–256, 2023.

[140] X. Yang, H. Zhang, and J. Cai. Deconfounded image captioning: A causal retrospect. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[141] D. Yao, C. Gong, L. Zhang, S. Chen, and J. Bi. Causalmta: Eliminating the user confounding bias for causal multi-touch attribution. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, page 4342–4352, New York, NY, USA, 2022. Association for Computing Machinery.

[142] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li. Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5668–5675, 2019.

[143] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and L. Zhenhui. Deep multi-view spatial-temporal network for taxi demand prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[144] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, and B. Yin. A comprehensive survey on traffic prediction. *arXiv preprint arXiv:2004.08555*, 2020.

[145] B. Yu, H. Yin, and Z. Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 3634–3640. International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[146] R. Yu, Y. Li, C. Shahabi, U. Demiryurek, and Y. Liu. Deep learning: A generic approach for extreme condition traffic forecasting. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 777–785. SIAM, 2017.

[147] Z. Yu, S. Liang, L. Wei, Z. Jin, J. Huang, D. Cai, X. He, and X.-S. Hua. Macar: Urban traffic light control via active multi-agent communication and action rectification. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 2491–2497, 2021.

[148] R. Zhan, C. Pei, Q. Su, J. Wen, X. Wang, G. Mu, D. Zheng, P. Jiang, and K. Gai. Deconfounding duration bias in watch-time prediction for video recommendation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4472–4481, 2022.

[149] A. Zhang, J. Gao, Y. Li, S. Li, Z. Chu, and L. Yao. A survey on causal inference. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2021.

[150] J. Zhang, Y. Zheng, and D. Qi. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[151] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi. Dnn-based prediction model for spatio-temporal data. In *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 1–4, 2016.

[152] L. Zhang, T. Hu, Y. Min, G. Wu, J. Zhang, P. Feng, P. Gong, and J. Ye. A taxi order dispatch model based on combinatorial optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 2151–2159, New York, NY, USA, 2017. Association for Computing Machinery.

[153] Q. Zhang, J. Chang, G. Meng, S. Xiang, and C. Pan. Spatio-temporal graph structure learning for traffic forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01):1177–1185, Apr. 2020.

[154] X. Zhang, L. Xie, Z. Wang, and J. Zhou. Boosted trajectory calibration for traffic state estimation. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 866–875. IEEE, 2019.

[155] K. Zhao, J. Hua, L. Yan, Q. Zhang, H. Xu, and C. Yang. A unified framework for marketing budget allocation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1820–1830, 2019.

[156] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[157] Y. Zhao, X. Fang, and D. Simchi-Levi. Uplift modeling with multiple treatments and general response types. 2017.

[158] C. Zheng, X. Fan, C. Wang, and J. Qi. Gman: A graph multi-attention network for traffic prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01):1234–1241, Apr. 2020.

[159] K. Zhong, F. Xiao, Y. Ren, Y. Liang, W. Yao, X. Yang, and L. Cen. Descn: Deep entire space cross networks for individual treatment effect estimation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, page 4612–4620, New York, NY, USA, 2022. Association for Computing Machinery.

[160] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun. Graph neural networks: A review of methods and applications. *AI open*, 1:57–81, 2020.

[161] P. Zhou, Y. Zheng, and M. Li. How long to wait? predicting bus arrival time with mobile phone based participatory sensing. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, pages 379–392, 2012.

[162] B. Zhu, Y. Niu, X.-S. Hua, and H. Zhang. Cross-domain empirical risk minimization for unbiased long-tailed classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3589–3597, 2022.

# Appendix A

# Author's Publications

(i) Papers published:

- **Mingqian Li**, Panrong Tong, Mo Li, Zhongming Jin, Jianqiang Huang, and Xian-Sheng Hua. 2021. Traffic Flow Prediction with Vehicle Trajectories. Proceedings of the AAAI Conference on Artificial Intelligence, 35(1), 294-302. https://doi.org/10.1609/aaai.v35i1.16104
- Panrong Tong, **Mingqian Li**, Mo Li, Jianqiang Huang, and Xiansheng Hua. 2021. Large-scale vehicle trajectory reconstruction with camera sensing network. In Proceedings of the 27th Annual International Conference on Mobile Computing and Networking (MobiCom '21). Association for Computing Machinery, New York, NY, USA, 188–200. https://doi.org/10.1145/3447993.3448617
- Xiaoyun Mo, **Mingqian Li**, Mo Li. 2022. Predicting Abnormal Events in Urban Rail Transit Systems with Multivariate Point Process. In: Gama, J., Li, T., Yu, Y., Chen, E., Zheng, Y., Teng, F. (eds) Advances in Knowledge Discovery and Data Mining. PAKDD 2022. Lecture Notes in Computer Science(), vol 13280. Springer, Cham. https://doi.org/10.1007/978-3-031-05933-9_4

(ii) Papers in submission:

- **Mingqian Li**, Mo Li, Panrong Tong, Zhongming Jin, and Jieping Ye. "Towards Sample Re-Weighting in Uplift Modeling".

(iii) Patents:

- **Mingqian Li**, Panrong Tong, Mo Li, Zhongming Jin, Jianqiang Huang, and Xian-Sheng Hua, "交通流量的预测方法、装置及系统," (Patent filed in China, not yet granted.)