A DATA-DRIVEN OPTIMIZATION METHOD FOR TAXI DISPATCHING PROBLEM

Narges Rezaei

A thesis
in
The Department
of
Quality System Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Applied Science
Concordia University
Montréal, Québec, Canada

July 2020

Concordia University

School of Graduate Studies

This is to certify that the thesis prepared

By: **Narges Rezaei**

Entitled: **A Data-Driven Optimization Method for Taxi Dispatching Problem**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining commitee:

Dr. Jun Yan ______________________ Chair

Dr. Mingyuan Chen (MIAE) ______________________ Examiner

______________________ Examiner

Dr. Anjali Awasthi ______________________ Examiner

Dr. Chun Wang ______________________ Supervisor

Approved ______________________

Dr. Mohammad Mannan Graduate Program Director

__________ 20 ______ ______________________

Dr. Amir Asif, Dean

Faculty of Engineering and Computer Science

Abstract

A Data-Driven Optimization Method for Taxi Dispatching Problem

Narges Rezaei

Taxi service has become one of the most important means of transportation in the world. Optimization of the taxi service can significantly reduce transportation costs, idle driving times, waiting times, and increase service quality. However, optimization of the taxi service due to its specific characteristics is a cumbersome task. In this research, we studied the taxi dispatching problem and proposed a mathematical programming machine learning-based approach to optimize the network. We presented a data-driven optimization methodology by combining machine learning techniques, that incorporate historical time-series data to forecast future demand, and mathematical programming. Specifically, Support Vector Regression and K-Nearest Neighbor are adopted to learn the passenger demand patterns based on time-series data. Then a MIP model is built to minimize total idle driving distance concerning balancing the supply-demand ratio in different regions. Moreover, we aimed at balancing supply according to the demand in different regions (nodes) of a city in order to increase service efficiency and to minimize the total ideal driving distance. We proposed a method that utilizes historical GPS data to build demand models and applies prediction technologies to determine optimal locations for vacant taxis considering anticipated future demand. From a system-level perspective, we compute optimal dispatch solutions for reaching a globally balanced supply-demand ratio with the least associated cruising distance under practical constraints. We implemented our approach to a real-world case study from New York City to demonstrate its efficiency and effectiveness.

Keywords : Data-Driven Optimization; Machine Learning; Mathematical Programming; Taxi Dispatching Problem

Acknowledgments

Foremost, I would like to express my sincere gratitude to my advisor Prof. Chun Wang for the continuous support of my Master study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor during my study for Master degree.
My sincere thanks also goes to Vasco Kollokian, Product Manager, AI & Data Science Solutions., at Tecsys Inc. for offering me the internship opportunity and leading me to work on diverse exciting projects.
I thank my fellow labmates at Concordia University, specially Xiaoming Li, for the discussions, helps and guidance to enlighten me during research in the last few years.
Last but not the least, I would like to thank my family: my parents, for supporting me spiritually throughout my life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Taxis are one of the most efficient transportation modes that deliver convenient services for passengers. Taxi service launched in the late 1990s and has become one of the crucial travel modes. Every taxi company must explore solutions to reduce costs while providing quality services to customers. The key to customer satisfaction is to optimize the demand-supply metric of drivers and passengers, in other words, to reduce the number of empty-loaded taxis. It is also of great importance to balance the supply-demand ratio in different regions to serve customers equally and enhance customer satisfaction.

Taking advantage of the recent developments of GPS technology, a large number of taxi companies and researchers aimed at creating more efficient vehicle dispatch systems in the last decades. Using recent information, advanced systems can provide strategy supports such as practical taxi dispatching, which can remarkably increase the efficiency of the taxi services.

There are roughly 200 million taxi rides in New York City each year. The exploitation of taxi supply and demand could significantly increase the efficiency of the city's taxi system. In New York City, taxi usage frequency is much higher than any other city in the US. Instead of providing an efficient booking system to customers, New York taxi drivers pick up passengers on the street. Therefore, predicting taxi ridership could result in valuable managerial insights to city planners. Such information could help managers in answering questions regarding locating the cabs, estimating the number of required cabs, and prediction of ridership variations. For this aim, we suggest a data-driven optimization framework to solve the spatio-temporal time-series taxi dispatch problem. We focus on assigning vacant taxis in an hourly time window to different regions of the city area using machine learning and optimization techniques.

1.1 Taxi Dispatching Process

In the Taxi Dispatching Process, taxis position at different locations, and they receive random passenger requests. The goal of the Taxi Dispatch Problem (TDP) is to schedule vacant taxis towards predicted passengers both spatially and temporally with the least total idle driving distance considering service quality.

In this problem, there is a central dispatch system that receives data, including the taxi's GPS location, with a timestamp periodically. The data is then processed at the dispatch center to predict the spatio-temporal patterns of passengers' demand. Based on the predictions, the dispatch center determines a dispatch solution and sends decisions to vacant taxis with dispatched districts to go.

Besides balancing supply and demand, another consideration in the taxi dispatch problem is minimizing the total idle driving distance of all taxis. A dispatch algorithm that introduces idle distance in order to serve passenger demands can increase the total profits of the taxi network in the long run.

1.2 An Optimization Framework to Taxi Dispatching Problem

One of the most challenging tasks in the taxi dispatching problem is the prediction of the future taxi demand, which plays a prominent role in the efficiency of taxi services and is one of the most advanced topics in the field.

In this study, we present a framework to allocate the available fleet of taxis to different regions of a given city, using spatio-temporal demand forecasting models and optimization techniques. The goals of our research are to decrease the taxis idle driving distance and to balance the supply-demand ratio for equivalent service quality in different regions. We use a heuristic estimation of idle driving distance to describe anticipated future costs associated with meeting customer demands using longitude and latitude of the center in each region, calculating Manhattan norm. Our proposed approach includes machine learning models to predict passenger demand, in combination with an optimization problem. We offer an adaptive system that forecasts taxi demand at a specific location within hourly intervals. We use that information to assign the taxis based on these predictions, which offer decision support for the taxi drivers and strategy support for taxi companies.

The objective of our optimization model is to allocate vacant taxis toward predicted passengers, both spatially and temporally, with the minimum total idle driving distance respecting service quality. Besides, to provide a simple structure for the problem, we consider our dispatch solution for a one hour time interval. In order to calculate the passenger demand, we utilize machine learning techniques and use that information as input parameters in our optimization framework. In the results and discussion section, we evaluate the performance of our proposed framework using the taxi data, January 2018, from New York City.

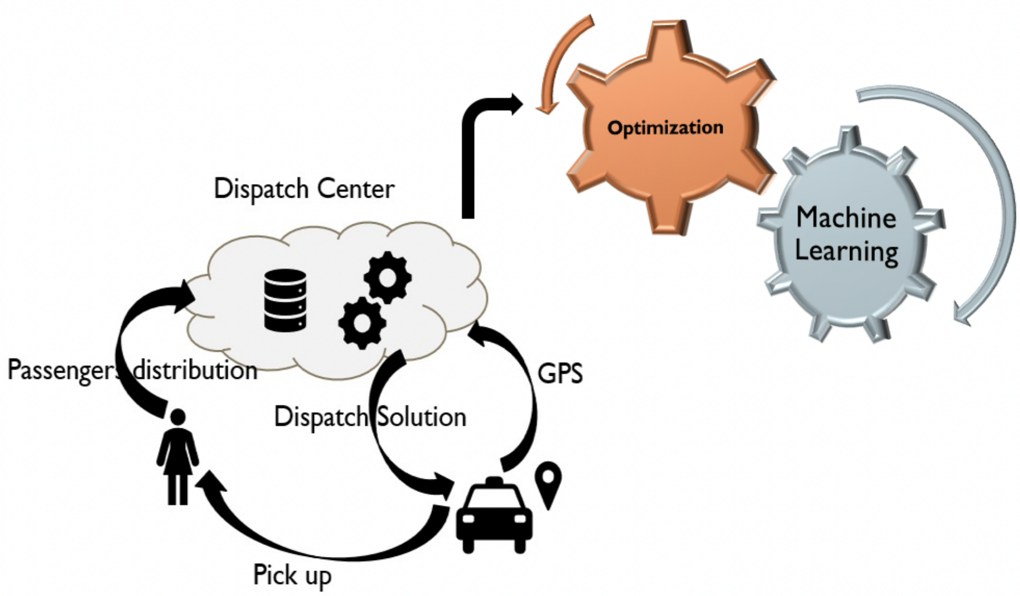

Figure 1: Taxi Dispatching Process; A combination of machine learning time series forecasting and MIP optimization.

1.3 Challenges for Data-Driven Taxi Dispatch Systems

The ultimate goal of a modern transportation system is to fulfill customers' demand while minimizing operational costs while minimizing greenhouse gas emissions. In the context of urban environments, on-demand mobility, including taxicabs and other ride-sharing services, has gained popularity in recent years due to the rapidly rising expenses of car ownership in cities. However, the operation of on-demand mobility services with limited service resources is far from optimal. Although existing methods for mobility-on-demand service of autonomous vehicles consider system-level optimality [38, 48], such systems still ignore many factors. For instance, there is no known framework to incorporate historical and real-time sensing data to improve dynamic dispatching performance. Besides, there is no efficient methodology to deal with demand uncertainties.

1.3.1 Uncertainty in Demand

Traditional taxi networks in metropolitan areas heavily rely on taxi drivers' experience to look for passengers on streets to maximize personal profit. However, such self-interested, uncoordinated behaviors of drivers usually result in a spatial-temporal mismatch between taxi supply and passenger demand. Greedy algorithms are widely employed by large taxi or ride-sharing service companies to optimize taxi networks. Such algorithms aim at finding the nearest vacant taxi to pick up a passenger or implement a first-come-first-served strategy to optimize the network. Considering a transportation system such as a taxi dispatch system or an on-demand ride-sharing system (e.g., uber, lyft, and Sidecar), the current applied service usually assigns the nearest driver to the demand point in which the driver can reach the customer in the shortest time once a request appears in the system. However, aiming to minimize each individual's waiting time does not guarantee the optimality of the total profit and efficiency of the provided service. This is due to the fact that passengers at over-supplied regions have shorter average waiting times than those at under-supplied regions, and the service may lose the customer in the under-supplied regions. Meanwhile, without a system-level regulator, drivers tend to stay within areas with the highest possibility of potential customers and traverse on streets hoping to pick up the next passenger in a short idle distance or idle time based on their own experience. Before a request enters the the system, drivers, do not have information about the demand; hence, there will be an extra idle driving distance, energy consumption, and unnecessary congestion or occupation of the roads. To the best of our knowledge, there have not been previous works that consider this type of real-time resource allocation problem from a system-optimal perspective, with the demand prediction based on either existing data or streaming data. Defining measures to evaluate service quality, considering available information, is essential for improving the performance of the system.

1.3.2 Dynamic Demand

Incorporating historical data and real-time information is vital for resource allocation for smart cities from a system-level optimality perspective due to the fact that the resources are limited. Ride-sharing or taxi services are more flexible compared to transportation systems such as subway, bus, and trains because they do not need a repeated schedule every day. In other words, in ride-sharing or taxi services, dispatch decisions should be made in real-time. However, efficient coordination of taxi networks based on the current system state is a challenging task in large scales. The choice of the most promising algorithm to model and predict a particular phenomenon is one of the most prominent activities of the temporal data forecasting. Forecasting (or prediction), similarly to other data mining tasks, uses empirical evidence to select the most suitable model for a problem at hand. It is difficult to find an accurate demand model based on a large volume of data for many applications, building an accurate model that includes appropriate information is a cumbersome task. Many application areas need a spatial-temporal model of demand prediction for regulating the supply more efficiently. For instance, in the field of clean and renewable energy, an adaptive robust dispatch method has been designed for wind power systems [22] but no probabilistic guarantee of the performance is ensured. Therefore, one of the challenges is to choose demand prediction models. In addition, an efficient modeling algorithm for a large dataset needs to be developed, and different performance measures need to be set to evaluate the models.

1.4 Contributions of the Thesis

The main contributions of this dissertation are using **Data Driven Optimization Framework** to fill the gap between demand prediction and dispatching in taxi services from a high-level perspective. We used machine learning approaches to predict passenger demand patterns in order to solve the taxi dispatch problem. Besides, we want to utilize the information provided by a large amount of sensing data to optimize taxi dispatching strategies in cities. Moreover, we design both optimal taxi dispatch models and passenger demand modeling algorithms. Furthermore, evaluations based on applying our methodology on a dataset from metropolitan areas in the NewYork City show that our framework reduces the total idle distance of all taxis, and supply is more balanced across different regions of the city.

1.5 Thesis Outline

The remainder of this research is organized as follows: Section 2 provides a survey on research conducted and related literature. In Section 3, the problem is defined as well as our proposed method which introduces the data-driven optimization method, two machine learning techniques, SVR, KNN, and the Mixed Integer Programming (MIP) formulation. In Chapter 4, experimental results based on NYC taxi data trip is illustrated. This chapter contains a data introduction, experimental results, and performance measures. The conclusion and future works are covered in Chapter 5.

Chapter 2

Related Literature

The problem of spatiotemporally imbalanced taxi supply and trip demand has been a significant difficulty to market effectiveness for a long time. When supply surpasses the demand, the driver's bare losses from idling around without fulfilling demand. On the other hand, passengers suffer from a long reply period of needs when there is a supply shortage. In this research, we present a data-driven optimization framework to dispatch the available fleet of taxis to different regions in a given area, using spatiotemporally demand forecasting models. The objective functions include decreasing taxi idle driving distance and matching the supply-demand ratio for equal service quality.

Our proposed approach includes machine learning models to predict demand in combination with an Operations Research (OR) model for dispatching taxis to different regions, which results in higher taxi utilization and customer satisfaction. Then, we solve the problem via offline optimization as a MIP problem.

TDP has been widely studied in recent years as a popular application of intelligent transportation. A recent relevant review can be found in [28, 41, 29]. Despite past efforts in handling the TDP, there is still much work needed to develop an applicable methodology. In order to design an efficient system, we should incorporate the historical passenger demand patterns into a demand prediction design. This leverages the taxi supply based on the Spatio-temporal dynamics of passenger demand, instead of taking the average or using probability distribution to create taxi dispatching. In order to suggest an applicable methodology, we need to construct a model based on the accurate future forecast, which can be obtained by applying machine learning techniques on historical data. The mainstream of this research can be categorized into two perspectives: optimization and machine learning. Moreover, the passenger demand in the taxi dispatch problem using optimization techniques can also be predicted by stochastic/deterministic models or using machine learning-based techniques. We categorize existing literature based on passenger distribution presumption into stochastic, deterministic, and machine learning forecasting approaches.

2.1 Stochastic Optimization Models

Stochastic modeling is a technique for estimating probability distributions of potential demands. The selection of an appropriate model is crucial as it shows the fundamental structure of the series, and the fitted model is valuable for future uses.

Most existing taxi dispatching systems consider the passenger demand following probability distribution:

Phithakkitnukoon et al. [32] used the naive Bayesian classifier with an advanced error-based learning method to estimate the number of vacant taxis at a given time, which can be utilized to improve the system to create online maps to navigate the passengers to stations. The objective was to predict the number of vacant taxis in a grid cell for a given time. For that aim, they estimated the probability of a certain number of vacant taxis within a cell given some observable drawn from historical data. They considered the number of vacant taxis to follow a Poisson distribution over time intervals.

Yuan et al. [46] formulated a probabilistic formulation to integrate the taxi behaviors on each road segment and parking place as well as the mobility patterns of passengers. The authors calculated the overall time-dependent distribution for both the parking places and the road segments.

Another recommendation system with no commitments to pick up has been proposed by Lee et al. [16]. They formulated the problem as taxi-customer negotiation in which the ultimate objective of the negotiation was to maximize the global utility while minimizing the total waiting time for both passengers and taxis. Customers were assumed to be generated by a stochastic process over time, which was modeled as a Poisson point process.

A rebalancing strategy for on-demand mobility taxi stations based on the fluid model has been proposed in (Pavone et al.)[31]. In which customers arrived stochastically at each station following a Poisson distribution. They showed that the optimal rebalancing policy could be generated as a solution to a linear problem. The objective was to make sure that every station reaches an equilibrium in which there are excess vehicles and no waiting customers.

Yang et al.[43] used a stochastic model for estimating customer waiting time and demand. Their objective was minimizing waiting time while maximizing taxi drivers' net utility by optimally locating the taxis.

Lees-Miller et al.[17] investigated empty vehicle redistribution on anticipating future requests. Their offered methods resulted in lower waiting times. The authors proved that the movement of empty cabs remarkably decreases the waiting times. They considered passengers' requests to follow a Poisson distribution.

Some of the existing recommendation systems are based on taxi-passenger matching models. An advisory system based on principles of stable matching has been proposed by Bai. R et al.[4]. They proposed an algorithm based on a game-theoretic model that regarded the problem as a non-cooperative game among taxi drivers, such that each taxi driver cannot find a better choice than their assigned passenger and route. The requests for a taxi service from the passengers were randomly distributed in 240 minutes, in addition to the random distributions of their locations and destinations. The time of each request was determined using a Poisson distribution.

Lowalekar et al. [23] investigated online Spatio-temporal matching of services in shared transportation. In order to overcome the myopic approaches, which are widely used in large-scale online matching, they presented a multi-stage stochastic optimization formulation to consider potential future demands.

2.2 Deterministic Optimization Models

Another category of researchers that considered the demand to be deterministic, while because of the random nature of travelers and destinations, this consideration can make efficient systematic dispatching a challenge. Different strategies and models have been proposed in the literature to cope with the vehicle dispatching problem with deterministic demand.

There are related works in order to build an advisory system based on deterministic demand.

Duan et al. [10] investigated the autonomous taxi (aTaxi) dispatching problem in hybrid request mode, which involved an immediate request and reserved services. Further, both centralized and decentralized autonomous dispatchers were planned to create short/long term routes for aTaxis. Centralized autonomous dispatchers integrated vehicle-to-passenger allocation with empty vehicle rebalance to ensure solution quality, while the decentralized one distributes partially to reduce the centralized dispatcher's workload. A certain percentage of historical requests from the New York City taxi dataset were randomly selected for passenger demand prediction.

Powel et al. [33] considered the potential profit for taxi dispatching based on fixed passenger demand. The methodology had the potential to maximize profit by offering potential locations to taxicab drivers, which can reduce overall cruising time. They created a location offering Spatial-temporal profitability (STP) map at the beginning of a cruise trip based on a profitability score.

Shi et al.[35] offered an outline to help drivers to find a taxi stand with waiting passengers. A discrete-event dynamic simulation model was applied to simulate the movements of taxis and to forecast various taxi systems' performance characteristics, such as taxi operation profit and customer waiting time. The customer demand pattern was assumed given and fixed, in a way that the average customer demand in a taxi stands for the first half-hour was one trip every two minutes.

An advisory system, which suggests taxis to change location to areas with higher demand by issuing advisory tokens to match the preallocation, has been developed by Choo et al.[9]. Passenger demand was projected by adding the number of pickups. The waiting time for clients was improved while taxis collaboratively try to determine the regions that require more free taxis and advisory tokens were then generated to request some nearby taxis to move to these regions.

Cheng et al.[8] proposed a service choice model that helps drivers in choosing which taxi stand to serve. For this aim, the average number of customers per hour based on historical data has been considered. The service choice model consisted of two options. Option A served the taxi stand, while Option B served the general network. The goal was to solve the problem as an optimization problem to maximize the expected revenue by calculating the expected rewards of each option.

Gao et al.[11] proposed a new mobile taxi-hailing system, which dispatches vacant taxis to taxi-hailing passengers proactively. The target was to maximize system utility, which was consisted of the total net profits of all taxis and the waiting time of passengers. The problem formulated as a special weighted bipartite matching problem.

The interests of passengers, drivers, and the company that may not align with one another has been addressed by Zheng et al.[52]. Passengers and taxi drivers will have preference orders for each other. A hybrid procedure was offered to balance the interests of passengers and taxi drivers. The objective was to find a constant matching between demand points and taxis.

Another taxi-passenger matching model was proposed by Situ et al.[37]. The goal was to optimize the total profit of the taxi network by matching the vacant taxis and passengers when taxi resources are insufficient. The authors used Ant Colony Construction, an enhanced form of the ant colony optimization to solve the model. The authors used real data from a real-world taxi company in Beijing to evaluate their methodology.

Miao et al. studied a series of TDP from different perspectives. In [26], a Receding Horizon Control (RHC) outline was proposed to dispatch taxis, that incorporated extremely spatio-temporally correlated demand and supply models. The objective was to match Spatio-temporal ratio between demand and supply for service quality with the minimum current while anticipating future taxi idle driving distance. Besides, RHC involved a variety of predictive models and robust optimization models which were capable of hedging against uncertainty. In [7] a hierarchical framework was proposed to implement strategies that can allocate vehicles to serve passengers in different locations considering the increasing demand and limited vehicle supply. Moreover, the framework involved two hierarchies. In the higher hierarchy, idle mileage was optimized based on current receding horizon control and predicted future requests, in the lower one, pick-up and drop-off schedules were obtained by solving a MIP model.

2.3 Machine Learning-Based Models

Recent machine learning approaches for TDP can be found in [42]. We noticed that most of the research related to the machine learning approach concerns demand prediction, which plays a vital role in TDP.

Liu et al. [21] analyzed several factors related to temporal and spatial densities of taxi pickup locations using a Generalized Additive Mixed Model (GAMM), which was a semi-parametric statistical approach. We note that in this research, we use GAMM for the following three purposes: (1) temporal dependence by capturing the interaction effects between a time metric and various time-varying variables, (2) applying a non-parametric additive function to model covariate effects using a polynomial spline estimation and (3) unobserved heterogeneity and to address data over-dispersion by including random effects.

Zhou et al. [53] studied Multi-step Citywide Passenger Demand Prediction (MsCPDP), which is more preferable than the next-step prediction. In order to overcome the inherent disadvantage of MsCPDP, which involves complicated spatiotemporal correlations in the distribution of passenger demand and lack of ground truth from pre-steps for the prediction of subsequent steps, they design a deep-learning-based prediction model with a spatiotemporal attention mechanism. The model named ST-Attn adopts the typical encoder-decoder framework without neural network units. Besides, the pre-prediction result was obtained by spatiotemporal kernel density estimation, which provides a reference to further accurate prediction.

Hu et al. [13] proposed a spatial-temporal prediction model for trip demand based on points of interest and multivariate long/short term memory model, in which the influence of meteorology information on the reliability of historical trip demand data and regional imbalance caused by POI were considered in temporal and spatial predictions, respectively. In order to balance the gap between customer demand and supply, a framework that involved four components: GPS data dimensionality reduction, pattern analysis, two-stage forecast model, and model verification was designed [19]. The non-linear SVM was used to recognize not only the mobility pattern but enhance the prediction accuracy to improve the taxi utilization.

Zhao et al. [51] proposed a framework to predict taxi and Uber demand. The approach involved a temporal-correlated entropy that measured the demand regularity and obtained the maximum predictability and five representative predictors, which can achieve the maximum predictability. Further, the experiment validations implied that the maximum predictability could help determine which predictor to use in terms of the accuracy and computational costs.

Shou et al. [36] studied the optimal passenger-seeking policies using the Markov decision process (MDP) and imitation learning for e-hailing platform. The e-hailing drivers' features were embedded into an MDP model, and the reward function was covered by leveraging an inverse reinforcement learning technique. The model was validated through a Mont Carlo simulation to show that the proposed MDP model is capable of capturing the supply-demand ratio.

Safikhani et al. [34] proposed a generalized Spatio-temporal autoregressive (STAR) model to explain the user demand for taxis through space and time since various factors such as commuting, weather, road work, and closures have a great impact on transit services. The LASSO-type penalized methods were introduced for tackling parameter estimation.

The availability of big data shows that trips from large scale networks tend to be periodic on spatiotemporal properties. Nevertheless, few studies aim to forecast the demand and supply of vehicles in high-density cities.

Ling et al. [20] proposed a two-stage forecast model based on big data to fill the gap between demand and supply in large scale the networks. The methodology combined both non-linear support vector machine and backpropagation neural network. The suggested framework not only revealed the mobility pattern, but it also improved the prediction accuracy for the gap between demand and supply of taxis, thus helps to improve the taxi utilization.

In ride-sharing services, such as UberPool and Lyftline, where multiple customers with similar itineraries are scheduled to share a vehicle, Al-Abbasi et al.[3], developed a framework, that uses deep Q-network (DQN) techniques to learn optimal dispatch policies by interacting with the environment. The method efficiently incorporated travel demand statistics and deep learning models to manage dispatching vehicles for improved ride-sharing services.

The objective was to efficiently dispatch the available fleet of vehicles to different locations in a given area in order to achieve the following goals: (1) satisfy the demand (or equivalently minimize the demand-supply mismatch), (2) minimize the waiting time of the customers (time elapsed between the ride request and the pickup), as well as the dispatch time which is the time to move to another zone to pick up new customers (maybe, future customers), (3) the extra travel time due to participating in ride-sharing, and (4) minimize the number of used vehicles/resources. The author used a convolutional neural network to predict future demand. The output of the network represented the expected number of ride requests in a given zone for 30 minutes ahead.

Although these works provide solid results for related taxi scheduling problems, they all considered a presumption for passenger demands, and none of them incorporates the historical passenger demand patterns into a demand prediction design, leveraging the taxi supply based on the Spatio-temporal dynamics of passenger demand. While our contribution is using machine learning approaches to predict the passenger demand pattern in order to solve the taxi dispatching problem.

The objective of this thesis is twofold: (1) to use the machine learning approaches to predict the passenger demand, and (2) to solve the taxi dispatching problem as a MIP model in order to minimize total idle driving distance while satisfying service quality equally in different districts. Our work presents a simple yet practical method for reducing cruising miles by assigning taxicab drivers to locations based on predicted demands. In our approach, historical data serves as experience that can be feed to the ML model in order to predict future demand.

Chapter 3

Data Driven Optimization

3.1 Problem Statement

The proposed taxi dispatch system aims to schedule and navigate empty taxis towards upcoming travelers with the lowest idle mileage. Considering the spatiotemporal patterns of demands in the city, the dispatch center dynamically assigns empty taxis to different regions to fulfill demand.

We utilize the supply-demand ratio of various areas as a criterion of service quality since sending a higher number of taxis to more requests is a standard system-level obligation, to ensure the demand of various areas similarly fulfilled. Similar service metric)service node utilization rate(has been applied in resource allocation problems and autonomous driving car mobility control [49]. To compute a dispatch solution, the system is equipped with Machine Learning (ML) techniques to predict spatiotemporal patterns of passengers' demands based on historical data. The data comprises the taxi's GPS location with a timestamp that periodically reported to the dispatch center.

In addition to harmonizing supply and demand, including future costs when evaluating a given solution is essential. It is hard to precisely forecast the future of the large-scale taxi service system in the real-world; hence, we utilize a heuristic idle driving distance to describe anticipated future costs associated with meeting passenger demands.

We assume that the optimization horizon is defined by set T. To provide a simple structure for the problem, we examine the dispatch solution for the next time-slot,

although there exists a potential for later considerations. This is mainly because our central dispatch system does not control the drop off location, which disables the optimization algorithm from supporting the mobility function between time intervals.
Public and private service systems categorize the demand sites into one or more service regions (or districts) to reduce the problem size. For simplicity in our problem, we assume that the taxi stations are located at the center of each district. Each district has two dimensions, corresponding to longitude and latitude. Our model consists of one dispatch center, n districts, and capacity m, which indicates the total number of available taxi units to serve passenger demands occurring at each region.

3.2 Sets, Parameters, and Decision Variables

With a large amount of historical data on taxi GPS and occupancy status, we extract necessary demand information. We assume that the city is divided into n regions. For instance, a city can be divided into administrative sub-districts. We also assume that during a time slot, the total number of passenger demands needs to be served by current vacant taxis at the $j-th$ region is denoted by r_j. In addition, the total number of demands in the entire city is denoted by $C = \sum_{j=1}^{j=n} r_j$, given that the demand in one time slot (i.e. a one-hour period) is an unknown parameter, and the maximum number of vacant taxis available in that time slot to be allocated is m.
The initial supply information consists of GPS position for the $i-th$ vacant taxi, denoted by $P_i^0 \in R^2$; while $i = 1, \ldots, m$.
The matrix $P^0 \in R^2$ represents the initial location of available taxis with each row as P_i^0. Before allocation, each taxi is in one of the taxi stations and we assume there exists a taxi station at the center of each region, meaning that the $i-th$ row of this matrix corresponds to the position of $i-th$ taxi (longitude and latitude) at the beginning of first time slot.
Specifically, each region has a predicted number of passengers demands which dispatch system needs to satisfy it by allocating vacant taxis. Note that the supply-demand ratio at each region before dispatching is unbalanced.
With the above initial information about supply and demand, we define $y_{ij} \in \{0, 1\}$ as a binary variable to calculate a dispatch decision at each region for current vacant taxis. This variable will get a value equal to 1 if taxi i is dispatched to region j, and

0 otherwise. We assume that the allocation time horizon is one hour, given predicted demand.
We define sets, parameters, and decision variables as follows to formulate the mathematical model of the problem.

Index	**Description**
i	the index of taxi
j	the index of region
t	the index of time slot
Sets	**Description**
M	the set of vacant taxis
N	the set of regions
T	the set of time slots
Parameters	**Description**
$m \in \mathbb{Z}_+$	the total number of vacant taxis
$n \in \mathbb{Z}_+$	the total number of regions
$r_j \in \mathbb{Z}_+$	the total number of predicted demand at region j
$e \in \mathbb{R}^n$	error vector
$C \in \mathbb{Z}_+$	the total number of predicted demand for all regions
$\alpha \in \mathbb{R}$	the upper bound vector of driving distance
$\beta \in \mathbb{R}$	the weight factor of objective function
$p^0_{i,k} \in P^0 \subseteq \mathbb{R}^2$	the initial position of vacant taxi i in terms of longitude and latitude for k=1 and k=2, respectively
$w_j \in W \subseteq \mathbb{R}^2$	longitude and latitude of $j-th$ region (taxi stations)
Variables	**Description**
$y_{i,j}$	the binary dispatch variable that represents the region j which taxi i should go
x_i	idle driving distance of the $i-th$ taxi to reach the dispatch location

Table 1: Notations of the mathematical model

We would like to find a dispatch solution that balances the supply-demand ratio while satisfying practical constraints and not introducing large idle driving distance.

3.3 The Mathematical Model and Formulation

A given route for a taxi needs a starting and a destination point so that the driver could follow the path shown on the GPS unit inside the taxi based on the information mentioned above. Since the designing routes is not the focus of this work, the dispatch center can only send a two dimensional GPS location for the taxi driver as the destination. In the reality, there are several taxi places on the street in an urban area. Taxis could choose an ideal station, or they would be randomly assigned to a preferred station by the monitoring system in every region. This preferred location for taxis is presented by $P^0 \in \mathbb{R}^2$. P_i^0, are the initial longitude and latitude positions of taxi i at the beginning of the assignment.

With the above initial information about supply and predicted demand, we defined the binary variable y_{ij} as the dispatch order variable, where $y_{ij} = 1$ if and only if the $i-th$ taxi is sent to the $j-th$ region. We define the below constraint to dispatched every taxi to one region.

$$\sum_{j=1}^{n} y_{i,j} = 1 \quad \forall i \in M, \tag{1}$$

3.3.1 Multi-Objective Optimization

This research aims to find solutions that balance the supply-demand ratio while satisfying the minimum idle driving distance. Once the center has made dispatch decisions, the solution details will be sent to vacant taxis.

In order to measure how supply matches demand at each region, we use a measure called supply-demand ratio. For region j, the supply-demand ratio is the total number of vacant taxis divided by the total number of passengers' demands during a given time slot.

We assume that the supply-demand ratio for each region j is equal to that of the entire city, so we have the following equation for $j = 1, \dots, m$,

$$\frac{\sum_{i=1}^{i=m} y_{i,j}}{r_j} = \frac{m}{C} \tag{2}$$

We rewrite the equation (2) as the following equation:

$$\frac{1}{m} \sum_{i \in M} y_{i,j} = \frac{r_j}{C} \tag{3}$$

However, the above equation can be too strict if employed as a constraint, and there may be no feasible solutions for the problem. This is because decision variables y_{ij}, are integer and taxis' driving speed is limited that they may not be able to serve the demands from any arbitrary region during a given time slot. Instead, we transform the constraint into a soft constraint by introducing a supply-demand mismatch penalty function J_E, which means that the supply-demand ratio should be balanced across the entire city. Based on the given illustrations, the first objective function of the dispatch problem is presented as follows:

$$J_E = |\frac{1}{m}\sum_{j\in N}(\sum_{i\in M} y_{i,j} - \frac{r_j}{C})| \tag{4}$$

The second objective function, which is considered in this research, aims at reducing the total driving distance from the initial location to the dispatch location. The dispatch center is required to send the location of the destination to vacant taxis. For this purpose, we locate the destination district with a longitude and latitude position. Besides, we assume that the taxi stations are located at the center of each district. The location of each taxi station at each region in the city is stored as a matrix W at the dispatch center, where each row w_j represents the two-dimensional geometric position of the taxi station at district j.

Once y_{ij} takes a value of 1, then $i-th$ taxi goes to the location $y_{ij}w_j$. Thus we have a driving distance p_i^0 to $y_{ij}wj$ for taxi i, in order to reach the dispatch position, which is going to produce additional cost (the taxi is driving empty to reach the position to respond to the demand). To estimate the distance without having information about the exact path, we use the Manhattan norm. Also, x_i is the estimated idle driving distance of the $i-th$ taxi to reach the dispatched location $y_{ij}wj$. In order to find the lower bound of idle driving distance, we consider below equation:

$$\sum_{k=1}^{2}|P_{i,k}^0 - \sum_{j\in N} y_{i,j}w_{j,k}| \leq x_i \quad \forall i \in M, \tag{5}$$

As we mentioned earlier for time slots greater than one, we need to consider drop off locations, which in our case are not directly controlled by the dispatch center. For this project, we just consider the first time slot and practice demand prediction with machine learning approaches as input for taxi dispatch problem, while expanding the

project to more time slots can be considered in future studies.
Within the defined time slot, the distance that each taxi can drive should be bounded by a constant vector of $\alpha \in \mathbb{R}^N$. This is to regard the speed limit to respond to the predicted demand within the time slot, so taxis cannot drive more than a certain distance to get to the dispatched region.
Below equation provides an upper bound for idle driving distance.

$$x_i \leq \alpha \qquad \forall i \in M, \tag{6}$$

The total idle driving distance of all the vacant taxis within the first time slot is calculated by the following equation.

$$J_D = \sum_{i=1}^{m} x_i \tag{7}$$

It is worth mentioning that the idle distance we estimate here is at the region-level distance to pick up predicted passengers — the distance is nonzero only when a vacant taxi is dispatched to a different region. We also require that the estimated distance is a closed-form function of the locations of the original and dispatched regions, without knowledge about certain traffic conditions or exact time to reach the dispatched region. Hence, in this work, we use Manhattan norm to approximate the idle distance.
For multi-objective optimization problems, no single solution can be found that simultaneously optimizes each objective. In this case, the objective functions are said to be conflicting, and there exists a set of Pareto optimal solutions that make trade-off among optimization of different objective functions. A solution is called non-dominated or Pareto optimal, if none of the objective functions can be improved in value without degrading some of the other objective values. Without additional subjective preference information, all Pareto optimal solutions are considered equally good. The goal may be to find a representative set of Pareto optimal solutions, and/or quantify the trade-offs in satisfying the different objective functions. By providing the set of Pareto optimal solutions to the Decision Maker (DM), he/she can choose the best solution based on his/her preferences. In this research, we define a weight parameter β when summing up the costs related to both objective functions.
Since solving a complex mixed integer programming model is not straightforward in

large scales, one standard relaxation method is to relax the binary assumption of the binary variables which is known as Linear Relaxation (LR) and can be derived replacing the binary constraint

$$y_{ij} \in \{0,1\} \qquad by \qquad 0 \leq y_{ij} \leq 1$$

Based on the given illustrations and notations, we propose the mathematical model of the problem as follows.

$$\begin{aligned} &\min_{y_{ij},x_i} .J = J_E + \beta J_D = \\ &\min |\frac{1}{m}\sum_{j\in N}(\sum_{i\in M} y_{i,j} - \frac{r_j}{C})| + \beta \sum_{i\in M} x_i \end{aligned} \tag{8}$$

$$\sum_{k=1}^{2} |P_{i,k}^{0} - \sum_{j\in N} y_{i,j} w_{j,k}| \leq x_i \quad \forall i \in M, \tag{8a}$$

$$x_i \leq \alpha_i \quad \forall i \in M, \tag{8b}$$

$$\sum_{j\in N} y_{i,j} = 1 \quad \forall i \in M, \tag{8c}$$

$$x_i \geq 0 \quad \forall i \in M, \tag{8d}$$

$$y_{ij} \in Y \subseteq \{0,1\}^{m\times n}. \tag{8e}$$

The problem we consider in this work aims at reaching good service to increase global customers' satisfaction, which is indicated by a balanced supply-demand ratio across different regions, instead of minimizing each customer's waiting time when a request arrives at the dispatch system. A similar criterion has also been used in mobility on demand systems [50]. In some situations, the taxi i will not pick up passengers in its original region but will be dispatched to another region. Such a dispatch decision results from the fact that global customers' satisfaction level should be increased instead of individual satisfaction level. For instance, when the original region of taxi i has a higher supply-demand ratio than the dispatched region, going to the dispatched region will help to increase customer's satisfaction in that region. By sending taxi i to some other region, customers' satisfaction in the dispatched region can be increased, and the value of the function J_E can be reduced without extra idle driving distance J_D.

3.3.2 Single-Objective Optimization

As there usually exist an enormous number of Pareto optimal solutions for a multi-objective optimization problem, thus we convert the original problem with multiple objectives into a single-objective optimization problem to reduce solution complexity known as e-constraint approach. While in experimental results, for the multi-objective formulation, we compare Pareto optimal solutions; in this section, we convert the problem into a single-objective model.

We define parameter e as a predefined supply-demand ratio error, which can be obtained from the best and the worst values of the supply-demand ratio error objective function. While we want to minimize the service quality in different regions, we can define an acceptable amount for e and consider J_E as a constraint (9a). Then we can solve the allocation problem for minimizing total idle driving distance. Thus the mathematical model is formulated as follows:

$$\min \quad \sum_{i \in M} x_i \tag{9}$$

$$\textbf{s.t.} \quad \left|\frac{1}{r_j}\sum_{i \in M} y_{i,j} - \frac{m}{C}\right| \leq e_j \quad \forall j \in N, \tag{9a}$$

$$\sum_{k=1}^{2} \left|P^0_{i,k} - \sum_{j \in N} y_{i,j} w_{j,k}\right| \leq x_i \quad \forall i \in M, \tag{9b}$$

$$x_i \leq \alpha_i \quad \forall i \in M, \tag{9c}$$

$$\sum_{j \in N} y_{i,j} = 1 \quad \forall i \in M, \tag{9d}$$

$$x_i \geq 0 \quad \forall i \in M, \tag{9e}$$

$$y_{ij} \in Y \subseteq \{0,1\}^{m \times n}. \tag{9f}$$

The new mathematical model allocates empty taxis towards predicted demand points while minimizing the total idle driving distance for all taxis. Note that we maintain a predefined global service quality by setting the supply-demand ratio error less than e. The optimal solutions to the single-objective optimization problem (9) are Pareto optimal solutions to the multi-objective optimization problem (8). Note that the difference between this model and the model presented in the previous section is that by assigning different weights to the objectives, we can drive several Pareto optimal solutions for the problem in the previous section. However, in the latter model, we can assign an upper bound based on DMs' opinion on the supply-demand error ratio and obtain Pareto optimal solutions that are of great importance to DMs with remarkably less computational effort.

3.4 Data Driven Method for Taxi Dispatching Problem

In this research, we present a framework to model the TDP as a time-series decision-making problem. More specifically, we use time-series data as an input for machine learning demand forecasting models such as Support Vector Regression (SVR) and K-Nearest Neighbor to predict the future passenger demand. The predicted demand will then be fed into the MIP model to dispatch an available fleet of taxis to optimize the objective function. The aim is to minimize the total taxis idle driving distance while satisfying the passenger demands at a certain service level (measured by supply-demand ratio). Note that the mathematical models are solved using Gurobi 29.0 academic version, and KNN and SVR are implemented using Python 3.7.4 on a laptop with an Intel i7 CPU, 16 GB RAM, and macOS 10.15.

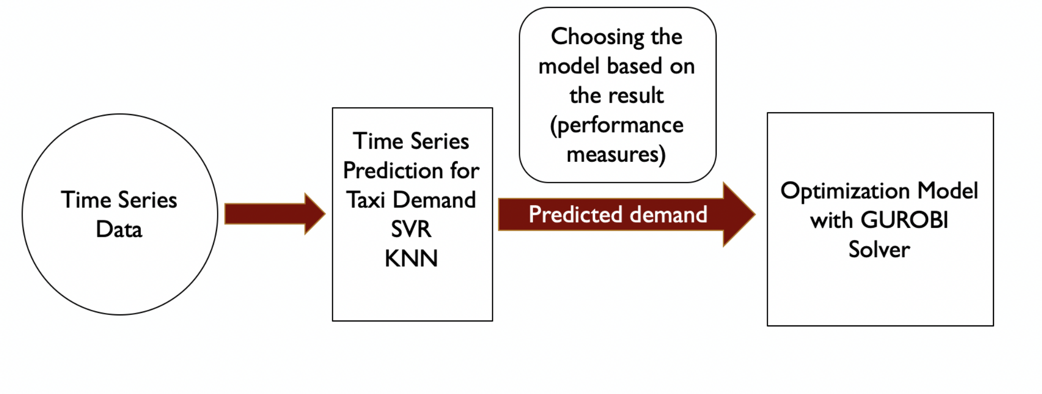

Figure 2: Data Driven Approach for Taxi Dispatching Problem

3.5 Time Series Forecasting Methods for Taxi Demand Prediction

The nature of the taxi demand allocation is a time series problem, which needs an accurate model to predict the distribution of future taxi demands considering historical data. Time series prediction can be defined as a procedure that obtains essential information from historical data and then provides future values. The study and forecast of time series have always been the main methods in an array of practical problems, including weather forecasting, transportation planning, traffic management, and so on. Generally, the goal of time series $y = y_1, y_2, ...$ is to estimate the value y_t at time t based on its previous k values $y_{t-1}, y_{t-2}, ..., y_{t-k}$. This problem can be formulated as a function $\mathbf{x} = y_{t-k}, y_{t-k+1}, ..., y_{t-1}$ for y_t so that $\hat{y}_t = f(\mathbf{x})$, where $\hat{y}_t$ is the closest estimation of y_t. Different approaches for time series forecasting, such as Auto-Regressive Integrated Moving Average (ARIMA), Support Vector Regression (SVR) and K-Nearest Neighborh (KNN) have been extensively studied and applied. However, it is usually challenging to know which technique is the best for a particular data set. The choice of the most promising algorithm to model and predict a particular phenomenon is the first decision in time series forecasting.

Time-series methods can be categorized into parametric and nonparametric methods. In nonparametric approaches (ML techniques), the model does not presuppose the data distribution. However, the parametric approaches assume that data follow a known distribution. The parametric algorithms are the state-of-the-art of time series prediction, and they are often related to statistical approaches. The employment of parametric approaches like ARIMA requires expertise both in the application and in computational mathematics, as the number of parameters that need to be estimated is more than nonparametric techniques. Moreover, DMs' insight and knowledge are vital for the parametric modeling procedure. Therefore, the cost of ARIMA models is high, based on the complexity of the application domain and computational mathematics.

Machine learning prediction approaches, in contrast to statistical models, define the data properties without the previous information of their distribution. These methods are extensively used for building precise prediction models based on data that has been provided by GPS technology [27]. In the last decade, GPS-location systems have attracted the consideration of both academics and businesses because of the novel type of accessible information. Particularly, the location-aware sensors and the information transmitted are tracking human behavior, and they can be used collaboratively to disclose their movement patterns. Trains, Buses, and Taxi Networks [18] are currently effectively discovering these traces. Using ML models for time series involves the transposition of the data sequence typically into an attribute value table to use as input to the machine learning regression algorithm. The ML models can be divided into two categories, including global and local approaches. The global approaches consider all the observations of the training series to build a model. SVR and Artificial Neural Network (ANN) are considered to be global nonparametric methods. With the development and improvement of SVR techniques [47, 6], a breakthrough in the area of demand forecasting occurred. The initial aim of SVR was to solve pattern classification problems, but later they have been widely applied in many other fields such as function estimation and regression problems. The impressive characteristic of SVR is that it is intended for a better generalization of the training data. In most cases, in SVR, the solution only depends on a subset of the training data points, called the support vectors. Furthermore, with the help of support vector kernels, the input points in SVR applications are usually mapped to a high dimensional feature space, which often generates good generalization outcomes. For this reason, the SVR methodology has become one of the well-known techniques in recent years, especially for time series demand forecasting problems.

The local approach partitions the time series into sub-sequences, where the closest or most important values, connected to the present value, are combined to produce the future values. KNN model is one of the most commonly used methods of nonparametric regression and classification in order to predict future values. Since usually a specific value is not influenced by observations that happened a long time ago, similarity-based methods, like the KNN classifier, are characterized by not constructing a model that explicitly describes the training dataset behavior. KNN is useful to predict highly nonlinear and complex time series patterns. The approach is local, so the closest or the most important values related to the current value are combined to produce the future value.

In this thesis, we use SVR and KNN to forecast the passengers' demands in a taxi dispatch system. To forecast taxi demand in the future, many features can be considered to build a good model. While in this work, we just focused on the number of pickups at different regions and pick up time to predict the passenger demand, which refers to the number of pickups submitted per unit time (e.g., every hour) and per unit region (e.g., each POI). Predicting passengers' demand is non-trivial for large-scale taxicab platforms because both accuracy and flexibility are essential.

Adapting the k-nearest neighbor and support vector regression models to improve the forecasting accuracy is a favorable return of the global positioning system (GPS), which through that, data could be observed as a channel to obtain the original data of the demand flow. In the following sections, the necessary procedure of the KNN and SVR models are introduced.

3.5.1 K Nearest Neighbors

Similarity-based approaches such as the KNN classifier, are categorized by not building a model that only defines the training dataset behavior. The overview of the training set is performed whenever we request a new classification from the algorithm. The idea behind the revision of KNN for time series prediction is very intuitive.

Given a series of data points

1. Computing the distance (Euclidean or Manhattan) between the point and all the other points in the training set.
2. Choosing the nearby K training data points.
3. Calculating the average or weighted average of the target output values of these K points, which are the final predicted results.

Implementing the basic version of KNN is straightforward by calculating the distances to all stored examples. The critical setting of KNN is the parameter K, which should be selected carefully. A large K will help to build a model with a lower variance but higher bias. By contrast, a small K will result in higher variance but lower bias. In the field of taxi demand forecasting, KNN is interpreted to find the nearest patterns in the history considering the current pattern. Those neighbors work as references to make predictions for the future standing at the current time.

Figure 3 presents a schematic view of how KNN works. Suppose we stand at the end of series B and try to make a prediction. By KNN, a similar series called series A is identified as the nearest neighbor of series B, so that the historical point can be a good reference to make the prediction.

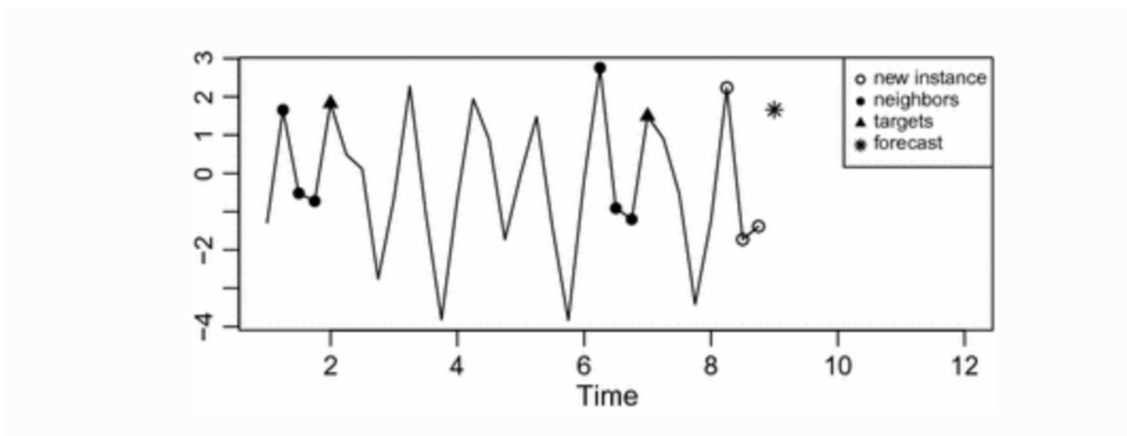

Figure 3: One-step-ahead forecast with 2-NN regression

Figure 3 [25] shows an example of one-step-ahead forecasting using lags 1–3 as explanatory variables. The last three values of the time series are the new instance to be regressed on and the two sets of consecutive black dots, the two nearest neighbors, whose targets are the triangles that are averaged to produce the forecast shown by an asterisk. The underlying intuition to using KNN on time series forecasting is that any time series contains repetitive patterns, so we can find previous similar patterns to the current series structure and use their subsequent patterns to predict the future behavior.

Several surveys have been conducted to analyze the performance of KNN with different distance measures. In worth mentioning that a few papers showed the efficiency of KNN in predicting highly nonlinear and complex patterns in time series [30, 45].

3.5.2 Support Vector Regression

SVR constitutes a machine learning technique based on the statistical learning theory, which was first proposed by Corinna Cortes and Vapnik in 1995 [40]. In other words, the basic idea behind SVR is raising the dimension and linearization. It has many unique advantages in solving small sample, nonlinear, and high dimensional pattern recognition. If the predicted variable is discrete, it is called classification, and if the predicted variable is continuous, it is called regression. SVR is suggested as a useful technique based on using a high-dimensional feature space and penalizing the ensuing complexity with an error function. Considering a linear model for illustration, the prediction is given by $f(x) = w^T x + b_o$, where w is the weight vector, b_o is the bias and x is the input vector. The objective is to minimize the error function given by;

$$J = 1/2||w||^2 + C \sum_{m=1}^{M} Loss(y_m, f(x_m)) \tag{10}$$

where w is the weight vector, x_m is the $m-th$ training input, y_m is the target output and $Loss(y_m, f(x_m))$ is the loss function.

For nonlinear functions, the data can be mapped into a higher dimensional space, called kernel space. Some common kernels are linear kernel function, polynomial kernel function, and Gaussian kernel function. Among them, Gaussian kernel function, also known as Radial Basis Function (RBF), is the most widely used one, which can map data into an infinite dimension.

Support vector regression (SVR) has two significant advantages:

1. The model produced by SVR depends only on a subset of the training data, called the support vectors, because the loss function ignores any training data close to the model prediction. Therefore, SVR is suggested to produce better generalization results.

2. With the help of support vector kernels, the inputs of SVR are usually mapped to

a high dimensional feature space.

The SVR model with RBF kernel requires two parameters, including (i) C, which is a regularization term that imposes a weight on the training set errors minimizing the model complexity; and (ii) σ, that reflects the Gaussian's width of the kernel function [12]. The construction of an SVR implies solving a quadratic problem with linear constraints, which depends on the set of input data, parameters, and the separation margin. During the training phase, the Lagrange multipliers that characterize the support vectors are obtained. These support vectors define the edges of the optimal separation hyperplane.

3.6 Sliding Window For Time Series Data

Time series is typically measured over successive times, representing as a sequence of data points [44]. The measurements taken during an event in a time series are arranged in proper chronological order. Time series forecasting is the use of a model to predict future values based on previously observed values. The use of previous time steps to predict the next time step is called the sliding window method. In statistics and time series analysis, this window is called time lag, and the number of previous time steps is called the window width or size of the lag [24]. Most predictive modeling algorithms will take some number of observations as input and predict a single output value. As such, they cannot be used directly to make a multi-step time series forecast. This involves making a prediction for one timestep, taking the prediction, and feeding it into the model as an input in order to predict the subsequent time step. This process is repeated until the desired number of steps has been forecasted. One approach where machine learning algorithms can be used to make a multi-step time series forecast is to use them recursively.

Recursive strategy uses forecasted values of the near future as inputs for the longer future forecasting [14]. The function of Recursive is defined as follows:

$$\begin{gathered}\hat{y}_{t+1} = f(y_t, y_{t-1}, \dots, y_{t-d+1}) \\ \hat{y}_{t+2} = f(\hat{y}_{t+1}, y_t, \dots, y_{t-d+1}) \\ \dots \\ \hat{y}_{t+n} = f(\hat{y}_{t+n-1}, \hat{y}_{t+n-2}, \dots, \hat{y}_{t-d+n})\end{gathered} \tag{11}$$

where t is the current time, $\hat{y}_{t+1}$, $\hat{y}_{t+2}$ ... , $\hat{y}_{t+n}$ are the forecasted values, Y_1, Y_2, ... , Y_t are the historical values and d is the dimension of inputs. It is called a univariate (or single) time series when d is equal to 1 and a multivariate time series when d is equal to or greater than 2.

3.6.1 Choice of the Window Size

When it comes to predict the future values in a time series problem, we need to answer this question: How many time windows (lags) we need to consider to look back in order to predict the future values.
It is evident that the larger the window the more information about time series can be considered. But it may decrease the sensitivity of the system which will produce too smooth (decreased noise) prognoses. The advantage for a smaller window size is increased sensitivity to changes in the underlying process. In order to find the optimal window size, we use Auto Correlation Function (ACF) and Partial Auto Correlation Function (PACF) graphs [39].
The ACF shows the correlation between the observation at current and th previous time spots while PACF shows The correlation between observations at two time spots given that we consider both observations are correlated to observations at other time spots. It means PACF gives "real" correlation between two time spots after taking out the influence of the other time spots.
By analyzing these two graphs, we can better understand the proper window size based on the dataset we use in our experimental results.

3.7 Performance Measures:

The evaluation of each model is based on the Mean Square Error (MSE) where the smaller values, indicate that the model has a better performance. Furthermore, we considered the prediction accuracy where the higher accuracy indicates the better performance of the model [15]. The considered performance metrics are defined as follows:

• **Accuracy:** Our goal is to predict precision values of Y, which consists of one or more positive values. To find the accuracy of the prediction, we compare $Y_{predict}$ to the actual data in the test split Y_{test} of a given dataset. Since each Y is a vector of multiple predictions, the average accuracy was calculated. That is the average of the differences between $Y_{predict} = (y_1^{'}, \ldots, y_p^{'})$ and $Y_{test} = (y_1, \ldots, y_p)$ should be calculated. The following equation shows how to calculate the accuracy for each model:

$$ACC = (1 - \frac{\mu(|Y_{predict} - Y_{test}|)}{\mu(|Y_{test}|)}) \times 100 \tag{12}$$

• **RMSE:** The Root Mean Square Error (RMSE) is used to measure the performance of the models [5]. RMSE is the square root of MSE. The square root is introduced to make scale of the errors to be the same as the scale of targets. We assume that we have determined i = 1,2, . . . ,n samples of model errors. RMSE formula is given as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{n=1}^{n}(Y_{predict} - Y_{test})^2} = \sqrt{MSE} \tag{13}$$

Chapter 4

Experimental Results

In the previous chapters, we proposed two machine learning methods in order to predict the passengers' demand in the taxi dispatch problem. In this paper, we implement our methodology on a real-world case study from New York City (NYC) taxi data set [2]. NYC has a remarkably rich history of the taxi industry. A dataset consisting of one-month (Jan 2018) NYC taxi trips is used for training and testing the models.

The advance forecast for taxi demand distribution provides valuable decision supports for any taxi company.

The taxicabs of New York City come in two types: yellow and green. Both types of taxis have the same fare structure, while green taxis are more restricted to some regions of the city.

We conduct our allocation experiment based on the New York taxi data set [2] to learn passenger requests of taxis, which serve as the input of the allocation algorithm. We note that our learning model is not restricted to the data set used in this experiment, and other forecasting models and data sets can also be incorporated. We implement the allocation algorithm in Python using an optimization toolbox called CVXPY [1]. We assume that all vacant taxis can follow the dispatch solution and go to suggested regions, although this solution is recommendation-based, and taxis can obey the recommendation in their favor. Within a target region, we assume that there is a taxi-stand in the middle of that region, and the longitude and latitude of this taxi-stand is the center of the region. We calculate the total idle mileage by Manhattan norm. The mileage between two points is approximated proportional to their geographical distance on the road map. GPS data directly provide the geometric location of a taxi.

4.1 Dataset

For several years, taxis in New York City have had GPS receivers to record information. In this thesis, we consider only the green taxis for Jan 2018. We have used Jan-2018 data to make predictions for the next one hour of the last date of the given dataset. The dataset has been split into training sets and testing sets (80% for training and 20% for testing).

Our objective is to predict the number of passengers' demands as accurately as possible for each district. Based on our dataset, there are 248 different location IDs. There are different features in the main dataset like pick up location ID, passenger counts, and trip distance. Among those, we used to pickup location ID and pick up date-time (consists of the date and time that pick up had occurred) for our experiment as follows:

- Pickup datetime: start time of the trip
- PU location ID: location ID at the start of the trip

Based on the taxi zone map of New York, we can find the IDs related to region names to find the corresponding longitude and latitude.

After preprocessing the dataset, the result is a new dataset with the total number of hourly pickups per region for January with 793531 records for each region.

Taxicab Dataset			
Selected Features: Pickup datetime : start time of the trip PU location ID : location ID at the start of the trip			
Taxicab Dataset			
Collection Period	Number of Taxis	Data Size	Record Number
01/01/2018-31/01/2018	989	70.9 MB	793531

Table 2: New York green taxi trips dataset for Jan 2018.

4.1.1 Determining the Input time lag (window size)

As we explained in section 3.6.1, ACF describes the correlation between two observations at a previous time step that includes direct and indirect dependence information. This means that if the input time window for our prediction is k, we would expect the ACF for our time series dataset be robust to a lag of k and the inertia of that relationship would carry on to subsequent lag values, trailing off at some point as the effect was weakened.

PACF only defines the direct connection among observation and its lag. This shows that there would be no correlation for lag values beyond k. For the PACF, we would expect the plot to show a strong relationship to the lag k and a trailing off of correlation from the lag onwards.

Below are ACF plots as well as PACF plots for one selected district based on the New York city dataset on Jan 2018.

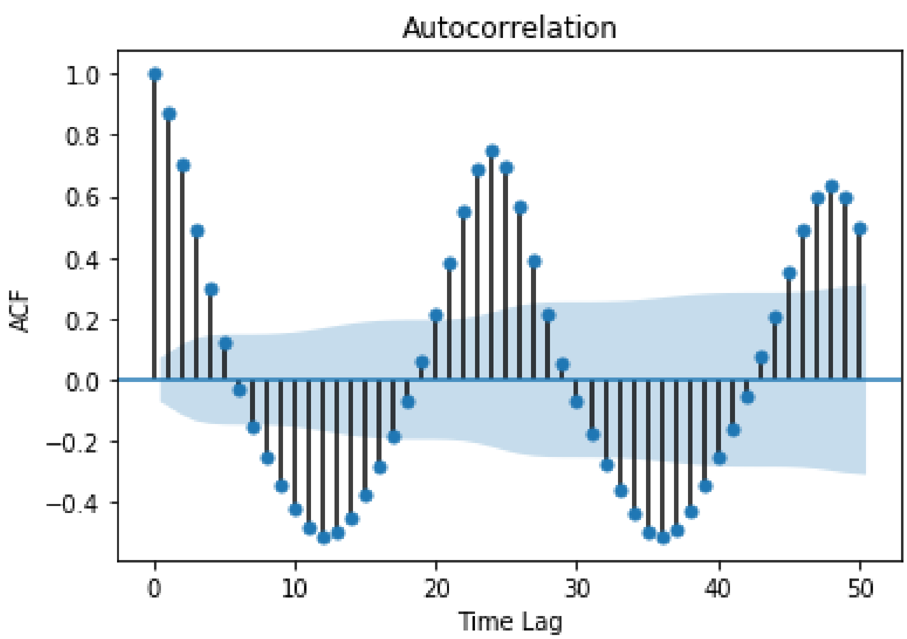

Figure 4: ACF for New York taxi passenger demand dataset on January 2018 over 50 time lags (hours) for Manhattan/Central Harlem North

Figure 4 shows how well the presented passenger demand of the NewYork taxi dataset for January 2018 is related to its past values over the last 50 hours.

The ACF plot shows seasonal peaks at daily lag for every 24 hours. A sine wave in which its amplitude decreases over time.

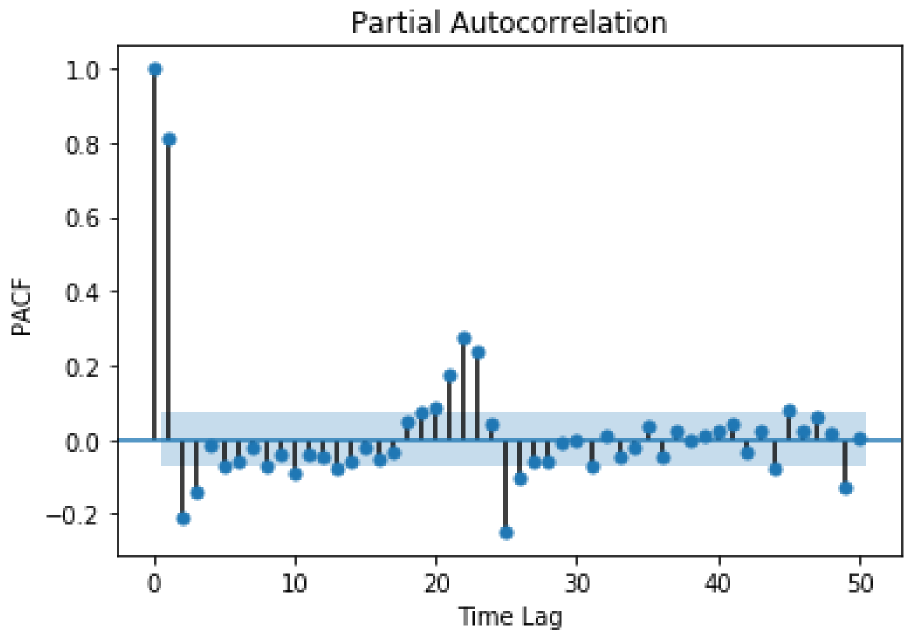

Figure 5: PACF for New York taxi passenger demand dataset on January 2018 over the 50 time lags (hours), for Manhattan/Central Harlem North.

Based on the figures, it becomes apparent that there is a statistically significant correlation in the time series dataset for this district's passenger demands at present time lag and time lag 24, and a trailing off of correlation from this time lag downward. By analyzing above mentioned plots for several different districts, we choose last 24 hours time interval, as our input argument (window size) for two machine learning methods in order to predict the future passenger demand.

4.2 Demand Prediction

4.2.1 Grid Search for Hyperparameter Selection

Grid search is the most basic approach towards automating hyperparameter optimization. To perform this search, the user defines a set of possible values for each hyperparameter. Then, every possible combination of those values (which form a grid) are evaluated for the given algorithm, and the combination of hyperparameters which achieves the minimum loss on the validation set is returned as the optimal configuration. For our experiment, we import GridSearchCV from the sklearn package in order to tune our machine learning hyperparameters.

4.2.2 Support Vector Regression

In this research, we used a support vector regression with a radial basis function kernel as the forecasting method. There is an unusual noise in the taxi demand series resulted from events such as irregular working schedule of companies, big concerts, parade, etc., which may cause over-fitting in the models. Due to the characteristics of better generalization, SVR is expected to have a good forecasting performance in these cases. For choosing the best hyperparameters, we used GridSearchCV from the sklearn package. Also, the sklearn.svm package has been used for SVR modeling.
Below table shows the list of tuned hyperparameters for our SVR model;

Hyperparameter	
C	1.0
Epsilon	0.1
Kernel	'rbf'
Degree	3

Table 3: Tuned hyperparameters for SVR model using grid search.

The same training and testing dataset are used for both KNN and SVR models. We use 80% of the dataset as the training data and 20% of the dataset as testing. Meaning that we have 576 training samples and 144 testing samples which each one of them contains sequences of 24 hours of the total number of pickups (passenger demands) for each region.
Below graphs show the results for selected regions in which the blue lines present actual demands, and the red lines show predicted demands for the selected regions.
The seasonality of the dataset can be seen in the graphs when the blue line goes down, demand has been decreased according to the night time, and by starting the day, it has been increased.

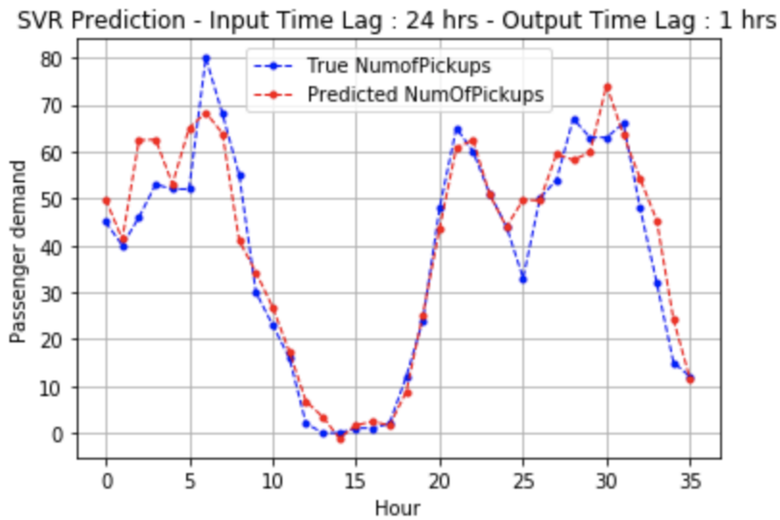

Figure 6: Comparing the actual value and the predicted value of passenger demand using SVR model for New York, January 2018 dataset location Brooklyn Heights.

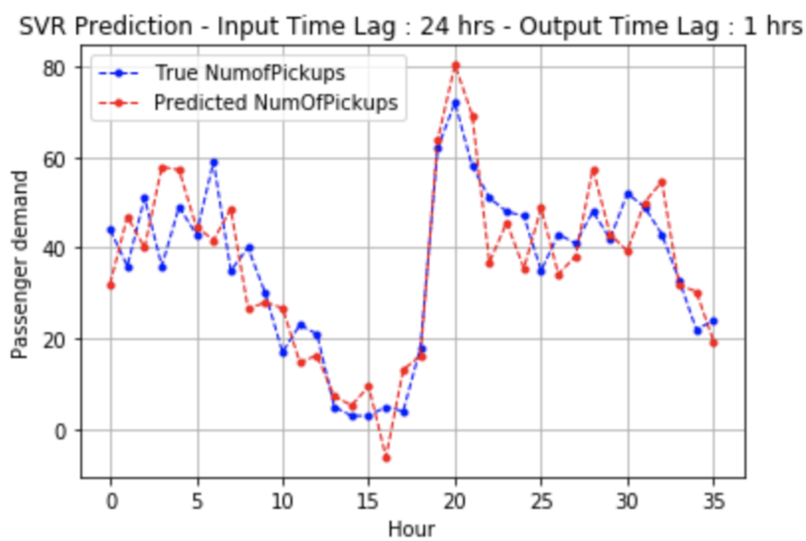

Figure 7: Comparing the actual value and the predicted value of passenger demand using SVR model for New York, January 2018 dataset location Manhattan Central.

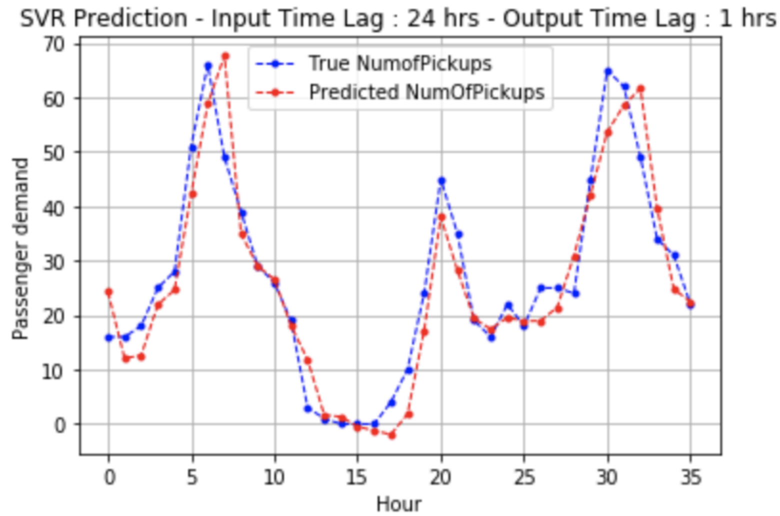

Figure 8: Comparing the actual value and the predicted value of passenger demand using SVR model for New York, January 2018 dataset location Brooklyn Dumbo, Vinegar Hill.

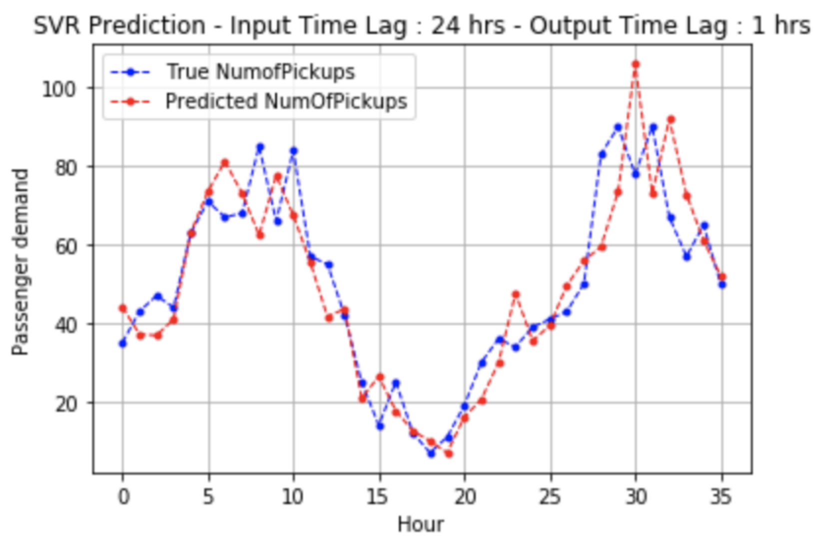

Figure 9: Comparing the actual value and the predicted value of passenger demand using SVR model for New York, January 2018 dataset location Queens Elmhurst.

4.2.3 K Nearest Neighbour

In this section, the KNN algorithm with $k = 5$ neighbors is used as the second prediction method.
We use sklearn.svm package in this experiment. The same training and the testing dataset is considered for both KNN and SVR models. We regarded 80% of the dataset as the training data, and 20% of the dataset for testing. It worth mentioning that we used a grid search for parameter tuning in this case study.
Below table shows the list of tuned hyperparameters for the KNN model.

Hyperparameter	
N_Neighbours	5
Weight	'distance'

Table 4: Tuned hyperparameters for KNN model using grid search

Below graphs show the predicted demand and actual demand for 4 selected regions, using the KNN model.

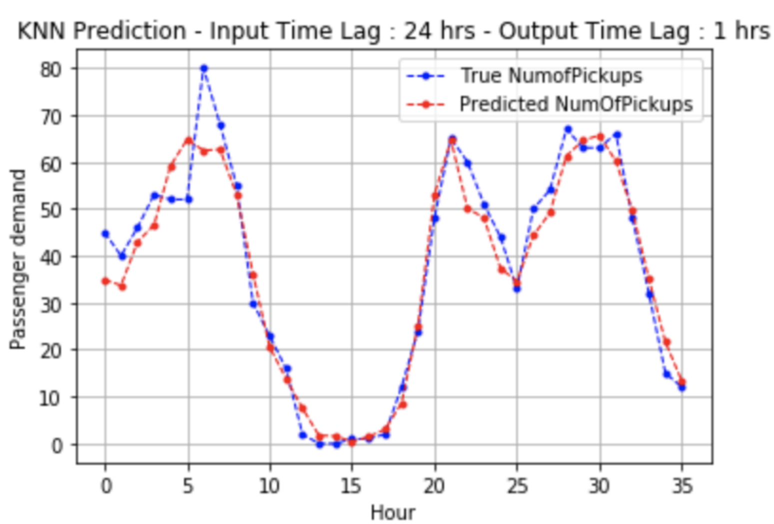

Figure 10: Comparing the actual values and the predicted values of passenger demand using KNN model for New York, January 2018 dataset location Brooklyn Heights.

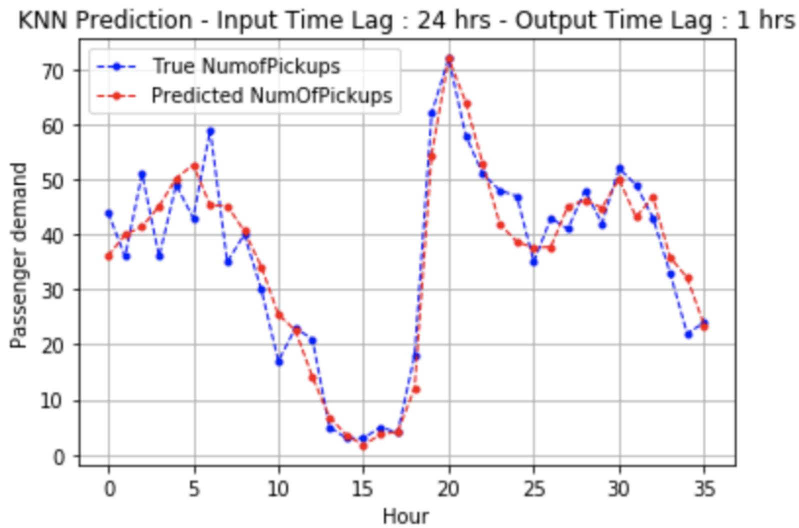

Figure 11: Comparing the actual values and the predicted values of passenger demand using KNN model for New York, January 2018 dataset location Manhattan Central Harlem North.

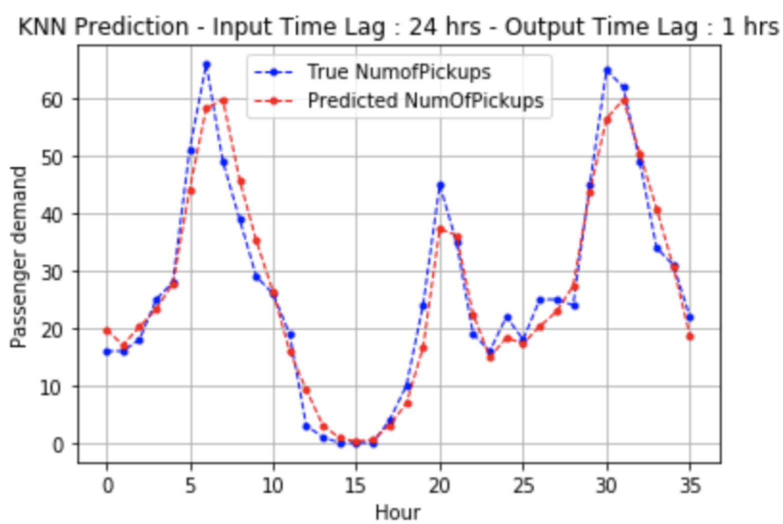

Figure 12: Comparing the actual values and the predicted values of passenger demand using KNN model for New York, January 2018 dataset location Brooklyn Dumbo/Vinegar Hill.

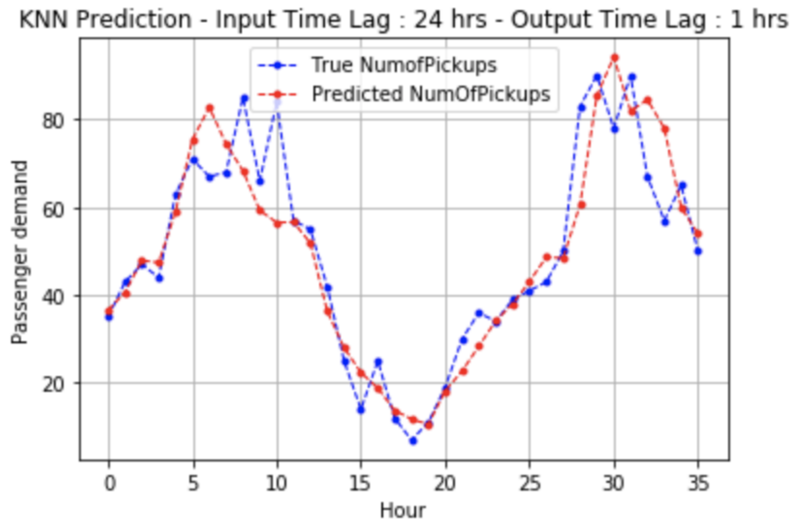

Figure 13: Comparing the actual values and the predicted values of passenger demand using KNN model for New York, January 2018 dataset location Queens Elmhurst.

4.3 Performance Measures for ML models

The prediction results for the dataset are presented in Table 5 demonstrated by a performance metric and the overall accuracy. Based on the results, the SVR has a smaller MSE with slightly higher accuracy. Thus, we can infer that the SVR performs better compared to the KNN for this particular dataset. Therefore, for the MIP optimization, we use the results from SVR demand prediction due to its superiority.

Model	Performance Metric	Accuracy
	MSE	
SVR	0.2066	81.25%
KNN	0.2294	79.86%

Table 5: Comparing the performance of SVR and KNN models for New York City taxi dataset.

4.4 Mixed Integer Programming for Taxi Dispatching Problem

As mentioned earlier, we have a centralized dispatch system which sends location recommendation to taxi drivers. We conduct the taxi dispatch problem, based on the New York taxi dataset for January 2018[2], explained in previous sections. The parameters and variables are defined in table 1. The geographic location (latitude and longitude) of taxis at the beginning of dispatching and predicted requests at each region are two of the input parameters in the taxi dispatching problem under consideration. After predicting the passengers' demands with the help of machine learning techniques, the results can be fed into our MIP model as the parameter; $r_j \in \mathbb{Z}_+$. Since the integer nature of demand, we rounded the predicted results. We implement the dispatch algorithm in Python using the optimization toolbox called CVXPY [1]. CVXPY is a Python-embedded modeling language for convex optimization problems. Inside a target region, we assume that there exists a taxi station in which a vacant taxi picks up passengers in that station. We note that our learning model is not restricted to the dataset used in this thesis, and other models and datasets can also be incorporated.

In our assumptions, we consider each taxi station as a node, and then we calculate the distance between every two nodes using the Manhattan norm. Hence, we consider the path of each taxi as connected road segments determined by every two consecutive points of the trace data we use in this section. Assume the latitude and longitude values of two consecutive points in the trace data are (p_{x1}, p_{y1}) and (p_{x2}, p_{y2}), for a short road segment, the mileage distance between the two points is approximated as being proportional to the value ($\|p_{x1}$-$p_{x2}\|$ + $\|\mathrm{p}_{y1}$-$p_{y2}\|$).

The geometric location of a taxi is directly provided by GPS data, which in our case, is the longitude and latitude of the initial taxi stations which taxis stand in before dispatching.

Based on our dataset, the total number of New York regions with pickups during Jan 2018 is 248. Also, we consider the average hourly pickups in each region and aggregated it in order to have an estimation of the total number of vacant taxis. Generally speaking, it is difficult to estimate the exact fleet size. Realistically this number should be determined by taxi companies as the vehicle resource is limited,

and there is only a certain number of taxis to satisfy the demand. There are 31 days in Jan 2018, and each day is 24 hours time-lag, so in total, we have $31 * 24 = 744$ hours, and by dividing the total number of pickups for each region over 744, we have the average hourly demand for each region. The result of aggregating these average amounts shows the total number of vacant taxis at 989. Also, each taxi has a unique ID number recorded in the dispatch system center.
We also assume that information of passenger's destination is not available to the system when making dispatch decisions since passengers just hail a taxi at taxi stations.

4.4.1 Multi-Objective Optimization

Figure 12 shows a 27% reduction in the supply-demand mismatch error by the solution provided by the taxi dispatch formulation compared with the historical data without allocation. We note that the blue line in Figure 12 is the global supply-demand ratio which is equal to total number of taxis in the city over total demand of the city.

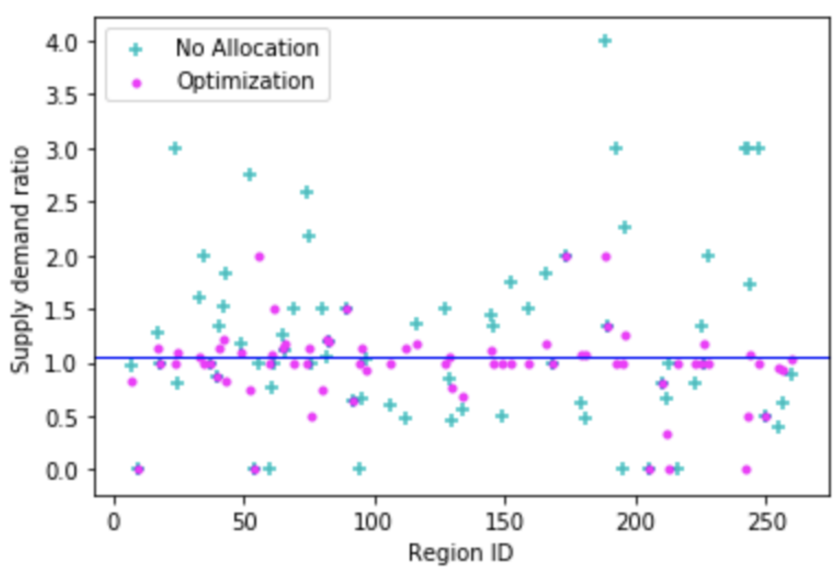

Figure 14: Comparison of supply-demand ratio at each region, for the solution of MIP model and historical data without allocation.

Design Parameters for Algorithms

Parameters like (α), the upper bound of driving distance, and (β) the objective function weight parameter in eq. (8), significantly affect the results and the corresponding optimal cost. Optimal values of parameters can vary from a given dataset to another. Therefore, we perform sensitivity analyses and change the value of a given parameter while keeping other parameters at the same level, and compare results. We compare the outcomes and explain the adjustment of the parameters according to experimental results based on a given historical dataset with GPS records.

The process of choosing values of parameters for Algorithm (8) is a trial and error process, by increasing/decreasing the values of the parameters and observing the resulting change in the dispatch cost until the desired performance is reached or some turning points occur that the cost is not reduced anymore. For instance, the objective weight is related to the objective function of the dispatch system, whether it is more important to reach fair service or reduce total idle distance. In addition, some parameters are related to additional information available to the system. for instance, α, can be adjusted according to the average speed of vehicles or traffic conditions during the considered period.

Sensitivity Analyses on β

The objective function consists of two parts: the idle geographical travel distance (mileage) cost and the supply-demand ratio mismatch cost. This trade-off between the two parts is addressed by β. In other words, the weight of idle distance increases with β. A larger β returns a solution with smaller total idle geographical distance while a larger error between the supply-demand ratio. The two components of the cost with different β by algorithm1 and historical data without algorithm (8) are shown in the below table.

β	0	2	10
Supply demand ratio error	0.005	0.022	0.020
Idle distance	98.9	10.72	10.723
Total cost	0.005	21.46	107.255

Table 6: Average cost comparison for different values of β

We calculate the total cost as (s/d error $+\beta\times$ idle distance) . Although with $\beta = 0$ we can dispatch vacant taxis to make the supply-demand ratio of each region closest to that of the whole city, a larger idle geographical distance cost is introduced compared with $\beta = 2$. Comparing the supply-demand ratio when $\beta = 0$ with $\beta = 2$ we have 77% increase, while the ratio error does not show a significant difference increasing β to 10.

Determining Ideal Distance Threshold

Figure 13 compares the error between local supply-demand ratio and global supply-demand ratio for different values of α. Since we directly use geographical distance measured by the difference between longitude and latitude values of two points (GPS locations) on the map, the threshold value is small: 0.1 degree, which is equal to 7 miles distance on the ground. We consider one degree of latitude and latitude equal to approximately 70 miles, which means 0.1 difference in GPS data corresponds to almost 7 miles distance on the ground. It means the upper bound driving distance for each taxi to respond to the passenger demand is 7 miles. When this parameter increases, the error between the local supply-demand ratio and global supply-demand ratio decreases, because vacant taxis are more flexible to traverse further to meet demand at farther regions.

For instance, when the length of the time slot is one hour, and α is the distance a taxi can traverse during 20 minutes of that time slot, this constraint means a dispatch solution involves the requirement that a taxi should be able to arrive the dispatched position within 20 minutes in order to fulfill predicted requests.

This parameter can be adjusted according to the travel speed information available for the dispatch system. This constraint also enables the dispatch system to consider the fact that drivers may be reluctant to drive idly for a long distance to serve potential customers. The threshold α is related to the length of a time slot. In general, the longer a time slot is, the larger α can be, because of constraints like a speed limit.

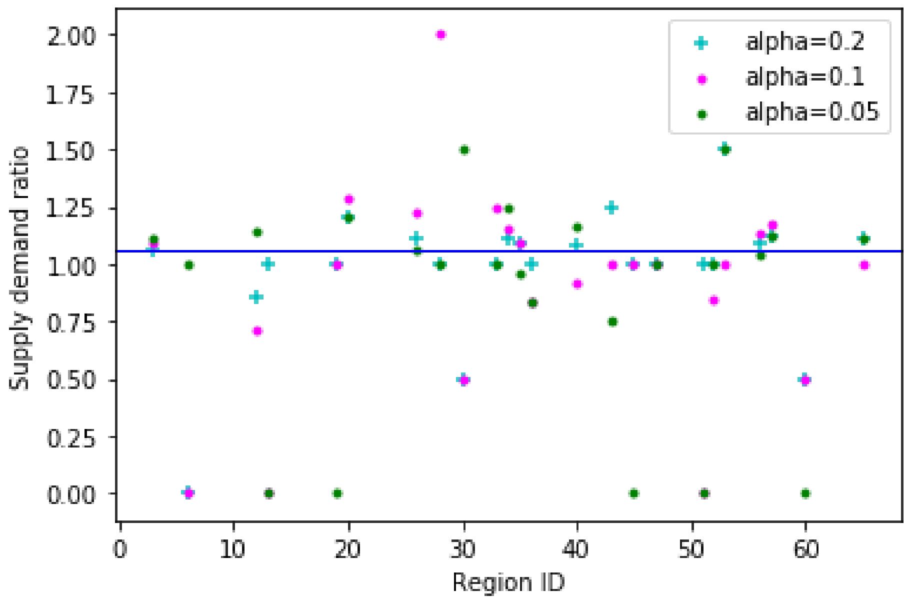

Figure 15: Comparison of supply-demand ratios at each region during one-time slot for different values of α.

4.4.2 Single-Objective Optimization

In the single-objective dispatch formulation, the cost function includes the idle geographical travel distance (mileage) cost. In this case, the supply-demand ratio mismatch is added as the constraint (9a). A larger error e returns a solution with smaller total idle geographical distance. This is because when we want to make the mismatch between supply-demand ratio in different districts smaller, taxis need to drive longer distances in order to respond to passenger demands at regions with lower predicted demands. The total cost (total idle geographical travel distance) with different values of e are shown in below table:

e	0.05	0.02	0.01	0.001	Without dispatch
Total cost	10.72	10.78	11.26	15.70	25.51

Table 7: Total cost comparison for different values of e.

As it becomes disclosed from the above table, by altering the amount of e, the total cost is increased by 57% compared to no allocation. We note that by "without dispatch" we mean that we calculated the average hourly demand for each region based on the historical data and estimated the number of required taxis in each region.

4.5 The Impact of Prediction on the Solution Quality

In order to demonstrate the impact of prediction of final solution quality, we compared the results of our data-driven optimization framework using SVR with optimization using the historical average. As we can see in the below table, using SVR in the objective function, which minimizes the total idle driving distance at a supply-demand ratio error of 5%, has reduced the total cost by 14% comparing to the historical average. We note that the unit of the data provided in Table 8 is degree of latitude which is approximately equal to 69 miles.

	Total idle driving distance
Historical Average	14.49
SVR	10.72

Table 8: Total idle driving distance comparison using SVR and historical average, e=0.05.

Chapter 5

Conclusion and Directions for Future Studies

Taxi service has become one of the most important means of transportation in the world. Optimization of the taxi service can significantly reduce transportation costs, traffic, greenhouse gas emissions, and increase service quality. However, optimization of the taxi service due to its specific characteristics is a cumbersome task. The existing research in the literature uses mathematical programming or machine learning approaches to handle the problem. The results of the approaches mentioned above can be significantly improved by hybridizing the two methodologies. In this research, we studied the taxi dispatching problem and proposed a mathematical programming machine learning-based approach to optimize the network. We presented a data-driven optimization methodology by combining machine learning techniques, that incorporate historical time-series data to forecast future demand, and mathematical programming. Moreover, we aimed at balancing supply according to the demand in different regions (nodes) of a city in order to increase service efficiency and to minimize the total ideal driving distance.

We proposed a method that utilizes historical GPS data to build demand models and applies prediction technologies to determine optimal locations for vacant taxis considering anticipated future demand. From a system-level perspective, we compute optimal dispatch solutions for reaching a globally balanced supply-demand ratio with the least associated cruising distance under practical constraints. We implemented our approach to a real-world case study from New York City. The results revealed that our proposed method could remarkably improve the performance of the taxi system by reducing total idle distance and increasing service quality. Evaluation results disclosed significant performance improvements in the network which linked to reducing the demand-supply ratio mismatch error by 27%.

While our framework focuses on the planning of the distribution of vehicles, the framework does not address routing decisions. Therefore, considering routing in the proposed model can significantly increase its applicability. Also, our work focuses on historical data as a proactive planning framework, and the framework can be expanded to a real-time estimator as a practical implementation. In that case, the framework requires a centralized authority to collect all available information and make decisions for every vehicle in real-time. Our allocation time horizon only supports one time lag as the dispatch system does not control the drop off location. Therefore, in future work, it would be interesting to take into account several time lags by considering a mobility pattern function. Last but not least, considering the traffic and road conditions can be other concerns in the future works.

Bibliography

[1] *CVXPY Convex optimization.* Available at `https://www.cvxpy.org`,.

[2] *NYC Trip Record Data.* Available at `https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page`,.

[3] Abubakr O Al-Abbasi, Arnob Ghosh, and Vaneet Aggarwal. Deeppool: Distributed model-free algorithm for ride-sharing using deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 20(12):4714–4727, 2019.

[4] Ruibin Bai, Jiawei Li, Jason AD Atkin, and Graham Kendall. A novel approach to independent taxi scheduling problem based on stable matching. *Journal of the Operational Research Society*, 65(10):1501–1510, 2014.

[5] Tianfeng Chai and Roland R Draxler. Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature. *Geoscientific model development*, 7(3):1247–1250, 2014.

[6] Qian Chen, Wenquan Li, and Jinhuan Zhao. The use of ls-svm for short-term passenger flow prediction. *Transport*, 26(1):5–10, 2011.

[7] Ximing Chen, Fei Miao, George J Pappas, and Victor Preciado. Hierarchical data-driven vehicle dispatch and ride-sharing. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 4458–4463. IEEE, 2017.

[8] Shih-Fen Cheng and Xin Qu. A service choice model for optimizing taxi service delivery. In *2009 12th International IEEE Conference on Intelligent Transportation Systems*, pages 1–6. IEEE, 2009.

[9] Fai Cheong Choo, Mun Choon Chan, Akhihebbal L Ananda, and Li-Shiuan Peh. A distributed taxi advisory system. In *2012 12th International Conference on ITS Telecommunications*, pages 199–204. IEEE, 2012.

[10] Leyi Duan, Yuguang Wei, Jinchuan Zhang, and Yang Xia. Centralized and decentralized autonomous dispatching strategy for dynamic autonomous taxi operation in hybrid request mode. *Transportation Research Part C: Emerging Technologies*, 111:397–420, 2020.

[11] Guoju Gao, Mingjun Xiao, and Zhenhua Zhao. Optimal multi-taxi dispatch for mobile taxi-hailing systems. In *2016 45th International Conference on Parallel Processing (ICPP)*, pages 294–303. IEEE, 2016.

[12] Steve R Gunn et al. Support vector machines for classification and regression. *ISIS technical report*, 14(1):5–16, 1998.

[13] Xingxing Hu, Qimei Cui, and Kwang-Cheng Chen. Temporal-spatial prediction of trip demand using neural networks and points of interest. In *2019 11th International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1–6. IEEE, 2019.

[14] Arief Koesdwiady, Alaa El Khatib, and Fakhri Karray. Methods to improve multi-step time series prediction. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2018.

[15] Pantea Koochemeshkian, Nuha Zamzami, and Nizar Bouguila. Flexible distribution-based regression models for count data: Application to medical diagnosis. *Cybernetics and Systems*, pages 1–25, 2020.

[16] Der-Horng Lee and Xian Wu. Dispatching strategies for the taxi-customer searching problem in the booking taxi service. In *Transportation Research Board 92nd Annual Meeting*, number 13–1975, 2013.

[17] John D Lees-Miller and R Eddie Wilson. Proactive empty vehicle redistribution for personal rapid transit and taxis. *Transportation Planning and Technology*, 35(1):17–30, 2012.

[18] Xiaolong Li, Gang Pan, Zhaohui Wu, Guande Qi, Shijian Li, Daqing Zhang, Wangsheng Zhang, and Zonghui Wang. Prediction of urban human mobility using large-scale taxi traces and its applications. *Frontiers of Computer Science*, 6(1):111–121, 2012.

[19] Lu Ling, Xiongfei Lai, and Li Feng. Forecasting the gap between demand and supply of e-hailing vehicle in large scale of network based on two-stage model. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3880–3885. IEEE, 2019.

[20] Lu Ling, Xiongfei Lai, and Li Feng. Forecasting the gap between demand and supply of e-hailing vehicle in large scale of network based on two-stage model. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3880–3885. IEEE, 2019.

[21] Qian Liu, Chuan Ding, and Peng Chen. A panel analysis of the effect of the urban environment on the spatiotemporal pattern of taxi demand. *Travel Behaviour and Society*, 18:29–36, 2020.

[22] Alvaro Lorca and Xu Andy Sun. Adaptive robust optimization with dynamic uncertainty sets for multi-period economic dispatch under significant wind. *IEEE Transactions on Power Systems*, 30(4):1702–1713, 2014.

[23] Meghna Lowalekar, Pradeep Varakantham, and Patrick Jaillet. Online spatio-temporal matching in stochastic and dynamic domains. *Artificial Intelligence*, 261:71–112, 2018.

[24] Deepesh Machiwal and Madan Kumar Jha. *Hydrologic time series analysis: theory and practice*. Springer Science & Business Media, 2012.

[25] Francisco Martínez, María Pilar Frías, María Dolores Pérez, and Antonio Jesús Rivera. A methodology for applying k-nearest neighbor to time series forecasting. *Artificial Intelligence Review*, 52(3), 2019.

[26] Fei Miao, Shuo Han, Shan Lin, John A Stankovic, Desheng Zhang, Sirajum Munir, Hua Huang, Tian He, and George J Pappas. Taxi dispatch with real-time sensing data in metropolitan areas: A receding horizon control approach. *IEEE Transactions on Automation Science and Engineering*, 13(2):463–478, 2016.

[27] Luis Moreira-Matias, João Gama, Michel Ferreira, and Luís Damas. A predictive model for the passenger demand on a taxi network. In *2012 15th International IEEE Conference on Intelligent Transportation Systems*, pages 1014–1019. IEEE, 2012.

[28] Abood Mourad, Jakob Puchinger, and Chengbin Chu. A survey of models and algorithms for optimizing shared mobility. *Transportation Research Part B: Methodological*, 2019.

[29] Santhanakrishnan Narayanan, Emmanouil Chaniotakis, and Constantinos Antoniou. Shared autonomous vehicle services: A comprehensive review. *Transportation Research Part C: Emerging Technologies*, 111:255–293, 2020.

[30] Antonio Rafael Sabino Parmezan and Gustavo EAPA Batista. A study of the use of complexity measures in the similarity search process adopted by knn algorithm for time series prediction. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pages 45–51. IEEE, 2015.

[31] Marco Pavone, Stephen L Smith, Emilio Frazzoli, and Daniela Rus. Robotic load balancing for mobility-on-demand systems. *The International Journal of Robotics Research*, 31(7):839–854, 2012.

[32] Santi Phithakkitnukoon, Marco Veloso, Carlos Bento, Assaf Biderman, and Carlo Ratti. Taxi-aware map: Identifying and predicting vacant taxis in the city. In *International Joint Conference on Ambient Intelligence*, pages 86–95. Springer, 2010.

[33] Jason W Powell, Yan Huang, Favyen Bastani, and Minhe Ji. Towards reducing taxicab cruising time using spatio-temporal profitability maps. In *International Symposium on Spatial and Temporal Databases*, pages 242–260. Springer, 2011.

[34] Abolfazl Safikhani, Camille Kamga, Sandeep Mudigonda, Sabiheh Sadat Faghih, and Bahman Moghimi. Spatio-temporal modeling of yellow taxi demands in new york city using generalized star models. *International Journal of Forecasting*, 2018.

[35] Wen Shi, CO Tong, and SC Wong. Influence of real-time information provision to vacant taxi drivers on taxi system performance. In *World Conference of Transport Research*, 2010.

[36] Zhenyu Shou, Xuan Di, Jieping Ye, Hongtu Zhu, Hua Zhang, and Robert Hampshire. Optimal passenger-seeking policies on e-hailing platforms using markov decision process and imitation learning. *Transportation Research Part C: Emerging Technologies*, 111:91–113, 2020.

[37] Xin Situ, Wei-Neng Chen, Yue-Jiao Gong, Ying Lin, Wei-Jie Yu, Zhiwen Yu, and Jun Zhang. A parallel ant colony system based on region decomposition for taxi-passenger matching. In *2017 IEEE Congress on Evolutionary Computation (CEC)*, pages 960–967. IEEE, 2017.

[38] Kevin Spieser, Kyle Treleaven, Rick Zhang, Emilio Frazzoli, Daniel Morton, and Marco Pavone. Toward a systematic approach to the design and evaluation of automated mobility-on-demand systems: A case study in singapore. In *Road vehicle automation*, pages 229–245. Springer, 2014.

[39] Prabhanjan Tattar, Tony Ojeda, Sean Patrick Murphy, Benjamin Bengfort, and Abhijit Dasgupta. *Practical Data Science Cookbook*. Packt Publishing Ltd, 2017.

[40] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[41] Hai Wang and Hai Yang. Ridesourcing systems: A framework and review. *Transportation Research Part B: Methodological*, 129:122–155, 2019.

[42] Peng Xie, Tianrui Li, Jia Liu, Shengdong Du, Xin Yang, and Junbo Zhang. Urban flow prediction from spatiotemporal data using machine learning: A survey. *Information Fusion*, 2020.

[43] Hai Yang, Cowina WY Leung, Sze Chun Wong, and Michael GH Bell. Equilibria of bilateral taxi–customer searching and meeting on networks. *Transportation Research Part B: Methodological*, 44(8-9):1067–1083, 2010.

[44] Kiyoung Yang and Cyrus Shahabi. A pca-based similarity measure for multivariate time series. In *Proceedings of the 2nd ACM international workshop on Multimedia databases*, pages 65–74, 2004.

[45] Bin Yu, Xiaolin Song, Feng Guan, Zhiming Yang, and Baozhen Yao. k-nearest neighbor model for multiple-time-step prediction of short-term traffic condition. *Journal of Transportation Engineering*, 142(6):04016018, 2016.

[46] Jing Yuan, Yu Zheng, Liuhang Zhang, XIng Xie, and Guangzhong Sun. Where to find my next passenger. In *Proceedings of the 13th international conference on Ubiquitous computing*, pages 109–118, 2011.

[47] Liu Yue, Yin Yafeng, Gao Junjun, and Tan Chongli. Demand forecasting by using support vector machine. In *Third International Conference on Natural Computation (ICNC 2007)*, volume 3, pages 272–276. IEEE, 2007.

[48] Rick Zhang and Marco Pavone. Control of robotic mobility-on-demand systems: a queueing-theoretical perspective. *The International Journal of Robotics Research*, 35(1-3):186–203, 2016.

[49] Rick Zhang and Marco Pavone. Control of robotic mobility-on-demand systems: a queueing-theoretical perspective. *The International Journal of Robotics Research*, 35(1-3):186–203, 2016.

[50] Rick Zhang and Marco Pavone. Control of robotic mobility-on-demand systems: a queueing-theoretical perspective. *The International Journal of Robotics Research*, 35(1-3):186–203, 2016.

[51] Kai Zhao, Denis Khryashchev, and Huy Vo. Predicting taxi and uber demand in cities: Approaching the limit of predictability. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[52] Huanyang Zheng and Jie Wu. Online to offline business: urban taxi dispatching with passenger-driver matching stability. In *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pages 816–825. IEEE, 2017.

[53] Yirong Zhou, Jun Li, Hao Chen, Ye Wu, Jiangjiang Wu, and Luo Chen. A spatiotemporal attention mechanism-based model for multi-step citywide passenger demand prediction. *Information Sciences*, 513:372–385, 2020.