Learning-based Methods for Optimising Shared Mobility Systems with Multimodal Data

Sen Yan

BEng, MSc

Supervisors:

Dr Mingming Liu

Prof Noel E. O'Connor

A Dissertation submitted in fulfilment of the requirements

for the award of Doctor of Philosophy (Ph.D.)

School of Electronic Engineering

Dublin City University

August 2025

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy s entirely my own work, and that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: Sen Yan **ID No.:** 21269571 **Date:** 5th August 2025

Acknowledgements

I would like to express my sincere gratitude to all those who have supported me throughout my research and the writing of this thesis.

I am especially grateful to my supervisory panel, including my supervisors, Dr. Mingming Liu and Prof. Noel E. O'Connor, and independent advisors, Dr. Kevin McGuinness and Prof. Xiaojun Wang, for their continuous guidance, encouragement, and invaluable feedback. Their advice and encouragement helped me stay focused and move forward, especially during times when progress felt uncertain.

I would also like to thank the examiner of my transfer report, Prof. Suzanne Little, and the panel of my viva examination, Prof. Shirley Coyle, Prof. Ivana Dusparic, and Dr. Robert Sadleir, for their thoughtful and constructive suggestions, which helped me strengthen the direction of my research at a key stage.

Many thanks also to my collaborators, Prof. David J. O'Connor, Prof. Alan F. Smeaton, Prof. Tomas E. Ward, Prof. Ji Li, Dr. Eric Arazo Sánchez, and Dr. Jaime B. Fernandez. I truly appreciate the opportunity to work with them and have learned a lot through our collaborations.

I am also thankful to Insight Research Ireland Centre for Data Analytics and Dublin City University for providing the environment and resources that made this research possible.

Finally, I am deeply thankful to my family and friends for their constant support, patience, and understanding. Their encouragement has been a steady source of motivation throughout this journey.

List of Publications

** Equal contribution*

Published

- Sen Yan, Mingming Liu, and Noel E. O'Connor. "Parking behaviour analysis of shared e-bike users based on a real-world dataset - a case study in dublin, ireland," in *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, IEEE, Jun. 2022, pp. 1-6. [**Conference**]

- Sen Yan, Shaoshu Zhu, Jaime B. Fernandez, Eric Arazo Sánchez, Yingqi Gu, Noel E. O'Connor, David O'Connor, and Mingming Liu. "Breathing green: Maximising health and environmental benefits for active transportation users leveraging large scale air quality data," in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Sept. 2023, pp. 5496-5503. [**Conference**]

- Sen Yan, Maqsood Hussain Shah, Ji Li, Noel O'Connor, and Mingming Liu. "A review on ai algorithms for energy management in e-mobility services," in *2023 7th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, IEEE, Oct. 2023, pp. 1-8. [**Conference**]

- Sen Yan, Hongyuan Fang, Ji Li, Tomas Ward, Noel O'Connor, and Mingming Liu. "Privacy-aware energy consumption modeling of connected battery electric vehicles using federated learning," *IEEE Transactions on Transportation Electrification*, vol. 10, no. 3, pp. 6663-6675, Sep. 2024, issn: 2372-2088. [**Journal**]

- Yue Ding*, Sen Yan*, Maqsood Hussain Shah, Hongyuan Fang, Ji Li, and Mingming Liu. "Data-driven energy consumption modelling for electric micromobility using an open dataset," in *2024 IEEE Transportation Electrification Conference and Expo (ITEC)*, IEEE, Jun. 2024, pp. 1-7. [**Conference**]

- Sen Yan, Noel E. O'Connor, and Mingming Liu. "U-park: A user-centric smart parking recommendation system for electric shared micromobility services," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 10, pp. 5179-5193, 2024. [**Journal**]

- Sen Yan, David J. O'Connor, Xiaojun Wang, Noel E. O'Connor, Alan. F. Smeaton, and Mingming Liu. "Comparative analysis of machine learning-based imputation techniques for air quality datasets with high missing data rates. In *2025 IEEE Symposia on Computational Intelligence for Energy, Transport and Environmental Sustainability (CIETES)*, IEEE, Mar. 2025, pp. 1-8. [**Conference**]

Accepted

- Hongde Wu*, Sen Yan*, and Mingming Liu, "Recent advances in graph-based machine learning for applications in smart urban transportation systems," in *Next-Generation Cities: An Encyclopedia*, vol. 3, pt. 3, ch. 4, Singapore: World Scientific Publishing, 2026, to appear. ISBN: 978-981-12-8950-7. [**Book Chapter**]

Submitted

- Sen Yan, Chinmaya Kaundanya, Noel E. O'Connor, Suzanne Little, and Mingming Liu, "Machine learning in micromobility: A systematic review of datasets, techniques, and applications," submitted to *IEEE Transactions on Intelligent Vehicles*, Jan. 2025. [**Journal**]

Contents

List of Figures

List of Tables

List of Abbreviations & Acronyms

ALSTM	Attention-based Long Short-Term Memory
ANN	Artificial Neural Networks
API	Application Programming Interface
ARIMA	Auto-Regressive Integrated Moving Average
ARNN	Attention-based Recurrent Neural Networks
ASTGCN	Attention-based Spatio-Temporal Graph Convolutional Network
BEVs	Battery Electric Vehicles
CCE	Categorical Cross Entropy
CNN	Convolutional Neural Networks
CO	Carbon Monoxide
CO_2	Carbon Dioxide
Conv-LSTM	Convolutional Long Short-Term Memory
DCC	Dublin City Council
DL	Deep Learning
DPD	Dynamic Parcel Distribution
DPNst	Destination Prediction Network based on Spatio-Temporal data
DT	Decision Tree
E-Bikes	Electric Bikes
E-Scooters	Electric Scooters
EPA	Environmental Protection Agency
EVs	Electric Vehicles
FedAvg	Federated Averaging
FedPer	Federated Personalisation
FedProx	Federated Proximal

FedRep	Federated Representation Learning
FedSGD	Federated Stochastic Gradient Descent
FL	Federated Learning
GCN	Graph Convolutional Network
GDPR	General Data Protection Regulation
GNN	Graph Neural Network
GPS	Global Positioning System
GRU	Gated Recurrent Unit
GTWR	Geographically and Temporally Weighted Regression
GWR	Geographically Weighted Regression
IDW	Inverse Distance Weighting
IoT	Internet of Things
KLD	Kullback-Leibler Divergence
KNN	K-Nearest Neighbours
LGBM	Light Gradient-Boosting Machine
LR	Linear Regression
LSTM	Long Short-Term Memory
MaaS	Mobility as a Service
MAE	Mean Absolute Error
MDLN	Multi-module Deep Learning Network
MI	Mutual Information
ML	Machine Learning
MLP	Multilayer Perceptron
MLR	Multiple Linear Regression
MSE	Mean Square Error
NB	Naive Bayesian
NO	Nitric Oxide
NO_2	Nitrogen Dioxide
NO_x	Nitrogen Oxides

O_3	Ozone
OLSR	Ordinary Least-Squares Regression
PM	Particulate Matter
PM_{10}	Particulate Matter 10
PM_1	Particulate Matter 1
$PM_{2.5}$	Particulate Matter 2.5
QR	Quick Response
RF	Random Forest
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
RQ	Research Question
SBC	Single-Board Computer
SMOTE	Synthetic Minority Over-sampling Technique
SMS	Shared Mobility Systems
SO_2	Sulphur Dioxide
SOC	State of Charge
SR	Spatial Regression
STGCN	Spatio-Temporal Graph Convolutional Network
SVM	Support Vector Machine
TCN	Temporal Convolutional Network
UID	Unique Identification Number
VED	Vehicle Energy Dataset
XGB	eXtreme Gradient Boosting

Abstract

Learning-based Methods for Optimising Shared Mobility Systems with Multimodal Data

Sen Yan

This thesis explores the use of learning-based methods in Shared Mobility Systems (SMS), utilising multimodal data to address three key operational challenges: improper parking behaviour, energy consumption prediction, and pollution-aware routing. The overarching goal is to improve the efficiency, sustainability, and user experience of SMS through data-driven, task-specific solutions.

The first challenge is addressed by developing U-Park, a user-centric parking recommendation system. It predicts trip destinations and parking availability in real time using multimodal inputs, including partial trip data, GPS trajectories, and environmental features. Combining an attention-based RNN and a contextualised parking model, U-Park improves the chances of finding available parking by up to 29.66%.

The second contribution focuses on privacy-aware energy consumption modelling for shared battery electric vehicles. A Federated Learning (FL) framework enables model training across distributed data sources without sharing raw data. FL algorithms and local models are evaluated on multimodal features such as speed, altitude, and derived variables. The proposed FedAvg-LSTM model reduces mean absolute error by up to 67.84% and supports deployment in edge-cloud environments.

For the third challenge, a pollution-aware route planning system is introduced. Multimodal data from fixed and mobile air quality sensors is used to construct a high-resolution PM2.5 map, combining temporal imputation with spatial interpolation. Models including IDW, RF, LSTM, and Conv-LSTM are evaluated for short-term forecasting. The resulting pollutant maps inform route selection, reducing average exposure by 25.88% with minimal extra travel distance.

These contributions highlight the value of integrating multimodal data and adopting tailored learning approaches. The thesis also discusses challenges such as data sparsity, integration uncertainty, and model explainability, and outlines future directions including ensemble learning, uncertainty-aware modelling, and multi-objective optimisation.

Chapter 1

Introduction

This chapter presents the motivation behind the research reported in this thesis, defines the research objectives, summarises the main contributions, and introduces the overall structure of the thesis.

1.1 Overview & Motivations

Urban mobility has always been fundamental in shaping how people live, access opportunities, and engage in social activities within modern cities. As urbanisation and economies expand globally, transportation demand has risen sharply, supporting urban growth but also introducing new pressures on infrastructure and the environment. While increased mobility has enabled social and economic development, it has also enhanced the reliance on private vehicles and strained public transportation systems, potentially leading to more traffic congestion, greenhouse gas emissions, and air pollution [1], [2]. In response, some conventional strategies have been adopted such as road expansion or increasing fleet sizes to meet capacity needs. However, these approaches are often reactive and unsustainable. An overemphasis on supply-side measures may reinforce car dependence and cause environmental degradation, especially in highly populated areas [3], [4].

To address these challenges in urban mobility, Shared Mobility Systems (SMS) have emerged as sustainable alternatives to private car ownership. In this thesis,

SMS stand for technology-enabled systems that support the shared use of both micromobility[1], e.g., bicycles, Electric Bikes (E-Bikes), Electric Scooters (E-Scooters), and vehicle-based modes, e.g., Electric Vehicles (EVs). Figure 1.1 illustrates an overview of SMS, capturing the service providers, customers, and the common modes of transport, along with major operational challenges such as facility location, routing, and device assignment. The lower part of the diagram outlines a typical trip process, starting from device selection and route planning regarding different constraints (e.g., energy anxiety and health concerns) to arrival and parking, reflecting the decision points users and systems must address throughout a shared mobility journey.

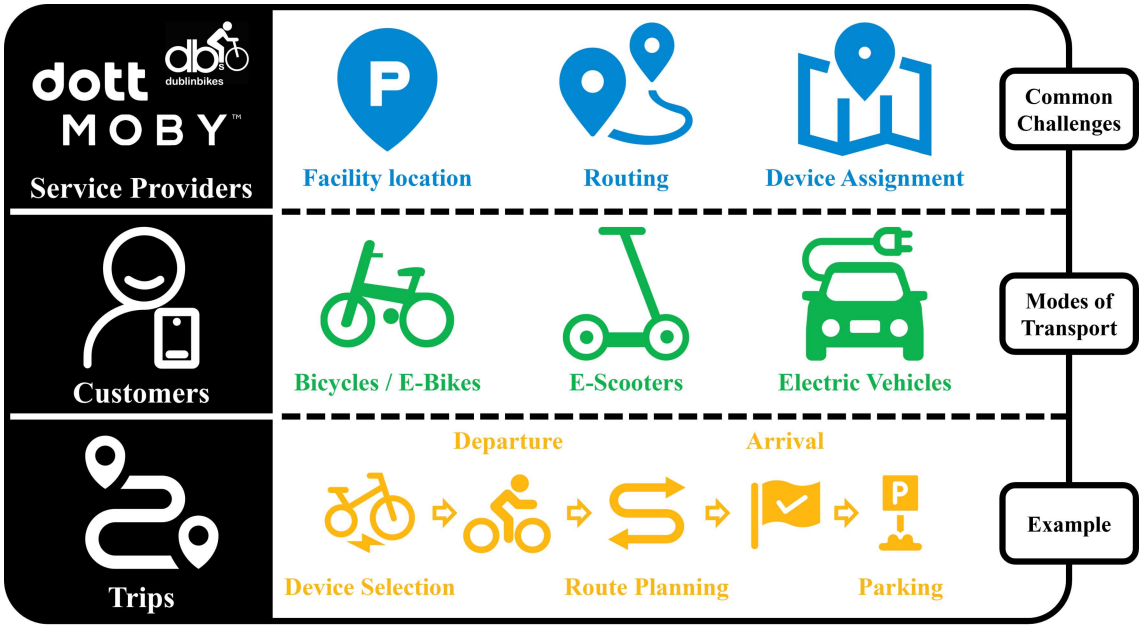


Figure 1.1: An overview of shared mobility systems.

Based on shared economy principles, SMS aim to reduce congestion, emissions, and resource consumption by increasing vehicle utilisation and decreasing reliance on privately owned vehicles [5], [6]. Nowadays, several cities worldwide, including Dublin [7], Paris [8], and Beijing [9], have already introduced large-scale programmes shared bicycle and E-Scooters, which have led to significant improvements in transport sustainability and flexibility. However, the effectiveness of SMS in real-world scenarios remains constrained by several operational challenges. Specifically, issues

[1] https://www.gov.ie/en/department-of-transport/publications/advice-note-for-local-authorities-shared-micromobility-services

such as uneven vehicle distribution, limited availability, inadequate infrastructure, unreliable parking behaviour, uncertain energy demands in electric fleets, and exposure to urban air pollution continue to hinder broader adoption and impact [10], [11], [12].

Addressing these operational challenges requires more effective coordination and management of existing infrastructure, resources, and user demands. Recently, multimodal data, describing the same world context through multiple complementary perspectives, has attracted more attention as a considerable resource to enhance the operational performance and sustainability of SMS [13], [14]. Within urban mobility, multimodal data typically includes heterogeneous and interconnected data sources, such as meteorological conditions [15], traffic flows information [16], air pollution data [17], user travel patterns [18], and Global Positioning System (GPS) trajectories [19]. Compared with single-source or unimodal data, multimodal datasets provide richer, more detailed, and contextually informed insights, improving predictive accuracy and supporting more responsive system-level decisions [13], [20].

However, the practical use of multimodal data remains challenging mainly due to inherent issues such as data sparsity, heterogeneity, and privacy constraints. For instance, real-world SMS data often contains significant missing values caused by sensor malfunctions, transmission delays, or simply the absence of observations at particular locations or times [21], [22]. In addition, multimodal datasets often exhibit complex cross-modal correlations and non-linear dependencies that conventional modelling approaches struggle to capture effectively. Traditional analytical methods, such as statistical regression or heuristic rules, typically rely on data completeness or linear dependencies, limiting their applicability in real-world multimodal settings [23], [24]. These limitations emphasise the need for advanced modelling techniques addressing these challenges using multimodal data more efficiently.

Learning-based methods, such as Machine Learning (ML) and Deep Learning (DL) techniques, have emerged as effective tools for multimodal data processing. Compared with traditional methods, learning-based models offer clear advan-

tages in capturing non-linear, dynamic relationships, extracting useful features automatically, and handling missing values through robust data imputation or prediction techniques [25], [26], [27]. These models has already been employed in many transportation-related applications, such as traffic management and vehicle energy management [28], [29]. Additionally, advanced ML paradigms such as transfer learning and ensemble learning have also demonstrated effectiveness in SMS based on previously learned knowledge or the combination of multiple predictive models to further improve accuracy and robustness to address data sparsity and heterogeneity conditions [27], [30]. Meanwhile, Federated Learning (FL) approaches has gained attention as promising solutions to address privacy concerns, enabling multiple stakeholders to collaborate efficiently without sharing private data directly [31]. These techniques present new opportunities for the effective use and secure integration of multimodal data in SMS.

Applying learning-based methods together with multimodal data in SMS presents strong potential to enhance system optimisation, operational efficiency, and user experience. For example, a recommendation system based on accurate prediction results of parking availability and user destination may help reduce vehicle misplacement, improve fleet utilisation, and ease congestion near high-demand docking areas [32]. Similarly, modelling the energy consumption of shared EVs using FL techniques can effectively reduce range anxiety (the concern that the battery may not last for the entire trip), optimise charging strategies, and support the expansion of electric mobility services, such as Battery Electric Vehicles (BEVs), E-Bikes, and E-Scooters, while maintaining user privacy [33]. Additionally, integrating short-term air pollution forecasts with route planning systems can lower health risks for vulnerable road users, such as cyclists and pedestrians, by recommending routes with lower pollutant exposure [17].

In summary, as global urban planning centres on building smarter, greener, and more resilient transport systems, addressing the operational challenges facing shared mobility has become gradually important. Applying advanced learning-based meth-

ods together with multimodal data provides a practical and promising avenue towards this goal, supporting more intelligent, reliable, and context-aware solutions for SMS. Motivated by the potential of this powerful combination, this thesis explores informed decision-making and optimisation strategies aimed at enhancing the operational efficiency and sustainability of SMS. Specifically, this thesis focuses on three typical and interrelated challenges in urban SMS:

(1) Inefficient parking behaviour, which limits service availability and increases user inconvenience;

(2) Energy consumption anxiety within shared EVs, which can constrain service scalability and weaken user confidence;

(3) Pollutant exposure risks, especially affecting vulnerable SMS users such as cyclists and pedestrians, which affects health impact of mobility decisions.

Jointly, these selected topics address distinct but interrelated aspects of system optimisation, illustrating how effectively learning-based approaches and multimodal data integration can drive improvements across diverse shared mobility scenarios.

1.2 Research Objectives & Contributions

As introduced above, this thesis aims to improve SMS by addressing three challenges using learning-based methods and multimodal data. Each challenge is discussed and solved in its corresponding case study in the following chapters. Accordingly, we briefly summarise the research objectives and contributions of each chapter below.

1.2.1 A User-Centric Parking Recommendation System

To address the challenge of parking management in SMS, this work presents the design of a real-time user-centric smart parking recommendation system based on multimodal data and learning based models. Three Research Question (RQ) are considered in his study:

- **RQ1:** Can the destination of SMS users be accurately predicted in real time based on historical travel records and partial GPS trajectories?
- **RQ2:** How can multimodal data (e.g., weather conditions) improve the prediction of parking space availability near the targeted destination?
- **RQ3:** Can a personalised parking recommendation system reduce the likelihood of improper parking and improve the overall user experience in SMS?

We hypothesise that integrating multimodal data into a learning-based recommendation pipeline can significantly improve the accuracy and relevance of parking suggestions, which in turn reduces improper parking behaviours and enables proactive user guidance throughout the entire journey. Taking it into consideration, our work explores three key objectives:

(1) To predict trip destinations during an ongoing trip by analysing historical mobility patterns and partial GPS trajectories;

(2) To predict the parking space availability around candidate destinations by including multimodal contextual data;

(3) To develop a personalised recommendation system that proactively suggests optimal parking locations, minimising user interaction while remaining adaptive to individual preferences.

These objectives jointly support a shift from reactive to proactive parking strategies, improving service efficiency, reliability, and user experience in SMS. The main novelties for this work are as follows:

- A novel smart parking recommendation framework for SMS, U-Park, is proposed, which proactively supports users from the beginning of their journey, avoiding last-minute parking decisions.
- U-Park combines multiple data sources including GPS trajectories, historical trip data, weather conditions, and infrastructure status, demonstrating the advantages of multimodal data fusion in intelligent urban mobility systems.

- A ML-based prediction pipeline is implemented, combining destination prediction, real-time parking availability, and trajectory analysis to ensure highly accurate, user-specific parking recommendations.
- U-Park is designed as a seamless, multimodal solution that functions across all trip stages, minimising user interaction while remaining responsive to explicit user preferences.

1.2.2 Privacy-Aware Energy Consumption Prediction

To address the challenge of energy consumption modelling in EVs, this work presents a privacy-aware prediction framework using multimodal data while respecting user privacy. This study mainly focuses on three RQ:

- **RQ1:** How to accurately model the energy consumption of shared EVs using real-world sensor data and simulation-based driving features?
- **RQ2:** Can FL achieve comparable or improved predictive performance compared to centralised models, while avoiding direct data sharing?
- **RQ3:** How do different FL strategies perform under realistic conditions, including centralised and decentralised environments?

We hypothesise that applying FL methods to multimodal mobility data enables accurate and privacy-preserving energy consumption modelling, and that specific FL strategies can offer robust performance under non-identical data distributions common in real-world scenarios. Based on these questions, this study focuses on three key objectives:

(1) To model the complex, dynamic energy usage patterns of EVs using real-world data of BEVs as an example, along with model-generated data to provide a comprehensive description of energy-relevant driving behaviour;

(2) To protect sensitive private data during model training by adopting FL as a privacy-preserving training framework;

(3) To evaluate the effectiveness of different FL strategies under realistic driving scenarios and different data distributions, demonstrating that strong predictive performance can be achieved without compromising data privacy.

This work supports the development of intelligent, optimised, and privacy-aware SMS. Our main novelties for this work are as below:

- A novel application of FL is introduced for the energy consumption prediction of shared BEVs, enabling privacy-aware training across multiple data sources while protecting data privacy.
- The performance of FL strategies is evaluated and compared, such as FedAvg and FedPer, based on multimodal data from ten BEVs, combining both sensor readings and physics-based model outputs.
- The proposed FedAvg-LSTM model demonstrates significant predictive accuracy, achieving up to 67.84% reduction in Mean Absolute Error (MAE) compared to baseline methods.
- The proposed method is discussed in two centralised and decentralised edge-cloud computing structures for BEV energy consumption modelling, enabling energy consumption predictions for EVs in various real-world scenarios.

1.2.3 Route Planning with Air Quality Constraints

To address the challenge of air pollution exposure, this work develops a comprehensive pollution-aware framework to support vulnerable road users, such as cyclists and pedestrians, in making safer travel decisions in urban environments. The framework integrates multimodal environmental data and predictive modelling to recommend routes with lower pollutant exposure while maintaining trip efficiency. This study aims to address the following RQ:

- **RQ1:** How to construct high-resolution real-time air pollution maps by integrating multimodal data with high missing rates?

- **RQ2:** Which imputation and/or prediction methods are most suitable for reconstructing spatiotemporal pollutant distributions under sparse and heterogeneous conditions?

- **RQ3:** Can predicted pollution levels be effectively integrated into real-time route planning systems to reduce exposure for vulnerable road users without significantly increasing trip distance?

We hypothesise that combining robust spatiotemporal prediction models with multimodal air quality data enables effective exposure-aware navigation, allowing users to make informed routing decisions that reduce pollution-related health risks with minimal increase in trip distance. To achieve this, this study dives into three objectives:

(1) To construct a high-resolution, hourly pollution map by merging multimodal data from different sources, including mobile and fixed sensors, and address the high missing rate through robust imputation and prediction models;

(2) To perform comparative evaluation of these models to reconstruct and forecast air pollutant concentrations, focusing on spatiotemporal consistency under sparse data conditions;

(3) To integrate the predicted pollutant distribution into a route recommendation system to enable real-time exposure-aware navigation.

These objectives jointly support the development of healthier and more sustainable SMS. The main novelties for this topic are:

- A comprehensive analysis of air quality data (Particulate Matter 2.5 ($PM_{2.5}$)) from multiple sources was conducted in Dublin, achieving high spatiotemporal resolution (0.5 km grid cells, hourly) while addressing missing data issues.

- A comparative evaluation of interpolation and prediction methods is conducted, supporting the selection of models that best balance accuracy and robustness under high missing rates.

- A pollution-aware route planning system is designed for vulnerable SMS users based on predicted results, enabling healthier mobility choices in urban environments.

1.3 Thesis Structure

This rest of the thesis is organised as follows:

- **Chapter 2** presents a comprehensive review of recent advancements in SMS, with a particular focus on learning-based approaches and the use of multimodal data. It also outlines key operational and environmental challenges, along with potential solutions covered in the relevant literature.

- **Chapter 3** proposes U-Park, a smart parking recommendation system aimed at improving parking behaviour in SMS. Using real-world data of shared E-Bikes from Dublin, we observe that 12.9% of users park improperly. To address this, U-Park integrates multimodal data, including historical trip records, real-time GPS trajectories, station availability, and weather conditions, to predict trip destinations and parking space availability. Learning-based models are employed to provide proactive personalised parking recommendations throughout the whole journey.

- **Chapter 4** presents a privacy-aware energy consumption modelling framework for shared BEVs, combining multimodal data with FL to enable accurate prediction while protecting data privacy. The proposed framework trains local models on-device using real-world and model-generated data, including GPS trajectories, vehicle speed, altitude, and derived features, without sharing raw data. We evaluate five FL algorithms and five local model candidates to identify the most effective setup. We further examine system behaviours under different conditions and discuss its practical deployment within edge-cloud infrastructures.

- **Chapter 5** presents a pollution-aware route planning system designed to reduce $PM_{2.5}$ exposure for vulnerable road users within SMS. Utilising multimodal air quality data collected from mobile sensors and fixed monitoring stations across Dublin, Ireland, we construct a high-resolution pollution map, addressing an initial missing rate of 89.64% through a three-stage framework comprising data imputation, spatiotemporal forecasting, and route optimisation. The system integrates various external features, including meteorological conditions, traffic flows, and land cover, to enhance model performance. This study demonstrates the effectiveness of personalised data-driven route planning in supporting healthier urban mobility choices for active SMS users.

- **Chapter 6** concludes the main findings of this thesis and discusses potential directions for future research within SMS.

Chapter 2

Literature Review

SMS are reshaping urban transportation by offering flexible and sustainable travel options. However, their complexity and dynamic nature introduce significant challenges for system management and service optimisation. In recent years, learning-based methods and multimodal data fusion have provided promising tools to address these challenges and improve operational efficiency. This chapter reviews relevant literature from three perspectives: the application of learning-based methods for system-level optimisation, the role of multimodal data in supporting SMS, and the key operational and environmental challenges along with potential data-driven solutions.

2.1 Learning-Based System-Level Optimisation

System-level optimisation aims to enhance the overall performance of complex systems by effectively coordinating resources, decisions, and operations [34], [35]. Traditionally, these tasks relied on static rules, heuristic approaches, or simple predictive models to estimate system demands and conditions [36], [37], [38]. For instance, mixed-integer linear programming has been applied in [39] to develop personalised charging strategies for EVs and in [40] to address joint task allocation and path planning problems. However, in complex and flexible environments such as SMS, these traditional methods often struggle to provide accurate and timely information, limiting their performance in system-level decision-making [23], [24].

Learning-based methods, especially ML models, can enhance decision support by improving the quality of system predictions. For example, Recurrent Neural Network (RNN) models have been applied in [41] to forecast future pickup and return rates for a bike-sharing system. By integrating these predictive insights into system-level operations, decision-making can become more adaptive, efficient, and proactive [25], [26], [27]. Consequently, learning-based methods have become more and more important in supporting system-level optimisation, helping complex systems respond more effectively to changing conditions.

Therefore, we first outline the main types of learning-based methods used for decision support, and then introduce their applications in system-level optimisation tasks in this section.

2.1.1 Learning-Based Methods for Decision Support

This section provides an overview of learning-based methods for decision support in SMS, focusing on the principles, variants, advantages, challenges, and specific applications of each algorithm, from classic ML models like Linear Regression (LR) to artificial neural networks such as Graph Convolutional Network (GCN).

Classic Machine Learning Models

Classic ML models have been widely used in SMS to support decision-making through data-driven predictions. Based on their core principles and primary objectives, these models can be categorised into statistical-based, distance-based, and tree-based models. An overview of these models is presented in Figure 2.1.

(1) Statistics-Based Models, such as LR [42], Multiple Linear Regression (MLR) [43], and Auto-Regressive Integrated Moving Average (ARIMA) [44], [45], are based on statistical theory and use probability models to describe and predict targets. They are usually applied to tasks with the hypothesis that the data follows a certain probability distribution, so they always focus on parameter estimation and hypothesis testing [46], [47], [48]. Although these models are sensitive to noise and

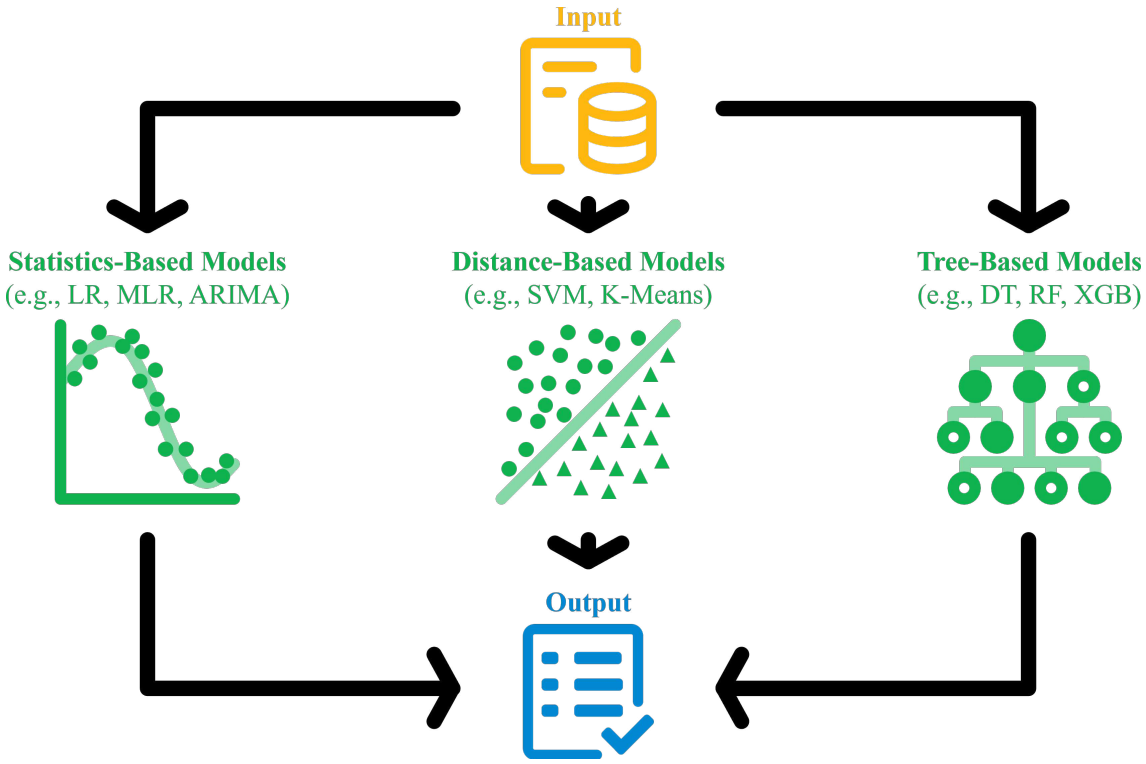


Figure 2.1: An overview of classic ML models.

outliers and struggle with nonlinear relationships, they can provide considerable prediction results and relatively strong explainability by analysing key statistical characteristics, making them suitable for some SMS applications, especially where model transparency and interpretability are important.

(2) Distance-Based Models rely on calculating the distance or similarity between data points for classification or clustering purposes. These models, including K-Nearest Neighbours (KNN) [49], Support Vector Machine (SVM) [42], and K-Means [50], employ distance metrics such as Euclidean or Manhattan distances to determine similarity, and they do not require strict statistical assumptions or parameter estimation [51], [52]. Although these models are computationally intensive and sensitive to the choice of distance metrics, they can still achieve competitive performance in scenarios with clearly structured data and limited noise, making them suitable for tasks where the relative position or similarity between data points is important.

(3) Tree-Based Models, such as Decision Tree (DT) [53], Random Forest (RF) [53], eXtreme Gradient Boosting (XGB) [54], and Light Gradient-Boosting Machine

(LGBM) [54], can classify or predict data by constructing decision trees that recursively split the dataset. These models can effectively handle nonlinear relationships and complex interactions and use branching logic to make decisions [55], [56], providing a clear and interpretable decision-making process. Although tree-based models generally require more computational resources than simpler models, they often achieve competitive results and high flexibility, making them suitable for complex SMS applications involving various data types.

Artificial Neural Networks

Based on the architectural design and methods for handling different types of data, we classify Artificial Neural Networks (ANN) models used in SMS into Multilayer Perceptron (MLP), Convolutional Neural Networks (CNN), RNN, Graph Neural Network (GNN), and Transformer models. An overview of these models is presented in Figure 2.2. Each type of network has unique processing capabilities, making them particularly suited for specific data types and tasks.

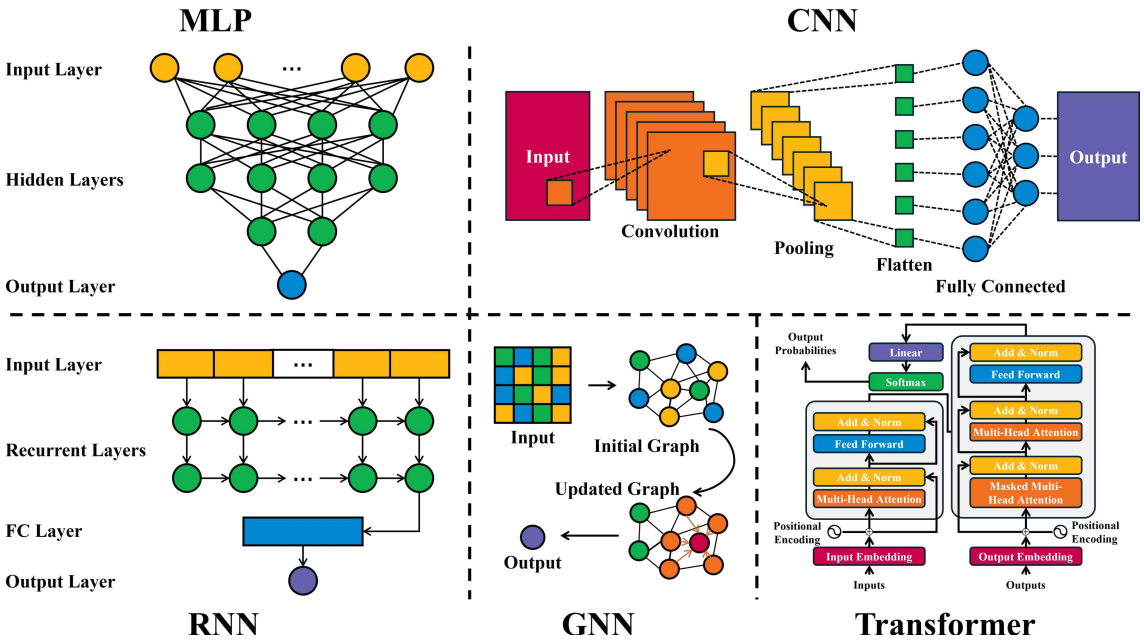


Figure 2.2: An overview of artificial neural networks.

(1) Multilayer Perceptron is a basic ANN model consisting of multiple fully connected layers, where each unit in one layer is connected to every unit in the previous layer. Compared with other ANN models, MLP does not have specific

capabilities for processing spatial or temporal data. While its ability to manage high-dimensional data, such as images or serial data, is limited, MLP remains well-suited for tabular tasks in SMS applications [57], [58]. For instance, it was employed in destination prediction as a baseline model in [49] and in modelling e-mobility energy consumption in [59].

(2) Convolutional Neural Network models use convolutional and pooling layers to process input data through local perception and shared weights, making it particularly effective at extracting spatial features. Through convolution operations, CNN models can efficiently capture spatial features in images, such as edges and textures, making them significantly effective in processing data with spatial structures, such as images and videos in SMS scenarios [60], [61], [62], [63], [64]. In addition to visual data, they have also been applied in many aspects in SMS, such as shared bike demand prediction in [65], destination prediction in [66], and energy consumption estimation for EVs in [67].

(3) Recurrent Neural Network models, including simple RNN, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and their variants, are designed with recurrent connections to handle sequential data, such as time series and text data with temporal or sequential dependencies [68], [69]. Simple RNN [49] has a basic recurrent structure suitable for sequences but struggles with long-term dependencies due to gradient vanishing. LSTM [49] adds gates to control information flow, making it more effective at capturing long-term dependencies. GRU [70] simplifies LSTM by combining the forget and input gates, making it faster and easier to train. Convolutional Long Short-Term Memory (Conv-LSTM) [70] integrates convolutional layers with LSTM, making it ideal for spatiotemporal data by capturing both spatial and temporal patterns. Overall, RNN models are highly effective for time series and spatiotemporal prediction tasks, making them particularly suitable for SMS applications involving dynamic temporal patterns.

(4) Graph Neural Network models, including GCN, Spatio-Temporal Graph Convolutional Network (STGCN), and other variants, are designed to process graph-

structured data [71], [72]. GCN uses convolutional layers to aggregate information from neighbouring nodes, making them suitable for tasks like node classification and link prediction in networks. It is widely used to obtain relationships in graph data, but may struggle with deep networks when node features become indistinguishable. STGCN extends GCN by incorporating temporal dynamics. Thus, it is ideal for spatiotemporal data like traffic flow or driver activity recognition. For instance, GCN has been employed in trip destination prediction in the context of taxi services in [44], while STGCN was used to predict the shared bike hourly demand in [73]. While the underlying transport modes differ in how they are affected by traffic conditions, the ML approaches applied remain methodologically relevant across contexts.

(5) Transformer Models are DL architectures originally developed for natural language processing, characterised by self-attention mechanisms that enable the modelling of long-range dependencies in sequential data [74], such as urban mobility time series over 12 to 48 hours, or traffic flow over daily or multi-day cycles. In SMS, Transformer models have been applied to tasks such as demand forecasting [75] and trajectory prediction [76], [77], using their strength in capturing complex temporal and sequential patterns. Although Transformers typically require substantial data volumes and computational resources, they provide strong predictive performance and flexible modelling capacity, making them well-suited for advanced SMS applications.

Ensemble Techniques

Ensemble techniques are advanced state-of-the-art approaches integrating two or more algorithms to improve model performance and robustness. In the context of SMS, ensemble methods typically involve processing data with one model to capture certain characteristics, followed by further refinement or prediction using another model.

(1) Data Transformations & Image Processing: Statistical techniques enable data transformation from various formats into images, enabling the application of

image processing and analysing techniques to address complex problems that are challenging to solve in other fields. For instance, a system was developed in [61] to monitor e-scooter drivers' attentiveness. It used short-time Fourier transform and wavelet transform to convert vibration data into images, which were then processed by various multimodal CNN models.

(2) Optical Flow & Object Detection: Optical flow is a technique used to describe the apparent position of objects of interest by tracking reference points across multiple frames over a certain period and calculating the displacement of corner points among these frames. Combined with object detection techniques, optical flow can approximate the speed of e-scooters and predict potential collision times with stationary objects. This idea has been employed to develop an accident prevention system in [78], [79] to provide real-time warnings to SMS users in situations of potential accidents.

(3) Semantic Segmentation & Object Detection: Combining semantic segmentation with object detection provides a balanced approach that enhances both accuracy and efficiency, especially in resource-limited scenarios. In [80], these techniques were integrated using "cells of interest" to classify objects rather than traditional pixel-based methods. This approach reduces the computational load while maintaining high precision, making it effective for applications in SMS, where precision and efficiency are both essential.

2.1.2 Applications of Learning-Based Methods

This section introduces different applications of learning-based methods in SMS, and Figure 2.3 provides a visual representation of some of these applications for a better understanding.

Demand Prediction

One of the most common applications of ML in SMS is demand prediction, which involves estimating the number of vehicles required within a given time window to

Figure 2.3: A pictorial representation of the applications of learning-based methods in SMS to improve efficiency, accuracy, and user experience.

meet customer demand. This task is usually regarded as a combination of time-series forecasting and spatial analysis. The complexity of urban dynamics and scalability across regions presents significant challenges for traditional approaches [17], [81]. In contrast, existing studies [82], [83], [84], [85] suggest that ML techniques such as time-series models, neural networks, and graph-based methods are well-suited to capturing these dynamics.

Trip Destination Prediction

Another important application of ML methods in SMS is the prediction of trip destinations, which requires understanding temporal patterns, spatial relationships, and user-specific behaviours. Recent studies have shown the effectiveness of ML techniques, such as time-series models, tree-based methods, and neural networks, in addressing these dimensions [86], [87], [88], as they provide more accurate, scalable, and personalised solutions compared with conventional methods [89], [90].

Energy Consumption Prediction

ML methods are integral to predicting energy consumption across various forms of SMS, particularly those involving EVs. This problem can be regarded as a combina-

tion of regression tasks and time-series forecasting, aiming to estimate precise energy consumption and investigate how it varies over time based on driving conditions, weather, and battery characteristics. Previous studies have presented the ability of ML methods in these tasks, such as linear models, ensemble methods, and neural networks [54], [91], [92], which can model the complex interactions between vehicle attributes, environmental factors, and trip features, enabling SMS to improve energy management and vehicle efficiency, and assist in route planning [33], [43], [93].

Route Planning with Constraints

Many studies have investigated route planning within SMS under different constraints, such as travel time [94], air quality [17], and trip mode preferences [95]. An essential factor making e-mobility unique from conventional transportation methods is the need to consider the real-time charge status of vehicle batteries. This application can be formed as a combination of prediction and optimisation tasks. The former estimates different objectives like energy consumption or travel time based on factors such as trip distance and driving habits, while the latter balances these objectives for optimised solutions. Various ML methods, including statistics-based, tree-based, and neural network models, have been applied to enable dynamic route adjustments based on real-time data [42], [50], [96].

Lane Recognition for Rider Safety

Lane recognition within SMS is important for ensuring rider and pedestrian safety. Riders such as e-scooter users may deviate into unauthorised areas due to traffic congestion or infrastructure limitations, increasing collision risks. This task can be formulated as a classification problem, where the system determines whether a rider is travelling within a designated lane. Object detection is often incorporated to identify lane boundaries and nearby obstacles from visual data. Recent studies [63], [64] have demonstrated the effectiveness of lightweight ML models, such as MobileNetV2 [97], which can be deployed on low-spec edge devices to enhance real-

time safety monitoring in SMS.

Object Detection for Safety & Navigation

Object detection within SMS involves localising, classifying, and identifying objects of interest from images or videos captured by cameras mounted on shared vehicles. The main purpose is to enhance safety and operational sustainability by detecting moving or stationary targets and enabling informed decision-making. Moving objects, such as pedestrians, vehicles, and other riders, are identified to alert users to potential hazards [79], [98], while stationary targets, such as potholes, traffic signs, and cones, are recognised to provide real-time navigation assistance [80].

Object Tracking & Trajectory Estimation

Object tracking and trajectory estimation in SMS aim to predict the future path of moving objects based on their current and historical positions, enhancing safety in navigation systems and autonomous operations. These methods can be particularly valuable for collision prevention in SMS. For instance, [62] presents a bicycle-based system that predicts pedestrian trajectories to anticipate potential collisions. Unlike traditional systems that rely on current positions and may trigger delays or false alarms, this approach forecasts whether and when a pedestrian might cross paths with a bicycle, enabling timely and more accurate warnings, which helps improve accident prevention in dynamic environments.

Other Applications

In addition to the applications mentioned, learning-based methods have also been employed in SMS for other purposes.

a. **Battery Health Prediction:** It significantly improves operational efficiency, reduces costs, and ensures user safety and satisfaction. The adoption of ML technologies enhances the sustainability, safety, and cost-effectiveness of e-mobility services [99], making SMS more robust and reliable for users.

b. **Trip Mode & Purpose Analysis:** This type of analysis provides operators with a deeper understanding of user behaviours and needs, allowing them to optimise vehicle deployment and provide personalised services based on actual user patterns. Employing ML methods enhances service customisation, operational efficiency, and user satisfaction [100], [101].

c. **User Safety Concerns:** ML methods have also been used to identify factors affecting the injury severity of E-Bike users in crashes [102]. Such analysis helps improve user safety and increases public confidence in SMS.

d. **Infrastructure Improvements:** It is crucial for addressing safety concerns in growing SMS use. The adoption of ML approaches effectively highlights risk areas and provides safe routes [103], contributing to more sustainable and secure urban transportation.

In general, ML methods are widely used in various aspects of SMS, including but not limited to demand prediction, destination prediction, energy consumption prediction, route planning, and vehicle and user safety concerns, such as lane recognition, trajectory estimation, object detection, and tracking. In addition to their better performance in result accuracy, these methods are also capable of managing dynamically changing real-time data. However, learning-based approaches relatively require higher data quality and computational resources. Additionally, as data-driven methods, it is necessary to consider data security and user privacy to avoid potential risks.

2.2 Multimodal Data in Shared Mobility Systems

SMS rely on diverse data sources, e.g., vehicle, environmental, and user behaviour data, to optimise operations and enhance user experiences. This section firstly introduces the key types of multimodal data typically collected in SMS, highlighting their roles and contributions to system efficiency and user satisfaction. Then, multimodal data fusion methods were discussed, especially early fusion, which integrates

features from different sources to capture interactions and improve prediction accuracy. Through this discussion, we aim to show how multimodal data fusion can effectively address challenges in SMS and support improvements in overall system performance.

2.2.1 Types of Multimodal Data in Shared Mobility

Data in SMS is collected from various sources, including different sensors, devices, and external platforms [104], [105], [106]. These data types cover information from different perspectives, from vehicle-specific attributes, e.g., location and status, to external factors, e.g., weather, traffic, and user behaviour. Integrating these diverse data sources is essential for building a comprehensive understanding of shared mobility operations, supporting system optimisation, enhancing user experiences, and promoting sustainability [19], [107]. This section introduces several key categories of multimodal data typically collected within SMS, with descriptions of their roles and contributions.

Transportation Data

Transportation data refers to all information directly related to the movement and usage of shared mobility devices, such as shared bicycles, E-Bikes, E-Scooters, and even cars and EVs. This type of data includes (1) **Location Data**, which tracks the real-time position of vehicles, enabling systems to monitor availability, predict travel patterns, and optimise routes [19]; (2) **Vehicle Status Data**, which provides insights into the current condition of devices, such as speed, State of Charge (SOC) or fuel levels, and operational status, supporting effective fleet management and reliable operations [108]; and (3) **Trip Data**, which includes origins and destinations, trip duration, and travel distance, assisting in understanding usage patterns and improving service planning [109].

In short, transportation data provides valuable information about vehicle locations, status, and trip details, which helps improve fleet management and enable

data-driven decision-making in SMS.

Environmental Data

Environmental data stands for information about external conditions that can influence the operation and demand for SMS. This includes (1) **Meteorological Data**, such as temperature, rainfall, and wind speed and direction, which can impact user preferences and vehicle performance [15]. For instance, users may prefer EVs over E-Bikes on rainy days; (2) **Air Quality Data**, which includes metrics like $PM_{2.5}$ or pollen levels that significantly affect vulnerable road users, as they may change their travel patterns according to personal preferences or health conditions [17]; and (3) **Traffic Data**, which provides real-time updates on road conditions, congestion levels, and accidents, supporting more effective route optimisation and influencing the availability and usage patterns of shared vehicles [16].

Overall, environmental data offers valuable insights into factors like weather, air quality, and traffic, enabling optimised route planning and personalised user experience in SMS.

User Behaviour Data

User behaviour data captures how individuals interact with SMS, providing insights into demand patterns and service preferences. This includes (1) **Usage Patterns**, which track user engagement with the system, such as peak usage times, preferred routes, vehicle types, and frequency of use [18]. Understanding these patterns is important for demand forecasting and ensuring service availability during periods of high demand; (2) **User Personas**, which consist of information like age, gender, and trip preferences. This data can be used to personalise services, offer targeted incentives, and tailor the system to meet specific user needs [110]; and (3) **Feedback Data**, which includes user ratings, reviews, and comments regarding the vehicles or the service. Such feedback provides actionable insights into areas for improvement, helping operators refine services [111].

In general, user behaviour data provides important insights into how users interact with SMS, supporting demand prediction and personalised service.

Infrastructure Data

Infrastructure data describes the physical environment where SMS operate, providing essential context for system management. It includes (1) **Parking Availability**, which tracks where shared vehicles can be parked and whether spaces are available or occupied [112]. Accurate parking information supports optimised vehicle allocation and maintains high availability at key locations; (2) **Charging Stations for EVs**, which offer real-time updates on charging station availability, status, and patterns of vehicle usage [113]. This information is important for maintaining the continuous operation of vehicle fleets; and (3) **Road Network Data**, which contains detailed maps, road types, and infrastructure features such as bike lanes, traffic lights, and pedestrian zones [114]. Such data is crucial for route planning, improving safety, and ensuring the efficiency of the device navigation through complex urban areas.

In summary, infrastructure data plays an important role in supporting vehicle deployment and enabling safe and efficient routing within SMS.

Payment & Transaction Data

Payment and transaction data capture the financial activities associated with SMS. This includes (1) **Payment Methods**, which record how users pay for their rides, such as through credit cards, digital wallets, or subscription models. Understanding payment preferences supports effective billing management, customer segmentation, and the development of targeted marketing strategies [115]; and (2) **Pricing Data**, which details different pricing structures, such as per-minute, per-mile, or subscription-based models [116]. It helps operators monitor revenue streams, identify pricing trends, and design promotional offers aligned with user behaviour and usage patterns.

Overall, payment and transaction data assist in understanding user spending

behaviour and improving customer segmentation within SMS.

Other Data

Other data types provide additional insights that further enhance SMS. This includes (1) **Social and Community Data**, which combines social media feedback with broader community factors such as urban planning and local transportation policies [117]. Incorporating this data helps to refine services and adapt solutions to specific community needs; and (2) **Public Transport Integration Data**, which captures information on transit schedules, routes, and stops. This data facilitates the planning of multimodal journeys and supports enhancing connections between SMS and public transport networks [118].

In summary, these complementary data types enable a more integrated and user-focused approach to the design and management of SMS.

2.2.2 Multimodal Data Fusion & Integration

Multimodal fusion methods are generally categorised into three types: early fusion, late fusion, and hybrid fusion. Each approach offers considerable flexibility, as they can be combined with various unimodal classifiers or regressors, supporting various applications across multimodal learning tasks [119].

Among these approaches, early fusion merges features extracted from different modalities directly, typically by concatenation. This approach allows the model to learn joint multimodal representations by capturing correlations and interactions between low-level features. One of its major advantages is the simplicity, as it requires training only a single unified model and involves a relatively straightforward processing pipeline compared to late or hybrid fusion methods [120], [121].

In late fusion, by contrast, the outputs of individual unimodal models are combined after their independent predictions, such as classification scores or regression outputs. Fusion techniques may include simple averaging, voting schemes, weighted combinations based on signal quality, or learning-based strategies [119]. A major

advantage of late fusion is its flexibility, as different models can be applied to each modality, and the system remains functional even when some modalities are missing [122]. However, because late fusion operates at the decision level, it cannot capture low-level interactions between modalities, which may reduce its capacity to efficiently use the complementary information across modalities [123].

Hybrid fusion combines elements of early and late fusion to capture the advantages of both [124]. By combining information at both the feature and decision levels, hybrid fusion enables models to learn fine-grained feature interactions while maintaining the modularity and robustness provided by decision-level fusion [125]. This combined strategy offers a more balanced and resilient modelling approach.

The multimodal SMS datasets used in our studies include trip behavioural data, GPS trajectories, environmental measurements, and other contextual information, all characterised by strong correlations and dependencies. Compared with other fusion strategies, early fusion captures feature-level dependencies more effectively, reduces information loss, and simplifies the integration process, ultimately improving prediction accuracy.

Accordingly, we focus on early fusion to combine features into a unified representation, enabling the model to capture feature interactions at the early stage, and in each study, we investigate how feature-level multimodal fusion contributes to prediction performance. Through this analysis, we aim to provide insights into the relationships in multimodal data sources and to show the effectiveness of multimodal fusion in addressing the challenges in SMS.

2.3 Challenges & Solutions in Shared Mobility Systems

The development and acceptance of SMS also brought several operational and environmental challenges, which should be addressed to support their sustainability and effectiveness. This section outlines three major areas of concern related to the

growth of SMS in urban environments.

Specifically, parking management remains a major concern, as improper parking behaviour reduces vehicle availability and increases operational costs. Meanwhile, the growth of electric devices in SMS has brought new challenges in energy management, where accurate consumption modelling is essential but often held back by privacy concerns, communication limitations, and the complexity of integrating diverse data sources. In addition, exposure to urban air pollution may cause significant health risks for active SMS users such as cyclists and pedestrians, but the development of effective pollution-aware mobility solutions is often constrained by missing data and the demands of real-time prediction.

Multimodal data and data-driven methods provide opportunities to address these challenges by improving the efficiency, sustainability, and user experience of SMS. These key challenges and the potential solutions enabled by multimodal and data-driven approaches are briefly introduced in this section below and detailed individually in the following chapters.

2.3.1 Parking Issues in Shared Mobility Systems

As SMS continue to expand within urban environments, parking management has become a critical operational challenge. Improper parking behaviour can disrupt pedestrian access, reduce vehicle availability, and increase operational costs for service providers [126]. For example, a study on shared E-Bike systems in Dublin, Ireland, found that 12.9% of users parked improperly, affecting both operational efficiency and management costs [7]. Commercial vehicles also encounter particular difficulties, as limited space and inadequate policies often lead to illegal parking in dense urban areas [127]. Traditional measures, such as penalty enforcement, designated parking zones, and incentive schemes, have provided partial relief but often remain reactive and inflexible in the face of dynamic and evolving mobility patterns [19], [128], [129].

To address these challenges, learning-based methods have been explored to sup-

port more proactive and adaptive parking management. ML models trained on historical trip records [130], real-time GPS trajectories [131], and contextual information such as weather conditions and traffic density [132] have shown strong potential in predicting parking demand and user destinations, as well as providing personalised parking recommendations. Specifically, classic ML methods such as Spatial Regression (SR), LR, and RF have been widely applied to predict parking availability and user destinations, using features extracted from temporal, spatial, and external datasets [45], [133], [134]. More recently, DL techniques, including CNN, RNN, and GNN, have further enhanced the ability to capture complex patterns in sequential mobility data [65], [73], [135], [136], [137]. Compared with sensor-based systems, which often require substantial infrastructure investments, learning-based approaches offer greater flexibility by integrating multiple data sources without reliance on physical deployment.

However, most existing learning-based parking solutions address only individual aspects of the parking management challenge. Few systems achieve the integration of destination prediction, real-time parking availability forecasting, and adaptive recommendation generation within a user-centric framework. Moreover, the fusion of multimodal data sources remains a substantial challenge, requiring models that can handle diverse feature types and scales. These limitations highlight the need for more comprehensive approaches, which are discussed in detail in Chapter 3.

2.3.2 Energy Management for Shared Electric Vehicles

As the development and adoption of private and shared EVs increase globally, energy management has become a critical challenge for SMS. Accurate energy consumption modelling is important for optimising vehicle operations, supporting infrastructure planning, reducing environmental impacts, and improving user experience [138], [139], [140]. Existing approaches for energy modelling include white-box methods based on vehicle dynamics [141], grey-box methods that combine physical models with data fitting techniques [142], and black-box methods that leverage ML and

DL algorithms [143]. For instance, ML models, including SVM [59], RF [53], and XGB [144], have been widely applied to predict energy consumption based on spatiotemporal features and external contextual factors. Deep learning methods, such as ANN [43] and CNN [67], have further improved prediction accuracy by capturing complex nonlinear relationships in large-scale sequential data. Each approach presents different balances between transparency, scalability, and adaptability.

Despite these advances, several challenges remain. Centralised energy modelling approaches often encounter difficulties related to data privacy, communication overheads, and the integration of heterogeneous data sources. To overcome these limitations, FL has been introduced as an effective approach, enabling vehicles to train predictive models collaboratively without sharing raw data [33], [145], [146]. FL-based methods offer enhanced privacy protection, scalability, and real-time adaptability, making them particularly well-suited to the decentralised nature of SMS [147]. However, the practical deployment of FL in energy management still faces technical challenges, including handling statistical heterogeneity across vehicles and ensuring communication efficiency within edge–cloud infrastructures [148]. The design and discussion of centralised and decentralised edge–cloud frameworks are further presented in Chapter 4 based on FL mechanism.

2.3.3 Air Quality Concerns for Active Road Users

Air pollution exposure has become a major concern for active road users, such as cyclists and pedestrians, in SMS. As they are more exposed to traffic emissions, these users are particularly vulnerable to pollutants such as $PM_{2.5}$ and Nitrogen Oxides (NO_x), which may cause serious respiratory and cardiovascular health risks [7], [149], [150]. As a result, improving mobility safety for vulnerable groups has become an essential challenge in the development of sustainable transport networks. However, designing pollution-aware mobility solutions remains a complex challenge. It is still difficult to obtain reliable air quality data, especially with fine spatial and temporal resolutions. Practically, sensor networks often produce fragmented datasets with

high missing rates, limiting the potential for accurate real-time exposure management.

To address these gaps, classic spatial interpolation methods such as Inverse Distance Weighting (IDW) and KNN have been applied to handle missing data issues [151], [152]. Meanwhile, ML techniques, e.g., RF [153], and DL models such as CNN and GCN [154], [155], have also been explored to capture the complex spatial patterns of urban pollution efficiently. Beyond interpolation, short-term air quality prediction has also developed from traditional statistical models, such as ARIMA [156], to more recent DL architectures, including Temporal Convolutional Network (TCN) [157] and Conv-LSTM [158]. These approaches enable smarter and more dynamic route planning based on the predicted pollution levels, allowing for healthier and more personalised mobility choices for SMS users.

Recent studies have demonstrated that even small changes in route choice may result in significant differences in pollution exposure [159], [160]. Consequently, more and more research has been conducted on including air quality predictions into real-time route planning frameworks [161]. At the same time, the inclusion of multimodal data, such as meteorological data and traffic conditions, has been proven to enhance the performance of pollution forecasting models [162], [163]. However, there are still several challenges in this domain, such as high missing data rates, the demand for rapid prediction in real-world conditions, and the need to balance route safety, travel time, and accessibility [164], [165], which continue to limit the practical implementation. A more detailed discussion of air quality data imputation, spatiotemporal forecasting, and route optimisation for active SMS users is provided in Chapter 5.

2.4 Summary

This chapter has reviewed existing research on learning-based methods and multimodal data in the context of SMS. Specifically, we examined the role of ML techniques in supporting system-level optimisation, the types and fusion strategies of

multimodal data in SMS, and the key operational and environmental challenges currently facing urban mobility services.

Based on this review, we identify three pressing challenges that support the technical contributions of this thesis. First, parking management remains a persistent issue, where improper parking behaviour reduces service efficiency and increases operational costs. Second, accurate energy consumption modelling for shared EVs is essential but complicated due to privacy concerns and data heterogeneity, especially in decentralised environments. Third, air pollution exposure poses significant health risks to vulnerable road users, but practical reduction through route planning is often constrained by missing data and the need for real-time prediction.

These challenges are highlighted not only for their practical relevance but also because they represent broader categories of problems where learning-based and multimodal approaches are particularly effective. Addressing these issues provides a foundation for developing scalable, intelligent, and sustainable solutions for future SMS. The following chapters explore each of these challenges in detail and propose targeted solutions based on the concepts introduced in this review.

Chapter 3

A User-Centric Parking Recommendation System

SMS, such as shared bicycles and E-Scooters, play an important role in sustainable urban transportation. However, effective parking management remains a critical operational challenge, as improper parking behaviours reduce vehicle availability and increase management costs. This chapter addresses the first of the three key challenges identified in Chapter 1, i.e., how to improve parking accuracy in SMS through data-driven user-centred approaches. It introduces U-Park, a smart parking recommendation system using multimodal data and learning-based methods to provide proactive and personalised parking support for SMS users. This work is conducted in collaboration with Mingming Liu and Noel E. O'Connor, and published in [7], [32].

3.1 Introduction

With the increasing demand for sustainable urban mobility, SMS, including shared E-Bikes, have become a vital component of Mobility as a Service (MaaS) frameworks. Shared E-Bikes offer a flexible, environmentally friendly alternative to private cars, particularly for short-distance commuting, helping to reduce traffic congestion and carbon emissions [166], [167], [168], [169]. Recent studies indicate that shared E-

Bikes can replace up to 76% of short car trips, with users travelling between 9.8 km and 17 km per week [170]. Additionally, E-Bikes offer health benefits for a wide range of users, as the electric assistance mechanism reduces physical exertion while still encouraging active transportation [171]. Compared to traditional bicycles, E-Bike users tend to cover longer distances and ride at higher speeds, averaging 3.0 km per day, whereas pedal bike riders travel only 2.6 km per day [172].

The widespread adoption of shared E-Bikes has been supported by policy changes and regulatory advancements. For example, in Ireland, new legislation has integrated E-Bikes into road traffic laws, treating them similarly to traditional bicycles[1]. However, despite their growing popularity, parking management remains a major operational challenge in SMS. Analysis of real-world MOBY shared E-Bike data from Dublin reveals that 12.9% of users fail to park correctly, highlighting widespread non-compliance and the need for more effective parking strategies [7]. Similar to traditional bicycles, improperly parked E-Bikes can obstruct pavements, pedestrian pathways, and public spaces, creating inconvenience and safety risks for pedestrians and other road users. Additionally, poor parking reduces vehicle availability, disrupts fleet accessibility, and lowers overall system efficiency. However, E-Bikes present additional operational challenges due to their reliance on dedicated charging and maintenance infrastructure. Misplaced E-Bikes, especially low-battery vehicles, may remain inoperable for long periods if parked far from charging stations. This delays battery swaps, increases maintenance workloads, and further reduces service reliability. Additionally, the higher operational costs of tracking, retrieving, and repositioning improperly parked E-Bikes place a financial burden on service providers. Ultimately, these issues degrade the user experience and threaten the long-term sustainability of SMS. While existing parking management strategies, such as penalty enforcement, incentive mechanisms, and designated parking zones, help improve compliance, they remain reactive rather than proactive, failing to offer real-time, user-centric parking recommendations. Moreover, many current systems

[1] https://www.gov.ie/en/press-release/12185-government-approves-next-steps-for-escooter-and-ebike-legislation/

lack adaptive decision-making, making them ineffective at responding to dynamic parking demand fluctuations [71], [87], [173], [174], [175], [176].

To address these challenges, data-driven approaches integrating real-time mobility data and ML techniques offer a promising solution for optimising parking management. Recent research suggests that ML-based parking recommendation systems can enhance trip planning [133], [134], destination prediction [44], [49], [66], [86], [177], [178], and trip demand forecasting [45], [65], [70], [73], [135], [136], [179]. While sensor-based approaches leveraging Internet of Things (IoT) infrastructure can provide accurate real-time parking data, they often lack flexibility and adaptability to dynamic user needs [180], [181], [182], [183]. Conversely, multimodal ML-based solutions enable the integration of heterogeneous data sources, offering greater personalisation and adaptability, enabling real-time and proactive parking recommendations [137], [184]. However, to the best of our knowledge, no existing smart parking system fully integrates destination prediction, parking availability forecasting, and personalised parking recommendations for SMS.

This chapter introduces U-Park, a multimodal, user-centric smart parking recommendation system for SMS using ML techniques. U-Park employs historical mobility data, real-time GPS trajectories, parking space availability, and other external factors such as weather information to deliver accurate and adaptive parking recommendations. In addition to the analysis findings from [7], our work also builds upon prior research in this area [71], [185]. In the former study, an Attention-based Spatio-Temporal Graph Convolutional Network (ASTGCN) was developed to predict bike availability at the start of user trips, achieving a MAE of 1.00, while in [185], a Bayesian classifier was devised for route prediction with Markov chains. This classifier can use a user's partial trajectory to update the posterior probabilities of the user's destination based on historical trip data. However, a key limitation of this work is that validation was performed solely with synthetic data, and it did not compare with advanced approaches like DL-based methods.

To increase user satisfaction, encourage SMS adoption for the daily commute,

and optimise system efficiency, U-Park introduces a novel approach using multimodal data based on ML models to address the challenges of parking management within SMS by including the following key features:

- U-Park addresses users' parking needs in SMS proactively from the journey's start, eliminating the need to search for parking at the last minute.

- U-Park ensures seamless functionality across all journey stages, minimising user input while remaining flexible enough to adapt to explicit instructions if provided.

- U-Park uses a user's current trajectory and historical mobility patterns to continually refine parking predictions, ensuring optimal accuracy and recommendations.

This chapter is structured as follows: In Section 3.2, we provide an overview of related work in existing SMS and ML techniques applied. The research problem addressed in our study is detailed in Section 3.3. The multimodal data used in this chapter is introduced in Section 3.4, and Section 3.5 delves into the overall design of our enrolment system. The details of our prototype implementation are presented in Section 3.6, and Section 3.7 presents our study results along with relevant discussions, and then the limitation of our current work is described in Section 3.8. Finally, we conclude our work in Section 3.9 and discuss future plans and improvements.

3.2 Related Work

In this chapter, we consider the relevant literature from four perspectives: parking management solutions, current SMS, ML methods in travel demand prediction, and ML methods in trip destination prediction.

3.2.1 Parking Management Solutions

Many solutions have been investigated from various perspectives to address the challenges in parking management for SMS. These include optimising the layout of parking areas, examining the role of social norms, the impact of warning messages and monetary incentives, as well as developing systems for recommending parking spots.

To optimise parking area layout, a collaborative optimisation model was proposed in [186] using an improved genetic algorithm to minimise resident walking distances and enterprise costs. The designed model considered constraints including parking area coverage and bicycle allocation, ultimately optimising both the layout of shared bicycle parking areas and the number of bicycles deployed. On the other hand, the authors of [187] focused on the impact of social norms, warning messages, and monetary incentives, and conducted a comparative experiment using the control variable method. The results suggest that behavioural incentives are more effective than social norm intervention in shared bike parking management.

Some recommendation systems have also been designed for parking management by recommending parking locations based on sensor networks and IoT technologies [180], [181]. As noted by the authors, this approach requires extensive infrastructure deployment, including sensors at intersections and parking lot entrances. However, these systems often struggle to meet the diverse and dynamic needs of users, as they tend to provide generic recommendations. Additionally, they typically require user interaction to initiate a parking search, which can be inconvenient for SMS users during their trips. In contrast, our system generates recommendations without relying on user input. By leveraging ML algorithms and trip data collected from users, our recommendations take into account diverse user trip patterns and dynamic needs.

In another previous study [177], a two-stage framework for destination prediction was proposed. This framework used hand-crafted rules, e.g., using several frequently-visited destinations in history as the candidate samples, to generate can-

didate destinations in the first stage and then employed a prediction model to predict the final destination from the candidate set, considering trip and customer attributes.

In contrast, our system, introduced in Section 3.5, relies entirely on ML-based models for trip destination prediction. Moreover, our system does not only rely on historical trip records but also incorporates trip trajectories. This means that our model not only provides a preliminary forecast based on historical data before the trip begins but also delivers in-journey predictions and recommendations. Furthermore, while the previous framework concluded with destination prediction, our approach integrates the destination as an input for the subsequent stage of the model.

3.2.2 Current Electric Shared Micromobility Services

Improper Parking Behaviours

Recent research studies have extensively investigated inappropriate parking behaviours, with a focus on identification and mitigation [7], [188], [189], [190], [191]. For instance, in [188], data from five U.S. cities was analysed to detect instances of improper parking. Research in [189] demonstrated the correlation between improper parking behaviour and parking space availability, while [192] proposed a real-time data-driven system employing ML techniques to manage parking spaces, aiming to reduce shortages and conflicts among customers. However, these analyses and methods primarily operate at the trip conclusion and necessitate specific user-provided input data.

Pricing & Incentive Policies

SMS employ diverse pricing structures, particularly pay-as-you-go models, with notable variations. In Dublin, Ireland, for example, MOBY[2], ESB[3], and Tier[4] provide shared E-Bikes, all featuring a consistent 1 euro starting fee but varying per-minute rates (5, 15, and 20 cents per minute, respectively). Similar pricing variations are observed in other regions as well[5]. Moreover, SMS companies globally implement diverse incentive policies to promote vehicle repositioning. In a study [190], a reward of ¥1.43 (approximately €0.20) per minute was proposed in China to encourage shared bike relocation. These policies consider multiple factors, such as travel distances and user preferences. Because of their practicality and proven effectiveness, we adopted similar mechanisms in our case study.

Relevant Datasets

In recent years, numerous research studies have delved into the realm of SMS. However, due to the limited accessibility of data directly from SMS providers, research and exploration in this domain have been restricted. Some studies have introduced datasets with similar characteristics[45], [133], [134], [135]. These datasets include trip history data, such as aperture and arrival times and locations, trip duration, distance, and trip trajectory data with timestamps, GPS coordinates, and vehicle indices, as seen in [134]. Nonetheless, a common limitation across these datasets is the absence of user-related features, which hinders the provision of personalised services. Therefore, after careful consideration, we opted to utilise the MOBY dataset. Although it has a limited volume, it contains information about anonymous users, enabling us to address the aforementioned challenges effectively.

In summary, addressing improper parking in SMS systems is important, and

[2]https://www.dublincity.ie/residential/transportation/covid-mobility-measures/dublin-city-covid-19-mobility-programme/journey-planning

[3]https://www.irishtimes.com/business/2022/08/17/esb-launches-shared-ebike-scheme-across-dublin

[4]https://irishcycle.com/2022/06/20/new-electric-bicycle-share-now-available-in-some-north-dublin-areas

[5]https://www.expertreviews.co.uk/scooters/1416160/how-much-do-electric-scooters-cost-everything-you-need-to-know-whether-youre-buying

this is highly influenced by parking space availability [189]. In the next sections, we examine ML techniques used in SMS for trip demand and destination prediction. The latter helps predict the endpoint of a current journey, prompting users to park correctly, while the former represents the number of departures and arrivals in an area or station, determining parking space availability.

3.2.3 ML Methods in Travel Demand Prediction

Hourly demand forecasting for SMS has been addressed using various ML algorithms and models in different cities globally. In this section, we provide an overview of previous methods and techniques for hourly demand prediction.

In previous studies [45], [133], [134], classic ML methods like SR and LR have been employed to forecast hourly demand for shared E-Bikes. In [134], for example, the authors focused on predicting E-Bike travel demand. The proposed method, Geographically and Temporally Weighted Regression (GTWR), achieved a significantly lower relative error of -1.07% compared to 7.85% for Ordinary Least-Squares Regression (OLSR) and -1.43% for Geographically Weighted Regression (GWR), demonstrating its superior predictive performance.

Recently, DL techniques, including LSTM in RNN, CNN, and GCN, have gained significant attention in micromobility research [65], [73], [135], [136], [137]. These models are adept at capturing temporal patterns. For instance, in [65], Conv-LSTM was employed to predict next-hour bicycle demand using rental histories alongside meteorological data. The use of GCN-based methods is also on the rise in this prediction task, given their ability to model non-Euclidean data localities [136]. In [73], bike pickup and return demand were predicted using the STGCN model, which outperformed three baseline models, RNN, LSTM, and GRU, in terms of prediction accuracy and computational efficiency. Inspired by cognitive attention in DL, attention mechanisms have been employed to enhance the prediction performance of GCN models. In [137], ASTGCN was proposed and applied to predict the bike availability at each bike parking station. This approach exhibited superior performance when

compared to other existing methods, including STGCN and XGB, when tested on real-world datasets. Our present research builds upon these concepts by adopting the method proposed in [137] to design the parking space availability prediction module within the SMS architecture.

Finally, extensive research has highlighted that prediction accuracy can be affected by various journey attributes, including spatial attributes (e.g., origin and destination), temporal attributes (e.g., hour of the day and day of the week), and weather conditions (e.g., temperature and wind speed) [65], [73], [133], [134], [136]. Consequently, these journey attributes have also been included in the ML-based method used in this study.

3.2.4 ML Methods in Trip Destination Prediction

Methods and models used in trip destination prediction in relevant papers are detailed below. Generally, travel destination prediction can be categorised based on the data utilised into two types: history-based and GPS-based. The former focuses on the prediction based on the trip records, including the temporal and spatial features of only the origins and destinations, while the latter aims to predict the destination based on time-series data reflecting the trip trajectory.

Trip History-Based Prediction

Trip history datasets provide temporal and spatial information about the origin and destination of each journey. Researchers have employed various techniques, including ARIMA, Conv-LSTM, CNN, GCN, to predict destination using trip history data [44], [66], [177]. For instance, a Destination Prediction Network based on Spatio-Temporal data (DPNst) was introduced in [66], integrating LSTM, CNN and a Fully Connected Neural Network, each extracting user behaviour, spatial features, and external features like weather conditions, respectively. The results were compared to alternative models, including LSTM and Naive Bayesian (NB) [185].

Our review illustrated that complex models with strong performance are adapt-

able to large datasets. However, considering the limitations of our trip history dataset, we chose ANN in our study because of its flexibility, adaptability, and suitability for handling small datasets, making it an ideal choice for our research.

Trajectory-Based Prediction

In parallel, numerous research studies have explored destination prediction based on GPS data for various applications [49], [70], [178]. For instance, in [49], the authors proposed an Multi-module Deep Learning Network (MDLN) to solve the problem of destination prediction. As the journey trajectory is a temporally-ordered GPS location collection, RNN [193] and LSTM [70] are also widely used in trajectory-based destination prediction thanks to their ability to handle temporal and sequential data effectively. Besides, recent research has indicated that the initial k and final k segments of a GPS trajectory significantly contribute to destination prediction [49], [194], [195], so we adopted the same method.

To sum up, ML models for both hourly demand and destination prediction commonly incorporate external factors such as weather and points of interest in addition to spatial and temporal information related to trip origins and destinations [44], [66], [134], [179]. RNN models play a crucial role in these tasks due to their effectiveness in time-series prediction problems [44], [49], [65], [66], [70], [73], [135], [136], [177], [179]. Considering our dataset size, the model complexity, and prediction accuracy, we decided to employ a simple ANN for initial destination prediction based on trip history at the start of a journey, while using an RNN model during the trip based on GPS data in our study.

3.3 Problem Formulation

In this section, we introduce our research problem and provide an in-depth explanation of each question. To enhance clarity, we use a shared E-Bike system as an example of SMS systems, but U-Park applies to other SMS systems as well. We segment the user journey into three stages: pre-journey (from the moment users de-

cide to start a trip and scan the Quick Response (QR) code attached to the E-Bike until they are unlocked), in-journey (from the point of unlocking the bike until users are near their destination and ready to end the trip), and post-journey (from the moment users are near the end of the trip until they receive final recommendations from U-Park, lock up the bike, and complete the payment process). Specifically, we consider the trip to be concluding if the user's current position is within a certain distance, e.g., *dis* metres, of the predicted destination.

3.3.1 Overall Research Problem

Notations: In essence, U-Park's primary objective is to solve improper parking behaviours in SMS by providing parking station recommendations as a trip nears completion. Given this context, we can formulate the parking station recommendation problem. With an SMS system, the operator has established a total of S predefined parking stations. We introduce the set of parking stations, denoted as $\mathbb{S}$, as follows:

$$\mathbb{S} = \{s_1, s_2, \ldots, s_S\} \tag{3.1}$$

For a given user, their historical record $\mathbb{H}$ and GPS record $\mathbb{G}$ contain information about a total of N past trips within the SMS. Leveraging $\mathbb{H}$ and $\mathbb{G}$, our goal is to devise a system to make recommendations for suitable parking stations near the destination of the user's ongoing trip. To achieve this, we use $\mathbb{T}_i$ to represent their i^{th} trip. For each trip $\mathbb{T}_i$, we define its GPS record $\mathbb{T}_i^G$ and historical record $\mathbb{T}_i^H$ as follows.

Firstly, let us define a GPS record of a single trip $\mathbb{T}_i^G$ by

$$\mathbb{T}_i^G = [(t_1^i, p_1^i); (t_2^i, p_2^i); \ldots; (t_{L_i}^i, p_{L_i}^i)] \tag{3.2}$$

where the sequences $(t_1^i, \ldots, t_{L_i}^i)$ and $(p_1^i, \ldots, p_{L_i}^i)$ represent lists of timestamps and GPS positions, both having the same length denoted as L_i. Thus, the size of $\mathbb{T}_i^G$ is

$L_i \times 2$. The initial position p_1^i and the final position $p_{L_i}^i$ correspond to the origin and destination of the trip $\mathbb{T}_i^G$, while the first timestamp t_1^i and last timestamp $t_{L_i}^i$ represent its departure time and arrival time, respectively. Given the context above, the user's historical record for a single trip $\mathbb{T}_i^H$ is defined as:

$$\mathbb{T}_i^H = [(t_1^i, p_1^i, t_{L_i}^i, p_{L_i}^i)] \tag{3.3}$$

which means the history of one specific trip $\mathbb{T}_i^H$ contains the spatial and temporal characteristics of its origin and destination and the size of $\mathbb{T}_i^H$ is 1×4. Expanding on the context given above, the user's historical record for N trips is:

$$\mathbb{H} = \{\mathbb{T}_1^H, \mathbb{T}_2^H, \ldots, \mathbb{T}_N^H\} \tag{3.4}$$

and similarly, the user's GPS record for N trips is:

$$\mathbb{G} = \{\mathbb{T}_1^G, \mathbb{T}_2^G, \ldots, \mathbb{T}_N^G\} \tag{3.5}$$

Finally, based on the list of all trip destinations within this user's GPS records, denoted as $(p_{L_1}^1, p_{L_2}^2, \ldots, p_{L_N}^N)$, we define the set of destinations for this user as follows:

$$\mathbb{D} = \{d_1, d_2, \ldots, d_D\} \tag{3.6}$$

where any $d_j \in \mathbb{D}$ represents a unique destination, and D is the length of $\mathbb{D}$, so for any given trip $\mathbb{T}_i$ in $\mathbb{G}$, we have the destination $p_{L_i}^i \in \mathbb{D}$.

Problem: For an ongoing trip $\mathbb{T}_k = [(t_1^k, p_1^k); \ldots)]$, the primary goal of our U-Park system is to determine a suitable parking station $\hat{s_k} \in \mathbb{S}$ and make recommendations aiming at increasing the user's chance to obtain an available parking space near the predicted destination $\hat{d_k} \in \mathbb{D}$, which is predicted by ML models within U-Park system.

Remark: The $\mathbb{S}$ and $\mathbb{D}$ defined in this section represent different sets. The

former is the predefined station set, while the latter is the set of destinations from this user's previous trips. The destinations may or may not be included in $\mathbb{S}$. Only for new users without history records, we specify that $\mathbb{D} = \mathbb{S}$. To address our primary research question, we divide it into three ML tasks corresponding to the three stages introduced earlier. We define these tasks as follows and provide detailed explanations in the subsequent sections: (i) history-based destination prediction in the pre-journey stage; (ii) trajectory-based destination prediction in the in-journey stage; and (iii) parking space availability prediction in the post-journey stage.

3.3.2 History-Based Destination Prediction

Notations: As introduced above, this history-based prediction model is applied at the pre-journey stage to predict the destination of this trip. In addition to the departure and arrival timestamp and position introduced in (3.3), each specific trip $\mathbb{T}_k$ includes a set of extra features f_k^H, such as temporal features (e.g., day of the week) and weather conditions (e.g., temperature). Denoting N_f^H as the number of f_k^H, the length of the extended historical record for $\mathbb{T}_k$ becomes $M_H = N_f^H + 4$. In other words, the size of the extended historical record for N trips in total amounts $N \times M_H$.

Problem: When dealing with an upcoming trip $\mathbb{T}_k = [(t_1^k, p_1^k, f_k^H)]$, the research question at this stage is to solve the classification task of determining the category within $\mathbb{D}$ that the trip $\mathbb{T}_k$ belongs to, or in other words, to identify a $d_H^k \in \mathbb{D}$ as the destination for $\mathbb{T}_k$.

Remark: This problem is solved every time a user gains authorisation to unlock a vehicle in this SMS. Since the model only relies on the user's historical record and external features, the prediction result for the current trip $\mathbb{T}_k$ remains constant until the dataset is enriched by new records.

3.3.3 Trajectory-Based Destination Prediction

Notations: As previously mentioned, this trajectory-based prediction model is employed during the in-journey stage to improve the prediction result made at the pre-journey stage. In addition to the timestamp and GPS position sequences in (3.2), each specific trip $\mathbb{T}_k$ with a length of L_k incorporates supplementary features f_k^G, such as temporal features (e.g., day of the week) and weather conditions (e.g., temperature) obtained for each timestamp. Denoting N_f^G as the number of f_k^G, the dimensions of the extended GPS record for $\mathbb{T}_k$ can be represented as $L_k \times M_G$, where $M_G = N_f^G + 2$ denotes the number of features for each timestamp.

Problem: At this stage, the research question is to solve the classification task of determining the category in $\mathbb{D}$ this ongoing trip $\mathbb{T}_k = [(t_1^k, p_1^k, f_k^G); (t_2^k, p_2^k, f_k^G); \dots]$ corresponds to, or in other words, to designate $d_G^k \in \mathbb{D}$ as the destination for $\mathbb{T}_k$.

Remark: Concerning General Data Protection Regulation (GDPR), GPS data will be converted and then integrated into our system rather than being directly input. The processing of GPS data will be introduced in Section 3.6. This problem is resolved each time a new GPS position is observed during the current trip. Since this model relies on external features and the user's real-time trajectory, it is feasible to update the prediction result when a new GPS point is included.

3.3.4 Parking Space Availability Prediction

Notations: Lastly, we articulate the parking space availability prediction problem as follows. Let A_t^k denote the number of available parking docks or spaces at station $s_k \in \mathbb{S}$ at time t. $\mathbf{A}_t$ represents the vector consisting of the numbers of the available parking docks or spaces at all S stations in total at time t. Consequently, the length of $\mathbf{A}_t$ is S. Moreover, we use f_t^k to denote the value of external features at station s_k at time t, such as temporal features (e.g., day of the week) and weather conditions (e.g., temperature) for model training, and we use M_F to represent the length of f_t^k. Similarly, we denote the feature set values of all parking stations at time t by $\mathbf{f}_t$, so the size of $\mathbf{f}_t$ is $S \times M_F$.

Problem: Mathematically, the learning objective of this prediction model, which is applied in the post-journey stage, is to find a function $\mathbf{H}(.)$ that addresses the following problem:

$$\mathbf{A}_{t+1:t+n} = \mathbf{H}(\mathbf{A}_{t-m+1:t}; \mathbf{f}_{t-m+1:t}) \tag{3.7}$$

where m and n represent the input and output lengths of this model, respectively. Besides, $t+1:t+n$ denotes the output as a sequence of vectors from time $t+1$ to $t+n$.

Remark: Predictions for this problem persist until the user confirms the destination of this trip or reaches locations with the highest prediction occurrences. This model is influenced by the arrival time and external features, and the arrival time can be obtained using any commonly used map Application Programming Interface (API).

To explain the reason for making multiple prediction results, we consider the following example: A user plans to take a shared E-Bike trip from *Point A* to a crowded bike station *Point C*, which is close to a less crowded station *Point B*. Using multiple prediction results for each bike station in the next hour, at certain intervals (e.g., 15 minutes), U-Park can forecast that *Point C* might have fewer parking spaces available than *Point B* upon arrival with high confidence. In response, U-park will recommend the user to park at *Point B* and walk to *Point C* rather than park at *Point C* to avoid potential time costs in looking for parking spaces. This provides a more convenient solution and helps prevent wasted time searching for parking spots. As a result, users can better deal with uncertainty arising during their travel, improving overall experience and satisfaction.

3.4 Multimodal Data & Dataset

In our work, data from three different sources was employed to provide trip-relevant details from multiple perspectives, including (1) the trip details and trajectories pro-

vided by *MOBY BIKES*[6] (**MOBY dataset**), (2) the meteorological data collected from the *Irish Meteorological Service*[7] (**Meteorological dataset**), and (3) the parking station information sourced from *NOW Dublinbikes*[8] (**Dublinbikes dataset**). In this section, we detailed our analysis of the MOBY dataset and introduced how these 3 datasets were integrated and contributed to solving our research problems.

3.4.1 Dataset 1: MOBY Dataset

The MOBY dataset is provided in a fully anonymous and GDPR-compliant manner, purely for academic research purposes as part of the *Smart DCU*[9] programme, consisting of two patterns of tables reflecting user rental reports and GPS trajectories.

Rental Reports

This section contains around 58,000 journey records, collected from January 2020 to September 2021 in Dublin, Ireland, with 37 features in total, including information about the user, E-Bike (e.g., *Bike ID*), consumption (e.g. *Total price*), time (e.g., *Rental duration*), position (e.g., *Return location*), and distance for each journey. Some briefly descriptive statistical information for total duration and distance of journey records are reported in Table 3.1. Some features in this dataset, such as *Voucher code* and *Status*, do not provide highly relevant information for our analysis and were therefore excluded during data preprocessing. Additionally, records containing "null" or "0" values in key features like *Return time* and *Distance* were dropped to simplify the analysis. To further refine the dataset, we applied the **Z-score** method to identify and eliminate outliers in features with exceptionally high or low values, ensuring more reliable results.

In Figure 3.1, the histogram of trip duration compares E-Bike usage between weekdays and weekends, based on 97.53% of the available data. The records used in this chart were selected by excluding trips shorter than 5 metres, with dura-

[6] `https://mobybikes.com/`
[7] `https://www.met.ie/climate/available-data/historical-data`
[8] `https://www.dublinbikes.ie/en/home`
[9] `https://smartdublin.ie/smart-districts/smart-dcu`

Table 3.1: Summary of descriptive statistics for duration and distance for all E-Bike journey records.

Features	**Minimum**	**Maximum**	**Median**	**AVG**	**STD**
Trip Distance (metres)	1.00	7914.00	371.00	586.66	653.90
Trip Duration (minutes)	0.17	4587.18	21.62	56.53	131.93

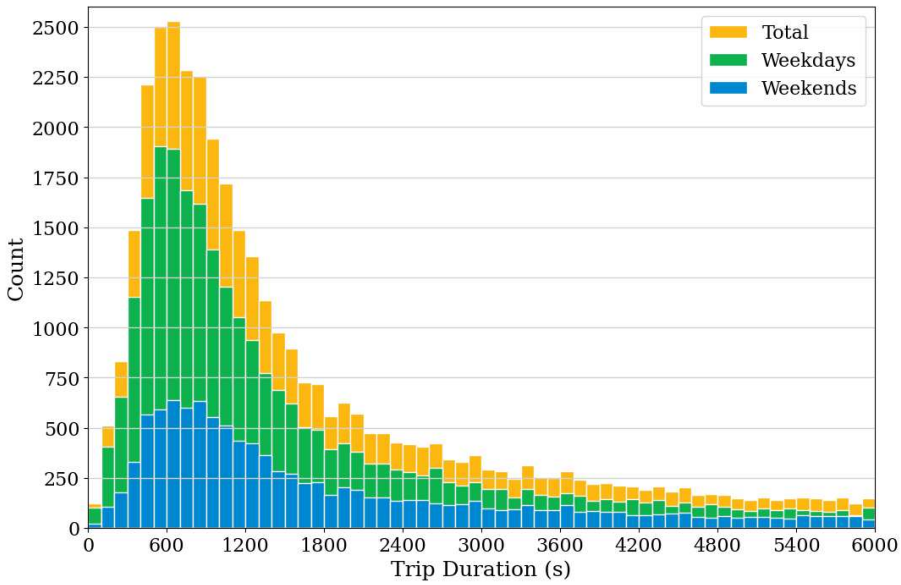


Figure 3.1: The histogram of trip duration based on partial data (97.53%).

tions under 1 second, or with average speeds exceeding 0.2 km/h, to remove outliers and unreliable readings. The results indicate that most trips last between 600 and 800 seconds, with weekday usage higher than weekends, although the overall distribution remains similar. This suggests that E-Bikes are primarily used for short daily commutes, with weekends showing a more dispersed pattern that may reflect leisure-oriented trips. Further supporting this, Figure 3.2 presents the total number of recorded trips and the percentage distribution of each E-Bike type across different distance intervals. The results show that trip frequency increases sharply until reaching a peak of 3,248 records at a distance range of 200–250 metres, after which it gradually declines to near 0 beyond 2,500 metres. This confirms that most trips cover very short distances, reinforcing the role of E-Bikes as a last-mile transport solution rather than a long-distance alternative. Additionally, the **DUB-General** bike type dominates across all distance intervals, consistently accounting for at least 90.59% of total users, while **Private** E-Bikes are rarely used, appearing only once

in the 850–900 metre range.

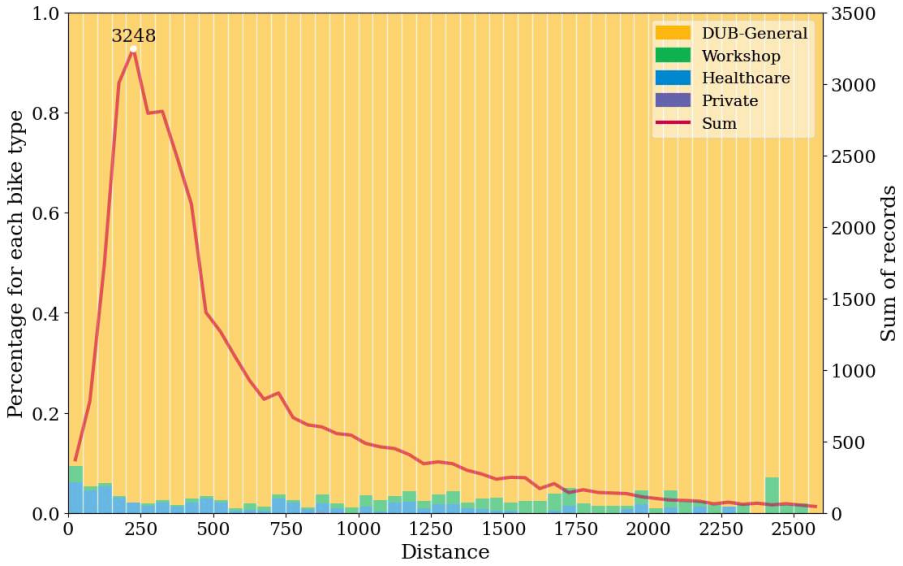


Figure 3.2: Percentage and sum of records for each E-bike type.

GPS Trajectories

This part of the dataset captures the GPS coordinates of each E-Bike at different time points, containing five features (*Bike ID*, *Latitude*, *Longitude*, *Valid*, and *Time*) across more than 4 million records collected from January 2020 to September 2021. During data processing, we identified duplicate records, particularly in the *Time* feature, where multiple GPS signals were received from different satellites at the same timestamp. To address this, we replaced the duplicated coordinates with their **mean values** for each device at a given timestamp. Some redundant features were removed, including *Valid*, which consistently held the value "1" across all records. Finally, after removing a small percentage of records with missing values, the cleaned dataset provides a refined representation of E-Bike trajectories. As illustrated in Figure 3.3, the trajectory of bike No. 40 on a specific date is shown, where the trip began on Wellington Road in Phoenix Park, highlighted in blue, and ended on Heytesbury Street in Dublin 8, marked in red. The colour of each road segment, ranging from green to red, represents the riding speed, from slow to fast, with values from 0 to around 25.31 m/s. The E-Bike journey follows a structured path, moving from a less dense area toward the city centre with a generally steady riding pace.

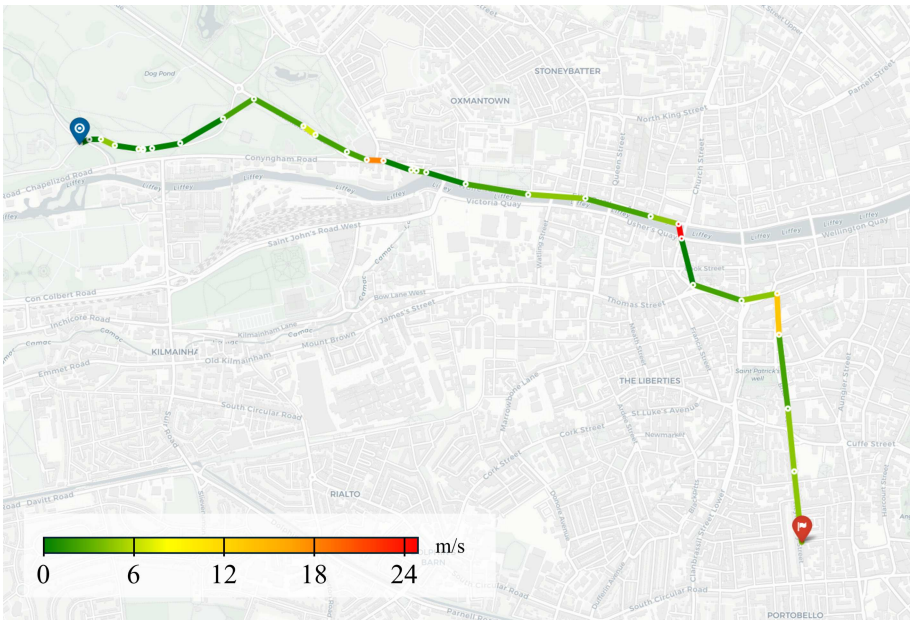


Figure 3.3: Trajectory of bike No. 40 on a specific date.

Joint Analysis & Processing

To prepare the dataset for the implementation of U-Park, we considered all user data in rental reports for the history-based model, filtered out users with less than 10 trip records in our dataset, and then selected the data from the top 5 users with the most trips for the trajectory-based prediction model. As shown in Figure 3.4, these users contributed 716 complete journey records in total, occupying about 14.37% of the total trips, so we believe that such experiments can reflect the performance of our proposed model to a certain extent.

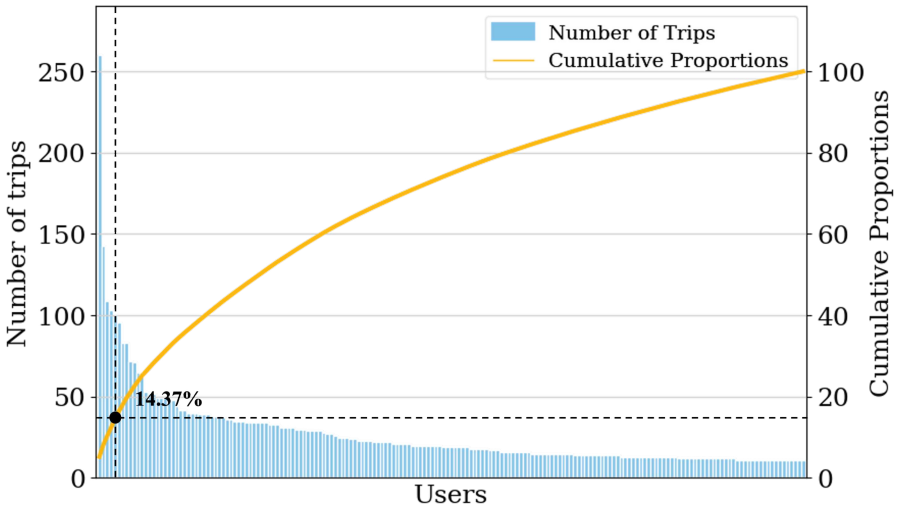


Figure 3.4: The number of trip records for each user.

Table 3.2 illustrates the diverse trip patterns among the selected users. Specifi-

cally, Users 1, 3, and 5 have a similar number of origins and destinations within their trip histories. In contrast, there is a significant imbalance in the numbers of origins and destinations for Users 2 and 4. The difference in trip patterns could be a challenge for the prediction model. It also highlights the necessity of effective strategies that can accommodate and adjust individual user behaviours. To ensure compliance with GDPR regulations, we performed the following steps: 1) splitting the feature *Starting Time* into separate components for the hour and weekday, and 2) transforming GPS information into road segment data, containing only road labels rather than exact GPS coordinates, by *Mapbox Map Matching API*[10].

Table 3.2: Statistics for selected user data.

User ID	**Trips**	**Origins**	**Destinations**	**Proportions**
User 1	260	25	25	0.0522
User 2	143	2	11	0.0287
User 3	109	40	31	0.0219
User 4	103	18	9	0.0207
User 5	101	26	14	0.0203

It is worth mentioning that a significant imbalance in trip origins and destinations was observed for these users. To address this issue, we employed the Synthetic Minority Over-sampling Technique (SMOTE) algorithm [196]. SMOTE works by calculating the K nearest neighbours of each minority class sample, randomly selecting N samples from these neighbours for random linear interpolation, and generating new minority class samples accordingly. This resulted in an augmented training set with more samples in minority categories to address the imbalance problem.

3.4.2 Dataset 2: Meteorological Dataset

The meteorological dataset comprises historical hourly records from three monitoring stations: "Casement", "Dublin Airport", and "Phoenix Park". It includes multiple parameters such as precipitation amount, air temperature, and vapour pressure. Each rental record was matched to the nearest station using Euclidean

[10]`https://docs.mapbox.com/api/navigation/map-matching`

distance and its rental time, while GPS trajectory samples were paired based on Euclidean distance and timestamp.

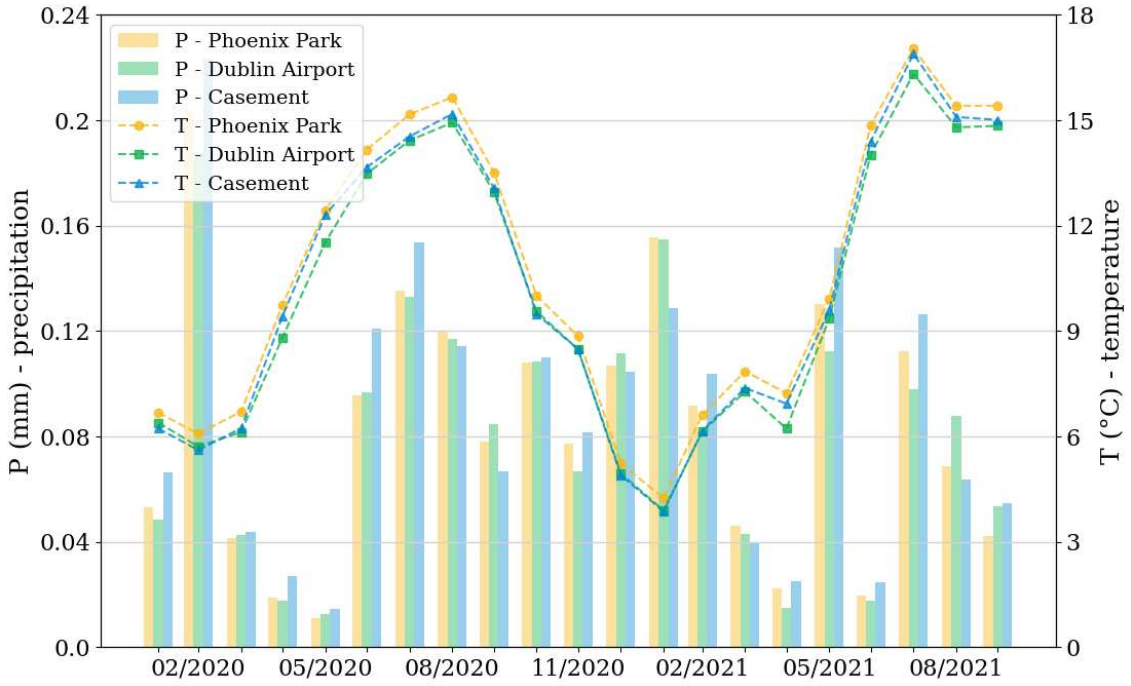


Figure 3.5: Temperature and precipitation of three monitoring stations.

An analysis of temperature and precipitation trends across these stations, as illustrated in Figure 3.5, reveals distinct seasonal patterns. Specifically, Precipitation exhibits periodic fluctuations, with peaks in winter and lower levels in summer, reflecting a typical seasonal rainfall cycle influenced by regional atmospheric circulation. Temperature also follows a clear seasonal trend, reaching its lowest levels in winter and peaking in summer, with a sharp rise after May. Trends across the three stations are highly consistent, indicating a relatively uniform temperature distribution across the city. However, **Phoenix Park** tends to record slightly higher temperatures, possibly due to its urban park environment, while **Casement** shows slightly higher precipitation in most months, which may be linked to local topography and exposure to prevailing weather systems. A notable inverse relationship is observed between precipitation and temperature, where higher rainfall often coincides with lower temperatures, particularly in winter. However, this relationship is not strictly linear and may be affected by some transient meteorological events.

3.4.3 Dataset 3: Dublinbikes Dataset

Inspired by the successful application of the ASTGCN model in bike availability prediction, as detailed in [71], we initially intended to adapt its structure to predict parking space availability in our study. However, the MOBY dataset lacks specific data on parking space availability at the parking stations. To address this, we assumed that the distribution of available parking spaces in our dataset would resemble that of a comparable open dataset, the Dublinbikes dataset used in [71]. We then aligned the parking stations from both datasets based on geographical proximity. The statistical histogram of parking space is illustrated in Figure 3.6.

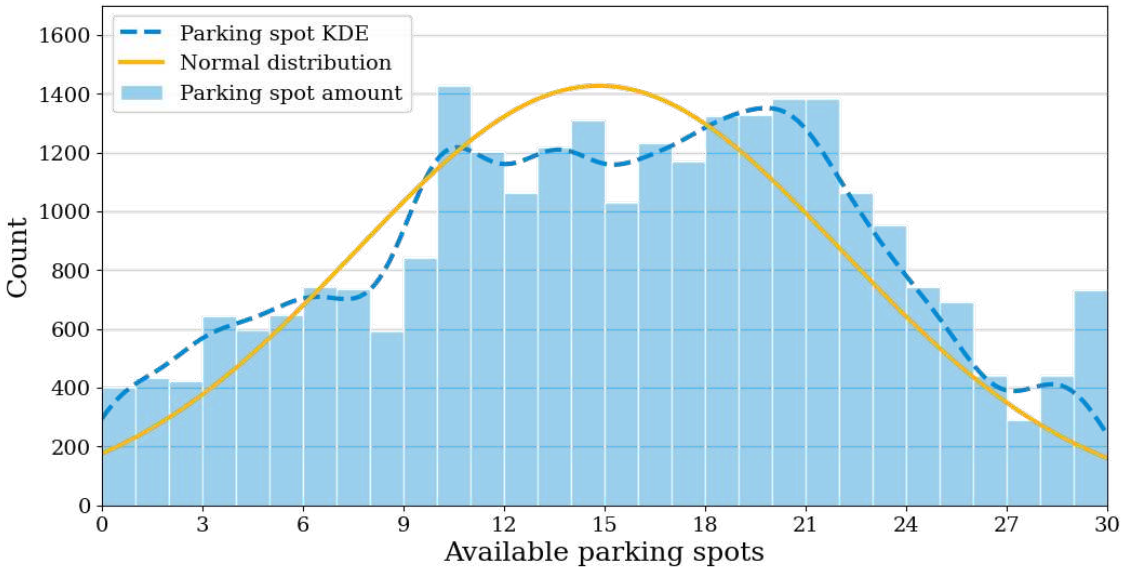


Figure 3.6: Histogram of parking space in station 32 on York Street East.

3.5 Methodology & System Design

This section presents the system structure of U-Park to address the research questions discussed in Section 3.3. An overview of the system architecture is provided in Figure 3.7, with hardware and user interface modules involved in all three journey stages: pre-journey, in-journey, and post-journey, as defined in the previous section. Again, we use a shared E-Bike system as an example of SMS systems for clarity.

U-Park consists of three main modules: a hardware module (e.g., E-Bikes with various sensors), a user interface module (e.g., a mobile application), and a prediction

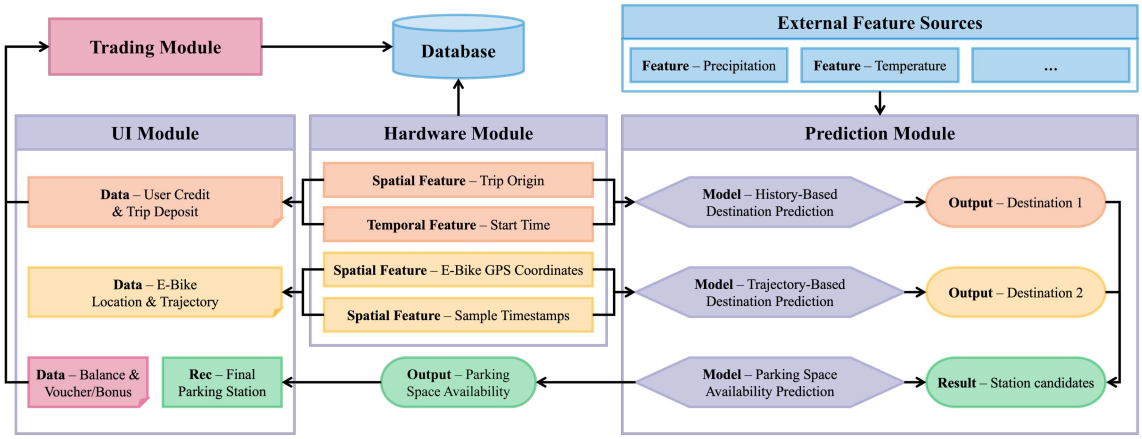


Figure 3.7: The proposed system architecture for U-Park.

module based on ML prediction algorithms. Additionally, it incorporates external feature sources, a database, and a trading module to manage financial transactions.

3.5.1 Hardware Module

The hardware module attached to U-Park's E-Bikes serves three primary functions: collecting physical data from the bikes, calculating trip deposits, and controlling motor rotation. This module can be built on a Single-Board Computer (SBC), e.g., a Raspberry Pi with various sensors connected to interact with the cloud database. When an unlocking or locking request is received, the SBC will unlock or lock the E-Bike accordingly and calculate the trip deposit, and the motor status will be stored in the database and updated in real-time throughout this trip. Additionally, this module will upload the GPS coordinates of E-Bikes to our database, allowing both administrators and users to track the current locations of E-Bikes.

3.5.2 User Interface Module

An Android mobile application has been developed to interact with users. It collects trip origin data, communicates with the prediction model, and delivers recommendations back to users. The application handles user credits, payments, and data collection, which is then uploaded to the database through the trading module. Furthermore, the application employs map APIs and QR code scanning to enable users to view their current position, locate nearby E-Bikes, and unlock available

E-Bikes by scanning QR codes.

When users open and log into the mobile application, their positions are located on the map according to GPS coordinates. The map also displays the locations and availability of E-Bikes using colour-coded markers, e.g., in Figure 3.11. Users can input their intended destination d_U^k for a trip $\mathbb{T}_k$. Once the user provides a destination, the system proceeds directly to the post-journey stage, as described in Section 3.5. However, our primary focus is on scenarios where users may not provide input to the system, allowing us to address more complex and versatile situations.

Users can scan the E-Bike's QR code to start the credit validation and deposit payment process when clicking on an available E-Bike marker. The deposit amount is determined by the total duration from the origin to the predicted destination, as determined by the history-based model in Section 3.5. The equation employed in this step (3.10) is detailed in the following section. If the user has sufficient remaining credit to cover the deposit, the device is unlocked after payment, and the journey status is updated in our database. Otherwise, the user is advised to add credit to their account and try again.

The E-Bike GPS is continuously captured and uploaded until the user ends the trip. Upon trip completion, the balance amount $F_{balance}$ is calculated by (3.8), which is determined by the trip duration, the minimum distance to predefined stations, and the deposit paid at the start of this trip.

$$F_{balance} = f \cdot (t_{L_k}^k - t_1^k) + F_{parking} - F_{deposit} \tag{3.8}$$

In (3.8), f represents the charge per second, $t_{L_k}^k$ is the actual arrival timestamp at the destination d_k and $F_{parking}$ is the predefined parking fee defined in Section 3.6. If $F_{balance} \geq 0$, it indicates that the user needs to be charged again at the end of the trip, and if $F_{balance} < 0$, a refund is provided.

3.5.3 Prediction Module

The prediction module serves two purposes: destination prediction and parking space availability prediction. We employ a two-step prediction model to improve the accuracy of destination prediction. When users initiate a trip, we first predict a list of potential destinations, i.e., a set of d_H^k, using our history-based prediction model. Subsequently, we refine it by our trajectory-based prediction module result, i.e., d_G^k. In other words, destination prediction is divided into two stages: pre-journey and in-journey. The final estimated trip destination $\hat{d}_k$ is determined by both d_H^k and d_G^k.

The workflow of the entire prediction module is illustrated in Figure 3.8, where items coloured in yellow, orange, and green represent history-based and trajectory-based destination predictions as well as the parking space availability prediction, respectively. The red item summarises the final result of the destination prediction task, while the blue items calculate probability distributions to make a recommendation regarding which parking station to use. For clarity, we summarise the notations used for destination prediction in Table 3.3.

Table 3.3: Notations for destination prediction tasks.

Notation	Explanation
d_k	the actual destination of trip $\mathbb{T}_k$ which will be recorded when the journey ends
d_H^k	the predicted destination of trip $\mathbb{T}_k$ which results from our history-based model
d_G^k	the predicted destination of trip $\mathbb{T}_k$ which results from our trajectory-based model
d_U^k	the user-planned destination of trip $\mathbb{T}_k$ which is determined by the user when the journey starts
$\hat{d}_k$	the overall result of the destination prediction module which is determined by d_H^k and d_G^k
$\hat{s}_k$	the overall result of U-Park, the final parking station recommended to the user when $\hat{d}_k$ is decided

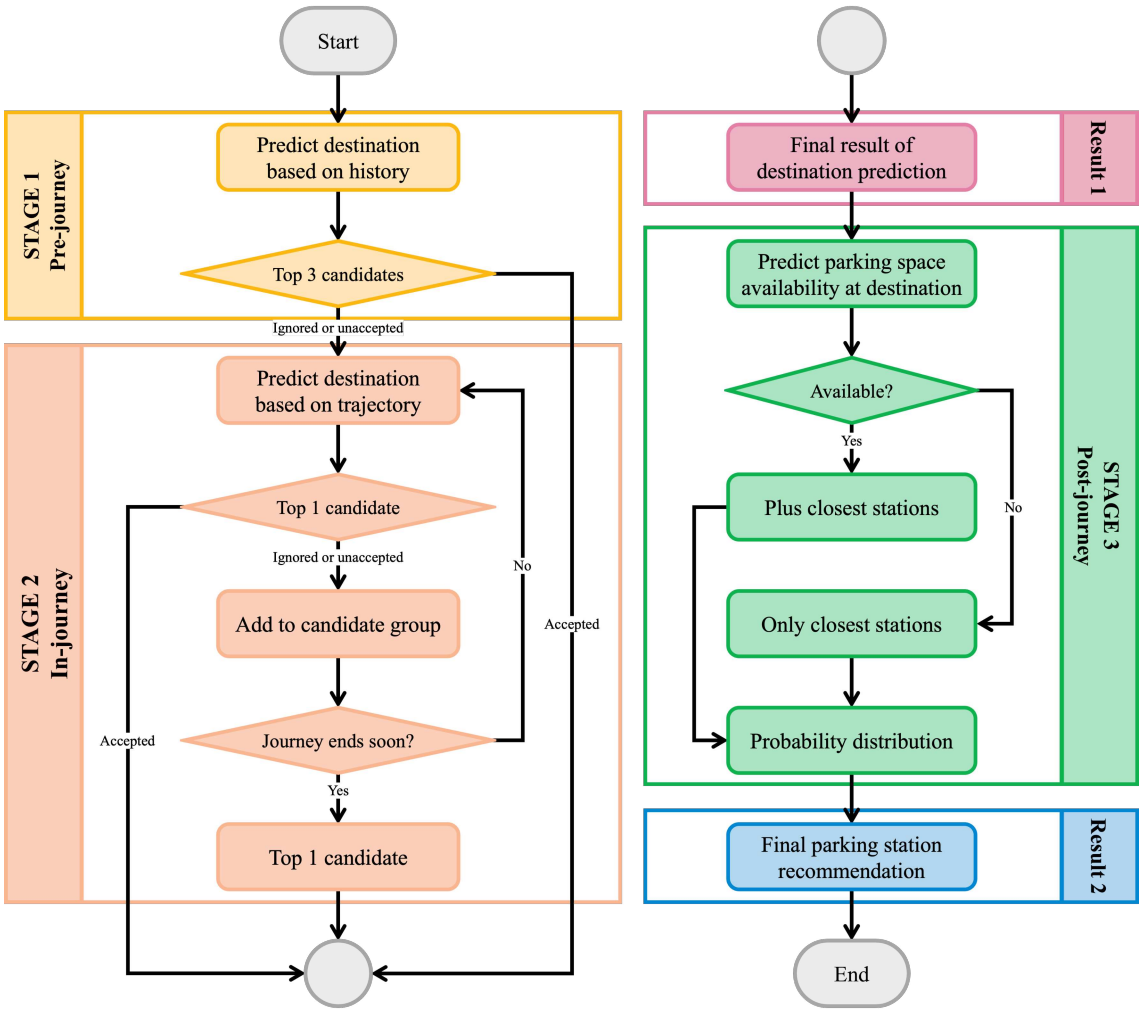
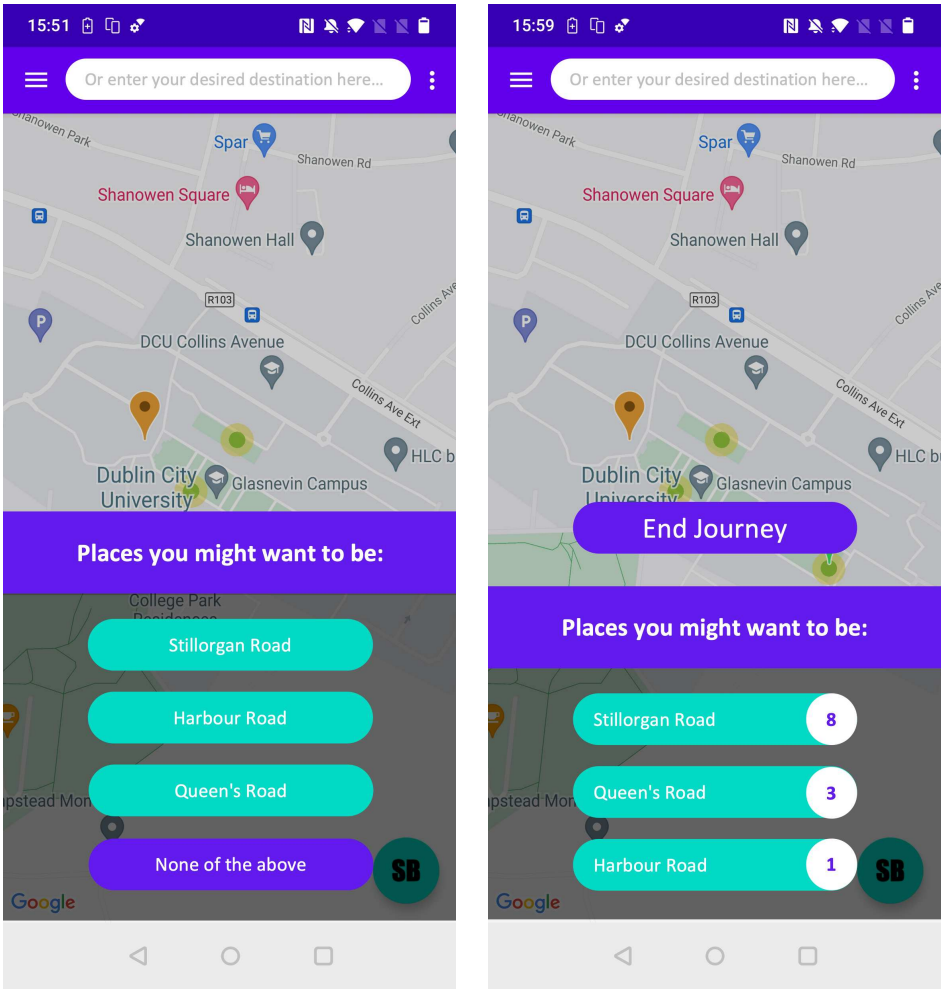


Figure 3.8: Prediction module workflow.

Pre-journey

Our prediction module focuses on destination prediction based on the user's history at this stage. Our classification model, as introduced in Section 3.5, is a probabilistic classifier, which results in a probability distribution over a set of categories instead of the most likely category. Consequently, we consider multiple destination candidates with the highest probabilities and calculate the trip deposit $F_{deposit}$ accordingly.



(a) Pre-journey recommendation. (b) In-journey recommendation.

Figure 3.9: Sample user interfaces of recommendations.

Following the notations, when a user starts a trip $\mathbb{T}_k$ from the origin p_1^k, U-Park collects necessary data, including the departure location, starting time, weather conditions, and battery remaining power. Temporal, spatial, and weather-related data are used for destination prediction, while the remaining battery power is compared with the expected travel distance to determine if it is sufficient to complete this

trip. If not, U-Park helps the user find other suitable E-Bikes. The predicted destination candidates $\mathbb{D}_H^k$, represented as a set of d_H^k, are then recommended for user confirmation. Depending on the user responses, two cases are considered in this step.

Case 1: If any result d_x in $\mathbb{D}_H^k$ matches d_U^k, i.e., $\exists d_x \in \mathbb{D}_H^k \ d_x = d_U^k$, and the users confirm it by selecting it from the list given in our mobile application, or the users enter a destination in the input box as shown in Figure 3.9(a), we denote this case as $d_U^k \in \mathbb{D}_H^k$. In this case, our prediction module adopts d_U^k as the final destination result, i.e., $\hat{d}_k \leftarrow d_U^k$, and calculates the deposit accordingly. Then, the system proceeds directly to the post-journey stage. Mathematically, the deposit for trip $\mathbb{T}_k$ is determined by the timestamp t_U^k when the user arrives at d_H^k, estimated by maps API.

Case 2: If none of the predicted results in $\mathbb{D}_H^k$ matches d_U^k, i.e., $\forall d_x \in \mathbb{D}_H^k$ $d_x \neq d_U^k$, or if the user disregards the recommendation in this stage, we denote this case as $d_U^k \notin \mathbb{D}_H^k$. In this scenario, the system estimates the expected deposit value based on the user's historical destinations $\mathbb{D}$ and proceeds to the in-journey stage. To estimate the expected deposit, we define $\mathbb{D}_k$ as the set of destinations for all trips in the user's historical record starting from the origin p_1^k, so $\mathbb{D}_k \subset \mathbb{D}$. Let D_k denote the number of elements in $\mathbb{D}_{p_1^k}$, so we have $D_k \leq D$. For any destination $d_l^k \in \mathbb{D}_k$, the arrival timestamp is represented by t_l^k. We use N_l to denote the total number of trips starting at p_1^k, and N_l^k to denote the number of trips starting at p_1^k and ending at d_l^k. Therefore, the weight w_l^k of destination d_l^k is calculated as:

$$w_l^k = \frac{N_l^k}{N_l} \tag{3.9}$$

where $\sum_{l=1}^{D_k} w_l^k = 1$, so the deposit for trip $\mathbb{T}_k$ is determined by the timestamps of arrival at all destinations in $\mathbb{D}_k$ and their corresponding weights.

To sum up, the equation for deposit calculation is given by

$$F_{deposit} = \begin{cases} f \cdot (t_U^k - t_1^k) & \text{if } d_U^k \in \mathbb{D}_H^k \\ f \cdot \sum_{l=1}^{D_k} w_l^k \cdot (t_l^k - t_1^k) & \text{if } d_U^k \notin \mathbb{D}_H^k \end{cases} \tag{3.10}$$

where f is the same constant value used in (3.8) representing the charge per second.

In-journey

During this stage, the continuous GPS coordinates of the E-Bike are used to predict the trip destination. Predicted trip destinations d_G^k are recommended to the user at specific intervals and saved in a sorted list $\mathbb{D}_G^k$, based on the frequency of occurrences. Users can customise recommendation frequency, confirm destinations at any time, or choose to ignore the recommendations while cycling. When users decide to end their journey and make a payment, they click the corresponding button in the mobile app. Three cases need to be considered in this step, depending on user responses.

Case 1: If any predicted result d_y in $\mathbb{D}_G^k$ matches d_U^k, i.e., $\exists d_y \in \mathbb{D}_G^k \; d_y = d_U^k$, and the user selects it from the list in our mobile application, as shown in Figure 3.9(b), we denote this case as $d_U^k \in \mathbb{D}_G^k$. In this case, our prediction module adopts d_U^k as the final result, i.e., $\hat{d}_k \leftarrow d_U^k$. The system enters the post-journey stage without further destination prediction.

Case 2: If the user ignores all the recommendations and is not ready to end the trip, we denote this case as $\nexists d_U^k$. The system will use the result at the top of $\mathbb{D}_G^k$, which is the prediction with the highest occurrence value, e.g., a set of predicted destinations followed by the number of their occurrence in Figure 3.9(b). Then, U-Park will begin the post-journey stage but continue tracking user responses to the predictions until an end command is issued.

Case 3: If the user does not choose any predicted destination but is ready to end the trip by clicking the "End Journey" button shown in Figure 3.9(b), we denote this case as $d_U^k \notin \mathbb{D}_G^k$, and set the user's current location p_{-1}^k as the final result, i.e., $\hat{d}_k \leftarrow p_{-1}^k$. The system will enter the post-journey stage.

Algorithm 1 demonstrates the workflow of our destination prediction module,

in which HistoryBasedModel(.) and TrajectoryBasedModel(.) represent the prediction models introduced earlier in this section. They are used to compute the predicted destination results based on trip history and trip trajectory, respectively. Specifically, if a user enters a desired destination, as shown in Algorithm 1, U-Park will adopt user input as the final destination result, skip destination prediction, and proceed directly to the post-journey stage.

Algorithm 1 Workflow of destination predictions

Input: An ongoing trip $\mathbb{T}_k = [(t_1^k, p_1^k); \dots]$, extra features f_k, distance threshold dis, (user-planned destination d_U^k)
Output: Prediction of destination $\hat{d}_k$

```
if d_U^k is initially given then
    d̂_k ← d_U^k
    return d̂_k
end if
while length(T_k) = 1 do
    D_H^k ← HistoryBasedModel(T_k)
    if ∃d_x ∈ D_H^k d_x = d_U^k (i.e., d_U^k ∈ D_H^k) then
        d̂_k ← d_x
        return d̂_k
    end if
    wait for more trip data to be added
end while
while (UserStop(T_k) = False) & (d̂_k = nothing) do
    D_G^k ← TrajectoryBasedModel(T_k)
    if ∃d_y ∈ D_G^k d_y = d_U^k then
        d̂_k ← d_y
        return d̂_k
    else if UserStop(T_k) = True then
        d̂_k ← p_{-1}^k
        return d̂_k
    end if
end while
```

Post-journey

When a user confirms a destination predicted by U-Park or the E-Bike's location approaches the final predicted destination $\hat{d}$, within a distance of dis metres, our final prediction module estimates parking space availability at nearby stations. This prediction is based on destination location, weather conditions, and estimated arrival time, as provided by the map API. The results represent the availability of parking docks or spaces, helping to decide whether a particular station is feasible to recommend for parking. For instance, if a prediction suggests more than 5 available

parking positions upon arrival, with an MAE of less than 1 [137], we are more confident in recommending that station, as it implies the user is likely to find suitable parking without difficulty. However, if the predictions indicate that there are fewer than, for example, 2 available parking positions, it implies that the user may have difficulty finding parking at their intended destination. In this case, U-Park will identify "available stations" based on the predicted results of nearby stations. An "available station" is defined as a station s_u where the parking space availability E_u exceeds a predefined threshold E. In Figure 3.10, we demonstrate the procedure for determining parking station $\hat{s_k}$ based on the probability distribution.

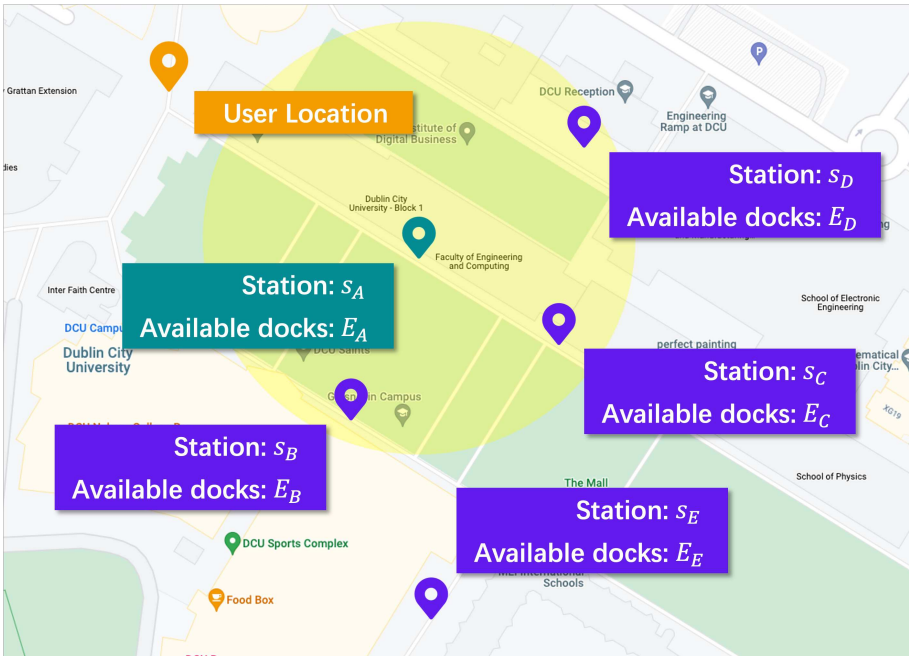


Figure 3.10: Parking station determination.

In the scenario shown in Figure 3.10, Station s_A represents the parking station at $\hat{d}$, with an associated available parking space denoted as $E_{\hat{d}} = E_A$. Stations s_B, s_C and s_D are the closest available stations to Station s_A, i.e., within the region of the yellow circle of radius m metres. In contrast, Station s_E lies outside the region and is not factored into the decision-making process. Formally, we express this as $\forall E_v > E$, where Station $s_v \in \{s_B, s_C, s_D\}$. In this scenario, we assume that $E_A < E$, indicating that Station s_A is not available for parking. Thus, the parking station recommended to the user is chosen from only Stations s_B, s_C and s_D. The probability of recommending Station s_v to this user can be calculated by (3.11). For

instance, the probability of Station s_C being recommended to the user in this case is given by $P_C = E_C/(E_B + E_C + E_D)$.

$$P_v = \frac{E_v}{E_B + E_C + E_D} \qquad v \in \{B, C, D\} \tag{3.11}$$

The parking station recommendation method based on the probability works better than simply advising every cyclist with the same destination to park at Station s_v because it avoids the rapid change in parking availability at Station s_v caused by the same recommendations. Instead, it helps balance the parking availability within a specific area.

3.5.4 Trading Module & Database

At the beginning of a trip, the trading module charges the user for the trip deposit as per the result of the prediction module, and users are billed for trip duration and parking behaviour at the end of the trip. For instance, parking at suggested stations may earn users rewards. Data from the hardware, user interface, and trading modules is stored in a database, enabling interactions among these components.

3.6 Experiments & System Implementation

In this section, we introduce and provide an overview of the implementation of the proposed system, U-Park, along with the experiments we conducted based on the multimodal data detailed above as a use case.

3.6.1 Hardware Module

The hardware section is implemented using various sensors connected to the E-Bike via a central controller, the *Raspberry Pi 3 Model B*, which is capable of processing the physical status information collected by the sensors, including GPS time series and digital lock status of the E-Bike. The GPS time series, required for our prediction module, is then transmitted to our database for further use, while the digital

lock status is used for interactions between the mobile application and the Raspberry Pi to lock or unlock an E-Bike. Specifically, the E-Bike's geographic location is identified by *NEO-6M* GPS module, and its axial position is captured by the *MPU6050* gyroscope and accelerometer module.

3.6.2 User Interface Module

Our Android mobile application is developed using *IntelliJ IDEA* and installed and tested on an *OnePlus 7pro* with a *Snapdragon*^{TM} *855* processor. With Google Maps API[11], the interactive map is implemented, presenting the locations of the user and E-Bikes, as well as predefined station locations with circles representing their parking areas and digits showing parking availability in real-time. In Figure 3.11, the orange marker represents the current user location, obtained via the Android locating service. The green marker shows the current location of an E-Bike. Circular areas filled with different colours define parking zones with different parking fees, and the digit within each parking zone denotes its current parking availability. The size of parking zones is dynamically scaled to account for changes in GPS positioning accuracy in different areas for optimising performance in non-open areas.

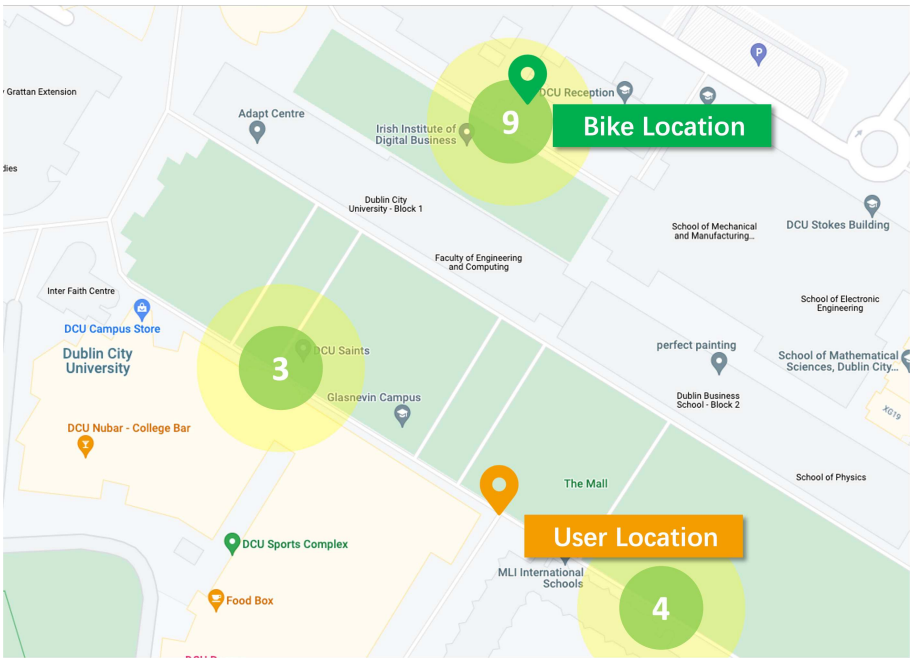


Figure 3.11: User location, E-Bike position and predefined parking zones.

[11] `https://developers.google.com/maps/documentation/directions`

The QR code scanning function, implemented by the *ZXing* library[12], captures the E-Bike's Unique Identification Number (UID), GPS coordinates, and current timestamp. The E-Bike's UID serves for management and maintenance, while the others are sent to our prediction module to predict the destination, which is subsequently recommended to the user for confirmation, and to examine whether the remaining battery power is enough for the predicted trip. If not, U-Park waits for the user's response and suggests an alternative E-Bike nearby that has sufficient battery power. Conversely, if the battery power is enough, the predicted destination is used to calculate the trip deposit. The calculated deposit and the user's real-time credit are then transmitted to the database for validation. Data transmission between the user-end and the trading module is completed based on *Web3j* API[13].

We used *DynamoDB* for data storage and hardware testing. Interactions with *DynamoDB* are facilitated through the *Amplify* framework provided by *Amazon Web Services*[14] for database management. The GPS data contains trip details, including trip ID, latitude, longitude, and timestamp. Additionally, we add extra information when a trip ends, including the moving duration, moving fee, minimum distance to stations, and parking fee. This enables us to validate trip deposits and balances. Trip records also include rental and return timestamps, along with GPS coordinates and physical attributes. A statistical description of this dataset is shown in Table 3.4.

Table 3.4: Statistical description of test trip report collected.

Object	**t (s)**	$\mathbf{F}_m$	**d (m)**	$\mathbf{F}_p$
mean	83.50	0.0025	36.16	0.0636
std	89.32	0.0027	33.27	0.0414
min	11.00	0.0003	1.94	0.0000
25%	24.00	0.0007	10.61	0.0500
50%	30.50	0.0009	20.35	0.0750
75%	112.50	0.0034	56.75	0.1000
max	318.00	0.0095	95.19	0.1000

t (s) – Trip Duration in Seconds, $\mathbf{F}_m$ – Moving Fee, **d (m)** – Parking Distance in metres, $\mathbf{F}_p$ – Parking Fee.

[12] `https://github.com/dm77/barcodescanner`
[13] `https://docs.web3j.io/4.10.0/`
[14] `https://aws.amazon.com/dynamodb/`

3.6.3 Trading Module & Database

Since this work focuses more on the design of the recommendation system, the trading module and database can be realised using various technologies depending on specific project requirements. Therefore, we will only discuss the management of user credits in this section.

According to the incentive and punishment policy we introduced in Section 3.2, the parking fee $F_{parking}$ mentioned in Section 3.5 and applied in U-Park is determined by a tiered fee approach, as shown in (3.12), where D_1 and D_2 are two distance thresholds and D_{min} represents the minimum distance, measured in metres, between the E-Bike's current GPS location and the locations of all available stations. $F_{penalty}$ is the penalty imposed for the failure to park the E-Bike appropriately.

$$F_{parking} = \begin{cases} 0 & D_{min} \leq D_1 \\ 0.5 \times F_{penalty} & D_1 < D_{min} \leq D_2 \\ F_{penalty} & D_{min} > D_2 \end{cases} \tag{3.12}$$

According to [197], the lowest mean walking speed indicates that a 50-metre commute takes less than 1 minute, leading to a reward of €0.2 in our study. Considering potential user heterogeneity, we set D_1 and D_2 to 50 and 100 metres, respectively. As an illustrative example, this choice results in corresponding penalties for user credits. To clarify, parking within the green circular zones in Figure 3.11 means that $D_{min} < 50$, leading to a parking penalty $F_{parking}$ of 0 for this trip. Parking within the yellow circular zones leads to half of the penalty while parking outside results in the full penalty charge.

3.6.4 Prediction Module

As outlined in Section 3.5, our prediction module consists of three individual stages: destination prediction based on trip history, destination prediction based on partial trajectory, and parking space availability prediction based on station history.

Each prediction task is addressed by various ML models. Initially, features of the data used in our system only included trip origin and destination labels, starting time, i.e., rental time, and hourly rental demand at the starting station (derived from statistics). After adding in meteorology data (e.g., precipitation amount and air temperature), the modified dataset, presented in Table 3.5, was used in the experiment section.

Table 3.5: Sample data of trip records from the modified dataset.

O	**D**	**W**	**H**	**R**	**PA**	**AT**	**WBT**	**DPT**	**VP**	**MSLP**	**Road segment**
215	199	1	16	2	0.0	21.0	18.4	16.7	19.1	1019.2	..., Stillorgan Road, ...
215	214	5	14	2	0.0	5.9	4.7	3.1	7.6	1009.6	..., Stillorgan Road, ...
261	261	0	20	2	0.0	13.5	7.9	0.1	6.2	1016.9	..., Queen's Road, ...
217	261	5	19	1	0.0	9.5	7.5	5.1	8.8	1006.8	..., Harbour Road, ...
261	210	2	13	1	0.0	18.4	14.8	12.0	13.9	1024.8	..., Idrone Lane, ...

O – Rental ID, **D** – Return ID, **W** – Rental Weekday, **H** – Rental Hour, **R** – Hourly rentals, **PA** – Precipitation Amount (mm), **AT** – Air Temperature (C), **WBT** – Wet Bulb Temperature (C), **DPT** – Dew Point Temperature (C), **VP** – Vapour Pressure (hPa), **MSLP** – Mean Sea Level Pressure (hPa).

History-Based Destination Prediction

In our model, the dataset is divided into training, validation, and test sets with an 8:1:1 ratio, and a basic ANN model is implemented including two hidden layers with 64 and 32 units, respectively, and a dropout layer between them. The number of units in each layer is optimised by the Hyperband algorithm [198], and they may vary when applied to other users' trip records. The loss function used is Categorical Cross Entropy (CCE), and the model performance is evaluated by Categorical Top-3 and Top-1 Accuracy metrics. To simulate the absence of certain features, we experimented with different combinations of feature sets in our work. The feature sets adopted are as follows: (1) rental location ID (Init); (2) rental hourly demand (D); (3) rental hour and rental weekday (T); (4) precipitation amount, air temperature, wet bulb temperature, dew point temperature, vapour pressure, and mean sea level pressure (W).

Trajectory-Based Destination Prediction

Based on road segments transformed from the partial trajectory, we split the dataset into training data, validation data, and test data with an 8:1:1 ratio, and trained

several models to predict the destination candidate.

Simple RNN: A simple RNN model is implemented with a structure including a dense layer with 64 units, a dropout layer, and a dense layer with 32 units.

ARNN: An Attention-based Recurrent Neural Networks (ARNN) model is implemented with a structure including a dense layer with 64 units, a Bahdanau attention layer, a dropout layer, and a dense layer with 32 units.

LSTM: An LSTM model is implemented with a structure including an LSTM layer with 128 units, a dropout layer, and a dense layer with 32 units.

ALSTM: An Attention-based Long Short-Term Memory (ALSTM) model is implemented with a structure including an LSTM layer with 128 units, a Bahdanau attention layer, a dropout layer, and a dense layer with 32 units.

The number of units in each layer is also tuned by the Hyperband algorithm. Specifically, for trips with a length of $L_k > (50 + 10 - 1)$, we randomly select 50 numbers smaller than $(L_k - 4)$ and accordingly extracted 50 samples starting from the trip origin. Each sample is processed by considering only the first 4 and the last 4 road segments to create a sub-trajectory for this trip. For instance, for a trip $\mathbb{T}_k$ of which the length is 70, if a generated random number is 20, we take the first 20 points (i.e., $[(t_1^k, p_1^k), \ldots, (t_{20}^k, p_{20}^k)]$) as one sample, and only use the first 4 (i.e., $[(t_1^k, p_1^k), \ldots, (t_4^k, p_4^k)]$) and last 4 points (i.e., $[(t_{17}^k, p_{17}^k), \ldots, (t_{20}^k, p_{20}^k)]$) to produce a sub-trajectory. For the selected user, 46 trips were suitable for this prediction task, resulting in 2,300 sub-trajectories containing the first four and last four road segment labels. We tested different feature combinations for this model. The initial feature combination (Init) is only a serial of road segment labels. Subsequently, we added temporal attributes (T) and weather conditions (W) as features. The loss function used is CCE, and the model's performance is assessed using Categorical Accuracy.

Parking Spot Prediction & Recommendation

According to the histogram of parking space demonstrated in Figure 3.6, we approximate that the number of available spots follows a normal distribution, and then set

up the baseline model for our study as follows: Initially, we model the actual number of parking spaces at each station using a normal distribution, i.e., $N \sim \mathcal{N}(\mu, \sigma^2)$, where μ is the average number of available spots and σ represents its standard deviation. Next, we compute the probability that the simulated number of vacancies exceeds zero for each simulation run. Lastly, this process is repeated, ten thousand times in our case study, to derive the average probability of obtaining at least one available parking space.

Following the same experiment setup as in ASTGCN [71], we use the data on a similar of available spots at each E-Bike station over the initial 3-hour period to predict the parking space availability 45 minutes later. Specifically, each data point represents the average number of available spots calculated over 15-minute intervals, so the historical data length is $3 \times 60 \div 15 = 12$, while the prediction length is $45 \div 15 = 3$. The dataset is divided into training, validation, and test sets with a 6:2:2 ratio. Features used include the number of available spots, time of day, weekday, and weather condition description. The model is optimised by the Adam algorithm, and other parameters, such as learning rate, are set the same as introduced in [71].

After obtaining the prediction results, we optimise our recommendation based on the strategy introduced in Section 3.5 which considers the parking availability within a specific area rather than merely one parking station. The distance threshold *dis* used in our experiments is 300, which means U-Park will find available stations as recommendation candidates within 300 metres of the destination and provide parking suggestions based on predicted parking spots.

3.7 Results & Discussion

3.7.1 Feature Selection

We introduced various feature combinations in the previous section. To reduce the influence of random factors, we conducted 10 evaluations for each combination and

recorded the mean values of each metric in Figure 3.12 and Table 3.6. We present the results for five users for case-specific analysis. However, the outcomes may vary when considering other users.

History-based Prediction Model

It is evident from Figure 3.12 that the Top-3 accuracy consistently outperforms the Top-1 accuracy across all users and feature sets. Moreover, we found that including additional features beyond the initial rental location ID significantly improved model performance. For instance, for User 2, Top-3 and Top-1 accuracy increased by approximately 11% and 31%, respectively, when incorporating demand (d) and weather-related (w) features. However, the optimal feature set can vary depending on the user. Specifically, User 4 achieved the highest Top-1 accuracy (around 67%) with a combination of rental location ID (init), hourly demand (d), and weather-related features (w).

Trajectory-based Prediction Model

Table 3.6 also demonstrates that the performance of the trajectory-based model varies with different feature sets. Different to the history-based model, a combination of the initial input (road segment labels), time (t), and weather-relevant (w) attributes consistently achieved the highest accuracy across all selected users. Moreover, the trajectory-based model significantly outperforms the history-based model, reaching 100% accuracy for the majority of users and achieving 97.33% even for the relatively challenging case of User 2.

Table 3.6: Trajectory-based results with different features.

User ID	**init**	**t**	**w**	**tw**
User 1	0.8875	1.0000	1.0000	1.0000
User 2	0.8907	0.9733	0.9667	0.9720
User 3	0.7133	1.0000	1.0000	1.0000
User 4	0.9592	1.0000	1.0000	1.0000
User 5	0.6750	1.0000	1.0000	1.0000
User_{avg}	0.8251	0.9947	0.9933	0.9944

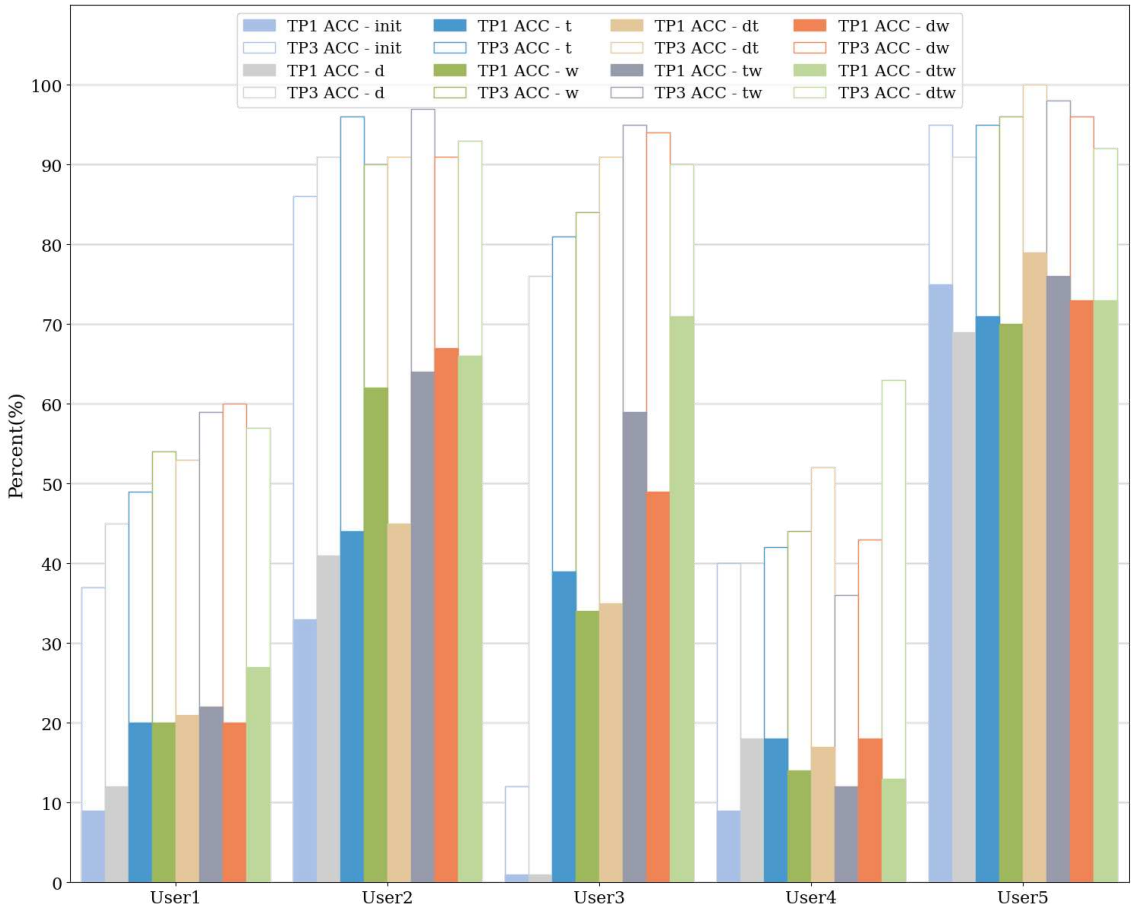


Figure 3.12: History-based results with different features.

3.7.2 Model Comparison for Trajectory-based Prediction

Figure 3.13 Compares the performance of various trajectory-based models, including simple RNN, ARNN, LSTM, ALSTM). It is obvious that incorporating trajectory information can significantly improve system performance, with accuracy improvements ranging from 29% (User 1) to 82% (User 4) compared to the model based on history. The significant differences across users suggests that individual travel behaviours have a strong impact on prediction performance. Users with consistent and repetitive travel patterns, such as User 4, will enable the model to learn stable trajectory-to-destination mappings, resulting in greater benefits from sequential modelling. In contrast, users like User 1, whose trips involve more diverse destinations and irregular routes, constrain the model's ability to generalise and therefore lead to smaller improvements. Interestingly, different ML models exhibit varying effectiveness among different users. While all models achieve 100% accuracy for certain users with consistent travel patterns, the best model may vary based on

individual user behaviour.

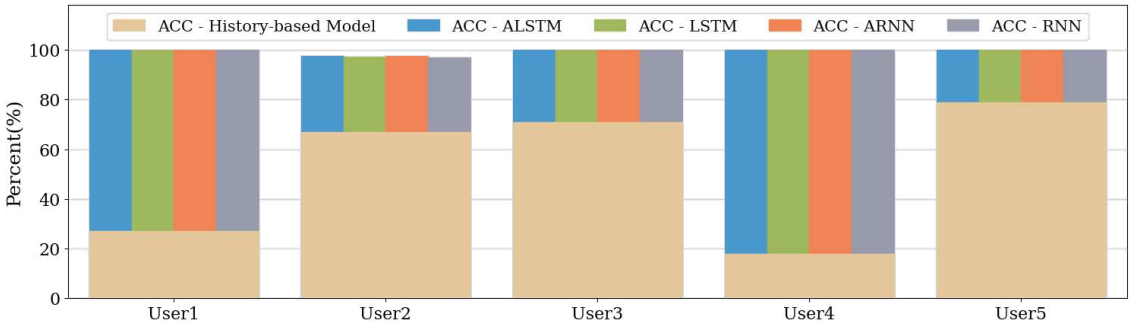


Figure 3.13: Trajectory-based results with different models.

3.7.3 Selection of K-value

As mentioned above, the first k and last k points in the journey play a significant role in the prediction results. Thus, we conducted experiments to assess the impact of different values of k, i.e., input size, and summarised the results in Table 3.7, from which we found that the highest prediction accuracy was achieved when $k = 4$. Consequently, the optimal choice for k in this prediction task is 4, which balances the prediction accuracy and input size.

Table 3.7: Impact of input size on prediction results.

K Value	1	2	3	4	5	6	7	8	9	10
Test Loss	0.0856	0.1027	0.1113	**0.0654**	0.0828	0.0804	0.0694	0.0977	0.1069	0.1057
Test Accuracy	0.9698	0.9609	0.9478	**0.9789**	0.9667	0.9774	0.9763	0.9641	0.9618	0.9571

3.7.4 Parking Space Availability Prediction & Impact

We illustrate the performance of our parking space availability prediction with examples from five parking stations in Table 3.8, which includes each station's location (latitude and longitude), basic statistical information of available spots (average value and standard deviations), as well as different MAE and Root Mean Squared Error (RMSE) metrics based on the predictor used.

To best illustrate the efficacy of the proposed system, we evaluated the probability of finding a parking spot for five real users across three different settings: (a) without any recommendation system as a baseline (*User X baseline*); (b) using the parking availability predictor (*User X predicted*); and (c) using our U-Park system

Table 3.8: Performance (MAE & RMSE) of parking spot prediction.

Spot	**Latitude**	**Longitude**	**Available Spots**		**Prediction Results**	
			mean	std	MAE	RMSE
1	53.3569	-6.2680	14.4737	4.7863	0.8880	1.3993
2	53.3511	-6.2700	14.4107	4.6208	1.0018	1.5732
5	53.3431	-6.2707	14.3387	5.6082	0.8363	1.3962
6	53.3430	-6.2733	23.7705	5.0778	0.8853	1.7610
8	53.3420	-6.2648	13.2831	8.3851	2.0066	3.2325

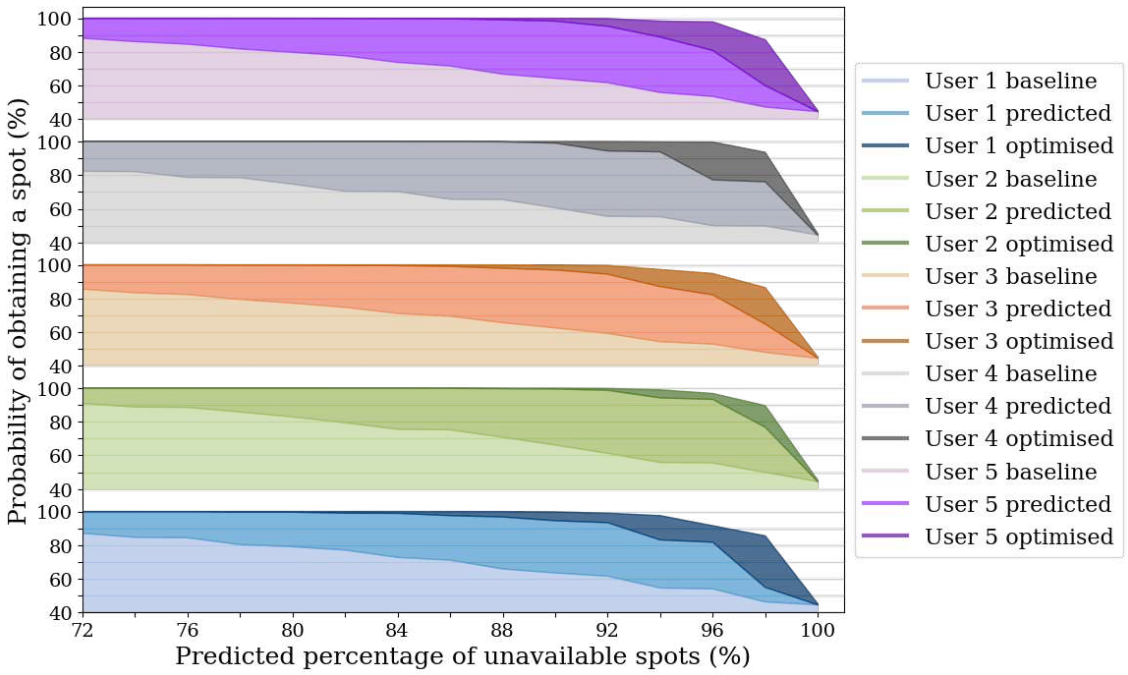


Figure 3.14: Impact of parking space prediction and optimisation.

(*User X optimised*), which recommends the station with the highest likelihood of finding a spot near the user's destination. Our key findings are shown in Figure 3.14. Specifically, for a given predicted percentage of unavailable spots, we calculate the probability of finding a parking spot in each of the three settings for all five users. The probabilities vary by user, as they are averaged based on historical data from all stations each user has visited. Furthermore, our results demonstrate that the prediction module can significantly help improve the probability of all users obtaining a parking spot compared to the baseline result, thanks to its high prediction accuracy, i.e., low MAE. U-Park further optimises performance over the predicted results, especially when parking spot unavailability ranges from 88% to 96%, leading to an average improvement of 24.91% and a peak of 29.66% across the five real users analysed. Clearly, this demonstrates the efficacy of our system in real-world scenarios, especially where parking availability is limited.

3.8 Limitations

Our framework proposed in this chapter represents the initial stage of developing smart parking management for SMS. Although our system has been tested using real-world data, it has not been implemented in a practical context. The final prediction stage, focusing on parking space availability for docked SMS, relies heavily on real-world data from shared E-Bike stations, which we had to approximate with comparable datasets during the study as an alternative. Reliable and comprehensive data are crucial for accurate predictions, but acquiring open data sources with detailed GPS time series and parking space information at each station remains a challenge, as such information is typically held by service providers. Incorporating an automated mechanism for model selection and hyperparameter optimisation would enhance our system. Our current work has primarily utilised simple ML models due to the limited dataset size, as more complex models could lead to overfitting. Access to larger and more diverse datasets, along with advanced ML models, is expected to substantially enhance the performance and universality of our recommendation

system within the existing framework. Besides, given that most E-Bike docking sites are generally located in open areas in our case study in Dublin, Ireland, our analysis was based on the assumption that it applies to these environments. Therefore, we did not specifically address non-open areas in our system.

3.9 Summary

Table 3.9: Summary of travel demand & destination prediction methods.

Study	Method	Prediction Target	Data Used
[133]	Regression models	E-Bike demand	E-Bike trip records, regional context, weather data, etc.
[134]	GTWR	E-Bike demand	E-bike trip records, regional context, weather data, etc
[45]	SARIMA, GARCH	E-Scooter demand	E-Scooter trip records.
[65]	Conv-LSTM	Bicycle demand	Bicycle trip records
[135]	MFCN	E-Scooter demand	E-Scooter trip records
[136]	LSGC-LSTM	Bicycle demand	Bicycle trip records
[73]	STGCN	Bicycle demand	Bicycle trip records
[137]	GGNN	Bicycle demand	Bicycle trip records
[44]	HC-LSTM	Bicycle demand	Bicycle trip records, regional context, weather data, etc
[66]	DPNst	Bicycle destination	Bicycle trip records
[177]	XGB, ANN	Bicycle destination	Bicycle trip records, regional context, etc
[70]	Conv-LSTM	Trajectory path	Walk, bike, bus and car GPS records, weather data, etc
[178]	HMM	Car destination	Car GPS records
[49]	MDLN	Car destination	Car GPS records
[193]	ARNN	Car destination	Car GPS records, regional context, etc
This work	(A)RNN, (A)LSTM	Bicycle demand & destination	Bicycle trip records, bicycle GPS records, weather data, etc

This chapter introduces U-Park, a smart parking management solution for SMS to enhance parking prediction and user experience based on multimodal data. We outlined its architecture, integrating multiple data sources, including historical trip records, real-time GPS trajectories, parking space availability, and environmental factors, across three predictive stages to provide personalised and adaptive parking recommendations. For the prediction of trip destinations in the first two stages, we

significantly improved accuracy to over 97.33% by employing a partial trajectory-based ARNN model, which dynamically refines predictions from a trip history-based model. Additionally, our parking space availability prediction model empowers U-Park to suggest the optimal parking station to users based on the availability of predicted parking spaces, improving the probability of obtaining a parking spot by 24.91% on average and 29.66% on maximum when parking availability is limited. To contextualise the adopted approach, a summary of the methods and data types used in this study compared with existing work is provided in Table 3.9.

Chapter 4

Privacy-Aware Energy Consumption Prediction

BEVs are an increasingly important component of SMS, offering cleaner alternatives to conventional transport. However, accurately estimating their energy consumption remains a key challenge, particularly when considering diverse real-world factors such as environmental conditions, driving behaviour, and traffic dynamics. This chapter addresses the second major challenge identified in Chapter 1, i.e., how to model shared BEV energy consumption using multimodal data while preserving user privacy. We introduce a FL-based approach enabling decentralised privacy-aware energy prediction without compromising model performance. This work is conducted in collaboration with Hongyuan Fang, Ji Li, Tomas E. Ward, Noel E. O'Connor, and Mingming Liu, and published in [33].

4.1 Introduction

The transition from conventional internal combustion engine vehicles to EVs has emerged as a major global initiative due to increasing concerns related to the environment and public health. Governments worldwide are introducing policies, laws, and regulations to promote EVs. For instance, the Members of the European Parliament recently approved regulations to encourage the production of zero- and

low-emission vehicles, such as EVs and plug-in hybrids[1]. In Ireland, the Zero Emission Vehicles Ireland office has been established to switch an expected 30% of the private car fleet to electric by 2030[2]. Furthermore, numerous studies have indicated a significant reduction in air pollution, noise pollution, and dependence on fossil fuels [199], [200], [201] resulting from the adoption of EVs.

Energy modelling is essential for policymakers to assess the impact on the grid, identify necessary infrastructure requirements based on vehicle-to-grid technology, and inform energy policy to support the transition to clean transportation to enhance the overall driving experience while mitigating transportation's environmental impact [202], [203]. It is beneficial to automakers and companies in understanding the performance of EVs and optimising the energy usage patterns, leading to improved battery management and energy efficiency [204], [205]. Besides, accurate predictions of energy consumption can enable drivers to make informed decisions regarding trip planning, route selection, and charging strategies [149], [206].

To this end, various methods have been applied in this research field, ranging from system dynamics models to statistical and data-driven models [43], [53], [59], [67], [143], [144], [207]. However, with the increasing volume of multimodal data generated by connected vehicles, conventional energy modelling methods cannot often efficiently manage the produced data. Specifically, these methods often require data collection from various sources to create and train a centralised model. This centralised approach can introduce challenges in data management, such as data transfer and the risk of data leakage. Instead, FL offers a promising solution that enables vehicles to share local models rather than raw data [208] in a decentralised manner for better data privacy, security, and scalability. While FL-based methods may demand increased computational resources and more robust communication infrastructure [209], their superior learning capabilities can generally lead to improved model performance in accuracy, precision, and generalisability for many real-world

[1] https://www.europarl.europa.eu/news/en/press-room/20230210IPR74715/fit-for-55-zero-co2-emissions-for-new-cars-and-vans-in-2035

[2] https://www.gov.ie/en/campaigns/18b95-zero-emission-vehicles-ireland/

applications [210], [211], [212], [213], [214], [215], [216].

In our preliminary study [138], FL methods show great potential to deal with privacy issues of connected vehicles. Edge-cloud computing is believed to be the key to meeting all challenges in big transportation data [217], [218], [219], [220]. Based on these foundations, our research aims to achieve energy consumption modelling for EVs based on FL methods and investigate the effectiveness of models together with edge-cloud computing techniques. By evaluating and comparing the performance of different models for BEV energy consumption prediction with various configurations, we provide recommendations for their practical implementation in different real-world scenarios. The main contributions of this work are outlined below:

- An extensive review of existing methods employed in previous research focusing on energy consumption modelling for EVs.
- A comparative study is conducted on five local model candidates and five FL algorithms to identify the optimal combination for BEV energy consumption modelling.
- The performances are evaluated for the models with different setups, including the number of iterations, data splitting ratio, and input data size.
- By leveraging distributed data sources and FL, the proposed method offers the potential to overcome data privacy concerns and enhance the accuracy and efficiency of energy modelling for EVs.
- The proposed method is discussed in two centralised and decentralised edge-cloud computing structures for BEV energy consumption modelling. This enables more accurate predictions of energy consumption in various real-world scenarios.

The structure of this chapter is as follows: In Section 4.2, we provide a review of the relevant literature on energy consumption modelling for EVs and FL algorithms and their applications. The research question and object are explained in detail in

Section 4.3. Section 4.4 provides the description and analysis of our dataset, and Section 4.5 describes the methodology employed in our work. Section 4.6 presents the experiment setups and corresponding results, and relevant discussions and limitations are included in Section 4.7. Finally, we conclude our work in Section 4.8 and discuss future plans and potential improvements.

4.2 Related Work

In this section, a comprehensive overview of the literature on energy consumption modelling for EVs is provided first. Secondly, an introduction to FL algorithms is presented with some applications. The unique advantage of FL algorithms in addressing privacy concerns is discussed at the end.

4.2.1 Energy Consumption Modelling

The literature on estimating energy consumption modelling for EVs has used three distinct categories of methods: white box, grey box, and black box. This section aims to provide a brief introduction to white and grey box methods while focusing primarily on black box methods.

White Box Methods

Methods that integrate physical constraints into their modelling process, known as white box methods, are invariably built upon theories of vehicle dynamics. These approaches typically estimate energy consumption by employing vehicle dynamics principles with data obtained from a single vehicle. For instance, [221] explores power relationships with velocity, acceleration, and road grade for EVs through statistical analysis and an analytical model based on vehicle dynamics principles. The data used in this study, including battery status, speed, acceleration, and position, was collected from a single test vehicle. For readers interested in further exploration of this topic, we recommend referring to the studies in [141], [222], [223], [224], [225],

[226], [227].

Grey Box Methods

Researchers have applied grey box methods to estimate energy consumption by combining physical knowledge with data fitting techniques. These methods, capable of handling larger datasets through data fitting techniques, primarily focus on the input and output of models using data collected from individual vehicles. For example, the authors of [228] investigated energy consumption modelling for EVs using data collected from custom-designed EVs. Their approach blends physics-based equations with empirical data, resulting in a hybrid power model that offers improved energy consumption estimations and addresses range anxiety concerns. Furthermore, statistical techniques such as the Monte Carlo method [229], multinomial logistic regression [230], and Gaussian process regression [231] have been utilised for similar purposes. For those interested in further exploring this topic, we recommend referring to the studies mentioned in [142], [232], [233], [234].

Black Box Methods

Instead of physical knowledge, black box methods use data fitting techniques, such as ML models, because of their capacity to make predictions and analyses of datasets. Some classic ML models, such as Support Vector Machine [59], Decision Tree and RF [53], have been applied and estimated in previous relevant work. Furthermore, DL methods such as ANN [43] and CNN have also been employed to solve the same problem. Methods mentioned above are widely used to estimate energy consumption for EVs based on clustered data collected from multiple vehicles. For instance, in [144], the authors employed XGB to forecast energy consumption based on the real-world data collected from 55 electric taxis from 2017 to 2018 in Beijing, China. The method demonstrated an impressive performance with a RMSE of 0.159 kWh. For further information, we refer interested readers to the explanations included in [43], [53], [59], [67], [143], [144], [207].

4.2.2 Privacy-Aware in Transportation Data

The utilisation of transportation data has gained significant traction in both research and industry applications. However, the integration of transportation data without the consideration for privacy can give rise to various risks and consequences. Recent scholarly investigations [217], [218], [219], [220] have elucidated and examined the potential risks associated with the usage of big transportation data, including privacy infringement, data breaches, misuse, and the erosion of user trust. For instance, direct usage of source data may expose sensitive information. [220] highlighted how the divulgence of origin-destination information of individual users can compromise their privacy, allowing malevolent actors to deduce their residential or workplace locations. Another pertinent issue related to big transportation data exists in open datasets. As outlined in [217], by cross-referencing publicly available datasets from the New York City Taxi and Limousine Commission, the cash tips paid by celebrities could be discovered and released easily. The perception of data vulnerability and the potential for misuse or privacy breaches can undermine user trust, resulting in a hesitancy to share data or participate in related research endeavours.

Simultaneously, the studies referenced above have also highlighted the advantages of incorporating privacy-aware methodologies in the context of transportation data. These benefits can be summarised as follows:

- From the users' perspective, the implementation of privacy-aware approaches is paramount to safeguarding user data, ensuring that sensitive information such as personal identities, location data, and behavioural patterns are adequately protected. By prioritising privacy, users can have greater confidence in sharing their data and engaging in various services and platforms.

- From the operators' standpoint, adopting privacy-conscious practices is instrumental in building trust and cultivating a positive reputation among users. This, in turn, can foster increased user participation, collaboration, and support for initiatives that prioritise privacy protection. Operators who prioritise

user privacy are more likely to attract and retain a loyal user base.

- In terms of privacy regulations and ethical considerations, adhering to privacy-aware practices is crucial for organisations to comply with relevant privacy laws and regulations. By implementing robust privacy measures, organisations demonstrate their commitment to respecting individuals' rights to privacy and autonomy. Aligning with ethical principles promotes a responsible and trustworthy environment for data collection and usage.

4.2.3 Federated Learning

To reduce the computational complexity and enhance the privacy preservation of the centralised approach, FL was developed and recently it has received considerable attention thanks to its ability to enable multiple data holders, such as EVs, to collaboratively train models without the need to share their underlying data, thereby providing a natural safeguard against data privacy breaches [31], [235], [236]. As a subfield of ML, FL methods bring revolutionary changes to the conventional system identification process. Instead of processing the data which is previously collected from a single vehicle or clustered from multiple vehicles, FL methods enable each vehicle to collect data and train the individual model locally without raw data sharing, so it is unnecessary to anonymise the data as what conventional methods did. Besides, the capability of FL methods to deal with large-scale data makes it possible to update the trained model and make the decision in time rather than optimise the system performance after another round of data clustering, processing and model training.

In the seminal paper [237], the Federated Stochastic Gradient Descent (FedSGD) and Federated Averaging (FedAvg) algorithms were first proposed. A recent study [238] employed CNN and bidirectional LSTM networks as local models and implemented FedAvg for global optimisation. The study revealed that the performance of the model improved compared to the original local model. In another study [239], the authors proposed the Federated Proximal (FedProx) framework, which aims to

address the challenges posed by statistical and system heterogeneity in the context of FL.

The domain of EVs is characterised by significant variability in data distribution across different devices. Recent research has proposed novel approaches such as Federated Personalisation (FedPer) [240] and Federated Representation Learning (FedRep) [241] to address the challenges arising from this statistical heterogeneity. These methods introduce the concept of dividing the model into standardised layers, which perform weight sharing across the devices, and personalised layers, which are responsible for generating the final personalised output based on the local data of each device. These methods represent a promising strategy for mitigating the negative impacts of statistical heterogeneity in FL for EVs. Several survey papers, including [242] and [243], have listed various comparative experiments where different FL approaches (e.g., FedAvg and FedPer) with diverse local models, such as LSTM and GRU, have been implemented and compared.

Among the five FL algorithms previously discussed, i.e., FedSGD, FedAvg, FedProx, FedPer, and FedRep, each exhibits unique strengths within this domain. While some studies have compared specific pairs of these algorithms, a comprehensive comparison of all five is still warranted to offer a more thorough understanding of their respective advantages and limitations.

4.3 Problem Formulation & System Model

In this section, we present our research object in terms of the research problem statement, the specification of vehicles, and the introduction of energy consumption calculation. The problem statement part provides a comprehensive overview of the research objectives, while the vehicle specification outlines the parameters of the vehicles used in the experiment. Additionally, the last part talks about the benchmark for calculating the energy consumed in transit.

4.3.1 System Setup

We consider a scenario where a BEV company wishes to improve its customers' satisfaction and convenience in the aspect of battery capacity management by enhancing the performance of its energy consumption prediction model while respecting the customers' privacy (e.g., without sharing their trip detail data such as location, time, speed, or the status of the remaining charge of the vehicle battery). Thus, the aim of our research is to implement and improve the energy consumption prediction by keeping the collected private dataset locally but updating the ML models globally with the help of different FL methods. To do this, we assume that each BEV in the system is equipped with an onboard computing/communication unit, which is connected to our sensor group including various sensors capturing different trip attributes (e.g., velocity, trip distance, temperature and road slope) and is able to process the data collected, train a local ML model accordingly and communicate and transfer data with other sections.

Similar to the model setup presented in our previous work [138], we now formulate the BEV energy consumption prediction as follows. Let N be the number of BEVs in our system and $\mathbb{N} := \{1, 2, 3, \ldots, N\}$ be the set for indexing these BEVs. For a vehicle $i \in \mathbb{N}$, we assume the number of its trip records is K and denote its j^{th} trip by $\mathbb{T}_{i,j}$ where $j \in \{1, 2, \ldots, K\}$. If our sensor group captures the trip features M times in trip $\mathbb{T}_{i,j}$, the trip features will be divided into M sets accordingly. Thus, at a given sampling moment $t \in \{1, 2, \ldots, M\}$, the feature set is represented by $f_{i,j}^t$ and consequently, the trip $\mathbb{T}_{i,j}$ could be defined based on its feature sets as:

$$\mathbb{T}_{i,j} = [f_{i,j}^1, f_{i,j}^2, \ldots, f_{i,j}^M] \tag{4.1}$$

Assuming that the length of the trip feature set is L, we could denote the feature set at the sampling moment t in the trip $\mathbb{T}_{i,j}$ by:

$$f_{i,j}^t = [f_{i,j}^t(1), f_{i,j}^t(2), \ldots, f_{i,j}^t(L)] \tag{4.2}$$

where each $f_{i,j}^t(k)$ represents the k^{th} feature in the feature set $f_{i,j}^t$, so accordingly, all the K historical trip records of the vehicle i could be defined as a collection of all trips during the day as:

$$\mathbb{D}^i = [\mathbb{T}_{i,1}, \mathbb{T}_{i,2}, \ldots, \mathbb{T}_{i,K}] := [\mathbb{D}^i(1), \mathbb{D}^i(2), \ldots, \mathbb{D}^i(L)] \tag{4.3}$$

where $\mathbb{D}^i(k)$ is a column vector of $\mathbb{D}^i$ where only the k^{th} feature of the trip data is included. Assuming that the last feature, i.e., $\mathbb{D}^i(L)$, represents the energy consumption data E^i of the vehicle i, we could use the other columns in $\mathbb{D}^i$ as the input feature F^i for training the local model for the vehicle i, which could be defined as:

$$F^i := [\mathbb{D}^i(1), \mathbb{D}^i(2), \ldots, \mathbb{D}^i(L-1)] \tag{4.4}$$

and output label data, i.e., the energy consumption data of the vehicle i (E^i), could be defined as:

$$E^i := \mathbb{D}^i(L) \tag{4.5}$$

Finally, for a given time t, we define the past m observations from the input and output sets as $F_{t-m+1:t}^i$ and $E_{t-m+1:t}^i$, respectively. The output vector has a dimension of m, representing the energy consumption, while the input matrix has a dimension of $m \times (L-1)$, representing the feature set at each given time point with the past m steps in standard time units. Here, we consider the overall energy consumption data during the m steps instead of the point-wise data of the vector, so we define the overall energy consumption within the past m observations as:

$$\hat{E}_{t-m+1:t}^i := \mathbf{1}^T \cdot E_{t-m+1:t}^i \ \forall t \in \mathbb{K} \tag{4.6}$$

where $\mathbf{1} \in \mathbb{R}^m$ is the column vector with all entries equal to 1, and $\mathbb{K} := \{m, m+1, \ldots\}$ is a feasible indexing set.

Given the notation above, a local model training process for vehicle i can be

defined to find a local hypothesis function $\mathbf{H}_i(\cdot)$ which is able to address the following problem:

$$\begin{aligned} &\min_{\mathbf{H}_i} \sum_{t\in\mathbb{K}} \left| \hat{E}^i_{t-m+1:t} - \widetilde{E}^i_{t-m+1:t} \right| \\ &\text{s.t. } \widetilde{E}^i_{t-m+1:t} = \mathbf{H}_i(F^i_{t-m+1:t}) \end{aligned} \tag{4.7}$$

4.3.2 Research Object

The studied vehicles are considered the same type of passenger BEVs as shown in Figure 4.1. It is equipped with an electric motor of a maximum of 102 kW at 3000 rpm and a battery pack with *120s5p* cells for a total usable energy of 45 kWh and a maximum power of 160 kW. In order to maximise reliability, vehicle models are constructed through the industrial development software AVL Cruise M[3].

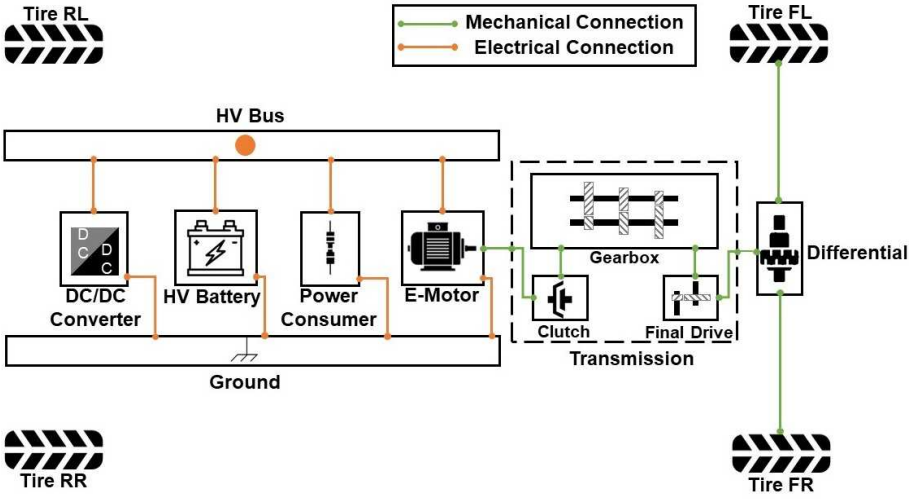


Figure 4.1: Standard vehicle architecture.

In this model, battery management and electric motor control systems are involved to ensure that the vehicle works under safe conditions, i.e., the power limitation of the battery pack and a current limit for the e-motor to match the battery limitation. The most used functionalities of velocity-dependent regenerative braking, electrical-mechanic parallel braking and battery current limitation are also represented. By modifying the cockpit output signals, the electric drive control calculates the e-motor load signal in both traction and recuperation conditions. If the

[3] https://www.avl.com/en/simulation-solutions/software-offering/simulation-tools-a-z/avl-cruise-m

braking effect of the electric motor torque is below the driver's request in terms of equivalent brake pressure, then the service brakes are supplied with pressure.

4.3.3 Energy Consumption Calculation

The model setup has been introduced in the previous sections, while here we present the method used to calculate energy consumption as the benchmark of our work based on the introduction in our previous work in [244]. The energy consumption of the studied EVs is all generated from its battery pack. In the model of the battery pack, battery cell current and voltage are applied in iterative calculations to simulate the battery cell dynamics. Starting from each iteration, the battery cell current I_{bc} needs to be calculated firstly by the following formula:

$$I_{bc} = \frac{P_{bp}}{n_{bc} \cdot V_{bc}} \tag{4.8}$$

where P_{bp} is the power of the battery pack, n_{bc} is the number of battery cells and V_{bc} is the terminal voltage of the battery cell. Here, a standard Resistor-Capacitor equivalent battery model is employed to expose the current-voltage dynamics of a lithium-ion battery cell. The battery's voltage dynamics must obey:

$$\begin{cases} V_{bc} = V_{oc}(SoC) - V_{p1} - R_0(SoC)I_{bc} \\ C_{p1}(SoC) \cdot \dfrac{\mathrm{d}V_{p1}}{\mathrm{d}t} = I_{bc} - \dfrac{V_{p1}}{R_{p1}(SoC)} \end{cases} \tag{4.9}$$

where V_{oc} is open circuit voltage, which is a function of the battery cell's State of Charge SoC; V_{p1} represents the transient voltage; R_0 and R_{p1} indicate the effective series resistance and the transient resistance; and each of them is a function of the battery cell's SoC; C_{p1} indicates the transient capacity, which is a function of the SoC as well. The SoC of the battery cell is calculated by:

$$SoC = SoC_0 - \int_0^t \frac{I_{bc}}{Q_{bc}} \, \mathrm{d}t \tag{4.10}$$

where SoC_0 is the initial SoC of battery cells and Q_{bc} is quantity of electric charge

of battery cells. The battery cell data and calibrated model parameters are soured from AVL Cruise M.

This work simplifies the battery package computational complexity by not considering the unbalance between cells in the battery pack, as their overall range will not be affected. So far, the energy consumption of each cell J used in the studied EVs can be calculated as:

$$J = \int_0^t (V_{oc}(SoC) \cdot I_{bc})\,\mathrm{d}t \tag{4.11}$$

4.4 Multimodal Data & Dataset

This section presents two datasets used in this study, each offering unique perspectives on vehicle energy consumption. The first dataset, **Vehicle Energy Dataset (VED)**, provides large-scale, sensor data collected directly from vehicles in the real world, providing some physical and operational attributes such as speed, fuel usage, and auxiliary power. On the other hand, the second dataset, the **AVL-Generated Dataset**, is formed by feeding selected VED trips into the vehicle dynamics model introduced in Section 4.3, combining observed inputs, e.g., GPS, altitude, and speed, with simulated outputs, e.g., temperature, actual velocity, and trip distance.

4.4.1 Dataset 1: Vehicle Energy Dataset

VED is a real-world open dataset featuring a single-source data modality, composed of onboard time-series telemetry data including GPS trajectories, fuel usage, energy consumption, speed, and auxiliary power usage collected from 383 personal cars in total (264 gasoline vehicles, 92 hybrid EVs, and 27 EVs or plug-in hybrid EVs) through onboard data loggers from November 2017 to November 2018 in Michigan, USA [245]. A visualisation of a partial trip is presented in Figure 4.2, where blue icons represent the starting point and ending point, and the colour of each segment denotes the driving speed, with green indicating faster movement and red indicating

slower movement.

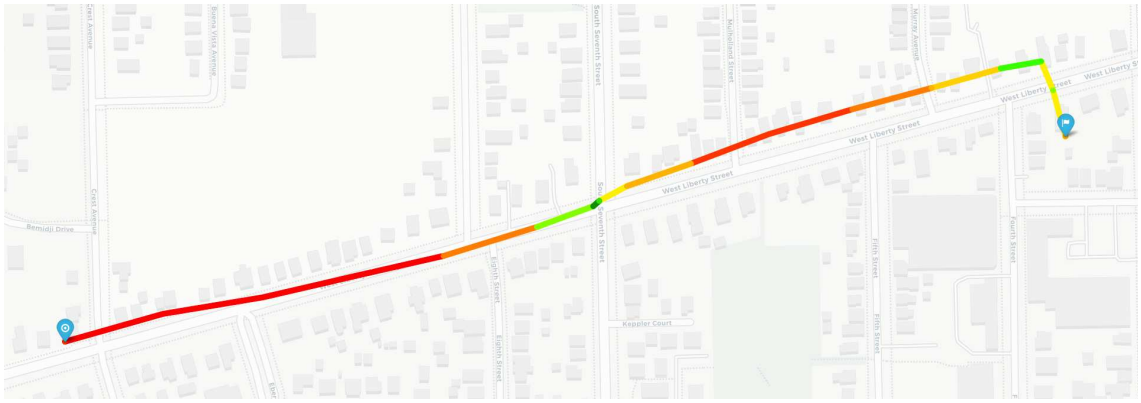
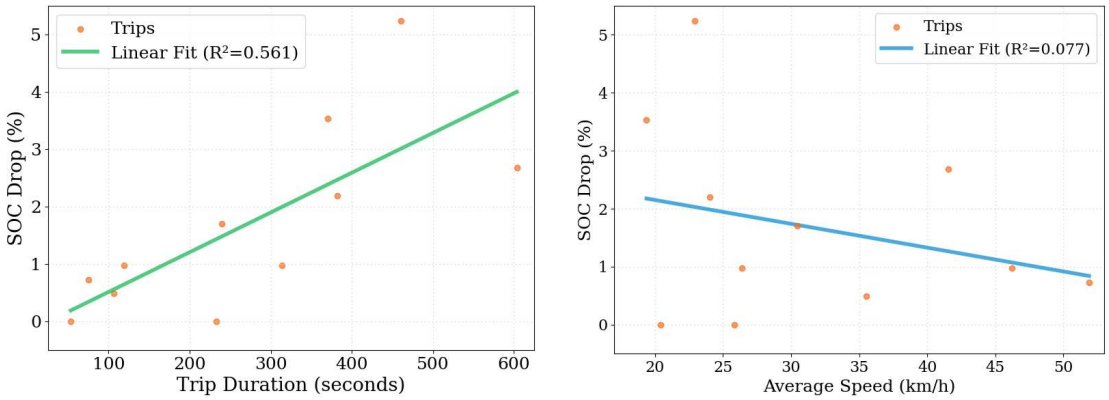


Figure 4.2: Trajectory of a partial EV trip in the VED dataset.



(a) SOC drop v.s. trip duration.

(b) SOC drop v.s. average speed.

Figure 4.3: Analysis of SOC consumption patterns across trips.

To better understand the factors influencing changes in SOC, we analysed one week of data in VED and examined two specific relationships: between the trip duration and SOC drop, and between the average speed of the whole trip and SOC drop. As shown in Figure 4.3, trip duration and average speed exhibit differing correlations with SOC drop. Trip duration shows a clear positive relationship, i.e., longer journeys usually lead to greater reductions in SOC. The concentration of data points suggests duration is a key factor. In contrast, the average speed of a whole trip shows only a weak correlation, with a shallow slope of the fitted line and low R^2 value, indicating a limited contribution to the SOC change.

4.4.2 Dataset 2: AVL-Generated Dataset

To construct a more diverse multimodal dataset, we selected 10 trips from VED with a duration longer than 1,800 seconds, and we then extracted the GPS coordinates, altitude, and speed data to serve as inputs to our vehicle model described above, with speed specifically used as the desired velocity. The model output includes multiple trip-related attributes such as temperature, actual speed, and trip distance, which we compiled into a comprehensive dataset. In total, the dataset consists of 10 tables for 10 vehicles, each containing 1,800 rows and 12 columns representing various features. Unlike the original VED dataset, this generated dataset combines both real-world sensed data, e.g., GPS, altitude, and speed, and simulated outputs generated by a vehicle dynamics model, resulting in a multimodal data including physical, behavioural, and model-generated attributes.

We analysed the data across all 10 vehicles and observed a correlation between average speed and total energy consumption over each 60-second interval. For instance, the relationship between average speed and energy consumption for Vehicle 1 is shown in Figure 4.4. It is evident that there is a slight delay in the change of the average speed relative to the energy usage. Following our analysis, we have determined that this delay is approximately 23 seconds.

The energy consumption values in 60-second intervals have a wide range from a minimum of -28.15 Wh to a maximum of 283.38 Wh. The data analysis results revealed the distinct statistical features of individual vehicles, which suggests that they have been exposed to different road conditions and other circumstances in the 1,800 seconds of observation. Additionally, Figure 4.5 indicates that there are many outliers in the data of Vehicle 1 and Vehicle 5, all of which were verified as valid data points rather than anomalies.

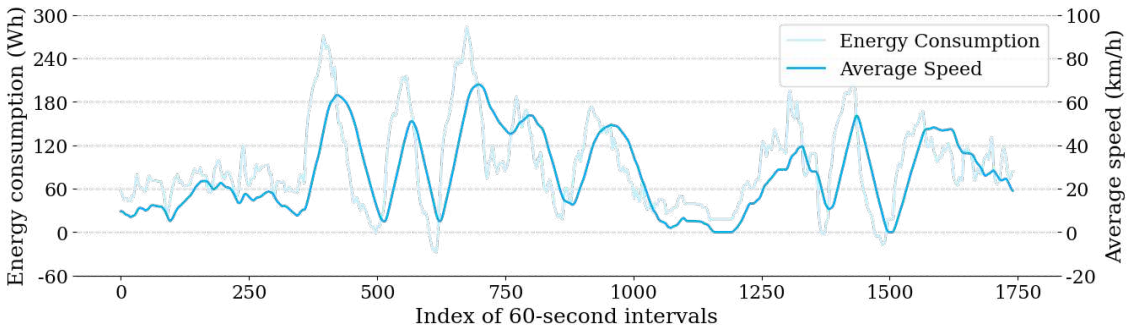


Figure 4.4: Effect of average speed on energy consumption in Vehicle 1.

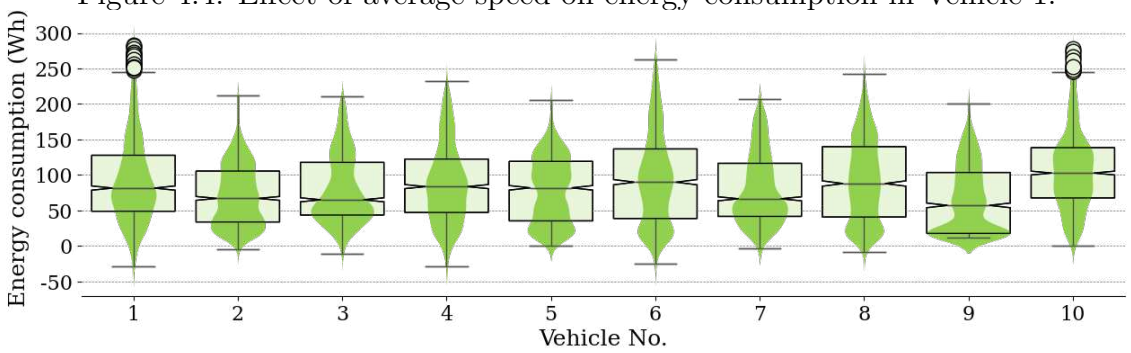


Figure 4.5: Box and violin plot of energy consumption for each vehicle.

4.5 Methodology

In this section, we briefly introduce centralised and decentralised FL structures, specify the ML models used as our local model candidates and provide a comprehensive overview of some FL optimisation algorithms which have been widely employed in a variety of domains.

4.5.1 Federated Learning Structures

In general, FL methods can be implemented in both centralised and decentralised structures, which are tailored to various real-world conditions. Specifically, a centralised FL implementation is illustrated in Figure 4.6, where each local model is trained on private data collected from a BEV and transmits the requisite information to the centralised server. After calculating and updating the global model, the unique information is sent back to each BEV to update its local model. In this scenario, a powerful centralised server is indispensable, and the requirement for BEV computing units is relatively low.

In contrast, for a decentralised FL structure, as demonstrated in Figure 4.7,

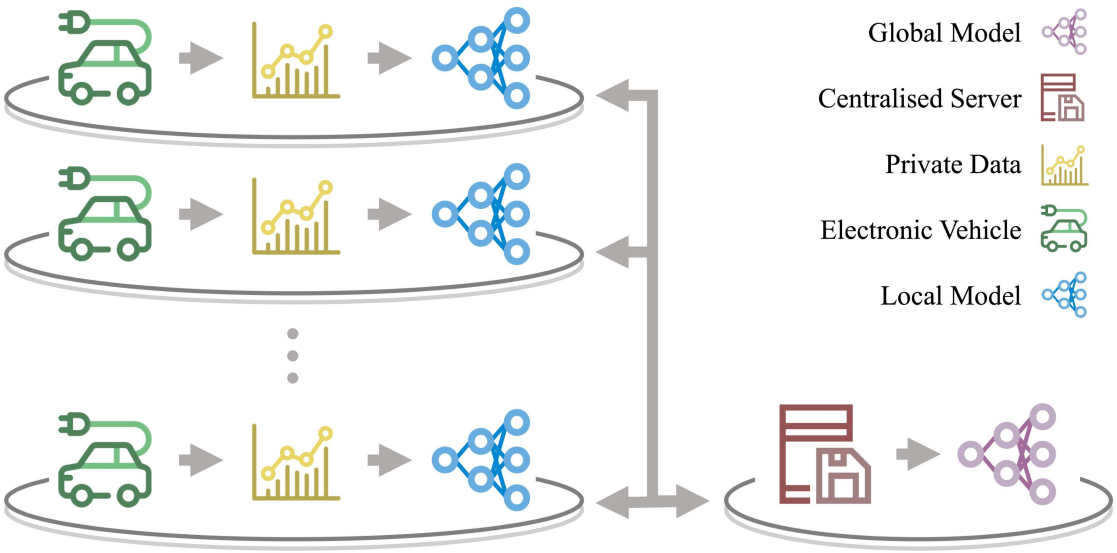


Figure 4.6: Centralised FL architecture.

BEVs share essential information and perform local model calculation and updating independently instead of relying on a server. In this case, the requirement for BEV computing units is greater, but the need for a centralised server is eliminated.

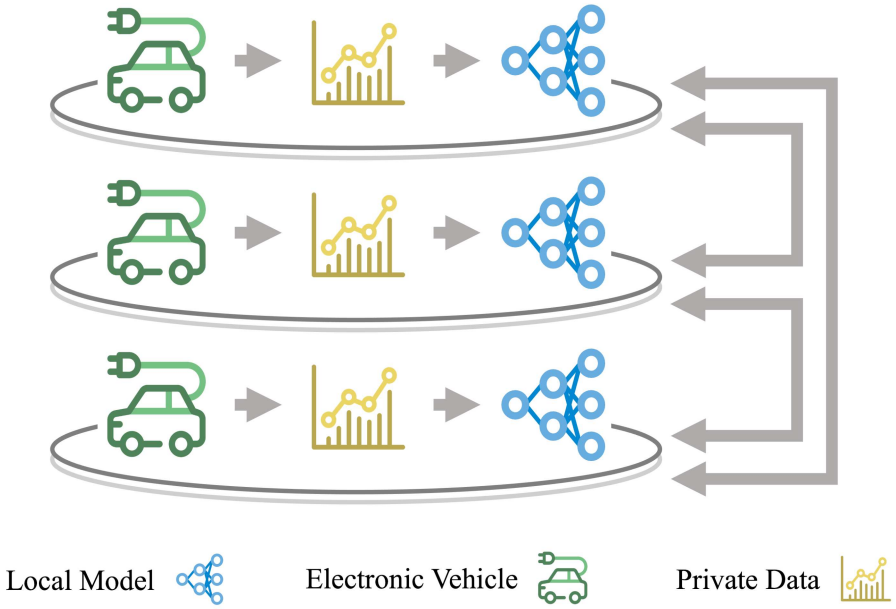


Figure 4.7: Decentralised FL architecture.

4.5.2 Local Model Selection

Several basic ML algorithms are adopted and compared to obtain the best candidate as our local model. As discussed in Section 4.2, we selected traditional ML algo-

rithms, e.g., RF and XGB, and DL algorithms, e.g., GRU and LSTM. The model implementation and performance are provided in the following sections.

4.5.3 Federated Learning Algorithms

We apply different FL optimisation algorithms in our prediction model section and compare their results to find the best-performing algorithm as our basic design together with the local candidate model. As introduced in Section 4.2, FedSGD, FedAvg, FedProx, FedRep and FedPer are widely used in similar systems, so these federated methods are adopted and their definitions are introduced in this part.

Algorithm 2 FedSGD for BEVs

Notations:

The V BEVs are indexed by v; the initial global gradient is g_0, the global gradient in the t^{th} round is g_t and the local gradient for vehicle v in the t^{th} round is g_t^v, the number of data points on vehicle v is n_v, the total number of data points is n, the loss function is **Loss**$(\cdot)$, and the local learning rate is η.

Server executes:

initialise g_0
for each round $t = 1, 2, \dots$ **do**
  for each BEV $v \in V$ **in parallel do**
    $g_{t+1}^v \leftarrow \text{SgdOpt}(v, g_t)$
  end for
  $g_{t+1} \leftarrow g_t - \eta \sum_{v=1}^{V} \frac{n_v}{n} \cdot g_{t+1}^v$
end for

SgdOpt(v, g)**:**

// run on vehicle v
$g \leftarrow \nabla \text{Loss}(g)$
return g

FedSGD: the direct transposition of stochastic gradient descent to the federated setting. The gradients are averaged by the server proportionally to the number of training samples on each node and used to make a gradient descent step. The pseudocode for the FedSGD algorithm is shown in Algorithm 2.

FedAvg: a generalisation of FedSGD, which enables local nodes to perform more than one batch update on local data and exchanges the updated weights rather than the gradients. The pseudocode for the FedAvg algorithm is shown in Algorithm 3.

FedProx: a modified algorithm based on FedAvg, characterised by two modifications. Firstly, partial models that have not been fully trained are allowed, and

Algorithm 3 FedAvg and FedProx for BEVs

Notations:

The V BEVs are indexed by v; the initial global weight is w_0, the global weight in the t^{th} round is w_t and the local weight for vehicle v in the t^{th} round is w_t^v, the number of data points on vehicle v is n_v, the total number of data points is n, the loss function is $\mathbf{Loss}(\cdot)$, the local data of vehicle v is D_v, the local minibatch size is B, the number of local epochs is E, and the local learning rate is η.

Server executes:

initialise w_0
for each round $t = 1, 2, \dots$ **do**
  $V_t \leftarrow$ (Full or partial set of V BEVs)
  for each BEV $v \in V_t$ **in parallel do**
    $w_{t+1}^v \leftarrow \text{AvgOpt}(v, w_t)$ **or** $\text{ProxOpt}(v, w_t, w_{t-1})$
  end for
  $w_{t+1} \leftarrow \sum_{v=1}^{V} \frac{n_v}{n} \cdot w_{t+1}^v$
end for

AvgOpt(v, w)**:**

// run on vehicle v
$\mathbb{B} \leftarrow$ (split D_v into batches of size B)
for each local epoch e from 1 to E **do**
  for batch $b \in \mathbb{B}$ **do**
    $w \leftarrow w - \eta \nabla \text{Loss}(w; b)$
  end for
end for
return w

ProxOpt(v, w^t, w^{t-1})**:**

// run on vehicle v; μ *is the proximal term coefficient*
$\mathbb{B} \leftarrow$ (split D_v into batches of size B)
for each local epoch e from 1 to E **do**
  for batch $b \in \mathbb{B}$ **do**
    $w^t \leftarrow w^t - \eta \nabla (\text{Loss}(w; b) + \frac{\mu}{2} \|w^t - w^{t-1}\|^2)$
  end for
end for
return w_t

Algorithm 4 FedPer and FedRep for BEVs

Notations:

The V BEVs are indexed by v; the initial standardised layer weight matrix for vehicle v is $W_S^{v,0}$, the standardised layer weight matrix for vehicle v in the t^{th} round is $W_S^{v,t}$, the initial personalised layer weight matrix for vehicle v is $W_P^{v,0}$, the personalised layer weight matrix for vehicle v in the t^{th} round is $W_P^{v,t}$, the number of data points on vehicle v is n_v, the total number of data points is n, the loss function is **Loss**$(\cdot)$, and the local learning rate is η.

Server executes:

*// for **FedPer:** W_S are base layers, W_P are head layers;*
*// for **FedRep:** W_S are head layers, W_P are base layers.*
initialise $W_S^0 := \{W_S^{1,0}, W_S^{2,0}, \dots\}$
initialise $W_P^{1,0}, W_P^{2,0}, \dots$
for each round $t = 1, 2, \dots$ **do**
  for each BEV $v \in V$ **in parallel do**
    $(W_S^{v,t+1}, W_P^{v,t+1}) \leftarrow \text{Opt}(v,\, W_S^{v,t},\, W_P^{v,t})$
  end for
  $W_S^{t+1} \leftarrow \sum_{v=1}^{V} \frac{n_v}{n} \cdot W_S^{v,t+1}$
end for

Opt$(v,\, W_S^v\, ,\, W_P^v)$**:**

// run on vehicle v
$(W_S^v,\, W_P^v) \leftarrow (W_S^v,\, W_P^v) - \eta \nabla \text{Loss}((W_S^v,\, W_P^v))$
return $(W_S^v,\, W_P^v)$

all partial models participating in the training are integrated regardless of their accuracy. Secondly, the objective function of the local model is composed of the loss function plus the proximal term. The pseudocode for the FedProx algorithm is shown in Algorithm 3.

FedPer: a federated learning optimisation method based on SGD which requires activating all clients and aggregating only the base layer parameters into individual personalised models. Its pseudocode is shown in Algorithm 4.

FedRep: a federated learning optimisation method based on SGD which requires activating all clients and focuses on learning a shared representation of the data, rather than learning individual models for each device. Its pseudocode is shown in Algorithm 4.

4.6 Experiments & Results

This section outlines data pre-processing and feature selection for our experiments, and several experimental cases, which encompass various experimental settings along with their corresponding results. By default, we assumed an input of the data from

all vehicles would result in the best performance.

4.6.1 Pre-processing

As introduced above, we found a strong connection between energy consumption and the average speed in a certain time step (TS), which makes it highly possible to predict energy consumption based on other features. Thus, we divide the data in each 60-second interval I^i as follows:

$$I^i := \{F^i_{t-59}, F^i_{t-58}, \ldots, F^i_{t-1}, F^i_t\} = F^i_{t-59:t} \tag{4.12}$$

where F^i_t is the set of trip features at time t for Vehicle i. With a full length of 1,800 seconds, the trip record data of one vehicle is divided into 1741 intervals.

Moreover, standardisation of the data is implemented to shift the distribution such that the mean is zero and the standard deviation is one. This preserves the essential information regarding the outliers, making the algorithm less susceptible to them, in comparison to Min-Max normalisation.

4.6.2 Feature Selection

As discussed in Section 4.2, most researchers working on similar research problems used speed, distance, acceleration and altitude as their features. Accordingly, we adopted these trip attributes and employed Principal Component Analysis and RF Regression to select features generated by mathematical transformations (e.g., square, logarithm, and exponent), and based on the results, we selected acceleration a, speed v, the square root of speed $\sqrt{v}$, the cube of speed v^3 and the square root of the distance moved in each second $\sqrt{\Delta d}$ as the features for model training.

4.6.3 Case Studies

Case 1: Local Model Comparison

Setups: In this section, we applied various ML models and compared their performance as the candidates of our local model deployed on each BEV. Here we implemented and evaluated RF, XGB, ANN, GRU and LSTM models.

Machine Learning Models: We initially applied the RF and XGB to implement the BEV energy consumption prediction on decentralised datasets, which were separately split into a training set and a test set with a ratio of $3:1$. In the setup of the RF model, 16 estimators were applied with a maximum tree depth of 9 to run 16 jobs in parallel, while for the XGB model, the regressor from the XGB library with a gradient-boosted tree booster was used to run 50 jobs in parallel.

Deep Learning Models: We adopt a similar structure for all the deep learning models in our work (i.e., ANN, GRU and LSTM) and replace the hidden layers with specific modules (i.e., dense layers, GRU layers and LSTM layers respectively). The deep learning model was set up based on three hidden layers containing 40, 32 and 16 neurons respectively with hyperbolic tangent as the activation function and the dropout layer was applied with a rate of 0.10 and 0.20 between the first and second and the second and third hidden layers separately. Coming from a dense layer, the model output of the model is a single value that by default captures the vehicle's overall energy consumption over the past 60 seconds. The loss function was chosen as the MAE which represents the average difference between the actual energy consumption of a BEV during a trip and the energy consumption predicted by the model. A lower MAE indicates that the model is better at predicting energy consumption and that the vehicle is likely to be more efficient, which can lead to reduced driving range anxiety and lower energy costs. The model was optimised by the Adam optimiser with the default learning rate, i.e., $1e^{-3}$. Each local dataset was split into batches with a size of 70 with 65 epochs.

Results: In Table 4.1, a comparative analysis of the credible and competitive results of these models is presented. The value in each row of this table represents

the average value of the MAE for the model trained on the corresponding vehicle's training set and tested on all vehicles' test sets.

Table 4.1: Local model performance (MAE) on each vehicle (Wh).

ID	RF	XGB	ANN	GRU	LSTM
V1	9.5375	9.0951	6.9207	5.3355	5.1759
V2	10.4122	9.7776	8.3997	6.8031	6.7997
V3	10.1580	9.4953	7.2892	6.3033	4.4117
V4	8.9370	8.6519	7.8394	6.6888	4.6765
V5	10.7788	10.6958	8.6680	5.6422	6.1466
V6	9.2424	8.8490	8.2999	4.9306	5.8707
V7	11.5519	10.5905	7.0247	5.3347	4.9729
V8	8.9563	9.0859	9.0698	7.5178	5.7530
V9	12.1133	11.5608	11.0439	10.7237	8.6670
V10	9.0282	8.8239	11.7233	7.1755	8.1254
V_{avg}	10.0716	9.6626	8.6279	6.6455	6.0599

Evaluations: As observed in Table 4.1, the LSTM model demonstrates the best performance in terms of MAE. It is noteworthy, however, that different datasets with varying attributes may result in different best-performing models. As such, in our study, we selected the LSTM model as the local model for our FL framework. Nonetheless, we acknowledge the need for adjusting the selection of the local model based on the prevailing conditions and characteristics of the dataset in future applications.

Case 2: Federated Learning Algorithm Comparison

Setups: As reviewed in Section 4.5, five algorithms, i.e., FedSGD, FedAvg, FedProx, FedPer and FedRep, were implemented and evaluated at this stage. It is evident that these algorithms share a similar structure and basic logic, with the exception of FedSGD, which calculates the gradient rather than weights, and FedPer and FedRep, which employ an additional parameter, namely the number of personalised layers. The performances of the different local model candidates, as shown in Table 4.1, led to the selection of LSTM due to its lowest MAE scores. Given the low complexity of the model, only one personalised LSTM unit was set, with the

other layers being standardised.

Results: The comparative analysis of different FL algorithms is presented in Table 4.2. The value in each row of this table represents the average value of the MAE for the model trained on the corresponding vehicle's training set and tested on all vehicles' test sets.

Table 4.2: FL method performance (MAE) on each vehicle (Wh).

ID	FedSGD	FedAvg	FedProx	FedPer	FedRep
V1	5.4300	4.0245	4.9574	3.8197	4.4066
V2	6.3252	4.8791	6.8512	4.8160	5.6075
V3	4.3714	3.8117	4.4467	3.6836	4.1196
V4	4.2921	3.4877	4.4171	3.3153	3.9459
V5	4.9364	4.5298	6.4715	4.6095	5.5250
V6	5.1704	3.7653	5.5837	3.8582	4.3317
V7	4.5819	4.1943	5.2512	4.3598	4.9148
V8	5.1873	4.3449	6.0334	4.1380	5.1330
V9	7.5800	7.4539	9.9346	7.5819	9.5547
V10	8.9892	7.5734	7.7679	7.6032	7.5043
V_{avg}	5.6864	4.8065	6.1715	4.7785	5.5043

Evaluations: As observed in Table 4.2, FedAvg and FedPer exhibit comparable performances on the dataset utilised in the study. As discussed in Section 4.2, the efficacy of personalised FL approaches is highly reliant on the personalised layers selected. Hence, the reason for FedPer and FedRep failing to surpass the performance of FedAvg could be attributed to the relatively low complexity of the local models employed in the study. Notwithstanding the comparable performances of FedAvg and FedPer, the time cost associated with both methods must be carefully considered. Taking into account the time required to complete 15 iterations (ITR) of FedAvg and FedPer, i.e., around 49 and 58 minutes respectively, we ultimately decided to adopt FedAvg as the global FL method for the subsequent experiments.

Case 3: Impact of Iteration

Setups: To investigate the impact of iteration numbers, the performances of the models with ITR of 15, 30, 45 and 60 were examined based on the LSTM-FedAvg

structure. These results were compared to the performance of the local models (i.e., with 0 ITR), which served as the baselines.

Results: Table 4.3 presents the impact of iteration on the model performance. The value in each row of this table represents the MAE value of the model trained and tested on the corresponding vehicle's training and test sets.

Table 4.3: ITR on model performance (Wh).

ID	LSTM	15 ITR	30 ITR	45 ITR	60 ITR
V1	2.0949	1.0118	0.6384	0.7127	0.6771
V2	1.4466	0.7181	0.5268	0.3533	0.3961
V3	1.5723	0.5057	0.5268	0.5463	0.3087
V4	1.6571	0.8222	0.8466	0.4838	0.6151
V5	1.1317	0.5640	0.5039	0.5823	0.2765
V6	2.2842	0.9905	0.6548	0.5882	0.8435
V7	1.4967	0.7576	0.5935	0.4758	0.4244
V8	1.7994	0.8203	0.8849	0.6794	0.4169
V9	1.3155	0.6690	0.5529	0.3905	0.2754
V10	2.2864	0.9877	0.7184	0.5738	0.7174
V_{avg}	1.7085	0.7847	0.6447	0.5386	0.4951

Evaluations: As demonstrated in Table 4.3, an increase in the number of ITR is accompanied by a decrease in the MAE for half of the objects (Vehicles 3, 5, 7, 8, and 9), indicating an improved model performance. However, for four objects (Vehicles 2, 4, 6, and 10), the best model performance occurred after 45 ITR. Vehicle 1 had the lowest MAE after 30 ITR. This trend can be attributed to the unique nature of the local data. Furthermore, while the model performance improved for some vehicles, the training process's time requirements increased. Specifically, executing 15, 30, 45, and 60 ITR required approximately 49, 98, 148, and 197 minutes, respectively, which implies that each iteration took about 49 minutes. After careful consideration of both model performance and time requirements, we chose to execute 15 ITR for this study.

Case 4: Impact of Data Split Ratio

Setups: Based on the LSTM-FedAvg structure, we conducted a comparative analysis of model performance across varying train-validation-test splitting ratios, i.e., 4:1:5, 5:1:4, 6:1:3, 7:1:2, and 8:1:1 to investigate the impact of data split ratio.

Results: The impact of data-splitting ratios on model performance is reported in Table 4.4. The value in each row of this table represents the MAE value of the model trained and tested on the corresponding vehicle's training and test sets.

Table 4.4: Impact of splitting ratio on model performance (Wh).

ID	4:1:5	5:1:4	6:1:3	7:1:2	8:1:1
V1	1.3546	1.3068	1.0626	0.7437	1.0118
V2	0.7424	0.6625	0.8641	0.6374	0.7181
V3	1.0652	0.9029	0.7644	0.7448	0.5057
V4	1.2096	1.0056	1.0278	0.7597	0.8222
V5	0.9935	0.7111	0.6575	0.7352	0.5640
V6	1.2573	1.4778	1.1662	1.1626	0.9905
V7	1.0656	0.9155	0.6053	0.5805	0.7576
V8	1.2109	1.0480	1.0135	0.9522	0.8203
V9	0.7191	1.1471	0.9642	0.7911	0.6690
V10	1.6850	1.2798	1.6003	1.0928	0.9877
V_{avg}	1.1303	1.0457	0.9726	0.8200	0.7847

Evaluations: Based on the results presented in the table, it can be observed that a data splitting ratio of 7:1:2 or 8:1:1 tends to yield the best performance on our dataset compared to the other ratios. In particular, a ratio of 8:1:1 demonstrates superior performance across all metrics. This indicates that an increase in the quantity of training data facilitates enhanced learning from the input under our experimental conditions. Consequently, in this work, we have selected a data splitting ratio of 8:1:1 due to its superior overall performance.

Case 5: Impact of Input Data Size

Setups: Concerning the input data sizes, we utilised data inputs with varying TS of 60, 90, 120, 150, and 180 timestamps to train the model based on LSTM-FedAvg, with the subsequent performance evaluation.

Results: The results of the experiments are presented in Table 4.5. The value in each row of this table represents the MAE value of the model trained and tested on the corresponding vehicle's training and test sets.

Table 4.5: Impact of input data size (TS) on model performance (Wh).

ID	**60 TS**	**90 TS**	120 **TS**	**150 TS**	**180 TS**
V1	1.0118	2.0234	2.4310	3.4794	7.2946
V2	0.7181	1.0923	1.2904	1.7201	3.9600
V3	0.5057	1.5067	1.6543	2.4812	3.4214
V4	0.8222	1.5299	2.0712	2.9217	3.2603
V5	0.5640	1.1884	1.4450	2.1414	2.4692
V6	0.9905	1.5533	1.7502	2.6796	3.7042
V7	0.7576	1.1970	1.6349	1.7855	2.5312
V8	0.8203	1.6027	1.7615	2.6084	3.5714
V9	0.6690	0.9991	0.9454	1.5623	1.6180
V10	0.9877	2.0208	2.2899	3.8614	8.7896
V_{avg}	0.7847	1.4714	1.7274	2.5241	4.0620

Evaluations: As can be seen in the table, the increase in input data size is typically associated with an increase in MAE, indicating poorer model performance. This result may be attributed to the decrease in the number of windows due to the increase in window size, resulting in a reduction of training data and consequent degradation of model performance.

Case 6: Decentralised Approaches

Setups: In order to incorporate various real-world scenarios, experiments were conducted based on the chosen FL approach in a decentralised setup. Using results from LSTM-based local models, we selected Vehicles 3, 4, and 6 as the top three performers (G), and Vehicles 2, 9, and 10 as the three weakest performers (W) to investigate the performance of decentralised FL approaches. From these 6 models, different test groups were formed based on the number of weaker performers as below:

- Three weak performers (**0G+3W**)
- One good performer with two weak performers (**1G+2W**)

- Two good performers with one weak performer (**2G+1W**)

- Three good performers (**3G+0W**)

Results: The findings are presented in Table 4.6.

Table 4.6: Performance (MAE) of decentralised FedAvg method (Wh).

ID	LSTM	0G+3W	1G+2W	2G+1W	3G+0W
V1 (W)	2.0949	0.2403	-	-	-
V2 (G)	1.4466	-	-	-	0.1956
V5 (G)	1.1317	-	0.1929	0.1892	0.2687
V6 (W)	2.2842	0.3488	0.2770	-	-
V9 (G)	1.3155	-	-	0.1413	0.2301
V10 (W)	2.2864	0.2815	0.3700	0.3106	-

Evaluations: The results indicate that after 15 ITR using the FedAvg algorithm, the performance of all local models in each case improved, demonstrating the efficacy of decentralised FL methods in enhancing BEV energy consumption modelling. However, an increase in the number of good performers does not always equate to an improvement in performance. Specifically, the MAE value of Vehicle 10 increased when interacting with Vehicles 5 and 6 (1G+2W) compared to being aggregated with two weak performers (Vehicles 1 and 6). Conversely, the interaction among good performers does not always result in the most suitable setup for an individual vehicle model. For example, the best model performance for Vehicle 2 occurs when interacting with Vehicles 9 and 10 (2G+1W), but it becomes worse when aggregated with Vehicles 5 and 9 (3G+0W).

4.7 Discussion & Limitations

This section presents a comprehensive discussion of the experiment results, focusing on the performance of the decentralised structure and FL applications in real-world settings where multimodal data is inherently present.

4.7.1 Decentralised Aggregation

As detailed in Section 4.6, the aggregation outcomes of models trained on locally stored, multimodal data were not significantly influenced by the number or percentage of well-performing participants. Instead, the similarity in multimodal data patterns among participants proved to be more influential. For instance, for two good performers, Vehicle 5 and Vehicle 9, their interactions with Vehicle 10 resulted in significantly better results than with Vehicle 2. This highlights that the similarity and dissimilarity in data modalities, such as differences in driving behaviour patterns, road conditions, and energy profiles, are crucial factors in federated aggregation. Therefore, exploring the similarity of user behaviours and driving patterns is necessary for achieving better aggregation results.

4.7.2 Real-World Application

FL methods are well-suited for edge-cloud computing frameworks where multimodal data remains distributed across heterogeneous devices. We propose the application of our work in a real-world edge-computing-based system, the framework of which is illustrated in Figure 4.8. The top layer comprises a cloud data centre, the middle layer consists of multiple edge infrastructures and the bottom layer is composed of base stations and BEVs. Different methods of data transmission and calculation have been considered and marked with different colours to accommodate various communication situations and hardware conditions. We now provide details of the four blocks from left to right in this framework:

- The first block in yellow illustrates the classic centralised method, wherein data provided by BEVs is transferred to the edge server through the base station and then passed to and calculated by the cloud server. This approach does not necessitate BEVs to calculate, but they can request the edge server to calculate the weights in local models.

- The subsequent block in red depicts the fully decentralised method, wherein BEVs communicate with each other directly. This necessitates that the data computing and transmission capabilities of BEVs be highly robust and reliable.
- The third block in blue represents cloud computing, wherein data provided by BEVs is sent to the cloud server directly through the base station, without the requirement of edge infrastructures.
- Lastly, the block in green indicates that data is only transmitted to the edge server, thus requiring the edge server to possess a high capability to calculate and deploy the data.

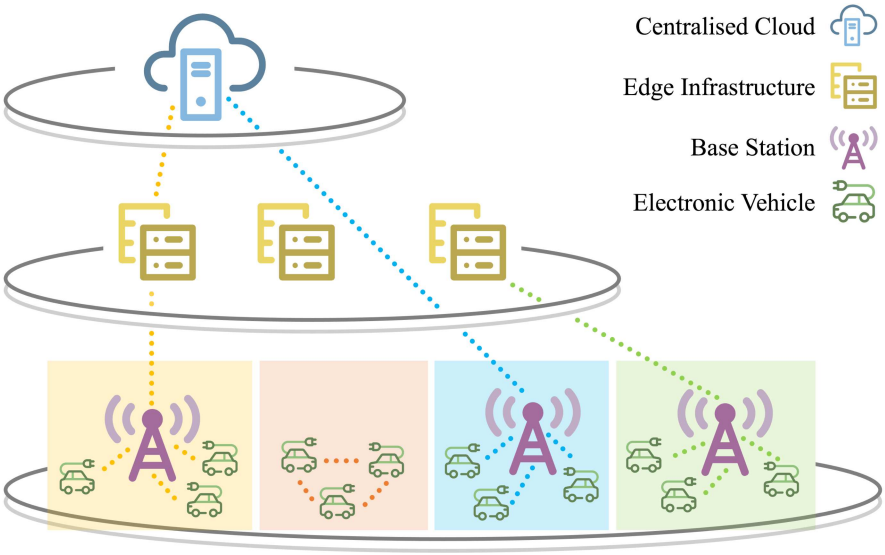


Figure 4.8: System framework based on edge computing.

4.8 Summary

This work aims to establish a privacy-aware energy consumption modelling framework for connected BEVs using multimodal data to address the challenge of energy consumption modelling for shared BEVs. To this end, we evaluated the performance of various FL algorithms on datasets comprising diverse input modalities such as GPS data, speed, altitude, and model-generated variables. Based on comparative experimental results and prior discussions, we identified LSTM and FedAvg

Table 4.7: Summary of EV energy consumption prediction methods.

Study	**Method**	**Prediction Target**	**Data Used**
[59]	MLR, SVM	Battery-electric bus energy consumption	vehicle mass, frontal area, road gradient, average speed, acceleration rates, etc
[53]	DT, RF, KNN	EV energy consumption	average speed, trip distance, nighttime lighting, road gradient, etc
[43]	XGB, LGBM	EV energy consumption	average speed, trip distance, heater usage, road gradient, etc
[144]	XGB	Electric taxi energy consumption	driving time, driving range, technical velocity, acceleration, temperature and traffic conditions
This work	LSTM-FedAvg	EV energy consumption	speed, distance, acceleration and altitude

as the most suitable local model and aggregation strategy, respectively. We further analysed the effects of key factors including the number of iterations, data splitting ratios, and input data sizes on model performance. A linear relationship was observed between iteration numbers and training time, and we recommend 15 iterations as an effective trade-off. For data splitting, the 8:1:1 ratio performed best on our multimodal dataset. Additionally, we found that increasing the temporal input length negatively impacted performance, with 60 timestamps yielding the most stable results. Overall, the use of multimodal inputs contributed to a significant improvement in prediction accuracy, with up to 67.84% reduction in MAE (from 1.5723 to 0.5057 on Vehicle 3). The average MAE across all vehicles was 0.7847, indicating an error of less than 1 Wh for one-minute intervals. To situate our method within the wider research landscape, a summary of the methods and data sources used in related work and this chapter is provided in Table 4.7.

In addition, we explored decentralised FL approaches for BEV energy consumption modelling by testing the selected FL framework on the three best and three worst performers. The results demonstrated the effectiveness of FL methods in this prediction task. Moreover, we provided a detailed description of how to apply FL methods with an edge-cloud computing framework with various setups. We believe that the explorations and relevant analysis we conducted are meaningful and beneficial for future researchers, business operators, and policymakers.

Chapter 5

Route Planning with Air Quality Constraints

Urban air pollution poses significant health risks for vulnerable SMS users, such as cyclists and pedestrians, who are directly exposed to traffic emissions. However, supporting pollution-aware travel behaviour remains challenging due to the high sparsity and inconsistency of urban air quality data. This chapter addresses the third key challenge identified in Chapter 1, i.e., how to enable effective route planning regarding air quality under extreme data deficiency. We present a data-driven framework that integrates multimodal air quality sources through interpolation, spatiotemporal prediction, and optimisation techniques to support safer and more sustainable travel choices in urban environments. This work is conducted in collaboration with Shaoshu Zhu, Jaime B. Fernandez, Eric Arazo Sánchez, Yingqi Gu, David J. O'Connor, Xiaojun Wang, Noel E. O'Connor, Alan F. Smeaton, and Mingming Liu, and published in [17], [246].

5.1 Introduction

Urban air pollution, particularly from NO_x and Particulate Matter (PM), poses a serious and escalating threat to public health [247], contributing to an estimated

7 million premature deaths per year worldwide[1] and approximately 3,300 deaths annually in Ireland alone[2]. These pollutants, primarily emitted from road transportation, industrial activities, and residential heating, are linked to respiratory and cardiovascular diseases, adverse birth outcomes, and elevated cancer risks.

Vulnerable road users, such as cyclists and pedestrians, are especially exposed to urban air pollution due to their close proximity to traffic and their longer durations in the open environment [7]. In densely populated areas, PM, especially $PM_{2.5}$, can penetrate deep into the lungs, leading to severe health consequences [149], [150], [248]. As urbanisation intensifies, there is a growing need for systems that help us to make safer travel choices by integrating air quality considerations into our urban route planning.

However, developing effective pollution-aware mobility systems relies on the availability of high-quality, high-resolution air quality data. Although initiatives such as the *Google Air Quality API*[3] provide large-scale forecasts, their data sources and methods are often not fully transparent or readily accessible and lack the granularity needed for personalised exposure management. This study employs multimodal datasets from mobile and fixed sensors across Dublin, Ireland, to construct an hourly pollution map with fine spatial resolution. However, despite the variety of data sources, the merging of multiple datasets has a missing rate of 89.64%, which leads to significant challenges for direct application in real-world systems. Consequently, addressing missing data is essential for building reliable pollution-aware routing solutions.

Previous studies have explored a range of imputation and prediction methods, including classic methods such as IDW, ML models such as RF and KNN [249], [250], [251], and DL models based on recurrent networks such as LSTM and GRU [252], [253], [254]. These approaches offer different strengths in reconstructing missing values and modelling the complex spatiotemporal dynamics of environmental data.

[1] `https://www.who.int/health-topics/air-pollution`
[2] `https://www.irishexaminer.com/news/arid-41018408.html`
[3] `https://developers.google.com/maps/documentation/air-quality/overview`

In this context, our work presents a complete framework aggregating missing data imputation, pollutant forecasting, and green route recommendation for vulnerable road users. The main contributions of this study are as follows:

- A comprehensive analysis of air quality data from multiple sources was conducted in Dublin, achieving high spatiotemporal resolution (0.5 km grid cells, hourly) while addressing significant data sparsity.
- A comparative evaluation of interpolation and prediction methods is conducted, selecting models that best balance accuracy and robustness under high missing rates.
- A pollution-aware route planning system is built and tested for vulnerable road users based on predicted pollutant distributions, enabling healthier mobility choices in urban environments.

The structure of this chapter is as follows. In Section 5.2, we review studies on air quality prediction, imputation for spatiotemporal data, and frameworks that integrate multimodal information for urban applications. The research problem and system design are introduced in Section 5.3. Section 5.4 details the multimodal data and datasets and how they contributed to this system, and Section 5.5 presents the methodology in terms of interpolation, prediction, and route planning. The dataset analysis and processing, prediction problem, experiment setup, and relevant results are presented in Section 5.6. Relevant discussions and analysis are included in Section 5.7, and finally, we conclude our work in Section 5.8 and discuss future plans and potential improvements.

5.2 Related Work

5.2.1 Data Imputation for Spatiotemporal Datasets

Accurate reconstruction of missing values is necessary for maintaining data completeness and supporting downstream tasks in spatiotemporal data analysis [255],

[256]. In the temporal domain, RNN and its variants, including simple RNN, LSTM, and GRU, have been widely used to model time series dependencies under incomplete observations [257], [258], [259]. For instance, an LSTM-based air pollution prediction framework was proposed in [259], where a genetic algorithm was used to optimise key hyperparameters such as window size and hidden units, leading to substantially improved prediction accuracy compared to traditional ML models. This illustrates the strong capacity of LSTM architectures to model noisy and irregular temporal patterns. Moreover, comparative studies such as [258] found that GRU can offer considerable performance while reducing model complexity, making it competitive for applications with limited data.

For spatial interpolation, classic statistical methods such as IDW and KNN remain competitive under conditions of high sparsity [21], [151], [152], [260]. Their reliance on local proximity, rather than extensive feature learning, allows them to operate effectively even when training data is fragmented or unevenly distributed. Additionally, learning-based spatial interpolation approaches have been explored to employ richer feature information. In particular, RF has been successfully applied to spatiotemporal interpolation tasks such as aerosol loading reconstruction [153], demonstrating its ability to capture nonlinear feature interactions while maintaining interpretability. Other works, such as [261], further confirmed the capability of tree-based ensembles in missing data imputation. In parallel, DL approaches, including CNN, GNN, and GCN, have also been applied to spatial interpolation problems [154], [155], [262]. These models can capture complex spatial patterns, such as regional clusters and dependencies between neighbouring areas, which might be missed by traditional methods. However, they generally require denser and more structured datasets to fully realise their advantages.

Building on these findings, this study employs multiple classic interpolation, ML, and DL methods to better understand and compare their performance on multimodal air quality datasets collected from various sources.

5.2.2 Spatiotemporal Prediction of Air Pollution

Short-term spatiotemporal prediction of air pollutants has developed from early statistical methods to more recent applications of ML and DL techniques [263]. Early approaches, such as historical averaging and ARIMA, were applied to air quality forecasting tasks [156], [264], [265]. Although these methods are more computationally efficient, they often rely on stationarity assumptions and struggle to capture the nonlinear and dynamic dependencies, especially in real-world scenarios, making them less suited for complex urban environments. Building on these early studies, classic ML models such as RF have been introduced for structured urban grids [266], [267]. RF models can capture nonlinear dependencies and are less sensitive to noise and redundant features, making them more robust to missing values and measurement noise, which are common challenges in environmental datasets.

More recently, DL methods have been explored to improve predictive performance in spatiotemporal tasks. TCN, which replaces recurrent layers with causal convolutions, has been successfully applied to sequence modelling tasks such as trajectory forecasting [157] and traffic flow prediction [268]. Compared with recurrent models, TCN offers advantages in training stability and can naturally capture longer temporal dependencies. Meanwhile, Conv-LSTM has remained popular for spatiotemporal prediction as it combines convolutional operations for extracting spatial features with LSTM cells for modelling temporal evolution [158], [269], enabling it to jointly learn spatial and temporal patterns and making it effective for grid-based pollutant forecasting, although its training often requires larger datasets and careful tuning to avoid overfitting.

These developments reflect a shift from simple statistical methods to more flexible learning-based models, motivating the comparative evaluation of different methods in this study.

5.2.3 Pollution-Aware Route Planning

Existing studies have reported significant differences in pollutant exposure across alternative commuting routes. For instance, [159] compared three cycling routes and observed large variations in ultrafine particle concentrations. Similarly, [160] proposed alternative paths for vulnerable SMS users based on spatial patterns of black carbon concentrations, aiming to reduce overall exposure. These findings suggest that adjusting route choice based on environmental conditions can meaningfully lower pollution exposure during travel.

More recent work has moved from static analyses to including real-time or predicted air quality information into route planning. For example, Empirical Bayesian Kriging has been used in [270] to interpolate pollutant concentrations and recommend routes with lower exposure. Building on this, a framework was designed in [161] to predict air quality distributions by combining CNN and RNN, enabling dynamic route adjustments based on predicted pollution levels. A similar approach was proposed in [271], where predicted near-road pollution levels were incorporated into a bicycle route planning tool that supports multi-criteria evaluation, balancing pollutant exposure with safety and accessibility. However, most of these approaches relied on limited types of external information and only used pollutant measurements alone. As a result, their ability to respond to broader urban changes remains restricted.

To address this gap, recent studies have explored the use of multimodal data sources to improve air quality modelling. Relevant research [162], [163], [272], [273] has shown that including demographic characteristics, trip purposes, meteorological variables, such as temperature, wind speed, and humidity, as well as traffic flow information, can substantially enhance predictive accuracy. These additional features provide greater support for decision-making in urban environments. At the network level, [274] further demonstrated how exposure-aware objectives can be integrated into multi-modal transport planning to balance health and efficiency considerations.

Beyond technical modelling, several studies have examined how exposure infor-

mation is perceived and responded to by users. [275] found that while some individuals reported increased awareness and behavioural adaptation when presented with personalised exposure data, others reacted with avoidance or denial. Complementing this, [276] used in-situ interviews and wearable sensors to reveal that perceived exposure often diverges from actual measurements. Environmental elements such as greenery, water, and aesthetics were found to enhance commuting satisfaction, suggesting that perceived environmental quality should also be considered in healthy route planning.

5.3 Problem Formulation & System Model

The core problem addressed in this study is recommending pollution-aware routes that minimise $PM_{2.5}$ exposure for vulnerable SMS users, and outlining the system architecture developed to address this and the attendant challenge of data sparsity. The problem is formulated as a weighted shortest path search on a graph, where edge weights represent pollutant-adjusted travel cost. The system we developed includes a mobile interface to interact with users, a backend server connected to our database, and an optimisation module combining multimodal pollution data into route generation.

5.3.1 Research Question

We recommend a green route for vulnerable SMS users, such as cyclists, which minimises their exposure to $PM_{2.5}$ while travelling from a specific origin to a destination within a defined geographical area. Specifically, we model the area of interest using a weighted directed graph $G = (V,\ E,\ W)$, where $V = \{1,\ \ldots,\ n\}$ is the vertex set, and $E = \{e_1,\ \ldots,\ e_m\}$ is the set of edges with corresponding positive weights in $W = \{w_1,\ \ldots,\ w_m\}$. The nodes in the graph represent intersections or points of interest, while the edges indicate the road segments that connect these nodes.

The weight w_i associated with the edge e_i represents the aggregated level of

pollutant exposure that a user may encounter while travelling along that edge, based on historical data measurements collected for that edge and the vulnerability of the cyclist to each pollutant being considered. $PM_{2.5}$ is the primary focus of this study due to its widely recognised health impacts [277], [278], particular relevance to active travellers [17], and consistent availability across all data sources.

To further illustrate this point, mathematically, let K denote the total number of pollutants being considered, and let p_{ij} be the mean pollutant level for the j'th pollutant measured at the i'th edge, where $i = 1, \ldots, n$ and $j = 1, \ldots, K$, and l_i be the length of the i'th edge, then w_i can be defined as follows:

$$w_i = l_i \sum_{j=1}^{K} \alpha_j p_{ij} \tag{5.1}$$

where $\alpha_1, \alpha_2, \ldots, \alpha_K$ are positive coefficients capturing the vulnerability of each pollutant to the users, subject to the constraint $\sum_{j=1}^{K} \alpha_j = 1$ for normalisation. It is worth noting that given the diversity of different pollutants when they are combined for impact assessment, a min-max scaling is employed for each type of pollutant, implying that $p_{ij} \in [0, 1]$ and $w_i \in [0, l_i]$ which can be seen as a parameter proportionally weighted based on the length of the i'th edge road segment l_i and the vulnerability of the users to the K types of pollutants. This further implies that if the users' vulnerability across different road segments is the same, the optimisation model should favour the segment with the shortest length.

Given this context, our optimisation problem is to find a path $p = \langle v_1, v_2, \ldots, v_h \rangle$ which consists of a sequence of vertices starting from the source vertex $v_1 \in V$ and terminating at the destination vertex $v_h \in V$ that minimises the sum of the weights of its constituent edges. Mathematically, we have:

$$\underset{p}{\operatorname{argmin}} \sum_{i=1}^{h-1} w(v_i, v_{i+1}) \tag{5.2}$$

where $w(u, v) \in W$ denotes the weight of the edge connecting vertices u and v.

5.3.2 System Model Design

This section introduces the system architecture developed to address the research questions outlined above. An overview is shown in Figure 5.1, which presents the components of our system: a mobile application that serves as the frontend for user interaction, a backend server for data exchange and processing, a database for data storage, and an optimisation module for generating routes based on road-related parameters.

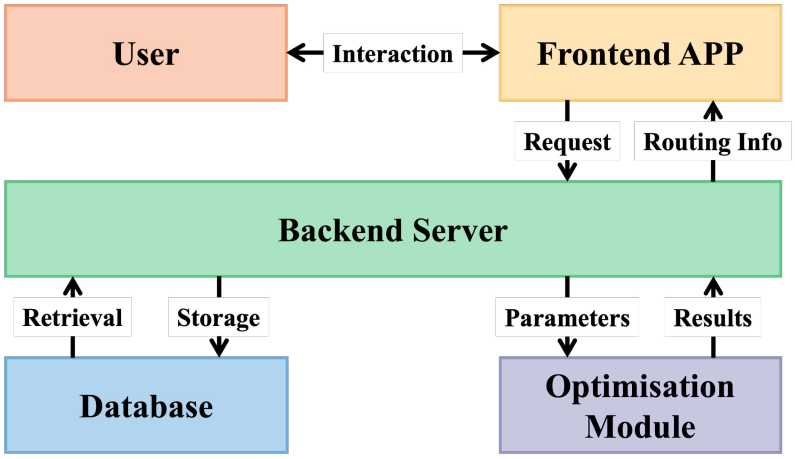


Figure 5.1: An overview of the system architecture

Frontend

The Android mobile application was developed using IntelliJ IDEA and tested on an Android Virtual Device with a *Google Play Intel Atom (x86)* processor. Map functionality was implemented using the *Google Maps API*[4], allowing users to specify origin and destination points and view routes optimised either for shortest distance or lowest pollutant exposure.

Backend

The backend server was built using Flask [279], a Python-based framework, to enable data-driven processing and real-time responses. Following the REST architecture, the server communicates with the frontend via JSON and computes route-related

[4] `https://developers.google.com/maps/documentation/directions`

metrics based on user inputs. On startup, the server loads geographic data for Dublin from *OpenStreetMap*[5] and integrates predicted results to construct a digital map enriched with pollution attributes. The optimisation module then evaluates pollution levels under different user-defined settings and generates a recommended route with reduced exposure.

Database & Optimisation Module

The interaction between the database and the backend is bidirectional. Data received from the backend is stored in the database, and the database is also used to retrieve and return data to the backend as per its requirements. Similarly, the optimisation module generates path recommendations based on the parameters received from the backend. It processes this information, calculates the optimised path based on the *Dublin urban road network models*[6] developed in [114], and returns results to the server. A more detailed explanation of the optimisation module will be provided in the subsequent section.

5.4 Multimodal Data & Dataset

This study integrates multiple datasets to capture the complex and dynamic nature of urban air pollution, providing a multimodal data foundation. Air quality data were provided by three different datasets: (1) the **Google Dataset** collected by *Google*[7] and *Dublin City Council (DCC)*[8]; (2) the **Environmental Protection Agency (EPA) Dataset** managed by the *Environmental Protection Agency of Ireland*[9]; and (3) the **Dynamic Parcel Distribution (DPD) Dataset** provided by *DPD Ireland*[10] and *DCC*. Additionally, this work employed extra datasets to represent unique modalities, including traffic volume data reflecting human mobility

[5] https://www.openstreetmap.org
[6] https://github.com/maxime-gueriau/ITSC2020_CAV_impact
[7] https://www.google.ie/
[8] https://www.dublincity.ie/
[9] https://www.epa.ie/
[10] https://www.dpd.ie/

patterns provided by *DCC*, spatial landcover information collected by EPA, and meteorological conditions gathered by *MET Eireann*[11]. Each data source provides information from different perspectives, such as spatial, temporal, behavioural, or environmental, which cannot be fully captured by any single modality alone. By combining these datasets, the modelling framework will have a better understanding of pollutant variations across both space and time, so in this section, we introduce these datasets used in this study, describing their characteristics, sources, and roles in supporting multimodal learning.

5.4.1 Dataset 1: Google Dataset

The Google dataset is collected between May 2021 and August 2022 as part of *Project Air View*[12]. Mobile sensors mounted on electric Street View vehicles equipped with the *Aclima* platform[13] measured pollutants at 1-second intervals while driving with normal traffic flow during weekdays from 9:00 to 17:00. The pollutants include Carbon Dioxide (CO_2), Carbon Monoxide (CO), Nitric Oxide (NO), Nitrogen Dioxide (NO_2), Ozone (O_3), and $PM_{2.5}$ (with size-resolved particle counts from 0.3-2.5 µm). The dataset offers high spatial resolution but limited temporal coverage, as sampling only occurred during daytime weekdays.

Two versions of the dataset are available[14], including *ADC-M*, containing raw 1-second measurements, and *ADC-R*, where data is aggregated for 50-meter road segments. In this study, we used *ADC-R*, which reports the median concentration per pollutant based on multiple drive passes. Each segment is labelled using its *OSM_ID*, enabling spatial alignment with external geospatial datasets.

[11] `https://www.met.ie/`
[12] `https://smartdublin.ie/project-air-view-research-opportunities/`
[13] `https://aclima.earth/`
[14] `https://data.smartdublin.ie/dataset/google-airview-data-dublin-city`

5.4.2 Dataset 2: Environmental Protection Agency Dataset

The EPA dataset is part of the *National Ambient Air Quality Monitoring Programme*[15] in Ireland and has been collected since 2017. It provides hourly averaged concentrations of key air pollutants, including Sulphur Dioxide (SO_2), NO_2, CO, O_3, and $PM_{2.5}$. Measurements are gathered from fixed monitoring stations across Ireland, each equipped with calibrated, high-precision instruments operating under standardised protocols.

The dataset provides high temporal resolution and long-term consistency as the data was recorded continuously, but its spatial coverage is limited due to its dependence on stationary infrastructure. Most monitoring stations are placed in urban centres or areas of regulatory interest, resulting in large gaps in fine-grained spatial information.

5.4.3 Dataset 3: Dynamic Parcel Distribution Dataset

The DPD dataset has been collected since September 2021, including data from two sensor types, i.e., mobile sensors installed on 102 DPD delivery vehicles and 22 fixed monitoring stations. All sensors are part of the Pollutrack network [280], [281], a Europe-wide initiative funded by DPD, and are specifically designed to measure real-time PM concentrations at breathing height. The pollutants monitored include Particulate Matter 1 (PM_1), $PM_{2.5}$, and Particulate Matter 10 (PM_{10}), enabling fine-grained differentiation between particle sizes.

Mobile sensors enable pollutant levels to be captured while vehicles travel through different urban routes during routine parcel deliveries, providing dense and dynamic spatial coverage. Meanwhile, the fixed stations ensure consistency and continuity at specific locations. This combination supports both spatial and temporal analysis, addressing the gap between fixed-site monitoring and emerging mobile sensing techniques.

[15] `https://www.epa.ie/publications/monitoring--assessment/air/national-ambient-air-quality-monitoring-programme-2017-2022.php`

5.4.4 Dataset 4: Traffic Volume Dataset

The traffic data is sourced from an open dataset collected hourly by *DCC*[16][17]. In addition to timestamps, the dataset includes two key metrics: *Sum Volume*, representing the total traffic volume recorded in the preceding hour, and *Avg Volume*, indicating the average traffic volume per 5-minute interval within the same period, which cannot be captured by environmental sensors alone but can provide a complementary behavioural context.

An analysis of this dataset reveals clear temporal patterns that differentiate weekdays from weekends. As shown in Figure 5.2, weekdays exhibit pronounced morning and evening peaks corresponding to commuting hours, while weekends show a more gradual increase in traffic volume around midday. This behavioural difference provides contextual cues closely associated with variations in air pollution levels caused by human mobility. Furthermore, weekday traffic tends to be more variable, especially during peak hours, whereas weekend patterns are relatively stable. On a weekly scale, traffic volume peaks on Fridays and drops significantly on Sundays, indicating a strong influence of work-related mobility.

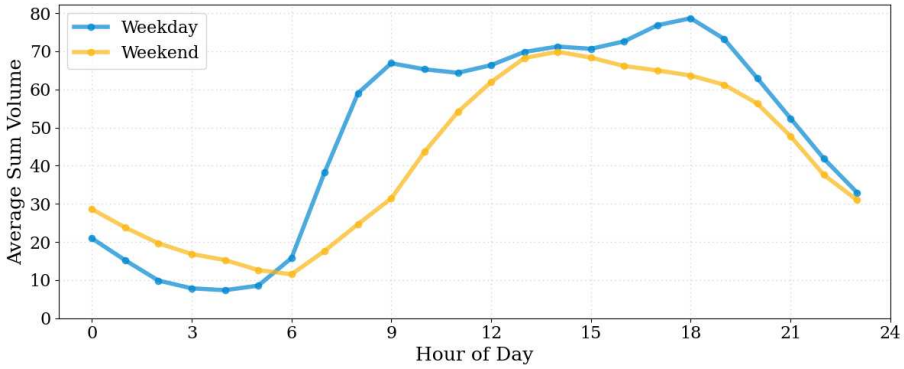


Figure 5.2: Hourly average traffic volume for weekdays and weekends.

5.4.5 Dataset 5: Landcover Dataset

The landcover data used in this study is sourced from the public *Corine Landcover* dataset[18], maintained by the EPA. As part of the 2018 update of the *Copernicus*

[16] https://data.smartdublin.ie/dataset/dcc-scats-detector-volume-jan-jun-2022
[17] https://data.smartdublin.ie/dataset/dcc-scats-detector-volume-jul-dec-2022
[18] https://gis.epa.ie/GetData/Download

Land Monitoring Service[19] from Ireland, it is based on the interpretation of satellite imagery and national in-situ vector data. This dataset provides useful information for classifying land according to usage types, with detailed classification guidelines available online[20]. GPS coordinates can be converted into a geospatial data frame and queried against the polygons defined in the *geometry* column of the dataset.

Based on this dataset, Figure 5.3 illustrates that Dublin's inner city is mainly covered by continuous urban fabric (shown in red), especially in the central and southern areas. This indicates that these zones are densely built-up with limited green or open spaces. Surrounding the core, smaller sections of discontinuous urban fabric (orange areas) represent lower-density residential neighbourhoods. Green urban areas (yellow zones) are relatively sparse and mostly concentrated in the western and northern parts, offering only limited greenery within the city centre. Industrial and commercial areas, marked in green, are only situated in the east, near the river and docklands. A small estuary zone (blue segment) also appears along the dockland, contributing to the overall land use diversity. In summary, the distribution of different land use types across Dublin's inner city will likely influence air quality dynamics and the effectiveness of urban sustainability measures.

5.4.6 Dataset 6: Meteorological Dataset

The meteorological dataset used in this study is the same as that described in Section 3.4, with the only difference being the time period covered. Since all other features remain consistent, no more details are provided here.

5.5 Methodology

This study introduces a three-stage optimisation module for the spatiotemporal analysis of urban air quality, covering missing data imputation, spatiotemporal $PM_{2.5}$

[19] `https://land.copernicus.eu/en/products/corine-land-cover`

[20] `https://land.copernicus.eu/content/corine-land-cover-nomenclature-guidelines/html/index.html`

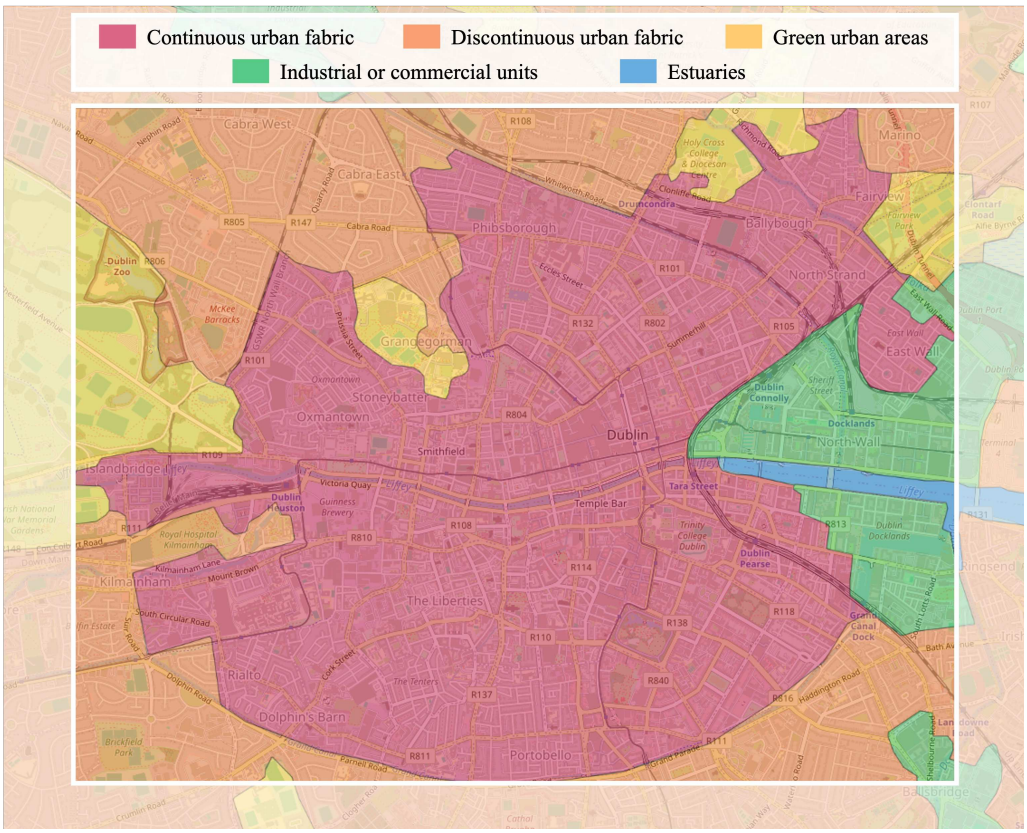


Figure 5.3: Overview of the landcover dataset.

prediction, and pollution-aware route planning. As shown in Figure 5.4, the optimisation module operates on a structured hourly grid constructed from multimodal data sourced from mobile sensing vehicles, fixed monitoring stations, and external contextual datasets, such as meteorological data, traffic volume, and landcover information.

In the first stage, a two-step interpolation process is used to address the high missing rate in our dataset, which could be defined as a grid system within the region of interest. Temporal models can first be applied to grid cells with relatively low missing rates, where they are trained to reconstruct $PM_{2.5}$ values based on local time series and then fine-tuned for use in target cells. Spatial models are subsequently applied to interpolate values across the grid at each time slice, using observations from nearby locations.

The second stage involves predicting the next-hour $PM_{2.5}$ values for all grid cells, based on recent spatiotemporal observations and a selected set of features. Classic methods, ML approaches, and DL models are examined for this short-term

forecasting task.

In the final stage, we demonstrate a downstream application using a case study on route planning, where predicted air quality data are used to identify paths with lower exposure for vulnerable SMS users.

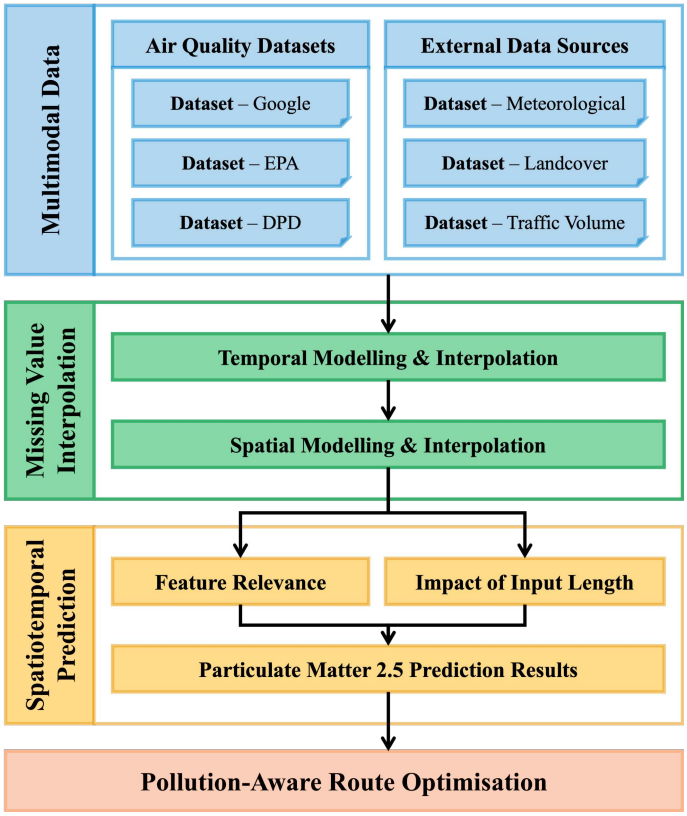


Figure 5.4: Framework of the pollution-aware route planning system.

5.5.1 Temporal & Spatial Interpolation Methods

To address the high missing rate in our air quality dataset, we employ a two-step interpolation strategy based on temporal and spatial modelling to reconstruct missing $PM_{2.5}$ values. Three recurrent neural networks were selected to learn temporal dependencies from sequential $PM_{2.5}$ data:

- **Simple RNN:** a simple architecture that models sequential data via recurrent connections, although limited by vanishing gradient issues.
- **GRU:** an improved variant of RNN with gating mechanisms that better capture dependencies without incurring significant complexity.

- **LSTM:** a widely used recurrent model with cell states and gates that enable learning of long-term dependencies and filtering of temporal noise.

After temporal patterns are reconstructed, spatial interpolation is performed independently at each hourly time slice using the following methods:

- **IDW:** a classic statistical method that estimates unknown values by computing distance-weighted averages of nearby observations.
- **KNN:** a non-parametric approach that fills missing values based on the average of the K nearest spatial neighbours.
- **RF:** an ensemble learning method that models nonlinear relationships using spatial features and external features.
- **CNN:** a DL model that treats the grid system as an image and applies convolutional filters to extract local spatial patterns.
- **GNN:** a message-passing architecture that updates node representations by aggregating information from neighbouring nodes within a defined graph.
- **GCN:** a graph-based model that performs spectral convolutions over a fixed adjacency structure, extending GNN with formal graph signal processing.

5.5.2 Spatiotemporal Prediction Methods

To enable short-term forecasting of $PM_{2.5}$ spatial distribution, we employ and compare a range of classic and learning-based models designed to capture both spatial and temporal dependencies in the fully interpolated dataset. All models below are trained to predict the pollutant concentrations for the next hour based on historical inputs aggregated at the grid-cell level.

Historical Average & ARIMA: a simple historical average method is used as a traditional baseline. It predicts based on the mean values observed at the same hour in the past. Another baseline model, ARIMA, is a widely used linear time-series model that captures trend and seasonality.

RF: a tree-based ensemble method known for its robustness and interpretability. It handles nonlinearity well and performs strongly on tabular data with structured feature sets. By adding attributes in past hours as an extra feature, it can capture not only spatial dependencies, but also temporal reliance, making it a strong candidate model for $PM_{2.5}$ prediction.

TCN: it replaces recurrent layers in traditional RNN models with causal convolutions, allowing the model to process entire sequences more efficiently and avoid issues like vanishing gradients. It is particularly suited to capturing short-term temporal patterns without recurrent memory structures.

Conv-LSTM: it integrates convolutional and LSTM layers to model spatial and temporal information jointly. The convolutional units extract spatial dependencies across the grid, while the LSTM units maintain temporal continuity across input sequences.

5.5.3 Pollution-Aware Route Planning

There are several well-known algorithms available to address (5.2), including Dijkstra's algorithm, Floyd-Warshall algorithm, and the Bellman-Ford algorithm [282], [283]. These algorithms are designed to identify the shortest or most efficient path between two points in a weighted directed graph. For ease of reference, we shall now present a description of Dijkstra's algorithm, which was applied in this case study, with reference to [284], [285] below. In Algorithm 5, the vertex set is denoted by Q, and the array $dist$ contains the current distances from the starting point s to other vertices, where $dist[v]$ represents the current distance from s to vertex u. The array $prev$ contains pointers to the previous-hop nodes on the shortest path from s to the given vertex, or equivalently, it is the next hop on the path from the given vertex to the source. The variable alt denotes the length of the path from the root node to the neighbour node v if it were to go through u. If this path is shorter than the current shortest path recorded for v, the current path would be replaced with this alt path.

Algorithm 5 Dijkstra's Algorithm

Input: $G = (V, E, W)$, a weighted directed graph, and $s, d \in V$, the starting point s and destination point d
Output: p, the shortest path for (5.2)

```
for all vertex v in graph G do
    dist[v] ← infinite
    prev[v] ← undefined
    Add vertex v to vertex set Q
end for
dist[s] ← 0
while Q is not empty do
    u ← argmin_i {dist[i] | ∀i ∈ Q}
    Remove u from Q
    for all neighbour v of u ∈ Q do
        alt ← dist[u] + w(u, v)
        if alt < dist[v] then
            dist[v] ← alt
            prev[v] ← u
        end if
    end for
end while
Set p as the shortest path to d using prev array
```

5.6 Experiments & Results

This section introduces a complete spatiotemporal processing for $PM_{2.5}$ interpolation, prediction, and pollution-aware routing using multimodal air quality data. As previously discussed, $PM_{2.5}$ is the focus of our study due to its significant impact on active travellers and its consistent presence in our datasets. We first integrate stationary and mobile measurements into an hourly grid, then apply a two-step interpolation strategy to fill missing values using temporal and spatial information. Based on the completed dataset, we train and evaluate models for short-term spatiotemporal prediction. Finally, we demonstrate the application of predicted pollution values through a route planning case study, showing that exposure can be reduced with only acceptable increases in travel distance.

5.6.1 Raw Data Overview & Spatiotemporal Pre-processing

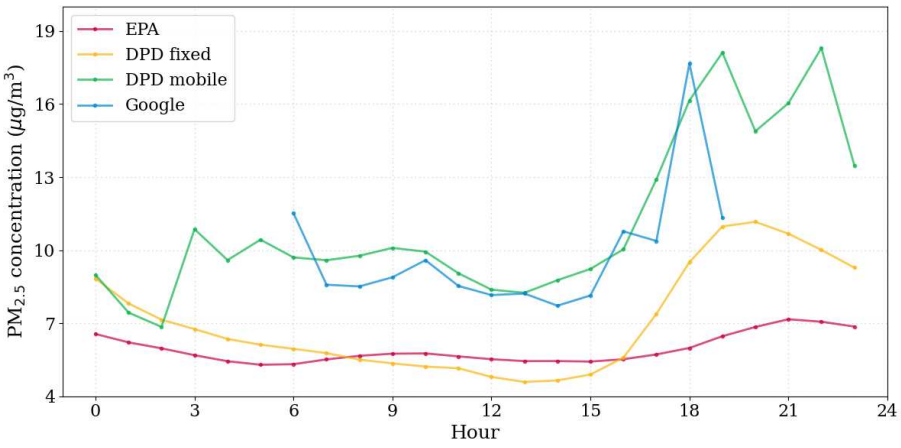
As outlined in Section 5.4, the air quality datasets used in this study include two different sources, i.e., fixed monitoring stations and mobile sensing vehicles. While data from stationary stations offer continuous temporal records, they are limited to a small number of fixed locations. In contrast, mobile vehicle data provide wider spatial coverage, but only along vehicle trajectories and without consistent temporal observations at any fixed point. To address these limitations and obtain a more balanced and detailed view of air pollution patterns, we divided the inner Dublin area into a grid system of 500 m $\times$ 500 m cells to enhance spatial resolution. This spatial division resulted in 70 grid cells arranged in 7 rows and 10 columns. Data within each grid cell was averaged on an hourly basis, resulting in 24 data points in each grid cell every day. A summary of all air quality datasets used in this study is provided in Table 5.1, where it is clear that $PM_{2.5}$ is the only air pollutant common to all sources.

Table 5.1: Summary of datasets including the number of records before and after filtering for Inner Dublin.

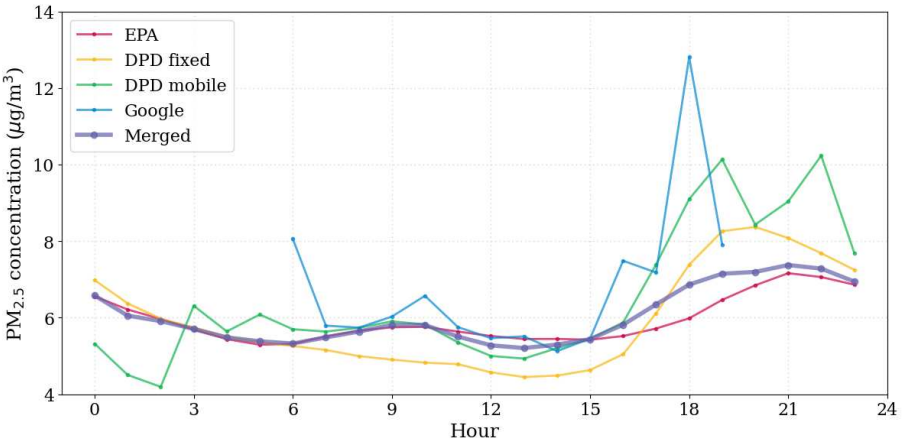
Dataset	Original Size	Inner Dublin	Time Range	Air Pollutants
Google	(5030143, 9)	(2176405, 9)	06/05/21 - 05/08/22	NO, NO_2, O_3, CO, CO_2, $\mathbf{PM_{2.5}}$
EPA	(1461684, 9)	(120994, 9)	04/01/22 - 31/12/22	NO_2, O_3, CO, SO_2, $\mathbf{PM_{2.5}}$, PM_{10}
DPD fixed	(6423046, 5)	(814168, 5)	01/10/21 - 31/07/22	$\mathbf{PM_{2.5}}$, PM_{10}
DPD mobile	(8029809, 5)	(618764, 5)	01/10/21 - 31/07/22	$\mathbf{PM_{2.5}}$, PM_{10}

While we acknowledge that each dataset may have its own sources of uncertainty, the EPA data, which is sourced from certified monitoring stations, was used as a reference standard. Specifically, we aligned the other datasets to the EPA data by calibrating them based on its mean and standard deviation. The calibrated datasets were then merged by computing hourly average values within each grid cell. The original $PM_{2.5}$ trends from these sources show some differences in pollutant levels and temporal patterns, as illustrated in Figure 5.5(a).

Specifically, the Google dataset shows notable spikes at certain times, such as around 6:00 and 18:00, while the DPD data collected from mobile vehicles tends to capture sharper peaks in the evening. In comparison, the EPA data exhibits a



(a) Hourly trends in original datasets.



(b) Hourly trends in calibrated and merged datasets.

Figure 5.5: The hourly trend of $PM_{2.5}$ across all grids in all datasets.

relatively stable pattern with lower overall concentrations, and the DPD stationary data shows smoother variations throughout the day. These differences result from the variations in sensor types, spatial coverage, and data collection methods, as introduced in Section 5.4. The merged results, shown in Figure 5.5(b), present a more consistent temporal pattern and were used as inputs for downstream tasks.

However, the dataset still exhibits very high sparsity and the causes of missing data vary across data sources and spatial locations. In mobile sensing datasets (Google and DPD mobile), gaps arise because data are only recorded when vehicles pass through a grid cell. As vehicle movements are neither spatially nor temporally uniform, this results in irregular coverage. In contrast, fixed monitoring stations (EPA and DPD fixed) only provide data within the grid cells where they are installed, and leave other areas unobserved. Additionally, even in cells with monitoring stations, data may be missing due to sensor faults, maintenance activities, or transmission errors. These factors lead to a high and uneven rate of missing values across the spatiotemporal grid, with over 89.64% of hourly grid-cell values missing, which highlights the importance of the following imputation experiments.

5.6.2 External Features & Feature Selection

As introduced in Section 5.4, traffic volume, land cover, and meteorological data were incorporated as external features by aligning them with the spatiotemporal aggregation method used for the original datasets. These inputs provide important external information from different perspectives. Specifically, meteorological data assists in capturing weather-driven transport and transformation; traffic flow contributes to understanding local emission strength; and land cover information reflects differences in fuel use and the influence of natural and built surfaces on the capacity for pollutant absorption. On the temporal side, raw timestamps were deconstructed to various temporal features, including hour of day, weekday, month, season, a rush-hour indicator, and a weekend flag. Each of these was represented as either a categorical or binary variable. At the end, all numerical features were stan-

dardised, and categorical features were one-hot encoded in our dataset, resulting in a structured input vector for each hourly grid cell.

Feature selection proceeded in two steps. For the missing-value imputation experiments, where the feature set was relatively small, we used Mutual Information (MI) and Pearson correlation analysis to rank and filter inputs to remove features with overlapping information or limited contribution to our target. For the downstream spatiotemporal forecasting task, which involved a much larger feature pool, we applied RF feature importance, Pearson correlation analysis, and single-feature ablation by iteratively dropping each feature and evaluating its impact on model performance to improve training efficiency. By combining RF importance scores, Pearson correlation metrics, and ablation results, we selected ten features, capturing over 80% of the variance and consistently improving predictive performance.

5.6.3 Missing Value Interpolation

This section describes a two-step approach for imputing missing $PM_{2.5}$ values. The first step uses time series modelling on grid cells with relatively low temporal missing rates, while the second step applies spatial modelling across grid cells at each time step. Feature selection was based on MI and Pearson correlation analysis, as described earlier, and included temperature, vapour pressure, wind speed and direction, spatial coordinates, month, hour of day, land cover, and traffic volume.

Temporal Modelling & Interpolation

We first focused on the 19 grid cells where the $PM_{2.5}$ data availability rate is lower than 95%, ensuring sufficient observations for model training. Five cells with the lowest missing rates among them were selected as source domains for pre-training our time series models. For evaluation, we randomly masked 20% of the observed values in each target cell's time series, and selected MAE, Mean Square Error (MSE), and Kullback-Leibler Divergence (KLD) as evaluation metrics.

We compared three models, including simple RNN, LSTM, and GRU, each

trained to predict masked hourly values based on the previous 24 hours of $PM_{2.5}$ observations. All models followed the same basic architecture: a single recurrent layer with 64 hidden units, followed by a dropout layer with a dropout rate of 0.2 applied to the final time step, and a fully connected linear layer mapping the output to a scalar $PM_{2.5}$ prediction. Models were implemented in PyTorch as sequence-to-scalar regressors, trained using the Adam optimiser with an initial learning rate of 1×10^{-3}, and evaluated using MAE, MSE, and KLD.

After pre-training, each model was fine-tuned on the remaining 14 target cells to adapt to local temporal patterns. During the first ten epochs, we froze all recurrent-layer parameters and trained only the final fully connected layer using the Adam optimiser, with a base learning rate of 5×10^{-4} and weight decay of 1×10^{-4}, minimising MSE. After epoch 10, the recurrent layers were unfrozen, and the optimiser was reinitialised with two parameter groups: the recurrent layer parameters trained at one-tenth the base learning rate, and the fully connected layer at the original rate. Both groups shared the same weight decay. A ReduceLROnPlateau scheduler monitored the validation MSE and halved the learning rate if the loss did not improve for the last 3 epochs. At the end of each epoch, we evaluated both training and validation sets using MSE, MAE, and KLD. The model state with the lowest validation MSE was retained, and this best-performing state was restored for final evaluation, with metrics reported as the average across all epochs.

Table 5.2: Model performance in pre-training, fine-tuning, and evaluation.

Model	**Pre-training (val set)**			**Fine-tuning (val set)**			**Evaluation (test set)**		
	MSE	**MAE**	**KLD**	**MSE**	**MAE**	**KLD**	**MSE**	**MAE**	**KLD**
RNN	25.3796	3.3958	0.9986	26.0893	3.6934	2.6211	19.4949	3.1990	6.6202
LSTM	25.3169	3.4474	0.4875	25.5268	3.4674	3.7840	16.6685	2.9676	8.6496
GRU	26.0046	3.4839	1.0417	25.4977	3.6028	2.5926	19.3059	3.1896	7.5810

The results of the three time series models on pre-training, fine-tuning, and final evaluation are summarised in Table 5.2. During pre-training on the source cells, all models achieved similar performance. LSTM showed the lowest MSE and KLD, while simple RNN slightly outperformed the others in terms of MAE. After fine-tuning, GRU achieved the best MSE and KLD, whereas LSTM remained competitive

with the lowest MAE. On the held-out test set, LSTM performed best with the lowest MSE (16.6685), MAE (2.9676), and a competitive KLD (8.6496).

These results suggest that while all three models can capture the temporal structure of $PM_{2.5}$ dynamics, LSTM showed more stable performance across training stages and generalised better in this setting. Based on this, it was applied to all 19 grid cells to fill in missing values in the time dimension, reducing the overall missing rate from 89.64% to 70.97%.

Spatial Modelling & Interpolation

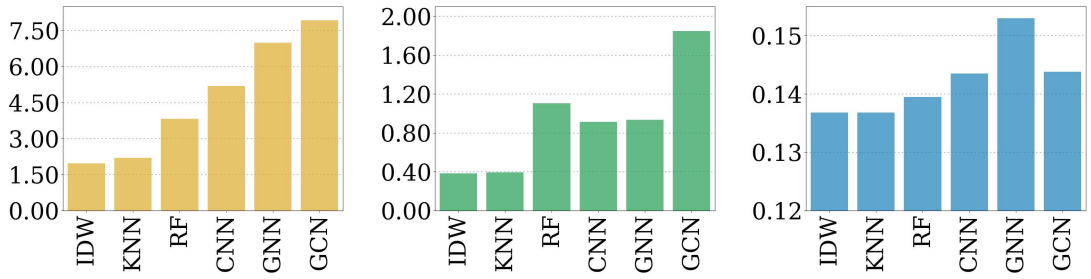
After temporal imputation, each hourly time step contained valid $PM_{2.5}$ observations in at least 19 grid cells, making it possible to move on to the spatial modelling and interpolation. We randomly masked 20% of the observed values per time slice to evaluate spatial interpolation performance as a validation set. Many interpolation and prediction methods were compared, i.e., IDW, KNN, RF, CNN, GNN, and GCN.

For classic and interpolation methods, IDW estimates missing values as weighted averages of observed values, where weights are based on the inverse of Euclidean distance, using a power parameter of 2. KNN follows a similar approach but averages the observations of the 5 nearest cells without applying distance-based weights. In the RF method, each hourly slice is treated as a separate regression task. At each time step, a unique RF model is trained using all available grid cells, with GPS coordinates and external features as inputs.

DL models, i.e., GCN, GNN, and CNN, shared the same general structure, consisting of a two-layer encoder applied to the spatial grid. The input was a vector of $PM_{2.5}$ values across all 70 grid cells, with missing entries filled with zeros, and the output was a reconstructed value for each grid cell. Training was conducted separately on each hourly slice using the masked reconstruction strategy described earlier, aiming to minimise the MSE between predictions and the true values of the masked grid cells. All models were trained for 100 epochs using the Adam optimiser

with a learning rate of 1×10^{-2}. Specifically, the GCN model uses symmetric graph convolution on a static undirected grid graph, where each cell is connected to its eight immediate neighbours. The GNN model builds on the same graph but adopts a more general message-passing approach, combining information from each node and its neighbours through separate linear transformations. The CNN model treats the grid as a 2D image and uses an encoder–decoder architecture with 3×3 convolution kernels to capture local spatial patterns. Its input includes the $PM_{2.5}$ values and a binary mask indicating observed locations, concatenated as a two-channel tensor.

Figure 5.6 presents the spatial interpolation performance of six methods in terms of MSE, MAE, and KLD. Across all methods, IDW produced the lowest MSE (1.9673) and MAE (0.3816), and similar with KNN in KLD (0.1368). KNN followed closely across all metrics. In contrast, learning-based models, including RF, CNN, GNN, and GCN, showed higher errors, especially in MSE. Although KLD differences were relatively less obvious, IDW and KNN still exhibited the best performance. These results suggest that simple spatial weighting methods such as IDW may be more effective in this case for interpolating sparse, irregularly distributed air quality data. Based on these observations, IDW was used to perform spatial imputation across all hourly slices, completing the missing-value imputation process.



(a) MSE (b) MAE (c) KLD

Figure 5.6: Spatial interpolation performance across different methods.

5.6.4 Particulate Matter 2.5 Spatiotemporal Prediction

Based on the fully interpolated dataset, we further evaluated the ability of different models to perform short-term spatiotemporal prediction. Specifically, models were

trained to predict the complete 7×10 spatial grid of $PM_{2.5}$ values at the next hour, using the previous N hours of historical data as input.

Feature Relevance

Input features were selected based on a joint analysis of RF feature importance, Pearson correlation, and single-feature ablation. Specifically, Pearson correlation revealed that variables such as *weather_temp* and *weather_dewpt* were negatively associated with $PM_{2.5}$ levels, while RF importance suggested *month*, *weather_msl*, and *weather_wdsp* as the top contributors. Ablation metrics (ΔMSE, ΔMAE, ΔKLD) further exhibited important features for this spatiotemporal prediction task, including *weather_vis*, *weather_wdsp*, and *hour*. An overview of feature relevance is in Table 5.3, where features are ranked by RF importance, and those selected for prediction are highlighted with a blue background. The ablation metrics are computed by:

$$\Delta M = M_{after\ removing\ feature} - M_{all\ features}, \quad M \in \{\text{MSE},\ \text{MAE},\ \text{KLD}\} \tag{5.3}$$

Impact of Input Length

To investigate the impact of input length on prediction performance, we tested a range of window sizes from 1 to 24 hours based on the RF model. As shown in Table 5.4, the 3-hour setting performed best overall, with the lowest MSE (4.6268) and KLD (0.1773), while the 12-hour window achieved the lowest MAE (1.4223). In comparison, 1-hour and 24-hour inputs resulted in higher errors across all metrics. These results suggest that a too short or too long window size may provide less relevant context for short-term air quality prediction.

Table 5.3: Summary of feature relevance under multiple metrics.

Feature	Correlation	Importance	ΔMSE	ΔMAE	ΔKLD
month	-0.2389	0.2473	-0.4245	-0.1211	-0.0072
weather_msl	0.2733	0.1401	+0.5254	+0.0900	-0.0036
weather_wdsp	-0.1805	0.0898	+0.7707	+0.1451	+0.0215
weather_wetb	-0.3752	0.0841	+0.5078	+0.0983	+0.0048
hour	0.1278	0.0752	+0.7286	+0.1347	+0.0072
weather_wddir	-0.1700	0.0731	+1.7626	+0.2457	+0.0173
weather_vis	-0.1613	0.0456	+1.2315	+0.1510	+0.0102
weather_w	-0.1661	0.0333	+0.1496	-0.0016	+0.0062
weather_temp	-0.3589	0.0279	+0.3030	+0.0469	+0.0025
weather_clht	0.1129	0.0272	+0.4302	+0.0747	-0.0012
wkd	0.0358	0.0189	+0.0125	-0.0280	+0.0004
land_use_522	-0.1085	0.0163	+0.3745	+0.0618	+0.0119
season_winter	0.2942	0.0158	-0.3537	-0.0776	+0.0066
weather_dewpt	-0.3650	0.0138	-0.0484	+0.0171	+0.0012
weather_rhum	0.0681	0.0133	+0.4718	+0.0726	+0.0207
season_summer	-0.2650	0.0116	+0.4059	+0.0801	+0.0021
weather_vappr	-0.3496	0.0107	-0.4373	-0.0906	-0.0092
season_spring	0.1626	0.0092	+0.1383	+0.0183	+0.0036
season_fall	-0.1719	0.0060	-0.3999	-0.0696	+0.0023
weather_ww	-0.1136	0.0054	+0.3559	+0.0809	+0.0117
weather_clamt	-0.1149	0.0044	+0.2170	+0.0455	+0.0013
weather_sun	-0.0221	0.0032	+0.6079	+0.1197	+0.0237
traffic_avg	0.0402	0.0028	+1.0355	+0.2200	+0.0249
traffic_sum	0.0416	0.0024	+0.1173	+0.0306	+0.0083
is_weekend	0.0582	0.0017	+1.4436	+0.2829	+0.0426
weather_rain	-0.0592	0.0015	+1.3300	+0.2520	+0.0500
is_rush_hour	-0.0427	0.0003	-0.1370	-0.0061	+0.0008
land_use_112	0.0322	0.0001	+0.5863	+0.1099	+0.0091
land_use_111	0.0166	0.0001	-0.0624	-0.0248	-0.0028
land_use_121	-0.0438	0.0001	-0.4076	-0.1096	-0.0051
land_use_141	0.0178	0.0000	+0.1832	+0.0250	+0.0012

CORR – Pearson Correlation, **IMP** – RF Feature Importance.
Feature Description:
weather_rain – Precipitation amount (mm), *weather_temp* – Air temperature (°C), *weather_wetb* – Wet bulb temperature (°C), *weather_dewpt* – Dew point temperature (°C), *weather_rhum* – Relative humidity (%), *weather_vappr* – Vapour pressure (hPa), *weather_msl* – Mean sea level pressure (hPa), *weather_wdsp* – Mean wind speed (knot), *weather_wddir* – Predominant wind direction (degree), *weather_ww* – Synop code for present weather, *weather_w* – Synop code for past weather, *weather_sun* – Sunshine duration (hours), *weather_vis* – Visibility (m), *weather_clht* – Cloud height (hundreds of ft, 999 if none), *weather_clamt* – Cloud amount, *land_use_111* – Continuous urban fabric, *land_use_112* – Discontinuous urban fabric, *land_use_121* – Industrial or commercial units, *land_use_141* – Green urban areas, *land_use_522* – Estuaries, *traffic_avg* – Average traffic volume, *traffic_sum* – Total traffic volume.

Table 5.4: Model performance with different window sizes.

Metric	1 H	3 H	6 H	12 H	24 H
MSE	5.0667	4.6268	4.9816	4.8379	4.9603
MAE	1.4405	1.4224	1.4502	1.4223	1.4342
KLD	0.1913	0.1773	0.1883	0.1783	0.1805

Prediction Results

Finally, we compared several classic and learning-based models using the best-performing feature set and the 3-hour input window introduced above to evaluate spatiotemporal prediction performance. As shown in Table 5.5, RF achieved the lowest MSE (4.6268) and MAE (1.4224), capturing short-term $PM_{2.5}$ variations with relatively high accuracy. Conv-LSTM achieved the lowest KLD (0.0941), showing a closer match in data distribution with slightly higher error values. In contrast, the historical average and ARIMA resulted in larger errors across all metrics, reflecting limited capacity to model spatiotemporal patterns in this case study. Among DL models, Conv-LSTM consistently outperformed TCN. Overall, the results suggest that tree-based methods, e.g., RF, are well suited to short-term $PM_{2.5}$ forecasting in a grid-based setting, particularly when combined with relevant features and an appropriate temporal context.

Table 5.5: Spatiotemporal prediction results with selected features and window size.

Metric	ARIMA	Historical Avg	TCN	Conv-LSTM	RF
MSE	10.4588	9.9460	11.8094	6.5811	4.6268
MAE	2.5020	2.4480	2.3019	1.6167	1.4224
KLD	15.8748	4.3425	1.8629	0.0941	0.1783

5.6.5 Pollution-Aware Route Optimisation

We developed a pollution-aware route planning module to help reduce $PM_{2.5}$ exposure for vulnerable SMS users by incorporating interpolated or predicted air quality data into the route selection process. Unlike conventional methods that consider only travel time or distance, this approach evaluates candidate routes based on trip length and pollution levels. For each route, $PM_{2.5}$ concentrations are summed across spatial segments to estimate total exposure, and a weighted score is applied to balance the trip distance and air quality, which can be customised by users according to their personal preferences or health conditions.

To evaluate the proposed method, we randomly selected 1,000 origin-destination

pairs within the study area, mainly focusing on walking and private micromobility modes such as bicycles and e-bikes. For each pair, three routes were computed using Dijkstra's algorithm. One example is illustrated in Figure 5.7, showing the shortest path (orange) and two greenest routes defined as those with the lowest estimated exposure, based on interpolated air quality for the current hour (green) and predicted values for the next hour (black), respectively. Three metrics were used to evaluate the greenest routes against the shortest ones, including the percentage increase in travel distance ($\mathbf{d} \uparrow$), the percentage reduction in $PM_{2.5}$ exposure ($\mathbf{p} \downarrow$), and the percentage reduction of the exposure per unit of travel distance ($\mathbf{p/d} \downarrow$).

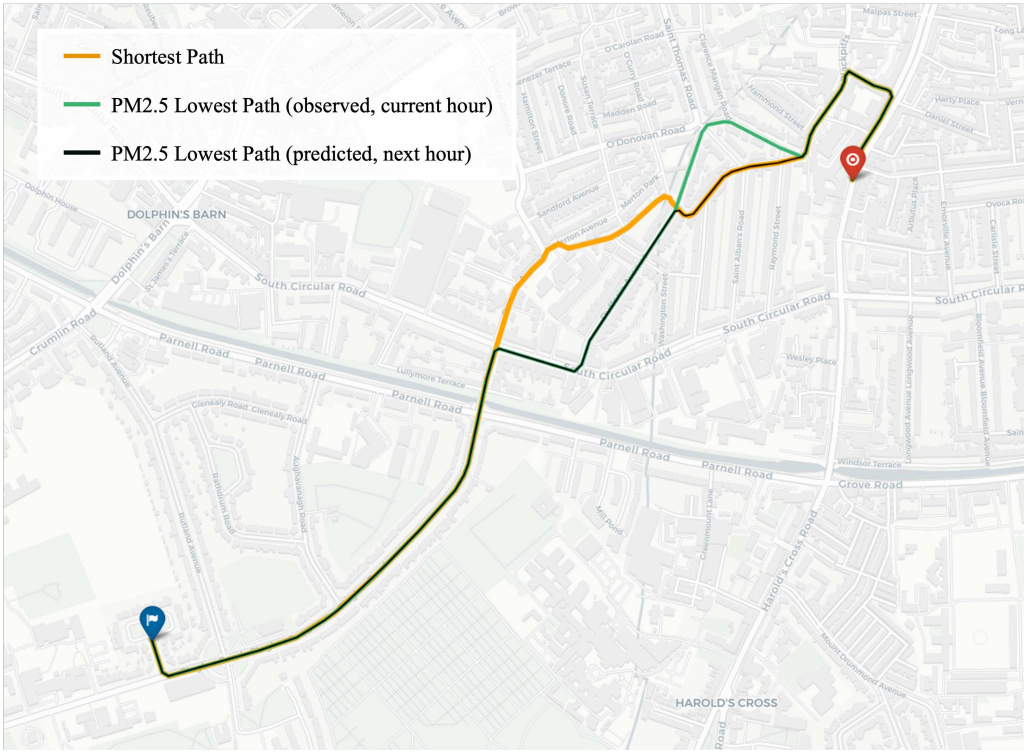


Figure 5.7: Shortest path and two optimised paths.

As shown in Table 5.6, routes based on interpolated pollution data increased distance by 33.11% and reduced $PM_{2.5}$ exposure by 26.02% on average, with an exposure per distance ratio of 0.4146. Results using predicted values were similar, with a 28.50% distance increase, a 25.88% exposure reduction, and a ratio of 0.3908 on average. The maximum reductions in the exposure per distance ratio reached 85.30% and 91.06% when using interpolated and predicted data, respectively. The similarity between outcomes based on interpolated and predicted data also suggests

that this approach remains effective even when real-time air quality measurements are unavailable.

Table 5.6: Comparison between the greenest and shortest paths.

	Greenest v.s. Shortest (Interpolated)			**Greenest v.s. Shortest (Predicted)**		
	d ↑	**p ↓**	**p/d ↓**	**d ↑**	**p ↓**	**p/d ↓**
mean	0.3311	0.2602	0.4146	0.2850	0.2588	0.3908
std	0.2886	0.1692	0.2018	0.2619	0.2032	0.2317
min	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
median	0.2676	0.2374	0.4418	0.2210	0.2163	0.3843
max	2.1378	0.7951	0.8530	1.8192	0.8905	0.9106

The values represent the average percentage difference in travel distance and $PM_{2.5}$ exposure between greenest and shortest routes, computed over 1,000 randomly generated trips. On average, selecting the interpolated greenest route increased travel distance by 33.11% and reduced $PM_{2.5}$ exposure by 26%.

5.7 Discussion & Limitations

This section analyses the performance of different classic and learning-based models in imputation, prediction, and application tasks introduced in Section 5.6. We begin with the discussion of temporal and spatial interpolation methods for missing data, followed by a comparison between classic ML and DL approaches for short-term spatiotemporal forecasting. The final part focuses on how multimodal data integration supports the pollution-aware route planning. Throughout the section, we reflect on the influence of data quality, model selection, and task requirements, and discuss several limitations that can be addressed in future work.

5.7.1 Interpolation & Prediction Methods for Missing Data

This study applied a two-step strategy combining temporal imputation and spatial interpolation to address missing values in urban air quality data. Among the temporal models evaluated, LSTM performed best across all evaluation metrics, including MSE, MAE, and KLD. Its gating mechanism may have helped reduce the impact of noise in the input sequence and preserve relevant historical information, leading to more stable results compared to RNN and GRU.

For the spatial interpolation step, classic interpolation methods, including IDW and KNN, outperformed learning-based models, such as RF and GCN. This may be due to the sparsity and irregular distribution of our data, where distance-based methods can directly utilise proximity information, but DL models often rely on denser and more structured input to generalise effectively. In this case, the limited data coverage may have affected their ability to learn useful spatial patterns.

These findings underline a key consideration in multimodal data processing tasks: advanced learning-based models are not universally superior, and their performance highly depends on data quality and density. Combining statistical interpolation with ML, especially when adjusted according to data sparsity and distribution, could offer a practical way to improve imputation performance in similar settings.

5.7.2 Learning-Based Methods for Spatiotemporal Prediction

The forecasting experiments were conducted and compared based on multiple classic ML and DL methods. RF achieved the lowest MSE and MAE, which is related to its ability to handle structured tabular inputs and capture nonlinear patterns without large amounts of training data. It also showed robustness to noise and redundancy, which can be beneficial in short-term $PM_{2.5}$ prediction tasks where data coverage is uneven across grid cells.

In comparison, DL models such as Conv-LSTM produced slightly higher MAE but achieved the lowest KLD, presenting better alignment with the overall distribution. This may reflect the strength of its architecture in learning both spatial and temporal structure, as it combines convolutional layers for spatial distribution with LSTM units for temporal patterns.

Overall, the results indicate that different models concentrate on various aspects of prediction, so the selection of models should depend on whether the task requires higher accuracy at individual locations or better preservation of spatial patterns. Where accuracy and consistency are both important, ensemble approaches that

combine models with complementary strengths, such as RF and Conv-LSTM, might improve overall performance.

5.7.3 Multimodal Data for Pollution-Aware Route Planning

The case study on pollution-aware routing shows that predicted $PM_{2.5}$ values can support exposure reduction for vulnerable SMS users, even without ground truth measurements. In most cases, meaningful improvements can be achieved with relatively small increases in travel distance. This suggests that the forecasting models used in this study provide considerable accuracy for routing decisions, especially when live sensor data are limited or unavailable. The usage of multimodal data, including stationary and mobile pollutant data, as well as environmental and contextual features, helped extend coverage across both temporal and spatial domains.

Future enhancements could consider additional pollutants, such as NO, NO_2, O_3, CO, CO_2, SO_2, and PM_{10}, and explore routing strategies aimed at reducing exposure to multiple pollutants. The performance of the optimisation method could also be evaluated under different pollutant weights.

5.8 Summary

This study proposed a complete framework to address the challenge of reducing pollution exposure for vulnerable SMS users by recommending routes based on predicted $PM_{2.5}$. To address the challenges caused by a multimodal dataset with a missing rate of 89.64%, we adopted a two-stage imputation strategy combining temporal sequence modelling and spatial interpolation. Comparative experiments demonstrated that LSTM models effectively reconstructed missing temporal patterns, while IDW performed best for sparse spatial interpolation. For short-term spatiotemporal prediction, RF and Conv-LSTM achieved considerable results, balancing prediction accuracy and distribution. A comparative summary of the methods and data types used in this study, alongside those employed in existing research,

Table 5.7: Summary of data imputation and prediction methods.

Study	**Method**	**Problem Addressed**	**Research Target**
[258]	Bi-GRU	Time serial data imputation	air pollutant
[259]	LSTM + GA	Time serial data imputation	air pollutant
[151]	IDW	Spatial data imputation	Walker Lake data
[260]	LSTM	Time serial data imputation	air pollutant, stock prices
[152]	KNN-ST	Spatiotemporal data imputation	temperature, air pollutant
[153]	RF	Spatiotemporal data imputation	aerosol optical depth
[155]	GCN	Spatial data prediction	cigarette consumption
[156]	LSTM	Spatiotemporal data prediction	air pollutant
[265]	ARIMA-LSTM	Time serial data prediction	air pollutant
[266]	RF	Time serial data prediction	air pollutant
[158]	GRU	Time serial data prediction	air pollutant
[157]	LSSTA	Spatiotemporal data prediction	coordinates trajectory
[268]	TCN	Spatiotemporal data prediction	traffic flow
[269]	Conv-LSTM	Spatiotemporal data prediction	air pollutant
This work	LSTM, IDW, RF, Conv-LSTM	Time serial data imputation, spatial data imputation, spatiotemporal data prediction	air pollutant

is provided in Table 5.7, highlighting the methodological positioning of our approach. These reconstructed and predicted pollutant maps were then used to generate route recommendations, leading to substantial reductions in estimated pollution exposure with only small increases in travel distance. Overall, the results highlight the potential of using multimodal environmental data and ML techniques to support healthier and more informed mobility choices in urban environments.

Chapter 6

Conclusions & Future Work

This chapter concludes the thesis by summarising the key work presented in each chapter and outlining potential directions for future research.

6.1 Conclusions

This thesis investigated the integration of multimodal data with learning-based methods to address three key challenges for SMS as outlined in Chapter 1, namely (1) parking management; (2) energy consumption modelling, and (3) air pollution exposure. By examining three use cases, i.e., parking recommendation, energy consumption prediction, and pollution-aware routing, this thesis demonstrates the value of employing data from multiple modalities and applying learning-based approaches to develop effective and applicable solutions.

The initial contribution of this thesis is the demonstration of the importance of multimodal data in SMS. Across all three case studies, the integration of multimodal data significantly enhanced the robustness and contextual sensitivity of the developed systems. Specifically, in the U-Park system, multimodal data enabled early and accurate destination predictions, which in turn supported personalised parking recommendations and improved user experience by providing smooth and continuous service even based on minimised user input. In the task of energy consumption prediction, the combination of raw sensor data with physics-based model

features allowed for detailed modelling of energy usage dynamics in EVs. Similarly, in pollution-aware routing, the inclusion of multimodal data provided information from more perspectives, contributing to the reconstruction of the high-resolution $PM_{2.5}$ spatiotemporal distribution, which is essential for route optimisation.

The second major contribution lies in the adjustment of learning-based models according to the unique characteristics of each task, supporting effective decision-making under real-world data constraints. Instead of relying on a single modelling approach, this thesis investigates various traditional and learning-based methods to address the challenges in SMS. Model selection was guided by the nature of each problem. For instance, an ARNN was applied to partial GPS sequences in the the U-Park system, achieving a trip destination prediction accuracy of 97.33%. In the energy consumption prediction work, LSTM models became the most effective local model under FL strategies, providing robust performance despite privacy constraints. In the air quality case study, classic methods such as RF and even IDW demonstrated strong performance in spatiotemporal prediction and spatial interpolation, respectively. However, DL models like LSTM and Conv-LSTM also achieved competitive results, with Conv-LSTM showing superior capacity to capture spatiotemporal dependencies and achieving stronger alignment in data distribution, i.e., better KLD, than baseline models. These results suggest that RNN-based methods have strong generalisability across different tasks involving temporal or spatiotemporal dependencies, which are common in SMS scenarios. Meanwhile, the considerable performance of classic models such as RF highlights the continued value of ensemble approaches, especially when dealing with structured missing data.

The final and perhaps most practical contribution of this thesis is the application of data-driven insights into actionable system-level improvements. In U-Park, parking recommendations increased the probability of finding available spaces by an average of 24.91%, with improvements reaching up to 29.66% in high-demand areas. In the energy prediction study, the proposed methods were designed for integration with real-world edge-cloud infrastructures, supporting more efficient fleet manage-

ment and charging strategies. In the routing system, personalised path suggestions reduced predicted $PM_{2.5}$ exposure by an average of 25.88%, with only minimal increases in travel distance, providing visible health benefits to vulnerable SMS users.

While the results presented in this thesis are encouraging, several challenges still remain in studies related to SMS. On the one hand, the quality and complexity of multimodal data will lead to notable limitations to effective learning. Missing values, uneven sampling, and inconsistent coverage across sources, e.g., GPS trajectories, sensor readings, weather data, are very common, which usually decreases predictive accuracy and complicates the integration of multimodal data sources. On the other hand, model explainability and transparency are important but under-explored in this thesis. While DL methods often achieve strong performance, their black-box nature makes them difficult to trust and deploy in real-world decision-making settings. For SMS, where model outputs may influence urban management, individual choices, and even public health, understanding the decision-making logic behind the model is essential. Addressing these challenges about data quality and model explainability would benefit the development of not only accurate and scalable, but also trustworthy and socially responsible learning-based systems.

In conclusion, this thesis presents how multimodal data and learning-based methods can be systematically integrated and applied to improve SMS performance. Their value lies not only in the accuracy of model results, but also in enabling more context-aware, privacy-conscious, and user-centric services. By addressing key challenges in prediction, optimisation, and system design across behavioural, energy-related, and environmental dimensions, this thesis provides a foundation for the development of more intelligent, efficient, and adaptable SMS services.

6.2 Answers to Research Questions

Building on the research objectives outlined in Chapter 1, the following summary highlights how each research question was addressed through the findings of the three case studies.

Parking Recommendation System

- **RQ1: Can the destination of SMS users be accurately predicted in real time based on historical travel records and partial GPS trajectories?**

  Yes. By integrating historical trip records with real-time GPS data, the proposed ARNN-based model dynamically refined destination predictions during the trip, achieving an accuracy exceeding 97.33%.

- **RQ2: How can multimodal data (e.g., weather conditions) improve the prediction of parking space availability near the targeted destination?**

  The inclusion of multimodal contextual data, such as weather conditions and infrastructure information, enabled the parking availability prediction model to identify suitable stations more effectively, supporting timely and reliable recommendations under varying demand conditions.

- **RQ3: Can a personalised parking recommendation system reduce the likelihood of improper parking and improve the overall user experience in SMS?**

  Yes. The U-Park system significantly improved the chance of securing a parking spot, with an average gain of 24.91% and up to 29.66% in constrained situations, demonstrating its potential to reduce improper parking and enhance user satisfaction through proactive and personalised guidance.

Energy Consumption Prediction

- **RQ1: How to accurately model the energy consumption of shared EVs using real-world sensor data and simulation-based driving features?**

  By utilising multimodal inputs, including GPS, speed, altitude, and model-

generated features, the proposed framework achieved high prediction accuracy. The LSTM-based local model reduced the MAE by up to 67.84% (from 1.5723 to 0.5057), with an average MAE of 0.7847 across all vehicles.

- **RQ2: Can FL achieve comparable or improved predictive performance compared to centralised models, while avoiding direct data sharing?**

  Yes. The evaluation of multiple FL strategies demonstrated that federated approaches (especially using LSTM with FedAvg) can achieve strong predictive performance while preserving data privacy, confirming their viability as alternatives to centralised training.

- **RQ3: How do different FL strategies perform under realistic conditions, including centralised and decentralised environments?**

  Comparative analysis showed that model performance is sensitive to factors such as iteration number, data split, and temporal input length. A configuration using 15 iterations, an 8:1:1 data split, and 60 input timestamps proved most effective, with FedAvg consistently outperforming other aggregation methods.

Pollution-Aware Route Planning

- **RQ1: How to construct high-resolution real-time air pollution maps by integrating multimodal data with high missing rates?**

  A two-stage imputation strategy combining LSTM-based temporal modelling and IDW-based spatial interpolation enabled the reconstruction of high-resolution $PM_{2.5}$ maps despite a missing rate of 89.64%, effectively handling both temporal gaps and spatial sparsity.

- **RQ2: Which imputation and/or prediction methods are most suitable for reconstructing spatiotemporal pollutant distributions under sparse and heterogeneous conditions?**

Experimental comparisons showed that LSTM performed best for temporal imputation, IDW for spatial interpolation, and both RF and Conv-LSTM offered strong performance for short-term spatiotemporal prediction, demonstrating a balance between accuracy and distributional alignment.

- **RQ3: Can predicted pollution levels be effectively integrated into real-time route planning systems to reduce exposure for vulnerable road users without significantly increasing trip distance?**

  Yes. The integration of predicted pollutant maps into the routing system enabled route recommendations that substantially reduced estimated $PM_{2.5}$ exposure for users, making it possible to support healthier urban mobility decisions with only small increases in travel distance.

6.3 Future Work

Although the systems developed in this thesis achieved considerable progress in applying learning-based methods and multimodal data to SMS, several directions still can be investigated to extend the contributions of this thesis and address some unresolved challenges identified across the case studies. This section summarises key opportunities for improvement and extension across all three case studies, aiming to provide a unified view of future research directions.

6.3.1 Improving Data Quality & Fusion

Although multimodal data were employed across all three case studies, high missing rates and inconsistencies between sources continued to constrain model performance. Despite employing rich multimodal datasets, all three studies encountered limitations related to data sparsity, inconsistency, or missing values. High-resolution environmental and behavioural data are often incomplete due to sensor malfunctions, transmission delays, or limited coverage.

Future work should investigate more robust spatiotemporal imputation techniques, uncertainty-aware fusion methods, and adaptive mechanisms for incomplete inputs may help address these issues. Moreover, the introduction of shared standards or interoperable protocols for data collection will enable smoother fusion and wider use. In addition, the introduction of standardised data formats or interoperable protocols would enable smoother data fusion and wider use.

6.3.2 Advancing Modelling Techniques

While this thesis demonstrated the effectiveness of RNN-based models and FL techniques in time-serial data processing, ensemble learning remains relatively underexplored and may offer further improvements by combining strengths of different modelling approaches. For instance, combining distribution-aware DL models, e.g., Conv-LSTM with classic models, e.g., RF, may enable better trade-offs between accuracy, generalisability, and computational cost. Furthermore, integrating GNN-based methods and hybrid spatiotemporal architectures may better capture the structured dependencies in SMS. For instance, GNNs may better capture the networked nature of shared mobility infrastructure (e.g., station locations, routes) for tasks like demand forecasting or parking availability. Additionally, incorporating uncertainty-aware techniques could enhance model reliability, especially when dealing with missing, noisy, or biased data.

6.3.3 Extending System Functions & Integration

Each case study addressed a unique operational challenge, but future efforts could explore their integration into a unified decision-support platform. For example, pollution-aware routing could be incorporated into the U-Park journey planning system, enabling users to find parking spots not only based on availability and preferences but also with respect to exposure risks.

System scope can also be extended in multiple directions. For instance, multi-objective optimisation techniques could help balance competing needs, such as travel

time, pollutant exposure, and energy usage. The inclusion of more vehicle types (e.g., cargo bikes, segways) and other pollutants (e.g., NO_2, O_3, or pollen) will bring benefits to a wider range of users and expand SMS to broader environmental contexts.

6.3.4 Model Explainability & User Trust

As data-driven systems gradually influence mobility decisions, improving model transparency and interpretability becomes more and more important. Future work should explore explainable model architectures, post-hoc explanation techniques (e.g., SHAP and LIME), or human-in-the-loop frameworks to provide transparency and support user confidence. For instance, SHAP can be applied to better explain feature contributions in ML models, which may help operators understand why a particular trip was predicted to consume more energy, e.g., due to steep road gradient or cold weather, thereby improving interpretability and user confidence in energy management systems. These efforts are crucial for the widespread adoption of intelligent SMS in practice.

Bibliography

[1] Mona Mahrous Abdel Wahed Ahmed and Nanis Abd El Monem, "Sustainable and green transportation for better quality of life case study greater cairo - egypt," *HBRC Journal*, vol. 16, no. 1, pp. 17–37, Jan. 2020, ISSN: 1687-4048. DOI: 10.1080/16874048.2020.1719340.

[2] Merieke Stevens, "Introduction to the special issue on mobility, climate change, and economic inequality," *Journal of Operations Management*, vol. 69, no. 1, pp. 4–8, Jan. 2023, ISSN: 1873-1317. DOI: 10.1002/joom.1233.

[3] Shanjun Li, Jianwei Xing, Lin Yang, and Fan Zhang, *Transportation and the Environment: A Review of Empirical Literature*. World Bank, Washington, DC, Oct. 2020. DOI: 10.1596/1813-9450-9421.

[4] Pedram Saeidizand, Koos Fransen, and Kobe Boussauw, "Revisiting car dependency: A worldwide analysis of car travel in global metropolitan areas," *Cities*, vol. 120, p. 103 467, Jan. 2022, ISSN: 0264-2751. DOI: 10.1016/j.cities.2021.103467.

[5] Aleksandra Koźlak, "The relationship between the concepts of sharing economy and smart cities: The case of shared mobility and smart transport," *International Journal of Sustainable Society*, vol. 12, no. 2, p. 152, 2020, ISSN: 1756-2546. DOI: 10.1504/ijssoc.2020.107894.

[6] Edgar Vazquez and Dario Landa-Silva, "Towards blockchain-based ride-sharing systems," in *Proceedings of the 10th International Conference on Operations Research and Enterprise Systems*, SCITEPRESS - Science and Technology Publications, 2021. DOI: 10.5220/0010323204460452.

[7] Sen Yan, Mingming Liu, and Noel E. O'Connor, "Parking behaviour analysis of shared e-bike users based on a real-world dataset - a case study in dublin, ireland," in *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, IEEE, Jun. 2022, pp. 1–6. DOI: 10.1109/vtc2022-spring54318.2022.9860871.

[8] Charilaos Latinopoulos, Agathe Patrier, and Aruna Sivakumar, "Planning for e-scooter use in metropolitan cities: A case study for paris," *Transportation Research Part D: Transport and Environment*, vol. 100, p. 103 037, Nov. 2021, ISSN: 1361-9209. DOI: 10.1016/j.trd.2021.103037.

[9] Xijie Xu, Jie Wang, Stefan Poslad, Xiaoping Rui, Guangyuan Zhang, and Yonglei Fan, "Exploring intra-urban human mobility and daily activity patterns from the lens of dockless bike-sharing: A case study of beijing, china," *International Journal of Applied Earth Observation and Geoinformation*, vol. 122, p. 103 442, Aug. 2023, ISSN: 1569-8432. DOI: 10.1016/j.jag.2023.103442.

[10] Elena Higueras-Castillo, Zoran Kalinic, Veljko Marinkovic, and Francisco J. Liébana-Cabanillas, "A mixed analysis of perceptions of electric and hybrid vehicles," *Energy Policy*, vol. 136, p. 111 076, Jan. 2020, ISSN: 0301-4215. DOI: 10.1016/j.enpol.2019.111076.

[11] Tim Unterluggauer, Jeppe Rich, Peter Bach Andersen, and Seyedmostafa Hashemi, "Electric vehicle charging infrastructure planning for integrated transportation and power distribution networks: A review," *eTransportation*, vol. 12, p. 100 163, May 2022, ISSN: 2590-1168. DOI: 10.1016/j.etran.2022.100163.

[12] Fayez Alanazi, "Electric vehicles: Benefits, challenges, and potential solutions for widespread adaptation," *Applied Sciences*, vol. 13, no. 10, p. 6016, May 2023, ISSN: 2076-3417. DOI: 10.3390/app13106016.

[13] Nathan Gaw, Safoora Yousefi, and Mostafa Reisi Gahrooei, "Multimodal data fusion for systems improvement: A review," *IISE Transactions*, vol. 54, no. 11, pp. 1098–1116, Dec. 2021, ISSN: 2472-5862. DOI: 10.1080/24725854.2021.1987593.

[14] Laura Alessandretti, Luis Guillermo Natera Orozco, Meead Saberi, Michael Szell, and Federico Battiston, "Multimodal urban mobility and multilayer transport networks," *Environment and Planning B: Urban Analytics and City Science*, vol. 50, no. 8, pp. 2038–2070, Jul. 2022, ISSN: 2399-8091. DOI: 10.1177/23998083221108190.

[15] Yue Ding, Sen Yan, Maqsood Hussain Shah, Hongyuan Fang, Ji Li, and Mingming Liu, "Data-driven energy consumption modelling for electric micromobility using an open dataset," in *2024 IEEE Transportation Electrification Conference and Expo (ITEC)*, IEEE, Jun. 2024, pp. 1–7. DOI: 10.1109/itec60657.2024.10599070.

[16] Satish Sajja, Harshit Pandey, Aman Sharma, Bhawana, Pvsn Murty, and Banajit Rajbongshi, "Real-time traffic and traffic flow prediction using machine learning algorithm for optimal route planning in electric vehicles," in *2024 Second International Conference Computational and Characterization Techniques in Engineering & Sciences (IC3TES)*, IEEE, Nov. 2024, pp. 1–7. DOI: 10.1109/ic3tes62412.2024.10877654.

[17] Sen Yan et al., "Breathing green: Maximising health and environmental benefits for active transportation users leveraging large scale air quality data," in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Sep. 2023, pp. 5496–5503. DOI: 10.1109/itsc57777.2023.10421940.

[18] Songhua Hu, Hangfei Lin, Xiaohong Chen, Kun Xie, and Xiaonian Shan, "Modeling usage frequencies and vehicle preferences in a large-scale electric vehicle sharing system," *IEEE Intelligent Transportation Systems Magazine*,

vol. 14, no. 1, pp. 74–86, Jan. 2022, ISSN: 1941-1197. DOI: 10.1109/mits.2019.2953561.

[19] Zineb Mahrez, Essaid Sabir, Elarbi Badidi, Walid Saad, and Mohamed Sadik, "Smart urban mobility: When mobility systems meet smart data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6222–6239, Jul. 2022, ISSN: 1558-0016. DOI: 10.1109/tits.2021.3084907.

[20] Khaled Bayoudh, Raja Knani, Fayçal Hamdaoui, and Abdellatif Mtibaa, "A survey on deep multimodal learning for computer vision: Advances, trends, applications, and datasets," *The Visual Computer*, vol. 38, no. 8, pp. 2939–2970, Jun. 2021, ISSN: 1432-2315. DOI: 10.1007/s00371-021-02166-7.

[21] Xiao Jiang, Zean Tian, and Kenli Li, "A graph-based approach for missing sensor data imputation," *IEEE Sensors Journal*, vol. 21, no. 20, pp. 23 133–23 144, Oct. 2021, ISSN: 2379-9153. DOI: 10.1109/jsen.2021.3106656.

[22] Katrina Blazek, Anita van Zwieten, Valeria Saglimbene, and Armando Teixeira-Pinto, "A practical guide to multiple imputation of missing data in nephrology," *Kidney International*, vol. 99, no. 1, pp. 68–74, Jan. 2021, ISSN: 0085-2538. DOI: 10.1016/j.kint.2020.07.035.

[23] Leandro do C. Martins, Rocio de la Torre, Canan G. Corlu, Angel A. Juan, and Mohamed A. Masmoudi, "Optimizing ride-sharing operations in smart sustainable cities: Challenges and the need for agile algorithms," *Computers & Industrial Engineering*, vol. 153, p. 107 080, Mar. 2021, ISSN: 0360-8352. DOI: 10.1016/j.cie.2020.107080.

[24] Man Luo, Bowen Du, Konstantin Klemmer, Hongming Zhu, and Hongkai Wen, "Deployment optimization for shared e-mobility systems with multi-agent deep neural search," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2549–2560, Mar. 2022, ISSN: 1558-0016. DOI: 10.1109/tits.2021.3125745.

[25] Hayssam Dahrouj et al., "An overview of machine learning-based techniques for solving optimization problems in communications and signal processing," *IEEE Access*, vol. 9, pp. 74 908–74 938, 2021, ISSN: 2169-3536. DOI: 10.1109/access.2021.3079639.

[26] Maxime Gueriau, Federico Cugurullo, Ransford A. Acheampong, and Ivana Dusparic, "Shared autonomous mobility on demand: A learning-based approach and its performance in the presence of traffic congestion," *IEEE Intelligent Transportation Systems Magazine*, vol. 12, no. 4, pp. 208–218, 2020, ISSN: 1941-1197. DOI: 10.1109/mits.2020.3014417.

[27] Julian Teusch et al., "A systematic literature review on machine learning in shared mobility," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 4, pp. 870–899, 2023, ISSN: 2687-7813. DOI: 10.1109/ojits.2023.3334393.

[28] Hanan Almukhalfi, Ayman Noor, and Talal H Noor, "Traffic management approaches using machine learning and deep learning techniques: A survey," *Engineering Applications of Artificial Intelligence*, vol. 133, p. 108 147, Jul. 2024. DOI: 10.1016/j.engappai.2024.108147.

[29] Julakha Jahan Jui, Mohd Ashraf Ahmad, M M Imran Molla, and Muhammad Ikram Mohd Rashid, "Optimal energy management strategies for hybrid electric vehicles: A recent survey of machine learning approaches," *J. Eng. Res.*, vol. 12, no. 3, pp. 454–467, Sep. 2024. DOI: 10.1016/j.jer.2024.01.016.

[30] Xiaocai Zhang, Zhixun Zhao, Yi Zheng, and Jinyan Li, "Prediction of taxi destinations using a novel data embedding method and ensemble learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, pp. 68–78, 2020. DOI: 10.1109/TITS.2018.2888587.

[31] Helio N. Cunha Neto, Jernej Hribar, Ivana Dusparic, Diogo Menezes Ferrazani Mattos, and Natalia C. Fernandes, "A survey on securing federated learning: Analysis of applications, attacks, challenges, and trends," *IEEE Ac-*

cess, vol. 11, pp. 41 928–41 953, 2023, ISSN: 2169-3536. DOI: 10.1109/access.2023.3269980.

[32] Sen Yan, Noel E. O'Connor, and Mingming Liu, "U-park: A user-centric smart parking recommendation system for electric shared micromobility services," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 10, pp. 5179–5193, 2024. DOI: 10.1109/TAI.2024.3428513.

[33] Sen Yan, Hongyuan Fang, Ji Li, Tomas Ward, Noel O'Connor, and Mingming Liu, "Privacy-aware energy consumption modeling of connected battery electric vehicles using federated learning," *IEEE Transactions on Transportation Electrification*, vol. 10, no. 3, pp. 6663–6675, Sep. 2024, ISSN: 2372-2088. DOI: 10.1109/tte.2023.3343106.

[34] Fatemeh Golpayegani et al., "Intelligent shared mobility systems: A survey on whole system design requirements, challenges and future direction," *IEEE Access*, vol. 10, pp. 35 302–35 320, 2022, ISSN: 2169-3536. DOI: 10.1109/access.2022.3162848.

[35] Yao Chen and Yang Liu, "Integrated optimization of planning and operations for shared autonomous electric vehicle systems," *Transportation Science*, vol. 57, no. 1, pp. 106–134, Jan. 2023, ISSN: 1526-5447. DOI: 10.1287/trsc.2022.1156.

[36] Chee Sheng Tan, Rosmiwati Mohd-Mokhtar, and Mohd Rizal Arshad, "A comprehensive review of coverage path planning in robotics using classical and heuristic algorithms," *IEEE Access*, vol. 9, pp. 119 310–119 342, 2021, ISSN: 2169-3536. DOI: 10.1109/access.2021.3108177.

[37] H. Mataifa, S. Krishnamurthy, and C. Kriger, "Volt/var optimization: A survey of classical and heuristic optimization methods," *IEEE Access*, vol. 10, pp. 13 379–13 399, 2022, ISSN: 2169-3536. DOI: 10.1109/access.2022.3146366.

[38] Yuqing Yang, Stephen Bremner, Chris Menictas, and Merlinde Kay, "Modelling and optimal energy management for battery energy storage systems in renewable energy systems: A review," *Renewable and Sustainable Energy Reviews*, vol. 167, p. 112 671, Oct. 2022, ISSN: 1364-0321. DOI: 10.1016/j.rser.2022.112671.

[39] Xinran Li, Wei Wang, Kun Jin, and Hao Gu, "A blockchain-enabled personalized charging system for electric vehicles," *Transportation Research Part C: Emerging Technologies*, vol. 161, p. 104 549, Apr. 2024, ISSN: 0968-090X. DOI: 10.1016/j.trc.2024.104549.

[40] Aayush Aggarwal, Florence Ho, and Shinji Nakadai, "Extended time dependent vehicle routing problem for joint task allocation and path planning in shared space," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Oct. 2022, pp. 12 037–12 044. DOI: 10.1109/iros47612.2022.9981570.

[41] Daniele Gammelli, Yihua Wang, Dennis Prak, Filipe Rodrigues, Stefan Minner, and Francisco Camara Pereira, "Predictive and prescriptive performance of bike-sharing demand forecasts for inventory management," *Transportation Research Part C: Emerging Technologies*, vol. 138, p. 103 571, May 2022, ISSN: 0968-090X. DOI: 10.1016/j.trc.2022.103571.

[42] Zhao Li et al., "Real-time e-bike route planning with battery range prediction," in *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, ser. WSDM '24, ACM, Mar. 2024. DOI: 10.1145/3616855.3635696.

[43] Irfan Ullah, Kai Liu, Toshiyuki Yamamoto, Rabia Emhamed Al Mamlook, and Arshad Jamal, "A comparative performance of machine learning algorithm to predict electric vehicles energy consumption: A path towards sustainability," *Energy & Environment*, vol. 33, no. 8, pp. 1583–1612, Oct. 2021. DOI: 10.1177/0958305x211044998.

[44] Hao Miao, Yan Fei, Senzhang Wang, Fang Wang, and Danyan Wen, "Deep learning based origin-destination prediction via contextual information fusion," *Multimedia Tools and Applications*, vol. 81, no. 9, pp. 12 029–12 045, Jan. 2021, ISSN: 1573-7721. DOI: 10.1007/s11042-020-10492-6.

[45] Narith Saum, Satoshi Sugiura, and Mongkut Piantanakulchai, "Short-term demand and volatility prediction of shared micro-mobility: A case study of e-scooter in thammasat university," in *2020 Forum on Integrated and Sustainable Transportation Systems (FISTS)*, IEEE, Nov. 2020. DOI: 10.1109/fists46898.2020.9264852.

[46] Douglas C Montgomery, Elizabeth A Peck, and G Geoffrey Vining, *Introduction to linear regression analysis*. John Wiley & Sons, 2021.

[47] Rudresh Jani, Palak Patel, Siddharth Joshi, and Soham Vyas, "Sam based computational analysis of root mean square error for spes using lr, rf & arima machine learning models," in *2023 3rd International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET)*, IEEE, Dec. 2023. DOI: 10.1109/icefeet59656.2023.10452190.

[48] Xiangqian Li, Keke Li, Siqi Shen, and Yaxin Tian, "Exploring time series models for wind speed forecasting: A comparative analysis," *Energies*, vol. 16, no. 23, p. 7785, Nov. 2023, ISSN: 1996-1073. DOI: 10.3390/en16237785.

[49] Jian Liang, Jinjun Tang, Fang Liu, and Yinhai Wang, "Combining individual travel preferences into destination prediction: A multi-module deep learning network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 13 782–13 793, Aug. 2022, ISSN: 1558-0016. DOI: 10.1109/tits.2021.3128153.

[50] Mahmoud Masoud, "A hybrid k-means and particle swarm optimization technique for solving the rechargeable e-scooters problem," *IEEE Access*, vol. 11, pp. 132 472–132 482, 2023, ISSN: 2169-3536. DOI: 10.1109/access.2023.3336810.

[51] Aniss Moumen, El Houcine Bouchama, and Younes El Bouzekri El Idirissi, "Data mining techniques for employability: Systematic literature review," in *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, IEEE, Dec. 2020. DOI: 10.1109/icecocs50124.2020.9314555.

[52] Muhammad Imran Faizi and Syed Muhammad Adnan, "Improved segmentation model for melanoma lesion detection using normalized cross-correlation-based k-means clustering," *IEEE Access*, vol. 12, pp. 20 753–20 766, 2024, ISSN: 2169-3536. DOI: 10.1109/access.2024.3360223.

[53] Irfan Ullah, Kai Liu, Toshiyuki Yamamoto, Muhammad Zahid, and Arshad Jamal, "Electric vehicle energy consumption prediction using stacked generalization: An ensemble learning approach," *International Journal of Green Energy*, vol. 18, no. 9, pp. 896–909, Feb. 2021. DOI: 10.1080/15435075.2021.1881902.

[54] Karim Ouf, Hassan Soubra, and Ahmed Mazhr, "E-bike energy needs estimation based on route characteristics and rider behavior," in *2023 Eleventh International Conference on Intelligent Computing and Information Systems (ICICIS)*, IEEE, Nov. 2023. DOI: 10.1109/icicis58388.2023.10391196.

[55] Yimin Liu, Xinping Yang, Siqing Niu, Shaoming Zheng, Qian Chen, and Ancheng Xue, "Defect rating for different production models of line protection devices based on decision trees," in *2019 IEEE 3rd Conference on Energy Internet and Energy System Integration (EI2)*, IEEE, Nov. 2019. DOI: 10.1109/ei247390.2019.9062248.

[56] João Carlos N. Bittencourt, Daniel G. Costa, Paulo Portugal, and Francisco Vasques, "Towards lightweight fire detection at the extreme edge based on decision trees," in *2024 IEEE 22nd Mediterranean Electrotechnical Conference (MELECON)*, IEEE, Jun. 2024. DOI: 10.1109/melecon56669.2024.10608598.

[57] Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci, "Deep neural networks and tabular data: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 6, pp. 7499–7519, Jun. 2024, ISSN: 2162-2388. DOI: 10.1109/tnnls.2022.3229161.

[58] Charalampos Bratsas, Kleanthis Koupidis, Josep-Maria Salanova, Konstantinos Giannakopoulos, Aristeidis Kaloudis, and Georgia Aifadopoulou, "A comparison of machine learning methods for the prediction of traffic speed in urban places," *Sustainability*, vol. 12, no. 1, p. 142, Dec. 2019, ISSN: 2071-1050. DOI: 10.3390/su12010142.

[59] Hatem Abdelaty, Abdullah Al-Obaidi, Moataz Mohamed, and Hany E.Z. Farag, "Machine learning prediction models for battery-electric bus energy consumption in transit," *Transportation Research Part D: Transport and Environment*, vol. 96, p. 102 868, Jul. 2021, ISSN: 1361-9209. DOI: 10.1016/j.trd.2021.102868.

[60] Hoa Nguyen, Minh Nguyen, and Qian Sun, "Electric scooter and its rider detection framework based on deep learning for supporting scooter-related injury emergency services," in *Geometry and Vision*. Springer International Publishing, 2021, pp. 233–246, ISBN: 9783030720735. DOI: 10.1007/978-3-030-72073-5_18.

[61] Eunji Kim, Hanyoung Ryu, Hyunji Oh, and Namwoo Kang, "Safety monitoring system of personal mobility driving using deep learning," *Journal of Computational Design and Engineering*, vol. 9, no. 4, pp. 1397–1409, Jul. 2022, ISSN: 2288-5048. DOI: 10.1093/jcde/qwac061.

[62] Nobuhiko Yamaguchi, Hiroki Tokumaru, Osamu Fukuda, Hiroshi Okumura, and Wen Liang Yeoh, "Bicycle-based collision prevention system using pedestrian trajectory prediction," in *2022 Tenth International Symposium on Com-*

puting and Networking Workshops (CANDARW), IEEE, Nov. 2022. DOI: 10.1109/candarw57323.2022.00013.

[63] Woo-Jin Jang, Dong-Hyun Kim, and Si-Hyung Lim, "An ai safety monitoring system for electric scooters based on the number of riders and road types," *Sensors*, vol. 23, no. 22, p. 9181, Nov. 2023, ISSN: 1424-8220. DOI: 10.3390/s23229181.

[64] Chinmaya Kaundanya, Paulo Cesar, Barry Cronin, Andrew Fleury, Mingming Liu, and Suzanne Little, "Using attention mechanisms in compact cnn models for improved micromobility safety through lane recognition," in *Proceedings of the 10th International Conference on Vehicle Technology and Intelligent Transport Systems*, SCITEPRESS - Science and Technology Publications, 2024. DOI: 10.5220/0012600300003702.

[65] Jaehyung Lee and Jinhee Kim, "Evaluation of spatial and temporal performance of deep learning models for travel demand forecasting: Application to bike-sharing demand forecasting," *Journal of Advanced Transportation*, vol. 2022, Rosa G. González-Ramírez, Ed., pp. 1–13, Jun. 2022, ISSN: 0197-6729. DOI: 10.1155/2022/5934670.

[66] Jian Jiang, Fei Lin, Jin Fan, Hang Lv, and Jia Wu, "A destination prediction network based on spatiotemporal data for bike-sharing," *Complexity*, vol. 2019, pp. 1–14, Jan. 2019, ISSN: 1099-0526. DOI: 10.1155/2019/7643905.

[67] Shatrughan Modi, Jhilik Bhattacharya, and Prasenjit Basak, "Estimation of energy consumption of electric vehicles using deep convolutional neural network to reduce driver's range anxiety," *ISA Transactions*, vol. 98, pp. 454–470, Mar. 2020, ISSN: 0019-0578. DOI: 10.1016/j.isatra.2019.08.055.

[68] Ljubisa Sehovac and Katarina Grolinger, "Deep learning for load forecasting: Sequence to sequence recurrent neural networks with attention," *IEEE Access*, vol. 8, pp. 36 411–36 426, 2020, ISSN: 2169-3536. DOI: 10.1109/access.2020.2975738.

[69] Hansika Hewamalage, Christoph Bergmeir, and Kasun Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *International Journal of Forecasting*, vol. 37, no. 1, pp. 388–427, Jan. 2021, ISSN: 0169-2070. DOI: 10.1016/j.ijforecast.2020.06.008.

[70] Asif Nawaz, Zhiqiu Huang, and Senzhang Wang, "Ssmdl: Semi-supervised multi-task deep learning for transportation mode classification and path prediction with gps trajectories," in *Lecture Notes in Computer Science*. Springer International Publishing, 2020, pp. 391–405, ISBN: 9783030602901. DOI: 10.1007/978-3-030-60290-1_31.

[71] Zhengyong Chen, Hongde Wu, Noel E. O'Connor, and Mingming Liu, "A comparative study of using spatial-temporal graph convolutional networks for predicting availability in bike sharing schemes," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, IEEE, Sep. 2021. DOI: 10.1109/itsc48978.2021.9564831.

[72] Hongde Wu and Mingming Liu, "Lane-gnn: Integrating gnn for predicting drivers' lane change intention," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Oct. 2022. DOI: 10.1109/itsc55140.2022.9922139.

[73] Guangnian Xiao, Ruinan Wang, Chunqin Zhang, and Anning Ni, "Demand prediction for a public bike sharing program based on spatio-temporal graph convolutional networks," *Multimedia Tools and Applications*, vol. 80, no. 15, pp. 22 907–22 925, Mar. 2020, ISSN: 1573-7721. DOI: 10.1007/s11042-020-08803-y.

[74] Saidul Islam et al., "A comprehensive survey on applications of transformers for deep learning tasks," *Expert Systems with Applications*, vol. 241, p. 122 666, May 2024, ISSN: 0957-4174. DOI: 10.1016/j.eswa.2023.122666.

[75] Yiming Xu, Xilei Zhao, Xiaojian Zhang, and Mudit Paliwal, "Real-time forecasting of dockless scooter-sharing demand: A spatio-temporal multi-

graph transformer approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8507–8518, Aug. 2023, ISSN: 1558-0016. DOI: 10.1109/tits.2023.3239309.

[76] Lang Peng, Zhirong Chen, Zhangjie Fu, Pengpeng Liang, and Erkang Cheng, "Bevsegformer: Bird's eye view semantic segmentation from arbitrary camera rigs," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, IEEE, Jan. 2023, pp. 5924–5932. DOI: 10.1109/wacv56688.2023.00588.

[77] Yifeng Bai, Zhirong Chen, Zhangjie Fu, Lang Peng, Pengpeng Liang, and Erkang Cheng, "Curveformer: 3d lane detection by curve propagation with curve queries and attention," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, May 2023. DOI: 10.1109/icra48891.2023.10161160.

[78] Rohith Puvvala Subramanyam, Abhay Naik, and Mahima Agumbe Suresh, "Accident prediction on e-bikes using computer vision," in *2023 IEEE Ninth International Conference on Big Data Computing Service and Applications (BigDataService)*, IEEE, Jul. 2023. DOI: 10.1109/bigdataservice58306.2023.00040.

[79] Khalil Sabri, Célia Djilali, Guillaume-Alexandre Bilodeau, Nicolas Saunier, and Wassim Bouachir, "Detection of micromobility vehicles in urban traffic videos," *Proceedings of the 21st Conference on Robots and Vision*, May 2024. DOI: 10.21428/d82e957c.abc3243f.

[80] Kai Su, Yoichi Tomioka, Qiangfu Zhao, and Yong Liu, "Yolic: An efficient method for object localization and classification on edge devices," *Image and Vision Computing*, vol. 147, p. 105 095, Jul. 2024, ISSN: 0262-8856. DOI: 10.1016/j.imavis.2024.105095.

[81] Mohamed Abouelela, Cheng Lyu, and Constantinos Antoniou, "Exploring the potentials of open-source big data and machine learning in shared mobility

fleet utilization prediction," *Data Science for Transportation*, vol. 5, no. 2, Apr. 2023, ISSN: 2948-1368. DOI: 10.1007/s42421-023-00068-9.

[82] Narith Saum, Satoshi Sugiura, and Mongkut Piantanakulchai, "Optimizing shared e-scooter operations under demand uncertainty: A framework integrating machine learning and optimization techniques," *IEEE Access*, vol. 12, pp. 26 957–26 977, 2024, ISSN: 2169-3536. DOI: 10.1109/access.2024.3365947.

[83] César Peláez-Rodríguez, Jorge Pérez-Aracil, Dušan Fister, Ricardo Torres-López, and Sancho Salcedo-Sanz, "Bike sharing and cable car demand forecasting using machine learning and deep learning multivariate time series approaches," *Expert Systems with Applications*, vol. 238, p. 122 264, Mar. 2024, ISSN: 0957-4174. DOI: 10.1016/j.eswa.2023.122264.

[84] Dinh Viet Cuong, Vuong M. Ngo, Paolo Cappellari, and Mark Roantree, "Analyzing shared bike usage through graph-based spatio-temporal modeling," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 5, pp. 115–131, 2024, ISSN: 2687-7813. DOI: 10.1109/ojits.2024.3350213.

[85] Yi Shi, Liumei Zhang, Shengnan Lu, and Qiao Liu, "Short-term demand prediction of shared bikes based on lstm network," *Electronics*, vol. 12, no. 6, p. 1381, Mar. 2023, ISSN: 2079-9292. DOI: 10.3390/electronics12061381.

[86] Wei Wang, Xiaofeng Zhao, Zhiguo Gong, Zhikui Chen, Ning Zhang, and Wei Wei, "An attention-based deep learning framework for trip destination prediction of sharing bike," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4601–4610, Jul. 2021, ISSN: 1558-0016. DOI: 10.1109/tits.2020.3008935.

[87] Aoyong Li, Pengxiang Zhao, He Haitao, Ali Mansourian, and Kay W. Axhausen, "How did micro-mobility change in response to covid-19 pandemic? a case study based on spatial-temporal-semantic analytics," *Computers, Envi-*

ronment and Urban Systems, vol. 90, p. 101 703, Nov. 2021, ISSN: 0198-9715. DOI: 10.1016/j.compenvurbsys.2021.101703.

[88] Ziyin Zhang, Fei Yang, Li Wang, Zhigang Wang, Tian Zhou, and Yansong He, "Enhancing urban vehicle destination prediction through trajectory metadata integration and machine learning," in *2023 7th Asian Conference on Artificial Intelligence Technology (ACAIT)*, IEEE, Nov. 2023. DOI: 10.1109/acait60137.2023.10528407.

[89] Athanasios Tsiligkaridis, Jing Zhang, Ioannis Ch. Paschalidis, Hiroshi Taguchi, Satoko Sakajo, and Daniel Nikovski, "Context-aware destination and time-to-destination prediction using machine learning," in *2022 IEEE International Smart Cities Conference (ISC2)*, IEEE, Sep. 2022. DOI: 10.1109/isc255366.2022.9922593.

[90] Juanjuan Zhao, Liutao Zhang, Jiexia Ye, and Chengzhong Xu, "Mdlf: A multi-view-based deep learning framework for individual trip destination prediction in public transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 13 316–13 329, Aug. 2022, ISSN: 1558-0016. DOI: 10.1109/tits.2021.3123342.

[91] Hakan İnaç, Yunus Emre Ayözen, Abdulkadir Atalan, and Cem undefinedağrı Dönmez, "Estimation of postal service delivery time and energy cost with e-scooter by machine learning algorithms," *Applied Sciences*, vol. 12, no. 23, p. 12 266, Nov. 2022, ISSN: 2076-3417. DOI: 10.3390/app122312266.

[92] Christos Gioldasis, Zoi Christoforou, and Aikaterini Katsiadrami, "Usage factors influencing e-scooter energy consumption: An empirical investigation," *Journal of Cleaner Production*, vol. 452, p. 142 165, May 2024, ISSN: 0959-6526. DOI: 10.1016/j.jclepro.2024.142165.

[93] Sugam Pokharel, Pradip Sah, and Deepak Ganta, "Improved prediction of total energy consumption and feature analysis in electric vehicles using machine learning and shapley additive explanations method," *World Electric*

Vehicle Journal, vol. 12, no. 3, p. 94, Jun. 2021, ISSN: 2032-6653. DOI: 10.3390/wevj12030094.

[94] Karim Aboeleneen, Nizar Zorba, and Ahmed M. Massoud, "Reinforcement learning-based e-scooter energy minimization using optimized speed-route selection," *IEEE Access*, vol. 12, pp. 66 167–66 184, 2024, ISSN: 2169-3536. DOI: 10.1109/access.2024.3395286.

[95] Maqsood Hussain Shah, Yue Ding, Shaoshu Zhu, Yingqi Gu, and Mingming Liu, *Optimal design and implementation of an open-source emulation platform for user-centric shared e-mobility services*, 2024. DOI: 10.48550/ARXIV.2403.07964.

[96] Darren M. Scott, Wei Lu, and Matthew J. Brown, "Route choice of bike share users: Leveraging gps data to derive choice sets," *Journal of Transport Geography*, vol. 90, p. 102 903, Jan. 2021, ISSN: 0966-6923. DOI: 10.1016/j.jtrangeo.2020.102903.

[97] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2018. DOI: 10.1109/cvpr.2018.00474.

[98] Kumar Apurv, Renran Tian, and Rini Sherony, "Detection of e-scooter riders in naturalistic scenes," *arXiv preprint arXiv:2111.14060*, 2021. DOI: 10.48550/ARXIV.2111.14060.

[99] Thommas Flores, Matheus Andrade, Morsinaldo Medeiros, Miguel Amaral, Marianne Silva, and Ivanovitch Silva, "Leveraging iot and tinyml for smart battery management in electric bicycles," in *2023 Symposium on Internet of Things (SIoT)*, IEEE, Oct. 2023. DOI: 10.1109/siot60039.2023.10390169.

[100] Hui Zhang, Li Zhang, Yanjun Liu, and Lele Zhang, "Understanding travel mode choice behavior: Influencing factors analysis and prediction with ma-

chine learning method," *Sustainability*, vol. 15, no. 14, p. 11 414, Jul. 2023, ISSN: 2071-1050. DOI: 10.3390/su151411414.

[101] Yingying Xing, Ke Wang, and Jian John Lu, "Exploring travel patterns and trip purposes of dockless bike-sharing by analyzing massive bike-sharing data in shanghai, china," *Journal of Transport Geography*, vol. 87, p. 102 787, Jul. 2020, ISSN: 0966-6923. DOI: 10.1016/j.jtrangeo.2020.102787.

[102] Zhengwu Wang, Shuai Huang, Jie Wang, Denisa Sulaj, Wei Hao, and Aiwu Kuang, "Risk factors affecting crash injury severity for different groups of e-bike riders: A classification tree-based logistic regression model," *Journal of Safety Research*, vol. 76, pp. 176–183, Feb. 2021, ISSN: 0022-4375. DOI: 10.1016/j.jsr.2020.12.009.

[103] Tiago Tamagusko et al., "Data-driven approach for urban micromobility enhancement through safety mapping and intelligent route planning," *Smart Cities*, vol. 6, no. 4, pp. 2035–2056, Aug. 2023, ISSN: 2624-6511. DOI: 10.3390/smartcities6040094.

[104] Carlos Lemonde, Elisabete Arsenio, and Rui Henriques, "Public transportation multimodality in the city of lisbon," *Transportation Research Procedia*, vol. 58, pp. 75–82, 2021, ISSN: 2352-1465. DOI: 10.1016/j.trpro.2021.11.011.

[105] Zhiren Huang, Alonso Espinosa Mireles de Villafranca, and Charalampos Sipetas, "Sensing multi-modal mobility patterns: A case study of helsinki using bluetooth beacons and a mobile application," in *2022 IEEE International Conference on Big Data (Big Data)*, IEEE, Dec. 2022, pp. 2007–2016. DOI: 10.1109/bigdata55660.2022.10020578.

[106] Martina Erdelić, Tomislav Erdelić, and Tonči Carić, "Dataset for multimodal transport analytics of smartphone users - collecty," *Data in Brief*, vol. 50, p. 109 481, Oct. 2023, ISSN: 2352-3409. DOI: 10.1016/j.dib.2023.109481.

[107] Jiangtao Zhu, Ningke Xie, Zeen Cai, Wei Tang, and Xiqun (Michael) Chen, "A comprehensive review of shared mobility for sustainable transportation systems," *International Journal of Sustainable Transportation*, vol. 17, no. 5, pp. 527–551, Mar. 2022, ISSN: 1556-8334. DOI: 10.1080/15568318.2022.2054390.

[108] Arash Mousaei, Yahya Naderi, and I. Safak Bayram, "Advancing state of charge management in electric vehicles with machine learning: A technological review," *IEEE Access*, vol. 12, pp. 43 255–43 283, 2024, ISSN: 2169-3536. DOI: 10.1109/access.2024.3378527.

[109] Isabel Paulino, Sergi Lozano, and Lluís Prats, "Identifying tourism destinations from tourists' travel patterns," *Journal of Destination Marketing & Management*, vol. 19, p. 100 508, Mar. 2021, ISSN: 2212-571X. DOI: 10.1016/j.jdmm.2020.100508.

[110] Laura Höpfl, Maximilian Grimlitza, Isabella Lang, and Maria Wirzberger, "Promoting sustainable behavior: Addressing user clusters through targeted incentives," *Humanities and Social Sciences Communications*, vol. 11, no. 1, Sep. 2024, ISSN: 2662-9992. DOI: 10.1057/s41599-024-03581-6.

[111] Nina Rizun and Bartosz Duzinkewicz, "Decoding customer experience: A comparative analysis of electric and internal combustion vehicles in the u.s. market through structured topic modeling," *IEEE Access*, pp. 1–1, 2025, ISSN: 2169-3536. DOI: 10.1109/access.2025.3561219.

[112] Xiao Xiao et al., "Parking prediction in smart cities: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 10, pp. 10 302–10 326, Oct. 2023, ISSN: 1558-0016. DOI: 10.1109/tits.2023.3279024.

[113] Omar Isaac Asensio, M. Cade Lawson, and Camila Z. Apablaza, "Electric vehicle charging stations in the workplace with high-resolution data from casual and habitual users," *Scientific Data*, vol. 8, no. 1, Jul. 2021, ISSN: 2052-4463. DOI: 10.1038/s41597-021-00956-1.

[114] Maxime Gueriau and Ivana Dusparic, "Quantifying the impact of connected and autonomous vehicles on traffic efficiency and safety in mixed traffic," in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Sep. 2020, pp. 1–8. DOI: 10.1109/itsc45102.2020.9294174.

[115] Alexandra Hoess, Jonathan Lautenschlager, Johannes Sedlmeir, Gilbert Fridgen, Vincent Schlatt, and Nils Urbach, "Toward seamless mobility-as-a-service: Providing multimodal mobility through digital wallets," *Business & Information Systems Engineering*, vol. 67, no. 2, pp. 149–170, Feb. 2024, ISSN: 1867-0202. DOI: 10.1007/s12599-024-00856-9.

[116] Rui Miao, Peng Guo, Wenjie Huang, Qi Li, and Bo Zhang, "Profit model for electric vehicle rental service: Sensitive analysis and differential pricing strategy," *Energy*, vol. 249, p. 123 736, Jun. 2022, ISSN: 0360-5442. DOI: 10.1016/j.energy.2022.123736.

[117] Xin Li, Yuan Li, Tao Jia, Lin Zhou, and Ihab Hamzi Hijazi, "The six dimensions of built environment on urban vitality: Fusion evidence from multi-source data," *Cities*, vol. 121, p. 103 482, Feb. 2022, ISSN: 0264-2751. DOI: 10.1016/j.cities.2021.103482.

[118] Elhadja Chaalal, Cindy Guerlain, Enric Pardo, and Sébastien Faye, "Integrating connected and automated shuttles with other mobility systems: Challenges and future directions," *IEEE Access*, vol. 11, pp. 83 081–83 106, 2023, ISSN: 2169-3536. DOI: 10.1109/access.2023.3294110.

[119] Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 423–443, Feb. 2019, ISSN: 1939-3539. DOI: 10.1109/tpami.2018.2798607.

[120] Luis Manuel Pereira, Addisson Salazar, and Luis Vergara, "A comparative analysis of early and late fusion for the multimodal two-class problem," *IEEE*

Access, vol. 11, pp. 84 283–84 300, 2023, ISSN: 2169-3536. DOI: 10.1109/access.2023.3296098.

[121] Radhika Shetty D S, "Multi-modal fusion techniques in deep learning," *International Journal of Science and Research (IJSR)*, vol. 12, no. 9, pp. 526–532, Sep. 2023, ISSN: 2319-7064. DOI: 10.21275/sr23905100554.

[122] Luis Manuel Pereira, Addisson Salazar, and Luis Vergara, "On comparing early and late fusion methods," in *Advances in Computational Intelligence*. Springer Nature Switzerland, 2023, pp. 365–378, ISBN: 9783031430855. DOI: 10.1007/978-3-031-43085-5_29.

[123] Konrad Gadzicki, Razieh Khamsehashari, and Christoph Zetzsche, "Early vs late fusion in multimodal convolutional neural networks," in *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, IEEE, Jul. 2020, pp. 1–6. DOI: 10.23919/fusion45008.2020.9190246.

[124] Muhamad Faisal, Jenq-Shiou Leu, and Jeremie T. Darmawan, "Model selection of hybrid feature fusion for coffee leaf disease classification," *IEEE Access*, vol. 11, pp. 62 281–62 291, 2023, ISSN: 2169-3536. DOI: 10.1109/access.2023.3286935.

[125] Henry Alexander Ignatious, Hesham El-Sayed, and Parag Kulkarni, "Multilevel data and decision fusion using heterogeneous sensory data for autonomous vehicles," *Remote Sensing*, vol. 15, no. 9, p. 2256, Apr. 2023, ISSN: 2072-4292. DOI: 10.3390/rs15092256.

[126] Dr Hanmant N. Renushe et al., "Implementation of anonymous vehicle reporting and communication system for wrongly parked vehicle," *Journal of Advanced Zoology*, Jan. 2024. DOI: 10.53555/jaz.v45i1.3284.

[127] Farah Ghizzawi, Alia Galal, and Matthew J. Roorda, "Modelling parking behaviour of commercial vehicles: A scoping review," *Transport Reviews*, vol. 44, no. 4, pp. 743–765, Jan. 2024, ISSN: 1464-5327. DOI: 10.1080/01441647.2024.2305202.

[128] Anthony Kimpton, Dorina Pojani, Connor Ryan, Lisha Ouyang, Neil Sipe, and Jonathan Corcoran, "Contemporary parking policy, practice, and outcomes in three large australian cities," *Progress in Planning*, vol. 153, p. 100 506, Nov. 2021, ISSN: 0305-9006. DOI: 10.1016/j.progress.2020.100506.

[129] Marko Mladenović, Thierry Delot, Gilbert Laporte, and Christophe Wilbaut, "A scalable dynamic parking allocation framework," *Computers & Operations Research*, vol. 125, p. 105 080, Jan. 2021, ISSN: 0305-0548. DOI: 10.1016/j.cor.2020.105080.

[130] Enrico Bassetti, Andrea Berti, Alba Bisante, Andrea Magnante, and Emanuele Panizzi, "Exploiting user behavior to predict parking availability through machine learning," *Smart Cities*, vol. 5, no. 4, pp. 1243–1266, Sep. 2022, ISSN: 2624-6511. DOI: 10.3390/smartcities5040064.

[131] Jamie Arjona, MªPaz Linares, Josep Casanovas-Garcia, and Juan José Vázquez, "Improving parking availability information using deep learning techniques," *Transportation Research Procedia*, vol. 47, pp. 385–392, 2020, ISSN: 2352-1465. DOI: 10.1016/j.trpro.2020.03.113.

[132] Saba Inam, Azhar Mahmood, Shaheen Khatoon, Majed Alshamari, and Nazia Nawaz, "Multisource data integration and comparative analysis of machine learning models for on-street parking prediction," *Sustainability*, vol. 14, no. 12, p. 7317, Jun. 2022, ISSN: 2071-1050. DOI: 10.3390/su14127317.

[133] Sergio Guidon, Henrik Becker, Horace Dediu, and Kay W. Axhausen, "Electric bicycle-sharing: A new competitor in the urban transportation market? an empirical analysis of transaction data," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2673, no. 4, pp. 15–26, Mar. 2019, ISSN: 2169-4052. DOI: 10.1177/0361198119836762.

[134] Xizhen Zhou, Yanjie Ji, Yidan Yuan, Fan Zhang, and Qinhe An, "Spatiotemporal characteristics analysis of commuting by shared electric bike: A case study of ningbo, china," *Journal of Cleaner Production*, vol. 362, p. 132 337, Aug. 2022, ISSN: 0959-6526. DOI: 10.1016/j.jclepro.2022.132337.

[135] Santi Phithakkitnukooon, Karn Patanukhom, and Merkebe Getachew Demissie, "Predicting spatiotemporal demand of dockless e-scooter sharing services with a masked fully convolutional network," *ISPRS International Journal of Geo-Information*, vol. 10, no. 11, p. 773, Nov. 2021, ISSN: 2220-9964. DOI: 10.3390/ijgi10110773.

[136] Jing Luo, Dai Zhou, Zhaolong Han, Guangnian Xiao, and Yunlong Tan, "Predicting travel demand of a docked bikesharing system based on lsgc-lstm networks," *IEEE Access*, vol. 9, pp. 92 189–92 203, 2021, ISSN: 2169-3536. DOI: 10.1109/access.2021.3062778.

[137] Jianguo Chen, Kenli Li, Keqin Li, Philip S. Yu, and Zeng Zeng, "Dynamic planning of bicycle stations in dockless public bicycle-sharing system using gated graph neural network," *ACM Transactions on Intelligent Systems and Technology*, vol. 12, no. 2, pp. 1–22, Mar. 2021, ISSN: 2157-6912. DOI: 10.1145/3446342.

[138] Mingming Liu, "Fed-bev: A federated learning framework for modelling energy consumption of battery electric vehicles," in *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, IEEE, Sep. 2021, pp. 1–7. DOI: 10.1109/vtc2021-fall52928.2021.9625535.

[139] Millend Roy et al., "Reliable energy consumption modeling for an electric vehicle fleet," in *ACM SIGCAS/SIGCHI Conference on Computing and Sustainable Societies (COMPASS)*, ser. COMPASS '22, ACM, Jun. 2022. DOI: 10.1145/3530190.3534803.

[140] Apurva Pamidimukkala, Ronik Ketankumar Patel, Sharareh Kermanshachi, Jay Michael Rosenberger, and Shams Tanvir, "A review on shared mobility

and electric vehicles," in *International Conference on Transportation and Development 2023*, American Society of Civil Engineers, Jun. 2023, pp. 333–342. DOI: 10.1061/9780784484883.029.

[141] Christopher Hull, J.H. Giliomee, Katherine A. Collett, Malcolm D. McCulloch, and M.J. Booysen, "High fidelity estimates of paratransit energy consumption from per-second GPS tracking data," *Transportation Research Part D: Transport and Environment*, vol. 118, p. 103 695, May 2023. DOI: 10.1016/j.trd.2023.103695.

[142] Jinhua Ji, Yiming Bie, Ziling Zeng, and Linhong Wang, "Trip energy consumption estimation for electric buses," *Communications in Transportation Research*, vol. 2, p. 100 069, Dec. 2022. DOI: 10.1016/j.commtr.2022.100069.

[143] Maksymilian Madziel and Tiziana Campisi, "Energy consumption of electric vehicles: Analysis of selected parameters based on created database," *Energies*, vol. 16, no. 3, p. 1437, Feb. 2023. DOI: 10.3390/en16031437.

[144] Jin Zhang, Zhenpo Wang, Peng Liu, and Zhaosheng Zhang, "Energy consumption analysis and prediction of electric vehicles based on real-world driving data," *Applied Energy*, vol. 275, p. 115 408, Oct. 2020. DOI: 10.1016/j.apenergy.2020.115408.

[145] Zhaoyang Du, Celimuge Wu, Tsutomu Yoshinaga, Kok-Lim Alvin Yau, Yusheng Ji, and Jie Li, "Federated learning for vehicular internet of things: Recent advances and open issues," *IEEE Open Journal of the Computer Society*, vol. 1, pp. 45–61, 2020, ISSN: 2644-1268. DOI: 10.1109/ojcs.2020.2992630.

[146] Vishnu Pandi Chellapandi, Liangqi Yuan, Christopher G. Brinton, Stanislaw H. Żak, and Ziran Wang, "Federated learning for connected and automated vehicles: A survey of existing approaches and challenges," *IEEE Transactions*

on Intelligent Vehicles, vol. 9, no. 1, pp. 119–137, Jan. 2024, ISSN: 2379-8858. DOI: 10.1109/tiv.2023.3332675.

[147] Yuhan Zhou, "Federated learning merges with v2x privacy protection," *Journal of Computing and Electronic Information Management*, vol. 15, no. 2, pp. 52–54, Dec. 2024, ISSN: 2413-1660. DOI: 10.54097/f10t6523.

[148] Mehreen Tahir and Muhammad Intizar Ali, "On the performance of federated learning algorithms for iot," *IoT*, vol. 3, no. 2, pp. 273–284, Apr. 2022, ISSN: 2624-831X. DOI: 10.3390/iot3020016.

[149] Jiayan Liu, Gang Lin, Sunhua Huang, Yang Zhou, Christian Rehtanz, and Yong Li, "Collaborative EV routing and charging scheduling with power distribution and traffic networks interaction," *IEEE Transactions on Power Systems*, vol. 37, no. 5, pp. 3923–3936, Sep. 2022. DOI: 10.1109/tpwrs.2022.3142256.

[150] Jean-Baptiste Renard, Jeremy Surcin, Isabella Annesi-Maesano, Gilles Delaunay, Eric Poincelet, and Gilles Dixsaut, "Relation between pm2.5 pollution and covid-19 mortality in western europe for the 2020-2022 period," *Science of The Total Environment*, vol. 848, p. 157 579, Nov. 2022, ISSN: 0048-9697. DOI: 10.1016/j.scitotenv.2022.157579.

[151] Zhanglin Li, "An enhanced dual idw method for high-quality geospatial interpolation," *Scientific Reports*, vol. 11, no. 1, May 2021, ISSN: 2045-2322. DOI: 10.1038/s41598-021-89172-w.

[152] Ningrinla Marchang and Rakesh Tripathi, "Knn-st: Exploiting spatio-temporal correlation for missing data inference in environmental crowd sensing," *IEEE Sensors Journal*, vol. 21, no. 3, pp. 3429–3436, Feb. 2021, ISSN: 2379-9153. DOI: 10.1109/jsen.2020.3024976.

[153] Qingqing He, Weihang Wang, Yimeng Song, Ming Zhang, and Bo Huang, "Spatiotemporal high-resolution imputation modeling of aerosol optical depth for investigating its full-coverage variation in China from 2003 to 2020,"

Atmospheric Research, vol. 281, p. 106 481, Jan. 2023, ISSN: 0169-8095. DOI: 10.1016/j.atmosres.2022.106481.

[154] Gabriel Appleby, Linfeng Liu, and Li-Ping Liu, "Kriging convolutional networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 3187–3194, Apr. 2020, ISSN: 2159-5399. DOI: 10.1609/aaai.v34i04.5716.

[155] Chao Deng et al., "Joint urban modeling with graph convolutional networks and crowdsourced data: A novel approach," *IEEE Access*, vol. 12, pp. 57 796–57 805, 2024, ISSN: 2169-3536. DOI: 10.1109/access.2024.3390156.

[156] S Abirami and P Chitra, "Regional air quality forecasting using spatiotemporal deep learning," *Journal of Cleaner Production*, vol. 283, p. 125 341, Feb. 2021, ISSN: 0959-6526. DOI: 10.1016/j.jclepro.2020.125341.

[157] Cuiliu Yang and Zhao Pei, "Long-short term spatio-temporal aggregation for trajectory prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 4, pp. 4114–4126, Apr. 2023, ISSN: 1558-0016. DOI: 10.1109/tits.2023.3234962.

[158] Aji Gautama Putrada, Nur Alamsyah, Mohamad Nurkamal Fauzan, Ikke Dian Oktaviani, and Doan Perdana, "Gru for overcoming seasonality and trend in pm2.5 air pollution forecasting," in *2023 Eighth International Conference on Informatics and Computing (ICIC)*, IEEE, Dec. 2023, pp. 1–6. DOI: 10.1109/icic60109.2023.10381963.

[159] Javier Luengo-Oroz and Stefan Reis, "Assessment of cyclists' exposure to ultrafine particles along alternative commuting routes in edinburgh," *Atmospheric Pollution Research*, vol. 10, no. 4, pp. 1148–1158, Jul. 2019. DOI: 10.1016/j.apr.2019.01.020.

[160] Borut Jereb et al., "Exposure to black carbon during bicycle commuting–alternative route selection," *Atmosphere*, vol. 9, no. 1, p. 21, Jan. 2018. DOI: 10.3390/atmos9010021.

[161] K. Samal, Korra Babu, and Santos Das, "Predicting the least air polluted path using the neural network approach," *ICST Transactions on Scalable Information Systems*, vol. 8, no. 33, p. 170 250, Oct. 2021. DOI: 10.4108/eai.29-6-2021.170250.

[162] Giovanni Gualtieri, Lorenzo Brilli, Federico Carotenuto, Carolina Vagnoli, Alessandro Zaldei, and Beniamino Gioli, "Quantifying road traffic impact on air quality in urban areas: A covid19-induced lockdown analysis in Italy," *Environmental Pollution*, vol. 267, p. 115 682, Dec. 2020, ISSN: 0269-7491. DOI: 10.1016/j.envpol.2020.115682.

[163] S Veera Manikandan, Y Abilash, S Hari Prasanth, J Alfred Daniel, and R Santhosh, "Optimized feature selection for air quality index forecasting using GPR and SARIMA models," in *2023 International Conference on Inventive Computation Technologies (ICICT)*, IEEE, Apr. 2023. DOI: 10.1109/icict57646.2023.10134373.

[164] Wenwei Yue, Di Zhou, Shangbo Wang, and Peibo Duan, "Engineering traffic prediction with online data imputation: A graph-theoretic perspective," *IEEE Systems Journal*, vol. 17, no. 3, pp. 4485–4496, Sep. 2023, ISSN: 2373-7816. DOI: 10.1109/jsyst.2023.3268717.

[165] Ao Wang, Yongchao Ye, Xiaozhuang Song, Shiyao Zhang, and James J. Q. Yu, "Traffic prediction with missing data: A multi-task learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 4, pp. 4189–4202, Apr. 2023, ISSN: 1558-0016. DOI: 10.1109/tits.2022.3233890.

[166] Michael McQueen, John MacArthur, and Christopher Cherry, "The e-bike potential: Estimating regional e-bike impacts on greenhouse gas emissions," *Transportation Research Part D: Transport and Environment*, vol. 87, p. 102 482, Oct. 2020. DOI: 10.1016/j.trd.2020.102482.

[167] Roberto Nocerino, Alberto Colorni, Federico Lia, and Alessandro Luè, "E-bikes and e-scooters for smart logistics: Environmental and economic sustainability in pro-e-bike italian pilots," *Transportation Research Procedia*, vol. 14, pp. 2362–2371, 2016. DOI: 10.1016/j.trpro.2016.05.267.

[168] Henrik Becker, Milos Balac, Francesco Ciari, and Kay W. Axhausen, "Assessing the welfare impacts of shared mobility and mobility as a service (maas)," *Transportation Research Part A: Policy and Practice*, vol. 131, pp. 228–243, Jan. 2020, ISSN: 0965-8564. DOI: 10.1016/j.tra.2019.09.027.

[169] Zakir Hussain Farahmand, Konstantinos Gkiotsalitis, and Karst T. Geurs, "Mobility-as-a-service as a transport demand management tool: A case study among employees in the netherlands," *Case Studies on Transport Policy*, vol. 9, no. 4, pp. 1615–1629, Dec. 2021, ISSN: 2213-624X. DOI: 10.1016/j.cstp.2021.09.001.

[170] S. Cairns, F. Behrendt, D. Raffo, C. Beaumont, and C. Kiefer, "Electrically-assisted bikes: Potential impacts on travel behaviour," *Transportation Research Part A: Policy and Practice*, vol. 103, pp. 327–342, Sep. 2017. DOI: 10.1016/j.tra.2017.03.007.

[171] Yingqi Gu, Mingming Liu, Matheus Souza, and Robert N. Shorten, "On the design of an intelligent speed advisory system for cyclists," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Nov. 2018, pp. 3892–3897. DOI: 10.1109/itsc.2018.8569725.

[172] Maarten Kroesen, "To what extent do e-bikes substitute travel by other modes? evidence from the netherlands," *Transportation Research Part D: Transport and Environment*, vol. 53, pp. 377–387, Jun. 2017, ISSN: 1361-9209. DOI: 10.1016/j.trd.2017.04.036.

[173] Daozhi Zhao and Di Wang, "The research of tripartite collaborative governance on disorderly parking of shared bicycles based on the theory of planned behavior and motivation theories—a case of beijing, china," *Sus-*

tainability, vol. 11, no. 19, p. 5431, Sep. 2019, ISSN: 2071-1050. DOI: 10.3390/su11195431.

[174] Yixiao Li, Zhaoxin Dai, Lining Zhu, and Xiaoli Liu, "Analysis of spatial and temporal characteristics of citizens' mobility based on e-bike gps trajectory data in tengzhou city, china," *Sustainability*, vol. 11, no. 18, p. 5003, Sep. 2019, ISSN: 2071-1050. DOI: 10.3390/su11185003.

[175] Gamze Dane, Tao Feng, Floor Luub, and Theo Arentze, "Route choice decisions of e-bike users: Analysis of gps tracking data in the netherlands," in *Geospatial Technologies for Local and Regional Development*. Springer International Publishing, Apr. 2019, pp. 109–124, ISBN: 9783030147457. DOI: 10.1007/978-3-030-14745-7_7.

[176] Dandan Xu, Yang Bian, Jian Rong, Jiachuan Wang, and Baocai Yin, "Study on clustering of free-floating bike-sharing parking time series in beijing subway stations," *Sustainability*, vol. 11, no. 19, p. 5439, Sep. 2019, ISSN: 2071-1050. DOI: 10.3390/su11195439.

[177] Yang Liu, Ruo Jia, Xue Xie, and Zhiyuan Liu, "A two-stage destination prediction framework of shared bicycles based on geographical position recommendation," *IEEE Intelligent Transportation Systems Magazine*, vol. 11, no. 1, pp. 42–47, 2019, ISSN: 1941-1197. DOI: 10.1109/mits.2018.2884517.

[178] Fang Zong, Yongda Tian, Yanan He, Jinjun Tang, and Jianyu Lv, "Trip destination prediction based on multi-day gps data," *Physica A: Statistical Mechanics and its Applications*, vol. 515, pp. 258–269, Feb. 2019, ISSN: 0378-4371. DOI: 10.1016/j.physa.2018.09.090.

[179] Kanokporn Boonjubut and Hiroshi Hasegawa, "Accuracy of hourly demand forecasting of micro mobility for effective rebalancing strategies," *Management Systems in Production Engineering*, vol. 30, no. 3, pp. 246–252, Jul. 2022, ISSN: 2450-5781. DOI: 10.2478/mspe-2022-0031.

[180] Tzu-Chieh Tsai and Yuan Chen, "An iot based parking recommendation system considering distance and parking lot flow," in *2021 International Conference on Information and Communication Technology Convergence (ICTC)*, IEEE, Oct. 2021. DOI: 10.1109/ictc52510.2021.9620850.

[181] Waheb A. Jabbar, Chong Wen Wei, Nur Atiqah Ainaa M. Azmi, and Nur Aiman Haironnazli, "An iot raspberry pi-based parking management system for smart campus," *Internet of Things*, vol. 14, p. 100 387, Jun. 2021, ISSN: 2542-6605. DOI: 10.1016/j.iot.2021.100387.

[182] Dina Nawara and Rasha Kashef, "Iot-based recommendation systems - an overview," in *2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, IEEE, Sep. 2020. DOI: 10.1109/iemtronics51293.2020.9216391.

[183] Dina Nawara and Rasha Kashef, "Context-aware recommendation systems in the iot environment (iot-cars)-a comprehensive overview," *IEEE Access*, vol. 9, pp. 144 270–144 284, 2021, ISSN: 2169-3536. DOI: 10.1109/access.2021.3122098.

[184] Zeinab Teimoori, Abdulsalam Yassine, and M. Shamim Hossain, "A secure cloudlet-based charging station recommendation for electric vehicles empowered by federated learning," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 9, pp. 6464–6473, Sep. 2022, ISSN: 1941-0050. DOI: 10.1109/tii.2022.3148997.

[185] Jonathan P. Epperlein, Julien Monteil, Mingming Liu, Yingqi Gu, Sergiy Zhuk, and Robert Shorten, "Bayesian classifier for route prediction with markov chains," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Nov. 2018. DOI: 10.1109/itsc.2018.8569895.

[186] Hao Chen, Wenxian Wang, Ligang Cheng, and Ping Li, "A cooperative optimization method for the layout of shared bicycle parking areas and delivery

quantity," *Scientific Reports*, vol. 14, no. 1, Feb. 2024, ISSN: 2045-2322. DOI: 10.1038/s41598-024-54647-z.

[187] Duan Su, Yacan Wang, Nan Yang, and Xianghong Wang, "Promoting considerate parking behavior in dockless bike-sharing: An experimental study," *Transportation Research Part A: Policy and Practice*, vol. 140, pp. 153–165, Oct. 2020, ISSN: 0965-8564. DOI: 10.1016/j.tra.2020.08.006.

[188] Anne Brown, Nicholas J. Klein, Calvin Thigpen, and Nicholas Williams, "Impeding access: The frequency and characteristics of improper scooter, bike, and car parking," *Transportation Research Interdisciplinary Perspectives*, vol. 4, p. 100 099, Mar. 2020. DOI: 10.1016/j.trip.2020.100099.

[189] Eva Heinen and Ralph Buehler, "Bicycle parking: A systematic review of scientific literature on parking behaviour, parking preferences, and their influence on cycling and travel behaviour," *Transport Reviews*, vol. 39, no. 5, pp. 630–656, Mar. 2019. DOI: 10.1080/01441647.2019.1590477.

[190] Liangpeng Gao, Yanjie Ji, Xingchen Yan, Yao Fan, and Weihong Guo, "Incentive measures to avoid the illegal parking of dockless shared bikes: The relationships among incentive forms, intensity and policy compliance," *Transportation*, vol. 48, no. 2, pp. 1033–1060, Feb. 2020. DOI: 10.1007/s11116-020-10088-x.

[191] Ruoying Chen, ""bike litter" and obligations of the platform operators: Lessons from china's dockless sharing bikes," *Computer Law & Security Review*, vol. 35, no. 5, p. 105 317, 2019. DOI: 10.1016/j.clsr.2019.03.011.

[192] Sunethra B, Sreeya C, Dhannushree U, Nagaraj P, Muthamil Sudar K, and Muneeswaran V, "A systematic parking system using bi-class machine learning techniques," in *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, IEEE, Apr. 2022. DOI: 10.1109/icscds53736.2022.9760903.

[193] Alberto Rossi, Gianni Barlacchi, Monica Bianchini, and Bruno Lepri, "Modelling taxi drivers' behaviour for the next destination prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 7, pp. 2980–2989, Jul. 2020. DOI: 10.1109/tits.2019.2922002.

[194] Jianming Lv, Qinghui Sun, Qing Li, and Luis Moreira-Matias, "Multi-scale and multi-scope convolutional neural networks for destination prediction of trajectories," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 8, pp. 3184–3195, Aug. 2020. DOI: 10.1109/tits.2019.2924903.

[195] Jinjun Tang, Jian Liang, Tianjian Yu, Yong Xiong, and Guoliang Zeng, "Trip destination prediction based on a deep integration network by fusing multiple features from taxi trajectories," *IET Intelligent Transport Systems*, vol. 15, no. 9, pp. 1131–1141, Jun. 2021. DOI: 10.1049/itr2.12075.

[196] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002. DOI: 10.1613/jair.953.

[197] Richard W. Bohannon and A. Williams Andrews, "Normal walking speed: A descriptive meta-analysis," *Physiotherapy*, vol. 97, no. 3, pp. 182–189, Sep. 2011. DOI: 10.1016/j.physio.2010.12.004.

[198] Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6765–6816, Jan. 2017, ISSN: 1532-4435.

[199] Ernani F. Choma, John S. Evans, James K. Hammitt, José A. Gómez-Ibáñez, and John D. Spengler, "Assessing the health impacts of electric vehicles through air pollution in the united states," *Environment International*, vol. 144, p. 106 015, Nov. 2020. DOI: 10.1016/j.envint.2020.106015.

[200] Şükrü İmre, Dilay Çelebi, and Funda Koca, "Understanding barriers and enablers of electric vehicles in urban freight transport: Addressing stakeholder

needs in turkey," *Sustainable Cities and Society*, vol. 68, p. 102 794, May 2021. DOI: 10.1016/j.scs.2021.102794.

[201] Jianwei Xing, Benjamin Leard, and Shanjun Li, "What does an electric vehicle replace?" *Journal of Environmental Economics and Management*, vol. 107, p. 102 432, May 2021. DOI: 10.1016/j.jeem.2021.102432.

[202] Akhtar Hussain, Van-Hai Bui, and Hak-Man Kim, "Optimal sizing of battery energy storage system in a fast EV charging station considering power outages," *IEEE Transactions on Transportation Electrification*, vol. 6, no. 2, pp. 453–463, Jun. 2020. DOI: 10.1109/tte.2020.2980744.

[203] Yanyan Ding, Xinwei Li, and Sisi Jian, "Modeling the impact of vehicle-to-grid discharge technology on transport and power systems," *Transportation Research Part D: Transport and Environment*, vol. 105, p. 103 220, Apr. 2022. DOI: 10.1016/j.trd.2022.103220.

[204] Anil K. Madhusudhanan and Xiaoxiang Na, "Effect of a traffic speed based cruise control on an electric vehicle s performance and an energy consumption model of an electric vehicle," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 386–394, Mar. 2020. DOI: 10.1109/jas.2020.1003030.

[205] Shemin Sagaria, Rui Costa Neto, and Patricia Baptista, "Modelling approach for assessing influential factors for EV energy performance," *Sustainable Energy Technologies and Assessments*, vol. 44, p. 100 984, Apr. 2021. DOI: 10.1016/j.seta.2020.100984.

[206] Yue Xiang, Jianping Yang, Xuecheng Li, Chenghong Gu, and Shuai Zhang, "Routing optimization of electric vehicles for charging with event-driven pricing strategy," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 1, pp. 7–20, Jan. 2022. DOI: 10.1109/tase.2021.3102997.

[207] Yuche Chen, Guoyuan Wu, Ruixiao Sun, Abhishek Dubey, Aron Laszka, and Philip Pugliese, "A review and outlook on energy consumption estimation models for electric vehicles," *SAE International Journal of Sustainable*

Transportation, Energy, Environment, & Policy, vol. 2, no. 1, Mar. 2021. DOI: 10.4271/13-02-01-0005.

[208] Weizheng Wang et al., "Secure-enhanced federated learning for AI-empowered electric vehicle energy prediction," *IEEE Consumer Electronics Magazine*, vol. 12, no. 2, pp. 27–34, Mar. 2023. DOI: 10.1109/mce.2021.3116917.

[209] Peiling Chen, Yujian Ye, Jianxiong Hu, Hongru Wang, Yonggao Yin, and Yi Tang, "Dynamic modeling of smart buildings energy consumption: A cyber-physical fusion approach," in *2021 IEEE Sustainable Power and Energy Conference (iSPEC)*, IEEE, Dec. 2021. DOI: 10.1109/ispec53008.2021.9735981.

[210] Muhammad Zahid, Yangzhou Chen, Arshad Jamal, Khalaf A. Al-Ofi, and Hassan M. Al-Ahmadi, "Adopting machine learning and spatial analysis techniques for driver risk assessment: Insights from a case study," *International Journal of Environmental Research and Public Health*, vol. 17, no. 14, p. 5193, Jul. 2020. DOI: 10.3390/ijerph17145193.

[211] Muhammad Zahid, Yangzhou Chen, Arshad Jamal, and Muhammad Qasim Memon, "Short term traffic state prediction via hyperparameter optimization based classifiers," *Sensors*, vol. 20, no. 3, p. 685, Jan. 2020. DOI: 10.3390/s20030685.

[212] Muhammad Zahid, Yangzhou Chen, Sikandar Khan, Arshad Jamal, Muhammad Ijaz, and Tufail Ahmed, "Predicting risky and aggressive driving behavior among taxi drivers: Do spatio-temporal attributes matter?" *International Journal of Environmental Research and Public Health*, vol. 17, no. 11, p. 3937, Jun. 2020. DOI: 10.3390/ijerph17113937.

[213] Masuod Bayat and Mohammad Mahdi Abootorabi, "Comparison of minimum quantity lubrication and wet milling based on energy consumption modeling," *Proceedings of the Institution of Mechanical Engineers, Part E:*

Journal of Process Mechanical Engineering, vol. 235, no. 5, pp. 1665–1675, May 2021. DOI: 10.1177/09544089211014407.

[214] Mohamed I. Ibrahem, Mohamed Mahmoud, Mostafa M. Fouda, Basem M. ElHalawany, and Waleed Alasmary, "Privacy-preserving and efficient decentralized federated learning-based energy theft detector," in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, IEEE, Dec. 2022. DOI: 10.1109/globecom48099.2022.10000881.

[215] Muhammad Firdaus, Harashta Tatimma Larasati, and Kyung-Hyune Rhee, "A secure federated learning framework using blockchain and differential privacy," in *2022 IEEE 9th International Conference on Cyber Security and Cloud Computing (CSCloud)/2022 IEEE 8th International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, IEEE, Jun. 2022. DOI: 10.1109/cscloud-edgecom54986.2022.00013.

[216] Mahdi Barhoush, Ahmad Ayad, and Anke Schmeink, "Accelerating federated learning via modified local model update based on individual performance metric," in *2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, IEEE, Jul. 2023. DOI: 10.1109/iceccme57830.2023.10253070.

[217] Godwin Badu-Marfo, Bilal Farooq, and Zachary Patterson, "A perspective on the challenges and opportunities for privacy-aware big transportation data," *Journal of Big Data Analytics in Transportation*, vol. 1, no. 1, pp. 1–23, Apr. 2019. DOI: 10.1007/s42421-019-00001-z.

[218] Pablo López-Aguilar, Edgar Batista, Antoni Martínez-Ballesté, and Agusti Solanas, "Information security and privacy in railway transportation: A systematic review," *Sensors*, vol. 22, no. 20, p. 7698, Oct. 2022. DOI: 10.3390/s22207698.

[219] Hadi Habibzadeh, Brian H. Nussbaum, Fazel Anjomshoa, Burak Kantarci, and Tolga Soyata, "A survey on cybersecurity, data privacy, and policy issues

in cyber-physical system deployments in smart cities," *Sustainable Cities and Society*, vol. 50, p. 101 660, Oct. 2019. DOI: 10.1016/j.scs.2019.101660.

[220] Dalton Hahn, Arslan Munir, and Vahid Behzadan, "Security and privacy issues in intelligent transportation systems: Classification and challenges," *IEEE Intelligent Transportation Systems Magazine*, vol. 13, no. 1, pp. 181–196, 2021. DOI: 10.1109/mits.2019.2898973.

[221] Xinkai Wu, David Freese, Alfredo Cabrera, and William A. Kitch, "Electric vehicles' energy consumption measurement and estimation," *Transportation Research Part D: Transport and Environment*, vol. 34, pp. 52–67, Jan. 2015. DOI: 10.1016/j.trd.2014.10.007.

[222] Chiara Fiori, Kyoungho Ahn, and Hesham A. Rakha, "Power-based electric vehicle energy consumption model: Model development and validation," *Applied Energy*, vol. 168, pp. 257–268, Apr. 2016. DOI: 10.1016/j.apenergy.2016.01.097.

[223] Rina Ristiana, Arief Syaichu Rohman, Carmadi Machbub, Agus Purwadi, and Estiko Rijanto, "A new approach of EV modeling and its control applications to reduce energy consumption," *IEEE Access*, vol. 7, pp. 141 209–141 225, 2019. DOI: 10.1109/access.2019.2941001.

[224] Ilyès Miri, Abbas Fotouhi, and Nathan Ewin, "Electric vehicle energy consumption modelling and estimation—a case study," *International Journal of Energy Research*, vol. 45, no. 1, pp. 501–520, Jul. 2020. DOI: 10.1002/er.5700.

[225] Yang Zhao, Zhenpo Wang, Zuo-Jun Max Shen, and Fengchun Sun, "Assessment of battery utilization and energy consumption in the large-scale development of urban electric vehicles," *Proceedings of the National Academy of Sciences*, vol. 118, no. 17, Apr. 2021. DOI: 10.1073/pnas.2017318118.

[226] Sagar Wankhede, Prajwal Thorat, Sanket Shisode, Swapnil Sonawane, and Rugved Wankhade, "Energy consumption estimation for electric two-wheeler

using different drive cycles for achieving optimum efficiency," *Energy Storage*, vol. 4, no. 6, May 2022. DOI: 10.1002/est2.361.

[227] Mahesh Kolte and Tanmay Khachane, "Electric vehicle range and energy consumption estimation using MATLAB simulink," in *INTERNATIONAL CONFERENCE ON TRENDS IN CHEMICAL ENGINEERING 2021 (ICoTRiCE2021)*, AIP Publishing, 2022. DOI: 10.1063/5.0114145.

[228] Joonki Hong, Sangjun Park, and Naehyuck Chang, "Accurate remaining range estimation for electric vehicles," in *2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)*, IEEE, Jan. 2016. DOI: 10.1109/aspdac.2016.7428106.

[229] Jun Chen, Man Liang, and Xu Ma, "Probabilistic analysis of electric vehicle energy consumption using MPC speed control and nonlinear battery model," in *2021 IEEE Green Technologies Conference (GreenTech)*, IEEE, Apr. 2021. DOI: 10.1109/greentech48523.2021.00038.

[230] Ahmad Almaghrebi, Fares Al Juheshi, Jarod Nekl, Kevin James, and Mahmoud Alahmad, "Analysis of energy consumption at public charging stations, a nebraska case study," in *2020 IEEE Transportation Electrification Conference & Expo (ITEC)*, IEEE, Jun. 2020. DOI: 10.1109/itec48692.2020.9161456.

[231] Fuguo Xu and Tielong Shen, "Look-ahead prediction-based real-time optimal energy management for connected HEVs," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 2537–2551, Mar. 2020. DOI: 10.1109/tvt.2020.2965163.

[232] Rui Zhang and Enjian Yao, "Electric vehicles' energy consumption estimation with real driving condition data," *Transportation Research Part D: Transport and Environment*, vol. 41, pp. 177–187, Dec. 2015. DOI: 10.1016/j.trd.2015.10.010.

[233] Cedric De Cauwer, Joeri Van Mierlo, and Thierry Coosemans, "Energy consumption prediction for electric vehicles based on real-world data," *Energies*, vol. 8, no. 8, pp. 8573–8593, Aug. 2015. DOI: 10.3390/en8088573.

[234] Jianhua Guo, Yu Jiang, Yuanbin Yu, and Weilun Liu, "A novel energy consumption prediction model with combination of road information and driving style of BEVs," *Sustainable Energy Technologies and Assessments*, vol. 42, p. 100 826, Dec. 2020. DOI: 10.1016/j.seta.2020.100826.

[235] Viraaji Mothukuri, Reza M. Parizi, Seyedamin Pouriyeh, Yan Huang, Ali Dehghantanha, and Gautam Srivastava, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021. DOI: 10.1016/j.future.2020.10.007.

[236] Helio N. Cunha Neto, Ivana Dusparic, Diogo M. F. Mattos, and Natalia C. Fernande, "Fedsa: Accelerating intrusion detection in collaborative environments with federated simulated annealing," in *2022 IEEE 8th International Conference on Network Softwarization (NetSoft)*, IEEE, Jun. 2022, pp. 420–428. DOI: 10.1109/netsoft54395.2022.9844024.

[237] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas, *Communication-efficient learning of deep networks from decentralized data*, 2016. DOI: 10.48550/ARXIV.1602.05629.

[238] Beibei Li, Yuqing Guo, Qingyun Du, Ziqing Zhu, Xiaohui Li, and Rongxing Lu, "P^3: Privacy-preserving prediction of real-time energy demands in EV charging networks," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 3, pp. 3029–3038, Mar. 2023. DOI: 10.1109/tii.2022.3182972.

[239] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith, *Federated optimization in heterogeneous networks*, 2018. DOI: 10.48550/ARXIV.1812.06127.

[240] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary, *Federated learning with personalization layers*, 2019. DOI: 10.48550/ARXIV.1912.00818.

[241] Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai, "Exploiting shared representations for personalized federated learning," in *Proceedings of the 38th International Conference on Machine Learning*, Marina Meila and Tong Zhang, Eds., ser. Proceedings of Machine Learning Research, vol. 139, PMLR, Jul. 2021, pp. 2089–2099.

[242] Wei Yang Bryan Lim et al., "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020. DOI: 10.1109/comst.2020.2986024.

[243] Mamoun Alazab, Swarna Priya RM, Parimala M, Praveen Kumar Reddy Maddikunta, Thippa Reddy Gadekallu, and Quoc-Viet Pham, "Federated learning for cybersecurity: Concepts, challenges, and future directions," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3501–3509, May 2022. DOI: 10.1109/tii.2021.3119038.

[244] Ji Li, Kailong Liu, Quan Zhou, Jinhao Meng, Yunshan Ge, and Hongming Xu, "Electrothermal dynamics-conscious many-objective modular design for power-split plug-in hybrid electric vehicles," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 4406–4416, Dec. 2022. DOI: 10.1109/tmech.2022.3156535.

[245] Geunseob Oh, David J. Leblanc, and Huei Peng, "Vehicle energy dataset (VED), a large-scale dataset for vehicle energy consumption research," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 4, pp. 3302–3312, Apr. 2022. DOI: 10.1109/tits.2020.3035596.

[246] Sen Yan, David J. O'Connor, Xiaojun Wang, Noel E. O'Connor, Alan. F. Smeaton, and Mingming Liu, "Comparative analysis of machine learning-based imputation techniques for air quality datasets with high missing data

rates," in *2025 IEEE Symposia on Computational Intelligence for Energy, Transport and Environmental Sustainability (CIETES)*, IEEE, Mar. 2025, pp. 1–8. DOI: 10.1109/cietes63869.2025.10995064.

[247] Robert J. Henning, "Particulate matter air pollution is a significant risk factor for cardiovascular disease," *Current Problems in Cardiology*, vol. 49, no. 1, p. 102 094, Jan. 2024, ISSN: 0146-2806. DOI: 10.1016/j.cpcardiol.2023.102094.

[248] Jean-Baptiste Renard, Jérémy Surcin, Isabella Annesi-Maesano, and Eric Poincelet, "Temporal evolution of pm2.5 levels and covid-19 mortality in europe for the 2020-2022 period," *Atmosphere*, vol. 14, no. 8, p. 1222, Jul. 2023, ISSN: 2073-4433. DOI: 10.3390/atmos14081222.

[249] Edoardo Arnaudo, Alessandro Farasin, and Claudio Rossi, "A comparative analysis for air quality estimation from traffic and meteorological data," *Applied Sciences*, vol. 10, no. 13, p. 4587, Jul. 2020, ISSN: 2076-3417. DOI: 10.3390/app10134587.

[250] Ihsane Gryech, Mounir Ghogho, Hajar Elhammouti, Nada Sbihi, and Abdellatif Kobbane, "Machine learning for air quality prediction using meteorological and traffic related features," *Journal of Ambient Intelligence and Smart Environments*, vol. 12, no. 5, pp. 379–391, Sep. 2020, ISSN: 1876-1364. DOI: 10.3233/ais-200572.

[251] Sheikh Rahmatulla Sakib et al., "Time series analysis and forecasting of air quality index of Dhaka City of Bangladesh," in *2023 IEEE World AI IoT Congress (AIIoT)*, IEEE, Jun. 2023. DOI: 10.1109/aiiot58121.2023.10174539.

[252] Abdellatif Bekkar, Badr Hssina, Samira Douzi, and Khadija Douzi, "Air-pollution prediction in smart city, deep learning approach," *Journal of Big Data*, vol. 8, no. 1, Dec. 2021, ISSN: 2196-1115. DOI: 10.1186/s40537-021-00548-1.

[253] R. Janarthanan, P. Partheeban, K. Somasundaram, and P. Navin Elamparithi, "A deep learning approach for prediction of air quality index in a metropolitan city," *Sustainable Cities and Society*, vol. 67, p. 102 720, Apr. 2021, ISSN: 2210-6707. DOI: 10.1016/j.scs.2021.102720.

[254] Wenjing Mao, Weilin Wang, Limin Jiao, Suli Zhao, and Anbao Liu, "Modeling air quality prediction using a deep learning approach: Method optimization and evaluation," *Sustainable Cities and Society*, vol. 65, p. 102 567, Feb. 2021, ISSN: 2210-6707. DOI: 10.1016/j.scs.2020.102567.

[255] Xinyu Chen, Mengying Lei, Nicolas Saunier, and Lijun Sun, "Low-rank autoregressive tensor completion for spatiotemporal traffic data imputation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 12 301–12 310, Aug. 2022, ISSN: 1558-0016. DOI: 10.1109/tits.2021.3113608.

[256] Mingzhe Liu, Han Huang, Hao Feng, Leilei Sun, Bowen Du, and Yanjie Fu, "Pristi: A conditional diffusion framework for spatiotemporal imputation," in *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, IEEE, Apr. 2023, pp. 1927–1939. DOI: 10.1109/icde55515.2023.00150.

[257] Muhammad Saad, Mohita Chaudhary, Fakhri Karray, and Vincent Gaudet, "Machine learning based approaches for imputation in time series data and their impact on forecasting," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, Oct. 2020. DOI: 10.1109/smc42975.2020.9283191.

[258] Baowei Wang, Weiwen Kong, and Peng Zhao, "An air quality forecasting model based on improved convnet and RNN," *Soft Computing*, vol. 25, no. 14, pp. 9209–9218, May 2021, ISSN: 1433-7479. DOI: 10.1007/s00500-021-05843-w.

[259] Ghufran Isam Drewil and Riyadh Jabbar Al-Bahadili, "Air pollution prediction using lstm deep learning and metaheuristics algorithms," *Measurement:*

Sensors, vol. 24, p. 100 546, Dec. 2022, ISSN: 2665-9174. DOI: 10.1016/j.measen.2022.100546.

[260] Hyun Ahn, Kyunghee Sun, and Kwanghoon Pio Kim, "Comparison of missing data imputation methods in time series forecasting," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 767–779, 2022, ISSN: 1546-2226. DOI: 10.32604/cmc.2022.019369.

[261] Aleksandar Sekulić, Milan Kilibarda, Gerard B.M. Heuvelink, Mladen Nikolić, and Branislav Bajat, "Random forest spatial interpolation," *Remote Sensing*, vol. 12, no. 10, p. 1687, May 2020, ISSN: 2072-4292. DOI: 10.3390/rs12101687.

[262] Minglong Lei, Pei Quan, Rongrong Ma, Yong Shi, and Lingfeng Niu, "Diggcn: Learning compact graph convolutional networks via diffusion aggregation," *IEEE Transactions on Cybernetics*, vol. 52, no. 2, pp. 912–924, Feb. 2022, ISSN: 2168-2275. DOI: 10.1109/tcyb.2020.2988791.

[263] Fareena Naz et al., "Comparative analysis of deep learning and statistical models for air pollutants prediction in urban areas," *IEEE Access*, vol. 11, pp. 64 016–64 025, 2023, ISSN: 2169-3536. DOI: 10.1109/access.2023.3289153.

[264] Vishesh Mittal, Suhas Sasetty, Rashmi Choudhary, and Amit Agarwal, "Deep-learning spatiotemporal prediction framework for particulate matter under dynamic monitoring," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2676, no. 8, pp. 56–73, Mar. 2022, ISSN: 2169-4052. DOI: 10.1177/03611981221082589.

[265] Junwei Xiao, Qingfang Wang, Jiawang Cui, and Jing Yu, "Multi-feature pm2.5 prediction with arima-lstm," in *2022 International Conference on Machine Learning, Cloud Computing and Intelligent Mining (MLCCIM)*, IEEE, Aug. 2022, pp. 229–233. DOI: 10.1109/mlccim55934.2022.00046.

[266] Puli Dilliswar Reddy and L. Rama Parvathy, "Prediction analysis using random forest algorithms to forecast the air pollution level in a particular location," in *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*, IEEE, Oct. 2022, pp. 1585–1589. DOI: 10.1109/icosec54921.2022.9952138.

[267] Jannah Mohammad and Mohammod Abul Kashem, "Air pollution comparison rfm model using machine learning approach," in *2022 IEEE 7th International conference for Convergence in Technology (I2CT)*, IEEE, Apr. 2022, pp. 1–5. DOI: 10.1109/i2ct54291.2022.9824248.

[268] Tongbin Liu, Yong Wang, Hongyu Zhou, Jian Luo, and Fangming Deng, "Distributed short-term traffic flow prediction based on integrating federated learning and tcn," *IEEE Access*, vol. 12, pp. 148 026–148 036, 2024, ISSN: 2169-3536. DOI: 10.1109/access.2024.3474300.

[269] Tafia Hasna Putri et al., "Fine-tuning of predictive models cnn-lstm and conv-lstm for nowcasting pm2.5 level," *IEEE Access*, vol. 12, pp. 28 988–29 003, 2024, ISSN: 2169-3536. DOI: 10.1109/access.2024.3368034.

[270] K. Krishna Rani Samal, Korra Sathya Babu, and Santos Kumar Das, "ORS: The optimal routing solution for smart city users," in *Lecture Notes in Electrical Engineering*, Springer Singapore, 2020, pp. 177–186. DOI: 10.1007/978-981-15-7031-5_17.

[271] Ji Luo, Kanok Boriboonsomsin, and Matthew Barth, "Consideration of exposure to traffic-related air pollution in bicycle route planning," *Journal of Transport & Health*, vol. 16, p. 100 792, Mar. 2020, ISSN: 2214-1405. DOI: 10.1016/j.jth.2019.100792.

[272] Yuxin Wang, Yizheng Wu, Zhenyu Li, Kai Liao, Chao Li, and Guohua Song, "Route planning for active travel considering air pollution exposure," *Transportation Research Part D: Transport and Environment*, vol. 103, p. 103 176, Feb. 2022, ISSN: 1361-9209. DOI: 10.1016/j.trd.2022.103176.

[273] Elias Willberg, Age Poom, Joose Helle, and Tuuli Toivonen, "Cyclists' exposure to air pollution, noise, and greenery: A population-level spatial analysis approach," *International Journal of Health Geographics*, vol. 22, no. 1, Feb. 2023, ISSN: 1476-072X. DOI: 10.1186/s12942-023-00326-7.

[274] Elham Mortazavi Moghaddam, Gholamreza Shiran, Ahmad Reza Jafarian-Moghaddam, and Ali Naaman, "Cyclists' exposure to traffic-generated air pollution in multi-modal transportation network design problem," *PLOS ONE*, vol. 18, no. 6, Sathishkumar V. E., Ed., e0286153, Jun. 2023, ISSN: 1932-6203. DOI: 10.1371/journal.pone.0286153.

[275] Heike Marquart, "Informing about the invisible: Communicating en route air pollution and noise exposure to cyclists and pedestrians using focus groups," *European Transport Research Review*, vol. 14, no. 1, Nov. 2022, ISSN: 1866-8887. DOI: 10.1186/s12544-022-00571-0.

[276] Heike Marquart, Kerstin Stark, and Julia Jarass, "How are air pollution and noise perceived en route? investigating cyclists' and pedestrians' personal exposure, wellbeing and practices during commute," *Journal of Transport & Health*, vol. 24, p. 101 325, Mar. 2022, ISSN: 2214-1405. DOI: 10.1016/j.jth.2021.101325.

[277] Joost Wesseling et al., "Assessment of pm2.5 exposure during cycle trips in the netherlands using low-cost sensors," *International Journal of Environmental Research and Public Health*, vol. 18, no. 11, p. 6007, Jun. 2021, ISSN: 1660-4601. DOI: 10.3390/ijerph18116007.

[278] Richard Sinnott and Siqi Zhong, "Real-time route planning to reduce pedestrian pollution exposure in urban settings," in *Proceedings of the IEEE/ACM 10th International Conference on Big Data Computing, Applications and Technologies*, ser. BDCAT '23, ACM, Dec. 2023, pp. 1–10. DOI: 10.1145/3632366.3632381.

[279] Miguel Grinberg, *Flask web development*. "O'Reilly Media, Inc.", 2018.

[280] Jean-Baptiste Renard, Eric Poincelet, Isabella Annesi-Maesano, and Jérémy Surcin, "Spatial distribution of pm2.5 mass and number concentrations in paris (france) from the pollutrack network of mobile sensors during 2018-2022," *Sensors*, vol. 23, no. 20, p. 8560, Oct. 2023, ISSN: 1424-8220. DOI: 10.3390/s23208560.

[281] Jean-Baptiste Renard and Christophe Marchand, "High resolution mapping of pm2.5 concentrations in paris (france) using mobile pollutrack sensors network in 2020," *Atmosphere*, vol. 12, no. 5, p. 529, Apr. 2021, ISSN: 2073-4433. DOI: 10.3390/atmos12050529.

[282] Tomasz Markowski and Piotr Bilski, "Optimization of autonomous agent routes in logistics warehouse," *International Journal of Electronics and Telecommunications*, pp. 559–564, Jul. 2021, ISSN: 2081-8491. DOI: 10.24425/ijet.2021.137846.

[283] Jeonghyeon Pak, Jeongeun Kim, Yonghyun Park, and Hyoung Il Son, "Field evaluation of path-planning algorithms for autonomous mobile robot in smart farms," *IEEE Access*, vol. 10, pp. 60 253–60 266, 2022, ISSN: 2169-3536. DOI: 10.1109/access.2022.3181131.

[284] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein, *Introduction to algorithms*. MIT press, 2022.

[285] Rafal Szczepanski and Tomasz Tarczewski, "Global path planning for mobile robot based on artificial bee colony and dijkstra's algorithms," in *2021 IEEE 19th International Power Electronics and Motion Control Conference (PEMC)*, IEEE, Apr. 2021, pp. 724–730. DOI: 10.1109/pemc48073.2021.9432570.