ABSTRACT

Title of Dissertation: INTRODUCING A GRAPH-BASED NEURAL NETWORK FOR NETWORKWIDE TRAFFIC VOLUME ESTIMATION

Sara Zahedian

Dissertation directed by: Professor Ali Haghani, Department of Civil and Environmental Engineering

Traffic volumes are an essential input to many highway planning and design models; however, collecting this data for all the roads in a network is not practical nor cost-effective. Accordingly, transportation agencies must find ways to leverage limited ground truth count data to obtain reasonable estimates at scale on all the network segments. One of the challenges that complicate this estimation is the complex spatial dependency of the links' traffic state in a transportation network. A graph-based model is proposed to estimate networkwide traffic volumes to address this challenge. This model aims to consider the graph structure of the network to extract its spatial correlations while estimating link volumes. In the first step, a proof-of-concept methodology is presented to indicate how adding the simple spatial correlation between the links in the Euclidian space improves the performance of a state-of-the-art volume estimation model. This methodology is applied to the New Hampshire road network to estimate statewide hourly traffic volumes. In the next step, a Graph Neural Network model is introduced to consider the complex interdependency of the road network in a non-Euclidean domain. This model is called Fine-tuned Spatio-Temporal Graph Neural Network (FSTGCN) and applied to various Maryland State networks to estimate 15-minute traffic volumes. The results illustrate significant improvement over the existing state-of-the-art models used for networkwide traffic volume estimation, namely ANN and XGBoost.

INTRODUCING A GRAPH-BASED NEURAL NETWORK FOR NETWORKWIDE TRAFFIC VOLUME ESTIMATION

by

Sara Zahedian

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2021

Advisory Committee:

Professor Ali Haghani, Chair

Professor Martin Dresner, Dean's Representative

Professor Paul Schonfeld

Professor Cinzia Cirillo

Dr. Kaveh Farokhi Sadabadi

© Copyright by

Sara Zahedian

2021

Acknowledgments

I would first like to express my deepest gratitude to my supervisor, Professor Ali Haghani, who made this work possible. His continuous inspiration, encouragement, and guidance pushed me to sharpen my thinking and brought my work to a higher level. My special thanks also go to my doctoral examination committee members, Professor Paul Schonfeld, Professor Martin Dresner, Cinzia Cirillo, and Dr. Kaveh Farokhi Sadababi, for their constructive comments and suggestions in this work.

I want to acknowledge my colleagues from my research lab, The Center for Advanced Transportation Technology at the University of Maryland, Dr. Przemysław Sekuła, Zachary Vander Laan, Amir Nohekhan, and thank them for their technical and emotional supports.

Finally, special thanks go to my mother and my family for their unconditional support and understanding during my years of doctoral study. Without their persistent encouragement, I would not have been able to make it through those difficult times.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Motivation

Traffic volumes are an essential component for computing various traffic performance measurements on a road network – including those used for the performance-based planning and programming process under the Moving Ahead for Progress in the 21st Century Act (MAP-21). While this data is needed at the statewide road network level for such purposes, large-scale networkwide volume data collection is infeasible. Transportation agencies often spend a significant portion of their budget collecting traffic count data (Zhong et al., 2004); however, continuous recording of traffic volume data is limited to a small percentage of road segments where continuous count stations (CCS) are installed (Wang & Kockelman, 2009). Consequently, agencies must determine how to best obtain statewide link-level volume estimates given the limited locations where reliable ground truth count data can be collected – a topic that has produced several proposed approaches. Most of these methods use the data collected by CCS stations to estimate the link-level hourly volume. However, the temporal and spatial correlation between link volumes is often overlooked due to its inherent complexity. The primary motivation behind this dissertation is to incorporate the spatio-temporal relationships between traffic volume in different segments of the road network in the traffic volume estimation framework.

In the last decade, the revolutionary advancement achieved in the data analysis area has created the opportunity to recognize these types of complicated patterns (i.e., temporal and spatial correlations in a road network). In particular, the massively available traffic data, on the one hand, and the ever-growing analytical methods, on the other hand, may help transportation experts to estimate traffic measurements more accurately. Therefore, introducing a method that leverages these advancements to solve the volume estimation problem is of great importance as it can improve the accuracy and reduce the cost of collecting traffic counts data.

While the volume data is not available for most links in a road network, link-level speed records are directly computable using probe vehicle data. The speed profiles of links are a valuable source of data that can also be used to estimate the networkwide traffic volume. A few studies have utilized advanced machine learning methods like deep learning to estimate statewide hourly traffic volume in the last couple of years. One of the state-of-the-art methodologies introduced by Sekula et al. (2018) uses a deep learning regression approach to estimate hourly traffic volume using the following data sources:

- Vehicle probe speed
- Vehicle probe counts
- Weather stations
- Road characteristics

This approach yielded appreciably higher estimation accuracy than other existing methods. However, it overlooks incorporating the road network's underlying characteristics and geometry to capture the Spatio-temporal correlation between the road segments. The proof-of-concept artificial neural network-based model introduced

in this study, although it solves the problem in the Euclidian space, which ignores the complex configuration of the transportation network, presents improved estimation accuracy relative to Sekula et al. (2018). Therefore, the incorporation of link traffic flow dependencies in the modeling framework can improve the current traffic volume estimation models.

1.2 Scope of the Dissertation

The primary purpose of this study is to improve the networkwide traffic volume estimation using a representative graph of the road network. Considering the underlying characteristics of the road segments in the form of a graph is a crucial step toward extracting their spatio-temporal correlations. The link traffic volumes are available for a handful of this graph's links (i.e., where CCS are deployed). This study aims to directly use these CCS data to estimate hourly traffic volume for all other graph links. Figure 1 presents the high-level architecture of the proposed graph-based model, while Figure 2 shows the architecture of the existing state-of-the-art ANN model (Sekula et al., 2018).

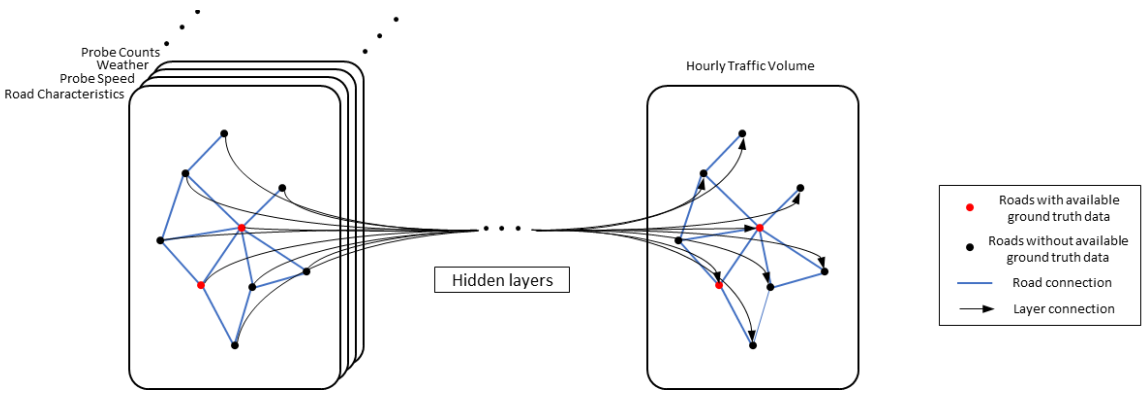


Figure 1. High-level architecture of the proposed graph-based model.

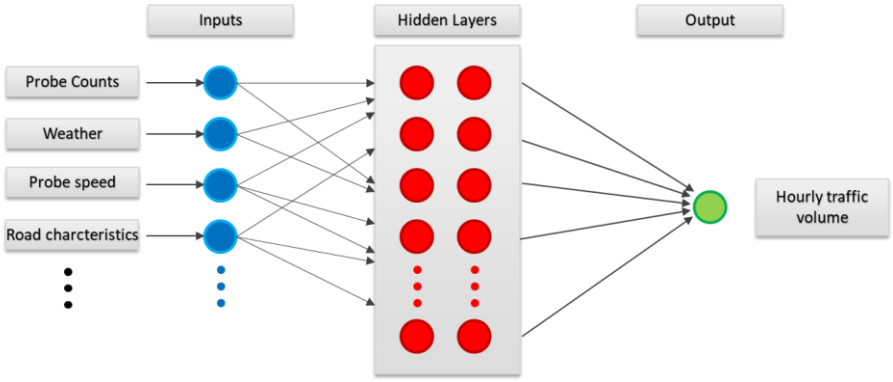


Figure 2. High-level architecture of the existing ANN model (Sekula et al., 2018)

The first step of this study introduces a proof-of-concept methodology to show how the direct incorporation of CCS volume data into the the-state-of-the-art model improves the networkwide hourly volume estimations. This framework aims to build off the initial work conducted in Sekula et al. (2018) by incorporating permanent CCS counts as a direct input to the model, thus accounting for the Spatio-temporal correlations between hourly link volumes. Whereas the previous work used CCS data solely for training and testing the proposed ANN, the current study further utilizes a subset of CCSs as an additional model input to improve the estimation accuracy, particularly focusing on choosing which CCSs to use for this purpose optimally. This task is addressed by assessing a handful of primary strategies to select the candidate CCSs, which will enter the model as explanatory variables and training new models. Using the New Hampshire road network as a case study, various estimation accuracy measures are employed to explore the effects of incorporating the CCS counts as additional features and compare the CCS selection scenarios. Note that throughout this study, traffic volume refers to the historical volume data unless otherwise stated.

Estimating historical traffic volume means that the introduced methods estimate statewide traffic volumes for time periods with known ground truth data of the CCS locations and networkwide traffic conditions such as traffic speed, weather, etc.

Given the results of the initial framework, this study introduces a graph-based methodology to integrate the links' volume correlations and traffic state characteristics into a single model. The proposed framework first introduces an algorithm to generate a graph representation of the road network. Then, this graph, besides the attributes available for each road segment and the ground truth traffic counts collected for a few links of the network, is inputted into one of the most recent machine learning methods named Graph Convolutional Networks (GCN). The introduced methodology includes an innovative model framework enabling the model to capture both temporal and spatial correlations between the links' traffic flows. Various components of the presented method are first tested using the data of the Worcester and Wicomico counties in Maryland. Then, the optimal framework is used to estimate 15-minute historical traffic flows for two distinct networks of the Maryland Beltway area and western Maryland (i.e., Allegany and Washington counties). Additionally, the framework is tested for real-time operation when the ground truth data might be delivered with some lags forcing the model to use previous time intervals' data for estimation.

1.3 Contributions

This study tries to address several existing gaps in previous studies and contributes to the transportation literature in the following directions:

1) Introducing a straightforward methodology built off of a state-of-the-art hourly traffic volume estimation model to prove how its performance improves by taking the network structure into account. In this work, attributes of a few road network links are fed to the model as complementary input. Besides, it is shown that network observability varies significantly based on the input links' selecting procedure. Higher observability and, in turn, a more accurate traffic volume estimation are obtainable for methods that tend to consider network structure and select input links evenly distributed over the network. The benefits of incorporating the network structure are illustrated in estimating hourly traffic volumes in the New Hampshire road network as a proof-of-concept.

2) Constructing a graph of the road network, which represents its underlying traffic characteristics and Spatio-temporal correlations. This graph is built upon the traffic patterns extracted from the probe vehicle movements in the network as a sample of the entire traffic.

3) Developing a GCN model that integrates the geometry of the road network and traffic conditions to estimate statewide 15-minute traffic flows using a handful of continuously collected volume data (i.e., CCSs data). The developed model borrows the idea of convolution operation from the Convolutional Neural Networks, a well-known model structure in computer vision. Contrary to the convolution operation in CNN models, which is applied to adjacent pixels of an image, the model in the present study adopts the convolution operation on the graph representation of the road network. Therefore, the model is capable of capturing the correlations between traffic volume in each link with the traffic volume in its adjacent links in the graph representation.

4) Introducing an innovative model framework to capture the spatio-temporal pattern of traffic volumes in both historical and real-time settings. The proposed framework consists of two parts. The first part uses the entire input data to train a specific GCN model for all time intervals to find the relations between attributes and traffic volumes and capture the spatial and temporal correlations between links' traffic volume. In the second part, the trained model in the first part is fine-tuned for each time step separately to focus on the traffic conditions in that time. Since the second part has a runtime in the order of seconds, the proposed framework is shown to be applicable to real-time traffic volume estimations.

1.4 Dissertation Structure

The rest of the dissertation is arranged as follows:

Chapter 2 summarizes the current research in the field of traffic volume estimation and reviews the literature of graph-based deep learning models and their application in transportation studies. The research gaps are discussed at the end of this chapter. In Chapter 3, a two-step proof of concept methodology is introduced. This methodology is designed to demonstrate the importance of incorporating the road network graph in a volume estimation model and is tested on the road network of the New Hampshire state. Chapter 4 introduces the proposed framework and elaborates on the GCN model, the graph generation algorithm, and the introduced model framework. Chapter 5 presents the data and networks used for the study experiments. Chapter 6 discusses the model performance analysis settings and introduces some experiments to test various components of the introduced framework. Using the findings of this chapter, chapter 7

provides the numerical results of applying the introduced methodology to various networks. Moreover, this section presents the results of applying the model to real-time traffic volume estimation. Chapter 8 concludes the finding of the study and provides some recommendations for future works.

Chapter 2: Literature Review

2.1 Overview

The concept of traffic volume estimation has always been an interesting topic in the transportation field with several applications. The number of vehicles passing a road segment is one of the essential inputs to many traffic analysis models at various levels, including planning, design, control, operation, and management.

However, the expensive procedure of collecting continuous traffic volume data has obligated researchers to use alternative and indirect measures. In the United States, states typically have 50 to 200 automatic traffic recorders (ATRs) within their highway network, permanently installed on or near the roadway and continually collecting 24-hour traffic counts (Wang and Kockelman, 2009). Additionally, short-period traffic counts (SPTCs) are collected at thousands of locations statewide via temporary sensor deployments. These data are commonly used to estimate aggregate traffic volumes called Annual Average Daily Traffic (AADT). AADT, as an alternative for the exact traffic volume, is widely studied in the literature of transportation and is used in many transportation projects (AASHTO, 2001). Therefore, the first part of this section, which provides a comprehensive review of traffic volume estimation models, is divided into two subsections. The first one investigates the works focused on estimating AADT, and the second one reviews the few pieces of research that aim to address the hourly traffic volume estimation problem.

Since the main contribution of this study is introducing a graph-based model for networkwide traffic volume estimation, the second part of this section provides a concise overview of Graph Neural Networks (GNN) as a recently developed machine learning technique. This part is also divided into two subsections. The first one presents the general concept and categories of GNNs, and the second one reviews the studies which applied the idea of GNN for solving transportation-related problems. In the end, we summarize the section and discuss the existing traffic volume literature gapes that this study aims to fill.

2.2 Volume Estimation Models

Numerous research papers have tried to develop models to estimate traffic volume – often in the form of AADT. Thus, the following subsection discusses various methods developed to estimate AADT. This subsection is followed by another shorter subsection that reviews the few works specifically focused on hourly traffic volume estimation.

2.2.1 AADT estimation

AADT, as an essential measure of aggregate traffic volume, is widely studied in the literature of transportation. The traditional Federal Highway Administration (FHWA) method for computing AADT is based on expanding SPTC data using daily and monthly factors estimated from groups of CCSs with similar traffic patterns. Besides, many statistical approaches like regression models have been developed to improve AADT estimation. One of the early works in this regard is done by Fricker & Saha (1987). In this study, statistical analysis is combined with subjective judgment to

forecast AADT in rural areas. They introduce two series of models, first an aggregate model based on the functional classification of a highway and the other one, a location-specific disaggregate model. They also present a six-step process for the following year's AADT prediction.

In another study, Aldrin (1995) proposed a statistical method that models daily car traffic based on variables such as road level, traffic trend, seasonal variations, day of week and time of day, special days, and statistical errors. This method is trained with simultaneous data from various CCSs, which enables capturing their inter-relationships.

Adding more details to the existing models, Zhao and Chung (2001) developed a multiple linear regression model to estimate AADT considering geographic information systems, general land use, and accessibility measurements. Later on, Zhao & Park (2004) introduced a geographically weighted regression model for AADT estimation to consider the spatial variation of locally estimated parameters.

Eom et al. (2006) used a spatial regression method to predict AADT for lower-class facilities. This model uses a geostatistical approach called Kriging to consider both spatial trends and spatial correlation. Other studies employed more advanced machine learning techniques to improve the AADT estimation accuracy, including Sharma et al. (2001), who developed an artificial neural network (ANN)-based method to estimate AADT. The main objective of their study is to improve AADT estimation in low-volume roads. This improvement is obtained by eliminating major sources of errors in the traditional factoring approach, including sampling error, seasonal and daily variations, as well as CCSs grouping error.

More recently, Castro-Neto et al. (2009) used the support vector machine for regression (SVR) to forecast AADT. SVR-based models that had been already used to solve transportation problems, like short-term traffic flow prediction and travel time estimation, had demonstrated significant improvements compared with previous studies (Ding et al., 2002). In their research, Castro-Neto et al. (2009) used the distribution of the training data to compute SVR prediction parameters and developed a method called SVR with data-dependent parameters (SVR-DP). Additionally, Rossi et al. (2014) studied the effect of clustering methods to identify road groups based on typical traffic patterns to improve group factor estimation while computing AADT.

One of the more recent studies concerning AADT estimation is the Khan et al. (2018) research. They applied ANN and support vector machines (SVM) to estimate AADT for various road classes using short-term counts. In their study, the SVM model is introduced as the best model, which not only outperforms regression and factor-based approaches but also yields better results compared to the introduced ANN model.

2.2.2 Hourly traffic volume estimation

One of the approaches to solve the hourly volume estimation problem is utilizing macroscopic traffic models. Shimizu et al. (1998) applied state estimation algorithms to build an hourly traffic volume estimation system. They model the hourly traffic volume by a linear time-varying discrete dynamic system and use filtering algorithms (i.e., Kalman filter, interval smoother, and the MIPA Kalman filter) to remove noises from the data collected by detectors.

Herrera & Bayen (2008) added mobile sensor data to the collected data from detectors to improve traffic state estimation. They used assimilation methods such as the Kalman filter to find the state of the highway at any point in time and space. Their results show significant improvement achieved by incorporating mobile sensor data.

Work et al. (2008) transformed GPS devices' data into usable traffic information using data assimilation algorithms to enhance traffic state estimation. Papageorgiou et al. (2010) presented METANET, which is a macroscopic simulation tool for estimating traffic variables such as traffic volume. These macroscopic model-based approaches, however, are not scalable to large networks like a state network.

Transforming AADT into hourly volume profiles is perhaps the most common approach used for obtaining reasonable hourly traffic volume estimates at the state level– a process explained in Schrank et al. (2015) that uses AADT and speed profiles to obtain the hourly traffic profile for a typical week. While this approach was not initially intended for hourly volume estimation purposes, it often does an excellent job in capturing typical traffic behavior; however, by design, it does not capture aberrations caused by unique weather conditions or incidents. To consider these factors, Sekula et al. (2018) introduced an ANN-based regression approach that estimates hourly traffic volume using multiple data sources such as vehicle probe counts and speeds, weather stations, and road characteristics. This approach yielded appreciably higher estimation accuracy than the widely used profiling method and provided a framework for transportation agencies to obtain scalable statewide hourly volume estimates. However, their approach does not consider any spatial correlation between the transportation network links. To address this problem, Zahedian et al. (2020) introduced a framework

that selects some CCSs based on their position in the transportation network and adds their features, such as their count data and Euclidian distance, to the ANN model inputs so that the model can make more accurate estimation by capturing some level of spatial correlations in the network. Yi et al. (2021) used Breadth-first search (BFS) on the traffic network to extract spatial dependency features and add those features to their introduced extreme gradient boosting tree (XGBoost) for volume estimation. These studies indicate that adding spatial features to machine learning models improves their performance for traffic volume estimation. However, these methods cannot directly consider the transportation network's graph structure in the training process.

2.3 Graph Neural Networks

Many machine learning tasks are revolutionized in recent years due to two main reasons: first, a significant increase in the amount and variety of the available training data, and second, rapid developments in computer hardware resources (e.g., GPUs). Researchers have leveraged these advancements to implement innovative machine learning algorithms like neural networks to a variety of problems in various research fields. In most of these fields, the data has a Euclidian distance representation. However, there are many applications where data is generated from the non-Euclidian domain.

Transportation networks are examples of such applications with complex relationships in the form of a general graph- i.e., with a non-Euclidian nature. For these networks, recently, many studies have been developed to address graph-based machine learning problems, such as Graph Neural networks (GNNs). The following sub-section provides

a concise review of the literature of GNNs, including general concepts and categories of this field of study. Further, another sub-section summarizes studies that use GNNs to solve transportation-related problems.

2.3.1 General frameworks and categories

End-to-end deep learning models like Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) perform effectively on Euclidian data. However, they fail to appropriately capture the hidden patterns in data presented in the form of a graph. GNNs, also known as geometric deep learning models, are specifically designed to fill this gap. This group of models tends to release the assumption of independence of data points and train a machine to recognize the statistical pattern of graphs with any complex forms.

In general, given the graph structure and node features as the inputs of a GNN, the outputs can be node-level, edge-level, or graph-level labels. The GNN can be trained in supervised, semi-supervised, or completely unsupervised frameworks depending on the available label information.

Implementing neural networks on graphs is studied since the late 1990s when Sperduti and Starita (1997) applied it to acyclic graphs. However, Gori et al. (2005) is the first study outlining the concept of GNNs. Wu et al. (2020) have published a comprehensive survey on GNNs. They define four main categories of GNNs:

- Recurrent graph neural networks (RecGNNs),
- Convolutional graph neural networks (ConvGNNs),
- Graph autoencoders (GAEs), and
- Spatial-temporal graph neural networks (STGNNs).

RecGNNs, as one of the early works on GNNs, are designed to learn node representation with recurrent neural architecture (Scarselli et al., 2009; Gallicchio et al., 2010). These early works that try to find a node representation by iteratively propagating neighbor features are incredibly costly in terms of computational time. However, they form the inspiring platforms of the ConvGNNs, which are the most popular category of GNN with applications in different areas.

ConvGNNs, which are promoted by the successful application of CNNs, try to expand convolution operation to a general graph. Similar to CNNs, which perform on grid graphs (i.e., images), ConvGNNs aim to aggregate a node and its neighbors' features to provide a high-level representation of that node. ConvGNNs are themselves divided into two categories of spectral-based and spatial-based models.

Bruna et al. (2013) first used the spectral graph theory to develop a graph convolution. Later, many researchers (Defferrard et al., 2016; and Levie et al., 2017) developed and extended this idea. Although spatial-based ConvGNNs were first introduced in 2009 by Micheli et al. (2009), they remained unpopular until recent years when Atwood and Towsley (2016) used the diffusion process to capture spatial dependency of graph-structured data.

The other two categories of GNNs (i.e., GAEs and STGNNs) are essentially built on RecGNNs and ConvGNNs. STGNNs, focusing on extracting the spatial-temporal dependencies in a graph, are explicitly applicable to transportation-related problems. The reason is the existence of the temporal and spatial correlations between the traffic patterns of different road segments. The following section introduces studies that are mainly used in this approach to address traffic measures estimation.

2.3.2 Application in transportation

As mentioned earlier, GNNs have a variety of applications across different areas of research, including the transportation domain. Elaboration on the underlying graph of GNNs seems advantageous in analyzing the intricate pattern of traffic in a transportation network. Although the introduction of GNNs is relatively new, many studies applied GNNs to transportation-related problems. Yao et al. (2018) introduced a Deep Multi-View Spatial-Temporal Network (DMVST-Net) to predict taxi demand. This framework aims to capture both spatial and sequential temporal relations at the same time. They incorporate Long-Short Term Memory (LSTM), CNN, and network embedding (Tang et al., 2015) to forecast taxi demand for a location within a time interval. In this study, they show how the combination of CNN and LSTM by considering the graph structure of the road network can outperform state of the art prediction methods like XGBoost (Chen & Guestrin, 2016) and ST-ResNet (Zhang et al., 2017) when applied to the taxi demand prediction problem.

In another application, Li et al. (2018) applied GNNs to predict speed in a road network. They introduce a GNN called Diffusion Convolutional Recurrent Neural Network (DCRNN) to consider spatial and temporal correlations between road segments while predicting short- and long-term speed. Their system architecture, illustrated in Figure 3, includes Recurrent layers of Diffusion Convolutions, which are a general form of spectral graph convolutions. It also incorporates encoder and decoder recurrent neural networks to generate speed predictions based on either previously measured or estimated speed.

They test the introduced DCRNN on two datasets of METR-LA (Jagadish et al., 2014) and PEMS-BAY. The results are compared with both baseline models like Historical Average, Auto-Regressive Integrated Moving Average model with Kalman filter (ARIMA kal), Support Vector Regression (SVR), and deep learning-based models like Feedforward Neural Network (FNN) and Recurrent Neural Network with fully connected LSTM hidden units (FC-LSTM) (Sutskever et al., 2014). The DCRNN model outperforms all the mentioned models for all forecasting horizons, proving the importance of considering the spatial and temporal correlation in a road network. Zhang et al. (2018) also introduced a GNN model applicable to the traffic speed forecasting problem. They put forward Gated Attention Networks (GaAN), which is also a variation of STCNNs. In this study, the neural attention network idea (Bahadanau et al., 2015) is used as a graph aggregator to reduce the size of the Spatio-temporal network. Applying this model to the METR-LA dataset, the authors show how GaAN beats all the state-of-the-art models, including the DCRNN (Li et al., 2018).

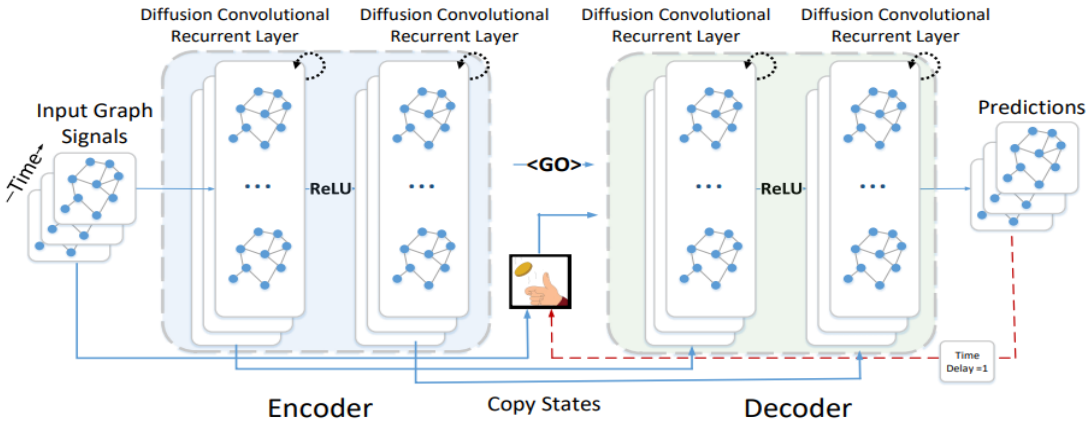


Figure 3. The system architecture of the DCRNN (Li et al., 2018).

Another STGNN method applied to traffic measurement estimation is developed by Yu et al. (2018), which is also concerned with traffic speed prediction. They introduce a Spatio-Temporal Graph Convolutional Networks (STGCN) to solve the time series furcating problem in a road network. The architecture of this Spatio-temporal graph convolutional network is illustrated in Figure 4.

One of the innovative ideas of this study is formulating the problem on graphs and constructing a model with complete convolutional structures. This implementation enables a much more efficient training process compared to the regular convolutional and recurrent units. This STGCN model is tested on two datasets of BJER4, a dataset of double-loop detectors in Beijing City, and PeMSD7 collected from Caltrans Performance Measurement System (Chen et al., 2001).

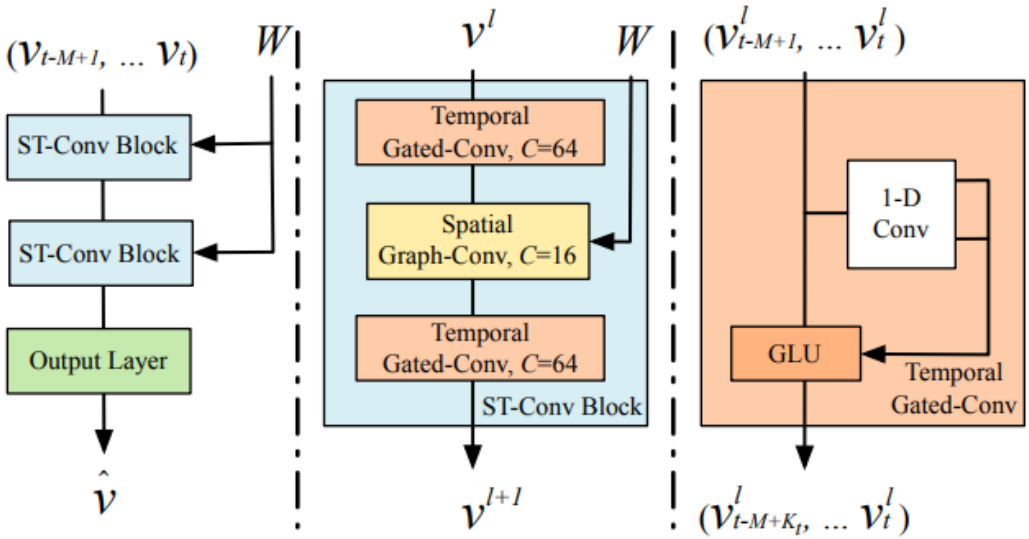


Figure 4. The system architecture of the STGCN (Yu et al., 2018).

The authors compare their model with the state-of-the-art baseline models and show how the introduced STGCN outperforms all of them, including the model proposed by

Li et al. (2018). The main advantage of this model is its significantly lower required computational time.

Recently, GNNs are also used for traffic flow prediction. Zhang et al. (2018) propose a Kernel-Weighted Graph Convolutional Network (KW-GCN) that combines node weights and learns the traffic features locally while considering the global structure of the road network. In this study, the weighted kernels are used to account for the diverse local traffic state. They test their model on the Beijing taxi dataset at intersection and road level. To predict traffic flow, they assume that they have traffic flow at all the graph nodes for the six-time intervals before predicting the flow. Their numerical results show that their proposed weighting approach leads to superior performance compared to other forecasting methods.

Li et al. (2021) developed a Multisensor Data Correlation Graph Convolution Network model, named MDCGCN, constructed of three main parts of recent, daily, and weekly components. The architecture of their proposed method is presented in Figure 5.

According to this figure, each of the three parts of recent (Y_h), daily (Y_d), and weekly (Y_w) include two multisensory data correlation convolution (MDCC) blocks and one 2D convolution block. Moreover, the daily period component consists of a benchmark adaptive mechanism (BA-Block). They test their model on PEMSD4 and PEMSD5 datasets and divide it into training, verification, and test sets temporally. This means that the same as previous research, they assume that the ground truth traffic volume data is available in previous times for the links on which they want to predict volume.

They compare their results with other forecasting models, including different variations of GCN, and show how the proposed MDCGCN is outperforming other models for long-term predictions such as 60-minute ahead.

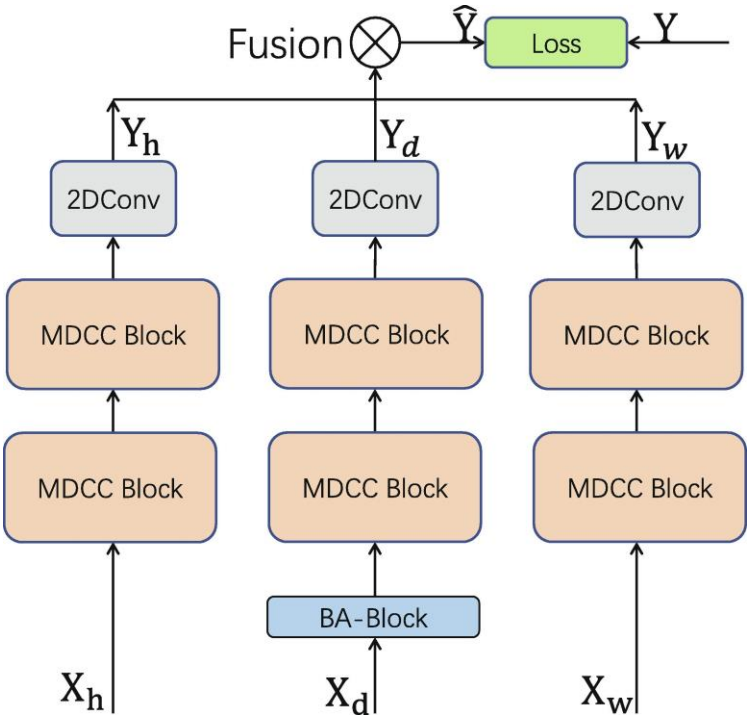


Figure 5. The model architecture of the MDCGCN (Li et al., 2021).

The studies mentioned above are only a sample of studies that developed in recent years, applying various GNN models for transportation-related problems. A recent study published by Jiang & Luo (2021) provides a comprehensive survey of these studies.

2.4 Chapter Summary

As we discussed in the first section of this chapter, the problem of networkwide traffic volume estimation has been of interest to many researchers for a long time. In recent years, advanced machine learning methods have come to the help of transportation researchers to develop more accurate link volume estimation models applicable to a variety of road networks. However, the fact that the traffic state of the links in a road network is correlated is mainly overlooked. Even researchers that used advanced machine learning methods (i.e., deep learning-based models) ignore the complicated data structure of the road network. This gap can be filled by introducing models that estimate traffic volumes while considering the underlying graph of the road network.

The second section of this chapter discussed a relatively new class of machine learning models called GNN. These models aim to expand deep learning models to apply them to problems with general graph-structured data. Transportation networks are one of those areas with complex graph-based relations. A few recent studies that used this kind of model in transportation-related problems were also introduced in this section.

These studies show how GNN-based models outperformed the previous state-of-the-art methods in solving problems like speed and flow prediction. However, to the best of our knowledge, their application in networkwide volume estimation, when the main challenge is estimating volume for locations where no previous traffic volume data is available, has not been investigated yet. The current study aims to address this problem by introducing a new GNN-based model that considers the complex characteristics of the traffic state and trip patterns in a road network.

Chapter 3: A Proof-of-Concept Methodology

3.1 Overview

This section introduces a two-step methodology designed to prove the importance of incorporating spatial correlations between roads in a statewide hourly traffic volume estimation model. The first step is selecting a subset of available CCSs (i.e., the source of ground truth volume data). The second step is incorporating the data of the selected CCSs as explanatory variables into a previously developed machine learning regression model (Sekula et al., 2018) for estimating hourly traffic volumes. In other words, for estimating hourly traffic volume in any specific road, the new model adds the selected CCSs data to its other available data to account for their possible dependency. The first part of this chapter, section *3.2*, introduces various strategies to select the subset of CCSs. Section *3.3* explains the process of training a fully connected neural network with selected CCSs data as additional inputs. Then, the introduced proof of concept framework is tested using the network of New Hampshire, the data of which is described in section *3.4*. Finally, the implementation and numerical results of the model are presented and discussed in section *3.5*.

3.2 Candidate CCS Selection Strategies

The road network is a connected graph with intercorrelated traffic flows in its links, which means ground truth volume data in a road segment may have a high correlation

with (and thus valuable for accurately estimating) volume data in other links. States generally have several CCSs installed continuously recording hourly volumes - each of which represents the traffic volume of the links associated with it.

This section seeks to utilize a subset of this recorded volume data (i.e., CCS counts and their features) as additional independent variables in a volume estimation model to capture the spatial and temporal interdependencies between links. In particular, explanatory (i.e., input) CCSs should be selected in a way to appropriately capture the intercorrelation between each of these CCS readings and other link volumes. In the following subsection, we introduce and briefly explain four basic strategies used here to select candidate CCSs as a proof of concept.

Random Strategy

This strategy randomly selects *n* CCSs among all the CCSs whose data is available and serves two purposes. First, any possible improvement in volume estimation based on this strategy demonstrates a benefit in adding CCS attributes to the volume estimation regardless of the applied selection strategy. Second, this strategy serves as a baseline for evaluating the performance of other selection strategies.

AADT-based strategy

AADT is an essential measure of traffic and can be computed for the links with permanent traffic volume recorders (i.e., CCSs) by aggregating count data on the average annual daily level. Equation 1 illustrates the objective function used to select CCSs based on the AADT value:

$$argmax_{A' \subset A, |A'|=n} \sum_{i \in A'} AADT_i \qquad (1)$$

where A is the set of all the CCSs within a state road network, A' is the selected subset, and $AADT_i$ is the AADT of the *i*th CCS $\in A'$.

Equation 1 immediately communicates that the *n* CCSs with the highest AADT values are selected. The reason for introducing this strategy is that roads with higher AADT are generally among the main arteries of the network, making it likely that traffic volume in these links may impact the traffic volume of many adjacent links. Therefore, adding these CCSs' data as inputs may improve volume estimation in busier areas, where traffic control is more critical.

CCSs distance-based strategy

The primary motivation for adding new variables is to provide information about exact traffic counts in some links of the network, which are highly correlated to other links and should help estimate volumes on them with higher accuracy. In this regard, the spatial distribution of the selected CCSs is of great importance. One can reasonably assume that each CCS represents the traffic volumes in its vicinity; consequently, a CCS selection strategy that maintains a uniform coverage over the entire network is a reasonable objective. Equation 2 presents the general form of the objective function used to represent this strategy:

$$max_{A' \subset A, |A'|=n} \sum_{i,j \in A'} (d_{ij})^m \quad (2)$$

where A and A' are the same as the previous definition, d_{ij} is the Euclidian distance between the two selected CCSs based on this strategy, and m is a parameter that normalizes the distance impact and is network specific. To choose a proper value of m, one can solve the optimization problem of this strategy for the study network and

different values of m. Therefore, the value of m, which visually presents a more uniform coverage over this specific network, will be chosen.

TMC coverage-based strategy

This last strategy is similar to the previous one, but rather than focusing on the geographical locations of the CCSs, it deals with the spatial distribution of the CCSs concerning their position in the Traffic Management Center (TMC) network. Specifically, this strategy selects a subset of CCSs evenly distributed over the TMC network, with each TMC assigned to its closest CCS within the selected subset. This strategy is presented as a linear program (LP) shown in a set of equations as follows:

$$\text{Min} \sum_{i \in A, j \in T} x_{ij} t_{ij} \tag{3}$$

$s.t.$

$$\sum_{i \in A} x_{ij} = 1, \forall j \in T \tag{4}$$

$$\sum_{i \in A} y_i = n \tag{5}$$

$$x_{ij} \leq y_i, \forall i \in A, j \in T \tag{6}$$

$$y_i, x_{ij} = 0 \; or \; 1, \forall i \in A, j \in T \tag{7}$$

where T is the set of TMC network links, A is the set of all the CCSs, t_{ij} is the Euclidian distance between the start point of $TMC_j, j \in T$ and $ATR_i, i \in A$, and y_i and x_{ij} are binary variables. $y_i = 1$ if ATR_i is selected based on this strategy and $y_i = 0$ otherwise. $x_{ij} = 1$ for $TMC_j, j \in T$ assigned to $ATR_i \in A$, otherwise $x_{ij} = 0$.

The objective function of this LP, Equation 3, is minimizing the total distance between TMCs and their associated CCSs in the selected subset. Furthermore, Equations 4 and 6 are satisfying the condition that each TMC must be assigned to one CCS from the selected subset. Finally, equation 5 imposes the constraint of the number of selected CCSs.

3.3 Fully Connected Feedforward Multi-Layer ANN

Inspired by animal brains, artificial neural networks (ANN) are computing systems consisting of layers of neurons. A general ANN contains three layer types: the input layer, hidden layers, and output layer. As a class of ANNs, a Fully Connected Feedforward Multi-Layer ANN includes multiple hidden layers where all the neurons in a layer are connected to all the neurons of the previous layer without forming a loop. The inputs to the first layer of an ANN are the data features, and the output of the last layer is the model estimation. The forward propagation rule in this model is presented in the following equation.

$$a_i^{(l+1)} = f\left(w_i^{(l+1)} a^{(l)} + b_i^{(l+1)}\right) \quad (8)$$

where $a_i^{(l+1)}$ is the output of the neuron i in the $(l+1)^{\text{th}}$ layer, $w_i^{(l+1)}$ and $b_i^{(l+1)}$ are the weight and bias vector between that neuron in the $(l+1)^{\text{th}}$ layer and all the neurons of the previous layer, and $a^{(l)}$ is the output vector of all the neurons in the l^{th} layer. $f(.)$ is the so-called activation function used to account for the nonlinear relationships between the data points.

Given this general configuration of the ANN model, Sekula et al. (2018) applied it to the hourly volume estimation problem. Their final model consists of three hidden

layers, 256 neurons, and exponential linear units (ELU) activation function in each hidden layer (Clevert et al., 2015). The detailed architecture of their model is shown in Figure 6. The ground truth data used for training and testing is the hourly traffic counts of the roads with CCSs.

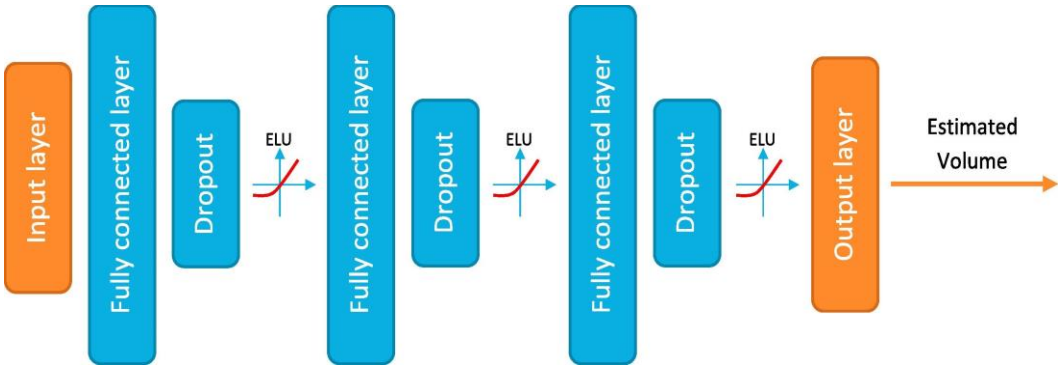


Figure 6. The detailed architecture of the fully connected ANN (Sekula et al., 2018).

The input layer of this model includes the following 84 features:

- *Vehicle probe volumes*. This is the number of vehicles in the sample GPS data (Marković et al., 2018). These volumes that include three classes of vehicles (less than 14k lb, between 14k and 26k lb, above 26k lb) are aggregated for 30-minute intervals. Thus, for each hour, there are six vehicle probe volumes (three classes of vehicles in two 30-min intervals). In addition to these six features, vehicle probe volumes of the 30 minutes before the observed hour are also added as input features - forming a total of 9 features.
- *Vehicle probe speeds*. This is the measured speed using vehicle probe data. This data is acquired from the Regional Integrated Transportation Information

System (RITIS). Two features of speed, including average hourly speed and approximate free-flow speed, are used.

- *Weather data.* A total of 36 features that describe the hourly weather data are used. This data obtained from Weather Underground (2017) includes features like Temperature, Visibility, Precipitation, and Weather Description.
- *Infrastructure data.* This data that forms seven features includes the number of lanes, speed limits, class of the road (motorway or trunk), and type of the road (Interstate, US, or state road).
- *Temporal data.* This forms around 29 features describing the time of day (1, 2, ..., 24), day of the week, and those special holidays during the observed period.
- *Volume profiles.* This is the hourly volume profile of a typical day computed from the well-known profiling method (Schrank et al., 2015).

In our proposed methodology, we use the same architecture and only change the network's input layer to add the attributes of the selected CCSs. In addition to the 84 previously introduced features that are describing a road, we add attributes of the selected CCSs to the model. Figure 7 shows a schematic depiction of our proposed model vs. the base model of Sekula et al. (2018).

The list of added attributes from n selected CCSs are as follows:

- *CCS volumes.* This is the volume in the selected CCSs during the observed hour. Thus for n selected CCSs, this adds $9n$ features.

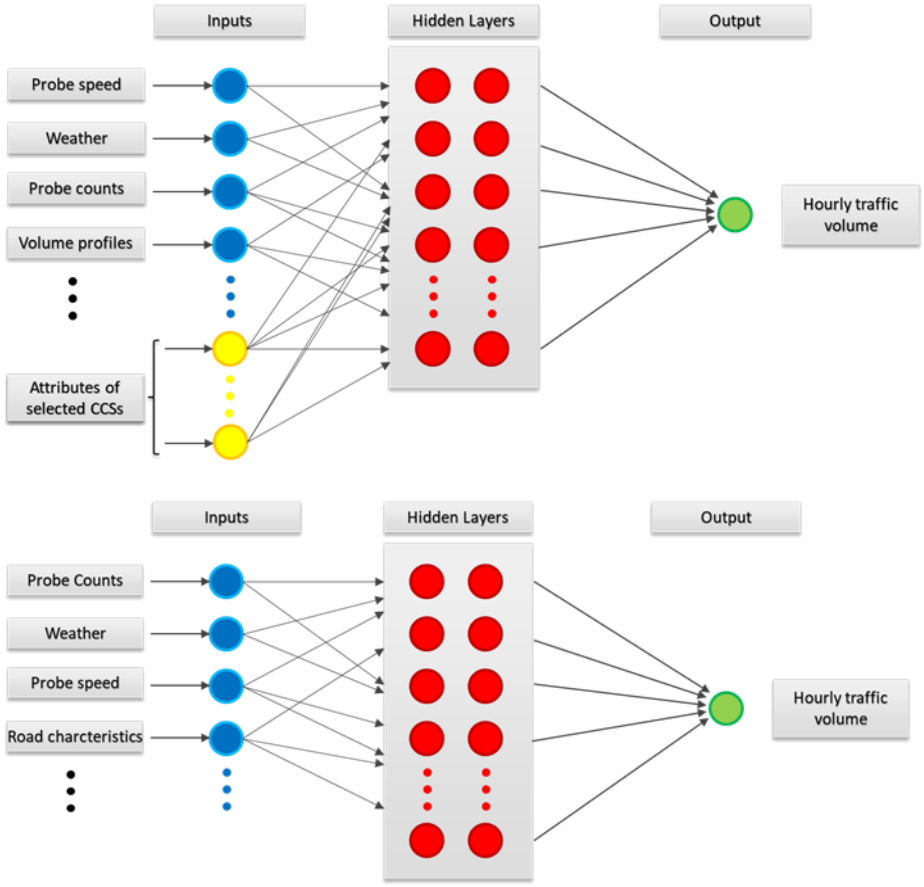


Figure 7. Schematic depiction of our proposed model (up) vs. the base model of Sekula et al. (2018) (down)

- *Euclidian distance to selected CCSs.* This is the Euclidian distance from each road to each selected CCS. This also adds n features for any observed road, which is the same over the observed hours.
- *Speed.* This is the speed on the selected CCSs during the observed hour measured from GPS probe data ($2n$ features).
- *AADT.* This is the average annual daily traffic of the selected CCSs, which is computed from their ground truth data (n features).
- *Reference speed.* This is the reference speed on the selected CCSs (n features)

- *Road characteristics.* This includes features like the number of lanes, type of the road, and Functional Road Classes (FRC) of the selected CCSs ($7n$ features).
- *Open Street Map (OSM) features.* This is the one-hot encoded OSM-based road class ($4n$ features).

After adding these features, a new model will be estimated based on each CCS selection strategy. The following section summarizes the process of selecting CCSs based on different strategies and training ANN models with CCSs' additional inputs using the New Hampshire road network as a case study.

3.4 Experiment Configuration and Data Description

As mentioned in section 3.2, in our introduced method, unlike Sekula et al. (2018) that uses the entire ground truth data for training and testing, a portion of ground truth data comprises the additional inputs. Additionally, in section 3.2, we described four basic strategies for selecting the subset of input CCSs (i.e., roads), which are the extra inputs. Therefore, this section will first introduce the New Hampshire road network and data sources used to implement the two-step model. This introduction is followed by using the New Hampshire road network to apply the CCS selection strategies and training new ANN models based on the chosen CCSs to observe the performance improvement of the hourly volume estimation model. Further, a description of the numerical results of the trained models is presented.

New Hampshire road network is an excellent example for testing the hourly volume estimation model because of its uneven distribution of traffic. Here we use the National

Performance Management Research Data Set (NPMRDS) TMC network of New Hampshire. Figure 8 illustrates the New Hampshire TMC network.

The data sources used as inputs to the machine learning regression model are based on the model proposed in Sekula et al. (2018) with slight modifications to fit the new methodology. For clarity, these hourly TMC-based data sources are summarized below:

Continuous Count Stations (CCS): New Hampshire DOT provided access to continuous count data from fixed locations throughout the state via a traffic management web application. These station locations were manually mapped to the TMC network via a manually created lookup table. Further, an automated process was developed to extract count data from the web application and assign it to the TMC network at the hourly level. Note that in the previous work (Sekula et al., 2018), CCS count data represented the ground truth data source used to calibrate and evaluate the model. However, in this study, the CCS count and additional attributes of each of the selected CCSs are also strategically introduced as model inputs. These attributes are CCS volumes, Euclidian distance to that CCS, speed, reference speed, number of lanes, AADT, Functional Road Classes (FRC), FHWA-approved Functional Classification System (F_system), type of the road, precipitation, temperature. It is noteworthy that these attributes are added per selected CCSs, which yields a total of $11 \times n$ additional features for n selected CCSs.

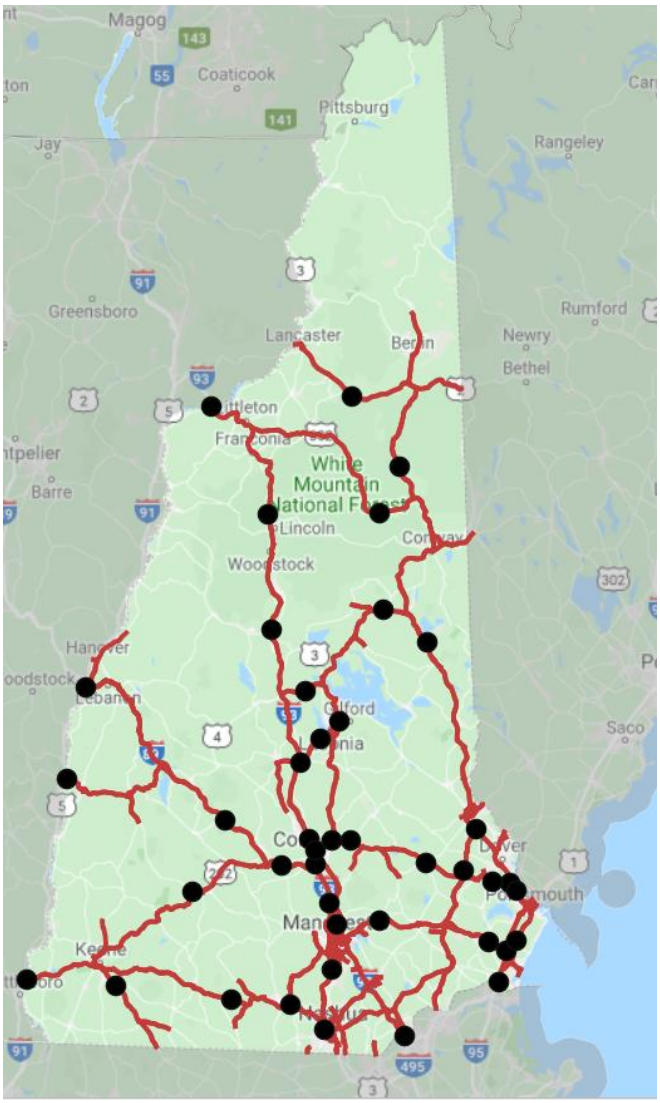


Figure 8. New Hampshire network (red lines: the NPMRDS TMC network; circles: the location of CCSs)

- *Probe counts*: GPS probe counts represent the number of unique probe vehicles traveling through a TMC segment in an hour and are obtained by extensively processing raw trajectory data. The raw trajectory data (i.e., GPS traces from a sample of vehicles on the road) must first be associated with the TMC road.
- *Probe speeds*: Hourly speed data on each TMC segment was downloaded through the I-95 Corridor Coalition Vehicle Probe Project.

- *Road / infrastructural characteristics*: The majority of TMC-based road characteristics (e.g., road classifications, number of lanes, AADT) were obtained via conflation from other data sources. HPMS-based data attributes were obtained through a data conflation effort conducted by the Texas Transportation Institute for the National Performance Research Dataset (NPMRDS., 2018). However, OpenStreetMap (Haklay & Weber, 2008) attributes were obtained via a conflation approach developed by Vander Laan & Sadabadi (2019).
- *Weather*: Historical weather information was obtained via the Iowa Environmental Mesonet (Accessed 2019), which archives granular weather data from weather stations. Aggregating weather station data provided TMC-based weather attributes (e.g., precipitation, temperature) at the hourly level by assigning the nearest station's attributes to each TMC.
- *Temporal Info*: Temporal features include flags to indicate the hour of the day, day of the week, presence of a holiday, etc.
- *Volume profiles*: Volume profiles capture the hourly traffic volume for an average week based on the TTI method described (Schrank et al., 2015).

One of the hyperparameters of the introduced methodology is the number of selected CCSs, n. Since the selected CCSs will be entered as input variables and therefore cannot be used for training and testing the ANN model, the value of n must balance the trade-off between the input CCSs and training and testing CCSs. In this example, we set $n = 8$, which is roughly 20% of the CCSs. By doing so, we leave enough data for training and testing.

3.5 Implementation of the Two-Step Model

This section demonstrates and discusses the results of the four CCS selection strategies using the New Hampshire dataset as a case study. As was previously mentioned, the New Hampshire dataset includes data from 42 active CCSs. Here, we choose eight specific CCSs for each selection strategy (roughly 20% of all the stations) as input variables. For each strategy, we first select the optimal set of 8 stations and then use the data of the remaining 34 stations for training and testing of the ANN model.

3.5.1 CCSs selection results

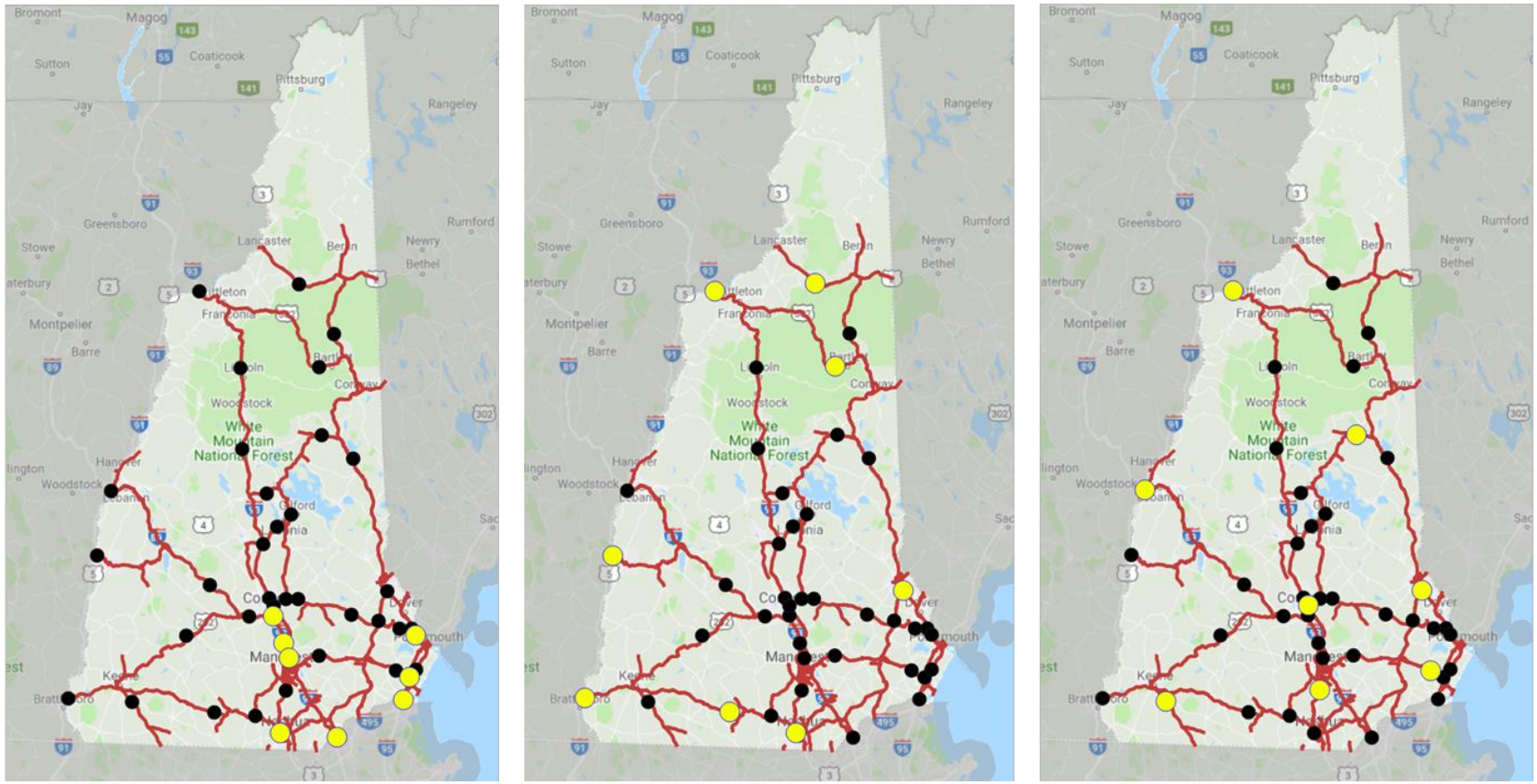
Selecting the candidate CCSs based on the random strategy is a simple task; however, to capture the characteristics of a random strategy, the process must be done for more than one random set. Thus, we generate ten distinct random sets and use the average of the final outputs of all these ten sets as the reported results for the random strategy. The selection of the optimal set of CCSs for the other three strategies is more straightforward. For the AADT-based strategy, we simply sort the AADT of the 42 active CCSs and select the top 8 stations to be used as input. Finding the optimal sets for the CCS distance-based and the TMC coverage-based strategies requires solving the associated LPs. Solving the LPs is done using the commercial optimization solver, FICO Xpress (2019). Note that for the CCS distance-based strategy, to choose a proper value of m, we solved the optimization problem of this strategy for the network of New Hampshire and different values of m. Each time we visually checked the spatial distribution of the selected subset to see which one presents a more uniform coverage over this specific network, resulting in $m = 0.5$. Figure 9 illustrates the selected sets of

CCSs based on the last three strategies in the map of New Hampshire and its TMC network.

Note that the AADT-based strategy yields selected stations accumulated in a small area due to the uneven distribution of the residential areas in the state. In the CCS distance-based strategy, selected CCSs are distributed around the state's border, yielding an uneven distribution considering the underlying network. On the other hand, the last strategy, which is more advanced and considers the TMC network, provides uniform distribution and coverage over the entire network.

3.5.2 ANN model results

After selecting the input CCS subset for each strategy, the rest of the CCSs are used for full cross-validation of the ANN model. Full cross-validation results in different training and testing sets for each strategy, a consequence of which is that a meaningful comparison of the absolute error metrics is not possible. To make the results comparable, two distinct ANN models are trained for each strategy, one of which has the attributes of the selected CCSs as input variables, while the other one does not. Since both models use the same set of training and testing data (i.e., data from 34 remaining stations), investigating the impact of added explanatory variables (i.e., attributes of the input CCSs) is possible. The learning process is performed with the AdaM (Adaptative Momentum) optimizer algorithm (Kingma & Ba, 2014) using the following parameters: learning_rate = 0.001, β_1=0.9, and β_2=0.999, and also utilizing the Dropout technique (Hinton et al.,2012). The input data is normalized with the following formula:



AADT-based strategy | CCS distance-based strategy | TMC coverage-based strategy

Figure 9. Selected sets of CCSs based on three strategies in the New Hampshire network (circles show the location of CCSs, the yellow ones are the selected stations, red lines are the NPMRDS TMC network)

$$x_{norm} = \frac{x - \bar{x}}{std(x)} \tag{9}$$

where x_{norm} is the normalized feature, x is the original feature, $\bar{x}$ is a mean of x and $std(x)$ stands for the standard deviation of x, and the loss function is defined using the Mean Absolute Error of the estimations. Previous research (Sekula et al., 2018) showed that this architecture and training procedure leads to good results and is not prone to overfitting even with relatively small datasets. To be consistent with that study, three error measures of Mean Absolute Percentage Error (MAPE), Mean Error to Maximum Flow Ratio (MEMFR) and R^2 were selected for evaluating the model results. The formulas for these metrics are as follows:

$$MAPE = \left(\frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right|\right) * 100 \tag{10}$$

$$MEMFR = \left(\frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_{max}}\right|\right) * 100 \tag{11}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{12}$$

Tables 1 to 4 illustrate the overall performance of the ANN model with and without CCS variables selected based on previously described strategies. Note that the values represented in Table 1 are computed by averaging the results of 10 distinct pairs of models trained based on ten different random sets. Looking at each table individually, it is apparent that adding extra CCS input data – regardless of strategy – yields a positive impact. However, to find the optimal strategy for selecting candidate CCSs, it is useful to look at the relative improvements achieved by each strategy (relative to their baseline counterparts without the additional data). Table 5 summarizes the relative

improvement of the mean values for all the strategies and shows that the TMC coverage-based strategy yields the best improvements in all of the evaluation measures. To better understand how each strategy improves the hourly traffic volume estimation, Figure 10 summarizes the relative improvements of the mean values for all strategies in a single figure. It re-emphasizes that all strategies improve the results with respect to the baseline conditions without including CCS data and clearly shows that the TMC coverage-based strategy yields the most improvement. These results corroborated via statistical testing confirm that the improvements in the mean error metrics achieved by the TMC coverage-based approach are statistically significant (student's t-test at 5% significance level). Additionally, Figure 11 illustrates the error distribution (a) without CCS inputs and (b) with CCS inputs of TMC coverage-based strategy for each metric using box-whisker plots, which are more informative about the distribution of each measure. These plots communicate that the median R^2 increases while the median of the other two metrics (i.e., MAPE and MEMFR) decrease, showing performance improvement when the ANN uses CCS inputs.

Additionally, Figure 11 shows that all quantiles improve for each error metric, although the improvement is more noticeable for MAPE and MEMFR. This improvement is also evident in the range of these measures. The lower range of the measures together indicates a smaller variation in the estimations of the model with CCS inputs. For a better illustration of this claim, the heat maps of all the data points used for testing the ANN models without and with CCS inputs of TMC coverage-based strategy are shown in Figure 12.

Table 1. Overall performance of the ANN with and without CCS inputs of random strategy

ANN Model	Without CCS inputs			With CCS inputs		
Measure	R^2	$MAPE$	$MEMFR$	R^2	$MAPE$	$MEMFR$
Minimum	-2.26	15.94	4.76	0.33	12.75	3.87
25th percentile	0.74	23.14	5.99	0.74	21.10	5.61
Median	0.81	28.34	7.29	0.84	26.91	6.76
75th percentile	0.88	40.51	8.51	0.89	36.48	8.44
Maximum	0.94	319.24	26.69	0.95	178.65	17.40
Mean	0.71	47.57	8.24	0.81	36.11	7.40

Table 2. Overall performance of the ANN with and without CCS inputs of AADT-based strategy

ANN Model	Without CCS inputs			With CCS inputs		
Measure	R^2	$MAPE$	$MEMFR$	R^2	$MAPE$	$MEMFR$
Minimum	-1.83	14.79	4.46	-0.03	13.08	3.75
25th percentile	0.72	23.88	6.02	0.76	22.82	5.55
Median	0.80	28.74	7.23	0.83	25.97	6.67
75th percentile	0.87	43.52	8.57	0.88	36.40	7.97
Maximum	0.93	331.52	25.98	0.95	141.47	19.49
Mean	0.71	46.08	8.23	0.79	37.53	7.47

Table 3. Overall performance of the ANN with and without CCS inputs of CCS distance-based strategy

ANN Model	Without CCS inputs			With CCS inputs		
Measure	R^2	$MAPE$	$MEMFR$	R^2	$MAPE$	$MEMFR$
Minimum	-0.13	15.91	4.64	0.46	12.06	3.75
25th percentile	0.77	22.31	5.87	0.78	19.57	5.31
Median	0.83	26.14	7.02	0.87	24.15	6.24
75th percentile	0.88	38.14	8.13	0.90	34.62	8.01
Maximum	0.94	272.84	18.60	0.96	179.24	15.19
Mean	0.80	38.82	7.54	0.83	32.79	6.92

Table 4. Overall performance of the ANN with and without CCS inputs selected by TMC coverage-based strategy

ANN Model	Without CCS inputs			With CCS inputs		
Measure	R^2	$MAPE$	$MEMFR$	R^2	$MAPE$	$MEMFR$
Minimum	-3.65	16.02	4.96	0.29	12.63	3.61
25th percentile	0.74	24.70	5.89	0.75	21.43	5.44
Median	0.81	29.95	7.48	0.84	25.34	6.70
75th percentile	0.88	47.39	8.99	0.90	41.81	8.18
Maximum	0.94	411.14	32.83	0.96	175.76	19.96
Mean	0.67	55.56	8.51	0.80	35.90	7.36

Table 5. The relative improvement of the mean values for all the strategies

Measure / Strategy	R^2	$MMAPE$	$MEMFR$
Random	13.88%	24.10%	10.18%
AADT-based	12.44%	18.56%	9.28%
CCS distance-based	4.30%	15.54%	8.25%
TMC coverage-based	19.39%	35.39%	13.44%

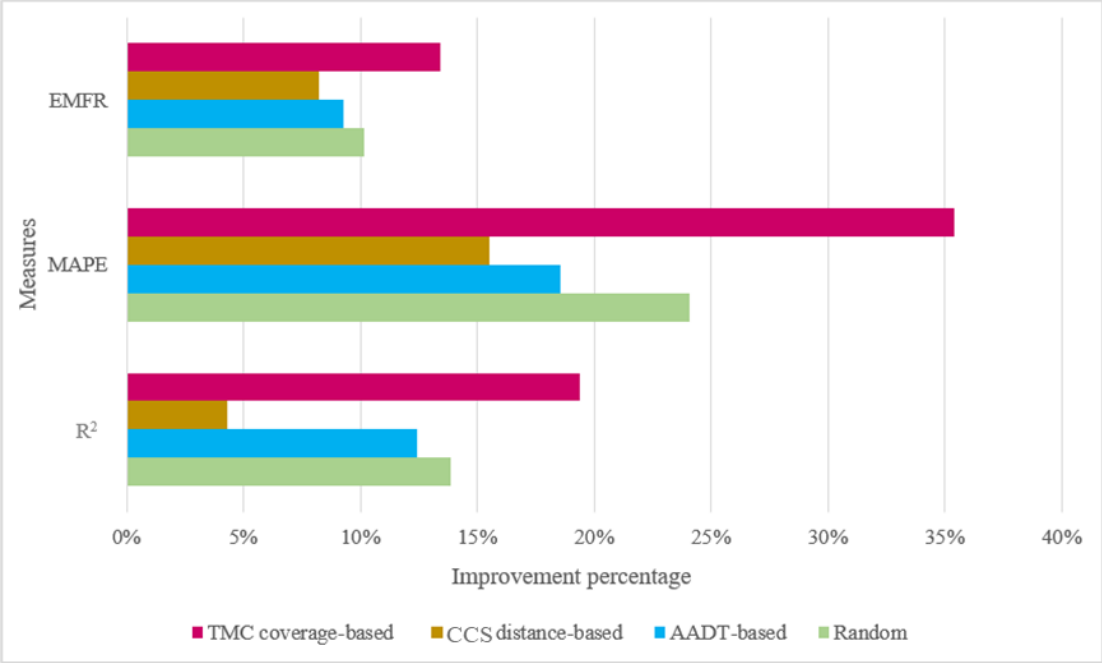


Figure 10. Relative improvements of the mean values for all the strategies

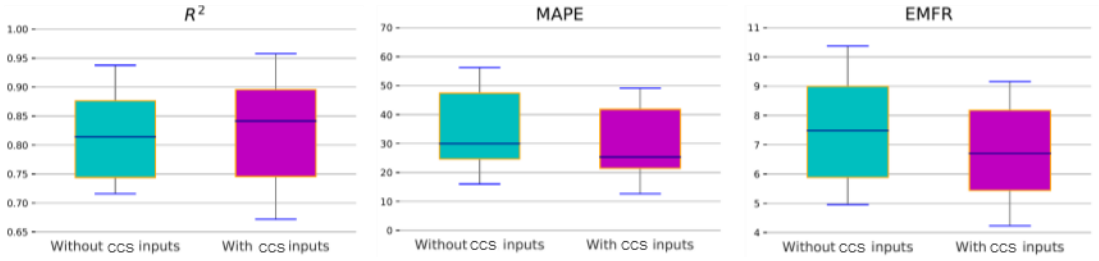


Figure 11. Error distribution without and with CCS inputs of TMC coverage-based strategy

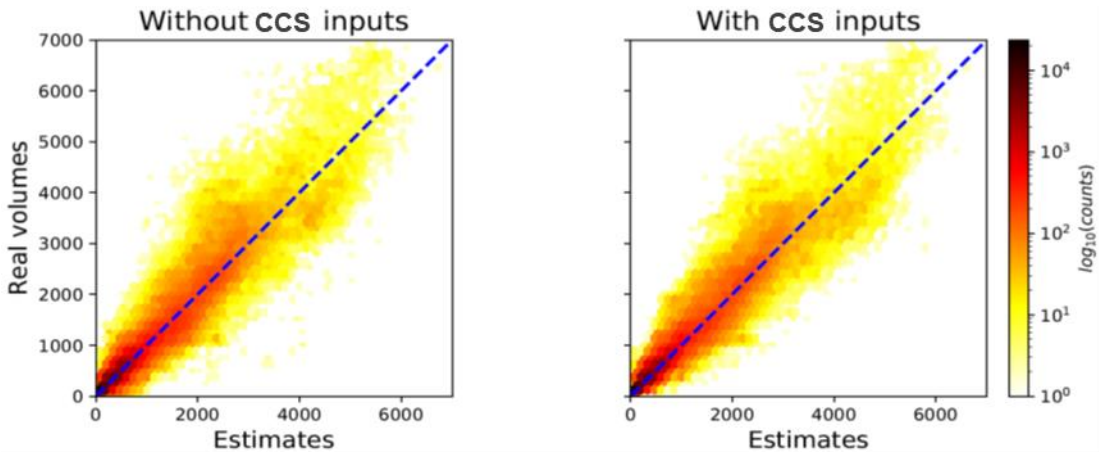


Figure 12. Heat maps of all data points used for testing the ANN models without and with CCS inputs of TMC coverage-based strategy.

A close look at this figure reveals that the estimates are more centered along the 45° line, and the number of outliers is reduced (a smaller number of scattered yellow dots). This indicates that not only does the model with CCS inputs provide a more accurate estimation of the volume counts in general, but it also improves the estimation in particular links where the base ANN model highly overestimates (or underestimates) the hourly traffic volumes.

3.6 Chapter Summary

In summary, this chapter reveals that incorporating the data of some CCSs as an input variable into the hourly traffic volume estimation model can significantly improve its performance if the input CCSs have been selected based on a reasonable strategy. This strategy can be as straightforward as TMC coverage-based strategy introduced here; however, more advanced strategies may be developed given a more comprehensive dataset – for example, perhaps incorporating road class into the optimization.

More importantly, the proof-of-concept model illustrated the essentiality of accounting for the dependencies in the traffic state of the network's links. The considerable estimation accuracy improvement in the proof-of-concept model, which is very limited in incorporating the traffic volume dependencies between road network links, motivates the direct incorporation of the road network graph structure. Therefore, the results and the improvement in estimation accuracy are the main incentives for proposing the graph-based model, where the road network graph will be a part of the regression model. In this way, any spatial correlation between the traffic volume of different road network segments will be captured through training a single model resulting in an elegant and accurate modeling framework.

Chapter 4: Proposed Framework

4.1 Overview

The previous chapter introduced a two-step methodology to prove how adding data from only a few links as input variables into the model can improve the accuracy of volume estimation in a road network. The results obtained from this methodology confirm that even indirect incorporation of the road network graph structure into the traffic estimation model improves its performance. Given these findings, the current chapter aims to propose a graph-based model that directly combines the traffic pattern's graph structure with the deep learning regression model to estimate traffic volumes. In the following sections, we first discuss the mathematical formula of the proposed model in section *4.2*. Then we introduce the novel graph generation method developed for this research in section *4.3*. In the end, the structure of the proposed model and its training process are described in section *4.4*.

4.2 Mathematical Formulation

The problem of estimating networkwide traffic volumes where ground truth data is only accessible for a small set of roads can be framed as graph-based semi-supervised learning. Kipf & Welling (2017) suggested the following loss function to smooth the available data over the graph using graph Laplacian regularization:

$$l = l_0 + \lambda l_{reg}\ , \qquad l_{reg} = \sum_{i,j} A_{ij} \left\| f(X_i) f\left(X_j\right) \right\|^2 = f(X)^T \Delta f(X) \tag{13}$$

Here, l_0 is the loss of the labeled part of the graph, $f(.)$ is a differentiable function like a neural network, λ is a weighing factor, and X is the matrix of node features X_i. $\Delta = D - A$ is the graph Laplacian of an undirected graph with an adjacency matrix $A \in R^{N \times N}$ and degree matrix D.

This formulation assumes that connected nodes probably share the same label, which may restrict the model capacity. Therefore, Kipf & Welling (2017) encoded the graph structure directly using a neural network model $f(X, A)$. This model is a multi-layer Graph Convolutional Network (GCN) with the following propagation rule:

$$H^{(l+1)} = \sigma\left(\widetilde{D}^{-\frac{1}{2}} \tilde{A} \widetilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)}\right) \tag{14}$$

where $\tilde{A} = A + I_N$ is the adjacency matrix with added self-connections that belongs to the undirected graph g, $\widetilde{D}$ is the degree matrix of $\tilde{A}$. W^l is the weight matrix in layer l, $\sigma(.)$ is an activation function and $H^{(l)} \in R^{N \times D}$ is the matrix of activation in layer l.

Using this propagation rule with rectified linear unit (ReLU) activation function and defining $\hat{A} = \widetilde{D}^{-\frac{1}{2}} \tilde{A} \widetilde{D}^{-\frac{1}{2}}$, their two-layer GCN model takes the following form:

$$\hat{y} = f(X, A) = softmax\left(\hat{A}\, Relu\left(\hat{A} X W^{(0)}\right) W^{(1)}\right) \tag{15}$$

This study uses the basic ideas of the propagation rule presented in equation (14) to develop a networkwide traffic volume estimation model. To do so, we first need to define the adjacency matrix to represent the traffic volume pattern correlations in the road network. Then we need to develop the model structure based on the research

objective, which is the Spatio-temporal traffic volume estimation. The following two sections elaborate on the methodologies developed to address these tasks.

4.3 Graph Generation

The geometry of a road network forms a graph per se; however, this graph cannot be efficiently used in a GNN model to solve the traffic volume estimation problem. There are two main reasons for this. First, most of the efficient GNN models are designed for node-level regression on undirected graphs (Wu et al., 2020), while roads are directed edges of the network. Besides, the physical connections and Euclidian distances in a road network do not represent the actual dependency between the links. A straightforward way of generating the representative graph, used in previous studies, is to put a corresponding node for each road and connect them based on their distance (Yu et al., 2018). However, using only the geometry of the road networks is not enough to indicate traffic volume correlations. Figure 13 presents an example to clarify this point. According to this figure, despite Link 2 and Link 3 being closer geometrically, they have significantly different traffic flow patterns compared with Link 1 and Link 3. In reality, the trip patterns in the networks determine the traffic volume correlations between links. Therefore, in this study, a graph generation method based on trip patterns is proposed.

The proposed graph generation method involves using probe vehicles' waypoints to extract trip patterns in a transportation network. Since these vehicles are a sample of the total vehicles in the network, they can be used to generate a weighted graph reflecting the flow dependencies in the network. The graph generation algorithm in this

study yields three representative graphs corresponding to three time periods of morning peak, afternoon peak, and off-peak hours.

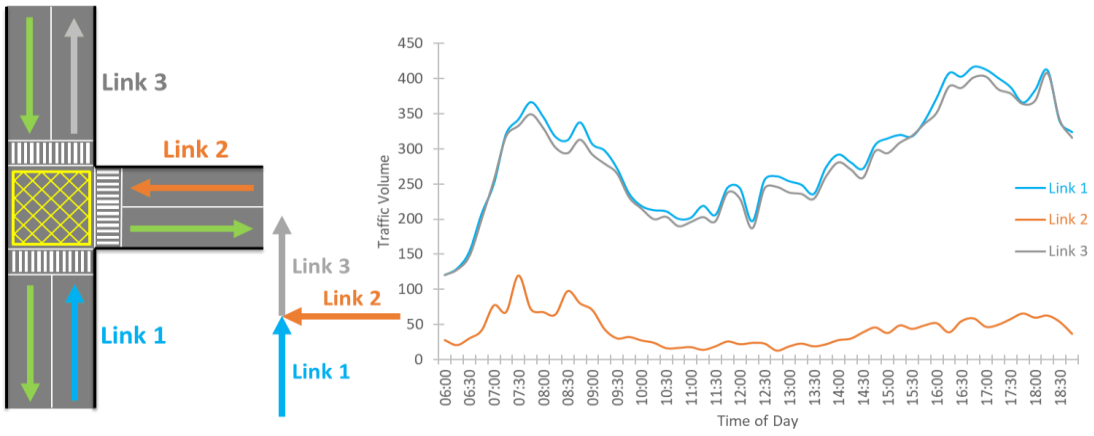


Figure 13. An example to show how road network geometry is not an appropriate indicator of traffic volume correlations.

The graph generation algorithm

1. For each probe trip, if the trip is connected, go to step 3; otherwise, go to step 2.
2. Make each probe trip connected by finding the shortest path between any two consecutive disconnected links in the trip.
3. Compute f_i^t as the total number of probe waypoints passing each road segment i in the network during each time interval t.
4. For each two road segments i and j, compute C_{ij}^t as the total number of probe waypoints that are common between the two roads (i.e., segments i and j are part of the same trip) during time interval t.
5. For each road segment i in the network, create a corresponding node called i.

6. If road segments i and j are connected in the road network, connect their corresponding nodes with a link whose weight is computed based on the following formula:

$$w_{ij}^{t} = \frac{C_{ij}^{t}}{0.5 \times \left(f_i^t + f_j^t\right)} \tag{16}$$

7. Once the weights are computed for all connections and time intervals, aggregate them over the three-time periods of the morning peak, afternoon peak, and off-peak hours by averaging the weights according to the following formula:

$$w_{ij}^{P} = \frac{1}{|S_P|} \sum_{t \in S_P} w_{ij}^{t} \quad \forall P \in \{AM - Peak, PM - Peak, Off - Peak\}, \qquad S_P = \{t | t \in P\} \tag{17}$$

8. Put the aggregated weights together to build the symmetric weighted adjacency matrices as presented in equation (18).

$$A_w^P = \begin{bmatrix} 0 & \cdots & w_{1n}^P \\ \vdots & \ddots & \vdots \\ w_{n1}^P & \cdots & 0 \end{bmatrix}, w_{ij}^P = w_{ji}^P \tag{18}$$

where n is the number of connections generated by the algorithm.

The intuition behind this graph generation algorithm is that two connected links of a road network are more correlated when they are part of the same path. Additionally, in reality, the links' flow correlations in each time interval of the morning peak, afternoon peak, and off-peak hours remain relatively the same from one day to another. Therefore, the last steps of the graph generation algorithm (steps 7 and 8) aggregate the time-dependent weights to get robust weighted adjacency matrices for each time interval. To better understand how this algorithm builds the representative graph with

additional information about the traffic condition, an illustrative example is presented in Figure 14.

In this example, for a specific time interval, a total of 10 trips are generated in the only origin of the network, O, 5 units of which are going to destination 1, D_1, using the connected path of $1 \rightarrow 2 \rightarrow 3$, and the remaining 5 are going to destination 2, D_2, using the connected path of $1 \rightarrow 2 \rightarrow 4$. Therefore, the values for f_i^t and C_{ij}^t are as presented in Table 6. Given this information, we can build a representative graph as represented in Figure 14.

The weights in this graph are computed as follows:

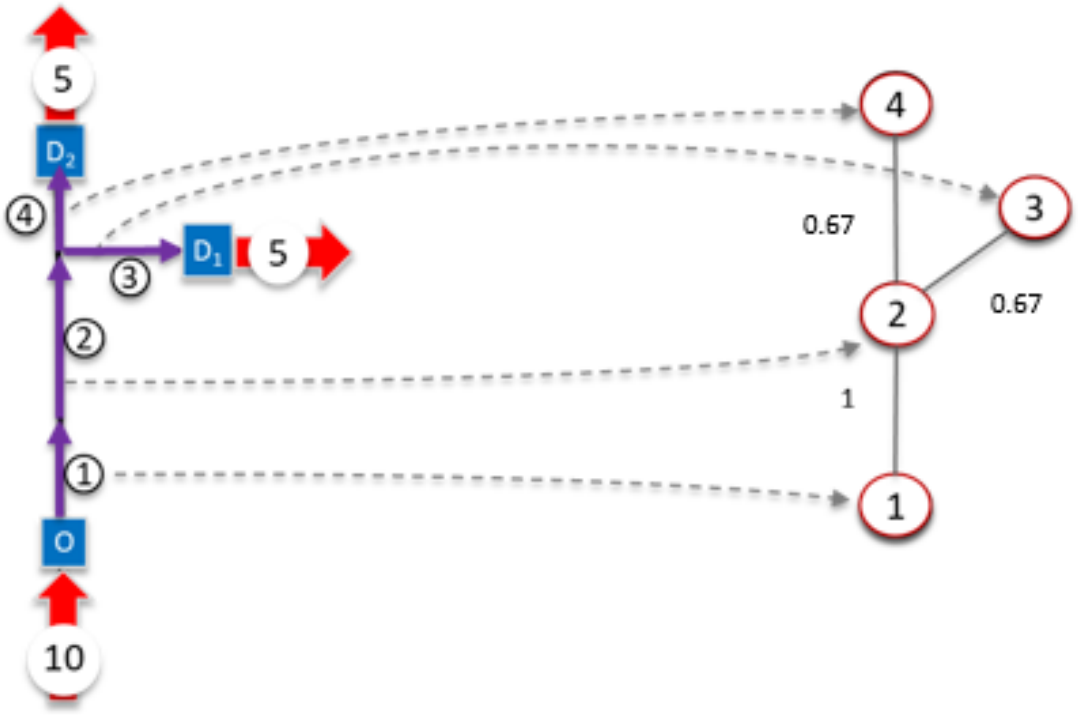


Figure 14. Graph generation example

Table 6. Computed flows for the example network.

Flow at each link				
i	1	2	3	4
f_i^t	10	10	5	5
The common flow between two connected links				
ij	1-2	2-3	2-4	3-4
C_{ij}^t	10	5	5	0

$$w_{12}^t = \frac{10}{0.5 \times (10 + 10)} = 1$$

$$w_{23}^t = \frac{5}{0.5 \times (10 + 5)} = 0.67$$

$$w_{24}^t = \frac{5}{0.5 \times (10 + 5)} = 0.67$$

$$w_{34}^t = \frac{0}{0.5 \times (10 + 10)} = 0$$

The presented example shows how the weights are calculated for a specific time interval. Once these weights are computed at different times and days, we will aggregate them to build static weighted adjacency matrices for each three time periods.

4.4 Model Structure

This section introduces an innovative model structure and training process that uses the basic mathematical formulation of the graph convolutional network described in section *4.2* and expands it to suit the traffic volume estimation problem. One of the main components of this model is the static graph adjacency matrix generated according to the algorithm introduced in section *4.3*. Given that the graph structure

accounts for the spatial correlation in the road network, the next challenge is to consider the dynamic characteristics of the traffic state to capture the temporal correlations.

For capturing the temporal correlations, the current study introduces a temporally dynamic GCN-based framework whose schematic architecture is illustrated in Figure 15. This framework is constructed from two main blocks. The left block, we name it the Spatio-Temporal GCN (STGCN) model, is a three-layer GCN trained using the entire data available for a time period (e.g., AM-Peak). Although the graph structure (i.e., the adjacency matrix) is static for a specific time period like morning peak, the node features dynamically change over the time intervals that belong to that period. Therefore, the input data of the STGCN model is a static graph whose features are changing dynamically. For instance, if we have data of an entire year, and the objective is to estimate networkwide traffic volumes for 15-minute time intervals in the morning peak, the input data to the STGCN model is the data of all the 15-minute time intervals in morning peak over the entire year. For each time interval, the inputs are the features such as probe speed, probe counts, road characteristics, temporal variables for all nodes (i.e., road network links), and ground-truth labels for a few nodes (i.e., links where CCSs are located).

The STGCN model is designed to capture the spatial and temporal variations in the network and helps the model learn the correlations between the traffic volumes and input features. The output of the left block is the STGCN model weights that are the initial weights of the Fine-tuned STGCN model (FSTGCN) in the right block. As shown in the figure, the fine-tuned model is designed to fine-tune the base model for each specific time interval. In other words, the STGCN model is fine-tuned for each

particular time interval to pay more attention to the state of the network in that time interval and captures the spatial correlations. To better understand how the introduced model works, Figure 16 illustrates the training process flowchart.

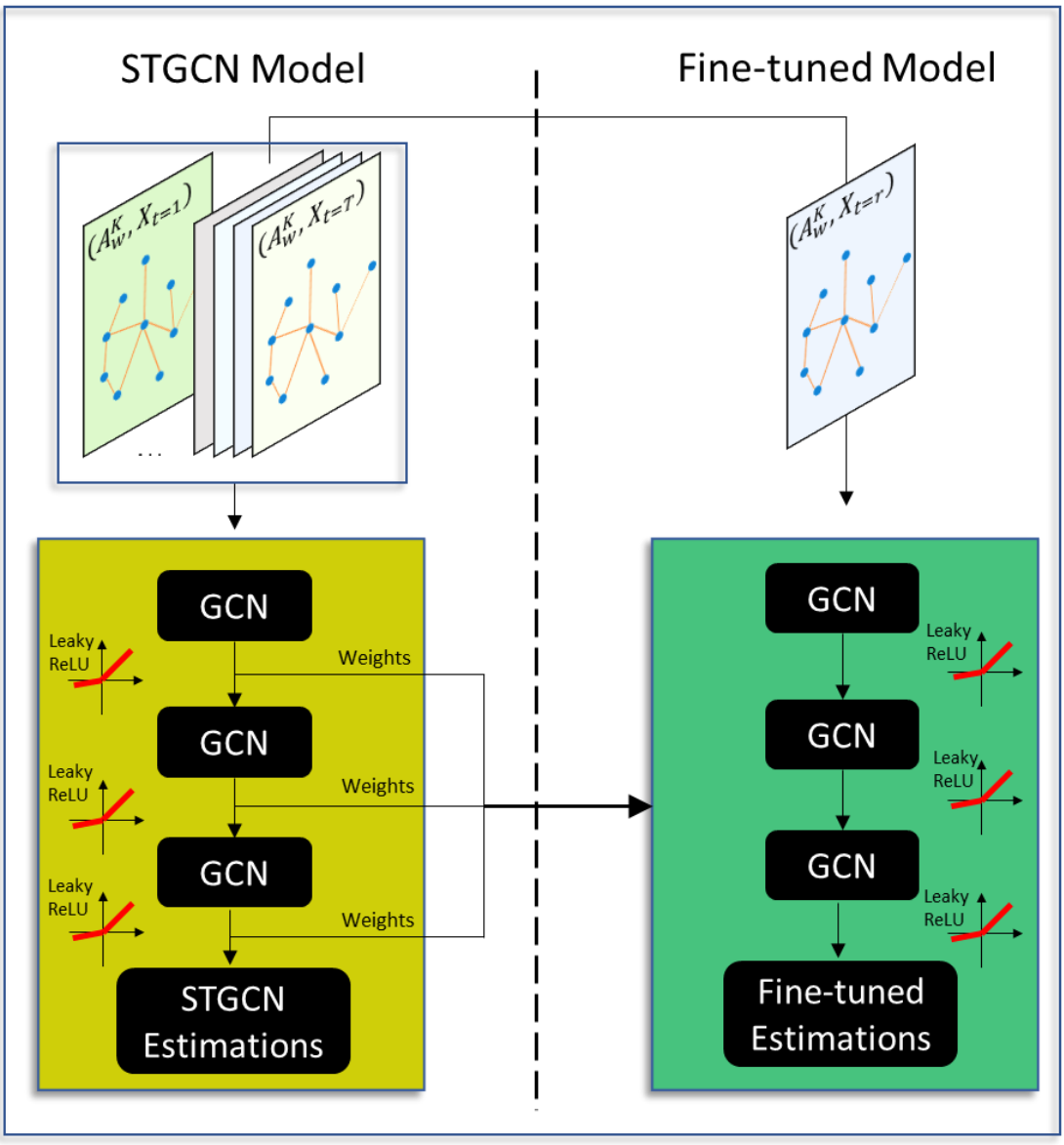


Figure 15. The schematic architecture of the introduced FSTGCN model.

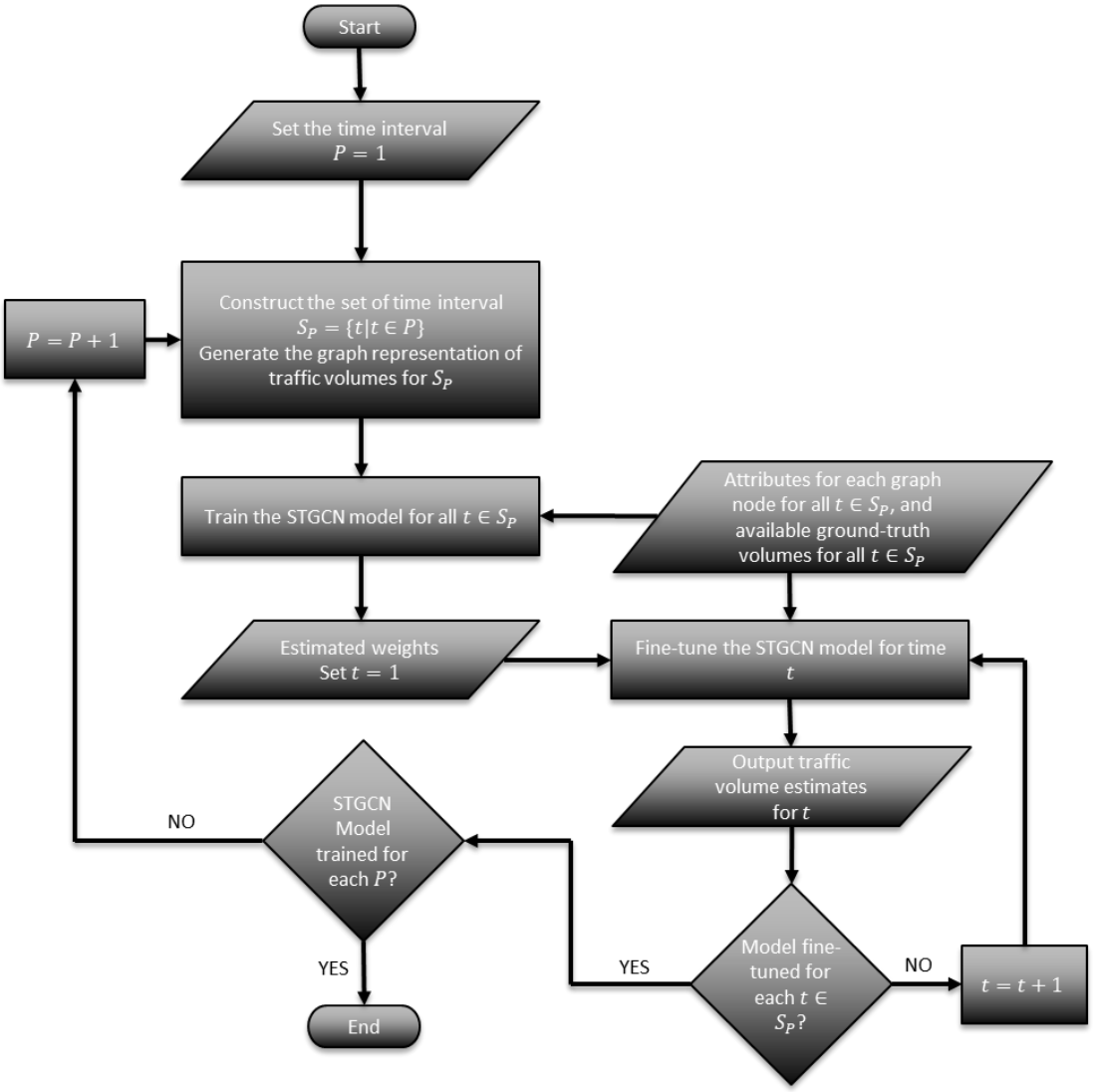


Figure 16. Training process flowchart.

4.5 Chapter Summary

In this chapter, the modeling framework in which the dependencies of link traffic flows are accounted for is introduced. As described in chapter two, the major gap in most of the previous studies is overlooking the interactions between traffic volume in different road segments. The interactions arise from the system users' route selection and should be incorporated in the modeling framework to produce robust traffic flow estimates.

The proposed approach in this study directly incorporates the traffic flow dependencies between road network links in its structure.

In the first section of this chapter, the mathematical formulation of the GCN model is presented and discussed. The GCN model is a graph-based semi-supervised deep learning model capable of estimating the labels of all nodes in a graph given the attributes of all nodes and the ground truth labels for some of the nodes. The mathematical formulation of the GCN model is followed by introducing the methodology for generating the graph representation of the road network traffic volumes. The proposed graph generation technique considers both the geometry of the road network and trip patterns to extract the relations between traffic volumes in different links. The traffic volumes utilized for generating the graph are the probe vehicle waypoints data, a sample of the entire vehicles traversing the roads. Finally, in the last section, the introduced GCN model and the graph generation technique are combined, and the graph-based model of FSTGCN is introduced for network-wide traffic volume estimation.

Chapter 5: Data

5.1 Overview

The proposed approach aims to estimate the 15-minutes traffic volumes by capturing the spatio-temporal dependencies between the traffic volumes in different segments of a road network besides learning the relations between traffic volume in a link and the attributes of that link. The introduced framework in the previous chapter is applied to the NPMRDS network of various areas in the state of Maryland using the 2019 data. The results of training and testing the model illustrate the proposed framework's performance in traffic flow estimation. In this chapter, the Maryland NPMRDS network is introduced, along with a brief descriptive analysis of the data used for numerical analysis of the presented framework.

5.2 Study Area

The focus for illustrating the performance of the proposed framework is on estimating the traffic volumes in the NPMRDS road network of various areas in the state of Maryland. The NPMRDS is a national database of probe vehicle-based speed and travel time data with free access for transportation authorities and agencies. The NPMRDS data is available across the national highway system (NHS) and has a spatial resolution based on Traffic Message Channel (TMC) location codes. The selection of the NPMRDS network is due to the availability of geometric and functional characteristics

of the segments in this network. The NPMRDS network in Maryland comprises about 4,430 miles of highways and interstate freeways with concentrations of road network in and around the urban regions of Washington DC and Baltimore. The NPMRDS network in the state of Maryland is shown in Figure 17. There are 45 CCSs on this network collecting traffic counts throughout the year. The location of the TMCs with installed CCSs is also illustrated in Figure 17 in yellow. As it can be seen, these stations are distributed throughout the entire network.

Since different regions of the Maryland NPMRDS network have different characteristics, three areas of this network are considered separately to investigate the proposed framework. These regions are Eastern Maryland, the Beltway area, and Western Maryland. The Eastern Maryland region incorporates Wicomico and Worcester counties, and the Western Maryland region comprises Garret and Allegany counties. These two regions typically have a low congestion level, with speeds close to the free flow speed. However, the difference between the Eastern and Western networks is the presence of different road classes in the Eastern region. The Beltway area includes all the TMC segments of the NPMRDS Maryland inside the I-495 Capital Beltway and the I-495 TMC segments. This area has a congested road network with high variations in speed profile throughout the day. The location of these three regions is illustrated in Figure 18.

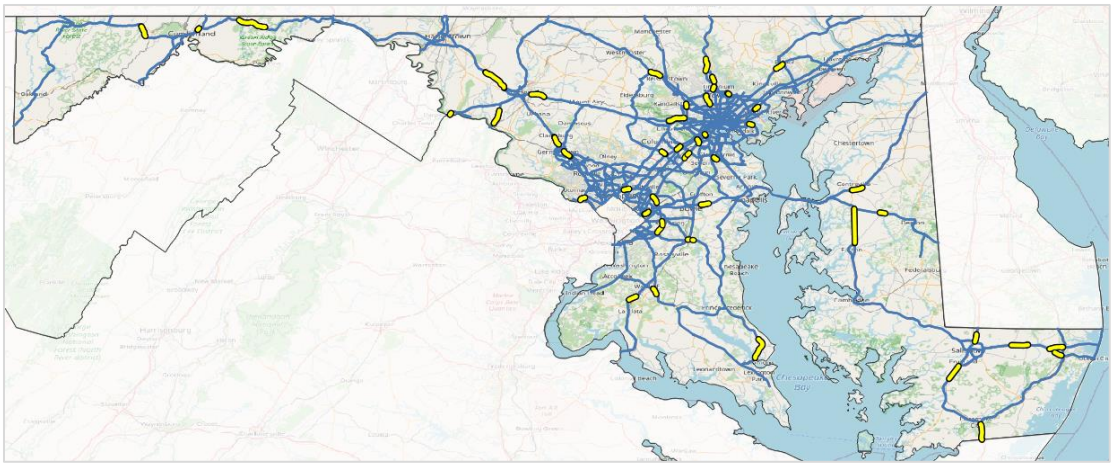


Figure 17. Maryland NPMRDS network

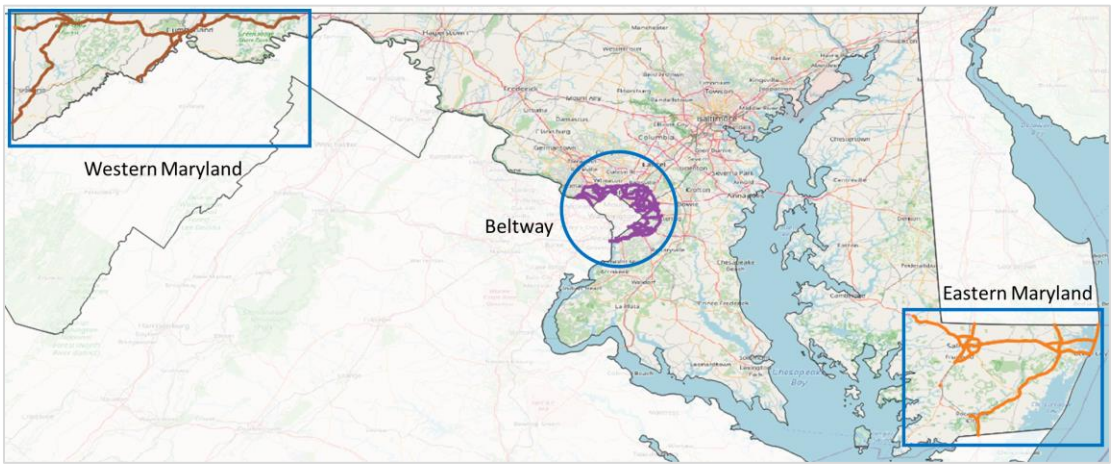


Figure 18. Study NPMRDS Maryland regions

5.3 Conflation of NPMRDS Network Attributes to OSM Network

The graph generation procedure introduced in this study requires a connected road network. However, the NPMRDS network, being a high-level performance-oriented definition, is not a connected network thus not appropriate for generating the traffic volumes graph representation. On the other hand, the OSM road network is a detailed map of the road network satisfying the connectivity requirement, thus suitable for the graph generation task. The downside of revering to the OSM network is that the

geometric and performance attributes are not reported for the OSM segments. Therefore, a mapping procedure is needed between the road links in the NPMRDS and OSM networks to transfer data attributes between these base maps. The conflation process consists of two high-level steps of setting up a crosswalk and data conflation. First, a list of matched segments from the NPMRDS network is generated for each segment in the OSM network in the crosswalk step. The attributes are linked from the NPMRDS network to the OSM network in the second step. Since the OSM network has much more granular segments, an NPMRDS TMS is often linked to many OSM segments. For cases where more than one TMC is associated with an OSM segment, the attributes of the TMC with the highest coverage in length are linked to the OSM segment.

5.4 Probe Vehicle Data

The procedure for generating the graph representation of the traffic flows, as discussed in 4.3, is based on the probe vehicle movements in the road network. The probe vehicle data is obtained from INRIX, one of the most renowned data vendors for transportation agencies. The INRIX data includes the records of vehicle waypoints in 2019 snapped to an OSM-based map modified by INRIX. The timestamps on which the waypoints are recorded are around 40 seconds on average. However, this value can vary significantly among trips and can be as much as several minutes. This difference between the timestamps means that the chain of traversed segments according to the raw data is not a connected route. Therefore, for each discontinuity, the shortest path between the traversed links is computed, and the links in this path are added to the

chain of links in a given trip. For instance, in Figure 19, a trip has records in segments 1, 2, 3, and 6, while there are no records in segments 4 and 5. Therefore, in the data preparation step, the missing segments should also be added to the chain of segments to form a connected trip.

After all the trips in the database have a connected chain of segments, the number of trips that pass each segment in each time period can be aggregated to compute the adjacency matrices.

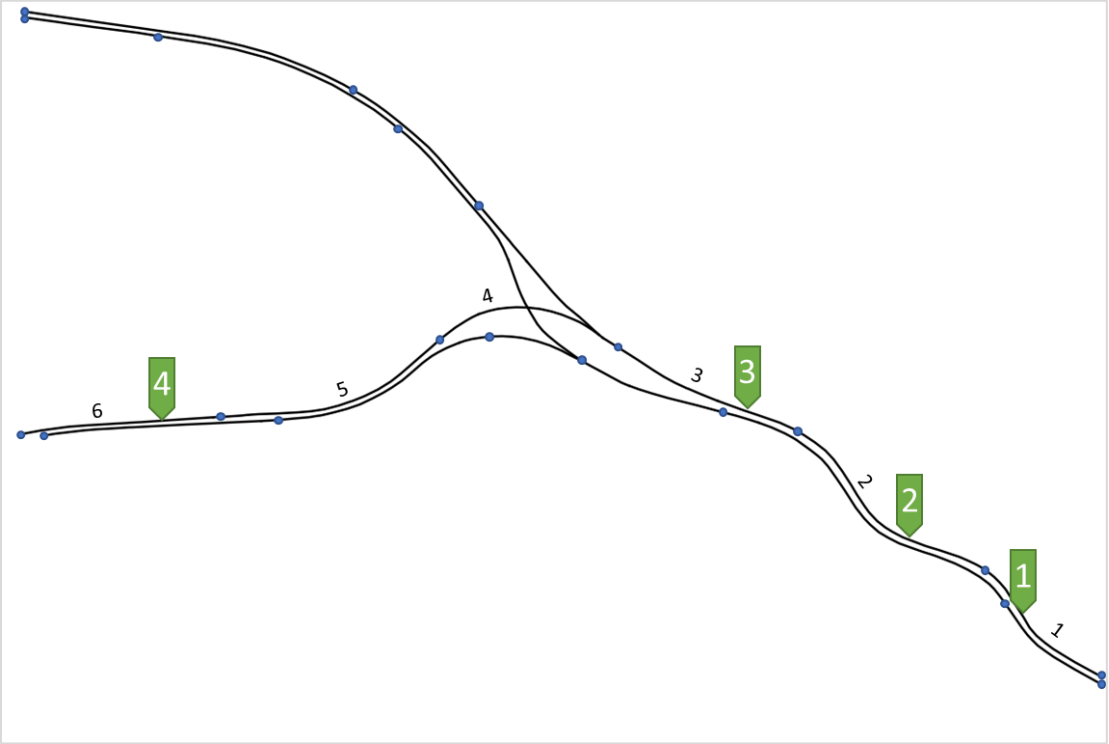


Figure 19. INRIX probe vehicle data discontinuity

5.5 Input Features

In this section, the data used for training and testing the model is briefly described. This descriptive analysis helps the readers obtain a broad perspective of the data and traffic volume characteristics in different regions of the Maryland NPMRDS network. The proposed framework requires the attributes for each road segment included in the model inputs. The features used in training the model are presented in Table 7.

Table 7. Input attributes for the proposed model

Variable	Details	Type
CCS data	traffic volume counts	Continuous
Probe vehicle speed	speed, average speed, reference speed	Continuous
Probe vehicle Count		Continuous
Weather data	temperature, precipitation,	Categorical
Infrastructure data	number of lanes, speed limits, class of the road (motorway or trunk), and type of the road (Interstate, US road, or MD road)	Categorical
Temporal data	The quarter of the hour, The hour of the day, The day of the week, The month of the year	Categorical

The distribution of traffic flow counts and speed for each of the three study regions are presented in Figure 20. As it can be observed in this figure, expectedly, the Eastern and Western Maryland regions roads carry less amount of traffic than the Beltway area roads. There are numerous instances where traffic flow exceeded 1,500 vehicles per

hour per lane in the Beltway area, illustrating that the segments are operating at or near their capacity. In the Eastern Maryland region, the speed shows two distinct congestion levels. Most of the time, there is no congestion, and speeds are close to the free flow speed; however, there are periods that the roads are congested, and the speeds are much lower than the free-flow speed. The congested durations in Eastern Maryland are observable during national holidays and summers.

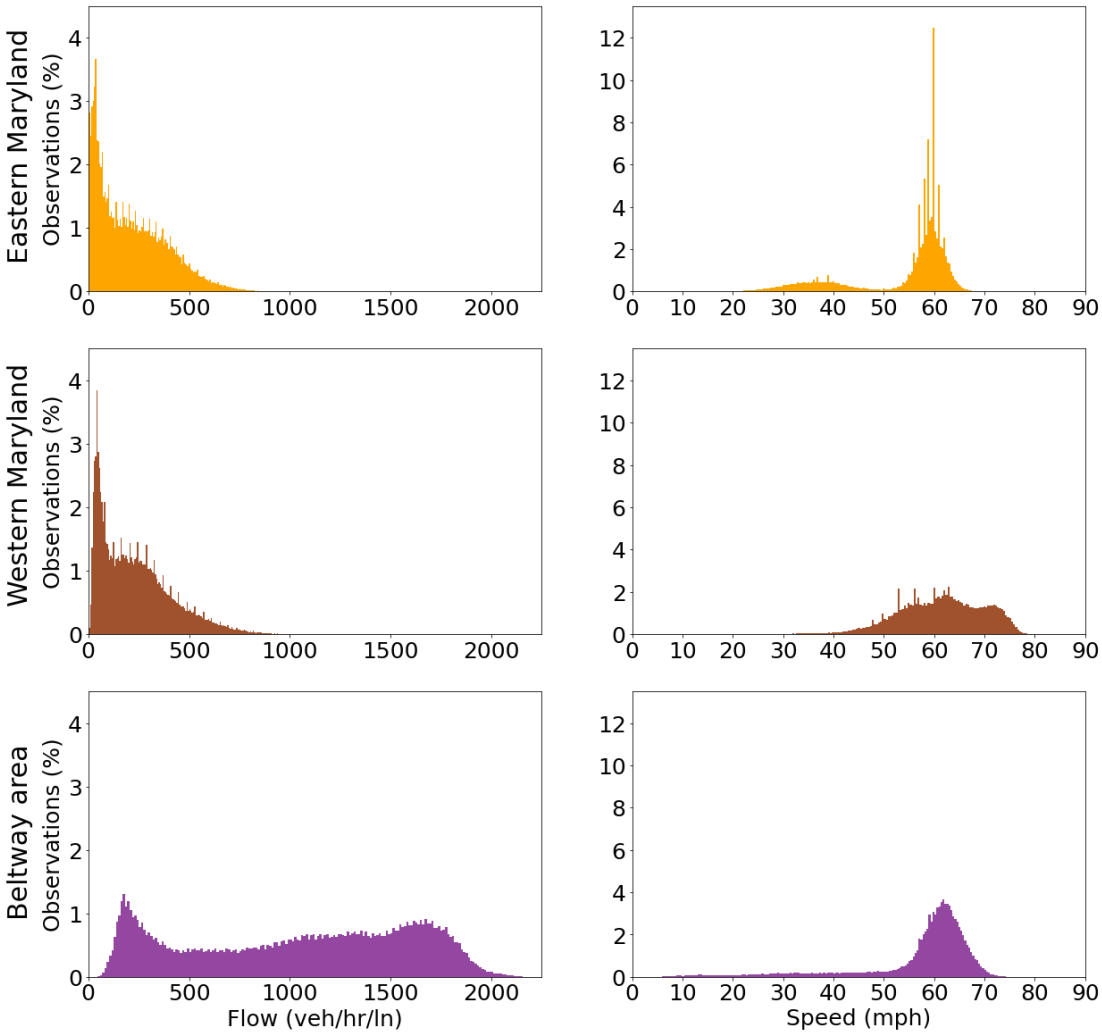


Figure 20. Distribution of flow and speed in Eastern Maryland, Western Maryland, and Beltway area

In Western Maryland, the speeds show variations around the free flow speeds since this region is free from recurring congestion. On the contrary, the Beltway area experiences recurring congestion in many of its segments; thus, the traffic speeds illustrate a significant amount of variability. The summary of the input data is presented in Table 8. The numbers in this table demonstrate that the traffic pattern in the Beltway area is entirely different from that of the Eastern Maryland and Western Maryland regions.

Table 8. Summary of the input data

	Western Maryland	Beltway area	Eastern Maryland
NPMRDS network length (miles)	298.91	342.65	276.19
Total number of NPMRDS segments	397	748	232
Total number of OSM segments	879	5175	2051
Number of CCSs	3	6	6
CCS FRC	1	1, 2	2, 3
Average AADT on CCSs	23,800	183,700	21,450
Average number of observations for each CCS in 2019	28,543	34,581	33,527

5.6 Chapter Summary

This section introduces the networks and data used for testing and analyzing the proposed graph-based framework. It first discusses the steps to build the desirable input data using available datasets and maps of NPMRDS, OSM, and INRIX. Additionally, the three case study networks of Eastern Maryland, Western Maryland, and Beltway are introduced, and some of their high-level traffic characteristics are discussed.

Chapter 6: Experiments

6.1 Overview

The FSTGCN model introduced in chapter 4 is constructed from various components that require investigation. This section discusses three experiments designed to explore the graph-based framework before providing the final numerical results in the next chapter. The first experiment investigates the effects of training size on the model performance. The second experiment explores the improvement obtained by adding the fine-tuning step to the graph-based framework. Finally, the last experiment compares the performance of the FSTGCN using different loss functions. However, before going through these experiments, we first briefly describe the models and criteria used to evaluate the introduced framework in the rest of this study.

6.2 Evaluation Models and Criteria

As previously stated, the two advanced machine learning models of ANN and XGBoost are currently used for network-wide traffic volume estimation (Sekula et al., 2018; Yi et al., 2021). The ANN model was fully described in section 3.3. Here, we first briefly introduce the XGBoost model and then discuss how we compare the study framework with ANN and XGBoost.

6.2.1 XGBoost model

XGBoost is the short name for "Extreme Gradient Boosting," an efficient and scalable implementation of gradient boosting framework (Friedman et al., 2000). XGBoost is a cutting-edge application of gradient boosting machines and has proven to push the limits of computing power for boosted trees algorithms. It was developed to improve model performance and computational speed. Boosting is an ensemble technique in which new models are added to adjust the errors made by existing models. The new models are created that predict the residuals of prior models and then added together to make the final prediction. The objective function of the XGBoost algorithm comprises a loss function over the training set and a regularization term penalizing more complex trees to reduce the overfitting:

$$Obj = \sum_{i} L(y_i, \hat{y}_i) + \sum_{k} \Omega(f_k) \tag{19}$$

Where $L(y_i, \hat{y}_i)$ can be any convex differentiable loss function and $\Omega(f_k)$, the complexity term, is defined as:

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda w^2 \tag{20}$$

where T is the number of leaves of the tree f_k and w is the leaf weights. After taking the Taylor expansion and removing the constant terms, the objective function for iteration *m* is as follows:

$$Obj^m = \sum_{j=1}^{T} \left[G_j w_j + \frac{1}{2}\left(H_j + \lambda\right) w_j^2 \right] + \gamma T \tag{21}$$

where G_j and H_j are defined in (22):

$$G_j = \sum_{i \in I_j} \frac{dL\left(y_i, \hat{y}_i^{(m-1)}\right)}{d\hat{y}_i^{(m-1)}}, H_j = \sum_{i \in I_j} \frac{d^2 L\left(y_i, \hat{y}_i^{(m-1)}\right)}{d\left(\hat{y}_i^{(m-1)}\right)^2} \quad (22)$$

I_j is the set of training instances in leaf j.

The best leaf weight w_j given the current tree structure will be:

$$w_j = -\frac{G_j}{H_j + \lambda} \quad (23)$$

6.2.2 Model settings and comparison criteria

As described in chapter 5, the study area for evaluating the introduced framework of chapter 4 is different networks inside the state of Maryland. Therefore, the ground-truth data used for the FSTGCN model is limited to the CCSs within the case study networks. However, for the two models of ANN and XGBoost, we use the entire state CCSs' ground truth data for training. There are two reasons for this. First, unlike the FSTGCN model that takes the input features from all the links in the study network, ANN and XGBoost only need the input features from locations where the ground-truth volume data is available. Thus, the ANN and XGBoost models demand much smaller memory and processing power for training. As a result, there are no capacity limitations to use the entire state CCSs' data for these two models. Secondly, one of the study's objectives is to independently evaluate the introduced FSTGCN model in different locations. Therefore, we limit both input features and ground-truth data to the study network for this model, which can be any of the Eastern Maryland, Beltway area, or Western Maryland NPMRDS networks introduced in chapter 5. Note that in this way, we are favoring the ANN and XGBoost model in terms of the given information since it is the

ground-truth volume data that is limited, and machine learning models almost always perform better with access to more ground-truth data.

At the same time, regardless of the model we are using, the input features must be available for any link that traffic volume is estimated for it. To understand this more clearly, the first experiment of this section is designed to investigate the ground truth data size used for training on the FSTGCN model performance.

Eastern Maryland is the network used for conducting the experiments in this chapter. The evaluation criteria are the two famous metrics of Absolute Prediction Error (APE), and Error to Maximum Flow Ratio (EMFR) defined according to Equations (24) and (25).

$$APE_i^t = \frac{|y_i^t - \hat{y}_i^t|}{y_i^t} * 100 \tag{24}$$

$$EMFR_i^t = \frac{|y_i^t - \hat{y}_i^t|}{y_{i,max}} * 100 \tag{25}$$

where y_i^t is the ground-truth and $\hat{y}_i^t$ is the estimated traffic volume in link i during time interval t, and $y_{i,max}$ is the maximum ground-truth traffic volume in link i.

One other point worth mentioning about the results presented in this chapter and the following one is that we are using full cross-validation for all models. Therefore, each time one CCSs' data of the study area is left out and the rest are used for training. This procedure is repeated until we test the model on all CCSs in the study area. The training process flowchart provided in Figure 16 is updated in Figure 21 to reflect the cross-validation step. In this figure, A represents the set of CCS stations in the study network

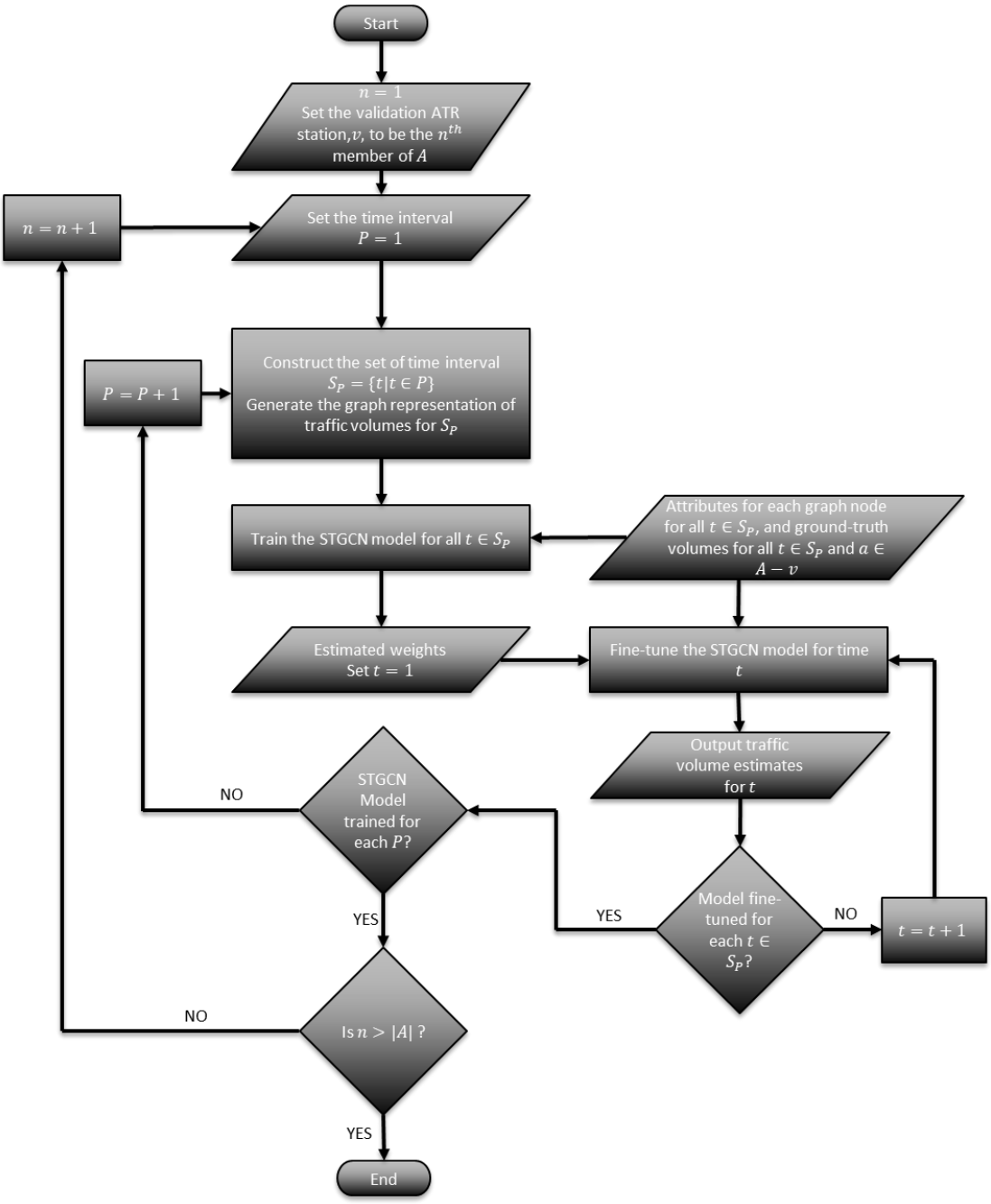


Figure 21. FSTGCN flow chart with cross-validation.

and all other variables are defined previously. The data used for training and testing is the entire 2019 data which means S_P is the set of all time intervals in period p throughout the year 2019.

Note that cross-validation leads to the same testing set for all models. However, based on what we discussed here, the number of CCSs used to train ANN and XGBoost is $45 - 1 = 44$ stations, as we have a total of 45 CCSs in Maryland. However, for the FSTGCN model, this value depends on the number of CCSs falling inside the study area.

Finally, the main hyperparameters used for the FSTGCN model are as follows:

- Each GCN layer is followed by the LeakyRelu activation function with the following formulation:

$$f(x) = \begin{cases} x & \forall x: 0 \leq x \\ 0.1x & \forall x: x < 0 \end{cases} \quad (26)$$

- There are dropout layers with a dropout rate set to 0.5 after the first two GCN layers.
- The output dimension is 256,128,1 for the three GCN layers in order.
- AdaM optimizer with a learning rate equal to 0.001 is used for optimization.

6.3 Experiment 1: Training ground-truth data size

As noted earlier in this chapter, we use much less ground-truth data to train the FSTGCN model than what is used to train the ANN and XGBoost models. Intuitively, using less amount of training data negatively affects the FSTGCN model performance. This section investigates this effect by training the FSTGCN model using two sets of CCSs on the Eastern Maryland network. The objective is to estimate traffic volume for Worcester county NPMRDS segments. This network, highlighted in Figure 22, itself has three CCSs (i.e., Stations 1, 2, and 3). The adjacent county of Wicomico, shown in gray in Figure 22, also has three CCSs (i.e., Stations 4, 5, and 6).

Here we first train a model only using the data and network of Worcester county. We refer to this model as the "Base Network" model. Then we add the network and data of

the adjacent county to the study area to see how using more training data improves the estimates on the Worcester county network. We refer to this as the "Expanded Network" model. Figure 23 shows the distribution of APE and $EMFR$ for these models compared to ANN and XGBoost. According to this figure, the Expanded Network model provides significantly better estimates than the Base Network model using more CCS data. However, even the Base Network FSTGCN model outperforms both ANN and XGBoost despite using much fewer CCSs ground-truth data for training (i.e., two stations vs. 44 stations).

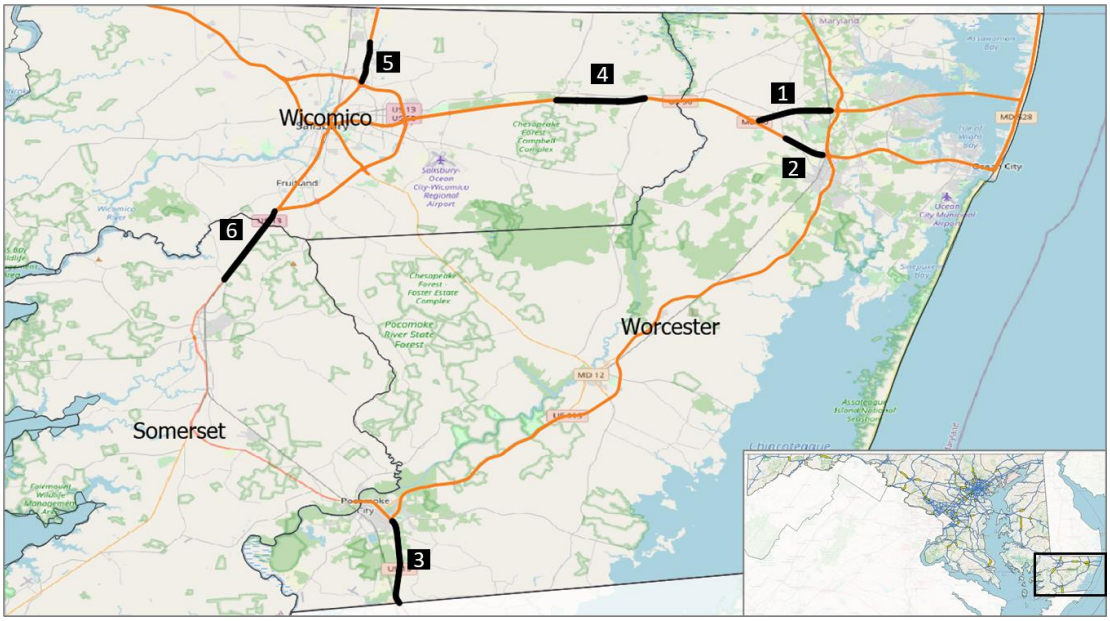


Figure 22. Eastern Maryland Network and locations of its CCSs.

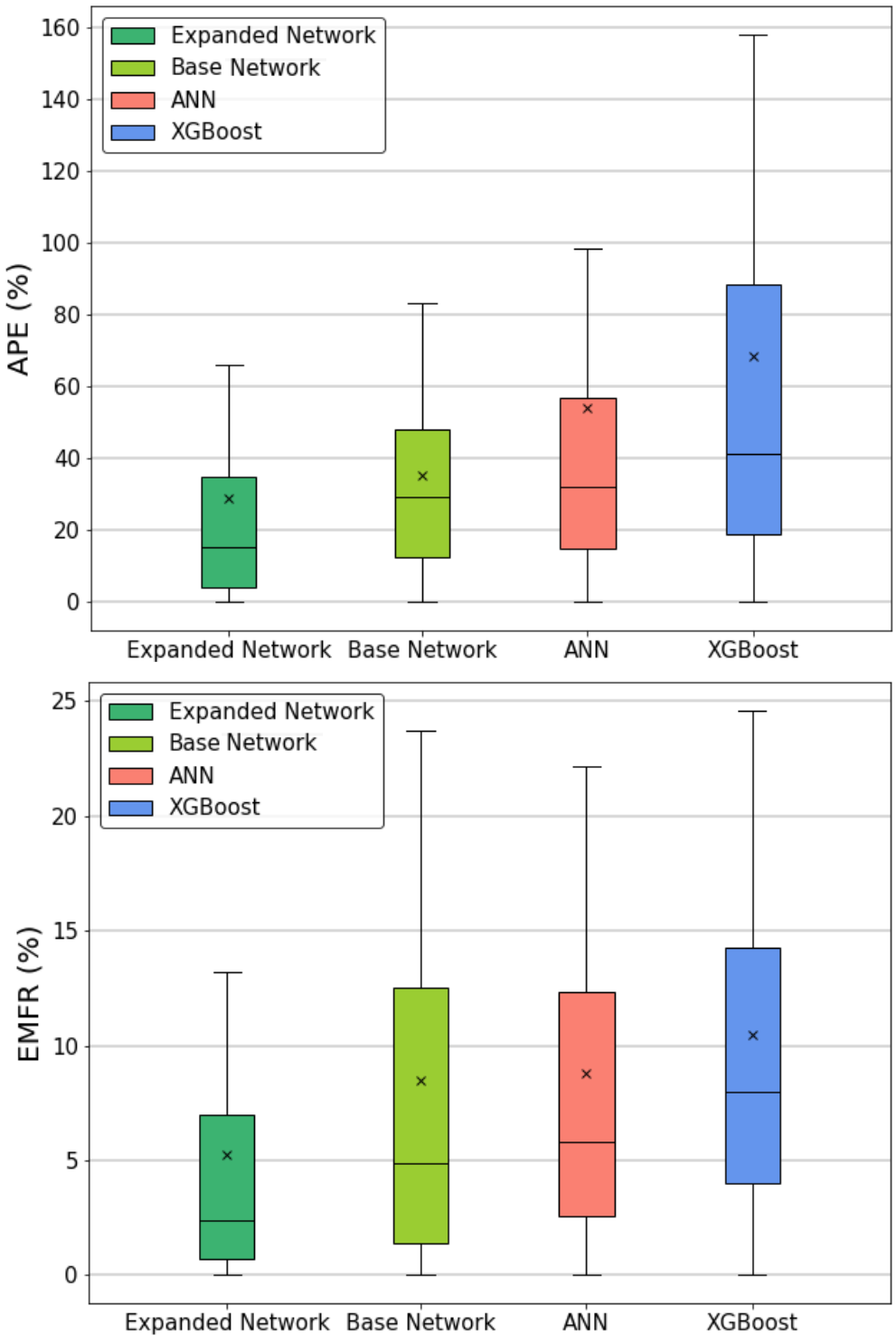


Figure 23. APE and EMFR distribution to investigate ground-truth data size effects.

6.4 Experiment 2: Fine-tuned model gain

As discussed in section 4.4, the FSTGCN framework introduced in this study is constructed from two primary components of the "STGCN model" and "Fine-tuned model." While in this framework, we only use the weights of the STGCN model as an input to the Fine-tuned model, we can use the STGCN model to get initial estimations of networkwide traffic volumes. This section compares the accuracy of such initial estimations with the final output of the introduced framework to investigate the benefits of fine-tuning for any time interval that we want to estimate networkwide traffic flows for it.

Here, we use the data and network of Eastern Maryland, illustrated in Figure 18. We first estimate traffic volumes using only the first part of the introduced framework, the STGCN model. Then, we compare it with the traffic volumes estimated using the entire framework and going through the fine-tuning process, i.e., the FSTGCN model. We also compare these estimations with those of ANN and XGBoost models. Same as before, the two metrics of APE and EMFR are used to compare models' performance. Figure 24 presents the distribution of APE and EMFR to compare the performance of the FSTGCN model with the STGCN, ANN, and XGBoost. According to this figure, the STGCN model itself has better performance than the two other state-of-the-art models of ANN and XGBoost. However, this performance is significantly improved by introducing the fine-tuning step to the study graph-based model. This suggests that the STGCN model itself can extract the relation between traffic volumes and the input features and improve the estimation accuracy compared to ANN and XGBoost using

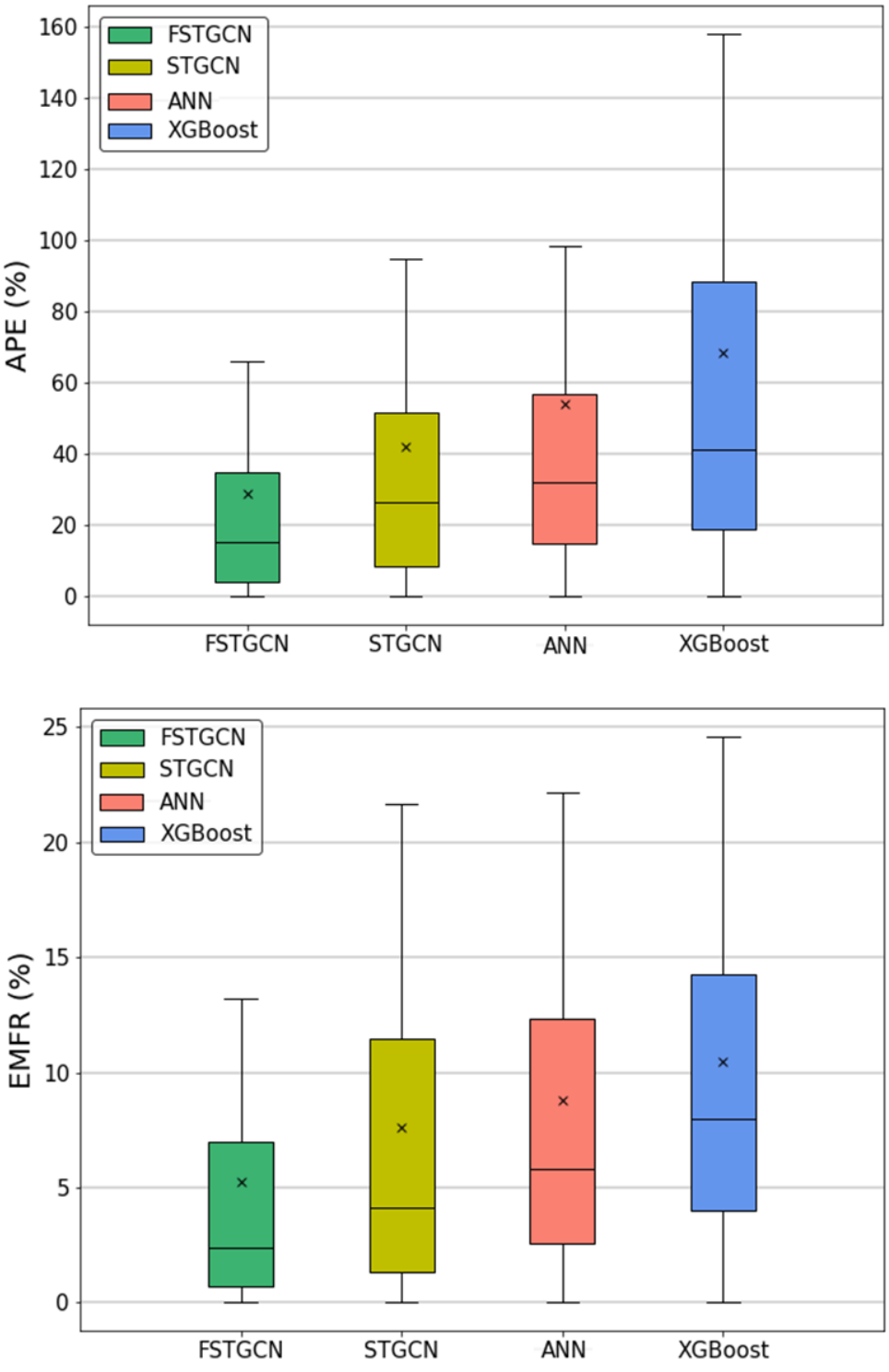


Figure 24. APE and EMFR distribution to investigate Fine-tuning gains.

the graph structure of the network. However, the most benefits from graph structure are gained with fine-tuning and focusing the model attention on the network's ongoing traffic condition.

6.4 Experiment 3: Loss function

The last experiment ran in this chapter is training the FSTGCN model using three different loss functions to see which one yields more accurate estimations of the network-wide traffic volumes. These three loss functions are Mean Squared Error (MSE), Mean Absolute Error (MAE), the summation of Mean Absolute Error, and Conservation of Flow (CoF). For the purposes of this study, these functions are defined as follows:

$$MSE = \frac{1}{|N| \times |T|} \sum_{t \in T} \sum_{i \in N} (y_i^t - \hat{y}_i^t)^2 \tag{27}$$

$$MAE = \frac{1}{|N| \times |T|} \sum_{t \in T} \sum_{i \in N} |y_i^t - \hat{y}_i^t| \tag{28}$$

$$MAE + CoF = \frac{1}{|N| \times |T|} \sum_{t \in T} \sum_{i \in N} |y_i^t - \hat{y}_i^t| + \lambda \sum_{t \in T} \sum_{c \in C} \sum_{i \in N} |\hat{y}_i^t x_i^c| \tag{29}$$

In this formula, N is the set of links in the road network (i.e., nodes in the representative graph), T is the set of time intervals that we want to estimate link traffic flows, C is the set of road segment junctions in the road network, and x_i^c is a value determining the relation of traffic flow in link i with junction c, defined as follows:

$$x_i^c = \begin{cases} 1: & \text{if the flow of link } i \text{ is entering junction } c \\ 0: & \text{if link } i \text{ is not directly connected to junction } c \\ -1: & \text{if the flow of link } i \text{ is exiting junction } c \end{cases} \tag{30}$$

To better understand the CoF part of the equation (28), Figure 25 shows an example of x_i^c computation in a network. According to this figure, the traffic flow of link 1 is entering the junction c_1 and traffic volume of link 2 is exiting it. Links 3 and 4 are not directly connected to the junction c_1. Therefore, $x_1^{c_1} = 1,\ x_2^{c_1} = -1,\ x_3^{c_1} = 0,$ $x_4^{c_1} = 0.$

With the same logic at c_2, $x_1^{c_1} = 0,\ x_2^{c_1} = 1,\ x_3^{c_1} = -1, x_4^{c_1} = -1$. Another value in *MAE + CoF* is λ, which is a hyperparameter to be set based on the network configuration. This value determines the weight of the CoF relative to the MAE in the loss function.

MSE and *MAE* are well-known loss functions used for various regression models. On the other hand, the summation of *MAE* and the conservation of flow is an innovative loss function introduced in this study to investigate whether adding a sense of conservation of flow can improve the FSTGCN model estimations.

Now that the experiment loss functions are defined, we train the introduced FSTGCN model using each loss function separately and compare the results. The network used for this experiment is a small part of the Eastern Maryland network separated by the black rectangular in Figure 26. The reason behind choosing this small network is that the *MAE + CoF* loss function requires a network with no missing links at connections. Conversely, the NPMRDS network used in this study only provides data on the high-level roads and often does not include connection links such as ramps. Therefore, we selected the small network to be able to add these connections and their data manually.

We use the same small network for all three loss functions to make a fair comparison between the results. The final λ value used for this network while using *MAE + CoF* loss function is $5e^{-8}$. This value is selected after training the model with a range of λ values to find the one that yields the best performance.

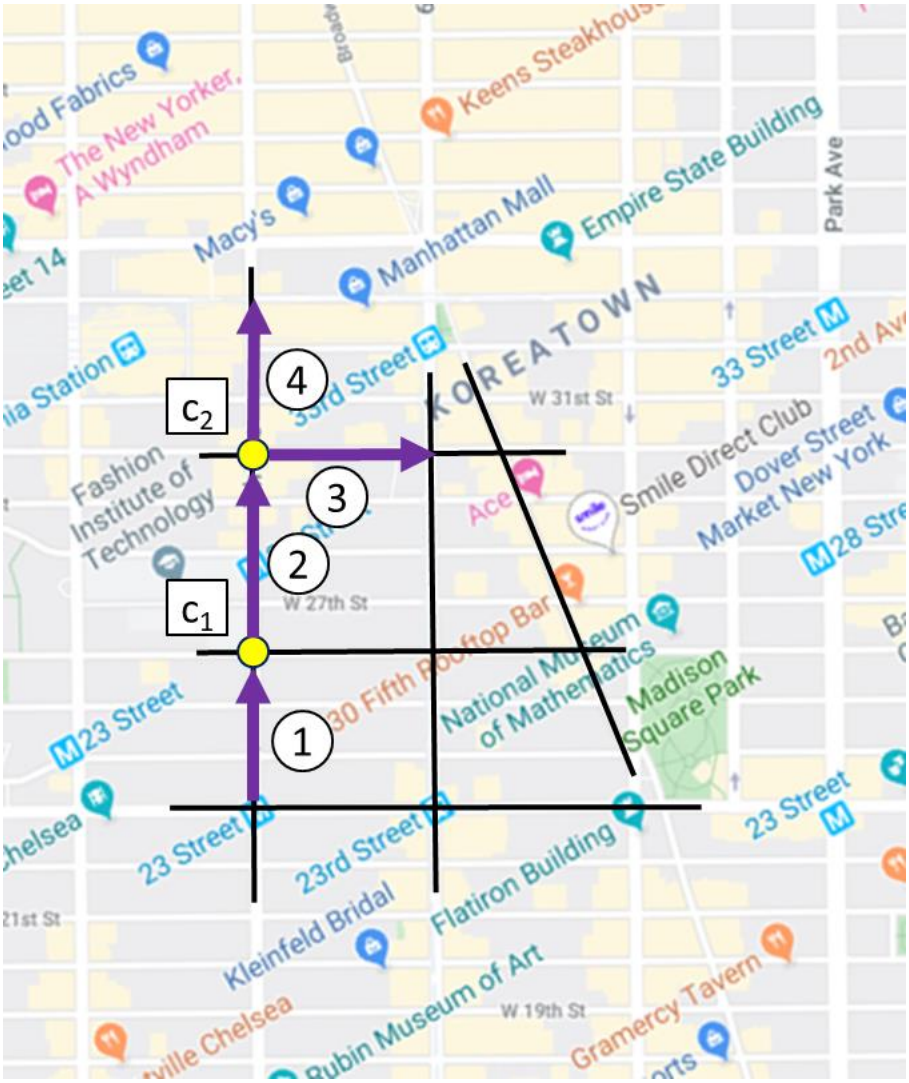


Figure 25. *MAE + CoF* loss function clarifying example.

The APE and EMFR of the models trained using the three discussed loss functions are presented in Figure 27. According to this figure, both *MAE* and *MAE + CoF* have

better performance compared to MSE loss function. However, the difference between *MAE* and *MAE + CoF* is not significant. Given the high computational cost of adding the conservation of flow to the loss function, which has enforced using fewer nodes in the GCN layers of the models trained for this experiment, *MAE* is the objective function we use for our numerical experiments in the rest of this study. However, the idea of *MAE + CoF* might be used in cases where the data is coming from a more granular network with a denser network of count sensors.

6.5 Chapter Summary

In the first section, this chapter presented the formulation of the XGBoost model trained to evaluate the FSTGCN model performance in estimating traffic volumes. Further, the model development procedure, along with the hyperparameters of the FSTGCN model, is discussed. Finally, the results of three experiments designed to investigate different settings of the FSTGCN model development are provided. These experiments illustrated the benefits of the proposed modeling framework and its capabilities compared to the ANN and XGBoost models. In the first experiment, possible benefits of expanding the study road network to include more CCSs in the training process are investigated. It is illustrated that inputting the data and graph representation of traffic volume of this expanded network can improve the model performance. In the second experiment, the fine-tuning step of the proposed framework is investigated to determine its benefits on model performance. The findings of the second experiment revealed the advantages of the fine-tuning step. However, even the STGCN model, which has not

undergone the fine-tuning phase, outperforms the ANN and XGBoost models indicating the benefits of adding graph structure to such models.

Figure 26. Experiment 3 study network.

One other experiment worth investigating with a graph-based model such as FSTGCN is to analyze the impact of the input CCSs location on the model performance. However, running such an experiment requires ground truth volume data on relatively close locations with the same road characteristics so that we can evaluate the effect of CCSs locations regardless of other influencing factors. In the currently available data for this study, it is impossible to set such an unbiased configuration to evaluate the effects of sensor location; however, given the availability of ground truth data on links close to CCSs, this experiment can be an interesting subject for future works.

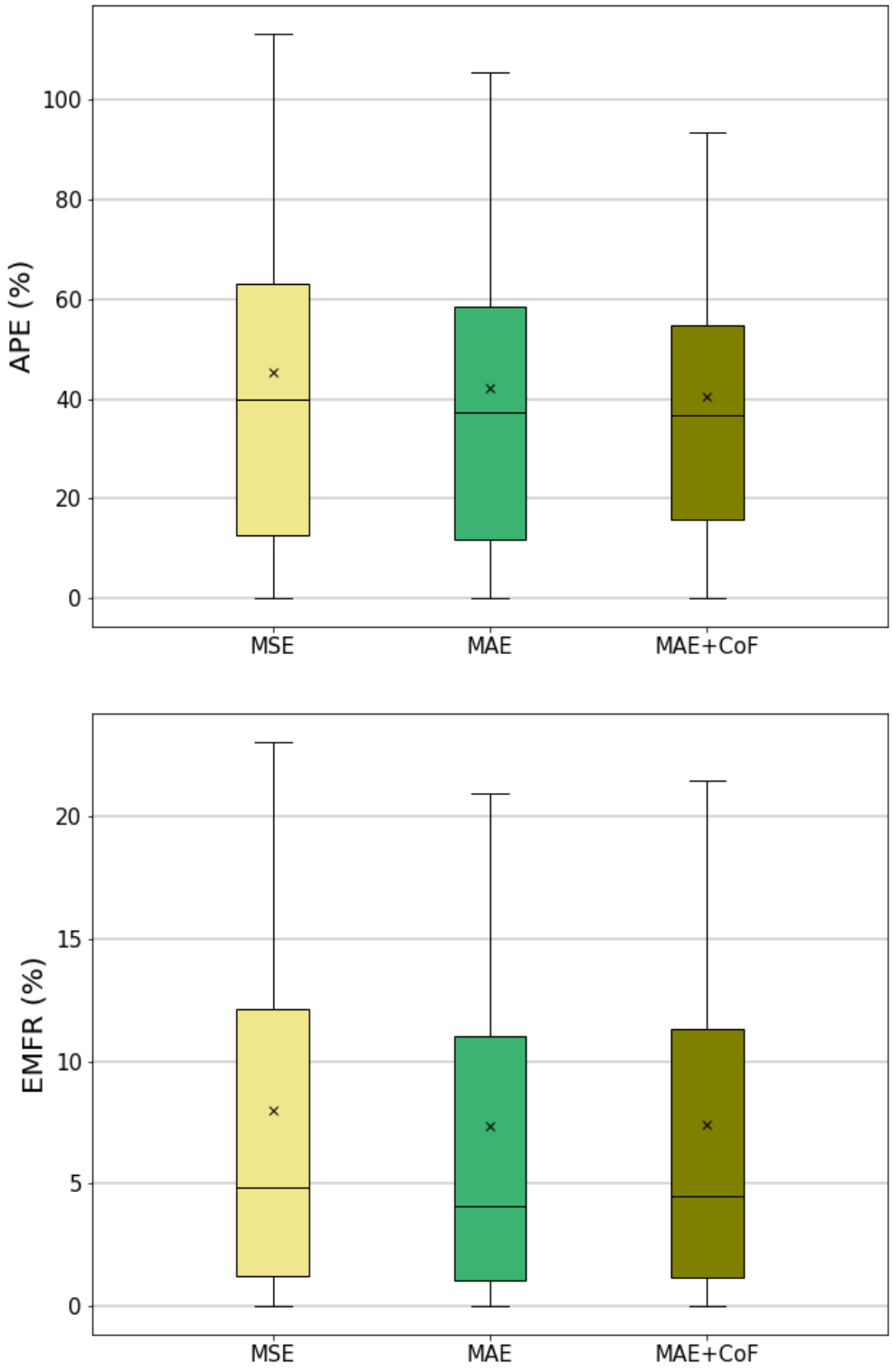


Figure 27. APE and EMFR distribution to investigate loss functions performance.

Chapter 7: Numerical Results

7.1 Overview

This section aims to provide the results of applying the proposed graph-based framework, the FSTGCN model, and the findings of the experiments designed in the previous chapter to the real-world data and compare its performance with the existing state-of-the-art models for network-wide traffic flow estimation. The superiority of this model is already investigated in chapter 6 using the Eastern Maryland network in various situations. Given the findings of chapter 6, we apply the model to two other previously introduced NPMRDS networks in Maryland, namely, Western Maryland and Beltway area. As discussed in chapter 5, these two networks have significantly different road characteristics and traffic patterns compared to each other and the Eastern Maryland network. Therefore, the results provided in this chapter enable an in-depth assessment of the introduced model performance in different traffic conditions and an exploration of the generalizability of the previous chapter's findings. Moreover, we provide results of applying the proposed FSTGCN model for traffic flow prediction to show its operational capabilities.

This chapter first provides the results using the FSTGCN model to estimate traffic volume for Western Maryland and Beltway area networks in section 7.2. Then we take one step further and apply the model for real-time volume prediction on the Beltway area network in section 7.3. The findings of this chapter are summarized in section 7.4.

7.2 Networkwide traffic flow estimation results

As mentioned before, the two networks of the Western Maryland and Beltway area are used for numerical analysis in this section. Here, we first present and discuss the graph-based framework performance in the Western Maryland network and then for the Beltway area network. In the end, we provide several aggregated analyses based on the results obtained from the case studies.

7.2.1 Western Maryland network

As presented in chapter 5, Western Maryland has a sparse NPMRDS network with relatively low traffic volumes. Figure 28 illustrates this network and its CCSs' locations. This network is passing through Garrett, Allegany, and Washington counties. All three CCSs are located on I-68, which has the highest concentrations of links in the network.

The same as the procedure described in chapter 6, we use the data of the entire year of 2019 for training and testing in this section. The results provided here are obtained from full cross-validation on CCSs. It means that each time the base model is trained using the 2019 yearly data of two CCSs and is separately fine-tuned for each time interval throughout the year. Then, the model is tested on the third CCS at each of those time intervals. This process is repeated to test all three stations individually. The training and testing process flowchart is presented earlier in Figure 21. The final results of the model are 15-minute traffic volumes for all links in the network during the year 2019. However, the evaluation of the model's accuracy is only possible on CCSs' locations where ground-truth traffic volumes are available. The distribution of *APE*

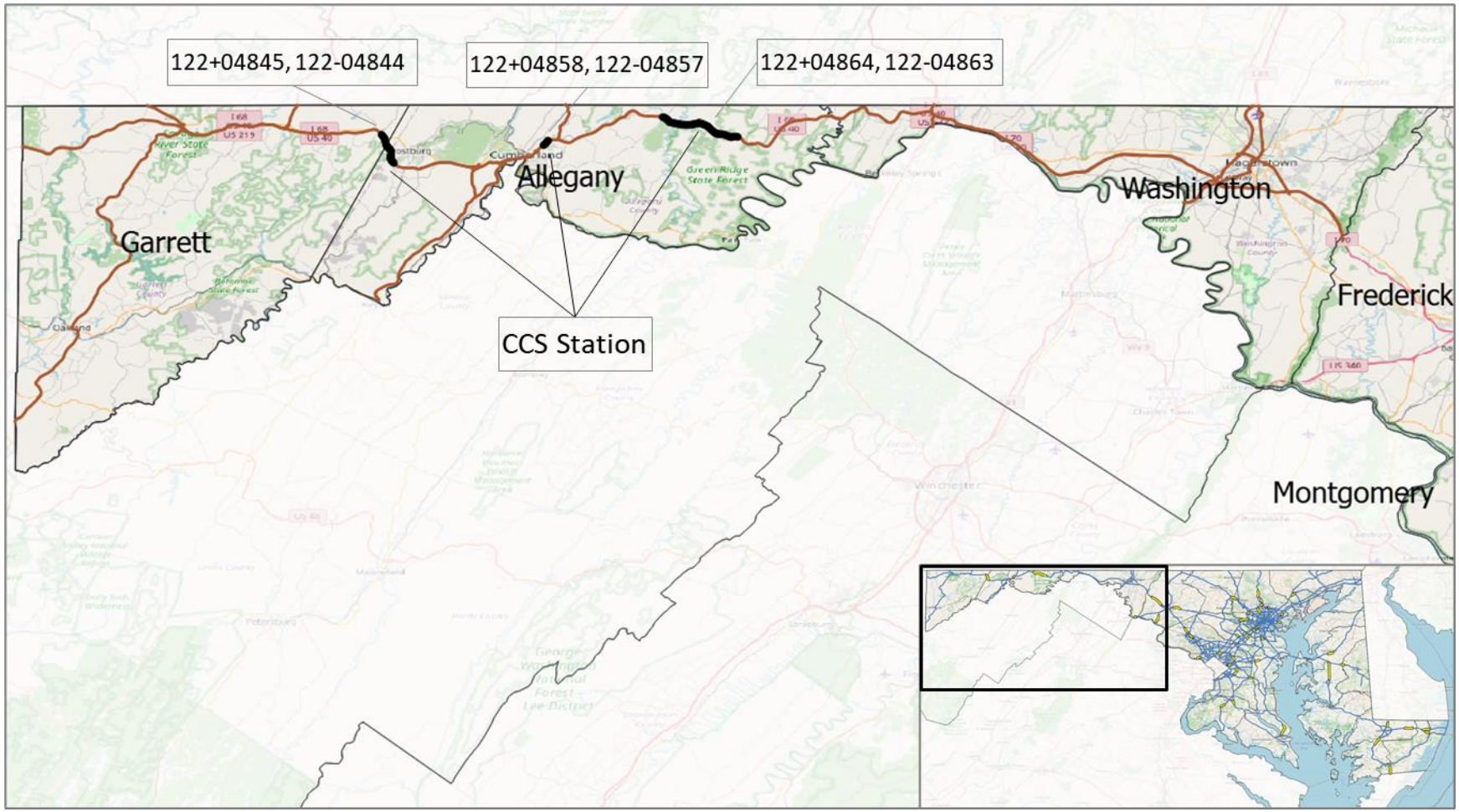


Figure 28. Western Maryland network and its CCSs' locations

and *EMFR* measures for the Western Maryland CCSs is presented in Figure 29. Moreover, these values are averaged for each TMC and provided in Table 9. According to Figure 29 and Table 9, the FSTGCN model outperforms the two other models based on all metrics before and after aggregating the results. Note that the ANN and XGBoost models are using the ground-truth data of 44 CCSs for training. Considering that the FSTGCN model only uses the ground-truth data of two CCSs, this model yields better results using approximately 5% ground-truth traffic volume data for training.

Another informative graphic is the daily patterns of traffic in a link. Figure 30 presents two sample daily traffic flow patterns estimated using the three models of FSTCGN, ANN, and XGBoost. This pattern is compared against the ground-truth traffic volume in the link. There are two sample days whose daily traffic flow patterns are plotted in this figure. The top plot illustrates a random day traffic pattern when all three models follow the actual traffic pattern.

Table 9. Western Maryland aggregated metrics group by TMC.

Location	FSTGCN		ANN		XGBoost	
	MAPE	MEMFR	MAPE	MEMFR	MAPE	MEMFR
122+04845	17.27	4.52	35.70	6.35	33.33	7.41
122+04858	22.36	4.86	24.34	6.36	33.94	8.69
122+04864	22.46	5.43	36.88	6.08	49.38	10.46
122-04844	20.17	5.26	42.65	6.35	40.30	7.83
122-04857	25.36	6.62	27.10	8.32	37.45	11.12
122-04863	23.09	5.15	45.38	6.33	46.84	9.75

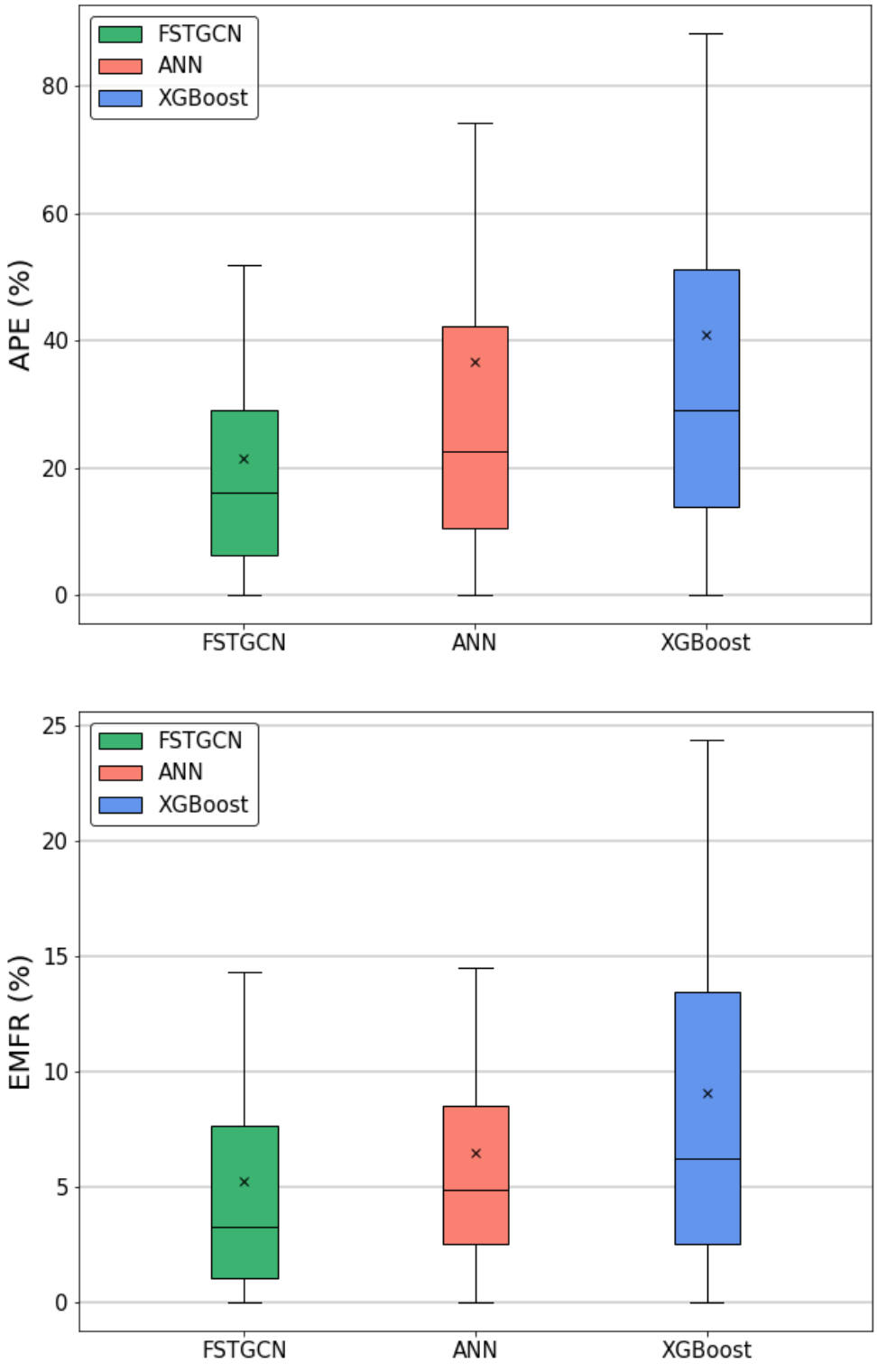


Figure 29. APE and EMFR distributions in Western Maryland.

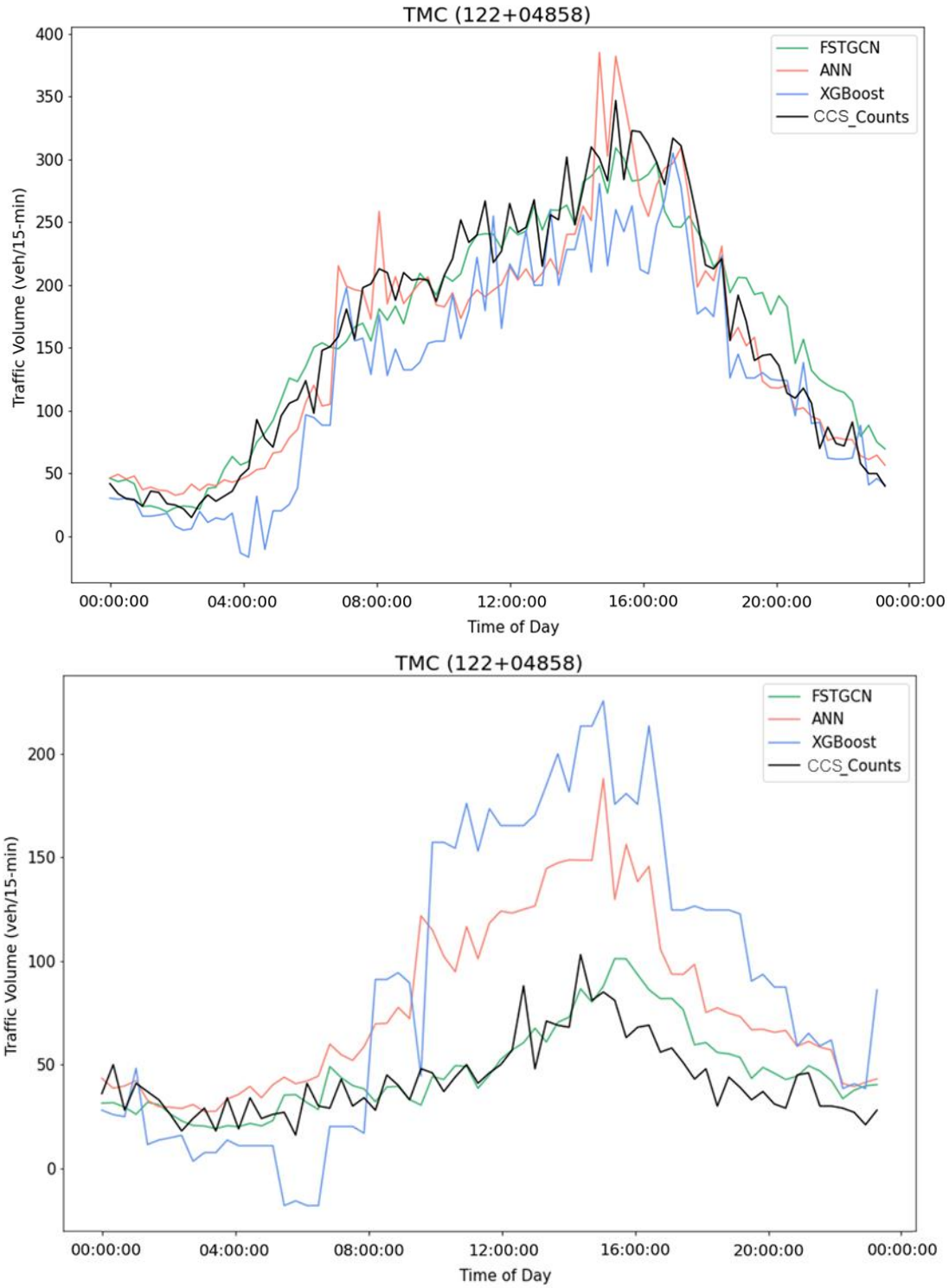


Figure 30. Daily traffic pattern samples in Western Maryland.

According to this plot, the FSTGCN and ANN estimations are very close to the actual traffic volumes. However, the bottom plot, which belongs to a snowy day in Maryland, presents an example of the FSTGCN model significantly outperforming the two other models. This finding indicates how adding spatial features to the model helps improving traffic volume estimation when an unusual traffic condition is observed in the network.

7.2.2 Beltway area network

Beltway area network is a congested network with a much higher number of links compared to Western Maryland. This network's map and its CCSs' locations are presented in Figure 31. There are five CCSs in this area that are mostly located on I-495 beltway links. These links are experiencing heavy traffics in rush hours, making traffic management challenging in the area.

We train the graph-based model through the same process as Eastern and Western Maryland networks (i.e., Figure 21). Note that we use four CCSs for training and one for testing as we have five CCSs in this network. Accordingly, the distribution of *APE* and *EMFR* for Beltway area CCSs are presented in Figure 32. Moreover, these values are averaged for each TMC and shown in Table 10.

According to Figure 32 and Table 10, all metrics are improved using the FSTGCN model. As far as daily traffic flow patterns are concerned, Figure 33 illustrates two sample days similar to the Western Maryland region. Although we see significantly different daily traffic patterns here, the same story we observed on the Western

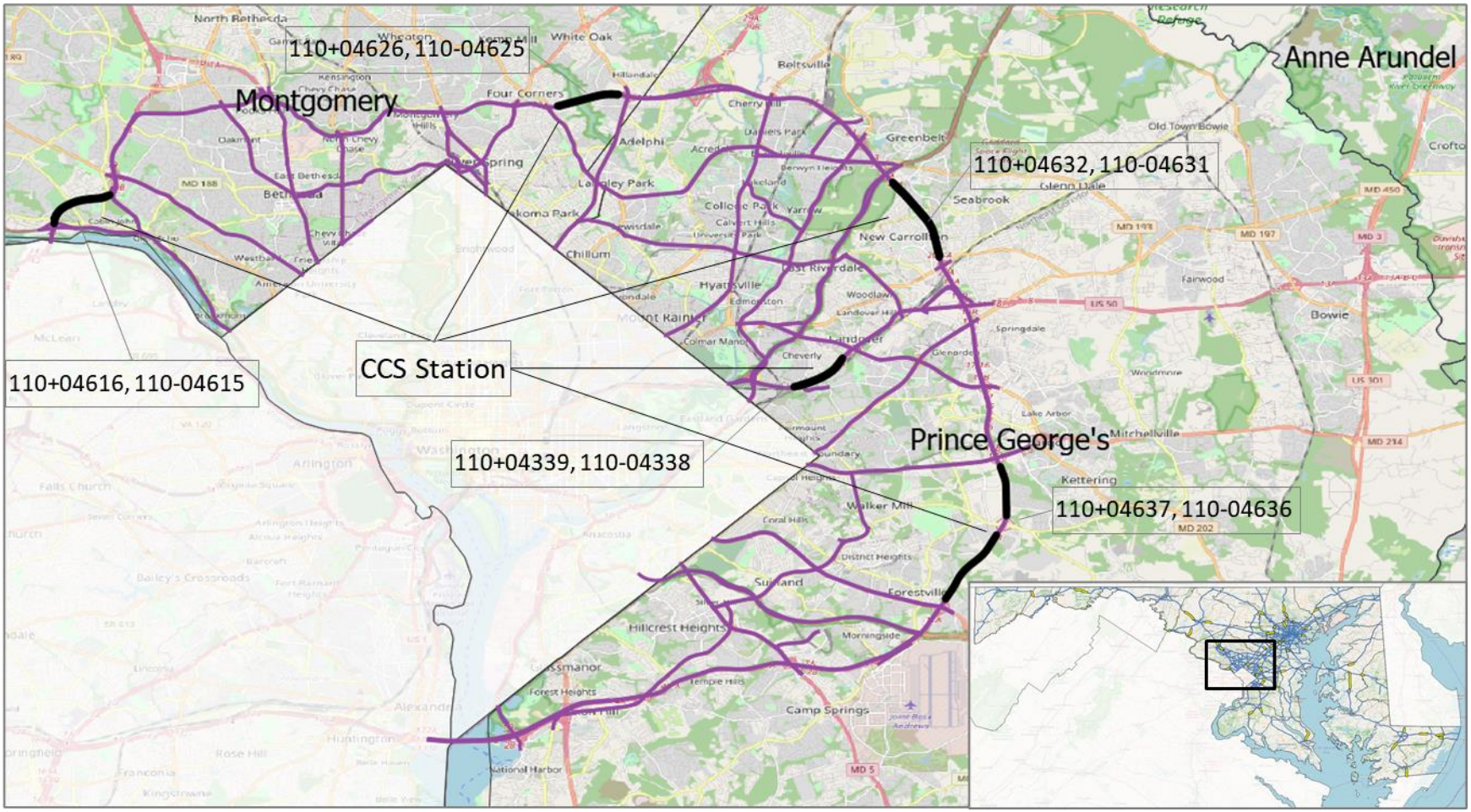


Figure 31. Beltway area network and its CCSs' locations.

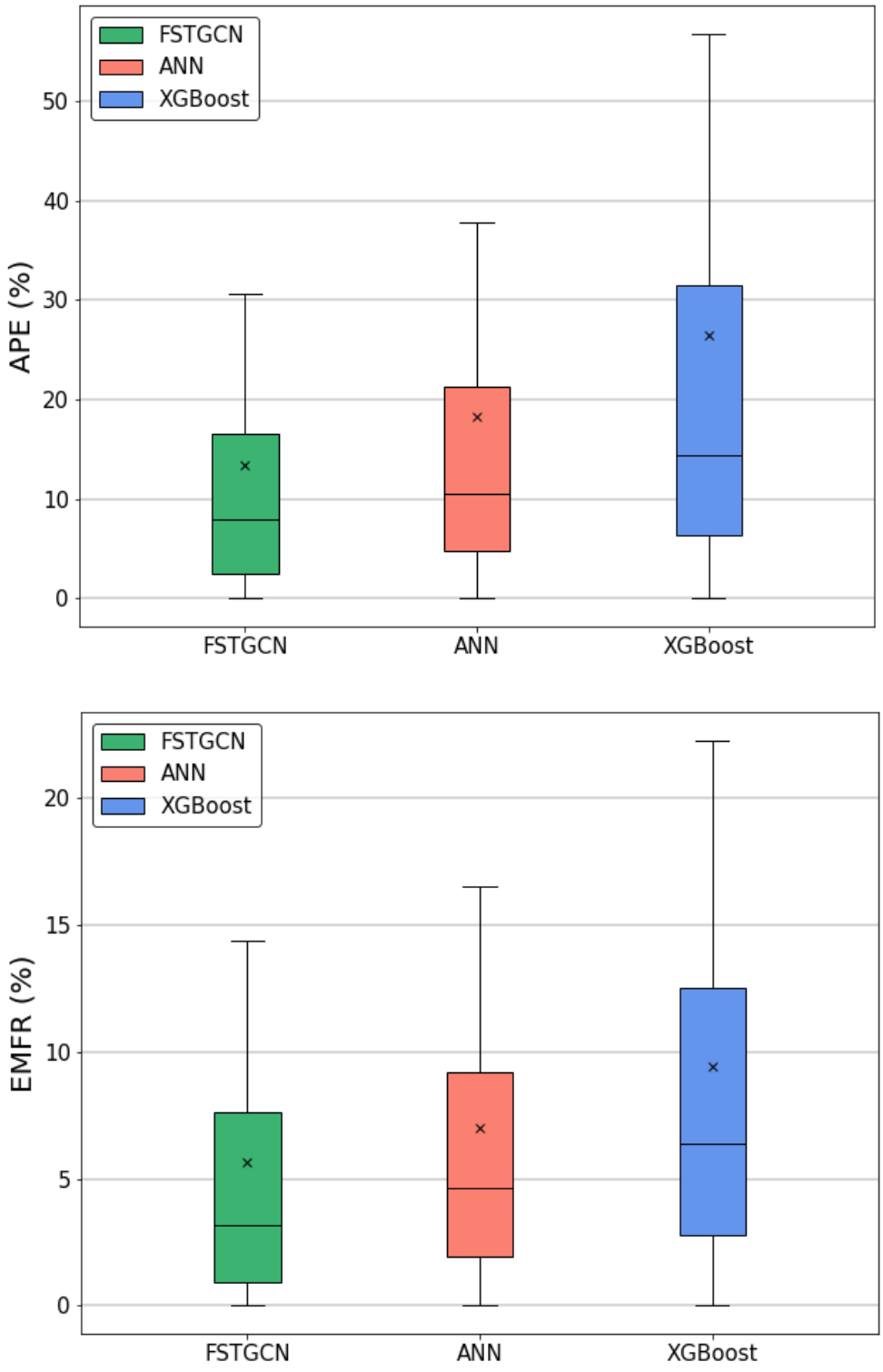


Figure 32. APE and EMFR distributions in the Beltway area.

Table 10. Beltway area aggregated metrics group by TMC.

Location	FSTGCN		ANN		XGBoost	
	MAPE	MEMFR	MAPE	MEMFR	MAPE	MEMFR
110+04339	27.05	11.93	47.34	14.41	52.25	17.75
110+04616	11.73	3.44	17.19	5.61	29.51	7.44
110+04626	9.28	4.69	12.25	5.70	17.36	6.43
110+04632	9.08	4.01	12.73	5.03	19.35	6.86
110+04637	8.08	4.39	11.14	5.03	20.29	8.04
110-04338	32.88	12.11	28.17	11.71	41.98	17.49
110-04615	9.66	5.17	13.54	5.94	21.95	8.02
110-04625	12.50	4.28	14.91	6.18	17.56	6.54
110-04631	7.31	3.17	12.48	4.86	21.18	7.01
110-04636	6.47	3.13	12.33	5.59	23.04	8.38

Maryland samples is repeated here. Both the FSTGCN and ANN models are closely following actual traffic volumes on a typical random day. Although not as good as the FSTGCN and ANN models, XGBoost also captures the typical traffic pattern. However, the FSTGCN significantly outperforms the other two for the snowy day when the network is experiencing much lower traffic volumes than usual.

The results provided for the Beltway area confirm that the FSTGCN model outperforms the existing models regardless of the study network geometry and its general traffic conditions. The following section puts the results of the three study areas together and compares them with the state-of-the-art ANN model from different aspects. This provides more information about the overall performance of the introduced model compared to the ANN.

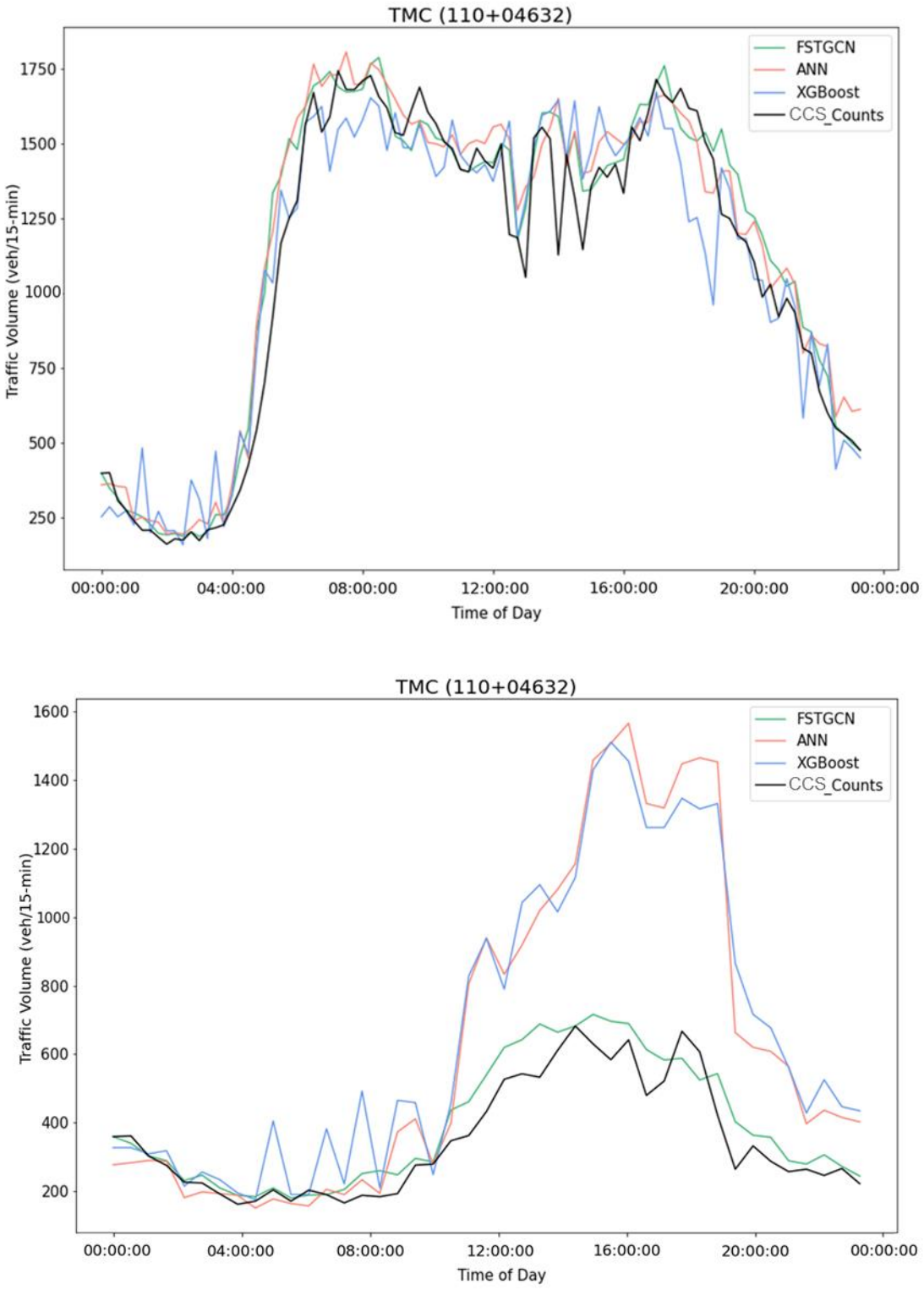


Figure 33. Daily traffic pattern samples in the Beltway area.

7.2.3 Overall numerical results

The results provided in previous sections and chapter 6 indicated that the FSTGCN model improves the accuracy of estimated traffic volumes in various traffic conditions. This section combines the results of training and testing the models over the three introduced regions of the Maryland NPMRDS network and analyzes them based on different traffic characteristics categories. However, before presenting the categorized comparison results, Table 11 presents the summary statistics of FSTGC, ANN, and XGBoost performances across all three networks based on the two metrics of APE and EMFR. According to this table, FTGCN outperforms ANN and XGBoost for all measures. Additionally, Figure 34 shows the heatmaps of all estimated traffic volumes obtained from the three models vs. actual values to see the results with more details. According to this figure, it is evident that FSTGCN has more accurate estimates as the observations are centered around the 45-degree line more rigorously compared to the other two models.

Table 11. Summary statistics of FSTGC, ANN, and XGBoost performances across all three networks.

Model	FSTGCN		ANN		XGBoost	
Measure	APE	EMFR	APE	EMFR	APE	EMFR
25th percentile	3.91	0.90	7.96	2.21	10.95	3.13
Median	12.34	3.05	19.18	5.05	27.67	7.37
75th percentile	26.75	7.66	38.84	10.09	60.99	15.16
Mean	19.81	5.54	34.22	7.60	53.15	11.54

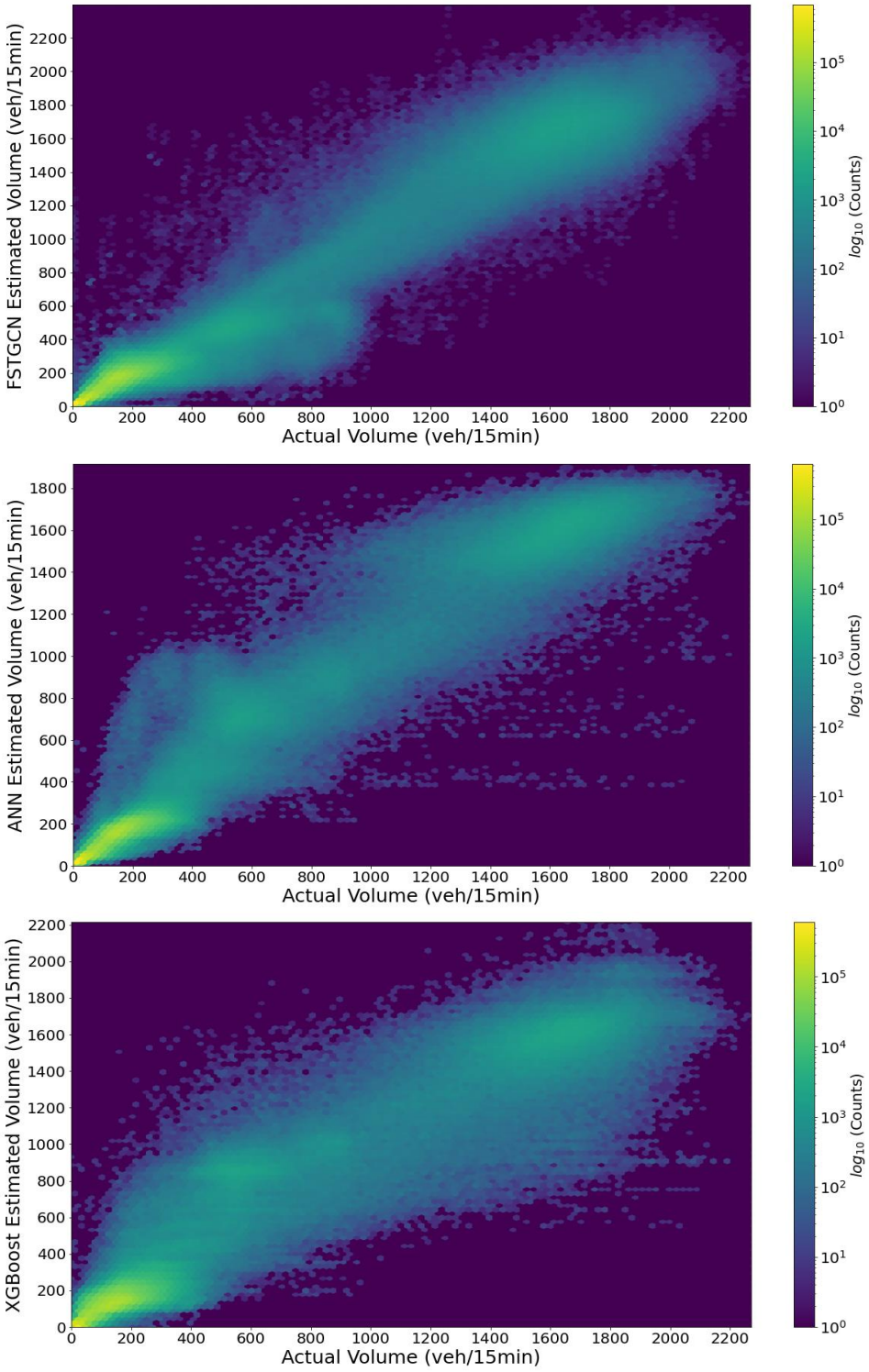


Figure 34. Heatmaps of the estimated volumes vs. actual volumes.

Based on the numerical results provided so far, the FSTGCN model is followed by the ANN model as the second-best model. Therefore, in what follows, we compare the FSTGCN model with the ANN based on different traffic characteristics. The first traffic characteristic we consider here is the Functional Road Class (FRC) of the links, where we can compare the estimations with the ground-truth traffic volumes. Figure 35 illustrates relative median error reduction using the FSTGCN model versus the ANN model in different FRC levels. The error metrics used here are *EMFR* and *APE* as before, and the values for each group (i.e., FRC level here) are computed based on the following formulas:

$$E_1 = \frac{Med\ (EMFR_{ANN}) - Med\ (EMFR_{FSTGCN})}{Med\ (EMFR_{ANN})} \times 100 \quad (31)$$

$$E_2 = \frac{Med\ (APE_{ANN}) - Med\ (APE_{FSTGCN})}{Med\ (APE_{ANN})} \times 100 \quad (32)$$

This means that for each FRC level, we compute the median *EMFR* and APE of all observations belong to that FRC level for both ANN and FSTGCN to see how much each metrics is reduced using FSTGCN versus ANN.

As presented in Figure 35, both EMFR and APE are reduced significantly for all FRCs. However, this reduction is more noticeable for FRC=3, which are the lower-level roads. This is a valuable observation because, in general, the traffic volume estimation is less accurate on lower-level roads. Therefore, the higher improvement gained for these roads using FSTGCN indicates the model's superiority in challenging situations.

Another level of aggregation used for comparison in this section is grouping the observations based on the congestion level in links. The congestion level we defined

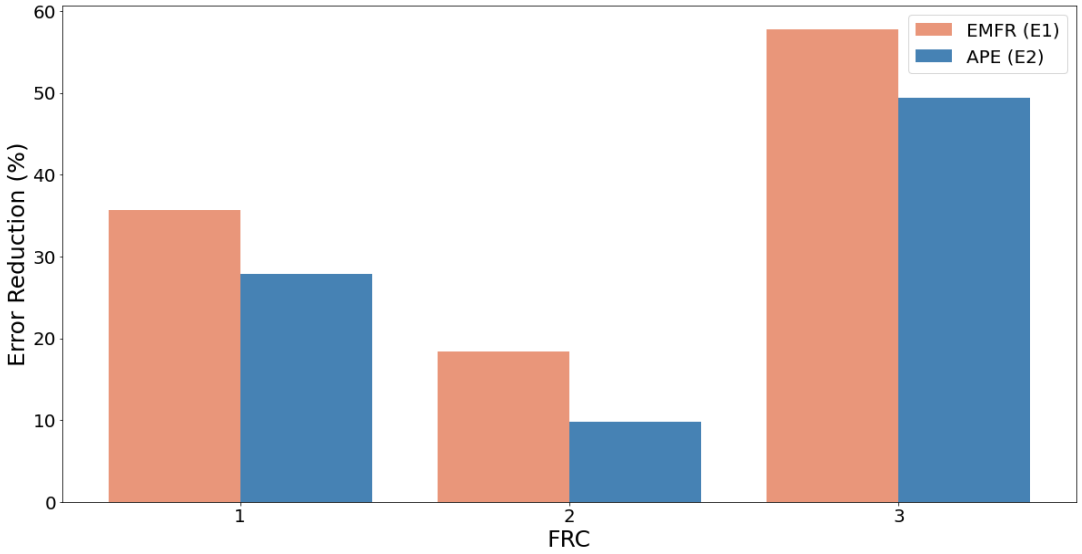


Figure 35. Error reduction based on FRC.

here is computed by dividing the ground-truth traffic flow at each time interval by the maximum flow observed in that link. This method produces a value between 0 and 1 corresponding to the lowest and highest congestion levels, respectively. Based on this definition, we divided the congestion level into ten bins. The error reduction results for each bin computed based on equations (31) and (32) are presented in Figure 36. According to this figure, the error is reduced for all congestion levels. This reduction is more significant for the bins corresponding to uncongested situations. The reason for lower improvements in congested situations is that in these conditions, the traffic speed is a robust indicator of traffic volumes; thus, the ANN model is already performing well. This observation is consistent with our previous findings regarding the superiority of the FSTGCN in traffic volume estimation in extreme and challenging conditions.

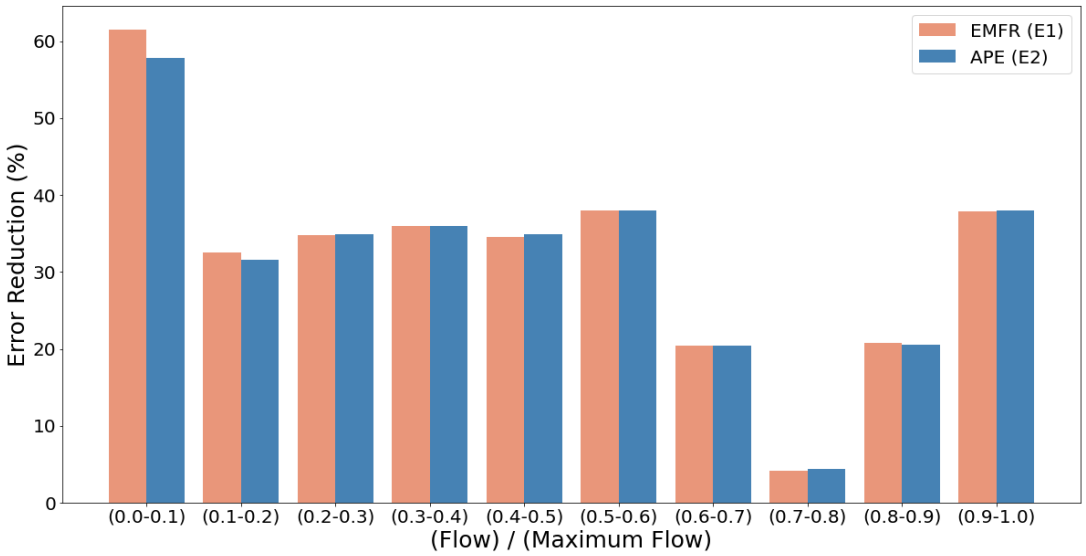


Figure 36. Error reduction based on congestion level.

7.3 Temporally aggregated results

This section aims to analyze the performance of FSTGCN compared to the ANN and XGBoost models when the 15-minute estimated volumes across all three study areas are aggregated temporally. Three aggregation levels of hourly volumes, daily volumes, and AADT values are used for temporal aggregation analysis in this section.

To compute the hourly and daily volumes, we simply sum the 15-minute estimations for each hour and day of the year, respectively. Further, similar to 15-minute estimates, the previously introduced error measures are calculated for these estimates. The distribution of APE and EMFR of hourly and daily volumes computed for FSTGCN, ANN, and GXBoost are presented in Figures 37 and 38. From these figures, we can see that for both hourly and daily aggregated volumes, the FSTGCN outperforms the other two models.

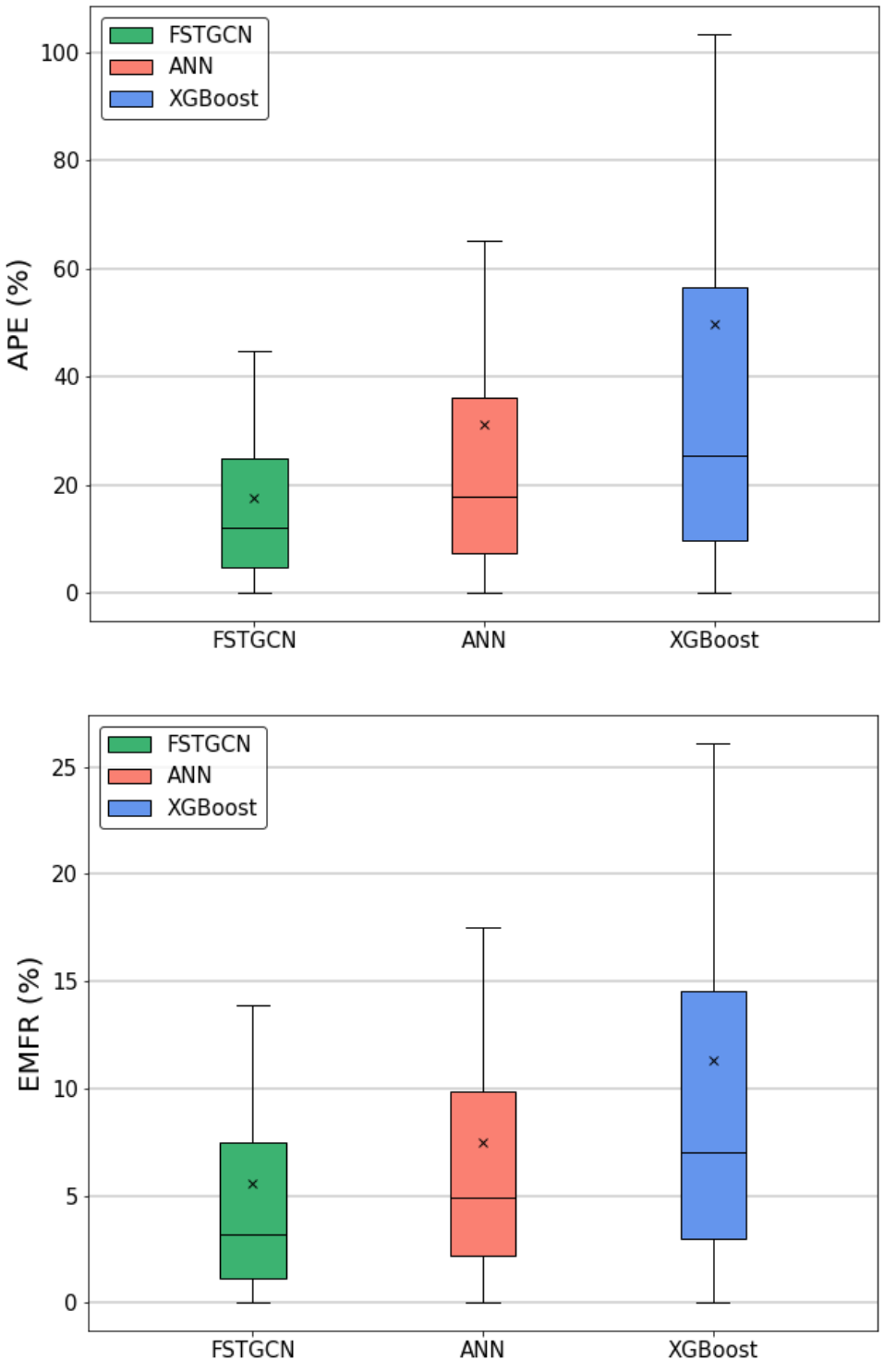


Figure 37. APE and EMFR distributions for hourly aggregated volumes.

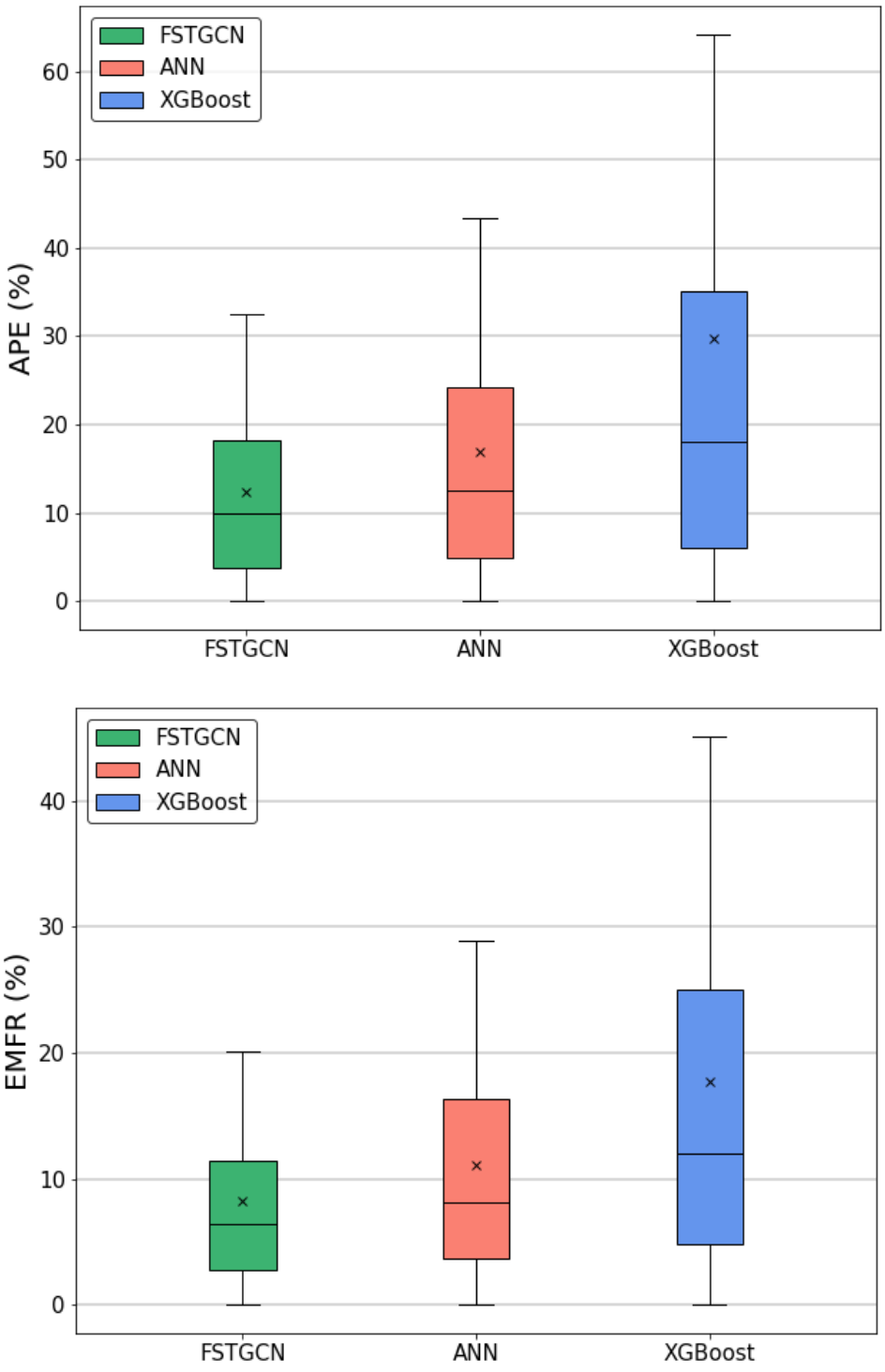


Figure 38. APE and EMFR distributions for daily aggregated volumes.

Having the hourly traffic volumes, we can compute GEH statistic, which is an empirical formula named after its inventor Geoffrey E. Havers (DMRB, 2005). GEH formula is presented in Equation (33):

$$GEH = \sqrt{\frac{2(y_h - \hat{y}_h)^2}{(y_h + \hat{y}_h)}} \tag{33}$$

where y_h is the observed hourly traffic volume and $\hat{y}_h$ is the estimated hourly volume. Figure 39 shows the distribution of GEH computed using the hourly traffic volumes estimated by all three models. According to this figure, FSTGCN has lower GEH values compared to the other two models. The 85-percentile GEH value for the FSTGCN, ANN, and XGBoost is equal to 8.33, 10.18, and 16.12, respectively.

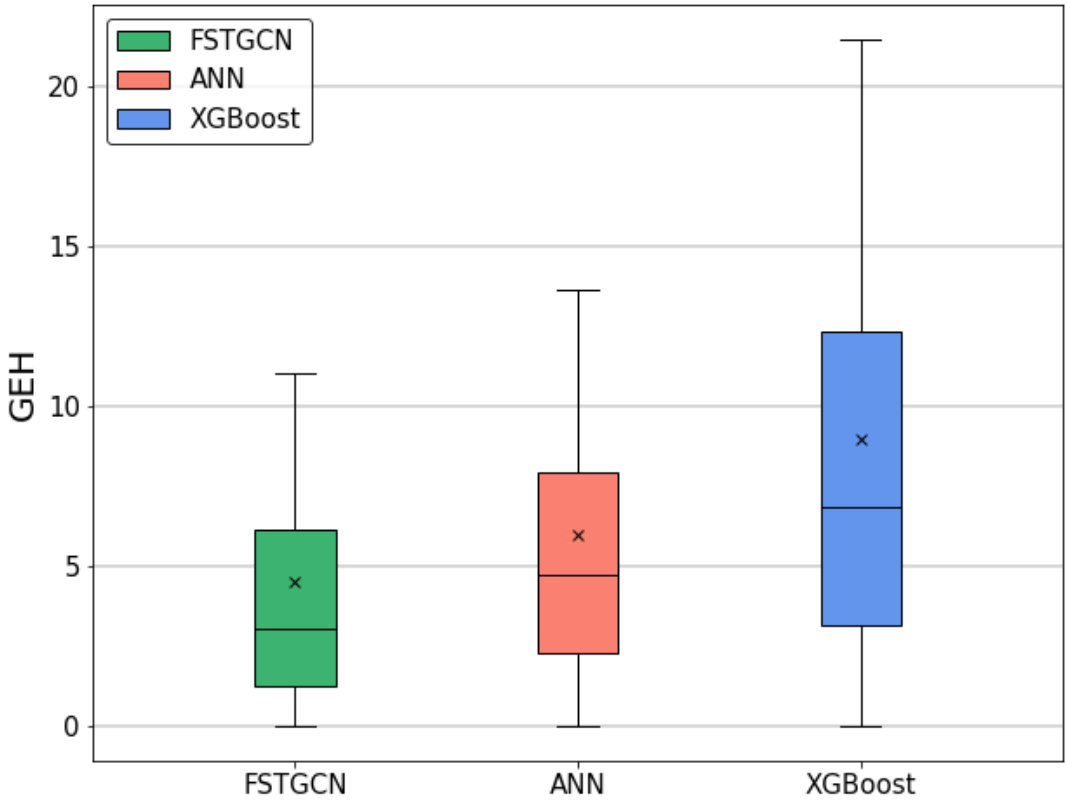


Figure 39. GEH distribution based on hourly traffic volumes estimated by FSTGCN, ANN and XGBoost.

According to FHWA Traffic Analysis Toolbox (2019), the 85-percentile GEH value for individual link flows less than 5 is acceptable. Although none of the models fall into this threshold, the significant reduction in 85-percentile GEH value gained by using FSTGCN proves this model's superiority.

The other temporally aggregated metric widely used for volume count analysis is AADT. The FHWA-recommended computation procedure of AADT (TMG, 2016) when there are missing data in traffic counts are presented in Equations (34) and (35):

$$MADT_m = \frac{\sum_{j=1}^{7} w_{jm} \sum_{h=1}^{24} [\frac{1}{n_{hjm}} \sum_{i=1}^{n_{hjm}} VOL_{ihjm}]}{\sum_{j=1}^{7} w_{jm}} \quad (34)$$

$$AADT = \frac{\sum_{m=1}^{12} d_m \, MADT_m}{\sum_{m=1}^{12} d_m} \quad (35)$$

where:

$AADT$ = average annual daily traffic,

$MADT_m$= monthly average daily traffic for month m,

VOL_{ihjm}= total traffic volume for ith occurrence of the hth hour of day within jth day of the week during the mth month,

i = occurrence of a particular hour of day within a particular day of the week in a particular month ($i = 1, \dots, n_{hjm}$) for which traffic volume is available,

h = hour of the day ($h = 1,2, \dots ,24$) – or other temporal intervals,

j = day of the week ($j = 1,2, \dots ,7$),

m = month ($m = 1, \dots ,12$),

n_{hjm}= the number of times the hth hour of day within the jth day of the week during the mth month has available traffic volume (n_{hjm}ranges from 1 to 5 depending on the hour of the day, day of the week, month, and data availability),

w_{jm}= the weighting for the number of times the jth day of the week occurs during the mth month (either 4 or 5); the sum of the weights in the denominator is the number of calendar days in the month (i.e., 28, 29, 30, or 31),

d_m= the weighting for the number of days (i.e., 28, 29, 30, or 31) for the mth month in the particular year.

The results of computing AADT according to estimates of each model and ground truth data for each CCS are presented in Table 12. Furthermore, these values are used to compute the absolute error percentage between the difference of observed and estimated AADT values, as shown in Equation (36).

$$E_i^j = \left| \frac{\widehat{AADT}_i^j - AADT_i}{AADT_i} \right| \times 100 \quad (36)$$

Where:

E_i^j= Absolute AADT estimation error in TMC i, using model j,

$\widehat{AADT}_i^j$= Estimated AADT in TMC i, using model j,

$AADT_i$= Computed AADT in TMC i, using the recorded traffic volume counts.

The distribution of Absolute AADT estimation error is presented in Figure 40 for different CCS locations and their corresponding TMC segments. According to Table 12 and Figure 40, although the FSTGC, with an average error of 10.78% compared to 13.60% for ANN and 21.45% for XGBoost, has a better average performance

estimating AADT, there are locations where ANN or even XGBoost estimate AADT more accurately. Given the results we have shown so far, this indicates that although ANN and XGBoost have significantly higher error estimating volume on 15-minute, hourly, and daily levels, these errors are often not biased in a way that makes inaccurate estimations of highly aggregated metrics such as AADT.

Table 12. Observed and Estimated AADTs on CCS locations.

CCS	CCS_Counts	FSTGCN	ANN	XGBoost
110+06339 110+06339	26,886	22,307	20,855	28,117
122+04864 122-04863	18,849	21,629	19,913	23,121
110+06366 110-06365	20,501	21,907	20,680	15,310
110+04626 110-04625	216,905	208,756	210,552	214,476
110+04637 110-04636	226,665	222,199	216,926	215,548
122+04845 122-04844	23,093	20,132	25,860	24,068
110+04632 110-04631	213,625	222,302	218,099	214,144
110+04339 110-04338	82,062	61,942	101,653	110,085
110+07791 110-06335	15,080	17,864	19,803	16,675
110+09016 110-07392	8,625	7,697	6,160	17,605
122+04858 122-04857	31,875	30,705	26,887	24,200

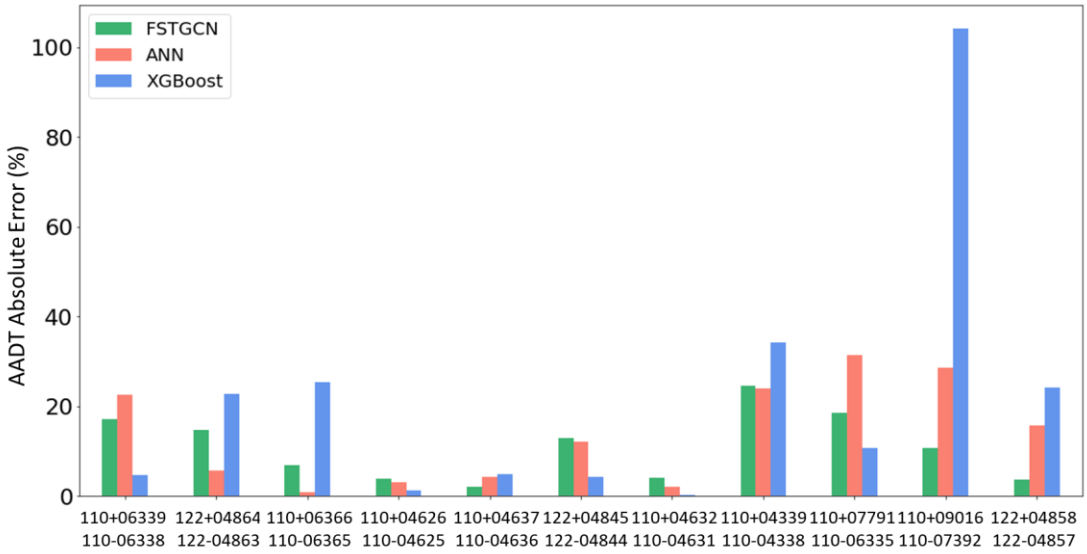


Figure 40. Comparison of AADT absolute error percentage

7.4 Graph-based model real-time application

So far, we analyzed the performance of the FSTGCN for network-wide traffic volume estimation and illustrated how this model outperforms the existing state-of-the-art models. In this section, we investigate the capability of the FSTGCN model for real-time applications.

Although the FSTGCN model is originally designed for historical traffic volume estimation, the model can be adopted to predict traffic flow in real-time. As discussed in chapter 4, this model is constructed from two sections of STGCN model training and Fine-tuning the STGCN model. For real-time applications, the STGCN part can be trained offline using the available historical data. Once the model weights are determined, the model can be Fine-tuned in real-time using the most recent data available for the study network.

Here we use the Beltway area network to analyze the accuracy of the FSTGCN model for real-time traffic flow prediction. To do so, we assume that the data of the first eight months of 2019 is available, and the objective is to see how the FSTGCN predicts traffic volume for the remaining four months when the online data is arriving with 15, 30, 45, and 60 minutes delays. It means that we train the STGCN model using eight months of data, and for each interval in the following four months, we fine-tune the model using the most recent data (i.e., last 15, 30, 45, or 60 minutes).

The training process for this analysis is represented in Figure 41, where Δt is the data arrival delay and can be equal to 15, 30, 45, or 60 minutes. Note that training of the STGCN, which is done offline, is taking about 90 minutes using a computer with Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20GHz, 40 cores, and NVIDIA Quadro M4000 GPU. However, fine-tuning for each time interval takes less than 10 seconds, which can be easily done in real-time.

The distribution of *APE* and *EMFR* for each Δt is illustrated in Figure 42. According to this figure, the FSTGCN model has an average *APE* of less than 20% and an average *EMFR* of less than 10%, even when the data arrives with a one-hour delay. These are acceptable values given that they present the model's accuracy predicting volume for the locations whose ground-truth traffic volume data has never been introduced to the model.

Additionally, Figure 43 presents some sample daily traffic volume patterns predicted by the FSTGCN model with different Δts against the actual values. An interesting observation here is that, although the model can predict the general traffic pattern, there is a lag in predictions directly correlated to the data arrival delay. This lag is more

evident when traffic increases fast to reach the peak during the morning hours. This observation can be used for future studies focused on traffic volume prediction using GCN-based models.

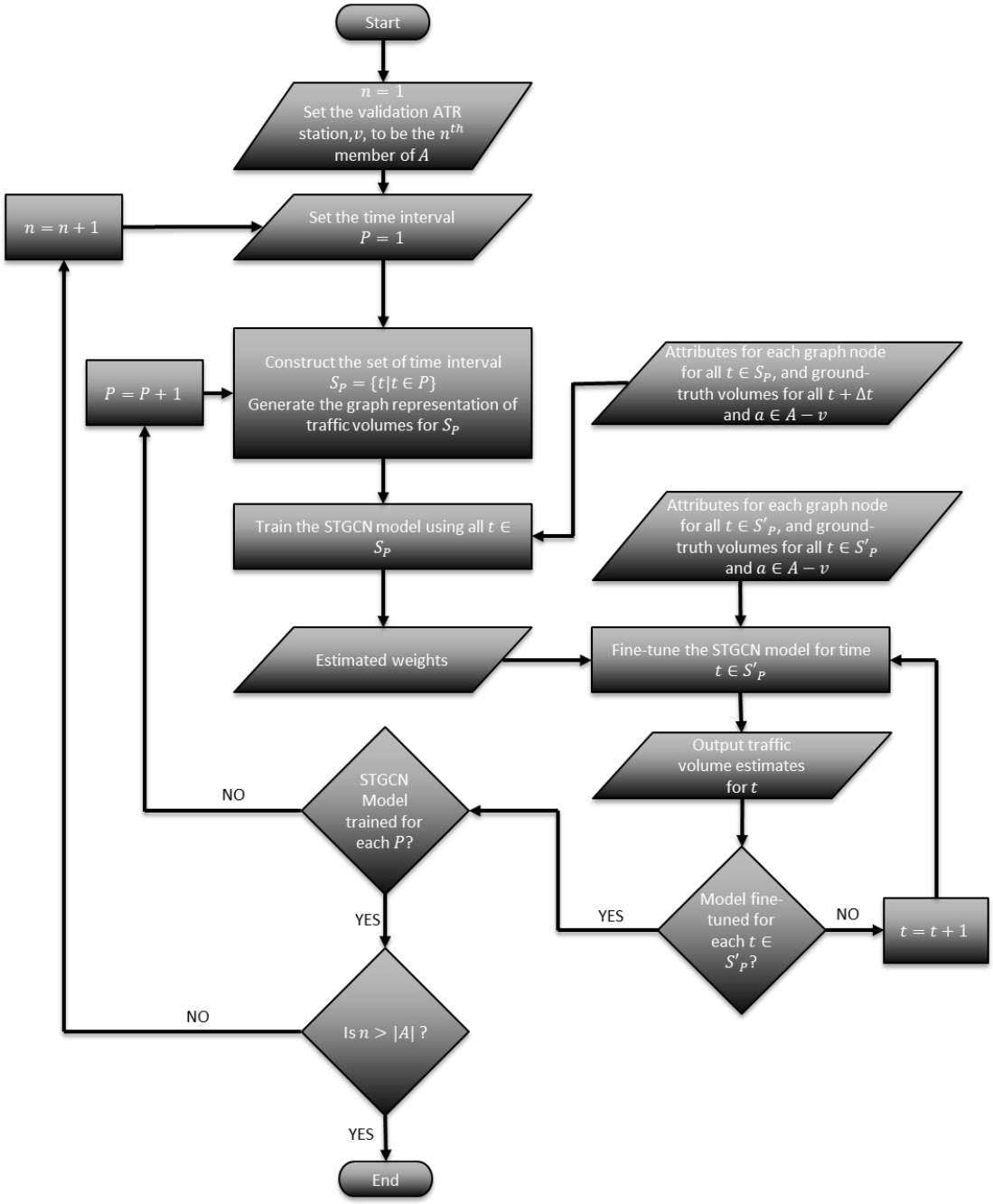
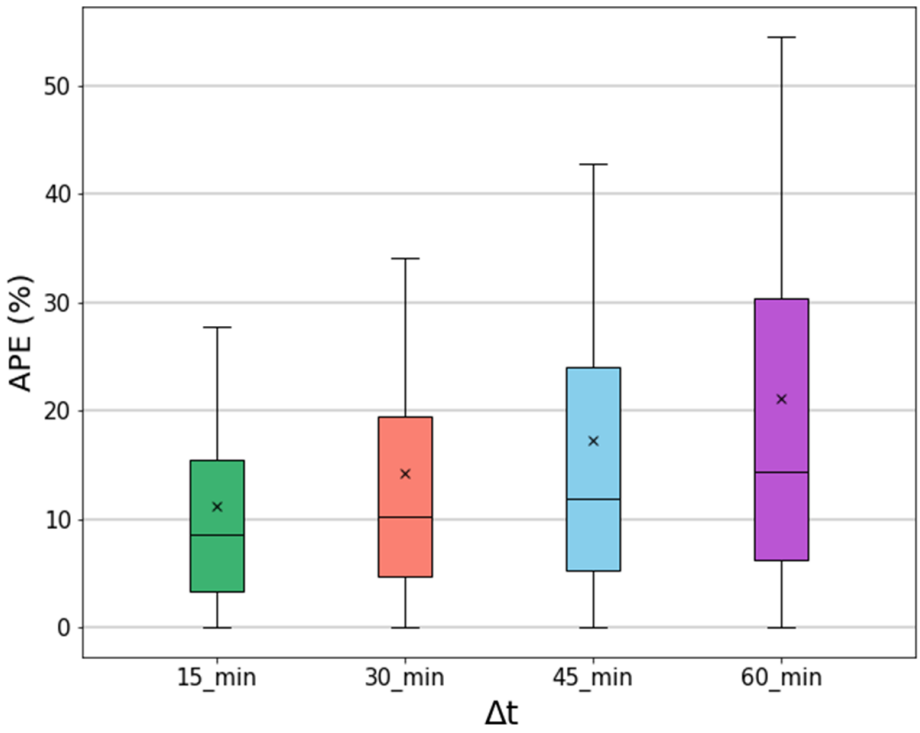


Figure 41. Training process flowchart for FSTGCN application in real-time.



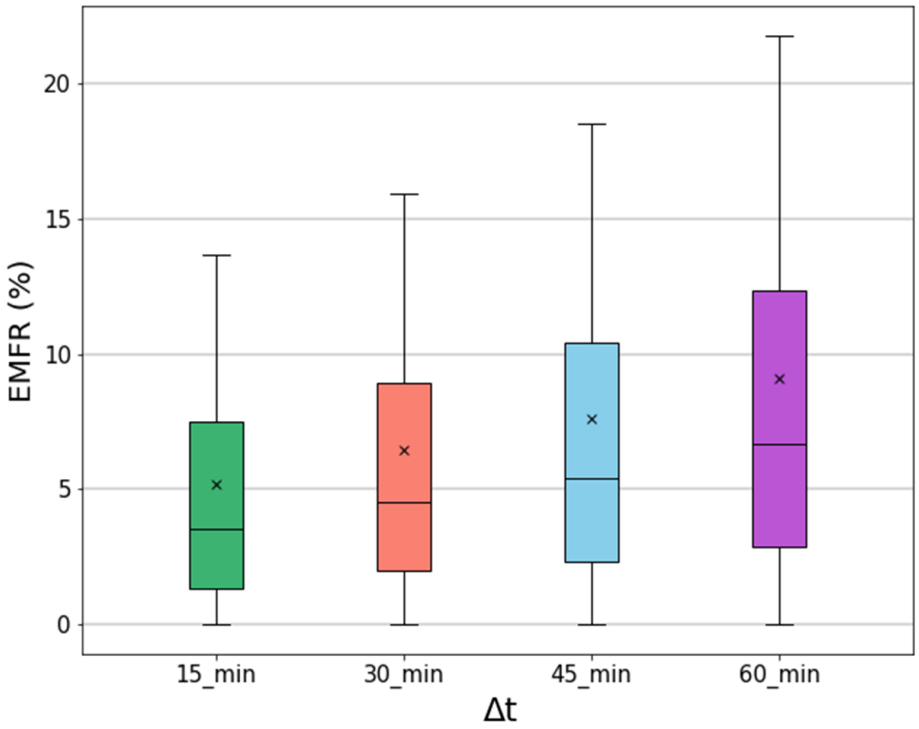


Figure 42. APE and EMFR distributions for predicting traffic flows in Beltway area.

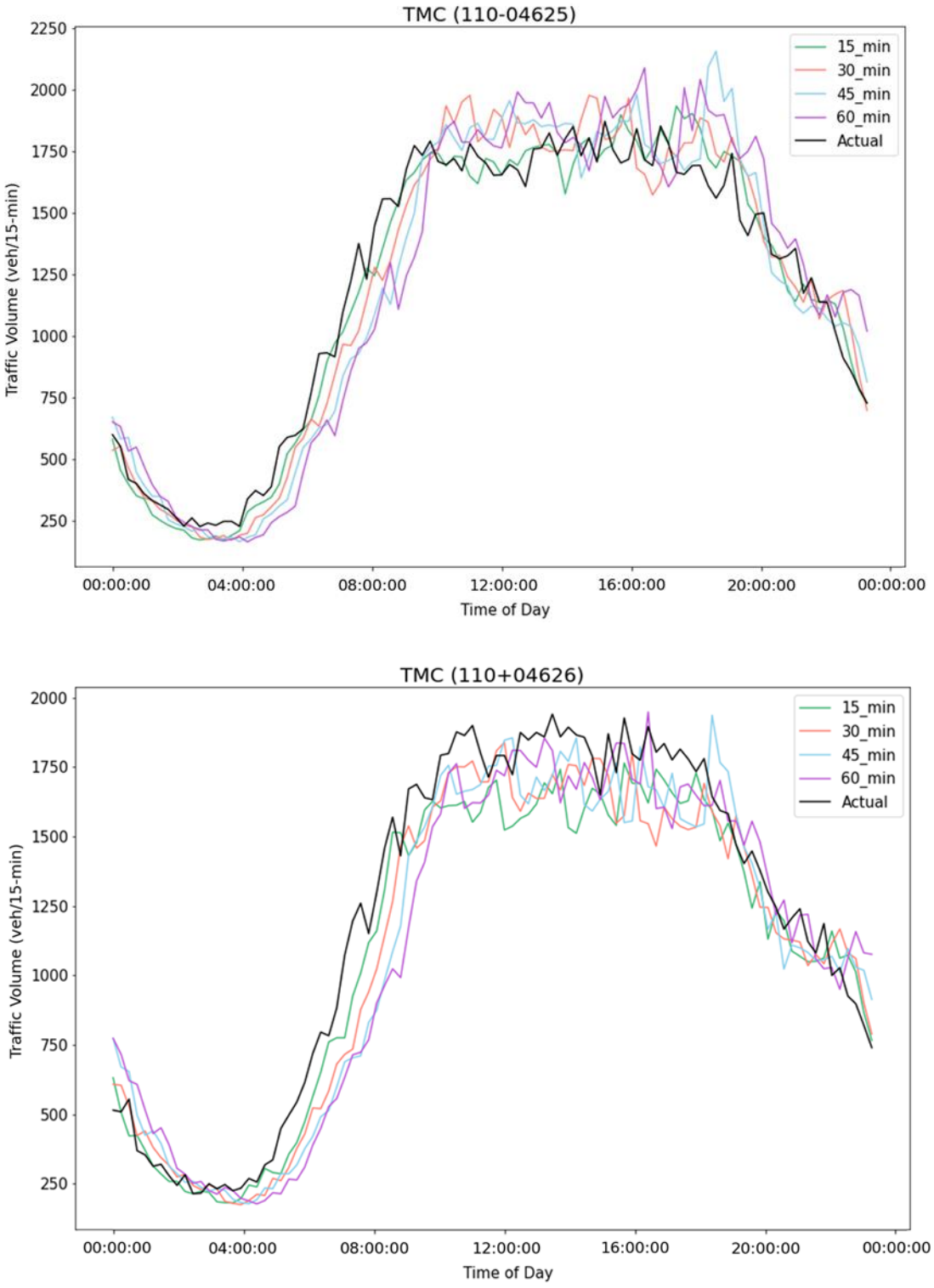


Figure 43. Daily traffic pattern samples predicted for Beltway area.

7.5 Chapter Summary

This chapter presented the results of applying the FSTGCN model and the ANN and XGBoost models to different regions in the Maryland NPMRDS road network. The results demonstrated the superiority of the proposed modeling framework relative to the XGBoost and ANN models, which on their own are capable state-of-the-art models in traffic volume estimation. Additionally, the framework is tested in the prediction of traffic volumes when there is a lag in reporting segment attributes such as speed profile. Further, the results of traffic volume estimation are aggregated over different functional road classes and various congestion levels, and it is shown that the superiority of the FSTGCN model is more evident in extreme conditions.

Chapter 8: Conclusions and Future Work

8.1 Research Summary and Contributions

This sturdy presented a graph-based model for networkwide traffic volume estimation. Traffic volume and speed are the two most fundamental inputs used by transportation agencies for quantifying traffic conditions, transportation system performance assessment, and cost-effective management of mobility projects and programs. While networkwide speed data are already available through data sources such as probe vehicle data, traffic volume remains a missing key in networkwide performance analysis.

The most common approach to compensate for the absence of traffic volume data is substituting it with the aggregate measure of AADT. The recent advancement in both available transportation large datasets and efficient pattern recognition algorithms provide the opportunity to estimate time-variant networkwide traffic volumes. However, the existing literature in this field lacks a comprehensive systematic framework for capturing spatio-temporal correlations that exist in a road network. The proposed framework aims to fill this gap by directly incorporating a graph representation of the road network in the volume estimation model.

Firstly, a two-step framework was developed to illustrate the significance of adding spatio-temporal features to the existing state-of-the-art traffic ANN developed for traffic volume estimation. This framework included selecting some CCSs based on

their location in the study network and adding their attributes as additional inputs to the ANN model. In this study, we illustrated how these additional features could improve the accuracy of the estimated volume in the case study network of New Hampshire state.

Encouraged by the findings of the proof-of-concept framework, this study proposed a graph-based methodology that directly incorporates a representative graph structure of the road network into the training process. This innovative methodology includes two main components of graph generation and model structure. This study's novel graph generation algorithm aims to build a systematic representation of a road network that considers its geometry and takes the existing trip patterns into account.

The innovative model architecture introduced in this study first uses the available data sources to extract correlations between the links' available features such as speed, road characteristics, temporal variables, etc., and traffic volumes by training a GCN-based model called STGCN. This model then goes through a fine-tuning process to consider the ongoing traffic condition in the road network while estimating volume for those links whose ground-truth volume data is not available. The fine-tuned model is called the FSTGCN model.

The FSTGCN model was analyzed by comparing its performance with two existing volume estimation models of ANN and XGBoost in various areas of Maryland state. The numerical results showed the significant improvement gained by using the introduced FSTGCN model for networkwide historical traffic volume estimation. In this study, APE and EMFR were used for models' performance analysis under various conditions. The overall results indicated 36 and 40 percent reductions in median APE

and EMFR obtained using FSTGCN instead of the state-of-the-art ANN model. These values are 42 and 27 percent for the average APE and EMFR, respectively. Additionally, the results illustrated more significant improvements when volume estimation is more challenging like on lower FRC roads, an unusual pattern in the network, or low congestion time intervals when speed is not a powerful indicator of the volume.

Considering the significant superiority of the FSTGCN model in historical volume estimation and its built-in structure, we expanded our analysis to estimate the FSTGCN model's performance for real-time traffic volume estimation. The results revealed the model's potential for real-time applications. The prediction accuracy measured by median APE and EMFR stayed under 20% and 7.5%, respectively, even when the most recent available data belongs to one hour before prediction time. Moreover, the fine-tuning processing time was calculated to be less than 10 seconds confirming its suitability for real-time applications.

As far as the overall computational cost of the models is concerned, FTGCN requires significantly more memory than ANN and XGBoost. This is because for training the STGCN part of the model, we need to read into memory the input features of the entire network on the whole study duration. In contrast, for ANN and XGBoost, we only input features of the CCSs to be read into memory. In this study, the largest network tested, the Beltway area network with 5157 OSM segments (i.e., 5157 nodes in the representative graph), required 70 GB of memory to read the entire 2019 data. Given the existing computational powers of today’s computers that go way beyond 70 GB,

the FSTGCN model can be applied to more extensive networks such as an entire state or country.

The proposed graph-based framework presented significant improvement over the existing methods and indicated its potential for real-time applications. However, some suggestions for expanding the model capabilities and practical implementations can be considered for future research. These suggestions are presented in the following section.

8.2 Potential Future Research

Several aspects of this study can be expanded in future works to result in more accurate traffic volume estimates. The following is a list of recommended directions for future research:

1. In this study, the attributes of the road segments are obtained from the NPMRDS network. However, since this network is a high-level network concerning the regional and statewide traffic performance, the connection links between the main road segments are missing; therefore, building a connected graph from this network requires additional steps and approximations. Developing the input data for a more granular network that includes the lower-level roads and, more importantly, the connection links can improve the model input data quality, thus increasing estimation accuracy.
2. Moreover, in this study, the weight matrix is assumed to be fixed during each considered period (e.g., AM-Peak, PM-Peak, or Off-Peak), a simplifying assumption. In reality, there can be variations in correlations between traffic

volumes at each time interval. However, updating the weight matrix of traffic volumes requires robust information on the turning movement patterns of vehicles in the network. The proposed framework can be expanded to estimate turning movements and update the representative graph weights iteratively. This way, the accuracy of the estimated volume increases, and turning movement patterns are evaluated simultaneously. Additionally, estimated volumes and turning movements can be combined to estimate OD patterns in the network.

3. The proposed methodology is designed to estimate historical volumes; thus, traffic volumes' temporal characteristics are captured by adding temporal features such as time of day, day of the week, month, etc. However, temporal features can be embedded in the model structure to predict the short-term traffic volumes with higher accuracy. There are GCN-based models developed explicitly for such dynamic tasks and can be combined with the findings of this study to build a framework designated for short-term traffic volume prediction. Advancing the proposed model for real-time applications can be beneficial in traffic management and operational strategies for congestion mitigation.
4. The computation of the weight matrix in this study was solely based on the movement of probe vehicles. However, the probe vehicle data is a small sample relative to the size of traffic volumes on the links. The relations between traffic volumes at different road network links can be explored from other perspectives, such as travel patterns from travel demand modeling frameworks, land-use distribution, and activity choice and scheduling of individuals. This

exploration can benefit the model in more robust estimation of weight matrix and, in turn, improved traffic volume estimation.

5. Another noteworthy application of the volume estimation models is the management of traffic operations when a sudden disruption, for instance, resulting from an accident, occurs in the network. These models can be employed to estimate and predict the traffic volume in the road links impacted by the incident, such as those used by rerouting vehicles. Given that the FSTGCN model gets the most recent speed data of the entire study area as input, it can detect the flow disruption as long as it directly impacts the speed in the area. However, we need ground truth data in adjacent links to investigate this more precisely in future works.
6. One other possible direction for future work is to explore the effect of CCSs locations on the model accuracy. As mentioned in chapter 6, this investigation is feasible if a denser network of count stations is available. Given such data, different settings of CCS locations can be selected for training and testing the model. The model performance in various links can be compared when nearby CCS data is fed into the model against when this data is not provided to the model.
7. Last but not least, a sensor placement optimization model can be developed on top of the introduced framework to investigate the impacts of CCS locations on networkwide volume estimation accuracy and optimize it accordingly. This optimization scheme enables the authorities to strategically plan the traffic count sensors placement in the network and improve the efficiency of the data

collection. A carefully planned sensor placement scheme improves the network observability and information gains through observations of the traffic count data. The currently available ground truth data is limited to a few existing CCSs, which is not enough for solving the sensor placement optimization problem. However, accessing a more widespread ground truth volume data or designing a simulation framework to simulate networkwide volumes can be the directions to solve the sensor allocation problem in the context of networkwide volume estimation.

References

AASHTO (2001). Policy on geometric design of highways and streets. American Association of State Highway and Transportation Officials, Washington, DC.

Aldrin, M. (1995). A statistical approach to the modelling of daily car traffic. *Traffic engineering & control*, *36*(9), 489-493.

Atwood, J., & Towsley, D. (2016). Diffusion-convolutional neural networks. In *Advances in neural information processing systems* (pp. 1993-2001).

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bruna, J., Zaremba, W., Szlam, A., & LeCun, Y. (2013). Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*.

Castro-Neto, M., Jeong, Y., Jeong, M. K., & Han, L. D. (2009). AADT prediction using support vector regression with data-dependent parameters. *Expert Systems with Applications*, *36*(2), 2979-2986.

Chen, C., Petty, K., Skabardonis, A., Varaiya, P., & Jia, Z. (2001). Freeway performance measurement system: mining loop detector data. *Transportation Research Record*, *1748*(1), 96-102.

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).

Clevert, D. A., Unterthiner, T., & Hochreiter, S. (2015). Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*.

Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, *29*, 3844-3852.

Ding, A., Zhao, X., & Jiao, L. (2002, September). Traffic flow time series prediction based on statistics learning theory. In *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems* (pp. 727-730). IEEE.

Eom, J. K., Park, M. S., Heo, T. Y., & Huntsinger, L. F. (2006). Improving the prediction of annual average daily traffic for nonfreeway facilities by applying a spatial statistical method. *Transportation Research Record*, *1968*(1), 20-29.

FHWA (Federal Highway Administration). (2016). Traffic Monitoring Theory, Technology and Concepts. *Traffic Monitoring Guide.*

FHWA (Federal Highway Administration). (2019). Traffic Analysis Toolbox Volume III: Guidelines for Applying Traffic Microsimulation Modeling Software 2019 Update. https://ops.fhwa.dot.gov/publications/fhwahop18036/index.htm

FICO Xpress 8.5.3.
http://www.fico.com/en/Products/DMTools/xpressoverview/Pages/Xpress Optimizer.aspx

Fricker, J.D., Saha, S.K. (1987). Traffic Volume Forecasting Methods for Rural State Highways. *FHWA/IN/JHRP-86-20. Purdue University, West Lafayette.*

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, *28*(2), 337-407.

Gallicchio, C., & Micheli, A. (2010, July). Graph echo state networks. In *The 2010 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE.

Gori, M., Monfardini, G., & Scarselli, F. (2005, July). A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.* (Vol. 2, pp. 729-734). IEEE.

Haklay, M., & Weber, P. (2008). Openstreetmap: User-generated street maps. *IEEE Pervasive computing*, *7*(4), 12-18.

Herrera, J.C., Bayen, A.M. (2008). Traffic flow reconstruction using mobile sensors and loop detector data. *TRB 87th Annual Meeting Compendium of Papers, Transportation Research Board of the National Academies, Washington DC, 13–17 January 2008.*

Herzmann, D., Arritt, R., and Todey, D. (2004). Iowa environmental mesonet, Available at: https://mesonet.agron.iastate.edu/. Iowa StateUniv., Dep. of Agron., Ames, IA, Accessed March 2019

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Jagadish, H. V., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J. M., Ramakrishnan, R., & Shahabi, C. (2014). Big data and its technical challenges. *Communications of the ACM*, *57*(7), 86-94.

Jiang, W., & Luo, J. (2021). Graph neural network for traffic forecasting: A survey. *arXiv preprint arXiv:2101.11174*.

Khan, S. M., Islam, S., Khan, M. Z., Dey, K., Chowdhury, M., Huynh, N., & Torkjazi, M. (2018). Development of statewide annual average daily traffic estimation model from short-term counts: A comparative study for South Carolina. *Transportation Research Record*, *2672*(43), 55-64.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Levie, R., Monti, F., Bresson, X., & Bronstein, M. M. (2018). Cayleynets: Graph convolutional neural networks with complex rational spectral filters. *IEEE Transactions on Signal Processing*, *67*(1), 97-109.

Li, W., Wang, X., Zhang, Y., & Wu, Q. (2021). Traffic flow prediction over muti-sensor data correlation with graph convolution network. *Neurocomputing*, *427*, 50-63.

Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*.

Marković, N., Sekuła, P., Vander Laan, Z., Andrienko, G., & Andrienko, N. (2018). Applications of trajectory data from the perspective of a road transportation agency: Literature review and Maryland case study. *IEEE Transactions on Intelligent Transportation Systems*, *20*(5), 1858-1869.

Micheli, A. (2009). Neural network for graphs: A contextual constructive approach. *IEEE Transactions on Neural Networks*, *20*(3), 498-511.

NPMRDS. (2018), National performance measurement research dataset (NPMRDS), https://ops.fhwa.dot.gov/perf_measurement/index.htm.

Papageorgiou, M., Papamichail, I., Messmer, A., & Wang, Y. (2010). Traffic simulation with METANET. In *Fundamentals of traffic simulation* (pp. 399-430). Springer, New York, NY.

QGIS Development Team (2018). QGIS Geographic Information System. Open Source Geospatial Foundation Project. http://qgis.osgeo.org

Rossi, R., Gastaldi, M., & Gecchele, G. (2014). Comparison of clustering methods for road group identification in FHWA traffic monitoring approach: Effects on AADT estimates. *Journal of Transportation Engineering*, *140*(7), 04014025.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2008). The graph neural network model. *IEEE transactions on neural networks*, *20*(1), 61-80.

Schrank, D., Eisele, B., Lomax, T., Bak, J. (2015). Appendix A: Methodology for the 2015 urban mobility scorecard, *Technical report, Texas Transportation Institute, Texas A&M University.* <https://static.tti.tamu.edu/tti.tamu.edu/documents/mobility-scorecard-2015-appx-a.pdf>.

Sekuła, P., Marković, N., Vander Laan, Z., & Sadabadi, K. F. (2018). Estimating historical hourly traffic volumes via machine learning and vehicle probe data: A Maryland case study. *Transportation Research Part C: Emerging Technologies*, *97*, 147-158.

Sharma, S., Lingras, P., Xu, F., & Kilburn, P. (2001). Application of neural networks to estimate AADT on low-volume roads. *Journal of Transportation Engineering*, *127*(5), 426-432.

Shimizu H, Yamagami K, Watanabe E. (1998). Applications of state estimation algorithms to hourly traffic volume system. *Proceedings of the Second World Congress on ITS, Yokohama, p. 72–7.*

Sperduti, A., & Starita, A. (1997). Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks*, *8*(3), 714-735.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems* (pp. 3104-3112).

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015, May). Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web* (pp. 1067-1077).

UK Highways Agency, Design Manual for Roads and Bridges (1996), Volume 12, Section 2, https://webarchive.nationalarchives.gov.uk/20140116153456/http://www.dft.gov.uk/ha/standards/dmrb/index.htm

Vander Laan, Z., and Farokhi Sadabadi, K. (2019). Automated Conflation Methodology to Associate OpenStreetMap Road Characteristic Data with a Traffic Message Channel Base Map, Submitted for publication.

Wang, X., & Kockelman, K. M. (2009). Forecasting network data: Spatial interpolation of traffic counts from texas data. *Transportation research record*, *2105*(1), 100-108.

Work, D. B., Tossavainen, O. P., Blandin, S., Bayen, A. M., Iwuchukwu, T., & Tracton, K. (2008, December). An ensemble Kalman filtering approach to highway traffic estimation using GPS enabled mobile devices. In *2008 47th IEEE Conference on Decision and Control* (pp. 5062-5068). IEEE.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Philip, S. Y. (2020). A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, *32*(1), 4-24.

Yao, H., Wu, F., Ke, J., Tang, X., Jia, Y., Lu, S., ... & Li, Z. (2018, April). Deep multi-view spatial-temporal network for taxi demand prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 32, No. 1).

Yi, Z., Liu, X. C., Markovic, N., & Phillips, J. (2021). Inferencing hourly traffic volume using data-driven machine learning and graph theory. *Computers, Environment and Urban Systems*, *85*, 101548.

Yu, B., Yin, H., & Zhu, Z. (2017). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*.

Zahedian, S., Sekuła, P., Nohekhan, A., & Vander Laan, Z. (2020). Estimating hourly traffic volumes using artificial neural network with additional inputs from automatic traffic recorders. *Transportation Research Record*, *2674*(3), 272-282.

Zhang, J., Zheng, Y., & Qi, D. (2017, February). Deep spatio-temporal residual networks for citywide crowd flows prediction. In the *Thirty-first AAAI conference on artificial intelligence*.

Zhang, Q., Jin, Q., Chang, J., Xiang, S., & Pan, C. (2018, August). Kernel-weighted graph convolutional network: A deep learning approach for traffic forecasting. In *2018 24th International Conference on Pattern Recognition (ICPR)* (pp. 1018-1023). IEEE.

Zhao, F., & Chung, S. (2001). Contributing factors of annual average daily traffic in a Florida county: exploration with geographic information system and regression models. *Transportation research record*, *1769*(1), 113-122.

Zhao, F., & Park, N. (2004). Using geographically weighted regression models to estimate annual average daily traffic. *Transportation research record*, *1879*(1), 99-107.

Zhong, M., Lingras, P., & Sharma, S. (2004). Estimation of missing traffic counts using factor, genetic, neural, and regression techniques. *Transportation Research Part C: Emerging Technologies*, *12*(2), 139-166.