공학박사학위논문

A Recurrent Neural Network for Estimating Speed Using Probe Vehicle Data in Urban Traffic Network

프로브 차량 자료를 이용한 도시교통 네트워크의 속도 추정 순환형 신경망 모형

2020년 2월

서울대학교 대학원

공과대학 건설환경공학부

양재환

Abstract

Urban traffic flows are characterized by complexity. Due to this complexity, limitations arise when using models that have commonly been using to estimate the speed of arterial road networks. This study analyzes the characteristics of the speed data collected by the probe vehicle method in links on the urban traffic flow, presents the limitations of existing models, and develops a modified recurrent neural network model as a solution to these limitations. In order to complement the limitations of existing models, this study focused on the interrupted flow characteristics of urban traffic. Through data analysis, we verified the separation of platoons and high-frequency transitions as phenomena in interrupted flow. Using these phenomena, this study presents a two-step model using the characteristics of each platoon and the selected dropout method that applies traffic conditions separately. In addition, we have developed an active imputation method to deal with frequent missing data in data collection effectively. The developed model not only showed high accuracy on average, but it also improved the accuracy of certain states, which is the limitation of the existing models, increased the correlation between the estimated value and the estimated target value, and properly learned the periodicity of the data.

Keyword : Data Estimation, Deep learning, Recurrent Neural Network, Probe Vehicle, Traffic Speed

Student Number : 2016-30289

Table of Contents

Chapter 1. Introduction

1.1. Study Background and Purpose

1.1.1. Importance of traffic speed estimation

Traffic systems are often expressed in networks, and the average characteristics of road segments are expressed in the properties of links, the most basic unit of the network. The flow properties specific to the link represent the state of the current system, which serves as a basis for understanding the current situation, providing it to the user, and further developing and improving short-term operation and long-term planning of the transportation system. Many studies have been conducted to estimate traffic conditions and traffic information on links and provide them to drivers or to establish traffic policies and strategies.

The average speed data of a link is the most basic data of traffic. Average speed, along with average travel time, is one of the most intuitive data available and is the most easy-to-access to drivers and operators. It is also the most objective indicator of the performance that a link currently shows. This study aims to estimate these speed data accurately.

In particular, the estimation accuracy of these speed data is more critical for urban traffic flow. Many countries with advanced transport infrastructures are in intensive urban development. In South Korea, the proportion of the urban population is 91.84% in 2018, and the proportion is steadily increasing. According to the mileage statistics of the Ministry of Land, Infrastructure and Transport, the driving distance of urbanized areas is 88.16%. In addition to a large amount of traffic in the urban area itself, the traffic to the arterial road that handles inter-regional traffic also passes through the urban traffic network, so urban traffic flow plays a fundamental role in all motor traffic in highly urbanized countries. However, these urban traffic flows have very complicated characteristics compared to the arterial road network. As a result, estimating specific data on urban traffic flows is a challenging task.

Nevertheless, estimation of the properties of urban traffic flows requires higher accuracy than arterial roads. There are several reasons for this. The most important reason is that the data estimation of urban traffic flows causes a small change in user route selection. This property is due to the complexity of the urban traffic network, which is different from the arterial road network. Also, in the case of urban traffic flows, frequent changes of state may occur, and the state estimation of traffic flows may change even with a small difference. In addition, since urban networks have a shorter segment length, and many segments are included in one path than the arterial road network, a small difference of one segment can be overlapped and amplified several times. For these reasons, the estimation of urban traffic flow requires a high level of accuracy.

In addition, investment to the existing loop type detector system is gradually

being reduced in policy, and efforts are being made to link it with private data and supplement it with new detection systems such as DSRC (Dedicated Short-Range Communications). However, for these newly applicated systems, much missing information is generated due to the properties of the system. This missing information reduces the accuracy of the estimation models, and at the same time, becomes the research objective of the estimation models themselves.

Existing studies often use traffic speed or travel time estimation models that were previously developed in other areas without any model modification. This approach has the disadvantage of providing no clue to the dynamics of traffic data, even if it guarantees high accuracy. Therefore, it is necessary to explore how the dynamics of traffic data affect this approach, with the introduction of the latest method of ensuring high accuracy.

1.1.2. Property of Urban Network Speed Data

Most of the traffic data are generally collected in units of points or segments on the road, usually represented as a link. In the case of such traffic data, there are two main characteristics.

The first is the time-series feature. Traffic data itself has characteristics as time-series, and the characteristics of the previous time affect the next time. It can also show regular patterns depending on the time of day. In particular, home-based traffic such as commuting is characterized by regular patterns over time.

The second is that it is affected by the traffic flow conditions from the nearby links. (Daganzo 1994) The traffic conditions of specific links depend on the shock waves propagated from the nearby links. Especially for arterial roads, such propagation characteristics are well represented, and many studies have been conducted for these kinds of roads.

In the case of urban networks, one more attribute is added here. Unfortunately, for urban networks consisting of a set of interrupted flows, the complexity limits the exact estimates (Vlahogianni et al. 2014). This complex correlation of traffic data is influenced not only by the spatiotemporal relationship between links, but also by many factors such as hierarchy, traffic volume, location characteristics, and traffic flow propagation. Naturally, speed data on urban networks also exhibit this complexity.

1.1.3. Problems of Speed Data Estimation on Urban Traffic

There are two problems in estimating the properties of urban traffic flows. The first problem is the estimation accuracy. Data estimation on urban traffic networks requires higher accuracy than that of regional arterial networks, but due to the characteristics of urban traffic flows, estimation often shows low accuracy. This low accuracy occurs for the following reasons. First, urban traffic flows have a low correlation between the data of links due to complexity. This low correlation results in lower forecastability between link information (Park et al. 2019). Second, propagation between links can be blocked due to traffic signals or intersections. In the case of traffic signals, the conditions between the links may vary due to signal operation and offset. Changes in signal operation can also cause different conditions overtime on the same link. Finally, urban traffic flows show more frequent state transitions than arterial flows, which can cause bias of data. Due to the frequent change of traffic states, it is more likely that the effects of false estimations will continue to be more effective than continuous flow.

The second problem is the high frequency of missing data. Urban traffic systems typically collect speed or travel time data by means of probe vehicles, which often cause missing data. A typical reason is the failure of data transmission/reception devices such as beacons receiving information of probe vehicles and terminals mounted on vehicles. In addition, the sample may not be sufficient due to the small traffic volume, and an error may occur in the received information.

Missing data and resulting information leaks are affected not only by the equipment itself but also by the current traffic conditions on the road. Probe vehicle data generated by commercial GPS information has also generated a large amount of missing data. Missing data generated through the above problems not only lowers the reliability of the data but also reduces the accuracy of the data estimation itself. In particular, in machine learning that requires a large amount of data, a problem may occur in that the number of data samples decreases.

1.1.4. Property and Limitation of Deep Learning

Deep learning is one of the most actively studied techniques in terms of data estimation. Deep learning models are widely studied in many areas as well as data estimation through their high accuracy.

Ever since the discovery of algorithms for learning through a Deep Belief Network studied by Hinton, deep learning has emerged as the most central algorithm in data estimation and machine learning. (Hinton et al. 2006) Many improvements have been made since then and are now being introduced to the transportation sector, but the number of studies is still limited(Nguyen et al. 2018).

Deep learning has the advantage of high accuracy, but it has the disadvantage that it is challenging to grasp the causality of input and outcome data. In particular, the characteristics of deep learning techniques that continuously abstract data face the problem of not being able to grasp the context as well known as the “Black Box

Problem". The disadvantage is that the analysis of relationships is almost impossible.

In addition, if there are missing values dependent on the data, the result may vary depending on the processing method. As a similar problem, the same feature as other machine learning techniques requires a great deal of data, as called "Big Data," which is a disadvantage of the deep learning model.

1.1.5. Purpose of the Study

This study aims to develop an accurate traffic speed estimation model for links on urban networks, which consists of interrupted flows that are difficult to estimate due to complexity. To do this, we analyze the characteristics of the interrupted flow speed data that makes it hard to apply general linear combination models. Furthermore, we develop a deep learning model that is expected to be the most accurate. In particular, we develop a model that can apply the properties of the interrupted flow, not merely apply some deep learning models. In addition, this study aims to produce a model that can be estimated for frequent missing data.

1.2. Research Scope and Procedure

1.2.1. Research Scope

The primary purpose of this study is to estimate the average speed data on links of urban roads as described above. Generally, urban road networks consist of interrupted flows. Because of this, this study precedes the analysis of interrupted flow to develop the model that can reflect characteristics of the interrupted flow. At the same time, it was found that the existing model could not adequately reflect the characteristics of interrupted flow. The main methodology for the estimation is the modified model of the recurrent neural network (RNN). Continuously, this study designs novel RNN cells to reflect the characteristics of interrupted flow.

The data used in this study are some part of the data collected from the DSRC system in Daegu between January and June 2018.

1.2.2. Research Composition and Procedure

As shown in Figure 1.1, this paper proceeds through five steps. This paper consists of six chapters, including the introduction.

In Chapter 2, the direction of the study was explored by analyzing the existing research. First, the existing studies related to data estimation was searched. These

studies include both traditional and machine learning techniques. Next, studies on traffic flow analysis were reviewed. Through the review, the existing research comprehensively analyzed, and the original contribution of the study was explained.

In Chapter 3, this study describes the DSRC data of Daegu, which is the data used in this study. Then, this study analyzed probe vehicle data collected through the DSRC system. Through this analysis, the characteristics of the data were identified. Simultaneously, this study was able to find out the result of applying the existing model and the phenomenon caused by the separation of platoons.

In Chapter 4, we describe a novel recurrent neural network method that modified the existing GRU to suit the characteristics of interrupted flow and data collected from the DSRC probe vehicle system.

In chapter 5, this study shows the improved performance of the model developed in several aspects. In addition to accuracy, it describes in detail how the existing model overcomes the limitations for each state.

Chapter 6 summarizes the analysis so far, presents the limitations of this study, and suggests future research directions.

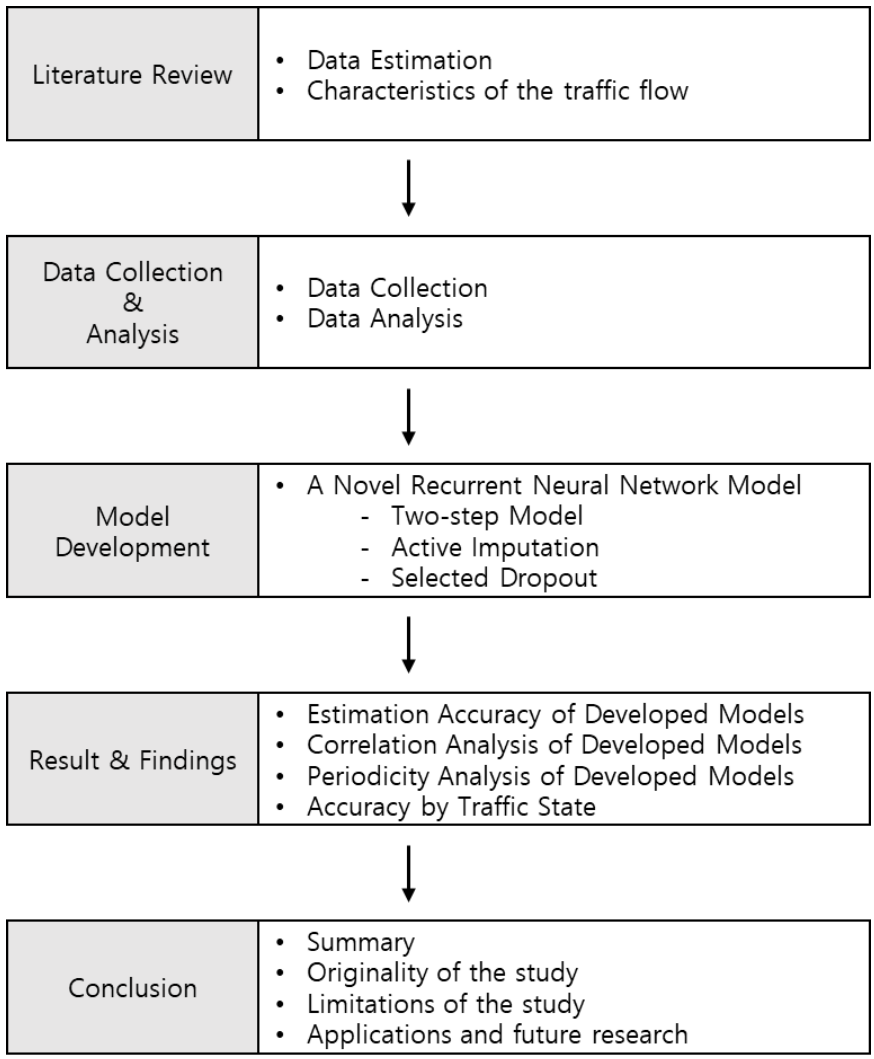


Figure 1. 1 Research Procedure

Chapter 2. Literature Review

2.1. Data Estimation

2.1.1. Traditional Estimation Methods

Traditional estimation models have been developed based on mathematical models, not data-driven. There is a representative model based on the kinematic wave proposed by Newell (Newell 1993). Also, a model using cellular automata on the highway (Nagel and Schreckenberg 1992), and the CTM(Cell Transmission Model) studied by Daganzo has been used (Daganzo 1994). Pan is an advanced form of CTM that studies mathematical models by selecting multivariate normal distribution-based predictors as a sub-system of the Stochastic Cell Transmission Model (SCTM). Also, there is a model that predicts real-time traffic volume using spatio-temporal correlation(Min and Wynter 2011).

In the case of data-driven research, since traffic data has traditionally been treated as a part of time-series data, studies on time-series waveform analysis such as Kalman filter and ARIMA (AutoRegressive Integrated Moving Average) have been conducted. The Kalman filter is a filter that estimates linear dynamics based on measurements that contain noise. Often used for time series prediction, a number of

studies have been conducted(Chen and Grant-Muller 2001; Chien et al. 2003; Van Lint 2008; Wang et al. 2006). ARIMA is also a time-series analysis tool that analyzes abnormal time series data using the differential form of the ARMA model. Many studies have also been conducted on this.

Table 2. 1 Traditional Estimation Methods

Author(s)	Year	Method	Target of Estimation
Newell	1993	Kinematic Wave	Highway Traffic
Nagel and Schreckenberg	1992	Cellular Automata	Freeway Traffic
Daganzo	1994	Cell Transmission Model	Highway Traffic
Pan	2013	Modified Stochastic Cell Transmission Model	Short-term Traffic State
Min and Wynter	2011	Spatio-temporal correlation analysis	Road Traffic
Chen and Grant-Muller	2001	Kalman Filter	Short-term Traffic Flow
Chien et al.	2003	Kalman Filter	Travel Time
Van Lint	2008	Kalman Filter	Travel Time
Wang et al.	2006	Kalman Filter	Traffic Flow
Chandra and Al-Deek	2009	ARIMA	Freeway Traffic Speeds
Smith et al.	2002	ARIMA	Traffic Flow
Williams and Hoel	2003	ARIMA	Traffic Flow
Ni et al.	2005	Markov Chain	Highway Traffic

2.1.2. Machine Learning Models

Machine learning can be classified into three categories: statistical, geometric, and regressive. Regression machine learning involves Deep learning, and most of the recent work involved deep learning.

2.1.2.1. Statistics Based Machine Learning Models

The most representative example of statistical machine learning is the Bayesian network model. It is a model based on Bayes' theorem and has been spotlighted as a model that enables probabilistic explanations along with decision trees in machine learning models.

Sun used the Bayesian network to predict traffic flows (Sun et al. 2006). In the 2018 study, Park conducted a study to predict the breakdown of traffic flows based on the Bayesian network (Park 2017).

2.1.2.2. Geometric Based Machine Learning Models

The geometric machine learning model is the simplest and has the highest reliability with high accuracy. Models in this category typically include k-nearest neighbors and support vector regression (SVR). The models estimate the data through the data that is geometrically closest to the situation to be estimated (k-

nearest neighbors) or find the hyperplane between the data to estimate (SVR). The K-Nearest Neighbor technique uses the average of k data that is geometrically closest to the current data to be predicted, and because it has no special analytical meaning, it does not have academic content and is typically not used for estimation nowadays. But it often used for comparison with models(Yu et al. 2011).

SVR is a regression analysis using the support vector machine (SVM). The support vector machine was developed by Cortes(Cortes and Vapnik 1995). This model is a well-defined machine learning model that was used mainly before the neural network was in the spotlight. As described above, the hyperplane through data is obtained and the model is used to grasp the characteristics of the data, and it is used for the prediction of short-term travel time through the regression analysis technique(Wu et al. 2004; Zhang and Xie 2007).

2.1.2.3. Deep Learning Models

As same as mentioned, it has been since 2006 that the deep learning model represented by the Artificial Neural Network has been in the spotlight, but it is not recent that the Neural Network has been applied to prediction or estimation of traffic data. Dongjoo Park used the Modular Neural Network for link travel time prediction and applied the Spectral Basis Neural Network(Park et al. 1999; Park and Rilett 1998). Byungkyu Park also used the Neural Network with backpropagation for traffic prediction and took good results(Park et al. 1998).

Since then, various neural networks have been introduced in traffic research.

The main models used are deformation models of recurrent neural networks that feedback outcome results. Models in this category are the Elman Neural Network (Elman NN), also known as the Simple Recurrent Network, and its variant, the State Space Neural Network (SSNN)(Ishak et al. 2003; Van Lint 2008).

Besides, Time Delay Neural Network (TDNN) and Nonlinear AutoRegressive with eXogenous inputs Neural Network (NARX NN) also have been used. In the case of TDNN, it is not a recurrent model but a neural network that takes current and previous time series values as inputs together(Lingras and Mountford 2001). NARX NN is a nonlinear autoregressive method that takes inputs from other exogenous time series values(Fusco et al. 2015).

The commonly used RNN model, like other deep neural networks, has shown poor performance for long-term data due to the gradient vanishing problem (Hochreiter 1998). To solve this problem, recurrent neural networks that customize RNN's memory term appear. It was. Representative models of this type are Long Short-term Memory (LSTM) and Gated Recurrent Unit (GRU). LSTM is a model developed by Hochreiter, and it is used in many fields by innovatively progressing the problem of RNN(Hochreiter and Schmidhuber 1997). GRU is a simplified version of LSTM's processing gate and is one of the most widely used RNN models with LSTM(Cho et al. 2014).

Research into applying this to traffic data is also actively underway. In Ma's study, LSTM was used for the traffic speed prediction study, and Fu's study introduced LSTM and GRU in the traffic flow prediction, showing better performance than ARIMA. Many studies have also used GRU and LSTM to predict

traffic flow and show better performance than previous models(Fu et al. 2016; Zhang and Kabuka 2018).

2.1.3. Other Estimation Models

Further studies have been conducted to estimate traffic data through signals collected from mobile data and others(de Fabritiis et al. 2008; Iqbal et al. 2014).

Table 2. 2 Machine Learning and other estimation methods

Author(s)	Year	Method	Target of Estimation
Sun et al.	2006	Bayesian Network	Traffic flow
Park	2017	Bayesian Network	Traffic State
Lee	2018	k-NN, SVM	Road Condition
Yu et al	2011	SVM, ANN, k-NN	Bus Arrival time
Wu et al.	2004	SVR	Travel time
Zhang and Xie	2007	SVR	Freeway Volume
Park et al.	1999	MNN	Travel time
Park and Rilett	1998	SNN	Travel time
Park et al.	1998	RBFNN	Traffic volume
Ishak et al.	2003	Elman NN	Travel Speed
Van Lint	2008	SSNN	Travel Time
Lingras and Mountford	2001	TDNN	Traffic volume
Fusco et al.	2015	NARX NN	Travel Speed
Ma et al.	2015	LSTM	Travel Speed
Fu et al.	2016	GRU, LSTM	Traffic Flow
Zhang and Kabaka	2018	GRU	Traffic Flow
De Fabritiis et al.	2008	Mobile data	OD Matrix
Iqbal et al.	2015	Mobile data	Traffic Volume

2.2. Traffic State Handling

Traffic flow state is difficult to estimate with only one type of data. Therefore, the traffic flow state is determined by considering various data in time or space. The classification of traffic flow conditions varies from two states to five states. There is also a methodology for determining breakdowns issued by unstable flows known to exhibit different dynamics from normal flows.

Traffic flow is converted to Unstable flow when “Break down” occurs in normal flow, which is thought to occur when demand exceeds capacity. (Kondyli et al., 2013)

For flow modeling, there are mainly a single regime model and a two-regime model. In the two-regime model, stable flows and unstable flows are modeled separately (Yao et al., 2009).

Treiber's study smoothed the traffic speed data using an adaptive kernel that reflects the mutual spatio-temporal effects of traffic flow. (Treiber et al., 2003) The model allows an indirect understanding of the impact of the neighbor links.

Traffic flow state determination uses more machine learning methods such as Markov chain and Clustering than traditional methodologies. In Xia et al. (2012), traffic flow state was determined by the clustering method, and in Dong and Mahmassani, Noroozi and Hellinga et al., Traffic state was determined by spatiotemporal Markov modeling. In some studies, the traffic state is clustered using only single data such as demand and speed, and the Jenks natural break method is used for one-dimensional clustering data. (Wen et al. 2017; Wu and Hung 2010)

The influence of data exchange between links can be determined by predictability. Forecastability strongly influenced by the variability of the data to be predicted. As a study of this inherent volatility, Georg developed the Forecastable Component Analysis technique. In this study, we defined the spectral density according to the time series and defined the Omega Value using Shannon entropy (Goerg 2013).

Yue's work was on the mutual Forecastability of each data. In this study, the time lag cross-correlation function (CCF) was used. CCF is an indicator of linear dependence between data and can have high predictive power between strongly related data. (Yue and Yeh 2008) Park indexed inherent volatility and mutual predictability. (Park et al. 2019)

Many studies have cited platoon separation as a hallmark of interrupted flow. (Akcelik 1996; Gartner et al. 1992; Yang et al. 2014) In these studies, traffic signals were identified as being divided into queued and unqueued vehicles by traffic signals, and traffic flow analysis was performed using them.

Table 2. 3 Traffic State Handling

Author(s)	Years	Method	Contents
Treiber and Kesting	2003	Adaptive smoothing	Spatio-temporal smoothing for speed
Dong and Mahmassani	2009	Markov Model	Breakdown identification
Wu and Liu	2011	Simulating Traffic Behavior	Traffic state
Xia et al.	2012	Clustering	Traffic state
Wu and Hung	2010	Jenks Natural Break	Traffic state
Wen et al.	2017	Jenks Natural Break	Traffic state
Gartner et al. (TRB)	1992	Platoon Separation	Traffic state
Akcelik	1996	Traffic Dynamics	Travel Time, Density, Speed
Yang et al.	2014	Travel Time Distribution	Travel Time
Noroozi and Hellinga	2014	Markov Model	Breakdown identification
Goerg	2013	Forecastability Component Analysis	Internal variability
Yue and Yeh	2008	Time-lag Cross-correlation Function	Forecastability by time-lag CCF
Park	2018	Bayesian Network	Breakdown forecast

2.3. Originality of This Study

This paper analyzes why the existing estimation models do not fit the urban network speed estimation. Subsequently, this paper develops and proposes a new estimation model. The originality of this study distinguished from previous studies is below.

1. Development of speed estimation model that can internally handle missing data

This study developed a model that can internally handle the missing data frequently in the probe vehicle system. To solve the frequent missing data problem, we developed the active imputation method that combines neural networks and random forest. By using the method, speed estimation is possible with data which is including missing or empty cells.

2. Improved estimation by applying the platoon separation

In complementing the limitations of the existing model, this study divides the data into periodic and aperiodic data by using the platoon separation, which is a common phenomenon in interrupted flow. This method solves the period reverse problem in the learning construction of LSTM and secures robustness against sudden change in speed.

3. Development of a model robust for traffic states

Interrupted flows have a high transition probability, unlike uninterrupted flows. For this reason, if the time lag occurs in the estimation, the expected error is larger than the interrupted flow. In this study, we proposed a methodology for solving the transition probability of each situation to secure the robustness of the transition state. The methodology consists of combining neural networks with the developed RNN model to perform traffic state adjustment.

Chapter 3. Data Collection and Analysis

3.1. Terminology

Below are some of the less common terms used in this paper.

- Low-Performance Platoon (LPP): A group of vehicles that exhibit a lower speed than the specified threshold during the same time.
- High-Performance Platoon (HPP): A group of vehicles that have a higher speed than a specified threshold during the same time
- Low-performance Platoon ratio (LPR): Counts of LPP / Counts of all OBE
- Naive Model: A model that uses only speed values as I / O values
- One Step Model: Model using LPP speed, HPP speed, LPR as input values and average speed as the output value
- Two-Step Model: A model that predicts LPP speed, HPP speed, and LPR with I / O values first and then calculates average speed using them.
- Internal missing imputation: Implement imputation on missing data to make missing data available as the input value
- Decaying Method: The method calculated by converging to the global average value over time when proceeding with internal missing imputation.
- Active Imputation: The method calculated by learning the ratios between the last observed values and the values calculated from other models when performing internal missing imputation.

3.2. Data Collection

3.2.1. Collected Data

This study used DSRC-based data collected from Daegu Metropolitan City from January to June 2018. The data records the recording time, vehicle speed, vehicle ID, and link passing time for each link. The data was converted into a one-minute data frame, and about 250,000 lines of data were used.

There are 642 links in total, and each link has a high data shadowing rate of about 50% on average. A total of 13 eastbound links in Dalgubeol, the main arterial road, were analyzed. In the case of Dalgubeol-daero, the average data shading rate is 34%.

In this study, we analyzed the data from 06:00 to 22:00 and removed the late-night data when the dynamics of traffic flow had no significant meaning for analyzing. In this case, the average missing rate is 15.4%.

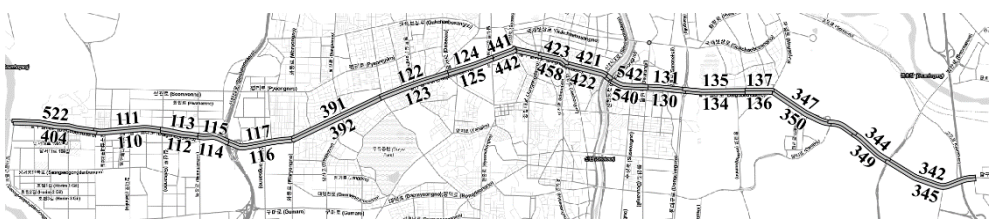


Figure 3. 1 Locations of the Data Collecting Area

Table 3. 1 Link Properties of the Data Collecting Area

Link Number	Traffic Signals*	Average Length (m)	Average Speed (km/h)	Signal Cycle (Sec.)		Lanes	Remarks
				Normal	Late-night		
112	1	779.8	25.447	180	200	5	-
114	0	583.2	28.708	180	200	5	Highway Ramp at end
116	1	1410.0	43.640	180	200	6	Highway Ramp at start
392	7	2017.4	29.388	180	200	4+1 (Bus lane)	Bus lane (At Peak)
123	4	1412.0	30.719	180	200	5	-
125	2	984.4	30.109	180	200	5	-
442	1	512.7	65.368	180	200	5	-
546	1	530.3	41.728	180	200	5	-
458	1	583.6	48.655	180	200	5	-
422	3	805.9	30.941	180	200	5	-
540	2	1071.6	30.587	180	200	4+1 (Bus lane)	Bus lane (At Peak)
130	2	775.8	42.660	180	200	4+1 (Bus lane)	Bus lane (At Peak)
350	4	1387.3	55.714	180	200	4+1 (Bus lane)	Bus lane (At Peak)

*: Both sides exclude

3.2.2. Data Collection Properties

The DSRC data used in this study is collected by vehicles equipped with On-Board Equipment (OBE). The passing time is recorded as the vehicle equipped with the OBE passes near the roadside equipment (RSE). In the DSRC system, this record time is used to calculate the link pass rate.

OBE is mounted on the electronic toll collection system (ETCS) terminal unit. The penetration rate of ETCS terminals is 80.6% as of 2018. Therefore, data collection rates for passenger cars and chartered buses are sufficient.

The system records a time record based on the first RSE range access time. The estimated speed is recorded based on the difference in time record between the length of the intersection and the RSE recorded in advance. The data includes the ID of the OBE, the link, and the average speed of the link traversal. The interval of data provision and update cycle is 5 minutes.

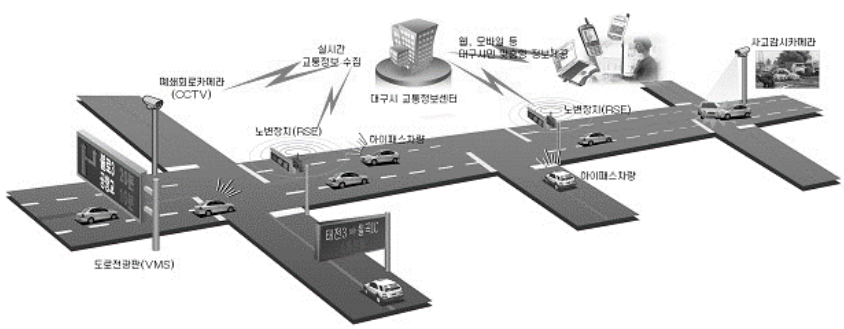


Figure 3. 2 Concept of the DSRC System

3.3. Data Analysis

3.3.1. Characteristics of Collected Data

3.3.1.1. Data Missing Characteristic

Data collected from the DSRC system causes various forms of data missing. This missing can be due to a variety of problems, from instrument problems to probe vehicle data (PVD) itself.

	110	112
2018-03-15 15:55	21.61	26.41
2018-03-15 16:00	27.88	31.22
2018-03-15 16:05	23.14	34.93
2018-03-15 16:10	22.69	37.79
2018-03-15 16:15	21.53	32.05
2018-03-15 16:20	25.47	39.79
2018-03-15 16:25	27.71	25.61
2018-03-15 16:30	29.30	28.86
2018-03-15 16:35	22.13	34.83
2018-03-15 17:15		
2018-03-15 17:20		
2018-03-15 17:25	28.50	
2018-03-15 17:30	25.44	29.65
2018-03-15 17:35	20.39	26.95
2018-03-15 17:40	23.73	31.94
2018-03-15 17:45	21.77	23.00
2018-03-15 17:50	23.26	30.84
2018-03-15 17:55	17.77	23.96
2018-03-15 18:00	26.64	29.59
2018-03-15 18:05	23.20	22.78

	392	422	442
2018-03-01 8:05	39.87	50.54	44.50
2018-03-01 8:10	44.44	62.19	65.58
2018-03-01 8:15	40.86	34.83	44.92
2018-03-01 8:20	44.96	54.26	53.96
2018-03-01 8:25	37.30	48.86	39.90
2018-03-01 8:30	41.93	39.66	59.03
2018-03-01 8:35	33.57	45.31	46.95
2018-03-01 8:40	40.24	51.92	64.56
2018-03-01 8:45	41.06		43.44
2018-03-01 8:50	36.43		48.96
2018-03-01 8:55	39.62		40.33
2018-03-01 9:00	45.09		61.76
2018-03-01 9:05	35.96		41.98
2018-03-01 9:10	40.17		52.09
2018-03-01 9:15	38.19		43.38
2018-03-01 9:20	42.62		53.91
2018-03-01 9:25	37.89		46.88
2018-03-01 9:30	39.31		59.16
2018-03-01 9:35	35.50		39.04
2018-03-01 9:40	39.35		45.96
2018-03-01 9:45	39.78		40.52
2018-03-01 9:50	37.78		48.22
2018-03-01 9:55			43.43
2018-03-01 10:00	40.26		46.42
2018-03-01 10:05	39.29		42.29
2018-03-01 10:10	38.46		52.26

Figure 3. 3 Cases of the Data Missing (Left: Short term, Right: Long Term)

Short-term data missing means no data is collected within some aggregation times. The most common cause is when an OBE-equipped vehicle does not cross the section. It also includes short-term device failures and short-term software failures. In the case of short-term data missing, since the temporal correlation in a link is high, imputation can be easily performed based on the before and after data.

However, in the case of long-term data missing, there are many differences. Long-term data missing means no data is collected for more than one hour. There are also some cases where some months and years of data are not collected. Common causes of the problem are long-term problems with the device. Other causes include system checks and errors in essential parts of the software. In the case of long-term loss, data is absent for a very long time, so the data of the upstream and downstream links are averaged and provided. However, these data have very low correlations, which causes problems in terms of accuracy.

In general, when such data missing occurs, a simple averaging method or a simple moving average method is used. For the simple temporal moving average method, MAE 4.89 is adequate for short term data missing. However, in the case of long-term data missing, the temporal average cannot be used because there is no close temporal data. Therefore, the average is calculated using spatially close data. In this case, the correlation between the data is not stable, which significantly reduces accuracy.

Table 3. 2 Performances of Simple Average Methods

Missing Term	Temporal Moving Average	Spatial Average
Long Term	8.08 km/h	11.71 km/h
Short Term	4.89 km/h	8.60 km/h

Table 3.2 shows the accuracy of the commonly used Simple Moving Average (SMA) method and the spatial average method. In the short term, the temporal SMA method shows an appropriate degree of performance, but in other cases, it shows a substantial error. The simple average of the surrounding data also shows a significant error, making it difficult to use.

If there is missing data in the input in machine learning, proceed with learning by performing data imputation. Imputation refers to a method in which other data is replaced so that learning can be performed when there is a blank or non-available data in the “input data.”

However, when the general imputation (linear combination imputation with nearby link material) is performed, the action of the signal is reversed to act as a case that disturbs learning. Therefore, this study explored another method.

Imputation for input data missing is not a process of finding an "estimated value," but rather a "value that fit learning." As a technical application method, the existing study finds a method that facilitates learning through the convergence of the global mean, learns the speed of convergence, obtains an imputation value, uses missing data, and improves accuracy. (Che et al. 2018) The method called ‘Decayed Method,’ to insert the decayed term at the transmission of the hidden term.

However, there are problems in applying the decayed term directly to the speed data. First, it is impossible to apply periodicity because it is a convergence value for the global mean. The speed of travel changes over time and does not converge to the global mean, as shown in Figure 3.4, which is not suitable for estimating the speed of travel. The second is that it cannot reflect the influence of the back and forth links. Therefore, in this study, we used the values derived from the random forest, which can be applied simply through the pre-training without a monotonic problem. In conclusion, this study developed the 'Active Imputation Method,' which learns the ratio between the value obtained with pre-trained random forest and the last observation. The method is a technical application to improve accuracy while dealing with Missing Data.

Horizontal: Time of days (by hour) Vertical: Mean speed of each hour(km/h)

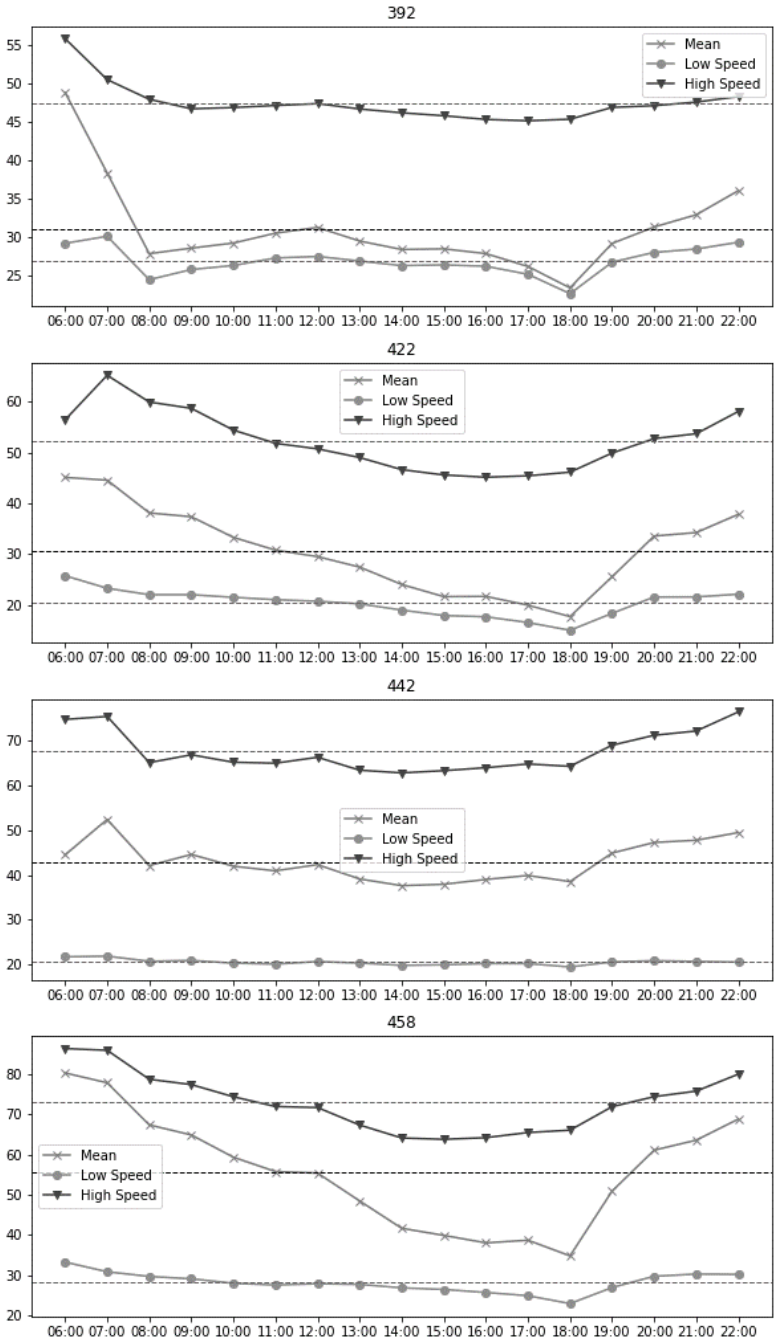


Figure 3. 4 Mean Speed of Each Time

3.3.1.2. Correlation Analysis

Table 3.3 shows the Spearman correlation coefficient between the mean speed value of the 13 links and the mean speed of the upstream and downstream links. Overall, it shows a positive correlation, but it shows a low correlation on many links.

Table 3.3 Correlation with Nearby Links

Link	Data Correlation	
	Upstream	Downstream
112	0.491	0.397
114	0.416	0.126
116	0.140	0.456
392	0.449	0.383
123	0.401	0.274
125	0.275	0.470
442	0.459	0.583
546	0.583	0.709
458	0.713	0.817
422	0.813	0.716
540	0.721	0.702
130	0.697	0.206
350	0.228	0.120

Table 3.4 analyzes the directional consistency and correlation of the speed data changes between upstream, downstream, and target links. Regardless of the upstream and downstream, the correlation of change amount was less than 0.5. In the case of change direction consistency, a random value (50%) was obtained in many links. This means that there is a low correlation between the speed value change of the nearby link data and the speed value change of the target link data.

Table 3. 4 Speed Change Analysis

Link	ΔSpeed Direction Coincidence Probability(%)		ΔSpeed Correlation	
	Upstream	Downstream	Upstream	Downstream
112	56.3	59.6	0.111	0.242
114	58.5	50.2	0.233	0.018
116	49.7	50.4	0.015	0.063
392	50.3	52.9	0.059	0.097
123	52.6	54.6	0.079	0.100
125	54.6	68.1	0.100	0.323
442	68.0	73.2	0.319	0.475
546	73.2	66.8	0.475	0.376
458	66.3	62.7	0.366	0.276
422	62.7	64.5	0.271	0.402
540	64.4	67.2	0.402	0.362
130	66.7	65.0	0.357	0.170
350	66.1	48.9	0.184	-0.016

3.3.1.3. Periodicity Analysis

Due to the difference between the signal cycle and the aggregation time unit, a periodicity occurs in the average speed data. As a result of time-lag autocorrelation analysis, which is generally used for periodicity analysis, periodicity was observed in speed data on 11 of 13 links. Figure 3.5 shows the autocorrelation analysis graphs for periodicity analysis. (figures for all links are included in the appendix) The exceptions are as follows: Weak periodicity was observed on one link (Link No. 114), and the start node is the highway entry ramp. Periodicity was not observed in one link (Link No. 116), and the start node is a highway exit ramp.

If periodicity appears in the data, structural problems arise in the learning of the LSTM. An LSTM learns the autocorrelation of data itself as the structure. As a result, a problem arises in that data having a higher autocorrelation period is more important than the latest data.

Horizontal: Time Lagging(Min.), Vertical: Pearson Correlation Coefficient.

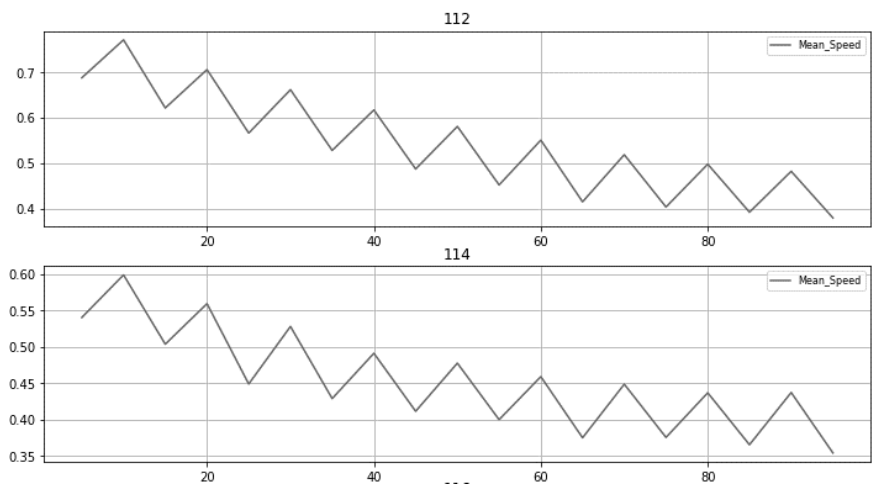


Figure 3. 5 Periodicity of Mean Speed Value

3.3.1.4. State Transition Analysis

In this study, the congestion state was designated as the case where the average speed was lower than the threshold obtained, according to Jenks Natural Break. In the case of interrupted flow, the state transition probability is higher than that of uninterrupted flow. In particular, the probability of transition from a non-congested state to a congested state is relatively high. The ratio of transition state samples is also very high, at 23.2%.

Table 3. 5 State Transition Probability Between Traffic State

Transition	Interrupted	Uninterrupted
Non-congested→Non-congested	63.16%	92.45%
Non-congested→Congested	36.84%	7.55%
Congested→Non-congested	20.50%	13.45%
Congested→Congested	79.50%	86.56%

In the case of uninterrupted flow, there is a high probability of maintaining the previous state. Thus there is little expectation of accuracy reduction in case of time lagging in estimation without catching the transition. However, in the case of interrupted flow, the continuous transition can occur when time lagging occurs. Therefore, interrupted flow needs correction. In addition, the transition probabilities between states are also different, so a way to compensate for this is required.

3.3.2. Results of Existing Models

3.3.2.1. Accuracy of Existing Models

The results of estimating the average speed using the existing estimation models are shown in Table 3.6. Simple Average is the most commonly used method and averaged the current average speed of the upstream and downstream links. It can be seen that due to the low correlation described above, an appropriate value cannot be derived. In the case of linear regression analysis using the least square method, it was higher than the simple mean but did not yield high accuracy. Other likelihood-based regression methods (Ridge, Lasso, etc.) did not yield different results than OLS. In particular, in the case of the linear combination model, the accuracy used is low even though the values used for fitting the estimated model are the same as those used for the estimation.

In order to solve the problem of low correlation, which may be the cause of low accuracy of the linear combination models, the LSTM model was used to estimate relatively higher accuracy than the existing model. In the LSTM model, train data and test data were separated.

Table 3. 6 Speed Estimation Accuracy of Existing Models

Model	MAE (km/h)	RMSE (km/h)
Simple Average	12.879	15.102
OLS Regression (Single Time)	5.987	8.163
OLS Regression (Time Series)	5.743	7.855
Naive LSTM (Random Forest Imputed)	4.307	5.852

3.3.2.2. Correlation Analysis

Figure 3.6 compares the correlation coefficient between the mean speed values estimated by the model using single-time data and the mean speed values of up / downstream. As can be seen from the graph, the single-time model was unable to estimate the correlation with the target data higher than the correlation of the input data. This is true even when the correlation between the input data and the target data is high. (Graphs for all links are listed in the appendix) In detail, the linear combination model using single-time data cannot increase the correlation between the target data and the result more than the correlation with the input data and the result.

Vertical: Spearman Corr. Coefficient(0~1.0)

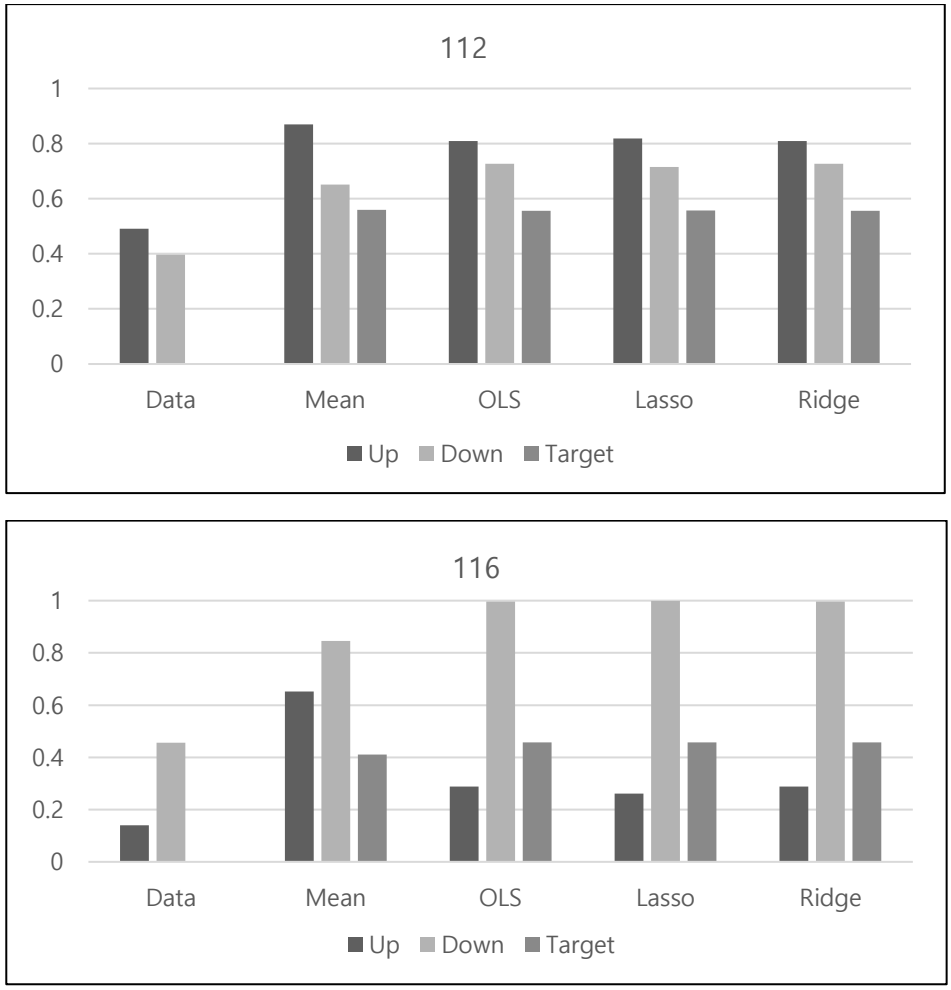


Figure 3. 6 Correlation between Up / Downstream Link Data and Single-Time Model Results

Even in the case of a model using time series data (30 minutes), the occurrence of correlation problems is not very different. For models using time series data, model fit/accuracy is slightly higher than for single-time models. (5.98 → 5.74) However, the correlation between the result and the target data is the same as using

single-time data. This phenomenon occurs in the case of close to monotone due to a high correlation between up / downstream and target data.

As a solution to this problem, we can suggest multi-layered models. In the case of multi-layered models, Random Forest and LSTM, the correlation of target data can be increased beyond the limit of surrounding data.

Vertical: Spearman Corr. Coefficient(0~1.0)

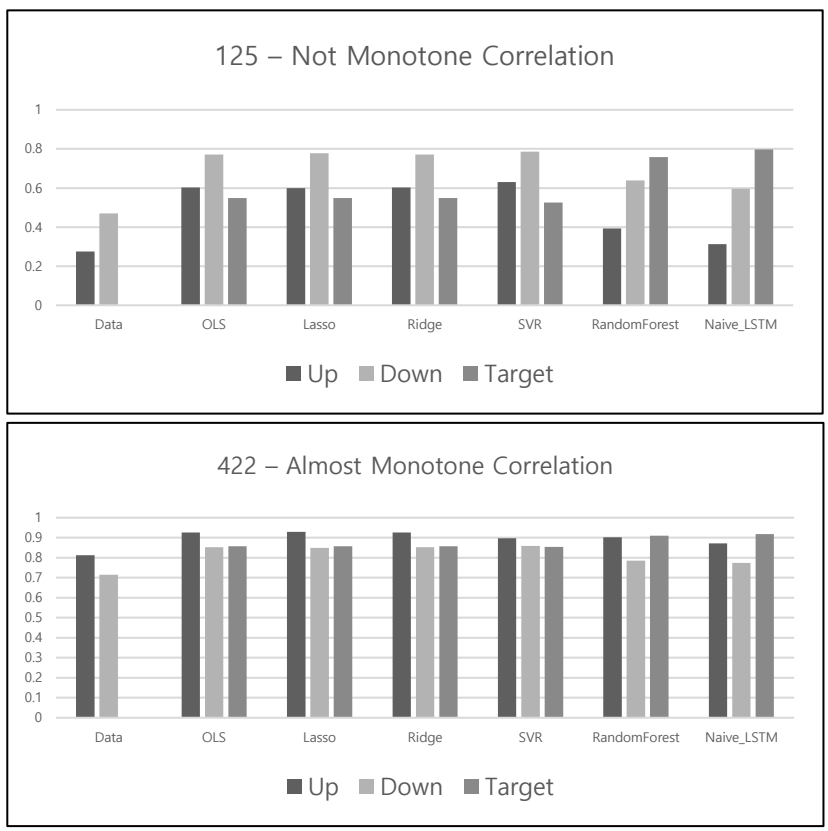


Figure 3. 7 Correlation between Up / Downstream Link Data and Time-series Model Results

Table 3. 7 ⊿Speed Correlation Analysis

Link	ΔSpeed Sign Coincidence Probability(%)			ΔSpeed Correlation		
	Upstream	Downstream	OLS	Upstream	Downstream	OLS
112	56.3	59.6	61.5	0.111	0.242	0.248
114	58.5	50.2	58.9	0.233	0.018	0.241
116	49.7	50.4	51.2	0.015	0.063	0.067
392	50.3	52.9	53.9	0.059	0.097	0.126
123	52.6	54.6	54.3	0.079	0.100	0.105
125	54.6	68.1	69.7	0.100	0.323	0.361
442	68.0	73.2	80.9	0.319	0.475	0.613
546	73.2	66.8	77.3	0.475	0.376	0.582
458	66.3	62.7	77.3	0.366	0.276	0.425
422	62.7	64.5	69.1	0.271	0.402	0.510
540	64.4	67.2	70.2	0.402	0.362	0.518
130	66.7	65.0	72.5	0.357	0.170	0.454
350	66.1	48.9	68.4	0.184	-0.016	0.199

Table 3.7 shows the direction correspondence between the speed data of the target link and the upstream, downstream, and OLS results. It shows the correlation between the proportions of the signs and the amount of change. Regardless of the upstream and downstream, the correlation of change amount was less than 0.5 in all cases. In the case of change direction agreement, a random value (50%) was derived from many links. This feature again proves that there is a low correlation between the change in the speed of peripheral link data and the difference in the speed of target link data. Also, linear regression, calculated by Ordinary Least Square (OLS), did not significantly improve the correlation problem.

3.3.2.3. Adequacy of Periodicity Estimation

Figure 3.8 shows the relative value of importance over time-analyzed by a local interpretable model-agnostic explanations (LIME) algorithm that evaluates the importance of input value. (Ribeiro et al. 2016) This algorithm can be used to confirm that the dependence is reversed. If the link traffic speed is static, this is not the problem. However, link traffic speed is dynamic due to its complexity. Therefore, if a situation that causes a sudden change in speed occurs, the LSTM will make an incorrect estimation.

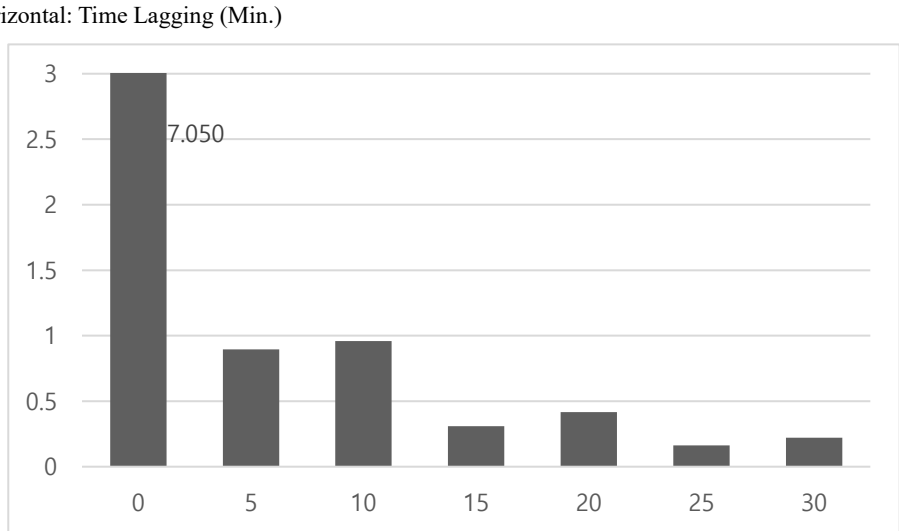


Figure 3. 8 Result of LIME Analysis of LSTM by Time

Besides, this "source" learning method sometimes causes the LSTM not to learn the actual periodicity present. Indeed, the LSTM on two links failed to learn periodicity. Figure 3.9 shows an example of periodic learning failures (Link 123).

Horizontal: Time Lagging(Min.), Vertical: Pearson Correlation Coefficient,
Top: Autocorrelation of Speed Data, Bottom: Autocorrelation of LSTM Result

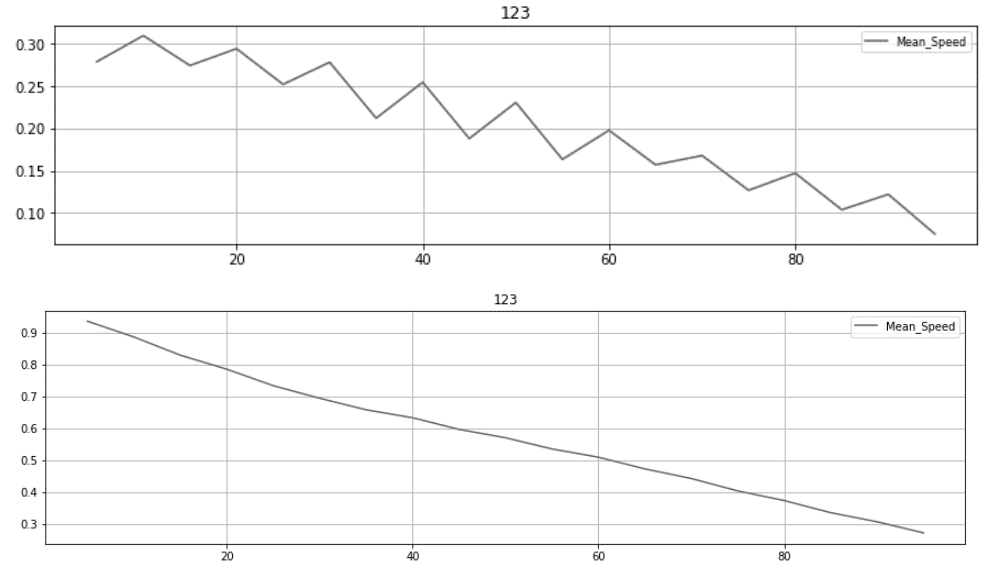


Figure 3. 9 Case of Periodicity Leaning Failure

MAE, Unit: km/h

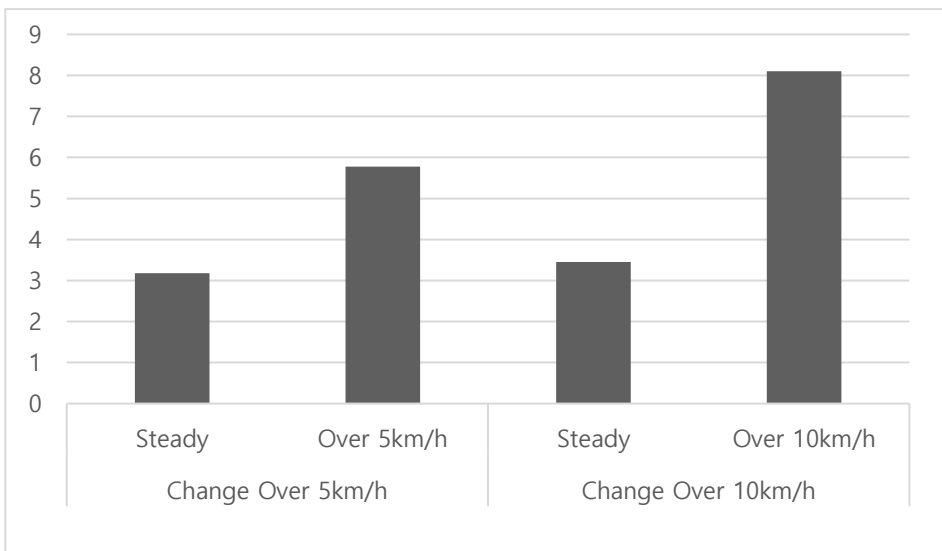


Figure 3. 10 MAE of LSTM for Sudden Change in Speed

Due to the correlation inversion problem of periodic learning, LSTM is vulnerable to state that sudden change in speed occurs. Figure 3.10 shows that the LSTM is vulnerable to drastic changes of more than 5 km/h and more than 10 km/h, respectively.

3.3.2.4. State Transition Analysis

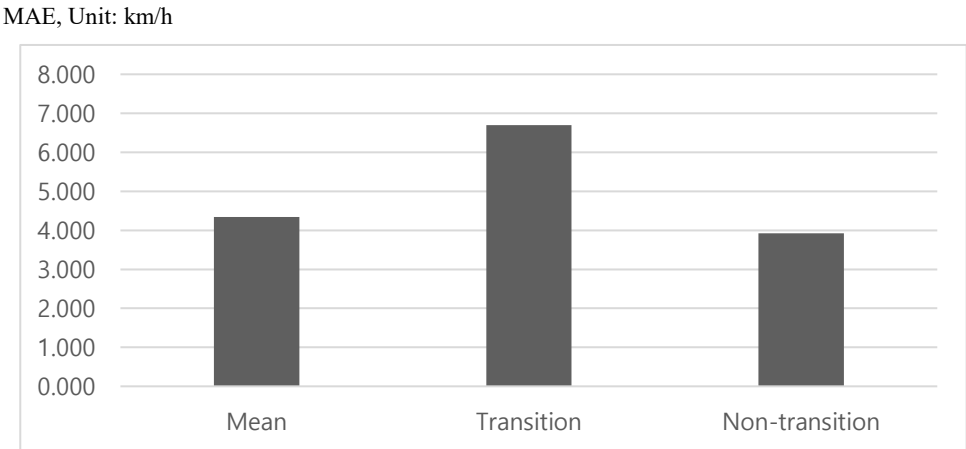


Figure 3. 11 MAE of LSTM for Transition State

Since the LSTM model performs hidden term transmission without adjusting the transition state, it is difficult to learn the transition state that changes in dynamics properly. Also, there is a problem of learning the congested state and non-congested state, showing different dynamics with the same structure. This is associated with periodic learning, leading to low accuracy due to sudden change. In this study, we tried to solve this problem by providing adjust, which performs separate learning according to state in Transmission of Hidden Term.

3.3.2.5. Summary of Existing Models Application

The results of applying the existing estimation model to the collected data can be summarized as follows. In the case of estimation using the existing estimation model, the linear combination model yielded very low accuracy and correlation. In order to solve the problem, the estimation was performed by applying the Deep Learning model. In this case, the low accuracy problem could be solved to a certain level. In the case of LSTM, one of the deep learning models, on average, high accuracy and correlation were obtained.

However, in the case of LSTM, the limitations of periodic learning and traffic conditions were found. In the case of the LSTM model, it is vulnerable to sudden changes due to the influence of periodic learning, and it is also vulnerable to the transition state.

Based on previous research, cell-transmission based RNN models such as LSTM and GRU are known to be able to cope with various conditions through tuning the transmission cell(Ravanelli et al. 2017; Wöllmer et al. 2011). Therefore, in this study, we propose to develop a modified RNN model that is properly tuned.

3.3.3. Analysis Using Separated Platoons

3.3.3.1. Separation of Platoons in Interrupted Flow

Basically, interrupted flow is restricted by the behavior of the vehicle due to traffic signals or intersections. Many previous studies have studied that the distribution of speed and travel time of a vehicle is divided into two or more distributions due to the queuing caused by signals, and it is found that it is primarily divided into queued vehicles and unqueued vehicles. (Akcelik 1996; Gartner et al. 1992; Yang et al. 2014)

Due to the separation of platoons (Gartner et al., 1992), the waiting vehicles at the previous intersection enter the next intersection with the formed platoon. Due to the offset operation between intersections, in the free-flow state with low traffic, platooned vehicles at the previous intersection and vehicles arriving immediately after the green signal can pass through the progression with high probability without queueing at the next intersection. In the car-following state, where the traffic volume is higher than the free-flow state, the proportion of vehicles that are not platooned increases, or the proportion of vehicles that have undergone queueing at the target segment increases because queueing is not resolved at the target segment. When congestion or traffic jams occur, most cars experience queueing due to the high volume of traffic. As such, the separation of the queued platoon and unqueued platoon occurs, and this phenomenon is called “the platoon separation” in this study.

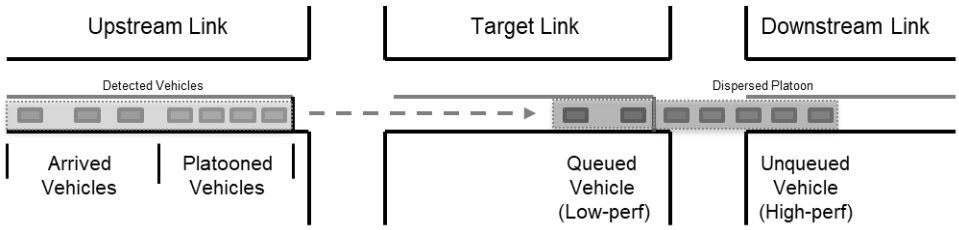


Figure 3. 12 Concept of the Platoon Separation

In this study, it was confirmed that the platoon separation occurs at each time through the speed data of each vehicle collected by DSRC, and it is divided into two types of traffic flows for the entire link. In general, platoon formation of flow occurs due to simultaneous start after queueing. (Akcelik 1996; Gartner et al. 1992)

The link will vary the average distribution of each platoon class, which is related to the link's performance. Figure 3.13 is a histogram showing the platoon separation at a specific time. Extending this to the entire data of one link also indicates the platoon separation (Figure 3.14).

Horizontal: Speed of a Vehicle(km/h), Vertical: Counts of Vehicles

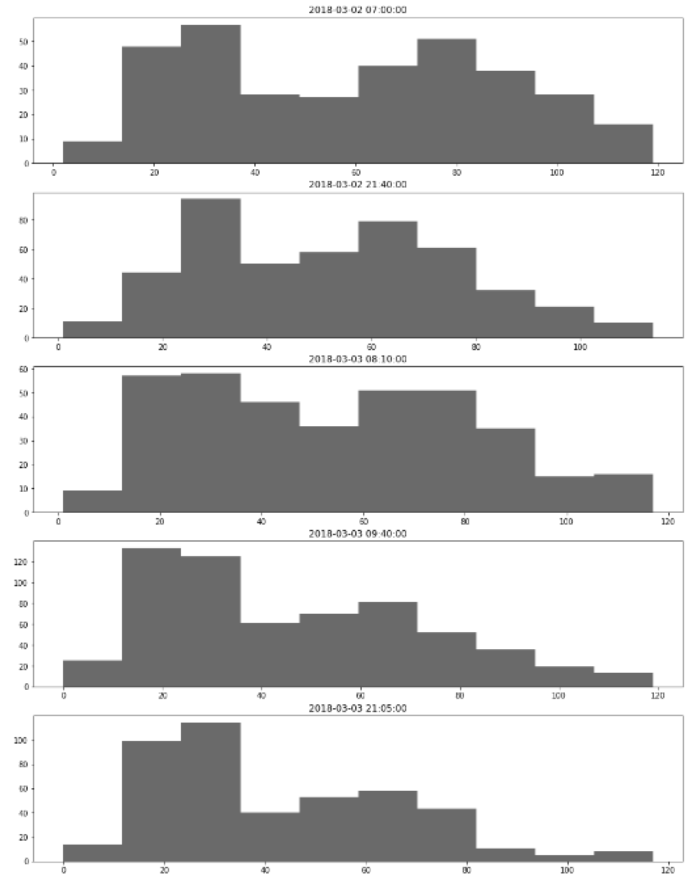


Figure 3. 13 Platoon Separation for Aggregation Time

Horizontal: Speed of a Vehicle(km/h), Vertical: Counts of Vehicles

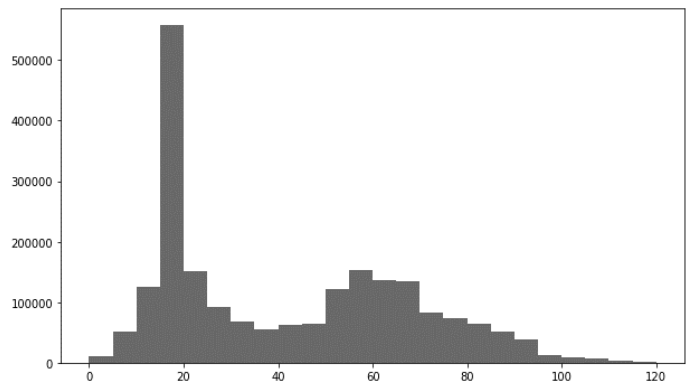


Figure 3. 14 Platoon Separation for a Link (Link 442)

3.3.3.2. Analysis for Separated Platoons

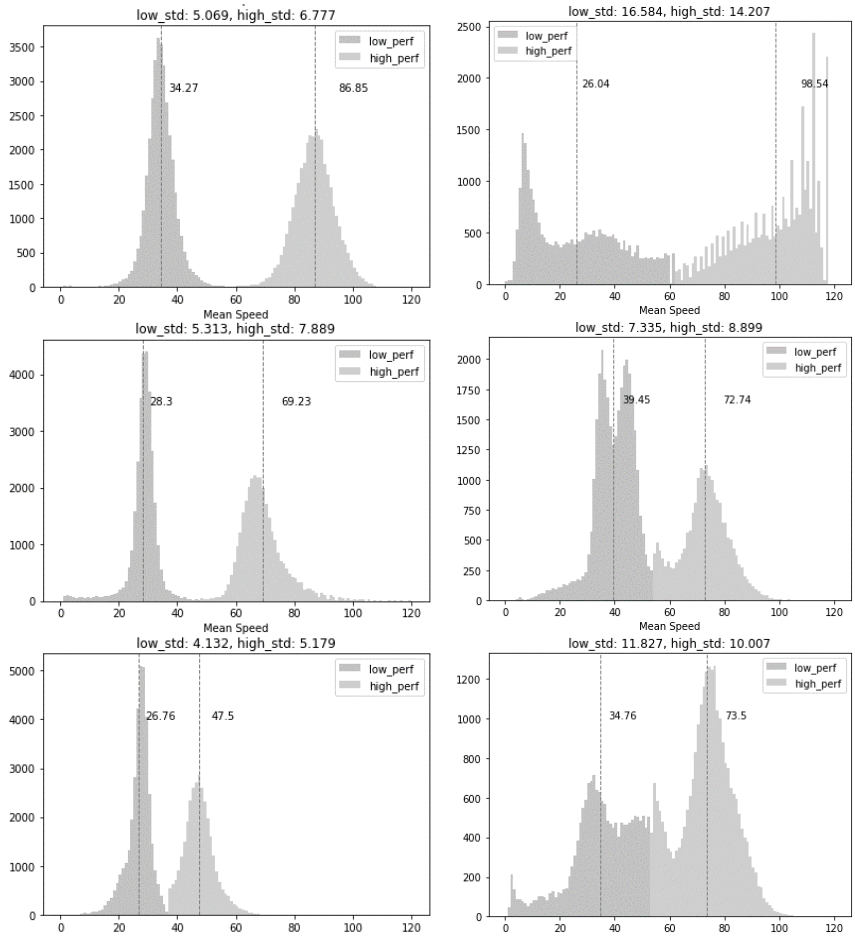
For the application of the platoon separation, this study classifies platoons into two types, the high-speed platoon, and the low-speed platoon. In order to reflect the performance of the link, each link was classified to obtain a threshold. In this study, Jenks Natural Break was used as a technique for the clustering of traffic features and searching for state identifying thresholds for a single property of urban traffic flows. (Wen et al. 2017; Wu and Hung 2010)

The Jenks Natural Break method searches for thresholds that minimize the variance in the classification group and increase the variance among the classification groups. The technique is appropriate when the classification of each platoon is clear. In the case of intermittent flow, if the length of the link is short enough, the separation of platoons occurs clearly, which is a suitable method.

Table 3. 8 Result of the Jenks Natural Break by Link

Link No.	Category	Threshold (km/h)
112	Interrupted	39.07
114	Interrupted	40.90
116	Interrupted	51.17
392	Interrupted	36.74
123	Interrupted	40.01
125	Interrupted	38.00
442	Interrupted	43.00
546	Interrupted	39.00
458	Interrupted	50.12
422	Interrupted	36.05
540	Interrupted	42.08
130	Interrupted	59.95
350	Interrupted	47.82

Figure 3. 15 Result of Platoon Classification



Horizontal: Mean speed at a Time Range(5min), Vertical: Counts
Left: Interrupted Flow, Right: Uninterrupted Flow

The classification result of platoons into high-performance/low-performance platoons is shown in Figure 3.15 (data for all links are included in the appendix). In the case of interrupted flows, the distinction between high-speed and low-speed vehicles was obvious. This separation can be said to be a phenomenon different from that of continuous flow. In the case of uninterrupted flow, the distinction was not clear because, in the case of continuous flow, the platoon effect did not occur, and

the speed of the entire traffic flow was similar.

In the case of interrupted flows, two Platoons are often distinguished within the same aggregation unit, because unqueued vehicles and queued vehicles are mixed in one aggregation unit.

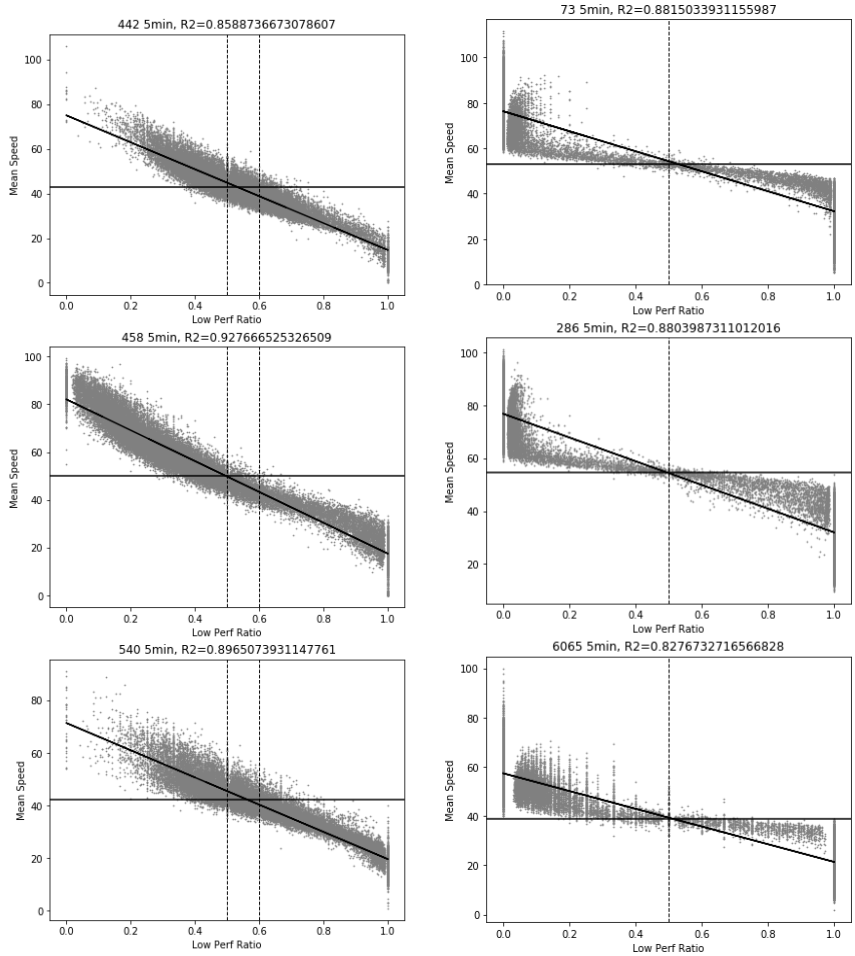
The statistical characteristics of the platoon speed distribution are apparent. LPP Speed shows low skewness and high kurtosis on average.

HPP Speed has a relatively low kurtosis compared to LPP Speed. In all cases, the D'Agostino-Pearson normality test was rejected. Table 3.9 shows the statistical characteristics of the platoon distribution.

Table 3. 9 Statistical Characteristics of the Platoon Speed Distribution

Location		Unit (min)	Low-performance Platoon		High-performance Platoon	
			Skewness	Kurtosis	Skewness	Kurtosis
112	Dalgubeol	5	-0.45	1.83	0.05	0.76
114	Dalgubeol	5	-0.53	1.31	0.49	2.92
116	Dalgubeol	5	-0.62	0.05	0.25	-0.56
392	Dalgubeol	5	-1.27	3.24	0.62	1.17
123	Dalgubeol	5	-0.63	2.08	0.95	1.85
125	Dalgubeol	5	-0.24	3.75	1.36	3.03
442	Dalgubeol	5	-1.13	8.34	0.17	0.50
546	Dalgubeol	5	-0.31	4.71	0.47	0.36
458	Dalgubeol	5	-0.54	2.07	0.02	-0.32
422	Dalgubeol	5	-0.50	2.15	0.54	0.15
540	Dalgubeol	5	-1.03	1.16	0.63	0.38
130	Dalgubeol	5	-0.34	4.14	-0.03	0.41
350	Dalgubeol	5	-2.01	8.23	1.28	3.45

Figure 3. 16 Relationship between LPR and Mean Speed

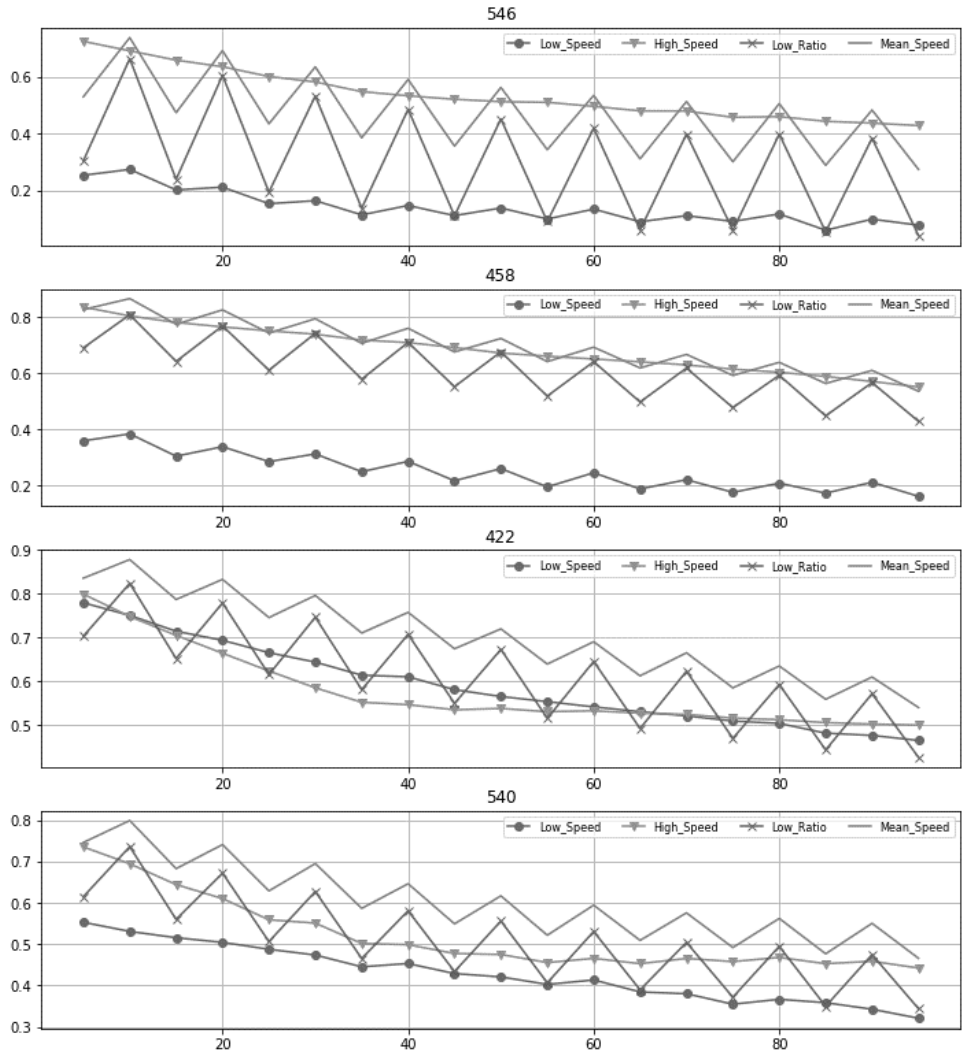


Horizontal: Low Performance Platoon Ratio at a Time Range(5min)(0~1.0)
Vertical: Mean Speed at a Time Range(5min)

As the average speed of platoons shows a high kurtosis distribution, a constant linear relationship is observed between the average speed of the LPR and the link. In the case of continuous flow, a shallow linear relationship is observed, but it is not well-fitted due to the lack of simultaneous samples. Figure 3.16 shows the relationship between LPR and average speed for interrupted and uninterrupted flows. As described above, in the interrupted flow, a clear linear relationship can be

observed. This linear relationship can be a powerful hint for estimating average speed. In this study, a periodical analysis was performed on LPP Speed, HPP Speed, LPR, and Mean Speed to confirm whether LPR is strongly involved in the periodicity of average speed.

Figure 3. 17 Periodical Analysis for Separated Platoons



The result of periodic analysis through time-lagged autocorrelation for each feature of separated platoons is shown in Figure 3.17. The analysis showed a periodicity of 10 minutes for LPR and average speed except for one link. One exception link is the presence of highway ramps in the ramp, which is not strongly affected by the periodicity of the signal. HPP Speed showed no periodicity on 11 of 13 links. LPP Speed showed weak periodicity on six links and strong periodicity on one link. In particular, LPR shows strong periodicity in all cases where the characteristic of interrupted flow is strong, and periodicity is observed. This phenomenon indicates that the most critical feature in generating periodicity of average speed is the periodicity of LPR for each signal cycle.

As described above, the occurrence of periodicity causes a correlation reverse problem in learning the multi-layer model RNN (such as GRU, LSTM). In order to solve this problem, this study developed a method of separating and learning data that causes periodicity by using the platoon separation phenomenon. In more detail, three data that can be estimated through the platoon separation phenomenon were preliminarily estimated, and then the average speed was estimated. Through this method, it is possible to reduce the dependence on the last input data and to apply the learning that has the dependence that decreases with time by separating the data with periodicity. In terms of technique, more accurate training can be done by limiting the train of the model.

3.3.4. Summary

Table 3.10 summarizes the problems raised by data analysis and the suggested methodologies in this study.

Table 3. 10 Summary of Data Analysis

Data Characteristics	Limitations of Existing Models and Methodologies	Application methodology
Low Correlation of Data	Low Accuracy of Estimation Models	Applying of RNN Models
Problems with Periodic Generation of Data	Correlation Application Problem According to Structural Learning of Periodicity, Challenging to Apply Sudden Change in Speed	Development and Application of Two-step Model Using Plating Effect
Higher transition probability compared to continuous flow	Estimation cannot follow changes when an event occurs, Low Accuracy for Transition	Selected Dropout Development and Application
Frequently Occurred Missing Data	Data Imputation Needed	Active Imputation Development and Application

Chapter 4. Model Development

4.1. Basic Concept of the Model

In this chapter, a data estimation model was developed based on the characteristics of data collection and interrupted flow analyzed in the previous chapter. This chapter describes the conditions of the development model, the development of a recurrent neural network suitable for the data, and the development of modules to apply transition probabilities.

4.1.1. Conditions of the Development

This development model estimates the average speed of traffic flow, which is representative information about link traffic. The average speed estimate aims to provide robust and compensating data for the short-term and long-term data missing of the probe vehicle system provided for interrupted flow.

In order to estimate the average traffic flow speed using the platoon separation which is the characteristic of interrupted flow, this development model predicts the LPP speed, HPP speed, LPR, and finally uses the average speed of whole traffic flow. Estimate

Since it is difficult to accurately estimate intermediate outputs (speed of LPP, speed of HPP, LPR) with low temporal correlation with other variables, we use a deep learning model with high accuracy. In addition, this development model aims to apply the difference in the transition probabilities of traffic states inherently in the model.

4.1.2. Structure of the Model

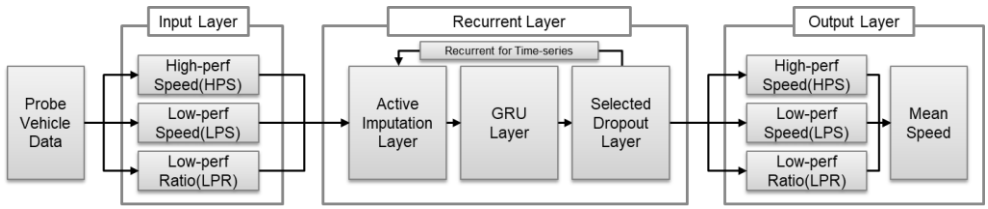


Figure 4. 1 Structure of the Developed Model

The model presented in this model can be divided into three parts: input layer, recurrent layer, and output layer. The role of the input layer is to divide the collected probe vehicle data into three data based on separated platoons analysis. The recurrent layer estimates three data through the developed modified recurrent neural network. This layer is the most important layer to compensate for missing data and to learn and estimate periodicity and state transitions. The output layer has the role of calculating the mean speed through the estimated three data.

The recurrent layer consists of three layers internally. The first layer is an active imputation layer, which produces a dataset that can learn by imputing missing data through a combination of random forest and a neural network. Secondly, the GRU layer learns and estimates through the GRU cell based on the input value. The last layer, the selected dropout layer, performs separate learning and estimation according to the traffic state. The above three machine learning layers are combined to fit the appropriate model.

4.1.3. Formations of Data

The final output data of this development model is the estimated average speed of the target time of Target Link. The situation assumed in this study is that there are various types of missing data in the input data, and the target time of the target link is Missing. Primary output data are LPP Mean Speed and HPP Mean Speed of Target Time of Target Link. The final output data is calculated using the primary output data. However, as the label data for the learning process, the final output data is also used to facilitate the learning. Figure 4.2 shows the formation of I/O data.

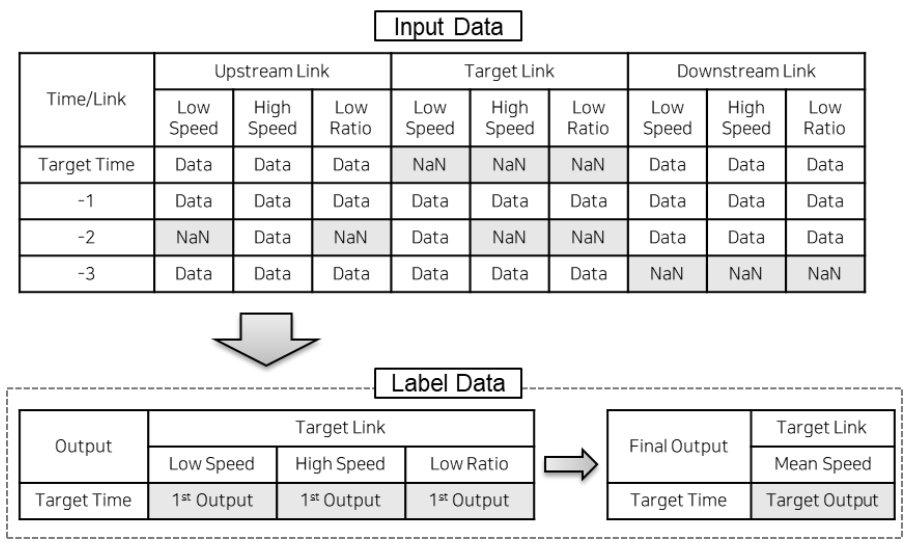


Figure 4. 2 I / O data type

4.2. Model Development

4.2.1. Notations

The symbols of the mathematical formulas used in this chapter are as follows. The model developed in this study is defined for time-series data denoted as $\mathrm{X} = (x_1, x_2, \dots, x_T)^{\mathrm{T}} \in \mathbb{R}^{T \times D}$ with time length T with D variables.

d: variable number

t: time number

x_t^d: The observed value of the t-th time of the d-th variable

x_t: vector of observations at the t-th time

r_t: Reset Gate value of t-th time

z_t: Update Gate value of t-th time

h_t: Hidden Term at t-th time

$\hat{x}_t^d$: Adjusted Input value of t-th time of d-th variable

$x_{t'}^d$: Last observation at time t with $t > t'$

$\tilde{x}^d$: average value of the d-th variable

I_t^d: Imputation target value of t-th time of d-th variable

W, U, b: Parameters of each linear combination

4.2.2. Gated Recurrent Unit(GRU)

The basic model of the recurrent cell used in this study is GRU. GRU was developed through the research of Cho. The GRU is expressed as formulas below (Cho et al. 2014).

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \cdots\cdots \text{Equation 4. 1}$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \cdots\cdots \text{Equation 4. 2}$$

$$\tilde{h}_t = tanh(W x_t + U(r_t \odot h_{t-1}) + b) \cdots\cdots \text{Equation 4. 3}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \cdots\cdots \text{Equation 4. 4}$$

4.2.3. Concept of GRU-D

In this study, we used the idea of GRU-D, a GRU studied by Che et al. GRU-D can handle missing data internally by adding a decayed term in GRU. The formula of GRU-D is as follows. (Che et al. 2018)

$$\gamma_t = \exp(-max(0, W_\gamma \delta_t + b_\gamma)) \cdots\cdots \text{Equation 4. 5}$$

$$\hat{x}_t^d = m_t^d x_t^d + (1 - m_t^d)(\gamma_{x_t}^d x_{t'}^d + (1 - \gamma_{x_t}^d)\tilde{x}^d) \cdots\cdots \text{Equation 4. 6}$$

$$\hat{\mathbf{h}}_{t-1} = \gamma_{h_t} \odot \mathbf{h}_{t-1} \cdots\cdots \text{Equation 4. 7}$$

$$r_t = \sigma(W_r \hat{x}_t + U_r \hat{h}_{t-1} + V_r m_t + b_r) \cdots\cdots \text{Equation 4. 8}$$

$$z_t = \sigma(W_z \hat{x}_t + U_z \hat{h}_{t-1} + V_z m_t + b_z) \cdots\cdots \text{Equation 4. 9}$$

$$\tilde{h}_t = \tanh(W \hat{x}_t + U(r_t \odot \hat{h}_{t-1}) + V m_t + b) \cdots\cdots \text{Equation 4. 10}$$

$$h_t = (1 - z_t) \odot \hat{h}_{t-1} + z_t \odot \tilde{h}_t \cdots\cdots \text{Equation 4. 11}$$

Where Mask m_t^d and delta time vector δ_t^d are defined as vectors in the following equation.

$$m_t^d = \begin{cases} 1, & \text{if } x_t^d \text{ is observed} \\ 0, & \text{otherwise} \end{cases} \quad \cdots\cdots \text{Equation 4. 12}$$

$$\delta_t^d = \begin{cases} s_t - s_{t-1} + \delta_{t-1}^d, & t > 1, m_{t-1}^d = 0 \\ s_t - s_{t-1}, & t > 1, m_{t-1}^d = 1 \\ 0, & t = 1 \end{cases} \quad \cdots\cdots \text{Equation 4. 13}$$

The input data is shown in the Figure 4.3 below.(Che et al. 2018)

$\boldsymbol{X}$: Input time series (2 variables); $\boldsymbol{M}$: Masking for $\boldsymbol{X}$;
$\boldsymbol{s}$: Timestamps for $\boldsymbol{X}$; $\boldsymbol{\Delta}$: Time interval for $\boldsymbol{X}$.

$$\boldsymbol{X} = \begin{bmatrix} 47 & 49 & NA & 40 & NA & 43 & 55 \\ NA & 15 & 14 & NA & NA & NA & 15 \end{bmatrix} \quad \boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{s} = [\,0 \quad 0.1 \quad 0.6 \quad 1.6 \quad 2.2 \quad 2.5 \quad 3.1] \quad \boldsymbol{\Delta} = \begin{bmatrix} 0.0 & 0.1 & 0.5 & 1.5 & 0.6 & 0.9 & 0.6 \\ 0.0 & 0.1 & 0.5 & 1.0 & 1.6 & 1.9 & 2.5 \end{bmatrix}$$

Figure 4. 3 An example of measurement vectors x_t, time stamps s_t, masking m_t, and time interval δ_t

4.2.4. Development of Modified GRU

4.2.4.1. Concept of the Model

In this study, we devised a method to apply the correlation characteristics of traffic speed data to GRU-D. There are four methods applied in this study, which are as follows. Each method can be said to be the most passive to the most active in order. Also, each case adopts a new approach in addition to the previous one.

1. In GRU-D, the target value of the trained decaying was defined as the

global mean value of the observed value. On the other hand, this study used the estimated imputation value obtained through the relational model with other links.

2. In the case of GRU-D, the linear transform matrix of the input term is limited to be diagonal. However, in this study, the limitation is re-examined to reflect the association between links.
3. Instead of decaying between the imputing value and the last observed value, we attempted to determine the trained ratio through the sigmoid function.
4. Attempts were made to determine the ratio of the substitute value to the present value, regardless of being missing. In this case, if the input value is not actually missing data, the last observed value and the input value are the same, so the trained imputation of the input data is not done internally, but it affects the learning by affecting the gradient of γ_{x_t}.

The best model was found by testing the models using the four methods one by one. The first model (Decayed Imputation Model) uses a method in which GRU-D changes the global mean value only to estimated imputation values. The second model did not further restrict the linear transformation matrix to diagonal matrices. The third model (Adjusted Input Model) replaces the decay function to the sigmoid function as a ratio function. The fourth model (Active Model) learn the proportions of imputation and last observed values regardless of missing. Additionally, the fifth model (Decayed-active model) uses decaying for all cases and does not use the

diagonal constraint for comparison with the above model.

4.2.4.2. Step-by-step Model Definition

The above-described model is expressed by the following equation.

$$\boldsymbol{\gamma_t = exp\{-max(0, W_\gamma \delta_t + b_\gamma)\}} \quad \cdots\cdots \textbf{Equation 4. 14}$$

or

$$\boldsymbol{\gamma_t = Sigmoid\{W_\gamma + b_\gamma\}} \quad \cdots\cdots \textbf{Equation 4. 15}$$

$$\boldsymbol{\hat{x}_t^d = m_t^d x_t^d + (1 - m_t^d)(\gamma_{x_t}^d x_{t'}^d + (1 - \gamma_{x_t}^d) I_t^d)} \quad \cdots\cdots \textbf{Equation 4. 16}$$

or

$$\boldsymbol{\hat{x}_t^d = \gamma_{x_t}^d x_{t'}^d + (1 - \gamma_{x_t}^d) I_t^d} \quad \cdots\cdots \textbf{Equation 4. 17}$$

$$\mathbf{\hat{h}_{t-1} = \gamma_{h_t} \odot h_{t-1}} \quad \cdots\cdots \textbf{Equation 4. 18}$$

$$\boldsymbol{r_t = \sigma(W_r \hat{x}_t + U_r \hat{h}_{t-1} + V_r m_t + b_r)} \quad \cdots\cdots \textbf{Equation 4. 19}$$

$$\boldsymbol{z_t = \sigma(W_z \hat{x}_t + U_z \hat{h}_{t-1} + V_z m_t + b_z)} \quad \cdots\cdots \textbf{Equation 4. 20}$$

$$\boldsymbol{\tilde{h}_t = \tanh(W \hat{x}_t + U(r_t \odot \hat{h}_{t-1}) + V m_t + b)} \quad \cdots\cdots \textbf{Equation 4. 21}$$

$$\boldsymbol{h_t = (1 - z_t) \odot \hat{h}_{t-1} + z_t \odot \tilde{h}_t} \quad \cdots\cdots \textbf{Equation 4. 22}$$

The formula conditions applied to each model are as follows.

5. Decayed Imputation Model
   - Change last observed data to imputation model estimated data in the existing model (GRU-D)
   - Using Equation 4.14 for $\boldsymbol{\gamma_{t_x}}$

- Diagonal constraint of $\boldsymbol{W_{\gamma_x}}$
- Using Equation 4.16 for $\hat{\boldsymbol{x}}_{\boldsymbol{t}}^{\boldsymbol{d}}$

6. Passive Model
   - Change last observed data to imputation model estimated data in the existing model (GRU-D)
   - Using Equation 4.14 for $\boldsymbol{\gamma_{t_x}}$
   - Remove diagonal constraint of $\boldsymbol{W_{\gamma_x}}$
   - Using Equation 4.16 for $\hat{\boldsymbol{x}}_{\boldsymbol{t}}^{\boldsymbol{d}}$
7. Adjusted Input Model
   - Change last observed data to imputation model estimated data in the existing model (GRU-D)
   - Using Equation 4.15 for $\boldsymbol{\gamma_{t_x}}$
   - Remove diagonal constraint of $\boldsymbol{W_{\gamma_x}}$
   - Using Equation 4.16 for $\hat{\boldsymbol{x}}_{\boldsymbol{t}}^{\boldsymbol{d}}$
8. Active Model
   - Change last observed data to imputation model estimated data in the existing model (GRU-D)
   - Using Equation 4.15 for $\boldsymbol{\gamma_{t_x}}$
   - Remove diagonal constraint of $\boldsymbol{W_{\gamma_x}}$
   - Using Equation 4.17 for $\hat{\boldsymbol{x}}_{\boldsymbol{t}}^{\boldsymbol{d}}$
9. Decayed-active Model

- Change last observed data to imputation model estimated data in the existing model (GRU-D)
- Using Equation 4.14 for $\boldsymbol{\gamma_{t_x}}$
- Remove diagonal constraint of $\boldsymbol{W_{\gamma_x}}$
- Using Equation 4.17 for $\hat{\boldsymbol{x}}_{\boldsymbol{t}}^{\boldsymbol{d}}$

This is summarized in the table below.

Table 4. 1 Properties of Each Developing Models

Model Type		Input Adjustment	Diagonal Constraint of Input Linear Transform
Existing	GRU-D	Decayed to Global Mean	W_{γ_x}
Developed	Decayed Model	Decayed to Imputation Value	W_{γ_x}
	Passive Model	Decayed to Imputation Value	None
	Adjusted Input Model	Weighted summation (for missing cases)	None
	Active Model	Weighted summation (for every cases)	None
	Decayed-Active Model	Decayed to Imputation Value (for every cases)	None

The structure of the models is represented as below. However, the mean value and last observed value are actually manipulated inside of the model, unlike those illustrated from the outside.

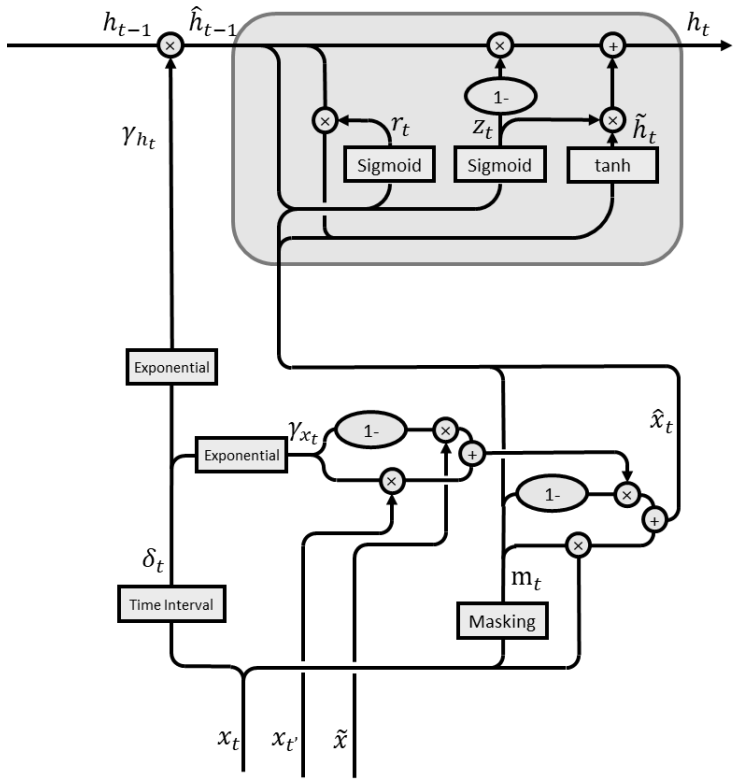


Figure 4. 4 Structure of GRU-D

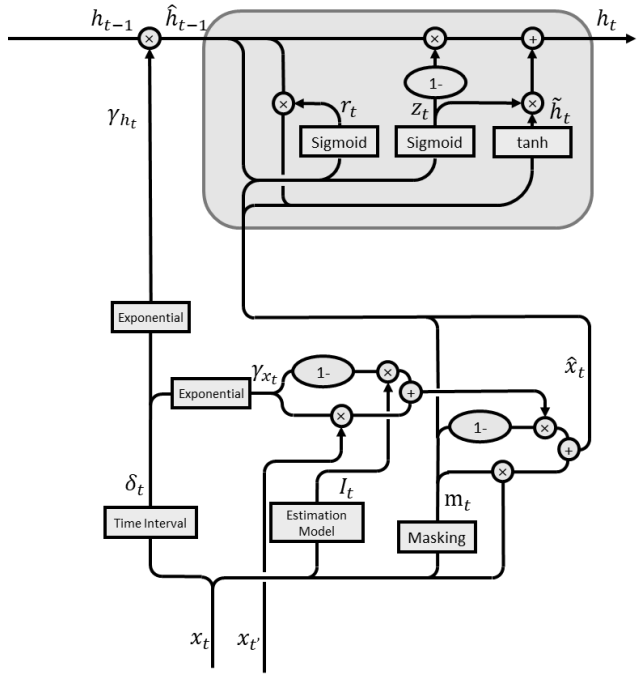


Figure 4. 5 Structure of Decayed, Passive Models

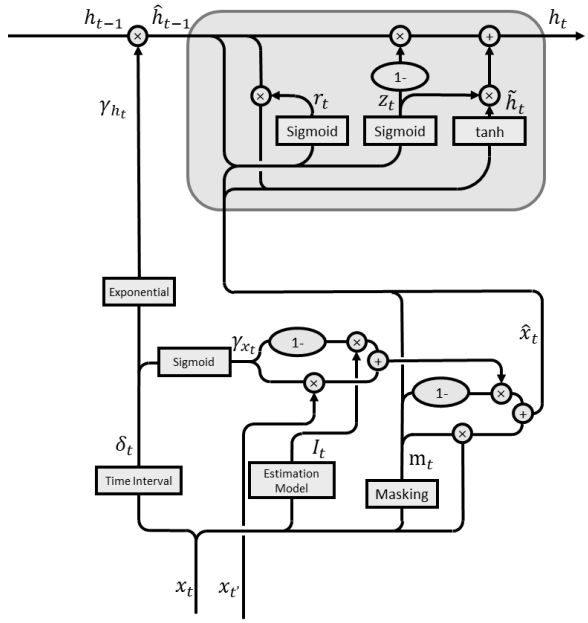


Figure 4. 6 Structure of Adjusted Input Model

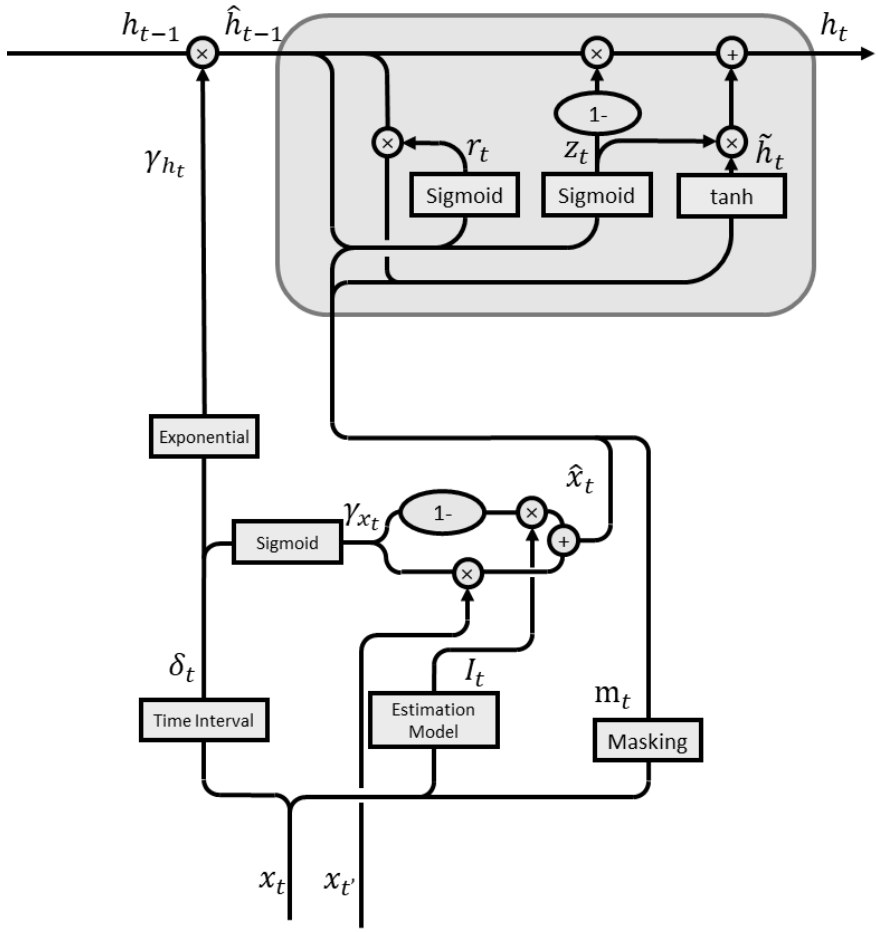


Figure 4. 7 Structure of Active Model

Each model can be used separately for individual data columns. Finally, in this study, we made the final concatenate model to learn and use the appropriate ratio between Decayed Model and Active Model for each data. This method is done by learning both the Decayed Model and the Active Model and then inserting a linear filter between the two models.

As will be described later, the active model showed the best performance in general naive estimation. However, as found in this study, we can assume that the

average speed of the low-performance platoon converges to a certain average value. Therefore, the decayed method may be more useful. In this study, we developed a model that uses both methods, considering the complexity of the calculation.

4.2.4.3. Performance of Internal Imputation Methods

As a result of examining the accuracy of the internal imputation candidates to select the internal imputation method, the models that performed the Imputation by Simple Moving Average (SMA), which is a linear combination method, showed lower performance than the trained imputation model. This is a natural result as the occurrence of low correlation problems cannot be controlled.

The accuracy of Active Imputation is higher than that of GRU-D because the direction of increase/decrease of the imputation value provided by GRU-D always goes to the average. In this study, we used the Active Imputation Method, which has the highest accuracy, as the Internal Imputation Method. However, as there is data with high kurtosis among platoon features, the missing term was adjusted by learning the value and ratio calculated through the decayed term at the same time.

Table 3. 11 Accuracy of Imputation Methods

Type	Model	MAE (km/h)	RMSE (km/h)
SMA Imputation	MLP	5.202	8.884
	LSTM	4.932	7.365
	GRU	5.053	7.892
Trained Imputation	GRU-D	4.580	5.931
	Decayed Imputation (GRU)	4.266	5.780
	Passive Imputation (GRU)	4.627	6.131
	Adjusted Input Imputation (GRU)	4.723	7.133
	Active Imputation (GRU)	3.839	5.096

4.2.5. Selected Dropout Filtering Method

In this study, we used the selected dropout layer to reflect the different transition probability of traffic states. This layer adjusts the hidden term assuming that each state has different propagation dynamics. Application to each filter is made in the following way.

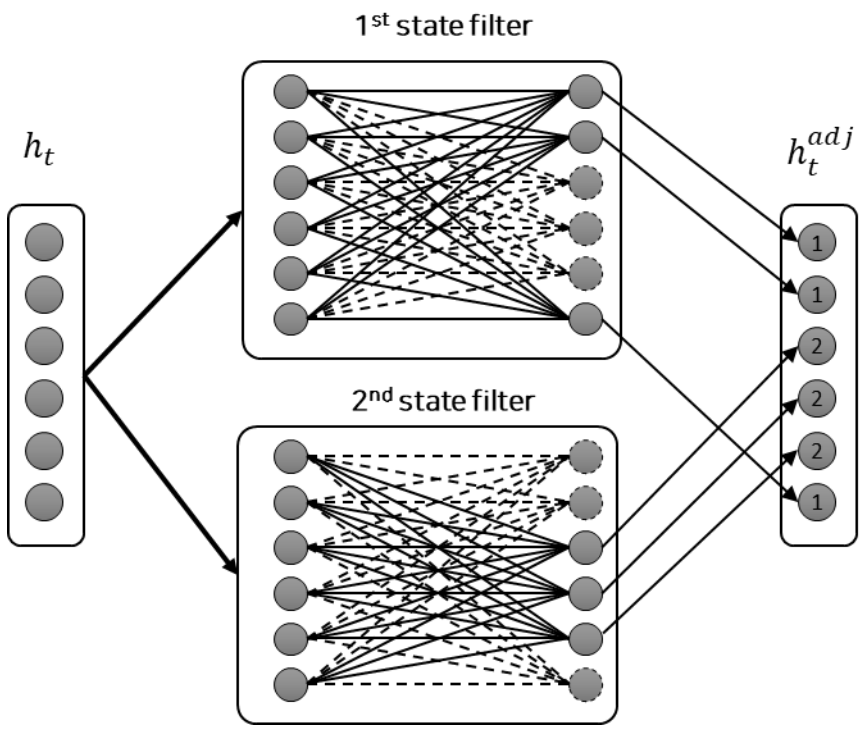


Figure 4. 8 Concept of Selected Dropout Method

In this way, the model can train the different transition probability according to the state of each link. Theoretically, if there are enough data, the model can use this

method to achieve at least the accuracy of not using the selected dropout. In this study, identifications through Jenks natural break were applied, as made in Chapter 3. Therefore, each state is reflected in the model by applying the selected dropout filter.

Figure 4. 9 Overall Description of the Model

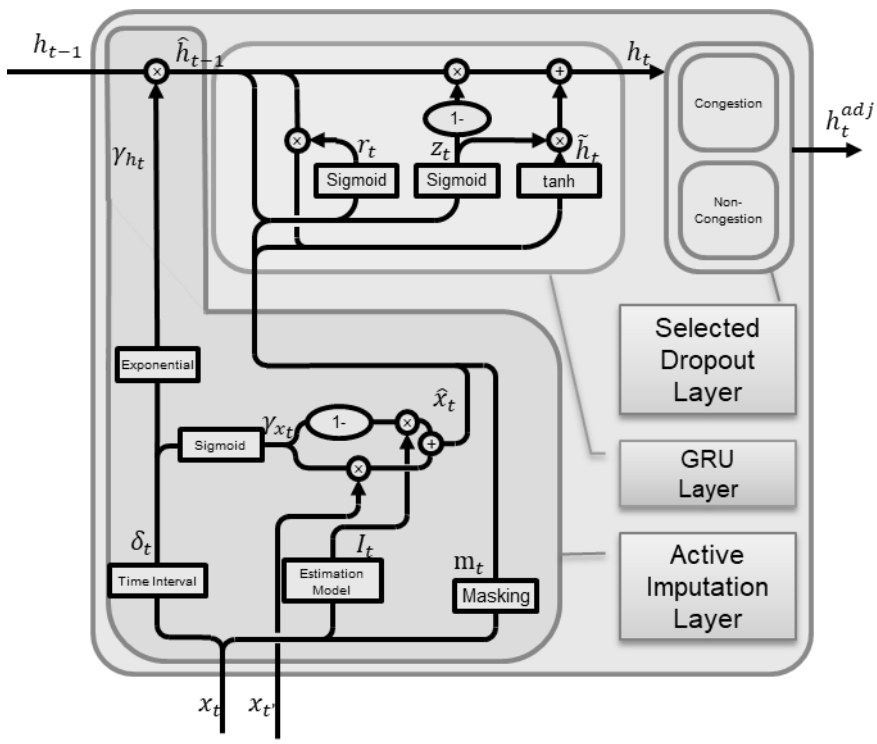


Figure 4.9 show the overall description of the developed model.

Chapter 5. Result and Findings

5.1. Estimation Accuracy of Developed Models

5.1.1. Average Accuracy of Developed Models

5.1.1.1. Average Accuracy for Mean Speed

Table 5.1 and Figure 5.1 shows the accuracy of the existing and developed models. All models based on deep learning were trained through 1000 epochs and prevented overfitting through validation data. Besides, to compare the same conditions, the naive model imputation of missing data into a random forest used for active imputation.

As a result, the filtered two-step model with the selected dropout filter showed the best performance. Linear regression models did not differ significantly, but in general, the support vector machine with the best results was unexpectedly low. As a result of applying LSTM to the two-step model, the selected dropout filter was not easily changed due to the characteristic that the hidden term and the cell term were transmitted separately. In addition, accuracy showed a slight drop.

Table 5. 1 Average Estimation Accuracy of Models

Type	Model	MAE (km/h)	RMSE (km/h)
Existing	Simple Average	12.879	15.102
	Linear Regression (Single Time)	5.987	8.163
	Linear Regression (Time Series)	5.743	7.855
	Linear Support Vector Regression	7.494	9.713
	Naive Random Forest	4.466	6.390
	Naive LSTM	4.307	5.852
	Naive GRU	4.415	6.005
Developed	Two-step Model (LSTM)	3.001	4.158
	Filtered Two-step(LSTM)	3.035	4.221
	Two-step Model (GRU)	3.207	4.638
	Filtered Two-step(GRU)	2.725	3.752

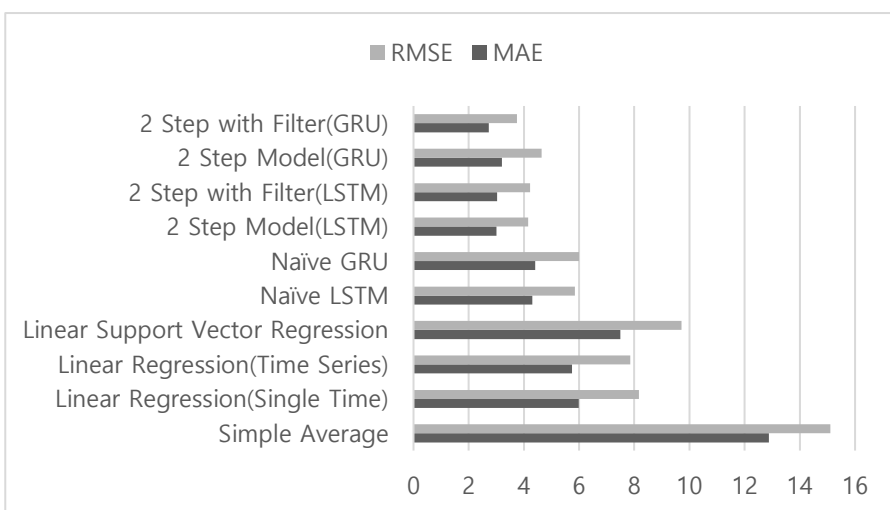


Figure 5. 1 Average Estimation Accuracy of Models

5.1.1.2. Average Accuracy for Platooning Feature

The results of estimating the three features (HPP Speed, LPP Speed, LPP Ratio) separated by the Plating Effect are shown in Table 5.2 and Figure 5.2. The average estimation accuracy for each feature did not show any significant difference depending on the filtering. This result means that the filtered model does not rely on the improvement of the accuracy of one feature but improves the accuracy based on the estimation of several variables. However, the difference in estimation accuracy between the filtered model and the non-filtered model varies greatly depending on the situation.

Table 5. 2 Average Accuracy of Model by Platooning Features

MAE	Filtered	Non-filtered
High-perf Speed(km/h)	2.866	2.901
Low-perf Ratio (%)	4.940	5.767
Low-perf Speed(km/h)	2.093	2.289
Mean Speed(km/h)	2.741	3.210

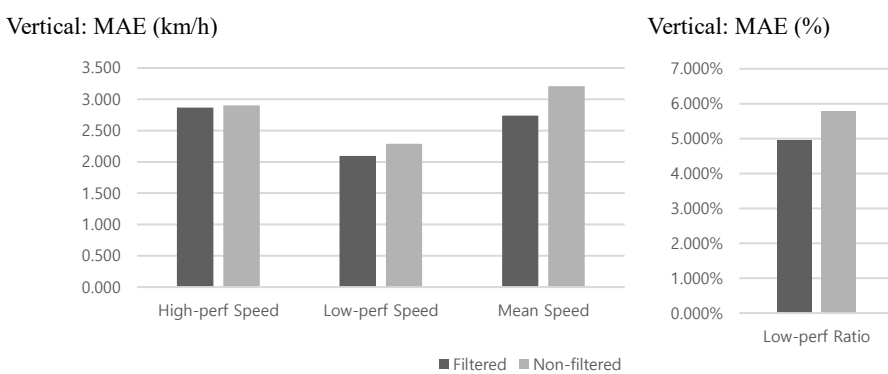


Figure 5. 2 Average Accuracy of Model by Platooning Features

5.1.2. Accuracy of Developed Models for Each Link

According to the estimation accuracy of each link, the two-step GRU model showed better accuracy than Naive LSTM on all links. Based on the MAE, the filtered model showed the best results in 11 links except for 458 and 422 links. In the case of link 458, the filtered two-step model showed better results in terms of RMSE. The non-filtered two-step model showed better results with one exception (392) compared to the Naive LSTM.

The non-filtered two-step model showed robust results for up to four traffic signals in the link but showed less accuracy than the naive LSTM model in seven cases. However, even in this case, the filtered two-step model showed better performance than the naive LSTM model. The segment length per signal of each link is almost constant, about 280m. Table 5.3 shows the result of each link.

Table 5. 3 Accuracy of Models for Each Link

Link No.	Traffic Signal on Link	Length of Link	Mean Speed	Naive LSTM		Two-step GRU (Non-filtered)		Two-step GRU (Filtered)	
				MAE	RMSE	MAE	RMSE	MAE	RMSE
112	1	779.8	25.447	3.823	5.125	2.753	3.997	2.480	3.428
114	0	583.2	28.708	2.919	3.784	2.542	3.272	1.874	2.423
116	1	1410.0	43.640	8.955	13.836	6.300	10.929	5.624	8.101
392	7	2017.4	29.388	2.191	3.045	3.001	3.656	1.468	2.085
123	4	1412.0	30.719	5.740	7.839	3.925	6.063	3.374	4.871
125	2	984.4	30.109	2.822	3.903	2.013	2.973	1.833	2.635
442	1	512.7	65.368	4.159	5.542	2.770	3.989	2.373	3.357
546	1	530.3	41.728	4.057	5.271	2.688	3.853	2.344	3.237
458	1	583.6	48.655	4.751	6.164	3.133	4.487	3.325	4.313
422	3	805.9	30.941	2.864	3.710	2.468	3.202	2.566	3.213
540	2	1071.6	30.587	3.805	4.966	3.487	4.461	2.401	3.234
130	2	775.8	42.660	6.182	8.017	4.153	5.860	3.604	4.984
350	4	1387.3	55.714	3.720	4.870	2.461	3.556	2.157	2.889

Unit: km/h

5.2. Correlation Analysis of Developed Model

5.2.1. Correlation Coefficient Analysis

As a result of examining the Spearman correlation coefficient with each model's upstream and downstream link data, the filtered two-step model showed the highest result in all cases. This was constant, regardless of accuracy.

Naive LSTM is similar to the random forest but shows a slightly higher level of correlation. This suggests that the relationship between the data can be estimated more clearly by reflecting the platooning and adjusting the transition.

Vertical: Spearman Corr. Coefficient(0~1.0)

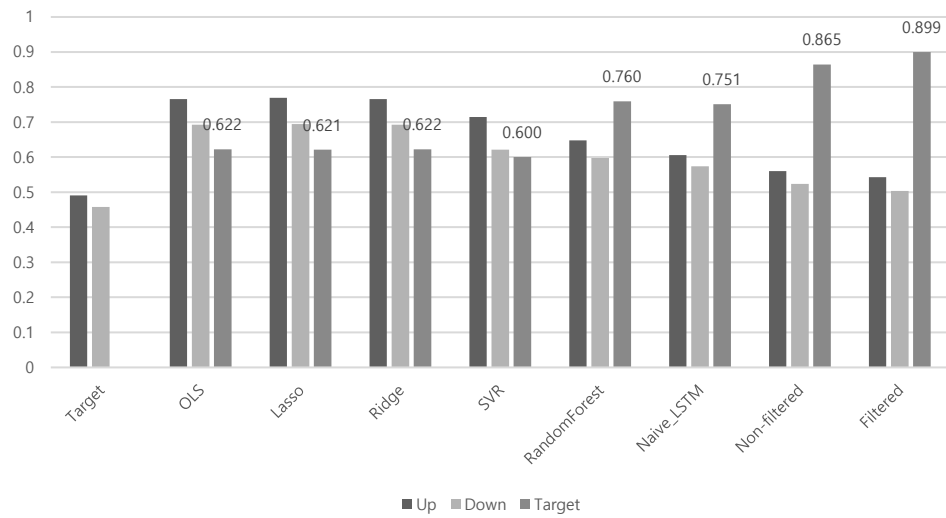


Figure 5. 3 Spearman Correlation Analysis for Models

The Spearman correlation was higher in the filtered model, even when the average accuracy of the filtered model was relatively low (links 458 and 422). In all cases, the correlation coefficient was higher for the non-filtered model than for the naive model, and the coefficient for the filtered model was improved compared to the non-filtered model. This means that even when the average accuracy of the Filtered Model is relatively low (links 458 and 422), the relationship between the data is clearly estimated.

Vertical: Spearman Corr. Coefficient(0~1.0)

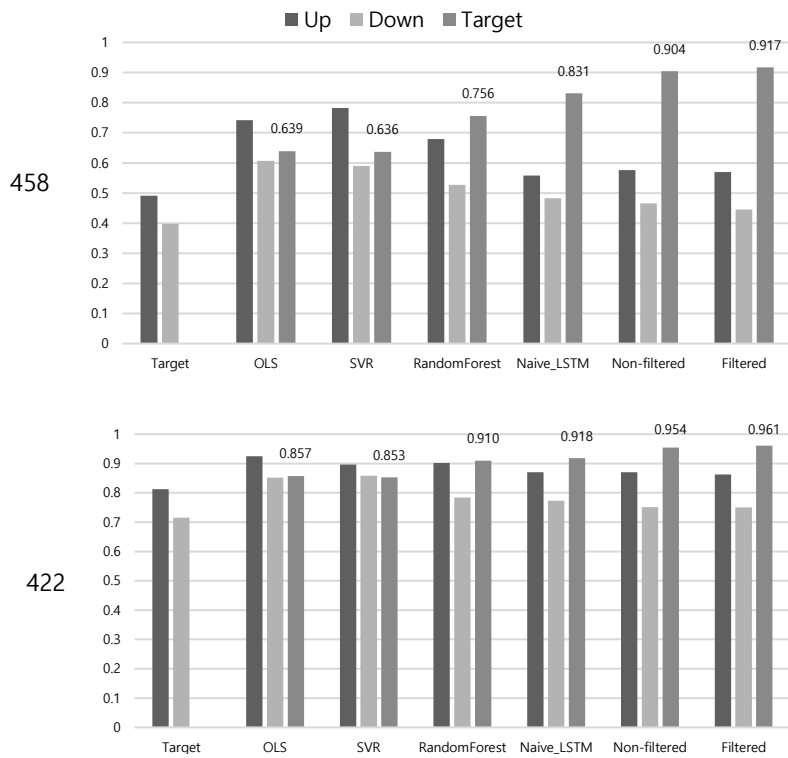


Figure 5. 4 Spearman Correlation Analysis for Low Accuracy Case

5.2.2. Speed Change Analysis

In this paper, as in the data analysis, two analyzes of variation were performed. One is to analyze whether the direction of change is consistent, and the other is to analyze the correlation coefficient of change.

As a result of the analysis of speed direction change, the naive LSTM has a lower direction consistency than the OLS. In contrast, the two-step model has increased consistency in all cases, regardless of whether it is filtered or not.

In the case of the correlation coefficient analysis of the speed value, the correlation of Naive LSTM was lower than that of the OLS or rather negative correlation. In contrast, the two-step model increased in all cases.

Table 5. 4 ΔSpeed Direction Coincidence Probability

Link	ΔSpeed Direction Coincidence Probability Probability(%)					
	Upstream	Downstream	OLS	Naive LSTM	Non-filtered	Filtered
112	56.3	59.6	61.5	63.1	76.5	77.4
114	58.5	50.2	58.9	34.5	68.3	72.7
116	49.7	50.4	51.2	37.7	66.1	68.6
392	50.3	52.9	53.9	37.7	60.8	64.8
123	52.6	54.6	54.3	41.7	70.3	74.5
125	54.6	68.1	69.7	71.6	79.8	80.7
442	68.0	73.2	80.9	82.8	87.1	89.4
546	73.2	66.8	77.3	78.1	83.6	85.0
458	66.3	62.7	77.3	68.3	78.0	77.7
422	62.7	64.5	69.1	65.0	74.5	75.6
540	64.4	67.2	70.2	62.7	70.8	75.0
130	66.7	65.0	72.5	78.8	85.3	87.0
350	66.1	48.9	68.4	62.5	77.6	82.0

Table 5. 5 ΔSpeed Correlation Analysis

Link	ΔSpeed Correlation Coefficient					
	Upstream	Downstream	OLS	Naive LSTM	Non-filtered	Filtered
112	0.111	0.242	0.248	0.265	0.651	0.726
114	0.233	0.018	0.241	-0.432	0.502	0.675
116	0.015	0.063	0.067	-0.298	0.520	0.631
392	0.059	0.097	0.126	-0.376	0.312	0.450
123	0.079	0.100	0.105	-0.226	0.593	0.721
125	0.100	0.323	0.361	0.390	0.692	0.744
442	0.319	0.475	0.613	0.683	0.851	0.904
546	0.475	0.376	0.582	0.596	0.799	0.859
458	0.366	0.276	0.425	0.374	0.690	0.726
422	0.271	0.402	0.510	0.417	0.686	0.734
540	0.402	0.362	0.518	0.362	0.624	0.757
130	0.357	0.170	0.454	0.581	0.794	0.856
350	0.184	-0.016	0.199	0.258	0.696	0.824

5.3. Periodicity Analysis for Developed Models

5.3.1. Periodicity reflection analysis

The analysis results of the development model and the periodicity learning of the naive LSTM are as follows. First, the two-step model succeeded in learning periodicity in all cases where periodicity appeared. However, the filtered model reproduced the autocorrelation value closer to the target data when analyzing periodicity than the non-filtered model. In the case of the Naive LSTM model, it is generally known that the periodicity can be learned, and most of the cases with periodicity have been learned.

However, the naive LSTM model did not learn periodicity for the case that has a low autocorrelation coefficient with periodicity (Link 123, Figure 5.5) And naive LSTM also fails to learn periodicity where small autocorrelation amplitude (Link 114).

As described above, two-step models learn the periodicity even if the naive model does not learn the periodicity. In the case of link 392, which showed deficient periodic characteristics, the Naive model was more accurate than the non-filtered model, but the correlation was weak. Table 5.6 below shows the case that the naive LSTM model does not learn periodicity.

Horizontal: Time Lagging(minutes)(5~95)
Vertical: Pearson autocorrelation coefficient(0~1.0)

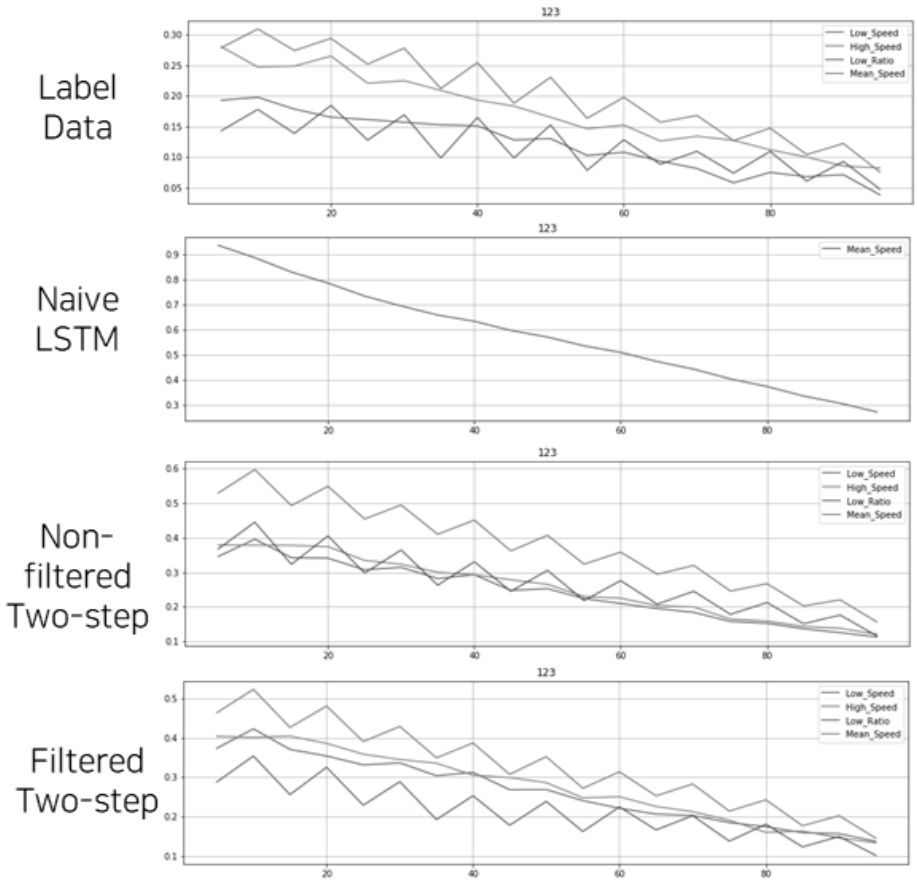


Figure 5. 5 Cases That the Naïve LSTM Fails to Learn Periodicity (Link 123)

Table 5. 6 Cases That the Naïve LSTM Fails to Learn Periodicity

Link	Periodicity Property	MAE		
		Naive	Non-filtered	Filtered
123	Low Autocorrelation Value	5.740	3.925	3.374
114	Small Amplitude of Autocorrelation	2.919	2.542	1.874
392	Vague Periodicity	2.191	3.001	1.468

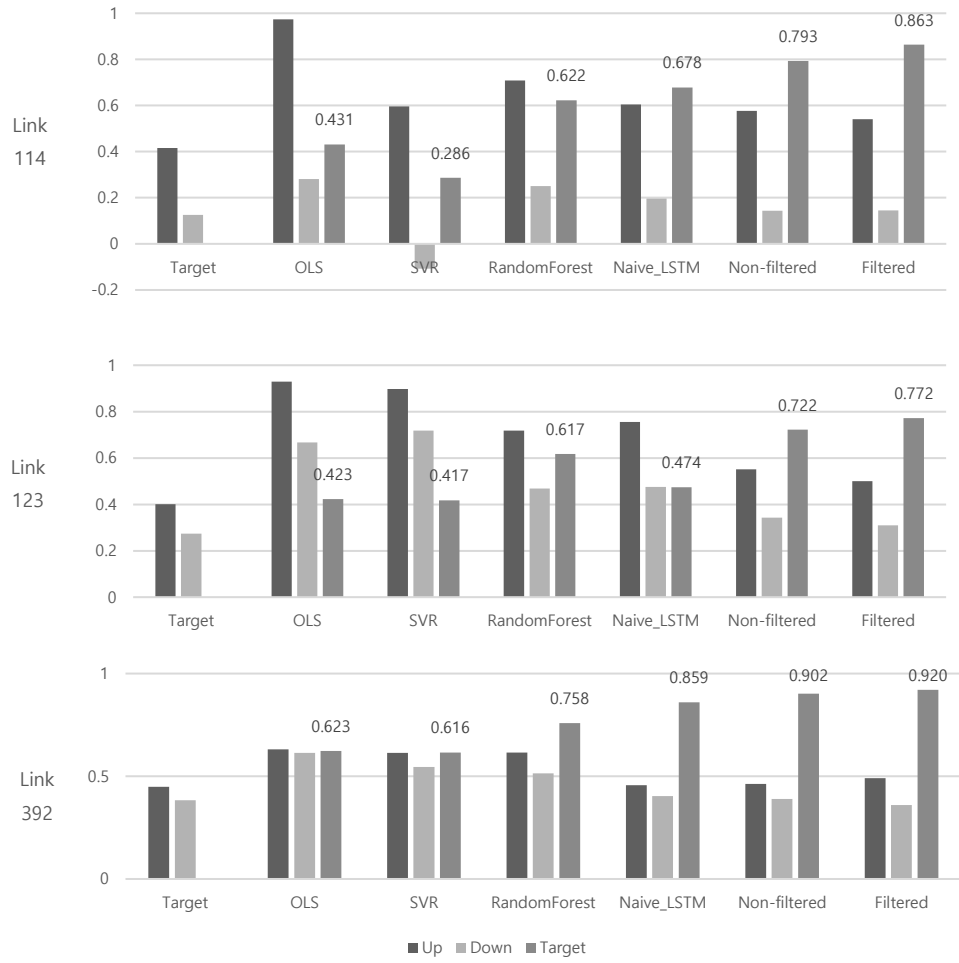


Figure 5. 6 Correlation Analysis for Periodicity Failure Cases of LSTM

For the case where Naive LSTM fails periodicity learning, the model's correlation and accuracy are as follows. Where the case that naive LSTM model did not learn periodicity due to the periodicity with a small amplitude of the correlation (Link 114), the improvement in accuracy was close to average. However, LSTM's Correlation is lower than the average (0.751), and the improvement to Correlation is

27.2%, above the average (19.7%). Change analysis also showed a significant decrease in correlation with Naive LSTM.

Second, the case that the naive LSTM model cannot learn the periodicity due to the low value of autocorrelation (Link 123), the correlation coefficient of naive LSTM is low. And the accuracy improvement is 40.7%, relatively high to the average (36.8%).

In cases where the periodicity was barely revealed (Link 392), the accuracy of the naive LSTM was higher than that of the non-filtered model, but the correlation was low. In the analysis of speed change, the two-step model showed a significant improvement.

5.3.2. Variable Dependency Analysis with LIME

Figure 5.7 shows the results of evaluating the model's time dependence using the LIME algorithm. As revealed in chapter 3, the Naive LSTM model reflects the correlation of data as the structure of the model itself.

This training feature can be seen as the superiority of the LSTM model in most cases. Still, if the periodicity cannot appears to be inherent in the model, the naive LSTM model cannot learn the periodicity. It also shows a correlation reverse vulnerability for periodicity.

In the case of the two-step model, both the filtered and non-filtered models show that the learning progresses in the direction of the periodicity that appears in the given data itself. Therefore, the correlation reverse does not occur, so it can be

seen that there is no vulnerability to periodicity.

Horizontal: Time lagging(min.)

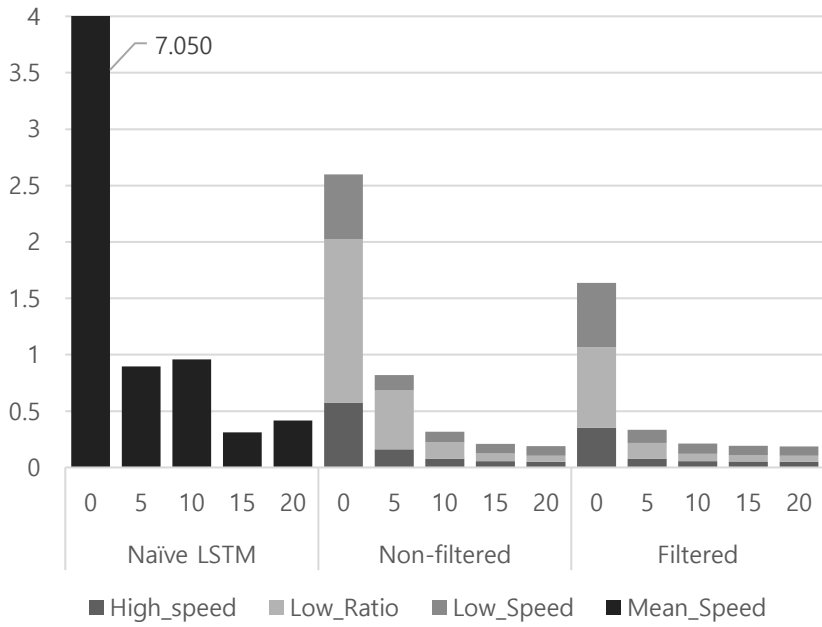


Figure 5. 7 Variable Dependency Analysis

5.4. Accuracy Analysis by Traffic State

5.4.1. Sudden Change in Speed

Table 5. 7 Accuracy of Models for Sudden Change in Speed

Criteria	State	MAE of Models(km/h)		
		Naive LSTM	Non-filtered Model	Filtered Model
5km/h	Change under 5km/h(A)	3.177	2.601	2.312
	Change over 5km/h(B)	5.777	3.886	3.263
	Difference (B-A)	2.600	1.285	0.951
10km/h	Change under 10km/h(A)	3.455	2.717	2.413
	Change over 10km/h(B)	8.103	5.154	4.113
	Difference (B-A)	4.649	2.437	1.700

Model accuracy analysis for sudden-change states is shown in Table 5.7 and Figure 5.8. All models showed a decrease in accuracy for the sudden-change state. However, the smallest is for the filtered model, and the largest is for the Naive LSTM. The larger the width of the change criteria, the greater the weakness of the LSTM

model. Since no definite correlation was found between the sample number and the accuracy of the sudden-change state, this does not mean a difference in dominance due to the sample number.

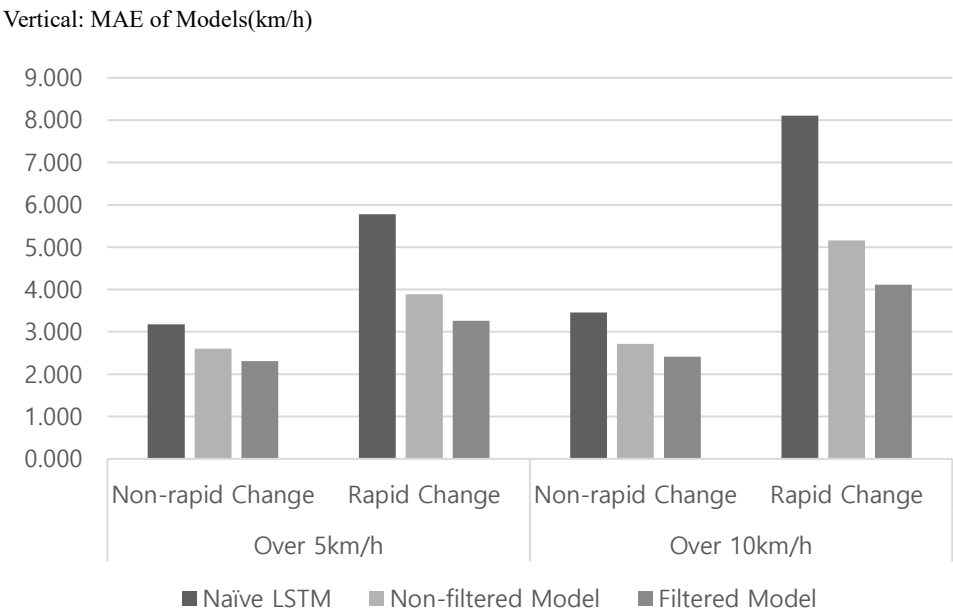


Figure 5. 8 Accuracy of Models for Sudden Change in Speed

Figures 5.9 and 5.10 show the difference in accuracy between the filtered and non-filtered models for platooning features for sudden-change states. In the change to sudden-change states, LPR showed the most significant difference, but this is not noticeable. Therefore, it can be seen that the model operates in the direction of increasing the accuracy of all variables for the sudden-change state.

As will be described later, this is considered to be the reason that the filtered model shows complete robustness for the transition, but the results are relatively

robust compared to other models for the sudden change.

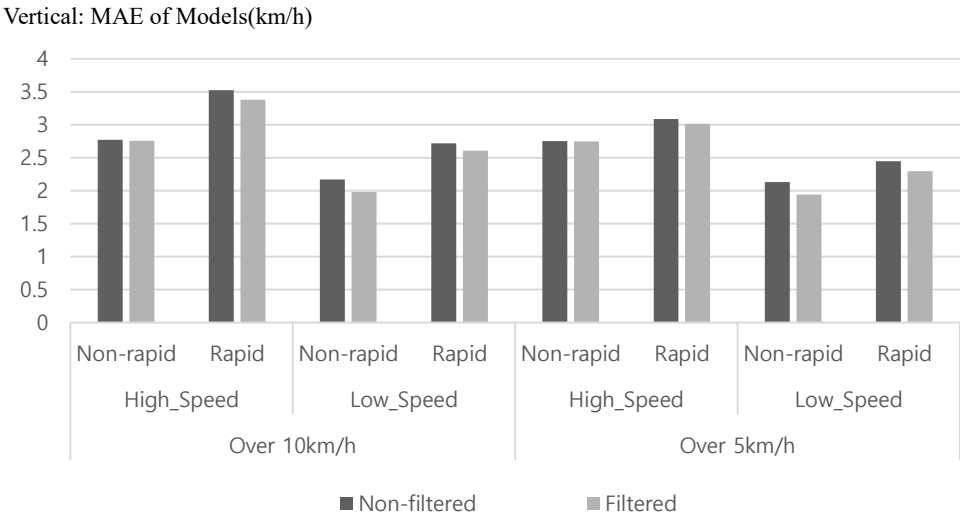


Figure 5. 9 Accuracy of Models for Sudden Change in Speed (LPP Speed, HPP Speed)

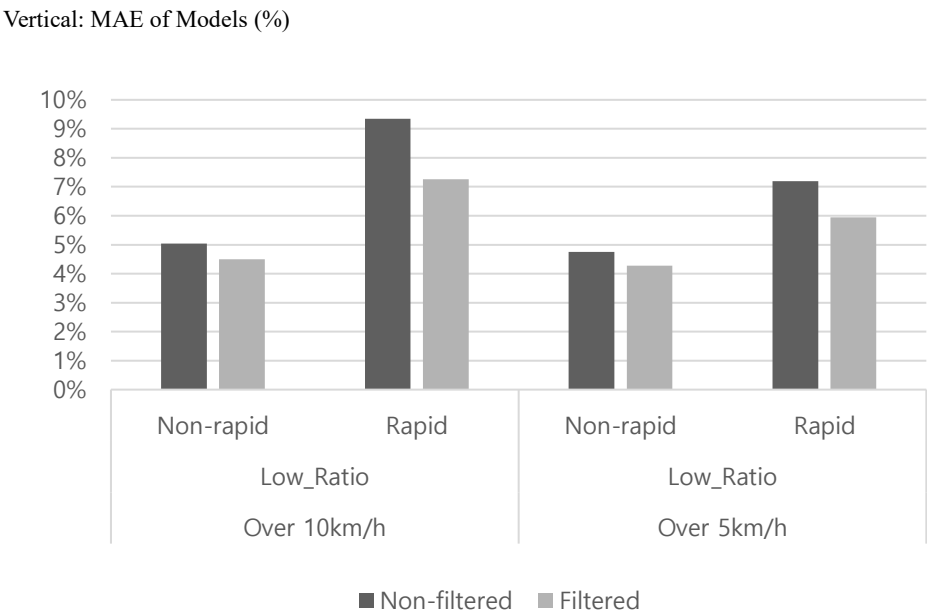


Figure 5. 10 Accuracy of Models for Sudden Change in Speed (LPR)

5.4.2. Transition State

Table 5.8 shows the accuracy of each model for the transition state in which the traffic state transitions to or resolves the congestion. The naive LSTM model shows an adequate level of accuracy for non-transition conditions but very low accuracy for transition conditions.

On the other hand, the filtered two-step model showed robust results in all cases. There was no correlation between the accuracy of each state and the number of state samples. The effect of learning can estimate the increase in accuracy by dividing different transition probabilities.

Table 5. 8 Accuracy of Models for Transition State

Condition	State	MAE of Models(km/h)		
		Naive	Non-Filtered	Filtered
Mean		4.341	3.210	2.741
Transition	Transition(A)	6.695	4.434	2.882
	Non-transition(B)	3.920	3.015	2.761
Difference (A-B)		2.775	1.419	0.121

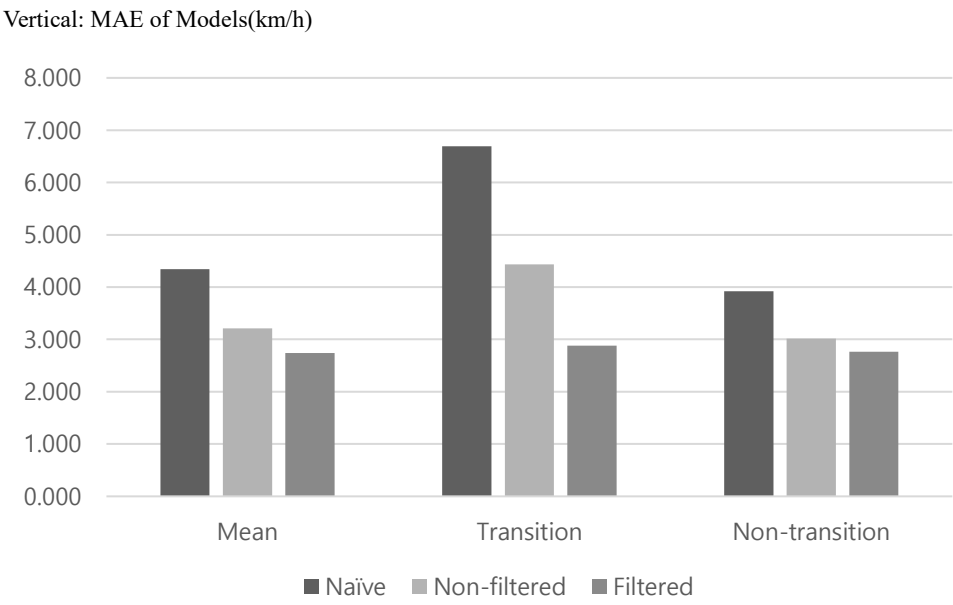


Figure 5. 11 Accuracy of Models for State Transition State

Figure 5.12 and Figure 5.13 are graphs showing the difference in the accuracy of the platooning features of the filtered and non-filtered models. Unlike the sudden-change state, the accuracy of the LPR is noticeably increased in this case. This result shows that there is a synergy between the selected dropout filter and the two-step model.

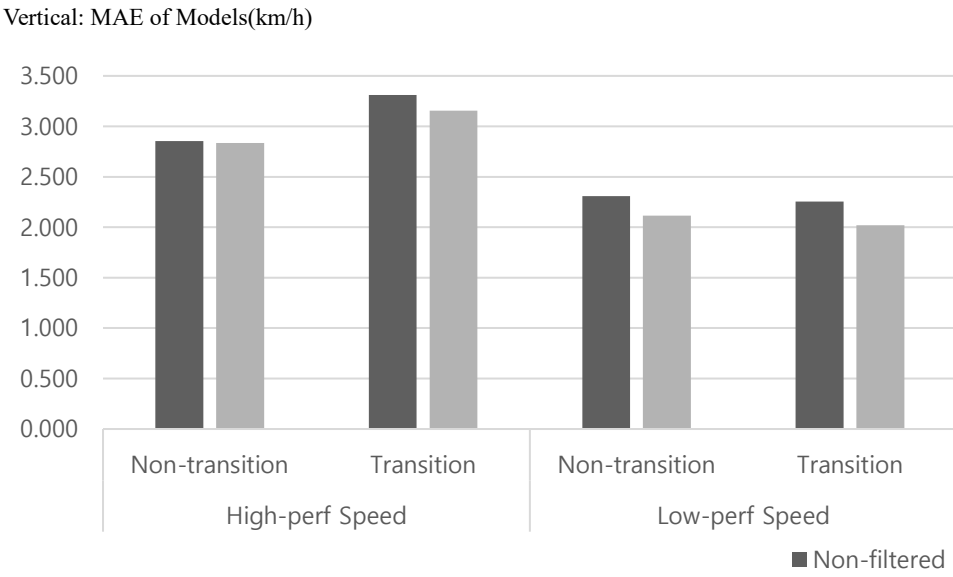


Figure 5. 12 Accuracy of Models for State Transition State (HPP Speed, LPP Speed)

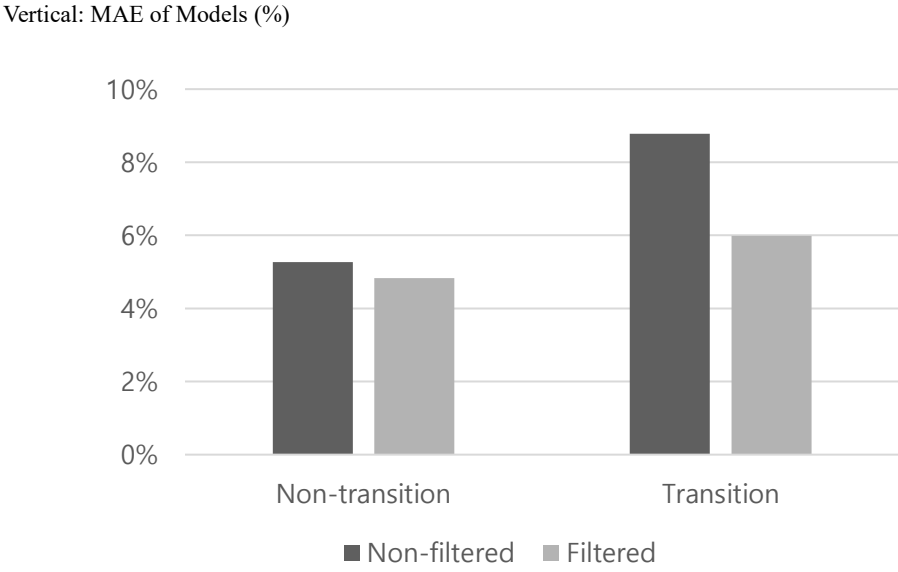


Figure 5. 13 Accuracy of Models for State Transition State(LPR)

5.5. Summary of the Result

In Chapter 5, this paper estimates the average travel speed on the link through the development model and overcomes the limitations of the existing model. Main model development conditions and introduction methods are as follows.

First, due to the low correlation with the surrounding links and the occurrence of periodicity, the existing model cannot guarantee high accuracy. In this paper, a modified recurrent neural network was developed to solve this problem.

Second, as a result of analyzing the platoon separation to solve the problem, according to periodicity, the link speed data is composed of a mixture of data showing non-periodic characteristics and data showing periodic characteristics. In this paper, we developed a two-step model to estimate the separation clearly.

Third, this paper proposed congestion filtering through the selected dropout layer to reflect different transition probabilities for each state in the data.

Fourth, the active imputation to appropriately respond to the imputation of missing data with high frequency was proposed.

As a result of estimating the model reflecting the above points, the following points are improved.

First, the development model enables high accuracy estimation.

Second, active imputation showed good performance in response to missing data.

Third, in contrast to the naive model, the two-step model learned the periodicity based on data rather than correlation.

Fourth, the filtered model showed robustness against accuracy for several traffic states, and the two-step model and the congestion filter method showed mutual synergy.

In conclusion, this paper demonstrates that the model developed through the above-described conditions shows excellent performance in general and for specific situations.

Chapter 6. Conclusion

6.1. Summary

This paper aims to develop a model for estimating the mean speed of a link that shows the characteristics of interrupted flow on an urban network. To this end, we analyzed the characteristics of the interrupted flow using DSRC system data collected from Daegu city from January to June 2018 and identified the problems when using the existing model. The problems of applying the existing model defined by the empirical foundation and data are as follows.

First, the speed data of the urban traffic network link shows a low correlation with the neighboring links. These characteristics reduce the goodness of fit of the linear combination model. To solve this problem, this paper proposes a neural network model using time series data. Also, it was found that the dependence reversal phenomenon occurs in machine learning due to the periodic characteristics, and the development of a recurrent neural network model using the platoon separation phenomenon solved the problem.

For the development, we designed a recurrent neural network model suitable for interrupted flow. It consists of several features. First, we developed a two-step model using the platoon separation. Second, a selected dropout filter to control the transition state probability is developed and applied. Third, an active imputation

method was developed and applied to deal with frequent missing data. Finally, this paper combines three neural network layers to develop an estimation model suitable for urban network traffic speed. This paper examined whether the limitations of the existing models could be overcome by analyzing the estimation values acquired through the development model.

In conclusion, the methods proposed in this paper overcome the limitations of the existing models, and in particular, yield good results in certain traffic states. We also found that there is a synergy effect between development models. Through this, it was possible to estimate the sections that were difficult to estimate with the existing model and to increase the accuracy of the overall estimation.

6.2. Limitation of the Study

As the most critical limitation, this study used the bimodal speed distribution of the vehicle, but the result is that only the average value of each feature, which is one of the parameters. This point is a limitation in that it cannot present the distribution of the speed of each vehicle and cannot present parameters except average values. This is because this study has limited bimodal distribution to separate parameters.

As another limitation, this study assumed that the bimodal distribution was the difference in the queueing due to the signal based on the previous research results, but did not analyze it clearly. This is because the data related to the signal could not be collected. In terms of the bimodal distribution described above, periodicity can

be expected to appear due to the difference in effective green for each aggregation cycle. This difference is assumed to be due to the difference in the signal cycle and the aggregation time. However, no clear test on this has been done due to data limitations. Also, there is a need to verify that there is transposability to something other than Daegu's local system.

Another problem with data limitations is the lack of traffic or density data, which may be positive in terms of limited data use but serves as a limitation for state identification. In addition, this study assumes platoon separation, which is a limited phenomenon in interrupted flow and is not suitable for uninterrupted flow.

It should be noted that the results of this study are the models used for the estimation and not the models used for forecasting. Even though it can be using the findings from this model for forecasting, it requires a more appropriate research.

6.3. Applications and Future Research

The suggestions for overcoming the limitations of this study and for further research are as follows. First, it is necessary to study whether the bimodal distribution and periodicity, which are the characteristics of the interrupted flow used in this study, are caused by queueing due to signal operation, as assumed in this study. The progress of this study requires data related to the signal phase.

Regarding the separation of platoons, the deep learning model used in this study aims at fitting numerical values, so it is necessary to analyze the distribution of actual

speed data through other models. For example, a Gaussian mixture model may be used to confirm the separation of the actual distribution.

As for periodicity, studies on the aggregation unit should be conducted to clarify whether periodicity in speed data is caused by a difference between aggregation units and signal cycles. It is expected that this result can be obtained by changing the aggregation unit. However, since the problem of dependence reversal is the same in a continuous flow, it is necessary to make a model applicable to continuous flow.

The result of this study can be used as a framework, and if the underlying data is provided, it can be expected to be able to be expanded to estimate other traffic data with propagation such as volume and density. Similarly, further research is needed to develop this model and use it for forecasting rather than estimation.

Lastly, estimating the attributes of a link has practical limitations, and it is necessary to apply machine learning to the estimation of path data. The results of this study are expected to be useful for estimating the cost function of such path data estimation.

Appendix

Appendix A. Correlation Analysis for Links

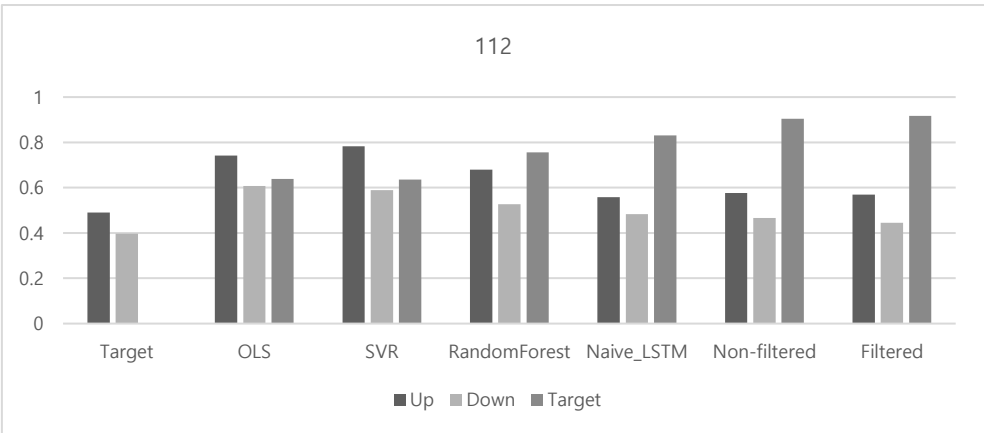


Figure A. 1 Correlation Analysis for Link No. 112

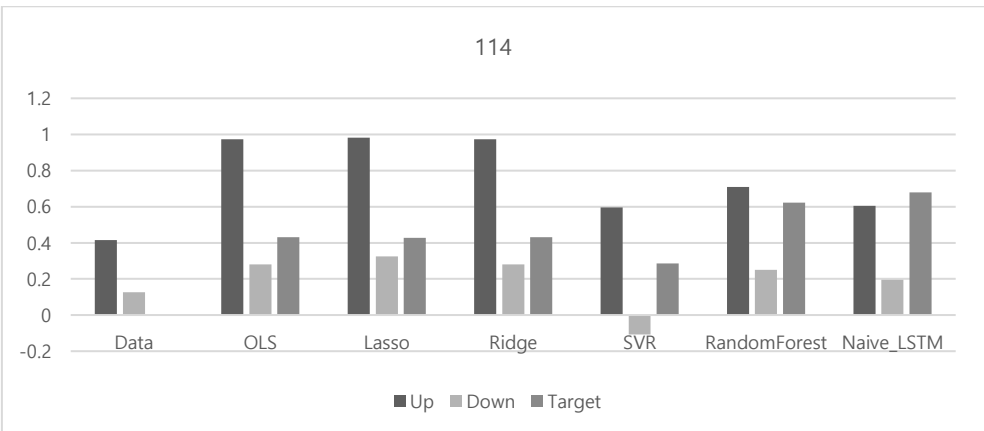


Figure A. 2 Correlation Analysis for Link No. 114

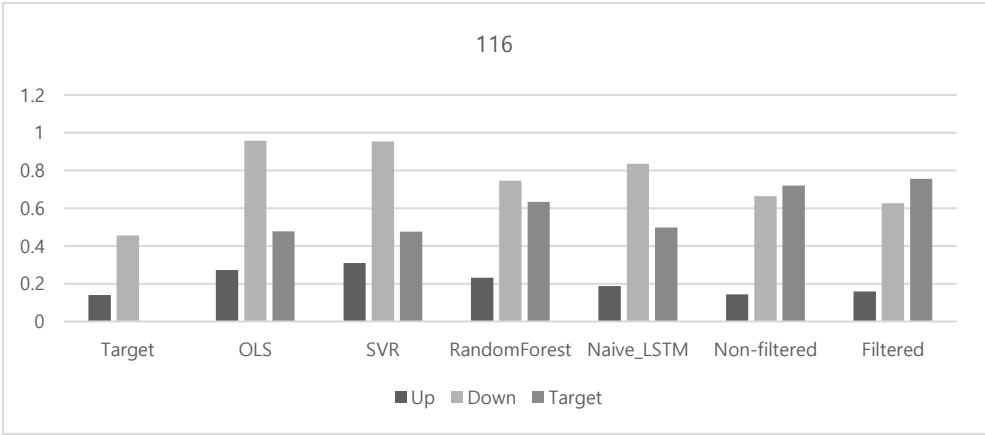


Figure A. 3 Correlation Analysis for Link No. 116

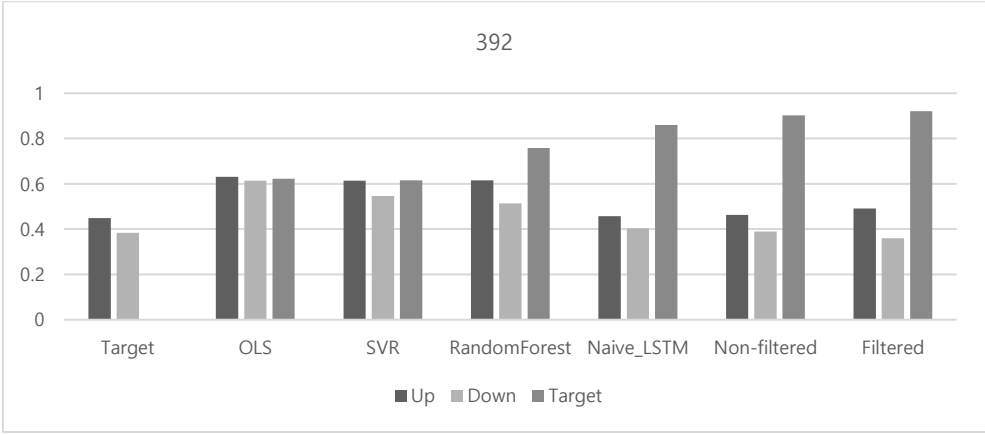


Figure A. 4 Correlation Analysis for Link No. 392

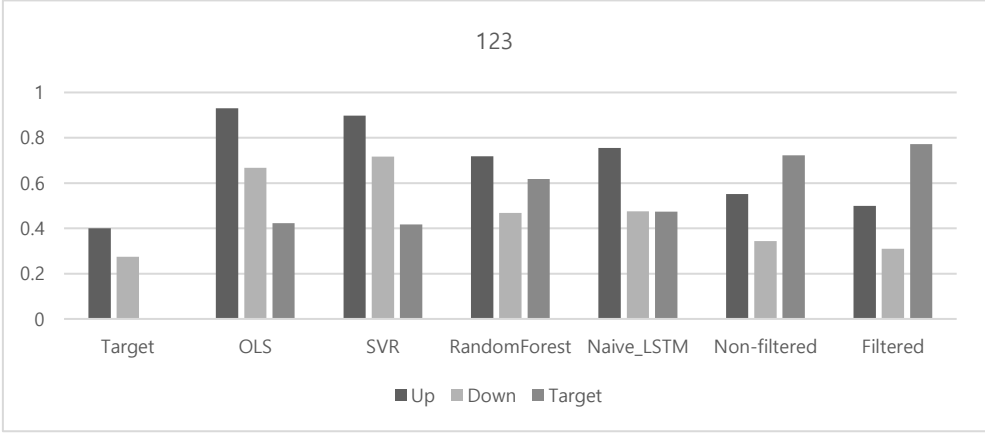


Figure A. 5 Correlation Analysis for Link No. 123

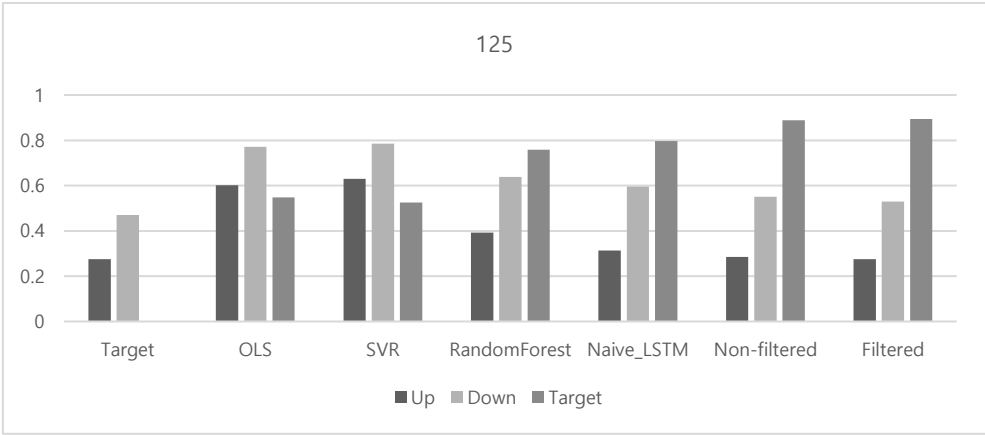


Figure A. 6 Correlation Analysis for Link No. 125

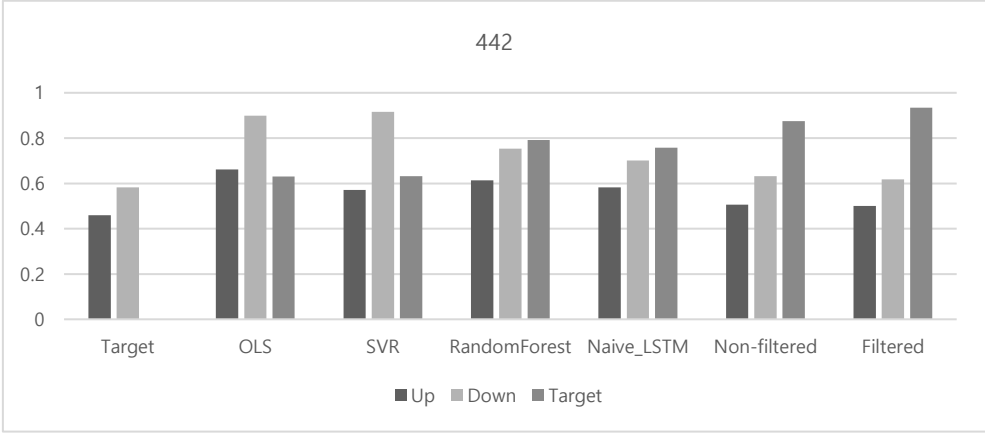


Figure A. 7 Correlation Analysis for Link No. 442

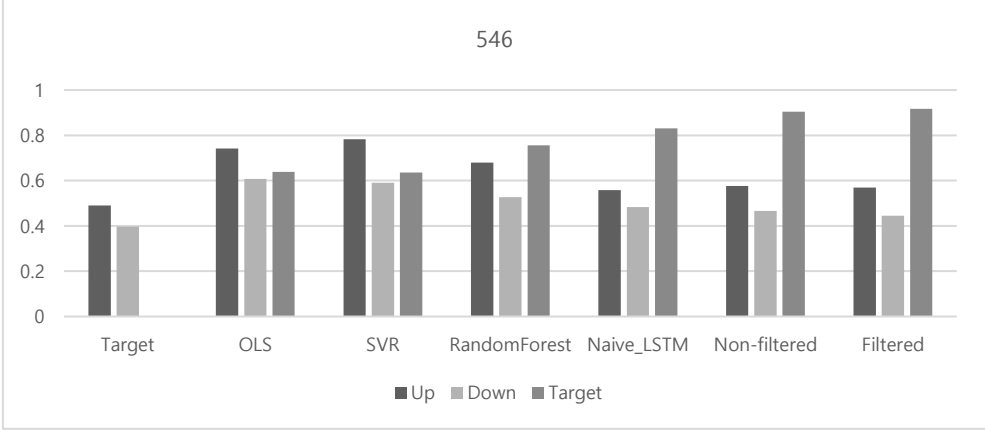


Figure A. 8 Correlation Analysis for Link No. 546

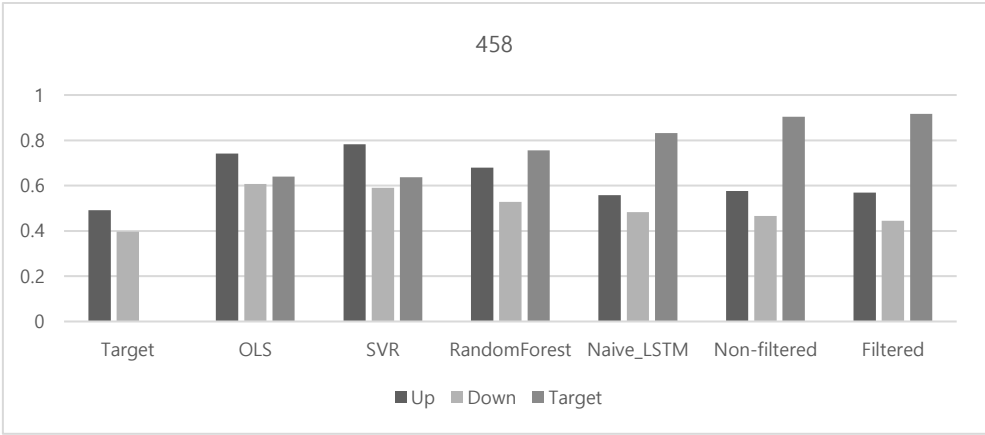


Figure A. 9 Correlation Analysis for Link No.458

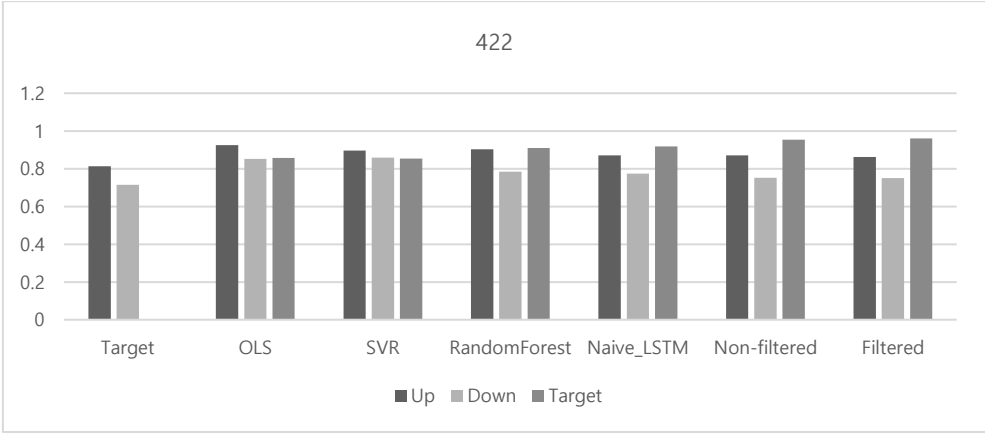


Figure A. 10 Correlation Analysis for Link No.422

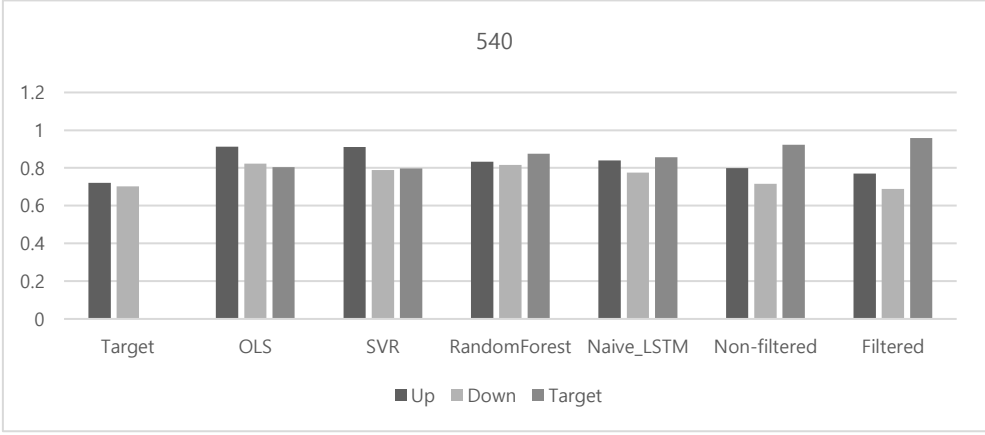


Figure A. 11 Correlation Analysis for Link No.540

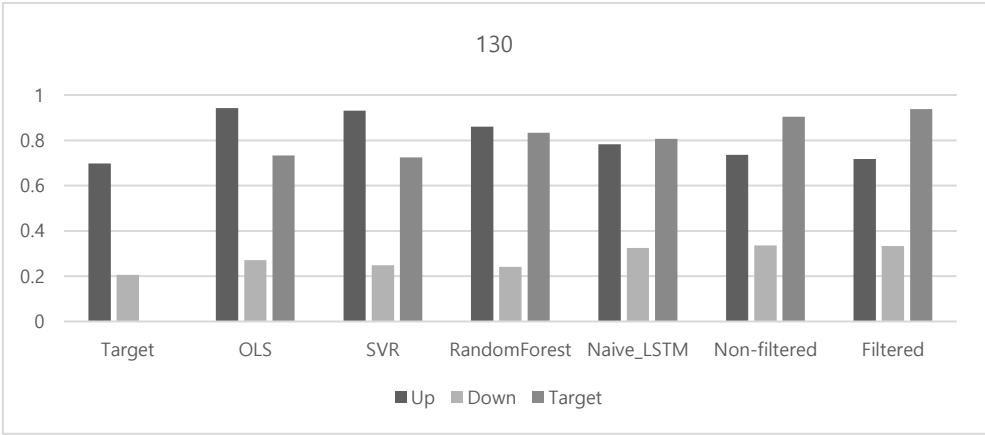


Figure A. 12 Correlation Analysis for Link No.130

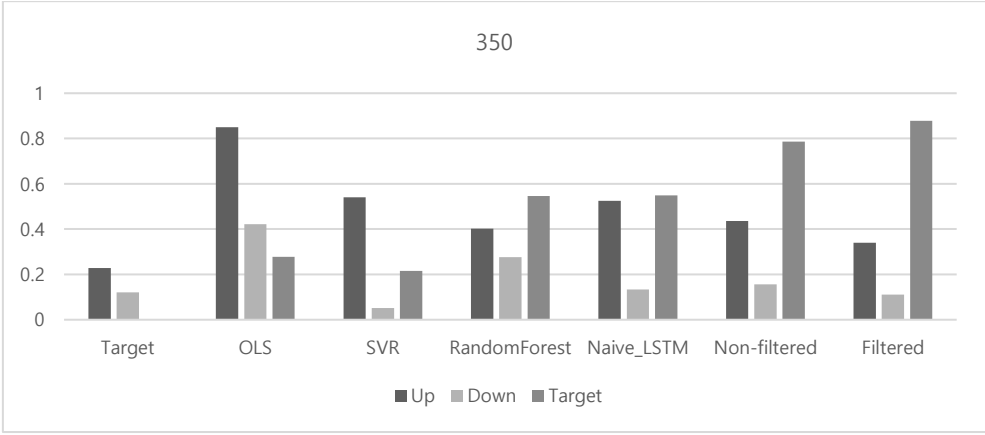


Figure A. 13 Correlation Analysis for Link No.350

Appendix B. Periodicity Analysis

B-1. Periodicity Analysis for Target Link Data

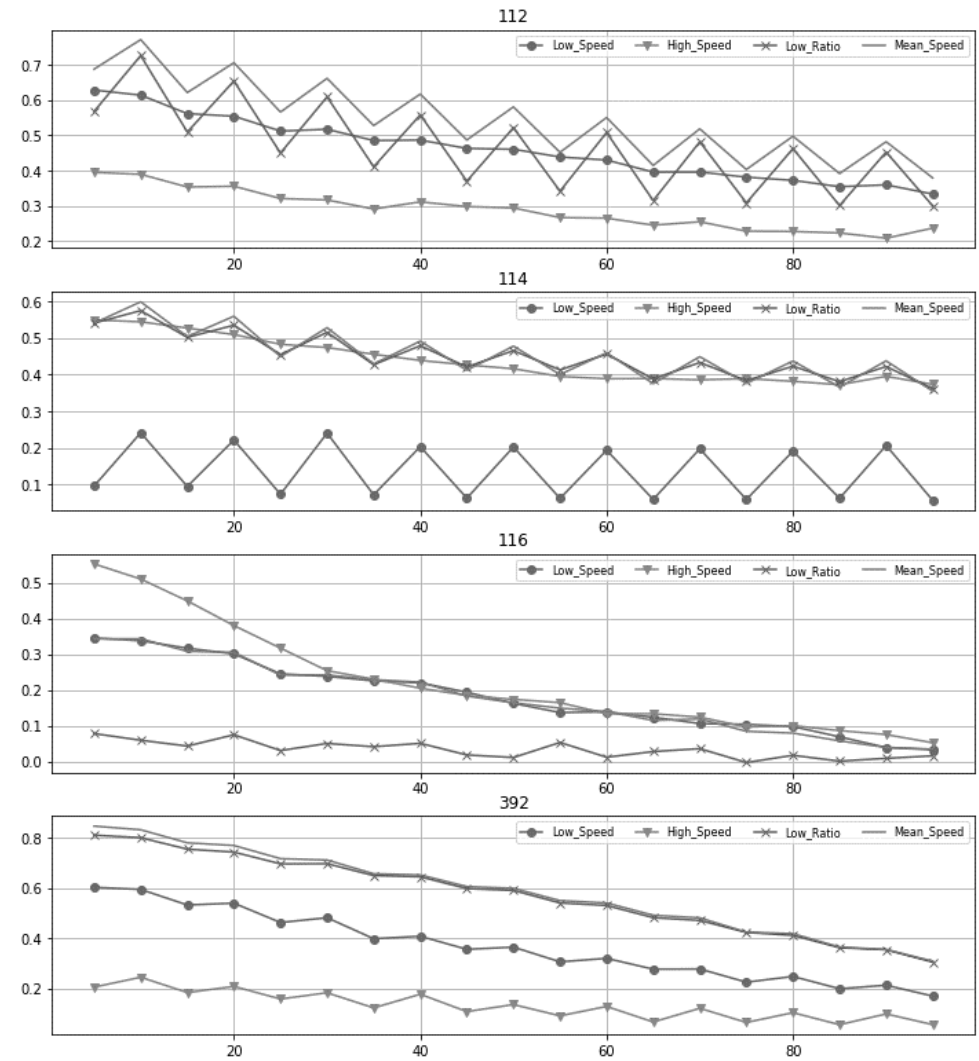


Figure B. 1 Periodicity Analysis for Target Link Data (Link 112, 114, 116, 392)

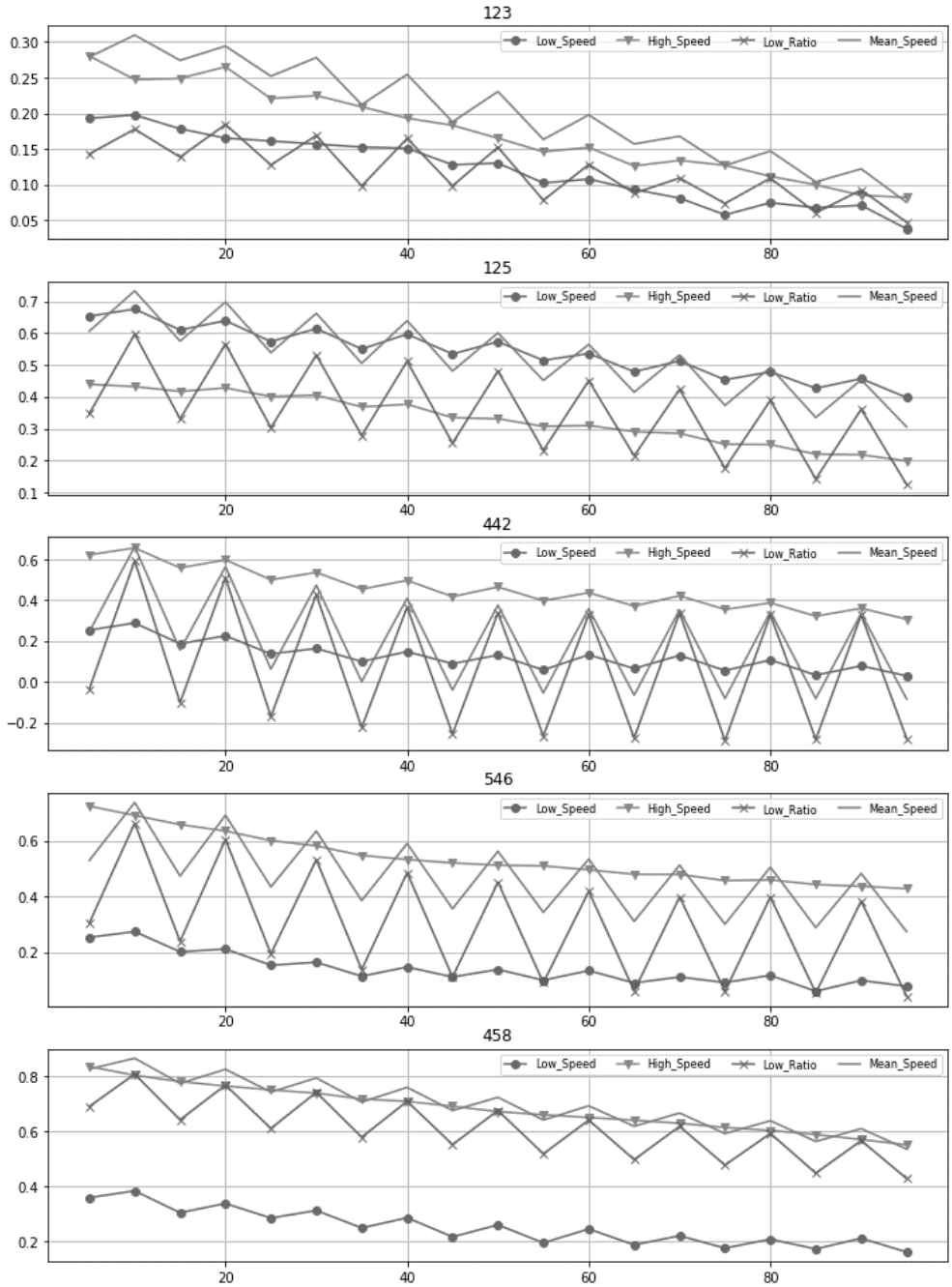


Figure B. 2 Periodicity Analysis for Target Link Data (Link 123, 125, 442, 546, 458)

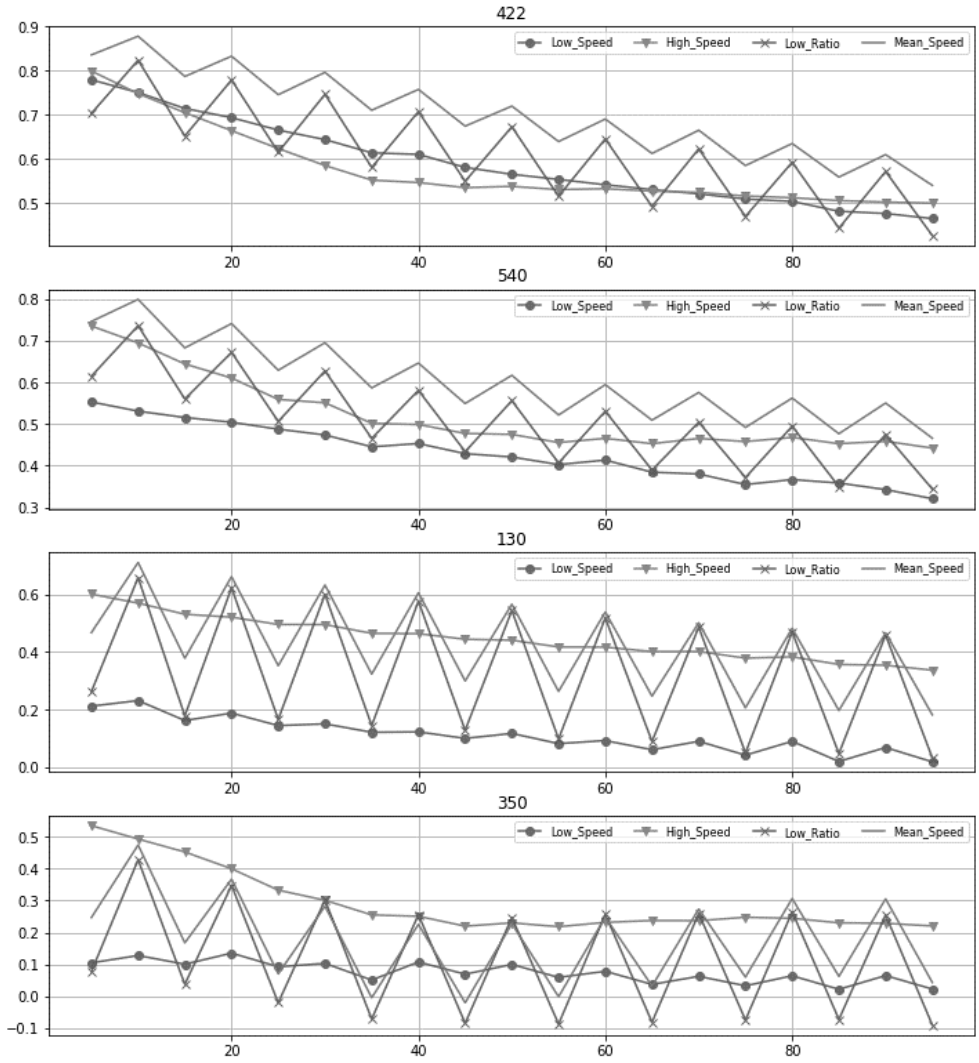


Figure B. 3 Periodicity Analysis for Target Link Data (Link 422, 540, 130, 350)

B-2. Periodicity Analysis for Naive LSTM Result

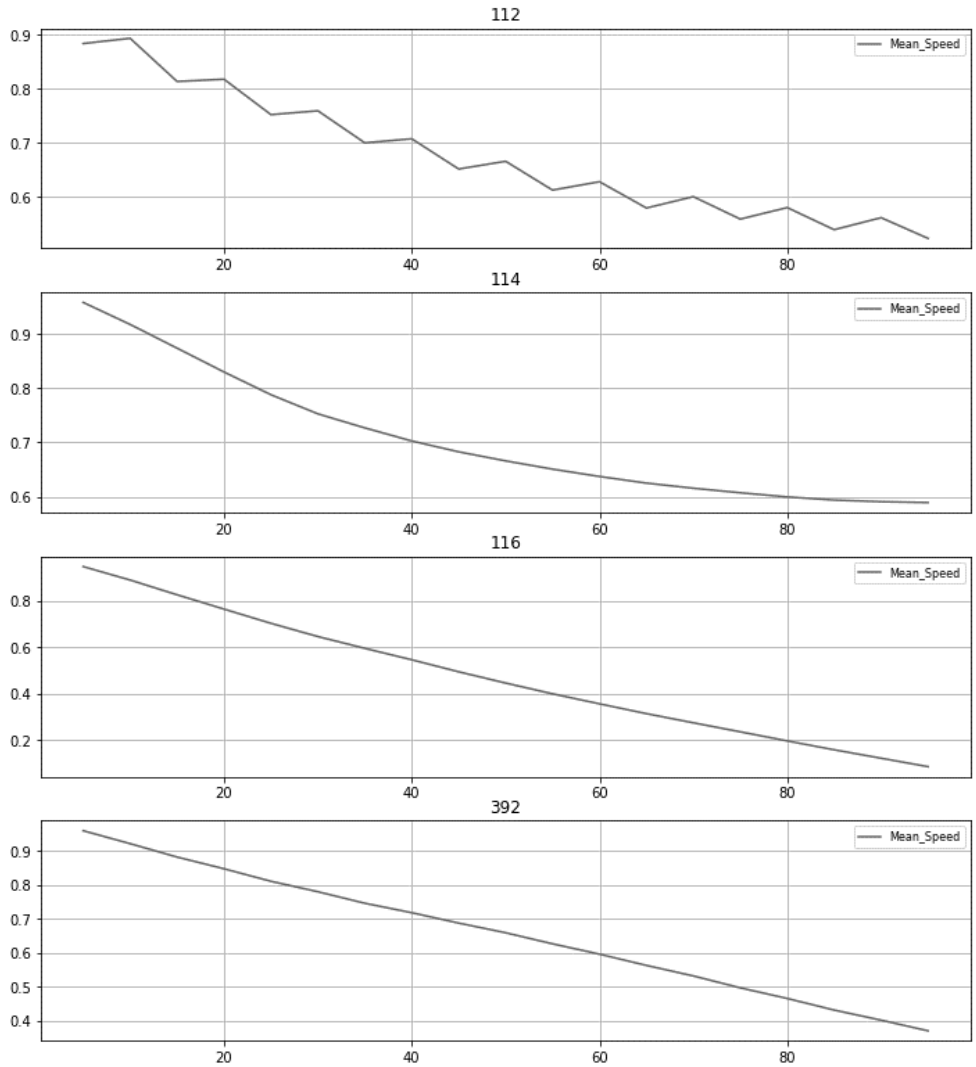


Figure B. 4 Periodicity Analysis for Naive LSTM Result (Link 112, 114, 116, 392)

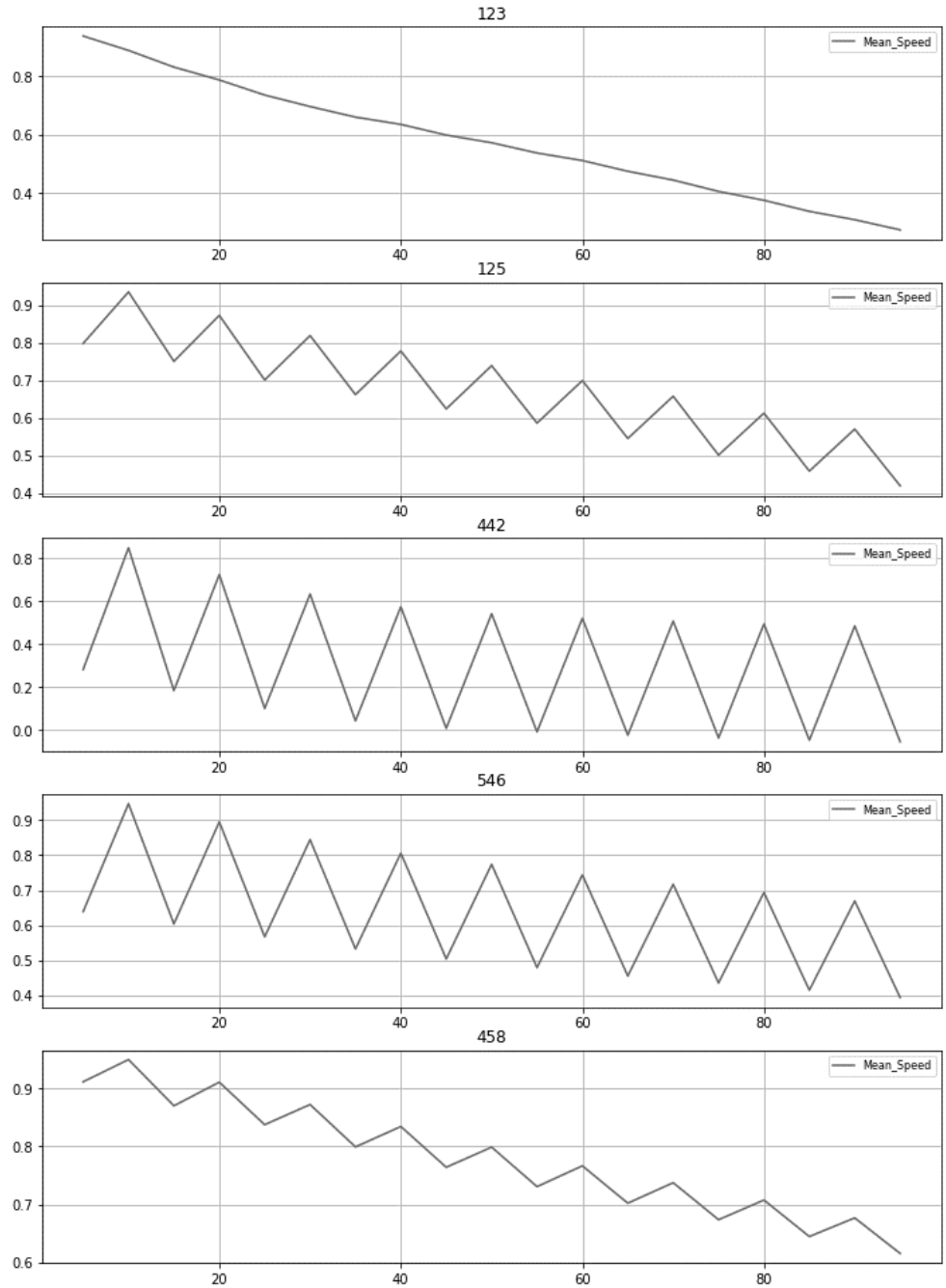


Figure B. 5 Periodicity Analysis for Naive LSTM Result (Link 123, 125, 442, 546, 458)

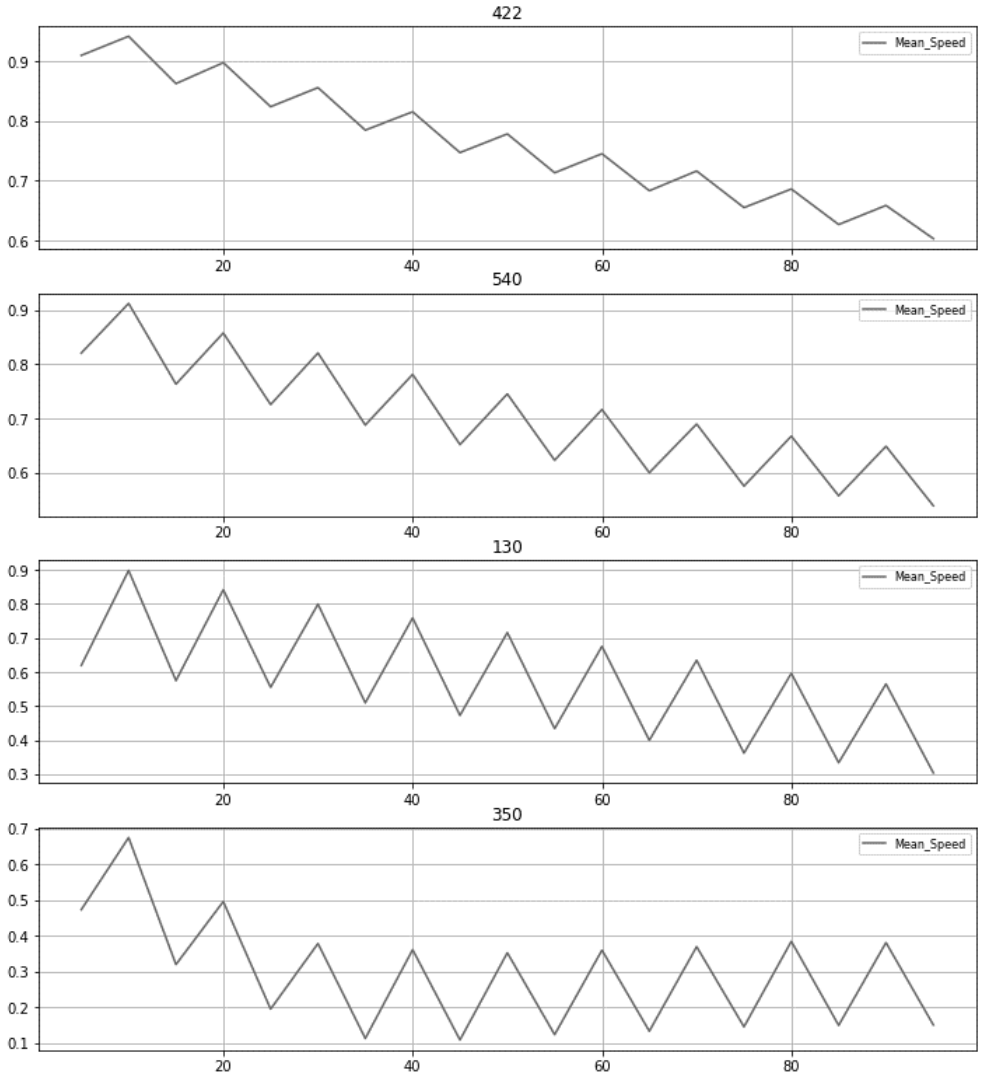


Figure B. 6 Periodicity Analysis for Naive LSTM Result (Link 422, 540, 130, 350)

B-3. Periodicity Analysis for Non-filtered Model

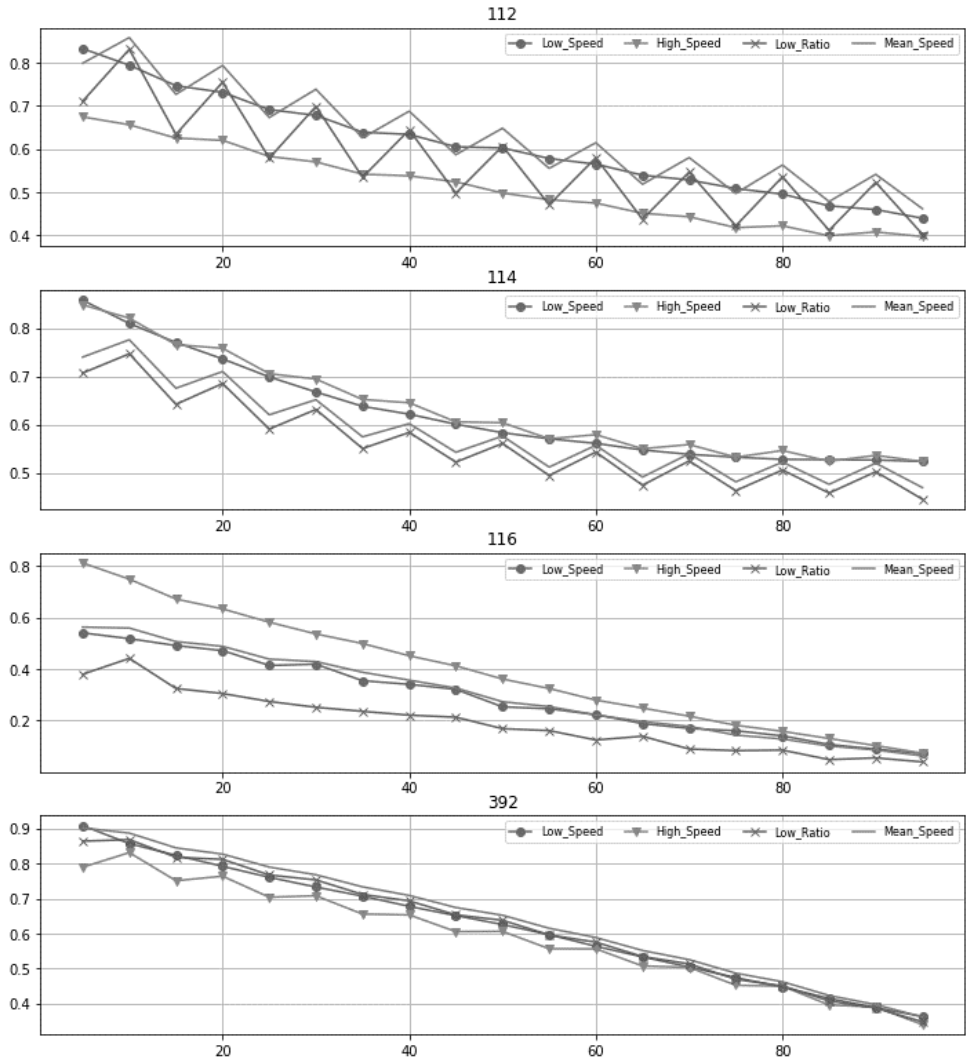


Figure B. 7 Periodicity Analysis for Non-filtered Model (Link 112, 114, 116, 392)

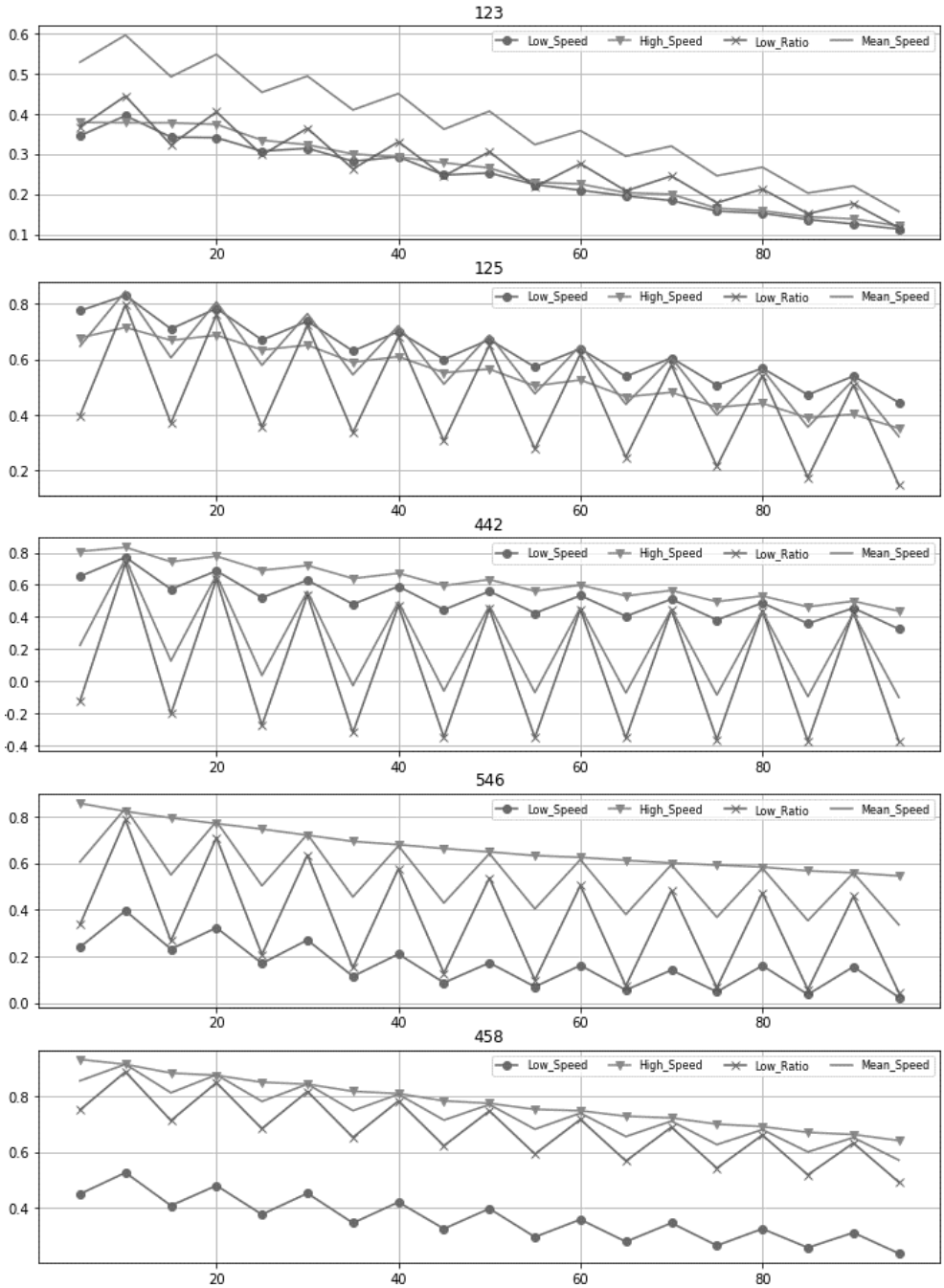


Figure B. 8 Periodicity Analysis for Non-filtered Model (Link 123, 125, 442, 546, 458)

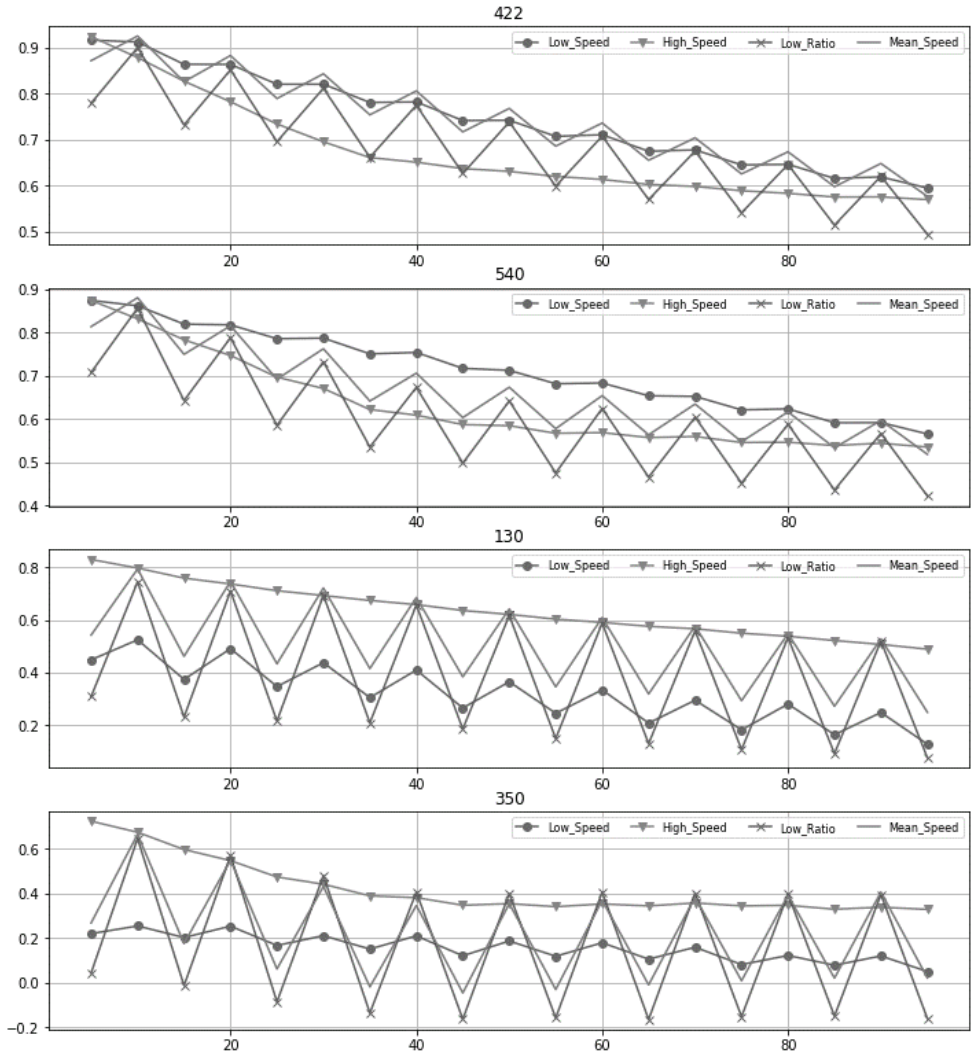


Figure B. 9 Periodicity Analysis for Non-filtered Model (Link 422, 540, 130, 350)

B-4. Periodicity Analysis for Filtered Model

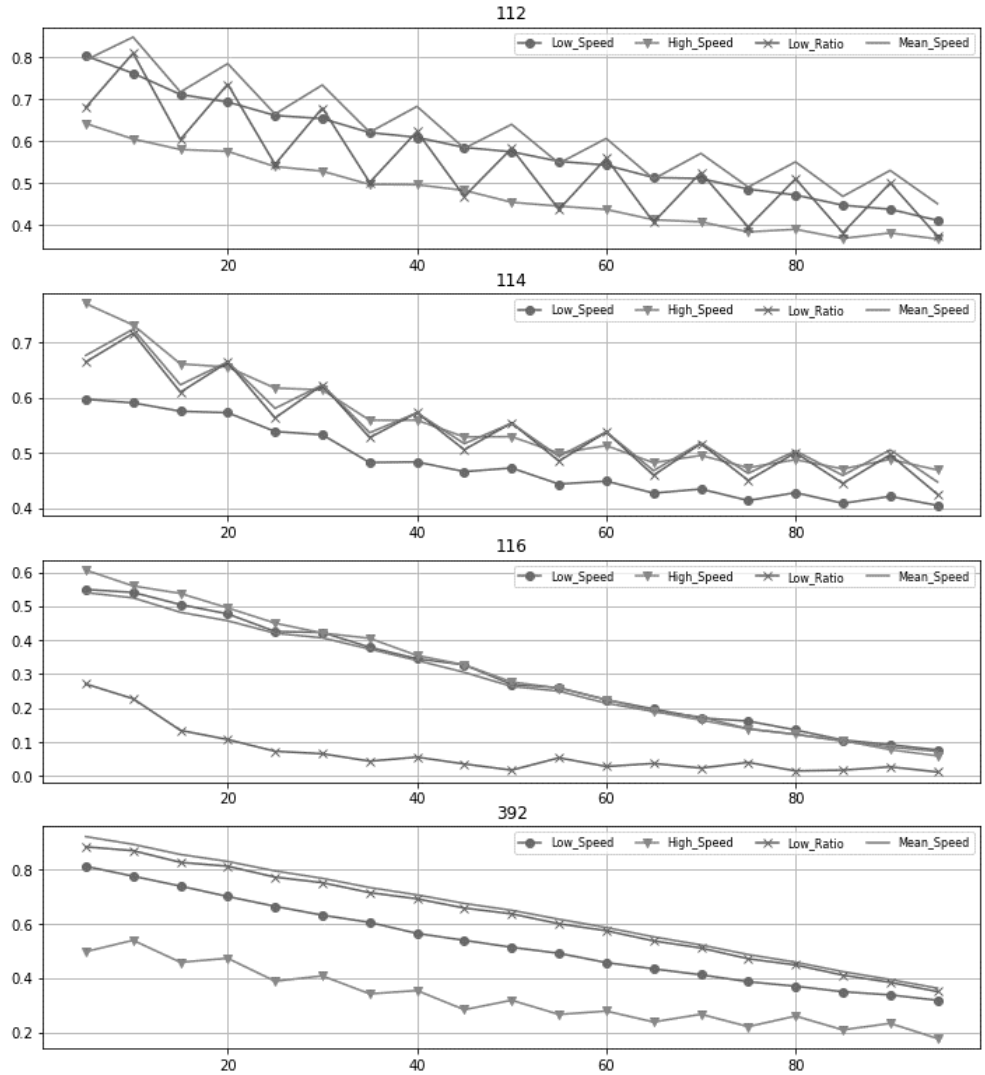


Figure B. 10 Periodicity Analysis for Filtered Model (Link 112, 114, 116, 392)

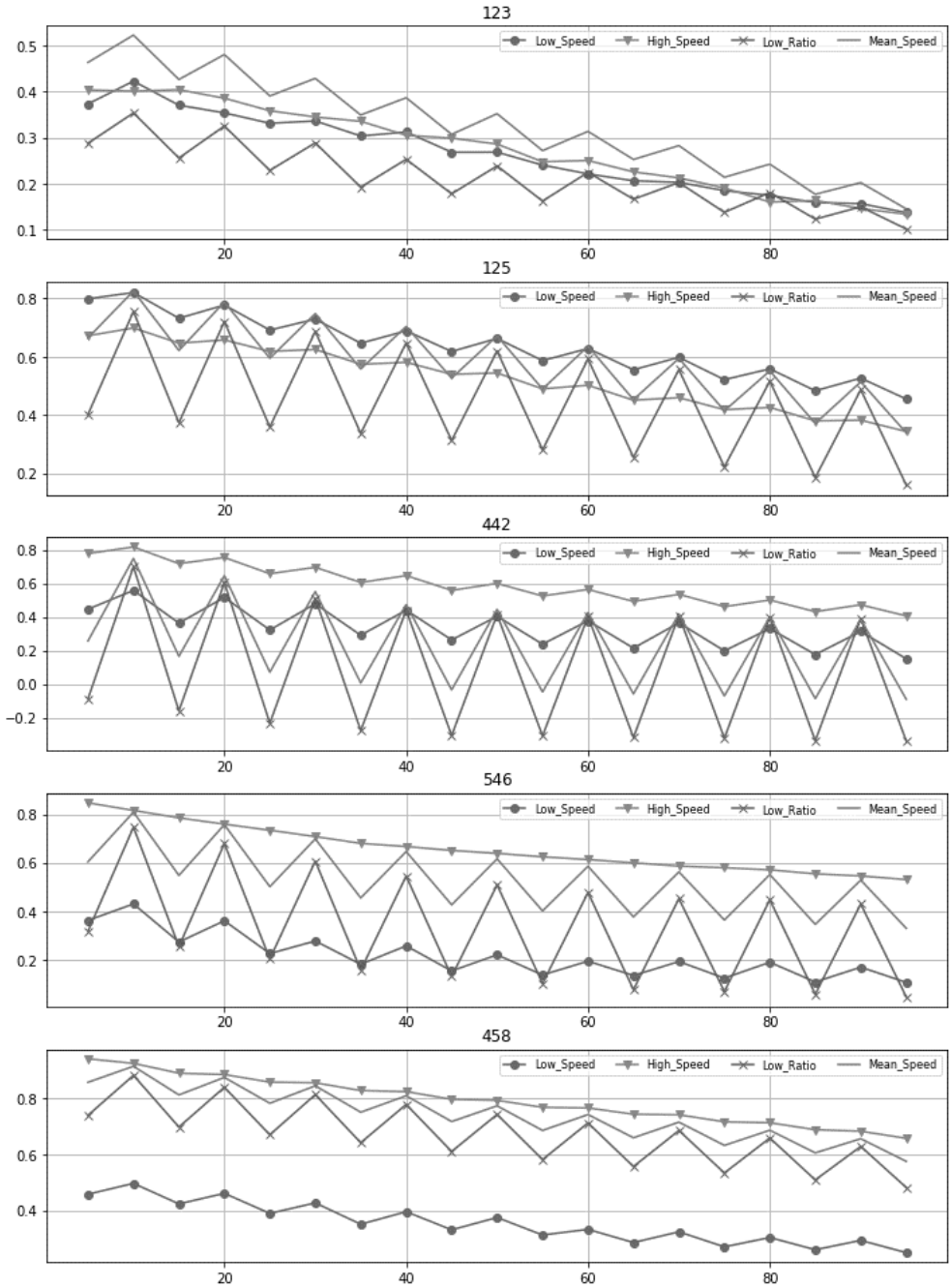


Figure B. 11 Periodicity Analysis for Filtered Model (Link 123, 125, 442, 546, 458)

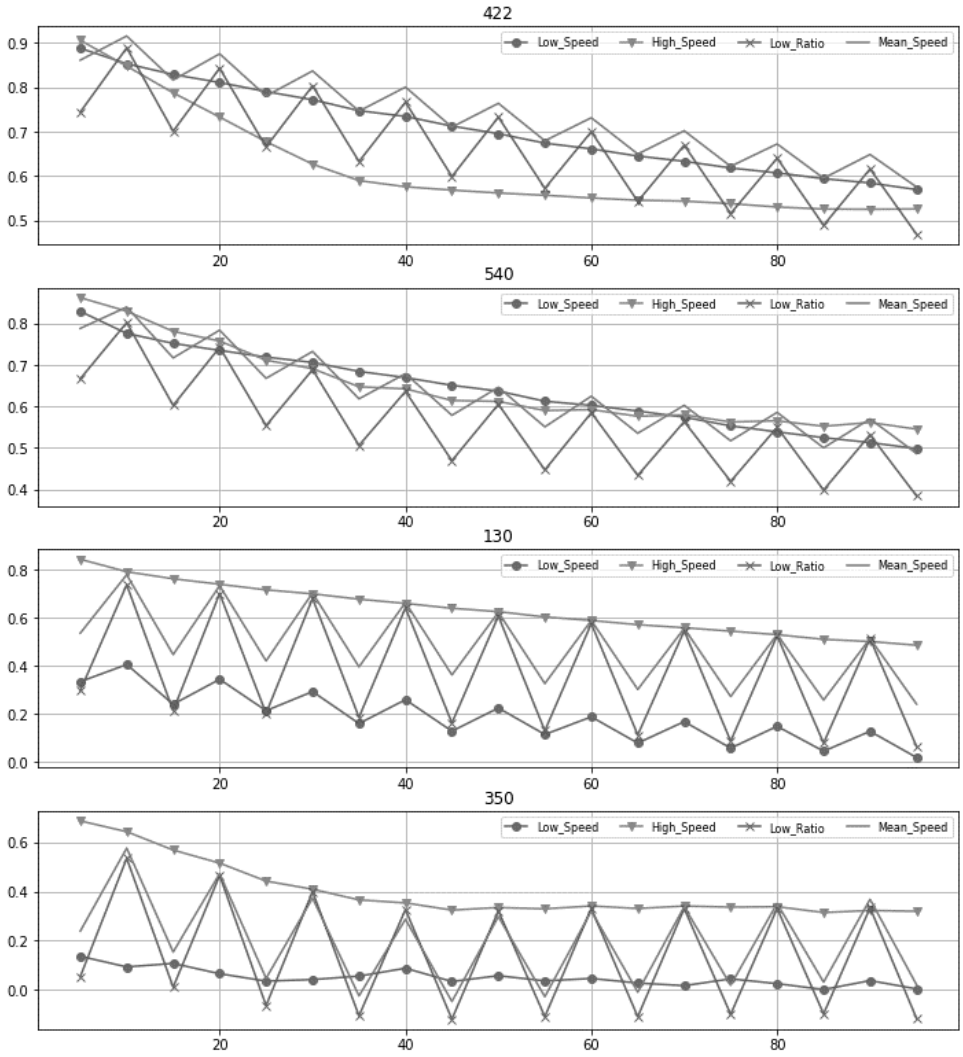


Figure B. 12 Periodicity Analysis for Filtered Model (Link 422, 540, 130, 350)

Appendix C. Platooning Features Diagram

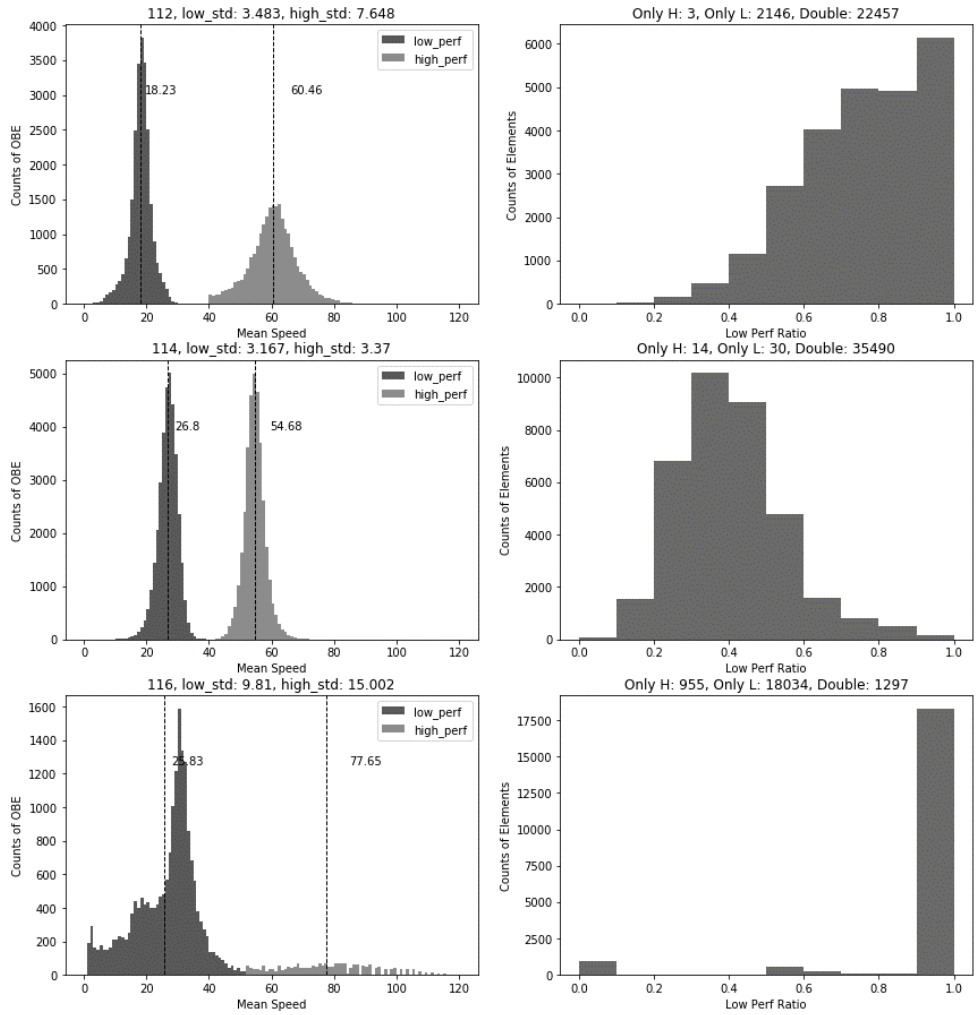


Figure C. 1 Histogram for LPP Speed, HPP Speed, LPR (Link 112, 114, 116)

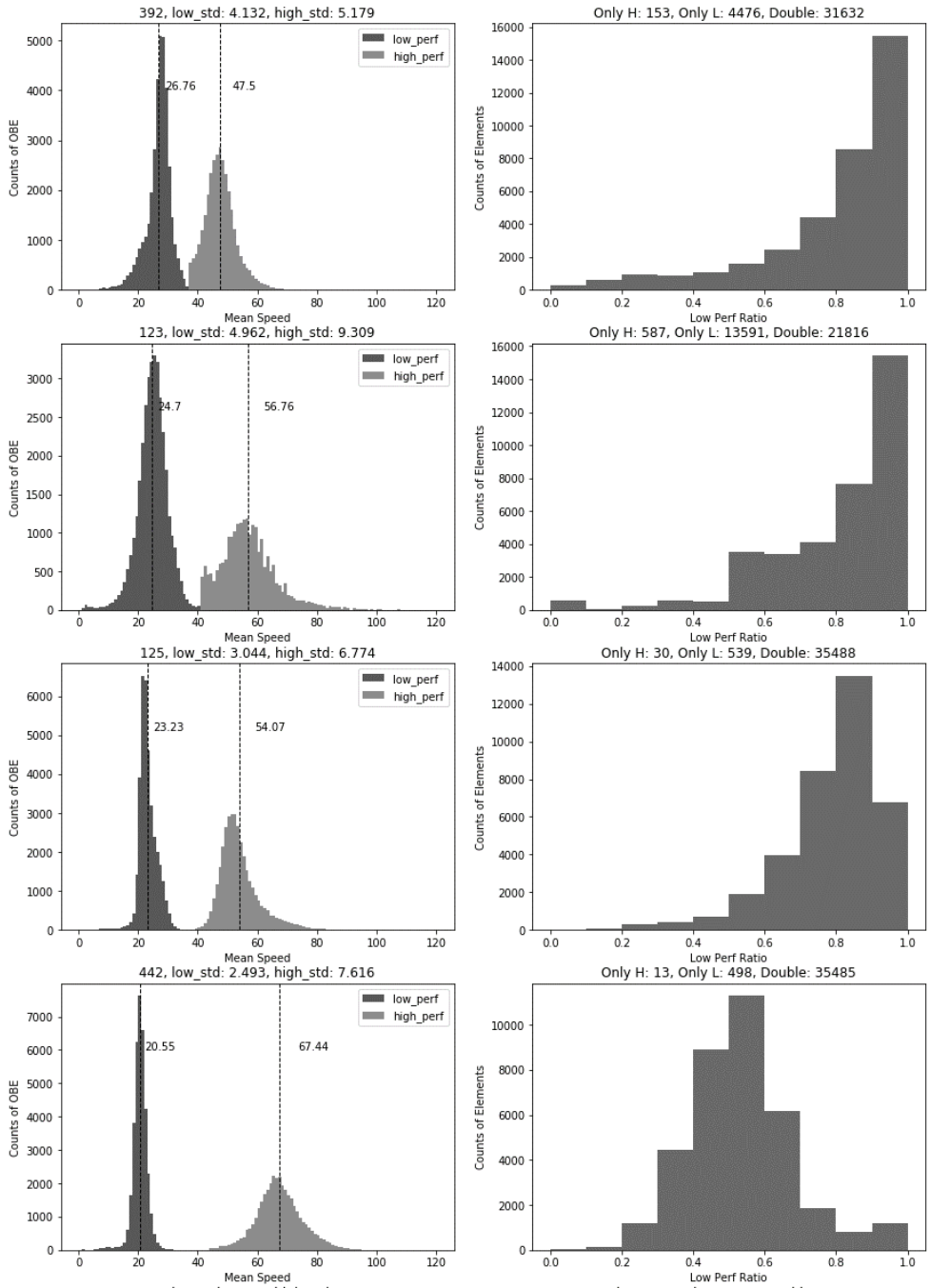


Figure C. 2 Histogram for LPP Speed, HPP Speed, LPR (Link 392, 123, 125, 442)

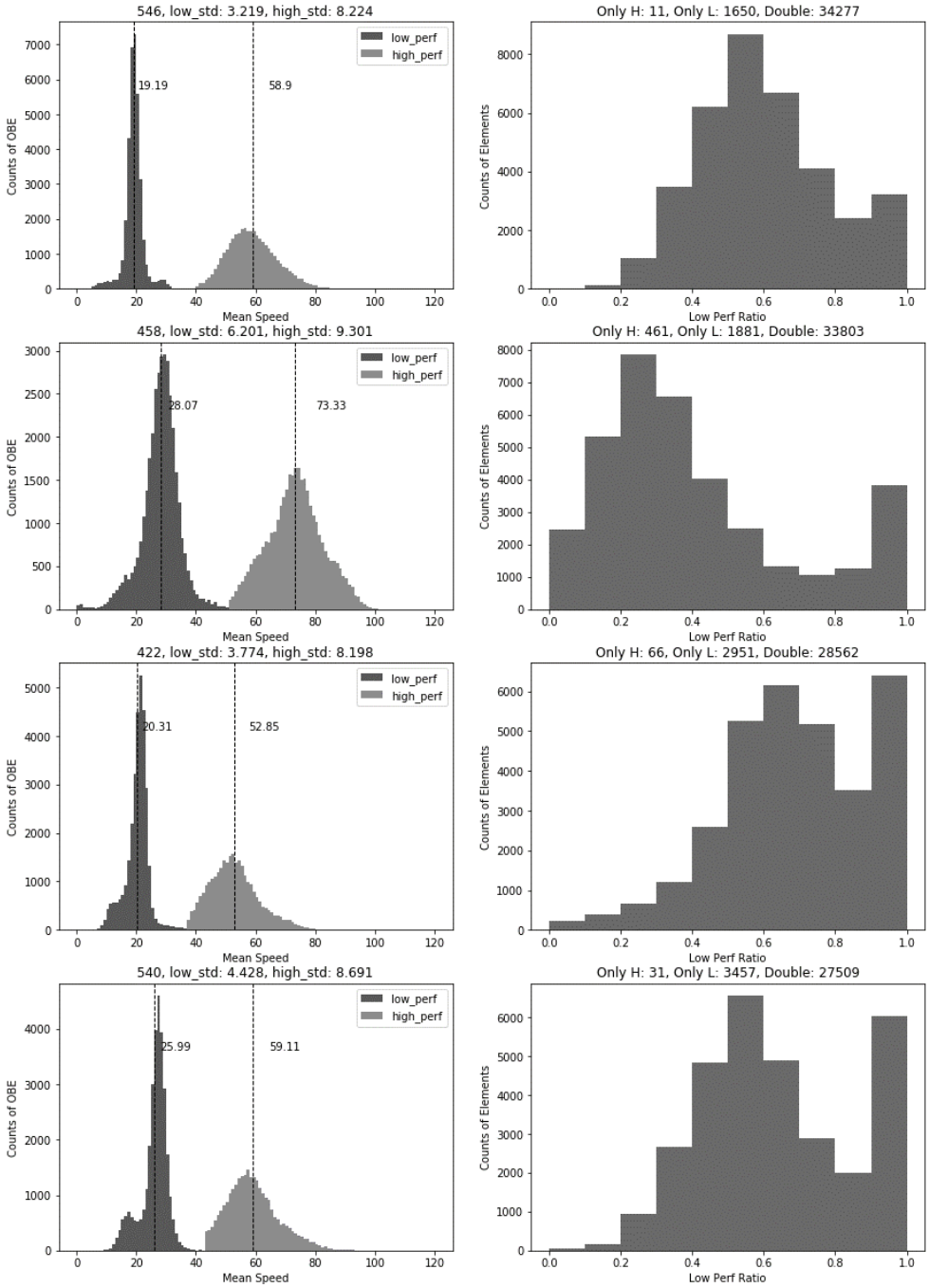


Figure C. 3 Histogram for LPP Speed, HPP Speed, LPR (Link 546, 458, 422, 540)

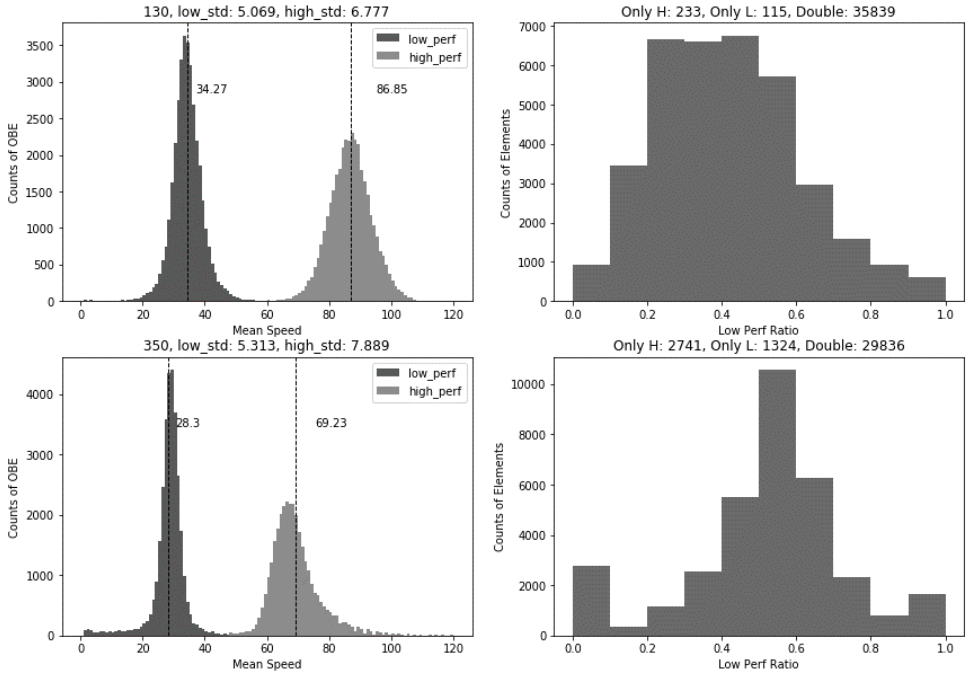


Figure C. 4 Histogram for LPP Speed, HPP Speed, LPR (Link 130, 350)

Bibliography

1. Akcelik, R. (1996). “Relating flow, density, speed and travel time models for uninterrupted and interrupted traffic.” *Traffic Engineering+ Control*, 37(9), 511–516.
2. Che, Z., Purushotham, S., Cho, K., Sontag, D., and Liu, Y. (2018). “Recurrent Neural Networks for Multivariate Time Series with Missing Values.” *Scientific Reports*, Springer US, 8(1), 1–12.
3. Chen, H., and Grant-Muller, S. (2001). “Use of sequential learning for short-term traffic flow forecasting.” *Transportation Research Part C: Emerging Technologies*, Pergamon, 9(5), 319–336.
4. Chien, S. I. J., Liu, X., and Ozbay, K. (2003). “Predicting Travel Times for the South Jersey Real-Time Motorist Information System.” *Transportation Research Record: Journal of the Transportation Research Board*, SAGE PublicationsSage CA: Los Angeles, CA, 1855(1), 32–40.
5. Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation.” *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, abs/1406.1, 1724–1734.
6. Cortes, C., and Vapnik, V. (1995). “Support-vector networks.” *Machine*

Learning, Kluwer Academic Publishers, 20(3), 273–297.

7. Daganzo, C. F. (1994). "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory." *Transportation Research Part B: Methodological*, Pergamon, 28(4), 269–287.
8. de Fabritiis, C., Ragona, R., and Valenti, G. (2008). "Traffic Estimation And Prediction Based On Real Time Floating Car Data." *2008 11th International IEEE Conference on Intelligent Transportation Systems*, IEEE, 197–203.
9. Fu, R., Zhang, Z., and Li, L. (2016). "Using LSTM and GRU neural network methods for traffic flow prediction." *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, IEEE, 324–328.
10. Fusco, G., Colombaroni, C., Comelli, L., and Isaenko, N. (2015). "Short-term traffic predictions on large urban traffic networks: Applications of network-based machine learning models and dynamic traffic assignment models." *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, IEEE, 93–101.
11. Gartner, N., Messer, C. J., and Rathi, A. K. (1992). *Revised Monograph on Traffic Flow Theory*. *US Department of Transportation Federal Highway Administration*.
12. Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). "A Fast Learning Algorithm for Deep Belief Nets." *Neural Computation*, MIT Press, 18(7), 1527–1554.
13. Hochreiter, S. (1998). "The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions." *International Journal of*

Uncertainty, Fuzziness and Knowledge-Based Systems, World Scientific Publishing Company , 06(02), 107–116.

14. Hochreiter, S., and Schmidhuber, J. (1997). "Long Short-Term Memory." *Neural Computation*, MIT Press 238 Main St., Suite 500, Cambridge, MA 02142-1046 USA journals-info@mit.edu , 9(8), 1735–1780.
15. Iqbal, M. S., Choudhury, C. F., Wang, P., and González, M. C. (2014). "Development of origin–destination matrices using mobile phone call data." *Transportation Research Part C: Emerging Technologies*, Pergamon, 40, 63–74.
16. Ishak, S., Kotha, P., and Alecsandru, C. (2003). "Optimization of Dynamic Neural Network Performance for Short-Term Traffic Prediction." *Transportation Research Record: Journal of the Transportation Research Board*, SAGE PublicationsSage CA: Los Angeles, CA, 1836(1), 45–56.
17. Lingras, P., and Mountford, P. (2001). "Time Delay Neural Networks Designed Using Genetic Algorithms for Short Term Inter-City Traffic Forecasting." Springer, Berlin, Heidelberg, 290–299.
18. Van Lint, J. W. C. (2008). "Online learning solutions for freeway travel time prediction." *IEEE Transactions on Intelligent Transportation Systems*, 38–47.
19. Min, W., and Wynter, L. (2011). "Real-time road traffic prediction with spatio-temporal correlations." *Transportation Research Part C: Emerging Technologies*, Pergamon, 19(4), 606–616.
20. Nagel, K., and Schreckenberg, M. (1992). "A cellular automaton model for freeway traffic." *Journal de Physique I*, EDP Sciences, 2(12), 2221–2229.

21. Newell, G. F. (1993). “A simplified theory of kinematic waves in highway traffic, part I: general theory.” *Transportation Research Part B*, 27(4), 281–287.
22. Nguyen, H., Kieu, L.-M., Wen, T., and Cai, C. (2018). “Deep learning methods in transportation domain: a review.” *IET Intelligent Transport Systems*, 12(9), 998–1004.
23. Park, B., Messer, C. J., and Urbanik, T. (1998). “Short-Term Freeway Traffic Volume Forecasting Using Radial Basis Function Neural Network.” *Transportation Research Record: Journal of the Transportation Research Board*, SAGE PublicationsSage CA: Los Angeles, CA, 1651(1), 39–47.
24. Park, D., and Rilett, L. R. (1998). “Forecasting Multiple-Period Freeway Link Travel Times Using Modular Neural Networks.” *Transportation Research Record: Journal of the Transportation Research Board*, SAGE PublicationsSage CA: Los Angeles, CA, 1617(1), 163–170.
25. Park, D., Rilett, L. R., and Han, G. (1999). “Spectral Basis Neural Networks for Real-Time Travel Time Forecasting.” *Journal of Transportation Engineering*, 125(6), 515–523.
26. Park, H. (2017). “Probabilistic prediction of traffic states using bayesian network.” Seoul National University, Seoul.
27. Park,H. , Kang, S., Kho, S., and Kim, D. (2019). *Effect of Inherent Variation and Spatiotemporal Dependency in Predicting Travel Speed in Urban Networks*. Washington DC, United States.
28. Ravanelli, M., Brakel, P., Omologo, M., and Bengio, Y. (2017). “Improving

speech recognition by revising gated recurrent units."

29. Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "'Why should i trust you?' Explaining the predictions of any classifier." *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 13-17-Augu, 1135–1144.
30. Sun, S., Zhang, C., and Yu, G. (2006). "A Bayesian network approach to traffic flow forecasting." *IEEE Transactions on Intelligent Transportation Systems*, 7(1), 124–133.
31. Vlahogianni, E. I., Karlaftis, M. G., and Golias, J. C. (2014). "Short-term traffic forecasting: Where we are and where we're going." *Transportation Research Part C: Emerging Technologies*, Elsevier Ltd, 43(February 2018), 3–19.
32. Wang, Y., Papageorgiou, M., and Messmer, A. (2006). "RENAISSANCE – A unified macroscopic model-based approach to real-time freeway network traffic surveillance." *Transportation Research Part C: Emerging Technologies*, Pergamon, 14(3), 190–212.
33. Wen, T.-H., Chin, W.-C.-B., and Lai, P.-C. (2017). "Link Structure Analysis of Urban Street Networks for Delineating Traffic Impact Areas." 203–220.
34. Wöllmer, M., Marchi, E., Squartini, S., and Schuller, B. (2011). "Multi-stream LSTM-HMM decoding and histogram equalization for noise robust keyword spotting." *Cognitive Neurodynamics*, 5(3), 253–264.
35. Wu, C.-H., Ho, J.-M., and Lee, D. T. (2004). "Travel-Time Prediction With Support Vector Regression." *IEEE Transactions on Intelligent Transportation*

Systems, 5(4), 276–281.

36. Wu, Y. H., and Hung, M. C. (2010). "Non-connective linear cartograms for mapping traffic conditions." *Cartographic Perspectives*, NACIS (North American Cartographic Information Society), (65), 33–50.
37. Yang, F., Yun, M.-P., and Yang, X.-G. (2014). "Travel Time Distribution under Interrupted Flow and Application to Travel Time Reliability." *Transportation Research Record: Journal of the Transportation Research Board*, 2466(1), 114–124.
38. Yu, B., Lam, W. H. K., and Tam, M. L. (2011). "Bus arrival time prediction at bus stop with multiple routes." *Transportation Research Part C: Emerging Technologies*, Pergamon, 19(6), 1157–1170.
39. Zhang, D., and Kabuka, M. R. (2018). "Combining weather condition data to predict traffic flow: a GRU-based deep learning approach." *IET Intelligent Transport Systems*, 12(7), 578–585.
40. Zhang, Y., and Xie, Y. (2007). "Forecasting of Short-Term Freeway Volume with v-Support Vector Machines." *Transportation Research Record: Journal of the Transportation Research Board*, SAGE PublicationsSage CA: Los Angeles, CA, 2024(1), 92–99.

Abstract

도시교통류는 복잡성을 내재하고 있다. 이 복잡성으로 인해, 일반적으로 지역간 간선 도로 네트워크의 속도를 추정하던 모형들을 사용할 경우 여러가지 한계점이 발생하게 된다. 본 연구는 도시교통류 상의 링크에서 프로브 차량 방식으로 수집된 속도자료의 특성을 분석하고, 기존 모형의 한계점을 제시하고, 이러한 한계점에 대한 해법으로서 변형된 순환형 신경망 모형을 개발하였다. 모형 개발에 있어, 기존 모형의 한계점을 보완하기 위해, 본 연구에서는 도시교통류의 단속류적 특징에 주목하였다. 자료 분석을 통해, 본 연구에서는 단속류에서 나타나는 현상으로서 차량군의 분리와 높은 빈도의 전이상태 발생을 확인하였다. 해당 현상들을 이용하여, 본 연구에서는 각 차량군의 특징을 이용한 이용한 2단계 모형과, 교통 상태를 분리하여 적용하는 선택적 드롭아웃 방식을 제시하였다. 추가적으로, 자료의 수집에 있어 빈발하는 결측 데이터를 효과적으로 다루기 위한 능동적 대체 방식을 개발하였다. 개발 모형은 평균적으로 높은 정확도를 보일 뿐 아니라, 기존 모형들의 한계점인 특정 상황에 대한 정확도를 제고하고 추정값과 추정 대상값의 상관관계를 높이며, 자료의 주기성을 적절하게 학습할 수 있었다.

주요어 : Data Estimation, Deep learning, Recurrent Neural Network,

Probe Vehicle, Data Correlation

학 번 : 2016-30289