

# 시공간적 영향력을 반영한 딥러닝 기반의 통행속도 예측 모형 개발

## Development of Traffic Speed Prediction Model Reflecting Spatio-temporal Impact based on Deep Neural Network

김영찬\* · 김준원\*\* · 한여희\*\*\* · 김종준\*\*\*\* · 황제웅\*\*\*\*\*

\* 주저자 : 서울시립대학교 교통공학과 교수

\*\* 교신저자 : 서울시립대학교 교통공학과 박사과정

\*\*\* 공저자 : 서울시립대학교 교통공학과 연구교수

\*\*\*\* 공저자 : 서울시립대학교 교통공학과 학부과정

\*\*\*\*\* 공저자 : 서울시립대학교 교통공학과 학부과정

Youngchan Kim\* · Junwon Kim\* · Yohee Han\* · Jongjun Kim\*  
· Jewoong Hwang\*

\* Dept. of Transportation Eng, Univ. of Seoul

† Corresponding author : Junwon Kim, mirageno21@uos.ac.kr

Vol.19 No.1(2020)

February, 2020

pp.1~16

pISSN 1738-0774

eISSN 2384-1729

[https://doi.org/10.12815/kits.](https://doi.org/10.12815/kits.2020.19.1.1)

2020.19.1.1

Received 8 November 2019

Revised 1 December 2019

Accepted 9 January 2020

© 2020. The Korea Institute of  
Intelligent Transport Systems. All  
rights reserved.

### 요약

4차 산업혁명 시대가 도래함에 따라 빅데이터를 활용하는 딥러닝에 대한 관심이 높아졌으며 다양한 분야에서 딥러닝을 이용한 연구가 활발하게 진행되고 있다. 교통 분야에서도 교통 빅데이터를 많이 활용하는 만큼 딥러닝을 연구에 이용한다면 많은 이점이 있을 것이다. 본 연구에서는 통행속도를 예측하기 위하여 딥러닝 기법인 LSTM을 이용한 단기 통행속도 예측 모형을 구축하였다. 예측에 활용한 데이터인 통행속도 데이터가 시계열 데이터인 것을 고려하여 시계열 예측에 적합한 LSTM 모델을 선택하였다. 통행속도를 보다 정확하게 예측하기 위하여 시간적, 공간적 영향을 모두 반영하는 모형을 구축하였으며, 모형은 1시간 이후를 예측하는 단기 예측모형이다. 분석데이터는 서울시 교통정보센터에서 수집한 5분 단위 통행속도를 활용하였고 분석구간은 교통이 혼잡한 강남대로 일부구간으로 선정하여 연구를 수행하였다.

핵심어 : 딥러닝, LSTM, 통행속도 예측, 빅데이터, 4차 산업혁명

### ABSTRACT

With the advent of the fourth industrial revolution era, there has been a growing interest in deep learning using big data, and studies using deep learning have been actively conducted in various fields. In the transportation sector, there are many advantages to using deep learning in research as much as using deep traffic big data. In this study, a short-term travel speed prediction model using LSTM, a deep learning technique, was constructed to predict the travel speed. The LSTM model suitable for time series prediction was selected considering that the travel speed data, which is used for prediction, is time series data. In order to predict the travel speed more precisely, we constructed a model that reflects both temporal and spatial effects. The model is a short-term prediction model that predicts after one hour. For the analysis data, the 5minute travel speed collected from the Seoul Transportation Information Center was used, and the analysis section was selected as a part of Gangnam where traffic was congested.

Key words : Deep learning, LSTM, Travel speed prediction, Big data, The fourth industrial revolution

## I. 서론

### 1. 개요

4차 산업혁명은 전 세계에서 등장과 동시에 매우 중요한 이슈가 되었으며, 이에 따라 사회가 빠르게 변화하고 있다. 4차 산업혁명으로 인공지능, 빅데이터, 사물인터넷 등 첨단 정보통신기술이 경제, 사회 전반에 융합되어 혁신적인 변화가 나타났으며, 그 중심에는 인공지능 기술과 빅데이터가 있다. 최근 몇 년간 빅데이터 기반의 딥러닝에 대한 연구들이 활발하게 이루어지고 있다.

교통분야에서도 4차 산업혁명으로 인해 환경이 급격하게 변화하고 있는데, 대표적인 분야가 교통정보 수집 분야이다. 기존의 교통정보 수집은 차량에서 도로 인프라로 일방적인 방향으로 이루어 졌으며, 지점정보 위주의 교통정보 수집이 이루어 졌다. 그러나 현재 차세대 지능형교통체계(Cooperative Intelligent Transportation System; C-ITS)의 등장으로 차량과 도로인프라가 정보통신기술과의 융합해 양방향으로 정보를 주고받고 있으며, 다양한 실시간 정보가 수집되어 다양한 형태의 교통 빅데이터가 생성되고 있다.

교통분야에서는 교통시스템의 효율을 향상시키기 위해 1990년대부터 인공지능을 적용한 연구를 진행해왔다. 최근에는 많은 연구에서 인공지능의 핵심 분야인 딥러닝을 적용하고 있으며, 특히 교통정보 예측 분야에서 효과가 나타나고 있다. 전통적인 교통정보는 지점검지기, CCTV, 택시와 같은 일부 차량의 GPS 자료를 통해 도로 소통정보를 추정해왔으나, 최근에는 딥러닝 기법을 교통정보 가공에 접목하고 있으며, 딥러닝 기법의 적용으로 교통정보 예측에서 다양한 교통 빅데이터를 융합해 사용함으로써 교통정보 예측의 정확도를 향상시키고 있다.

현재 다양한 교통빅데이터가 다양한 매체를 통해 수집되고 있으며, 향후 자율주행기술을 발전으로 보다 많은 교통빅데이터가 수집될 것으로 판단된다. 이에 교통정보 예측에 딥러닝 기법을 적용하여 교통빅데이터를 학습해 교통소통정보 예측의 정확도를 향상시키기 위한 다양한 연구가 진행되고 있다. 그러나 교통정보의 시계열적 특성만을 고려해 상하류부의 소통상황을 고려하지 않고 교통소통정보를 예측하여 예측정확도 향상에 한계가 있다 판단된다. 본 연구에서는 이와 같은 시공간적 영향력을 반영한 딥러닝 기반의 단기 통행속도 예측 모형을 개발하고자 한다.

### 2. 연구 방법

본 연구는 크게 4가지 단계로 구성해 진행하였다. 첫 번째 기존 딥러닝 이론 및 교통류 관련 이론 고찰, 두 번째 딥러닝 기반의 통행시간 예측 모형 개발, 세 번째 개발된 모형의 구현, 마지막 네 번째 개발된 모형의 평가 순으로 연구를 수행하였다.

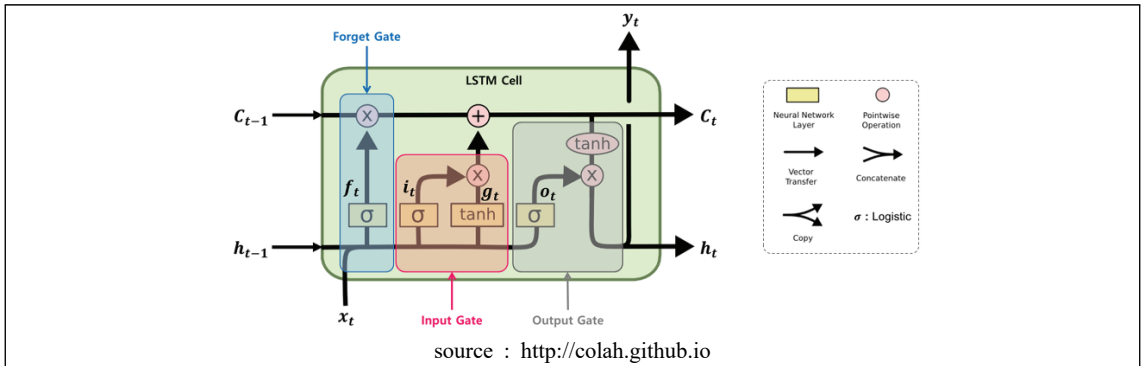
각 단계별 연구내용은 다음과 같다. 첫 번째 단계에서는 교통분야에서 딥러닝 이론을 접목한 연구사례를 분석하고, 교통류 분야에서 교통 소통상태에 대한 예측 및 분석에 대한 방법론을 검토하였다. 두 번째 단계에서는 도시부 특성을 반영해 시공간 및 통행시간 패턴을 고려한 딥러닝 기반의 예측모형을 개발하였으며, 세 번째 단계에서 개발된 모형을 효율적으로 학습할 수 있도록 모형을 구현하였다. 마지막 네 번째 단계에서는 서울시 교통정보센터(TOPIS)에서 수집·가공한 2017년 5분단위 구간통행속도 정보를 이용해 서울시 강남대로 “강남역 ~ 교보타워사거리” 구간을 대상으로 개발된 모형을 학습시켜 관측통행시간 정보와 예측통행시간 정보 간 절대평균오차율(MAPE)를 산출해 모형을 검증하였다.

## II. 이론적 고찰

### 1. LSTM(Long Short-Term Memory) 알고리즘

딥러닝 알고리즘은 크게 CNN와 RNN로 구분할 수 있다. 이중 RNN은 음성 또는 문자 등 순차적으로 등장하는 데이터 처리에 적합한 모델이다. 그러나 RNN의 경우 관련 정보와 그 정보를 사용하는 퍼셉트론 사이 거리가 멀 경우 역전파(backpropagation) 시 기울기 사라짐 문제(vanishing gradient problem)가 발생하여 학습 능력이 크게 저하된다. 때문에 시계열 자료를 분석하는데 한계가 있는데, 이 문제를 극복하기 위해 고안된 모형이 LSTM이다.

LSTM에서 셀(cell)의 상태(state)는 두 개의 벡터  $h_t$ ,  $c_t$ 로 나누어져 있다.  $h_t$ 는 단기상태(short-term state),  $c_t$ 는 장기상태(long-term state)를 의미한다. 이 중 LSTM의 핵심은 네트워크가 장기 상태( $c_t$ )에서 기억할 부분, 삭제할 부분, 새로 읽을 부분을 학습하는 것이다. 장기 상태( $c_t$ )는 Forget gate( $f_t$ ), Input gate( $i_t$ ,  $g_t$ ), Output( $o_t$ )를 통해 학습이 수행된다.



<Fig. 1> Structure of LSTM(Long Short-Term Memory) cell

Forget gate( $f_t$ )는 과거의 정보를 잊기 위한 게이트이다. 이전 단기기억( $h_{t-1}$ )과 현재 입력정보( $x_t$ )를 받아 시그모이드(sigmoid) 함수를 통해 활성화해 현재의 장기 상태( $c_t$ )로 내보낸다. 이때 시그모이드 함수의 출력 범위는 0에서 1사이인데, 그 값이 0이라면 이전 상태의 정보를 잊고, 1이라면 이전 상태의 정보를 온전히 기억하게 된다. Forget gate( $f_t$ )은 다음 식(1)과 같다.

$$f_t = \sigma(W_{x,h,f}x_t + W_{h,h,f}h_{t-1} + b_{h,f}) \dots\dots\dots (1)$$

- 여기서,  $x_t$  =  $t$ 시간대 입력 값
- $h_{t-1}$  =  $t-1$  시간대 단기 상태
- $W_{x,h,f}$  = Forget gate에서 입력값에 대한 가중치
- $W_{h,h,f}$  = Forget gate에서 단기기억에 대한 가중치
- $b_{h,f}$  = Forget gate에서 편향치
- $\sigma(k)$  = sigmoid 함수
- $\sigma(k) = \frac{1}{1 + e^{-k}}$

Input gate( $i_t \odot \tilde{c}_t$ )는 현재의 정보를 기억하기 위한 게이트이다. 단기기역( $h_{t-1}$ )과 현재 입력정보( $x_t$ ) 시그모이드 함수를 통해 활성화한 값과 같은 값을 쌍곡탄젠트(tanh) 함수를 통해 활성화 한 값을 매트릭스 곱(Hadamard product) 연산해 Forget gate( $f_t$ )를 통과한 장기 상태( $c_t$ )에 더한다.  $i_t$ 의 범위는 0~1이고  $g_t$ 의 범위는 -1~1로, 각각 정보의 강도와 방향을 의미한다. Input gate( $i_t, \tilde{C}_t$ )는 다음 식(2)과 같다.

$$i_t = \sigma(W_{xhi}x_t + W_{hhi}h_{t-1} + b_{hi}) \dots\dots\dots (2)$$

$$g_t = \tanh(W_{xhg}x_t + W_{hhg}h_{t-1} + b_{hg})$$

여기서,  $W_{xhi}, W_{xhg}$  = Input gate에서 입력값에 대한 가중치  
 $W_{hhi}, W_{hhg}$  = Input gate에서 단기기역에 대한 가중치  
 $b_{hi}, b_{hg}$  = Input gate에서 편향치  
 $\tanh(k)$  = 쌍곡탄젠트 함수

$$(\tanh(k) = 2\sigma(2k) - 1 = \frac{e^k - e^{-k}}{e^k + e^{-k}})$$

Forget gate( $f_t$ )와 Input gate( $i_t \odot g_t$ )에서 산출된 값을 통해  $t$ 시간대 장기 상태( $c_t$ )를 업데이트 하며 다음 식(3)와 같다.

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \dots\dots\dots (3)$$

Output gate( $o_t$ )는  $t$ 시간대 출력값( $y_t$ )과 단기 상태( $h_t$ )을 결정하는 단계이며, 출력값과 단기 상태는 동일한 값을 가진다. Output gate( $o_t$ )는 단기기역( $h_{t-1}$ )과 현재 입력정보( $x_t$ ) 시그모이드 함수를 통해 활성화한 값을 출력하며, 출력된 값과 앞 단계에서 갱신된  $t$ 시간대 장기 상태( $c_t$ )를 쌍곡탄젠트 함수를 통해 다시 활성화한 값을 매트릭스 곱 연산해 출력값( $y_t, h_t$ )을 결정한다. Output gate( $o_t$ ) 및 출력값( $y_t, h_t$ )은 다음 식(5)와 같다.

$$o_t = \sigma(W_{xho}x_t + W_{hho}h_{t-1} + b_{ho}) \dots\dots\dots (4)$$

$$y_t, h_t = o_t \odot \tanh(c_t)$$

여기서,  $W_{xho}$  = Output gate에서 입력값에 대한 가중치  
 $W_{hho}$  = Output gate에서 단기기역에 대한 가중치  
 $b_{ho}$  = Output gate에서 편향치

## 2. 딥러닝 기반의 소통수준 예측 관련 연구

2010년대 들어 AI(Artificial Intelligence) 기술의 급격한 발달로 다양향 분야에서 AI 기술을 접목하여 이미 인간의 수준을 뛰어넘은 결과를 도출하고 있다. 교통분야에서도 AI기술을 접목해 활용하고자 하는 많은 연구가 활발하게 이루어지고 있다. 특히 교통 빅데이터 기반의 미래 교통 소통 상황 예측에 대한 연구가 활발하게 이루어지고 있다.

Ma et al.(2015)는 시계열 학습에 장점이 있는 LSTM(Long Short-Term Memory)알고리즘을 이용해 단기통행

속도 예측 모형을 제시하였다. 모형의 효과분석을 검증을 위해 세 가지의 RNN(Recurrent Neural Network), 서포트 벡터 회귀모형(support vector regression), Arima 모형, Kalman-filter 모형과 통행속도 예측성능을 비교하였다. LSTM 모형은 중국 베이징에서 한달동안 2분 간격으로 검지기에서 수집된 통행속도 자료를 이용해 수행하였으며, 첫 25일 통행속도 자료는 학습에 다음 5일 자료는 모형 평가에 사용하였다. 분석 결과 LSTM 모형이 나머지 모형보다 좋은 성능을 나타냈으며, 실측시간과 표출시간 간 시간편차(time-lag)를 정확히 설정하였을 때 보다 좋은 성능을 나타냈다.

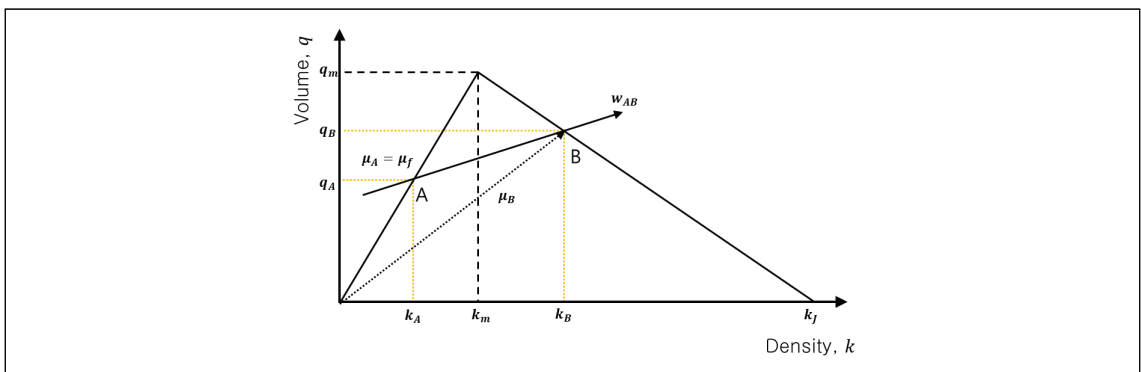
Choi(2018)는 청주시 DSRC 로그 데이터를 활용하여 딥러닝을 이용한 청주시 통행속도를 예측하는 연구를 수행하였다. 국내에서 구축된 ITS 시스템을 통해 수집된 데이터를 순환신경망기법을 적용하여 도로구간에 대한 통행속도를 예측하였다는 점에서 의미가 있다. 또한 순환신경망 기법 중 하나인 LSTM을 적용하여 통행속도를 예측할 수 있다는 가능성을 제시하였고 과거패턴과 기상데이터를 이용하여 통행속도를 예측하였다.

Jeon(2018)은 시공간 상관을 고려한 단기 통행속도 예측 모형을 개발하였다. 이 연구에서는 강남지역 244개의 도로 구간에 대한 통행속도를 활용해 도시 전체적인 공간적 상관관계를 고려한 딥러닝 모형을 적용하는 연구를 수행하였다. 도로의 속도를 이미지로 표현하여 도로의 공간구조를 데이터를 입력시켜줌으로써 보다 직관적인 방법을 통한 연구이다.

Wang and Li(2018)은 딥러닝 기법인 CNN과 LSTM 모델의 특징 모두를 활용하기 위하여 두 모델을 결합한 구조로 통행속도를 예측하였다. 기존에 있던 딥러닝 기법을 다양하게 결합하여 만든 새로운 모델들로 통행속도 예측을 시도하였다.

### 3. 교통류 이론

Newell(1993)은 교통량-밀도 관계를 기존 비선형적인 포물선 관계 대신 임계밀도( $k_m$ )을 중심으로 비혼잡상태와 혼잡상태의 2개 영역으로 구분해 삼각형 모양의 관계로 간편화 했다. 임계밀도를 기준으로 임계밀도 미만일 때 자유속도( $w_1 = \mu_f$ )를 갖고, 임계밀도 이상일 때  $w_2$ 의 속도를 갖는다는 개념으로 <Fig.2>와 같다.



<Fig. 2> Newell's simplified method

Daganzo(1994)는 Cell Transmission Model을 제시하였는데, 도로구간을 일정간격의 셀(Cell)로 구분하고, 단위시간에 따라 셀의 교통량과 밀도를 구해 연속류 도로의 교통류 상태를 나타내는 모형이다. 이 때 교통량과

밀도관계는 Newell의 모형을 준용해 선형관계로 고려하였으며 식(1)과 같다. Cell Transmission Model 모형은 교통류를 세밀하게 분석하기는 어렵지만 동일한 특성을 갖는 구간을 하나의 셀로 묶어, 셀별 교통류 특성(용량, 임계밀도, 혼잡밀도)에 따라 각 시간대별로 교통류를 전이시키기 때문에 현실적으로 교통류를 표현할 수 있다.

$$q = \min\{vk, q_{\max}, w(k_j - k)\}, \text{ for } 0 \leq k \leq k_j \dots\dots\dots (5)$$

where,  $w \leq v$  and  $q_{\max} \leq [1/v + 1/w]$

- 여기서,  $q$  = 교통량  
 $v$  = 자유통행속도  
 $k$  = 밀도  
 $q_{\max}$  = 도로 용량  
 $k_j$  = 혼잡밀도  
 $w$  = 혼잡발생시 후방충격과

### Ⅲ. 모형 구축

#### 1. 시공간적 영향력을 고려한 학습 모형 구축

본 연구는 주로 딥러닝 기법에 적용에 초점을 맞춘 기존 연구들과는 다르게 통행속도에 영향을 미치는 시간적인 영향력뿐만 아니라 충격과 모형의 기본 개념을 반영해 공간적인 요소들의 영향력을 고려하는 통행속도 예측 모형을 개발하였다. 과거 통행속도 패턴과 전시간대 통행속도와 같은 시간적 영향력과 인접한 주변 도로 및 공간적으로 일정거리 떨어진 주요도로의 공간적 영향력을 고려한 딥러닝 통행속도 예측 모형을 구축하였다.

본 연구에서는 다음 3가지를 고려해 분석 모형을 구축하였다. 첫째로 각 도로는 통행속도는 일정한 통행속도패턴을 보인다. 각 도로에서 나타나는 통행속도는 임의의 값을 갖는 것이 아니라 요일별·시간대별 패턴을 나타낸다. 특히 도시부 도로의 경우 통근·통학 통행으로 인해 통행속도의 패턴이 보다 명확하게 나타난다. 사고나 행사 같은 돌발 상황에 따라 다른 통행속도 패턴이 나타날 수 있으나, 그 외에는 일정한 패턴을 가지고 있다. 따라서 본 연구에서는 이와 같이 도시부 도로에서 나타나는 요일별·시간대별 패턴을 고려하기 위해 과거 통행속도 패턴을 반영한 통행속도 예측 모형을 구축하였다.

둘째, 각 도로의 통행속도는 인접도로의 영향을 받는다. 충격과 모형에 따르면, 각 도로의 통행속도는 상하류 인접한 도로의 소통수준에 크게 영향을 받는다. 만약 상류 진입구간 소통상황이 악화된다면 이는 대상구간의 진입 교통량 증가로 이어져 해당 구간의 소통상황이 악화될 것이다. 또한 하류 진입구간의 소통상황이 악화된다면 대상구간의 진출 교통량의 감소로 이어져 해당구간의 소통악화로 이어질 것이다. 따라서 본 연구에서는 이와 같은 공간적인 영향력을 고려해 인접해 있는 상류부, 하류부 도로의 통행속도를 입력변수로 활용하여 통행속도 예측모형을 구축하였다.

셋째, 한 도로의 통행속도는 일정거리의 영향권 내의 도로 교통상황에 상황에 영향을 미친다. 즉 어떤 지역이 막히기 시작하면 이와 인접한 곳에 위치한 도로의 소통이 악화되면서 교통체증이 주변 영향권으로 퍼지게 된다. 본 연구에서는 이를 반영하기 위해 분석도로에서 영향권 내 주요도로의 통행속도를 입력 변수로 활용하였으며, 영향권은 5km로 설정하였다. 또한 영향권 내에 있는 주요 연속류 도로 진입부의 교통소통 상황

을 입력변수로 활용하였다.

본 연구에서는 이와 같은 3가지 특성을 고려해 딥러닝 학습모형을 개발하였다. 우선 시간적 영향력을 위해 최근의 패턴을 고려하기 위한 하루 동일시간대 전 통행시간과 요일 패턴을 고려하기 위한 일주일전 동일 요일 동일시간대 통행시간 정보를 입력변수로 설정하였다. 또한 공간적 영향력을 고려하기 위해, 분석대상구 간 상류 직진 진입구간과 하류 직진 진출구간, 영향권 내 주요 진입도로 직전 시간대 통행시간 정보를 학습 모형의 입력변수로 설정하였다.

$$S_i(t) = [x_i(t-1), x_i^{day}(t), x_i^{week}(t), x_{i-1}(t-1), x_{i+1}(t-1), x_{inbound}(t-1)]^T \dots\dots\dots (6)$$

- 여기서,  $S_i(t)$  = t시간대 딥러닝 기반 통행시간 예측 모형 입력정보
- $x_i(t-1)$  = 분석대상 구간 직전 시간대 통행시간
- $x_i^{day}(t)$  = 분석대상 구간 전날 동일시간대 통행시간  
(평일의 경우 직전 평일 동일시간대 통행시간)  
(주말의 경우 1주전 동일요일 동일시간대 통행시간)
- $x_i^{week}(t)$  = 분석대상 구간 전주 동일요일 동일시간대 통행시간  
(주말의 경우 2주전 동일요일 동일시간대 통행시간)
- $x_{i-1}(t-1)$  = 분석대상 구간 상류 직진 진입구간 직전 시간대 통행시간
- $x_{i+1}(t-1)$  = 분석대상 구간 하류 직진 진출구간 직전 시간대 통행시간
- $x_{inbound}(t-1)$  = 분석대상 구간 영향권내 주요 진입도로 직전 시간대 통행시간
- $T$  = LSMT 학습 Sequence(학습시간 범위)

## IV. 딥러닝 학습 모형 평가

### 1. 분석 대상 구간

본 연구에서 개발된 모형을 평가하기 위해 분석의 대상이 되는 구간은 서울시 강남대로 “강남역 ~ 교보타워사거리” 구간을 선정하였다. 강남대로는 서울시에서 가장 혼잡한 단속류 도로 중 하나로, 교통정보 수집 샘플을 충분히 확보할 수 있어 해당 도로의 속도데이터가 정확하다고 판단해 본 연구의 분석 대상 구간으로 설정하였다. 이 중 강남역부터 교보타워 사거리 양방향 구간의 약 800m 거리의 구간을 분석대상 구간으로 설정하였다. 또한 강남역부터 교보타워 사거리 구간에 인접해있는 상류 및 하류 구간도 입력데이터로 활용하였다. 또한 영향권 내 주요 도로 진입링크로 강남대로 주변 주간선도로인 현충로, 서빙고로, 한남대로, 동호로, 고산자로, 왕십리로, 동일로, 사당로, 남부순환로, 현릉로, 양재대로, 올림픽로의 주요 구간을 입력데이터로 활용하였다.

### 2. 분석대상 데이터

서울시에서는 1년 365일, 24시간동안 서울특별시 485개 도로를 대상으로 차량통행속도 데이터를 수집하고 있다. 도시고속도로에서는 영상검지기, 루프검지기 등을 활용하여 정보를 수집하고 있으며, 일반도로에서는 서울시 카드 택시 7만 여대의 위치정보를 활용해 5분 단위 구간의 속도를 수집하고 있다.

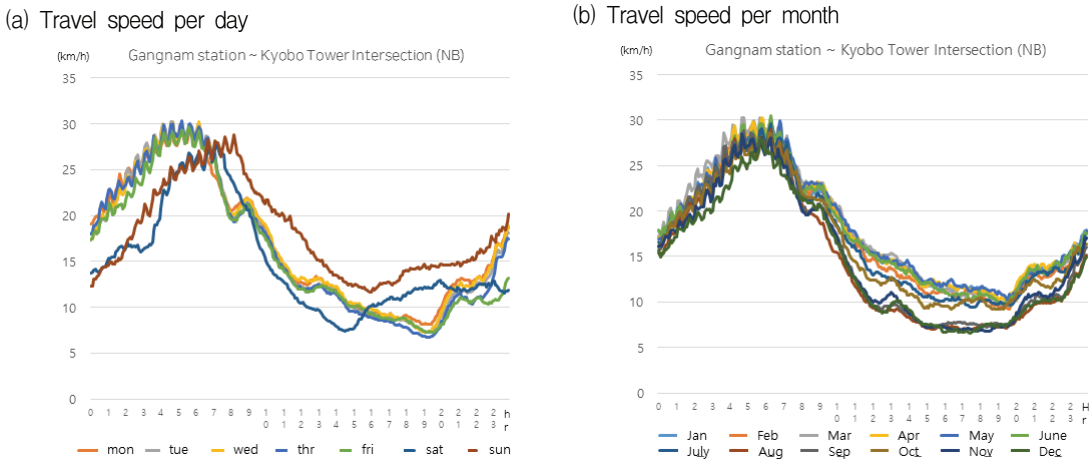
본 연구에서는 분석대상구간 링크에 대해 서울시 교통정보센터에서 수집한 2017년 1월부터 12월까지의 데이터를 가지고 활용하였다. 각 링크에 해당하는 5분 단위 통행속도 데이터를 사용했다. 이 중에서 제대로 수집이 되지 않거나 가공이 제대로 되지 않는 데이터는 중복 값 제거, 결측 값 보정을 통해 보완하였다. 또한 5분단위 통행속도 정보의 경우 변동이 심해 교통 패턴의 경향을 추출하기 어렵기 때문에 5분 단위 통행시간 정보를 지수평활화법을 이용해 산출하였으며 다음 식(7)과 같다.

$$\overline{TT}_i^{5min}(t) = \alpha \times TT_i^{5min}(t) + (1-\alpha) \times \overline{TT}_i^{5min}(t-1) \dots\dots\dots (7)$$

여기서,  $\overline{TT}_i^{5min}(t)$  = t시간대 i이동류 평활화 5분 통행시간  
 $TT_i^{5min}(t)$  = t시간대 i이동류 5분 통행시간  
 $\alpha$  = 평활화 계수

### 3. 입력데이터 구축

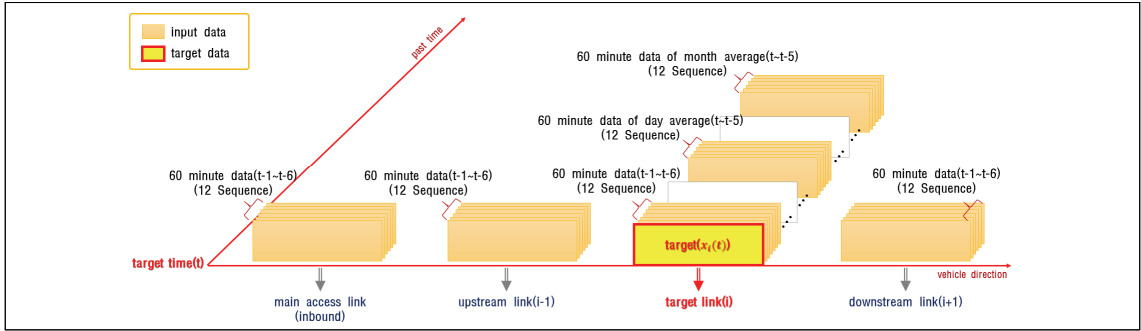
분석대상 구간의 통행속도는 일정한 통행속도패턴을 반영하기 위한 입력데이터 구축하기 위해 과거 통행속도가 어떻게 변화하는지 살펴보기 위해 요일별 시간대별 통행속도와 월별 시간대별 통행속도 그래프 분석하였다. <Fig. 5>의 (a)는 요일별 시간대별 통행속도 그래프이고, (b)는 월별 시간대별 통행속도 그래프이다. 이를 분석한 결과 과거 요일별, 월별 통행속도가 서로 상이한 것으로 나타났다. 이와 같은 특성을 반영하기 위해 요일별 시간대별 통행속도와 월별 시간대별 통행속도를 입력 변수로 구축하였다. 5분 단위의 기존 데이터에서 각 요일별, 월별 통행속도의 평균을 계산하여 앞서 말한 5분마다의 1시간의 데이터 셋을 만들 수 있도록 구성하였으며, 요일별 통행속도는 전체 1년간의 평균보다 분석시점 기준 5주전의 통행속도 데이터를 입력 변수로 활용하여 최근의 통행속도 패턴을 반영할 수 있도록 하였다.



<Fig. 3> Travel speed pattern

직전시간대 통행속도는 분석도로의 통행속도에 가장 큰 영향을 미치는 요소이기 때문에 요일/월 패턴, 상하류 구간, 영향권 내 주요 진입 구간의 통행속도 입력데이터는 학습목표시간 기준 이전 1시간 기준으로 구축하였다.





<Fig. 4> Travel time prediction model input DB configuration based on deep learning

분석대상 구간의 인접도로의 영향력을 반영하기 위한 입력데이터 구축을 위해 분석대상 구간인 “강남역 ~ 교보타워사거리” 양방향의 구간의 직진 상하류 구간인 “우성아파트 ~ 강남역” 구간과 “논현역 ~ 교보타워사거리” 구간과 좌회전 상하류 구간인 “테헤란로” 및 “사평대로” 구간의 통행속도 입력변수로 구축하였다.

분석대상구간 영향권에 있는 주요 진입 도로구간을 고려하기 위해 주요 진입도로 구간의 영향권을 5km로 설정하여 분석대상 구간에서 5km 떨어진 주요 진입구간을 진입도로로서 설정하였다. 분석대상구간인 강남대로에 많은 교통량이 진입할 것으로 예상되는 주간선도로 구간을 주요 진입도로로 설정하였다. 강남역에서 교보타워사거리로 가는 상행방향은 분석구간의 가운데 기준 아래쪽인 사당로, 남부순환로, 현릉로, 양재대로, 올림픽로를 주요도로로 설정하였다. 반대방향인 교보타워사거리에서 강남역으로 향하는 방면은 현릉로, 서빙고로, 한남대로, 동호로, 고산자로, 왕십리로, 동일로를 주요 진입도로로 설정하였다.

<Table 1> Variables of north bound

Data type	Variable
Past travel speed	monthly average speed, daily average speed, Speed before 1 hour
Adjacent road	Speed of upstream road, Speed of downstream road, Speed of cross road (Teheran-ro)
Nearby main roads	Speed of main roads (Sadang-ro, Nambusunhwan-ro, Heolleung-ro, Yangjae-daero, Olympic-ro)

<Table 2> Variables of south bound

Data type	Variable
Past travel speed	monthly average speed, daily average speed, Speed before 1 hour
Adjacent road	Speed of upstream road, Speed of downstream road, Speed of cross road (Sapyeong-daero)
Nearby main roads	Speed of main roads (Hyeonchung-ro, Seobinggo-ro, Hannam-daero, Dongho-ro, Gosanja-ro, Wangsimni-ro, Dongil-ro)

#### 4. 딥러닝 학습 수행

##### 1) 딥러닝 학습 모형 구축

본 연구에서는 Python기반의 딥러닝 라이브러리인 Tensorflow와 Keras를 사용하여 딥러닝 학습 모형을 구축하였다. 본 연구에서 구축한 딥러닝 학습모형은 2개의 LSTM 레이어와 3개의 Dense 레이어로 구성되어있

다. 이때 LSTM 레이어에서 출력하는 값을 Dense 레이어로 전달시 시퀀스 단위 출력을 입력으로 받을 수 있도록 하는 Time Distributed 레이어를 거친다. 또한 모형의 과적합 문제를 방지하기 위해 일정 비율의 퍼셉트론을 비활성화 하는 Drop Out 레이어를 추가하였다.

<Table 3> Model construction diagram

Layer Type	Output Shape	# of Parameters
Input	(None, 50)	
LSTM_1	(None, 12, 50)	12,400
LSTM_2	(None, 12, 50)	20,200
Dense_1	(None, 12, 64)	3,264
Dense_2	(None, 12, 128)	8,320
Dense_3	(None, 12, 32)	4,128
Dense_4(Output)	(None, 12, 1)	33

2) 가중치 초기화

딥러닝 학습에서 초기 가중치의 설정은 학습의 정확도를 결정하는 매우 중요한 역할을 한다. Hinton et al.(2006)은 초기 가중치를 적절한 값으로 초기화함으로써 기존 인공신경망 학습의 한계를 넘을 수 있었다. 초기 가중치 값을 0으로 설정해 학습을 시작하면, 오차역전파에서 가중치 값이 똑같이 갱신되기 때문에 가중치의 영향력이 고르게 나타나 학습이 되지 않는 문제가 발생한다.

본 연구에서는 Xavier Initialization을 이용해 가중치를 초기화 하였다. Xavier Initialization은 Glorot and Bengio(2010)가 제안한 초기화 방법으로, 이전 인공신경망과 다음 인공신경망의 개수를 통해 가중치를 초기화 하는 방법으로 균일분포를 따르는 방법과 정규분포를 따르는 방법이 있으며 다음과 식(8)과 식(9)와 같다.

- Xavier Normal Initialization

$$W \sim N(0, Var(W)) \dots\dots\dots (8)$$

$$Var(W) = \sqrt{\frac{2}{n_{in} + n_{out}}}$$

여기서,  $n_{in}$  = 이전 입력 퍼셉트론 수  
 $n_{out}$  = 다음 출력 퍼셉트론 수

- Xavier Uniform Initialization

$$W \sim U\left(-\sqrt{\frac{6}{n_{in} + n_{out}}}, +\sqrt{\frac{6}{n_{in} + n_{out}}}\right) \dots\dots\dots (9)$$

3) 최적화 모형

인공신경망의 최적화는 역전파 알고리즘(Backpropagation)을 통해 후방으로 전달되는 오차를 최소화할 통해 이루어지는데, 이때 많이 적용되는 최적화 알고리즘은 경사강하법(Gradient descent)으로 다양한 모형이 제시되고 있다.

본 연구에서는 ADAM(Adaptive Momentum Estimation) 모형을 이용해 최적화를 수행하였다. ADAM 모형은

현재 딥러닝 학습에 가장 광범위하게 이용되고 있는 최적화 알고리즘으로, 가장 좋은 성능을 보여주고 있다.  $w_{ij}$ 에 대한  $\text{momentum}(v_{ij})$ [이동방향]과  $\text{learning rate}$ [이동거리]를 조정하는 계수( $g_{ij}$ )를 다음 식과 같이 지수 이동 평균으로 계산해 최적화를 수행한다.

$$v_{ij}^{(t)} = \beta_1 v_{ij}^{(t-1)} + (1 - \beta_1) \frac{\partial L}{\partial w_{ij}^{(t)}} \dots\dots\dots (10)$$

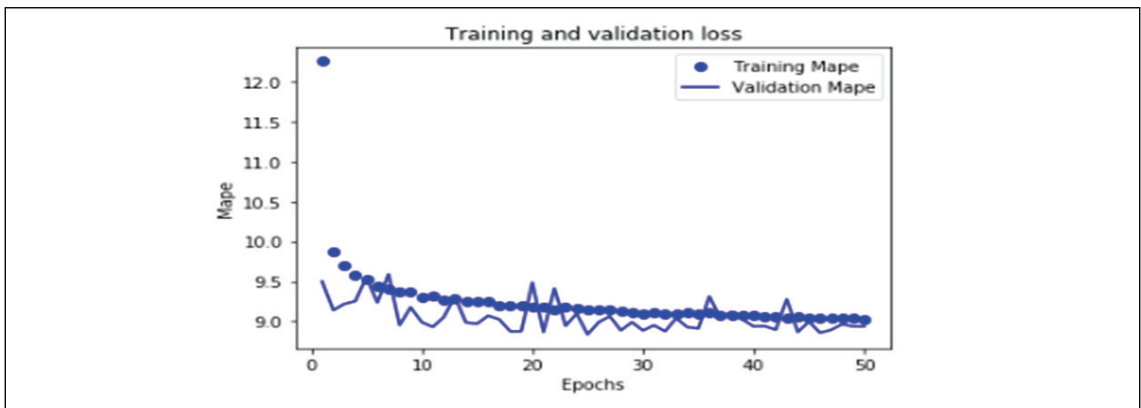
$$g_{ij}^{(t)} = \beta_2 g_{ij}^{(t-1)} + (1 - \beta_2) \left( \frac{\partial L}{\partial w_{ij}^{(t)}} \right)$$

### 3) 학습 및 검증

학습은 전체 96,456개 데이터셋 입력정보 중 70%인 67,519개 데이터셋을 학습입력정보로 활용하였으며, 나머지 28,937개 데이터 셋을 학습 완료 후 모델을 평가하기 위한 평가입력정보로 활용하였다. 학습은 50번의 epoch를 수행해 총 3,375,950회(67,519× epoch) 수행하였다.

예측에 대한 평가 지표로는 평균 절대 백분율 오차(Mean Absolute Percent Error; MAPE)을 활용하였다. MAPE는 정확도를 오차의 백분율로 표시하기 때문에 쉽게 가능하다는 장점을 갖고 있다.

학습과정에서 테스트데이터와 훈련데이터와 차이가 없는지 보기 위해 <Fig. 5>를 보았을 때, 비교적 큰 차이가 없는 모습을 보이고 있다는 것을 알 수 있다. 또한 훈련데이터에 대해서만 오차가 적은 과대적합 현상이 일어나지 않고 있다. 오히려 대부분의 과정에서 테스트데이터에 대한 평가인 Validation MAPE가 Training MAPE보다 낮은 것을 확인할 수 있다. 이를 통해 이러한 모델이 다른 데이터에도 알맞게 적용될 수 있다는 것을 볼 수 있었다.



<Fig. 5> Loss of Training and Test

## V. 분석 결과

### 1. 검증 시나리오

분석결과를 검증하기 위해서 본 연구에서는 다른 두 가지 방법과 비교를 통해 결과를 검증하였다. 첫 번째는 시간적, 공간적 영향을 모두 고려하여 예측한 연구에서 공간적 영향을 빼고 이전 1시간 통행속도만 입

력변수로 설정한 입력 데이터로 학습시켜 예측한 방법이다. 해당 방법은 사용하는 데이터는 동일하지만 입력변수의 개수가 11개 이상의 기존 연구에서 1개로 줄어들었다. 이로써 시간만 고려했을 때와 시간과 공간적 영향을 동시에 고려한 본 연구와 비교하여 모형의 효과성을 검증하고자 하였다.

두번째는 현재 서울시 교통정보센터 홈페이지에서 제공하고 있는 시간별 예측 속도와 실시간 통행속도와의 비교를 하여 이에 대한 오차율과 차이 값을 구해 본 연구를 통해 나온 결과와 비교하였다. 이를 통해 현재 공공기관에서 시민에게 제공하고 있는 정보의 정확도와 본 연구에서 개발한 모형의 예측 정확도를 비교하고자 한다.

## 2. 검증 시나리오와 비교 분석 결과

### 1) 전체 평균 결과 분석

본 연구에서 개발한 모형과 검증 시나리오간 통행속도 예측결과는 <Table 4>와 같이 나타났다. 본 연구 결과는 Study 항목으로 나타내었고, 이전 1시간 통행속도만 입력변수로 설정한 방법은 1 Variable 항목으로 두었다. 서울시 교통정보센터에서 제공하는 예측 속도와 실제 실시간 속도와의 차이를 분석한 방법은 서울시 교통정보센터 항목으로 설정하여 결과를 나타내었다. 방향별로 South Bound와 North Bound로 나누었는데, South Bound는 교보타워사거리에서 강남역으로 가는 방향이고 North Bound는 반대로 강남역에서 교보타워사거리로 가는 방향이다.

본 연구에서 개발한 모형에서 예측한 South Bound 통행속도의 오차율(MAPE)은 8.8%로 시간적 영향만 고려한 방법의 오차율과 비교하였을 때, 약 1%정도 오차율이 낮은 것으로 나타났으며, 서울시 교통정보센터의 예측 정확도에 비해서는 오차율이 4.5% 정도 낮은 것으로 나타났다.

North Bound의 예측 통행속도의 경우 South Bound 예측 통행속도와 비교하였을 때는 오차율이 좀 더 높은 것으로 나타났으나, 다른 2개 시나리오에 비해 오차율이 2.5%에서 3.0% 정도 낮을 것으로 나타나 다른 검증 시나리오에 비해 본 연구에서 개발한 모형의 예측력이 높은 것으로 나타났다.

<Table 4> Comparing results of study

Direction	Study		1 Variable (Speed before 1hour)		TOPIS	
	MAPE(%)	MAE(km/h)	MAPE(%)	MAE(km/h)	MAPE(%)	MAE(km/h)
South Bound	8.8	1.84	9.7	2.0	13.3	2.78
North Bound	9.9	1.32	12.5	1.77	13.0	1.84

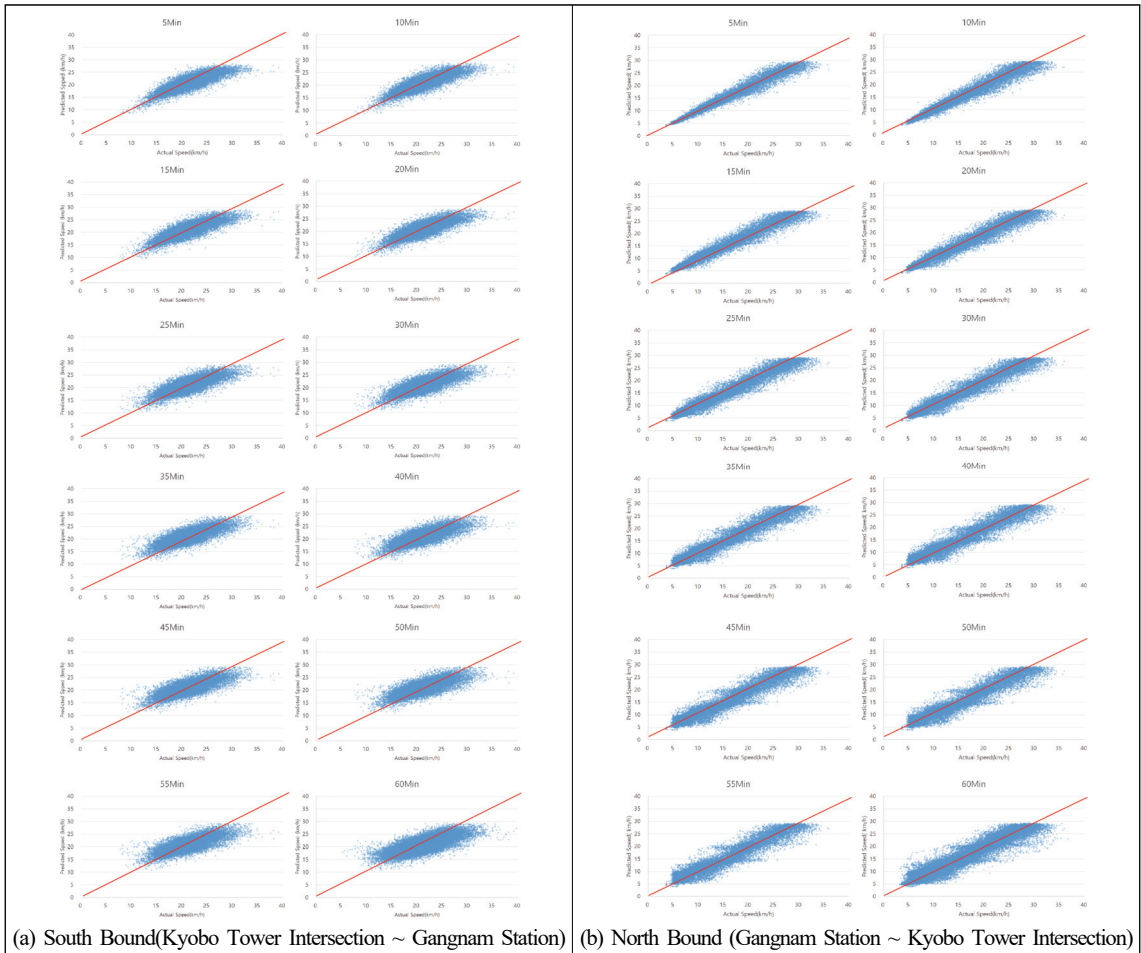
분석결과 시간적, 공간적 영향을 모두 고려하여 학습시킨 본 연구가 직전 1시간만 가지고 학습시킨 방법보다 더 좋은 결과를 나타나 모형의 효과성을 확인하였다. 또한 서울시에서 제공하는 정보와 비교하여도 더욱 우수한 결과를 내었으므로 보다 우위성을 가지고 있다는 것을 확인할 수 있었다.

### 2) 시간대별 세부 결과 분석

본 연구에서 개발한 모형은 향후 1시간 통행속도를 5분단위로 예측한다. 본 분석을 통해 향후 5분부터 60분까지의 통행속도 예측 정확도를 검토하였으며, 이를 위해 <Fig. 4>와 <Fig. 5>와 같이 예측통행속도와 실제 통행속도간 산점도 분석을 수행하였다.

따라서 산점도를 통해 실제 예측이 얼마나 정확히 이루어졌는지 확인하였다. 산점도는 직교 좌표계를 이용해 두 개 변수 간의 관계를 나타낸 그래프이다. <Fig. 6>이 바로 본 연구에서 예측된 통행속도와 실제 통행속도 간의 관계를 나타낸 산점도이다. 1시간 이후의 통행속도를 5분 단위로 예측하기 때문에 5분부터 60분까지 12개의 그래프가 나오게 된다. 해당 그림의 왼쪽 그래프는 교보타워사거리에서 강남역으로 가는 South Bound이고 오른쪽 그래프는 강남역에서 교보타워사거리로 가는 North Bound이다. 가로축은 실제 통행속도를 나타내고, 세로축은 예측된 통행속도를 나타낸다. 그리고 가운데 빨간색 선은 기울기가 1인 직선으로 해당 직선 위에 점이 있을수록 예측이 잘되었다는 것을 확인할 수 있다.

그림을 보면 대부분의 점들이 빨간색 직선 주위에 모여 군집이 기울기가 1인 직선 모양을 나타내고 있는 것을 볼 수 있다. 하지만 South Bound와 North Bound의 그래프를 보면 5분 후의 속도예측보다 60분 후의 속도 예측의 군집이 좀 더 퍼져있는 것을 알 수 있었다. 이를 통해 시간이 지날수록 예측의 성능이 조금씩 떨어지는 것을 확인할 수 있었다. 하지만 60분 후에도 파란점들이 비교적 빨간색 직선 위에 모여 있는 것을 볼 수 있다. 따라서 시간이 지날수록 예측 모델의 성능은 떨어지는 점이 있기는 하나 실제 속도와 예측한 속도가 비슷하다는 점을 확인할 수 있다.



<Fig. 6> Scatter plot between Predicted Travel Time and Actual Travel Time

<Table 5>는 분석 대상이 되는 강남역과 교보타워사거리 사이의 구간의 방향별 5분 단위 예측통행시간의 MAPE이다. 강남역에서 교보타워사거리까지의 North Bound의 오차율은 5분 후 예측에서는 5.5%로 가장 낮은 값을 보이고 있고 60분 후에는 13.5%로 가장 높은 값을 보이고 있다. 이를 통해 시간이 갈수록 오차율이 점차 증가하는 것을 알 수 있다. 5분 후 예측과 60분 후 예측의 오차율은 8%의 차이가 나고 있다. South Bound를 살펴보면 해당 방향 또한 7.4%인 5분 후 예측에서 60분 후에는 9.8%로 오차율이 늘어나고 있는 것을 확인할 수 있다. 하지만 그 차이는 2.4%로 반대방향에 비해서 크지 않은 것으로 나타났다.

<Table 5> 5minute prediction error rate of each direction

Time Bound	5 Min	10 Min	15 Min	20 Min	25 Min	30 Min	35 Min	40 Min	45 Min	50 Min	55 Min	60 Min	Total
	NB	5.5%	6.7%	7.4%	8.2%	9.2%	10.0%	10.8%	11.4%	12.0%	12.6%	13.0%	13.5%
SB	7.4%	7.6%	7.9%	8.3%	8.7%	9.0%	9.1%	9.3%	9.5%	9.6%	9.7%	9.8%	8.8%

## VI. 결론 및 향후과제

본 연구는 딥러닝 중에서 LSTM 모델을 활용하여 강남대로의 통행속도를 예측하는 연구를 진행하였다. 분석을 위해서 서울시 교통정보센터의 2017년 5분 단위 통행속도 데이터를 활용하였으며, 강남대로 중에서 교보타워사거리와 강남역 사이의 링크를 분석 대상으로 선정하였다. 문제가 있는 데이터에 대해서는 결측치를 채워 넣거나 중복된 값들을 삭제시키고 평활화를 시킴으로써 문제를 줄이고자 하였다. 딥러닝 모델을 구축하기 위해 필요한 Input 데이터를 만들기 위해서 도로의 과거 패턴과 인접한 상류부, 하류부 도로, 그리고 일정거리 떨어진 주요도로의 통행속도를 입력변수로 지정하였다. 분석 도로에 영향을 주는 입력변수들을 지정한 후에 LSTM 모델을 구축하여 딥러닝 학습을 시켰다. 학습은 Python의 딥러닝 라이브러리인 Keras를 통해 진행하였다. 학습이 잘되었는지 판단하기 위해 훈련데이터와 테스트데이터를 MAPE라는 지표로 서로 비교함으로써 훈련데이터에만 적합하지 않도록 모델을 시행착오를 거쳐 수정하였다. 또한 예측된 통행속도와 실제 통행속도를 비교하기 위해서 MAPE를 활용하여 결과를 산출하였다.

연구의 결과가 유의미한지 판단하기 위해서 다른 2개의 방법과 비교를 하였다. 하나는 본 연구의 입력변수 중 1시간 이전의 통행속도만 이용해 학습시킨 시간적 영향만 고려한 방법이고, 다른 하나는 서울시 교통정보센터에서 제공하는 예측속도와 실제 실시간 속도와의 차이를 산출한 방법이다. 본 연구 모델을 사용하였을 때 예측 오차는 South Bound는 8.8%, North Bound는 9.9%로 다른 방법들에 비해 적게는 1% 많게는 약 4.5%까지 차이가 나는 것으로 나타났다. 이를 통해서 연구 모델의 성능이 다른 방법들보다 우수한 것을 확인하였고, 본 연구가 유의미하다는 것을 판단할 수 있었다. 세부적인 결과를 확인하였을 때는 비교적 먼 미래를 예측할수록 예측의 정확성이 조금씩 떨어지는 것을 볼 수 있었다. South Bound의 경우, 5분 후의 예측 오차율이 7.4%인 반면 60분 후의 예측 오차율은 9.8%로 오차가 증가하였다. North Bound는 약 8%정도의 오차율이 증가하는 것으로 나타나 분석대상시간에서 멀어질수록 예측력이 떨어지는 것으로 나타났다.

본 연구는 컴퓨터의 성능을 고려하여 많은 데이터를 이용하지 못하였고, 더 좋은 학습 환경을 제공하지 못하였다는 한계를 가지고 있다. 향후 더 좋은 성능의 환경을 조성하거나 더욱 더 많은 데이터를 활용한다면 연구의 정확성을 높일 수 있을 것이다. 또한 지체수준에 따른 예측 성능 평가를 통해 결과에 따라 모형을 개

선할 경우 보다 정확한 교통예측정보 제공이나 이용자에게 맞는 출발 시간 제공 등의 활용이 가능할 것으로 기대된다.

## ACKNOWLEDGEMENTS

이 논문은 2019년도 서울시립대학교 교내학술연구비에 의하여 지원되었음.

## REFERENCES

- Choi Y. H.(2018), *The Prediction of Vehicle Speed Passing Urban Road Using Recurrent Neural Network Technique*, Chungbuk National University, pp.1-40.
- Chollet F.(2018), “Deep Learning with Python by Francois Chollet,” *Manning Publications(USA)*, pp.264-299.
- Daganzo C. F.(1994), “The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory,” *Transportation Research Part B: Methodological*, vol. 28, no. 4, pp.269-287.
- Glorot X. and Bengio Y.(2010), “Understanding the difficulty of training deep feedforward neural networks,” *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, PMLR 9, pp.249-256.
- Hinton G. E., Osindero S. and Teh Y. W.(2006), “The A fast learning algorithm for deep belief nets,” *Neural Computation*, vol. 18, no. 7, pp.1527-1554.
- Hochreiter S. and Schmidhuber J.(1997), “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp.1735-1780.
- Jeon H. J.(2018), *A Deep-learning Approach to Predict Short-term Traffic Speeds Considering City-wide Spatio-temporal Correlations*, Jung-ang University, pp.1-36.
- Lee M. S.(2016), *Forecasting short-term travel speed in a dense highway network considering both temporal and spatial relationship- Using a deep-learning architecture*, Jung-ang University, pp.1-48.
- Ma X., Tao Z., WangY., Yu H. and Wang Y.(2015), “Long short-term memory neural network for traffic speed prediction using remote microwave sensor data,” *Transportation Research Part C: Emerging Technologies*, vol. 54, pp.187-197.
- Mckinney W.(2013). *Python for Data Analysis*, HANBIT media(KOREA), pp.158-214.
- Minsky M. and Papert S.(1969), *Perceptrons: An Introduction to Computational Geometry*, The MIT Press.
- Newell G. F.(1993), “A simplified theory of kinematic waves. I: general theory; II: Queuing at freeway bottle- necks; III: Multi-destination flows,” *Transportation Research Part B*, vol. 27B, no. 4, pp.281-314.
- Rosenblatt F.(1958), “The perceptron : A probabilistic model for information storage and organization

in the brain,” *Psychological Review*, vol. 65, no. 6, pp.386-408.

Rumelhart D. E., Hinton G. E. and Williams R. J.(1986), “Learning representations by back-propagating errors,” *NATURE*, vol. 323, no. 9, pp.533-536.

The Seoul Transportation Information Center, <https://www.seoul.go.kr>, 2019.06.28.

Wang W. and Li X.(2018), “Travel Speed Prediction with a Hierarchical Convolutional Neural Network and Long Short-Term Memory Model Framework,” *European Transport Conference*, pp.6-15.