

교통 돌발 상황을 고려한 도로 속도 예측 기법

Road Speed Prediction Scheme Considering Traffic Incidents

박송희*, 최도진*, 복경수**, 유재수*
충북대학교 정보통신공학과*, 원광대학교 SW융합학과**

Songhee Park(shpark1586@cbnu.ac.kr)*, Dojin Choi(mycdj91@cbnu.ac.kr)*,
Kyoungsoo Bok(ksbok@wku.ac.kr)**, Jaesoo Yoo(yjs@cbnu.ac.kr)*

요약

교통 혼잡으로 인한 사회적 비용이 증가하면서 도로 속도를 예측하기 위한 다양한 연구들이 진행되고 있다. 도로 속도 예측의 정확도를 향상시키기 위해서는 교통 돌발 상황을 고려할 필요가 있다. 본 논문에서는 교통 돌발 상황을 고려한 도로 속도 예측 기법을 제안한다. 제안하는 기법은 연결된 도로들이 미치는 영향을 반영하기 위해서 예측 도로의 속도 데이터 뿐만 아니라 연결된 도로들의 속도 데이터도 이용한다. 또한, 돌발 상황으로 인한 혼잡을 예측하기 위해 속도의 변화량을 분석한다. 연결된 도로와 타겟 도로의 속도 데이터를 LSTM의 입력 데이터로 이용하여 1차적으로 도로 속도를 예측한다. 교통 돌발 상황으로 도로의 규칙적인 흐름이 깨지며 발생하는 예측 오차를 줄이기 위해 이벤트 가중치를 적용하여 최종적으로 도로 속도를 예측한다. 다양한 성능 평가를 통해 제안된 방법의 우수성을 입증한다.

■ 중심어 : | 도로 속도 예측 | 교통 혼잡도 | 교통 돌발 상황 분석 | 교통 예측 | 교통 데이터 분석 |

Abstract

As social costs of traffic congestion increase, various studies are underway to predict road speed. In order to improve the accuracy of road speed prediction, unexpected traffic situations need to be considered. In this paper, we propose a road speed prediction scheme considering traffic incidents affecting road speed. We use not only the speed data of the target road but also the speed data of the connected roads to reflect the impact of the connected roads. We also analyze the amount of speed change to predict the traffic congestion caused by traffic incidents. We use the speed data of connected roads and target road with input data to predict road speed in the first place. To reduce the prediction error caused by breaking the regular road flow due to traffic incidents, we predict the final road speed by applying event weights. It is shown through various performance evaluations that the proposed method outperforms the existing methods.

■ keyword : | Road Speed Prediction | Traffic Congestion | Traffic Incident Analysis | Traffic Prediction | Traffic Data Analysis |

* 이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단(No. 2019R1A2C2084257), 2017년도 정부(과학기술정보통신부)의 재원으로 한국연구재단-차세대정보·컴퓨팅기술개발사업(NRF-2017M3C4A7069432), 중소벤처기업부 '산업전문인력 역량강화사업'의 재원으로 한국산업기술진흥원(KIAT)의 지원을 받아 수행된 연구임(2019년 기업연계형연구개발인력양성사업, 과제번호 : S2755555).

접수일자 : 2020년 01월 28일
수정일자 : 2020년 02월 25일

심사완료일 : 2020년 02월 25일
교신저자 : 유재수, e-mail : yjs@chungbuk.ac.kr

I. 서론

교통 혼잡으로 발생하는 사회적 비용이 33조원을 넘어서면서 전국의 지자체들은 교통난을 해소하기 위한 다양한 교통 체계 개선 정책을 시행하고 있다[1][2]. 교통 혼잡을 예측할 경우 교통 혼잡 발생 비율을 줄일 수 있으며 정체에 대한 대안을 미리 제시하여 교통 혼잡으로 발생하는 다양한 문제들을 피할 수 있다. 일반적으로 교통 혼잡도는 도로 속도를 통해 판별된다. 도로 속도는 교통 상황의 중요한 지표이며 도로의 속도에는 다양한 요소가 영향을 미친다. 도로 속도에 영향을 미치는 요소로는 도로의 속도, 도로가 수용할 수 있는 교통량, 시간에 따른 교통 흐름, 연결된 도로의 영향, 사고, 공사, 날씨 그리고 명절과 같은 특별한 날 등이 있다. 이와 같이 많은 요소들이 교통 혼잡에 영향을 주므로 각 요소들 또는 결합된 요소들이 교통 혼잡에 미치는 영향을 분석할 필요가 있다.

도로 속도는 교통 혼잡도를 결정하는 중요한 지표로 활용되기 때문에 도로 속도를 예측하기 위한 다양한 연구들이 진행되고 있다[3-12]. 기존의 도로 속도 예측 기법들은 속도와 교통량을 주로 고려하였으나 교통 돌발 상황을 고려하지 않았다. 시간에 따른 속도와 교통량 변화를 이용한 예측은 도로의 규칙적인 흐름을 파악할 수 있다. 하지만 교통 돌발 상황 발생 시에는 도로의 규칙적인 흐름이 깨지므로 예측이 어렵다. 따라서 교통 돌발 상황에 따른 영향을 분석하여 도로 속도 예측의 정확도를 높일 수 있다.

도로 속도에는 차량 속도, 교통량 뿐만 아니라 다양한 환경적인 요소들이 영향을 미친다. 도로는 모두 연결되어 있으며 상호 영향을 미친다. [4-6]에서는 연결된 도로의 영향을 고려하여 도로 속도를 예측하였다. [5]에서는 연결된 도로의 영향과 교통사고, 도로 공사, 날씨, 특별한 날 등을 고려하여 교통 혼잡도를 예측하였다. 특히, [6]에서는 속도와 교통량의 영향을 고려하였으며, 이웃 도로들의 영향을 고려하였다. 타겟 도로는 속도를 예측하고자 하는 도로를 의미하고 이웃 도로는 타겟 도로 구간에 연결된 모든 하류 구간으로 정의하였다. 속도 데이터와 교통량 데이터를 각각 이용하여 예측한 속도의 성능을 비교한 결과, 타겟 도로와 이웃 도

로의 속도 데이터를 이용하여 예측한 도로 속도가 가장 높은 정확도를 보였다.

본 논문에서는 타겟 도로와 이웃 도로의 영향과 교통 돌발 상황을 고려한 도로 속도 예측 기법을 제안한다. 타겟 도로와 이웃 도로들의 영향을 고려하기 위하여 시계열 데이터 학습에 적합한 LSTM(Long Short-Term Memory)[13] 모델을 이용한다. 돌발 상황의 영향을 반영하기 위해 속도 변화량과 변화량에 따른 예측 속도와 실제 속도를 통계 분석하여 돌발 상황을 판단하고 이벤트 가중치를 정의한다. 기계학습 모델을 이용하여 타겟 도로와 이웃 도로들의 영향을 반영한 속도를 예측한 후 돌발 상황 발생 여부에 따라 이벤트 가중치를 반영하여 최종 도로 속도를 예측한다. 도로 속도 예측을 통해 교통 혼잡 상황을 판단할 수 있으며 미리 대안을 제시하여 교통 혼잡으로 발생하는 피해와 비용을 줄일 수 있다. 또한, 본 논문에서 제안하는 기법은 실시간 도로 상황을 제공하는 서비스나 속도 변화량 분석을 통해 도로의 정체 구간을 판별하거나 실시간 길안내 서비스에 활용될 수 있다.

본 논문의 구성은 다음과 같다. II장에서는 관련 연구에 대하여 기술하고 III장에서는 제안하는 도로 속도 예측 기법에 대해 설명한다. IV장에서는 제안하는 기법의 우수성을 확인하기 위해 성능 평가에 대해 기술한다. 마지막으로 V장에서는 논문의 결론에 대해 기술한다.

II. 관련 연구

도로 속도는 교통 혼잡도를 판단하는 중요한 지표이다. [4]에서는 베이지안 네트워크[14]를 이용하여 도로 속도를 예측하였다. 예측 도로의 상·하류 도로의 도로 속도를 예측 도로의 속도에 영향을 미치는 요인으로 설정하였다. 특정 노드에 대한 도로 속도 예측이 인근 링크들의 최근 정보에 의해 결정되도록 조건부 사건들로 정의하여 베이지안 네트워크 모형을 설계하였다. 또한, 일반적으로 돌발 상황에 대한 충분한 데이터를 확보하기가 어렵고, 이는 학습이 어렵기 때문에 이 문제를 해결하기 위해 실시간 도로 속도를 예측 모델에 반영하여 최종 도로 속도를 예측하였다.

도로의 혼잡도에는 다양한 요소들이 영향을 미친다. 기존의 도로 혼잡도 예측 기법은 과거 도로 혼잡도 데이터와 연결된 도로의 영향, 날씨, 사고, 특별한 날, 그리고 경찰에 의해 발생하는 정체를 고려하여 도로의 혼잡도를 예측하였다[5]. 인접한 도로의 영향과 시간에 따른 흐름을 반영하기 위해 인접한 도로와 예측하고자 하는 도로의 혼잡도를 2차원의 배열로 구성하여 생성한 스냅 샷들을 CNN(Convolutional Neural Network) [15]의 입력 데이터로 사용하였다. 도로 혼잡도에 영향을 주는 요소들을 반영하기 위해 날씨, 휴일, 도로 공사 그리고 특별한 날 등의 데이터를 정규화하였다. 정규화한 데이터와 예측한 도로의 상태를 완전 연결 신경망 네트워크의 입력으로 주어 최종 도로 혼잡도를 예측한다. 2차원의 입력 정보를 이용하여 인접한 도로의 영향을 고려하고 도로 속도에 영향을 미치는 다양한 데이터들을 고려하여 도로 혼잡도 예측의 정확도를 향상시켰다.

LSTM은 RNN(Recurrent Neural Network)[16]의 한 종류로 긴 의존 기간을 필요로 하는 학습을 수행할 능력을 갖고 있다. LSTM은 RNN의 메모리 셀에 입력 게이트, 망각 게이트, 출력 게이트를 추가하여 불필요한 기억을 지우고, 기억해야 할 것들을 결정하여 긴 시퀀스의 입력을 높은 성능으로 처리할 수 있다. 따라서 다양한 연구들에서 긴 시퀀스 데이터를 처리하기 위해 LSTM을 사용하고 있다[18][19]. [6]에서는 LSTM 모델을 이용하여 5분 후의 도로 속도를 예측하였다. 도로 속도에 영향을 미치는 요소를 분석하여 예측 속도의 정확도를 높이고자 하였다. 타겟 도로의 과거 속도와 교통량의 변화가 타겟 도로의 속도에 미치는 영향, 그리고 이웃 도로들의 속도와 교통량의 변화가 타겟 도로의 속도에 미치는 영향을 독립적으로 반영하여 예측하였다. 예측한 속도와 실제 속도의 RMSE(Root Mean Square Error) 값을 계산하여 비교한 결과, 타겟 도로와 이웃 도로들의 속도를 고려한 예측 속도가 가장 높은 정확도를 보였다.

[4]에서는 베이지안 네트워크 기반으로 예측하고자 하는 도로의 상·하류 도로의 과거 속도 데이터와 실시간 교통 정보를 이용하여 5분에서 60분 후의 도로 속도를 예측했으나 현재 시간에서 멀어질수록 예측 오차가

증가하였다. 돌발 상황이 도로 속도에 미치는 영향에 대비하여 실시간 교통 정보를 반영했으나 예측 시간이면 미래일수록 실시간 교통 정보의 영향이 적어지므로 예측 오차가 증가한다.

[5]에서는 과거 도로 혼잡도 데이터와 도로 속도에 영향을 미치는 요소인 날씨, 휴일, 도로 공사 그리고 특별한 날 등의 데이터를 이용하여 예측 혼잡도의 정확도를 높였다. 해당 연구에서는 각 도로 환경 데이터를 0~1사이의 값으로 정규화 했으나 도로 환경 데이터들은 서로 다른 특징을 보이므로 각 환경 데이터가 도로의 혼잡도에 미치는 영향을 연구할 필요가 있으며 각 데이터를 값으로 산출하는 방법을 제시하지 않았다.

[6]에서는 이웃 도로들의 영향을 고려하기 위해 LSTM 모델의 입력으로 타겟 도로와 이웃 도로들의 과거 속도 데이터를 입력으로 주어 가까운 미래의 도로 속도를 예측하였다. 하지만 상대적으로 먼 미래의 도로 속도를 예측할 경우 [4]와 같이 돌발 상황 발생 시 예측 오차가 증가하는 문제가 발생하므로 도로 환경 데이터를 추가적으로 고려할 필요가 있다. [12]에서는 RNN을 이용하여 도로 혼잡도를 예측하였다. 하지만 RNN은 시간 격차가 커지면 초기 데이터의 영향이 사라지는 기울기 손실 문제(Vanishing gradient problem)가 발생한다.

III. 제안하는 도로 속도 예측 기법

1. 전체 처리 과정

교통 혼잡 비용이 증가하면서 이 문제를 해결하기 위해 교통 혼잡도와 속도를 예측하기 위한 다양한 연구들이 진행되고 있다. 기존의 도로 속도 예측 기법에서는 돌발 상황이 발생한 경우 예측 정확도가 크게 저하되는 문제를 보완하기 위해 가장 최신의 도로 속도 데이터를 이용하였다. 하지만 예측 시점이 멀어지는 경우, 현재 시점에서 얻을 수 있는 가장 최신의 도로 속도 데이터를 반영하더라도 현재 시점과 예측 시점 사이에 발생하는 돌발 상황으로 인한 오차를 줄이는데 한계가 있다. 따라서 돌발 상황을 추가적으로 고려할 필요가 있다. CNN 모델을 이용한 기존 도로 혼잡도 예측 기법은 도

로 속도에 영향을 미치는 다양한 요소들을 고려하였으나 서로 다른 특징을 가지는 요소들을 특정 값으로 산출하는 방법을 제시하지 않았다.

본 논문에서는 기존 기법들이 갖는 문제점들을 해결하기 위하여 교통 돌발 상황을 고려한 도로 속도 예측 기법을 제안한다. 제안하는 기법은 예측을 위해 타겟 도로와 이웃 도로들의 속도 데이터를 활용한다. 또한, 사고나 공사와 같이 도로 속도에 급격한 변화 흐름을 가져오는 돌발 상황을 고려하기 위해 속도 변화량을 분석하여 이에 대한 영향을 반영한다. 따라서 돌발 상황 발생 시 도로 속도 예측 정확도가 떨어지는 기존의 문제를 보완할 수 있다.

[그림 1]은 제안하는 도로 속도 예측 기법의 전체 처리 구조를 나타낸다. 처리 과정은 학습 단계인 오프라인 처리와 예측 단계인 온라인 처리로 구분한다. 먼저, 온라인 처리에서는 실시간으로 수집되는 데이터를 정규화 과정을 거쳐 데이터 집합을 생성한다. 생성한 예측 데이터 집합을 학습된 LSTM 모델의 입력 데이터로 사용하여 학습 모델을 통해 1차적으로 속도를 예측한다. 현재 시점까지의 타겟 도로와 이웃 도로의 속도로 구성된 예측 데이터 집합으로 타겟 도로와 이웃 도로들의 영향을 반영하여 1차 속도를 예측한다. 하지만 1차 예측 속도는 교통 돌발 상황이 미치는 영향을 고려하지 않았기 때문에 돌발 상황 발생시 정확도가 저하되는 문제를 보완해야 한다. 따라서 1차 예측 속도에 교통 돌발 상황에 의해 급변하는 도로의 흐름을 반영하기 위한 이벤트 가중치를 적용하여 최종 속도를 예측한다. 오프라인 처리는 학습 데이터 집합을 LSTM 모델의 입력으로 넣어 예측을 위한 LSTM 학습 모델을 생성한다. 학습 데이터 집합은 타겟 도로와 이웃 도로들의 과거 속도 데이터로 구성된다. 학습을 통해 예측 속도와 실제 속도를 비교하면서 최적의 가중치를 갖는 학습 모델을 생성한다.

2. 정규화

신경망 모델을 이용한 예측 기법들은 높은 정확도를 보인다. 신경망 모델을 이용한 학습 결과에는 데이터의 형태가 많은 영향을 미친다. 따라서 LSTM 모델의 입력으로 사용할 5분 단위로 집계된 타겟 도로와 이웃 도로

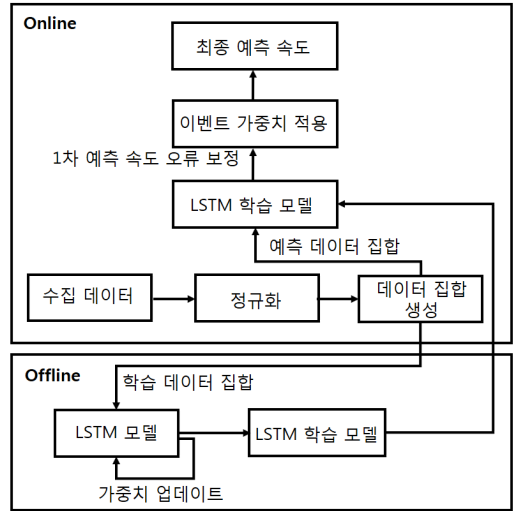


그림 1. 제안하는 기법의 처리 절차

들의 속도 데이터를 정규화한다. 이 단계의 목적은 원본 데이터를 신경망 모델 학습에 적합하도록 생성하는 것이다. 도로 속도 데이터 특성상 시간별 도로의 속도는 무시할 수 없는 요소이다. 따라서 데이터가 누락된 경우에는 Python pandas 라이브러리[20]에서 제공하는 fillna()함수를 이용하여 이웃하고 있는 값의 평균값으로 누락 데이터를 채울 수 있다. 연속으로 데이터가 누락된 경우에는 과거 같은 요일, 시 속도의 평균값으로 채운다.

신경망 모델의 입력 데이터는 각 데이터의 특성별로 독립적으로 정규화해야 한다[17]. 따라서 속도 데이터를 식 1과 같이 정규화한다. 신경망 모델이 쉽게 학습하기 위해서는 일반적으로 0~1사이의 값으로 정규화하며 이는 모든 특성이 비슷한 범위를 가지도록 균일해야 한다는 것을 의미한다. 식 1에서 S_R 은 도로 R 에서 수집한 전체 속도 데이터를 의미한다. S_R^t 은 도로 R 에서 수집한 t 시간의 속도를 의미한다. 각 도로의 일반적인 특성을 반영하기 위해 각 도로의 속도 데이터 중에서 가장 큰 값으로 해당 도로에서 수집한 속도 데이터를 나누어서 0에서 1사이의 값으로 모두 비슷한 범위를 가지도록 정규화 한다. 예를 들어, S_R 이 110, S_R^t 이 89라고 가정했을 때, S_R^t 을 정규화하면 0.81이 된다.

$$normalized S_R^t = S_R^t / MAX(S_R) \quad (1)$$

3. 데이터 집합 생성

도로 속도 예측을 위해서는 예측 모델을 생성하기 위한 학습 데이터 집합과 실제 예측을 위해 사용할 예측 데이터 집합이 필요하다. [표 1]은 데이터 정규화 과정을 수행한 데이터 예측 데이터 집합의 예시를 나타낸다. S_T^i 와 $S_{N_1}^i, S_{N_2}^i, S_{N_3}^i$ 는 i 시점의 타겟 도로와 타겟 도로의 하류 구간인 이웃 도로들의 데이터를 의미한다. 모든 데이터는 5분 단위로 기록되어 있으며 시간 단위는 5분이다. 예측을 하는 경우에는 실제 30분 후의 타겟 도로 속도인 S_T^{i+6} 를 알 수 없으므로 i 시점의 도로 속도 데이터 이용하여 [표 1]과 같이 예측 데이터 집합을 구성한다. 예측 데이터 집합은 학습된 LSTM 모델의 입력으로 사용한다.

표 1. 예측 데이터 집합 예시

구분	$S_{N_1}^i$	$S_{N_2}^i$	$S_{N_3}^i$	S_T^i
t	0.870698	0.905612	0.848839	0.851250
t+1	0.866772	0.907388	0.855186	0.845409
t+2	0.862847	0.909164	0.861533	0.839569
t+3	0.906615	0.917066	0.910139	0.877798
t+4	0.880805	0.910407	0.865375	0.829195
t+5	0.851535	0.942584	0.845357	0.842598
t+6	0.902145	0.920457	0.864527	0.864215
t+7	0.894255	0.901565	0.902145	0.854112
t+8	0.904232	0.920545	0.901565	0.945515

[표 2]는 학습 데이터 집합 예시를 나타낸다. 학습 데이터 집합은 예측한 속도와 실제 30분 후의 도로 속도를 비교하는 과정을 반복하여 LSTM 학습 모델을 생성하는데 이용한다. 학습 시에는 과거 데이터를 사용하기 때문에 $i+6$ 시점의 실제 속도 데이터를 얻을 수 있다. 따라서 i 시점과 $i+6$ 시점의 타겟 도로와 이웃 도로의 속도 데이터를 [표 2]와 같이 같은 열로 구성한다.

표 2. 학습 데이터 집합 예시

구분	$S_{N_1}^i$	$S_{N_2}^i$	$S_{N_3}^i$	S_T^i	S_T^{i+6}
t	0.870698	0.905612	0.848839	0.851250	0.864215
t+1	0.866772	0.907388	0.855186	0.845409	0.854112
t+2	0.862847	0.909164	0.861533	0.839569	0.945515
t+3	0.906615	0.917066	0.910139	0.877798	0.934532
t+4	0.880805	0.910407	0.865375	0.829195	0.928745
t+5	0.851535	0.942584	0.845357	0.842598	0.843523
t+6	0.902145	0.920457	0.864527	0.864215	0.874353
t+7	0.894255	0.901565	0.902145	0.854112	0.892345
t+8	0.904232	0.920545	0.901565	0.945515	0.902343

4. 예측 모델 생성

본 논문에서는 이웃 도로의 영향을 반영하여 도로 속도를 예측하기 위해 LSTM 학습 모델을 생성한다. 오프라인 처리에서 학습 모델을 생성하는 과정을 수행한다. [그림 2]는 제안하는 기법의 LSTM 모델 학습 절차를 나타낸다. LSTM 모델은 LSTM, Dropout, Dense 레이어로 구성한다. [표 2] 형식으로 구성된 학습 데이터 집합을 LSTM 레이어의 입력으로 넣는다. S_T^i 와 S_N^i 은 t 시점의 타겟 도로와 이웃 도로들의 데이터를 의미하며 이 데이터들을 이용하여 타겟 도로와 이웃 도로들이 타겟 도로의 속도에 미치는 영향을 반영할 수 있다.

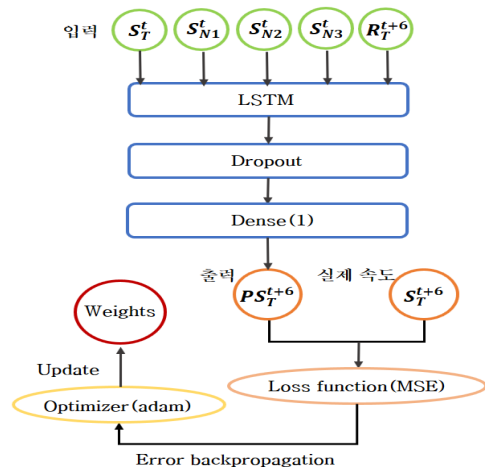


그림 2. 제안하는 기법의 LSTM 모델 학습 절차

학습 데이터 집합의 30분 후의 실제 도로 속도(S_T^{t+6})는 손실 값 계산 단계에서 사용한다. 과적합을 방지하기 위해 Dropout 레이어를 합하였으며 0.3의 비율을 적용한다. 모든 입력 뉴런과 출력 뉴런을 연결하는 Dense 레이어를 통해서 하나의 결과를 출력하도록 설정한다. PS_T^{t+6} 는 예측한 타겟 도로의 속도를 의미한다. 손실 함수 MSE(Mean Squared Error)를 이용하여 식 2와 같이 실제 속도와 예측 속도의 오차를 계산한다. 손실 값이 작을수록 정확하게 예측했음을 의미한다. 학습 속도를 빠르고 안정적이게 할 수 있도록 최적화 함수로 아담(adam)을 사용한다. 손실 값을 가장 작게 만들 수 있는 최적의 가중치를 구하기 위해 역전파 과정을 수행하며, 이 과정을 반복하여 최적의 가중치를 갖는 학습 모델을 생성한다.

$$MSE = \frac{1}{n} \sum_{i=1}^n (S_T^{t+6} - PS_T^{t+6})^2 \quad (2)$$

5. 1차 속도 예측

오프라인 처리에서 타겟 도로와 이웃 도로들의 영향을 반영하여 최적의 가중치를 갖는 학습 모델을 생성하였다. 이 단계에서는 학습된 LSTM 모델을 이용하여 30분 후의 도로 속도를 예측한다. 이 단계에서 예측한 속도는 1차 예측 속도라고 정의한다. 1차 속도 예측을 위해 타겟 도로와 이웃 도로들의 속도 데이터로 구성된 예측 데이터 집합을 이용한다. 예측 데이터 집합을 학습된 LSTM 모델의 입력 데이터로 넣어 최적의 가중치를 이용하여 30분 후의 도로 속도를 예측한다.

[그림 3]은 시간에 따라 실제 도로 속도와 1차 예측 속도를 나타낸 것이다. 1차 예측 속도는 30분 전의 속도 패턴을 유사하게 따라가고 있으며 돌발 상황으로 도로 속도가 급변하는 흐름을 뒤늦게 완만한 곡선으로 반영함으로써 실제 속도와 예측 속도의 차이가 커지는 것을 볼 수 있다. 따라서 도로 상황을 급변하게 하는 교통 돌발 상황을 고려할 필요가 있다. 도로 속도 변화량을 분석하고 예측 속도 보정을 통해 1차 예측 속도의 오류를 보정한다.

6. 예측 속도 보정

학습 모델만으로 예측한 1차 예측 속도는 바로 이전 시점의 속도 데이터가 가장 큰 영향을 미치므로 실제 속도를 뒤늦게 따라가는 흐름을 보인다. 교통 돌발 상황의 영향으로 급격한 속도 변화가 일어나는 구간을 뒤늦게 완만한 곡선으로 예측하여 예측 오차가 증가한다. 따라서 본 논문에서는 예측 정확도를 높이기 위해 1차 예측 속도를 보정하여 최종 속도를 예측한다.

교통사고나 도로공사 등의 돌발 상황으로 인해 도로의 규칙적인 흐름이 깨지는 경우 도로 속도가 급격히 감소한 후에 바로 회복되는 패턴을 보인다. 이를 반영하기 위해 실시간으로 기록되는 사고, 공사와 같은 실시간 이벤트를 이용한다. 사고 데이터를 사용하기 위해서는 실시간으로 기록되는 사고 정보를 읽어서 사고 발생 위치와 사고의 크기를 반영해야 한다. 하지만 사고가 발생하는 시점과 사고가 기록되는 시점이 다르며 대부분의 사고 데이터가 정확하지 않다. 따라서 속도 변화량을 분석하여 돌발 상황 발생 여부를 판단하고 돌발

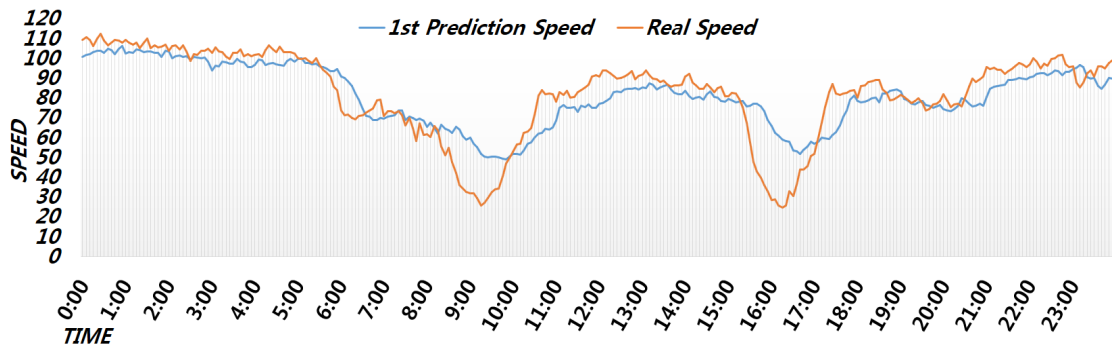


그림 3. 1차 예측 속도와 실제 속도 비교

상황 발생 시 이벤트 가중치를 적용하여 1차 예측 속도를 보정한다.

돌발 상황이 발생한 경우, 도로의 속도는 급격히 감소했다가 회복하는 패턴을 보인다. 감소 구간과 회복 구간에 각 감소 가중치와 회복 가중치를 적용하여 1차 예측에서 발생했던 오차를 감소시킨다. 먼저, 속도 변화량과 오차율의 관계 분석을 통해 감소 구간과 회복 구간을 정의한다. 감소 구간을 판단하는 기준을 정의하기 위해서 실제 속도 변화량과 예측 속도 변화량을 활용하며 이벤트 가중치 적용 시에는 예측 속도 변화량과 과거 실제 속도 변화량을 이용한다. 예측 속도의 변화량과 과거 실제 속도 변화량, 그리고 실제 속도의 변화량은 식 3~5와 같이 계산한다. 타겟 도로의 1차 예측 속도 변화량인 ΔPS_T^t 는 t 시점을 현재 시점으로 봤을 때, 15분 이전의 예측 속도와 차를 의미하며, t 단위는 5분으로 정의한다. t 시점에서 타겟 도로의 속도를 예측하고자 할 때, 얻을 수 있는 실제 속도는 30분 전의 데이터이므로 그로부터 15분 전인 45분전의 실제 속도와 차를 계산한다. 이는 식 4와 같이 계산한다. 마지막으로 실제 속도 변화량인 ΔS_T^t 은 t 시점으로부터 15분 이전의 실제 속도와 차를 의미한다.

$$\Delta PS_T^t = PS_T^{t-3} - PS_T^t \quad (3)$$

$$\Delta HS_T^t = S_T^{t-9} - S_T^{t-6} \quad (4)$$

$$\Delta S_T^t = S_T^{t-3} - S_T^t \quad (5)$$

속도 변화량에 따른 예측 속도의 오차율 통계 값을 이용하여 속도 감소 구간의 기준 값을 정의한다. 예측 속도의 오차율(Error rate)은 식 6과 같이 계산하며, 오차율이 1에 가까울수록 정확도가 높은 것을 의미한다. e_T^t 는 t 시간, 타겟 도로의 예측 속도 오차율을 의미한다. 1 이상은 실제 속도보다 낮게 예측했음을 의미하며 1 이하는 실제 속도보다 높게 예측했음을 의미한다.

$$e_T^t = S_T^t / PS_T^t \begin{cases} e > 1 : S_T^t > PS_T^t \\ e < 1 : S_T^t < PS_T^t \end{cases} \quad (6)$$

속도 감소 구간과 회복 구간을 판단하기 위해 과거 이벤트 발생 시 도로의 속도 변화량이 예측 오차율에 미친 영향을 분석하였다. 먼저, 1차 예측 속도의 변화량으로 속도 감소 추세를 알 수 있다. 하지만 예측 속도는 [그림 3]의 결과와 같이 뒤늦게 도로 속도의 흐름을 따라가며, 실제 속도보다 완만한 흐름을 보이므로 정확한 감소 추세를 알 수는 없다. 따라서 과거 실제 속도 변화량을 추가적으로 이용하여 속도 급감 구간을 판단한다.

[그림 4]는 과거 데이터를 이용하여 분석한 예측 속도와 실제 속도 변화량에 따른 각 오차율 통계를 나타낸다. 주황색 막대 그래프는 예측 속도 변화량을 의미한다. 예측 속도 변화량이 5이하인 경우에는 1에 가까운 수치로 높은 정확도를 보이며 예측 속도 변화량이 6이상 증가하는 경우 오차율이 낮아진다. 그리고 예측 속도 변화량이 14이상 증가하는 경우 오히려 오차율이 1 이상으로 높아지는데 이는 급격히 감소한 이후에 도로 속도가 바로 회복 되는 경우 때문에 이러한 현상이 발생한다. 통계 분석에 따라서 속도 감소 구간을 결정하는 첫 번째 기준을 예측 속도 변화량이 6이상일 때로 정의한다. 하지만 예측 속도는 실제 속도보다 완만한 흐름을 보이므로 정확한 감소 추세를 알 수 없다. 따라서 실제 속도 변화량을 추가적으로 고려하여 속도 감소 구간의 판단 기준을 정의한다. 파란색 막대 그래프는 실제 속도 변화량을 의미한다. 실제 속도 변화량이 4부터 10까지 정확도가 서서히 낮아지고 있으며 그 이후로 증감을 반복한다. 이는 실제 속도 변화량이 4이상 변화하면 예측 정확도가 낮아지고, 10일 때 급격한 변화로 인해 가장 정확도가 낮아진 것을 의미하며, 11이상으로는 회복 구간이 발생하면서 오차율이 증감을 반복하는 것을 의미한다. 따라서 실제 속도 변화량이 10이상 일 때를 두 번째 속도 감소 구간의 판단 기준으로 정의한다.

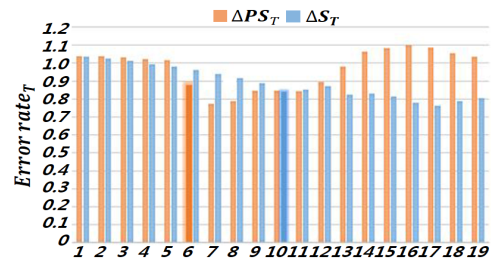


그림 4. 예측 속도와 실제 속도 변화량에 따른 오차율

예측 속도와 실제 속도 변화량에 따른 오차를 분석을 통해 두 가지 속도 감소 구간 판단 기준을 정의하였다. 기준에 부합하는 경우 1차 예측 속도에 가중치를 주어 속도를 보정한다. 가중치는 속도 급감 구간의 기준이 되는 두 변화량의 평균 오차를 값인 0.8로 정의하여 예측 속도에서 20% 감소한 속도를 반영한다. 시간이 지나면서 예측 모델이 감소 흐름을 반영하므로 30분 이후부터 가중치를 높여준다. [그림 4]의 주황색 막대그래프는 회복 구간이 발생하면서 오차율이 증가하는 것으로 보이지만 실제 속도 변화량 그래프인 파란색 그래프를 보면, 속도 변화량이 증가할수록 예측 오차율이 낮아지는 것을 볼 수 있다. 이는 속도가 크게 감소할수록 실제 속도보다 예측 속도를 더 높은 값으로 예측했으며, 그 차가 커지는 것을 의미한다. 따라서 [표 3]과 같이 예측 속도 변화량에 따라 감소 가중치(dw)를 다르게 정의한다.

표 3. 예측 속도 변화량에 따른 감소 가중치

ΔPS_T^t	dw
6	0.8
7	0.7
8	0.6
9	0.5
10	0.4
11	0.3
12	0.2
13	0.1

[그림 5]는 감소 구간 가중치 적용 알고리즘을 나타낸다. 두 가지 속도 감소 구간 판단 기준을 만족한다면 [표 3] 따라서 감소 가중치를 변경시킨다. 1차 예측 속도에 감소 가중치를 곱하여 1차 예측 속도를 보정하고 카운트를 증가시킨다. 카운트 1은 5분을 의미하며, 30분 이후에는 이미 속도 흐름의 추세를 반영할 수 있으므로 감소 가중치를 높여준다. 감소 가중치 1은 의미가 없는 값이므로 감소 가중치 반영을 중단한다. 감소 구간 판단 기준에 만족하더라도 회복 구간 판단 기준에 만족한다면 감소 가중치 반영을 중단하고 회복 가중치 적용 알고리즘을 수행한다.

Algorithm 1. Decrease Interval Weighting Algorithm

```

Notation :
    Speed Decrease Criteria1 = 6;
    Speed Decrease Criteria2 = 10;
    Decrease Weight (dw) = 0.8;
    Count = 1;
Input :  $PS_T^t, \Delta PS_T^t, \Delta HS_T^t$ 
Output :  $PS_T^{t'}$ 
.....
if  $\Delta PS_T^t > \text{Speed Decrease Criteria1}$ 
    and  $\Delta HS_T^t > \text{Speed Decrease Criteria2}$ 
then
    check_recovery_criteria;
    if  $\Delta PS_T^t > \Delta PS_T^{t-1}$  then
         $\text{switch}(\text{int}(\Delta PS_T^t)) \begin{cases} \text{case7} : dw = 0.7 \\ \text{case8} : dw = 0.6 \\ \text{case9} : dw = 0.5 \\ \text{case10} : dw = 0.4 \end{cases}$ 
        Count = 1;
    end if
     $PS_T^{t'} = PS_T^t * dw;$ 
    Count++;
end if
if Count == 6 then
    dw = dw+0.1;
    Count = 1;
end if
if dw == 1 then
    break;
end if
return  $PS_T^{t'}$ 
    
```

그림 5. 감소 구간 가중치 적용 알고리즘

속도 회복 구간을 판단하기 위해 예측 속도의 변화량과 속도의 관계를 분석하였다. [그림 6]은 예측 속도 변화량과 속도의 관계를 나타낸다. 붉은색 막대그래프를 보면, 1차 예측 속도 변화량이 급격히 10이상 증가했음에도 불구하고 완만한 곡선으로 속도가 낮아진다. 파란색 막대그래프를 보면 예측 속도 변화량이 11까지 치솟기 전에 실제 속도는 이미 음의 방향에서 양의 방향으로 전환되며 회복 구간을 형성하였다. 이는 과거 실제 속도 변화량인 ΔHS_T^t 가 양수에서 음수로 전환된 것과 같은 의미이다. 이처럼 이벤트가 발생한 경우, 도로의 속도는 급격히 감소했다가 회복하는 패턴을 보인다. 속도가 감소하는 구간에서는 과거 실제 속도 변화량이 양수이지만, 속도가 증가하는 구간에서는 음수로 나타

난다. 따라서 ΔHS_T^t 가 양수에서 음수로 전환되는 시점을 회복 구간으로 판단한다.

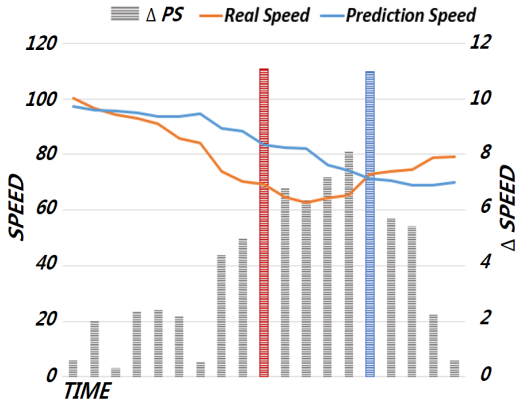


그림 6. 예측 속도 변화량과 속도의 관계

속도 회복 구간으로 판단되면 회복 가중치(rw)를 적용하여 속도를 보정한다. 회복 가중치는 감소 가중치와 반대로 예측 속도의 20% 증가한 값으로 보정할 수 있도록 1.2로 정의한다. [그림 7]은 회복 구간 가중치 적용 알고리즘을 나타낸다. ΔHS_T^t 가 음수로 전환되면 회복 구간으로 판단하고, 회복 가중치가 1보다 큰 경우에 1차 예측 속도에 회복 가중치를 적용하여 속도를 보정한다. 30분 후에는 1차 예측 속도가 회복 구간의 추세를 반영할 수 있으므로 가중치를 0.1씩 낮춰주며 적용한다.

Algorithm 2. Recovery Interval Weighting Algorithm

Notation :

Recovery Weight (rw) = 0.2;

Count = 0;

Input : PS_T^t

Output : $PS_T^{t'}$

if Count == 6 **then**

$rw = rw - 0.1$;

Count = 0;

end if

if $HS_T^t < 0$ and $rw > 1.0$ **then**

$PS_T^{t'} = PS_T^t * rw$;

Count++;

end if

return $PS_T^{t'}$

그림 7. 회복 구간 가중치 적용 알고리즘

IV. 성능 평가

제안하는 도로 속도 예측 기법과 기존 기법의 성능 비교 평가를 통해 제안하는 기법의 우수성을 입증한다. [표 4]는 성능 평가 환경을 보여준다. [표 5]는 성능 평가에 사용한 데이터를 나타낸다. 성능평가는 Intel core i5-4440 @ 3.10GHz, 8.00GB RAM, Windows 7 Ultimate K 64-bit 환경에서 실험 평가를 진행하였다. 제안하는 기법은 Python Anaconda custom 환경에서 Python 언어와 Keras 라이브러리를 이용하여 구현하였다. 성능평가에 사용된 속도 데이터는 한국도로공사 고속도로 공공데이터 포털에서 제공하는 VDS 수집체계로부터 획득한 구간 통행속도 자료를 수집하였다. 성능평가를 위해 2019년 06월 24일 월요일부터 9월 1일 일요일까지 수집된 속도 데이터를 학습 데이터 집합으로 사용하였으며 2019년 9월 2일 월요일부터 2019년 10월 6일 일요일까지 수집된 속도 데이터를 예측 데이터 집합으로 사용하였다.

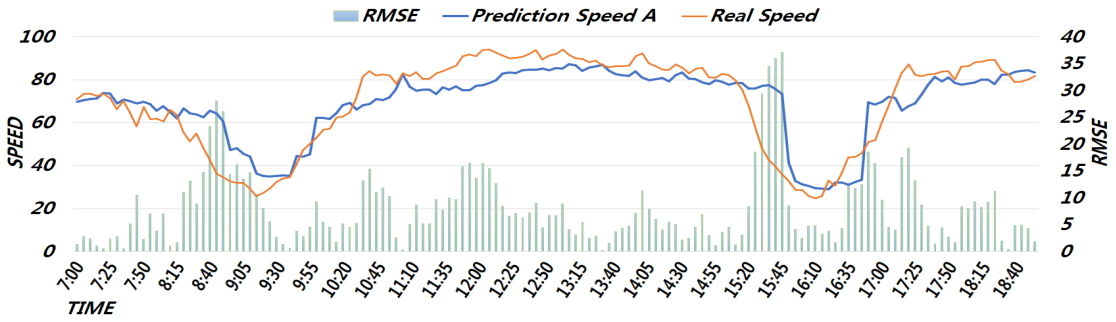
표 4. 성능 평가 환경

구분	내용
프로세서	Intel(R) Core(TM) i5-4440K 3.10GHz 4 Core
메모리	8.0 GB
운영체제	Windows 7 Ultimate K 64bit
사용 언어	Python 3
사용 플랫폼	Python3.5.6 Anaconda custom

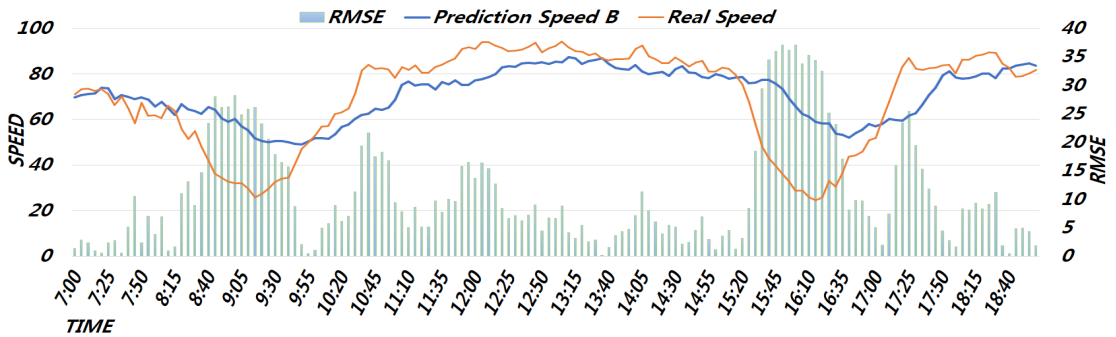
표 5. 성능 평가에 사용한 데이터

구분	수집 기간	크기
학습 데이터 집합	2019.06.24-2019.09.01	20,160 건
예측 데이터 집합	2019.09.02-2019.10.06	10,080건

제안하는 도로 속도 예측 기법의 성능평가는 크게 자체 성능평가와 기존 기법과의 비교 성능 평가로 이루어진다. 자체 성능평가로 제안하는 기법에서 고려한 교통 돌발 상황 시 이벤트 가중치를 적용했을 때와 적용하지 않았을 때의 결과를 비교하여 제안하는 기법의 유효성을 입증한다. 기존 기법과의 비교 성능 평가를 위해



(a) 이벤트 가중치를 적용한 예측 속도



(b) 이벤트 가중치를 적용하지 않은 예측 속도

그림 8. 이벤트 가중치 반영에 따른 예측 속도

RNN을 이용한 도로 속도 예측과 성능 비교를 수행한다. 본 논문에서는 이 기법을 RNN-P 기법이라 표기한다. 본 논문에서는 예측 정확도를 평가하기 위해 예측한 속도와 실제 속도를 기반으로 RMSE를 계산하였다. RMSE 값이 작을수록 예측 정확도가 높음을 의미한다.

제안하는 기법에서는 교통 돌발 상황이 발생한 경우 예측 속도에 이벤트 가중치를 적용하여 도로 속도 예측의 정확도를 높인다. 속도 변화량을 통계 분석하여 속도가 급변할 때 발생하는 감소-회복 패턴의 판단 기준을 정의하였으며 감소 가중치와 회복 가중치를 정의하였다. 본 논문에서는 이벤트 가중치를 반영하였을 때의 결과와 반영하지 않았을 때의 결과를 비교하여 자체적인 성능평가를 수행하였다. 이벤트 가중치를 반영하지 않은 결과는 1차 속도 예측 단계까지 수행한 제안 기법을 이용한다. 2019년 9월 9일 오전 7시 부터 오후 7시 까지의 예측 속도 결과를 이용하였다.

[그림 8]는 이벤트 가중치 적용 여부에 따른 예측 속도를 나타낸 것이다. [그림 8](a)는 이벤트 가중치를 적

용한 예측 속도(Prediction Speed A)와 실제 속도, 그리고 RMSE 값을 나타낸다. Prediction Speed A는 속도 감소 구간을 판단하여 감소 가중치를 적용함으로써 급격히 감소하는 도로 속도 구간을 높은 정확도로 예측하고 있으며, 속도 회복 구간 또한 회복 가중치를 적용함으로써 높은 정확도로 예측하고 있다. [그림 8](b)는 이벤트 가중치를 적용하지 않은 예측 속도(Prediction Speed B)와 실제 속도, 그리고 RMSE 값을 나타낸다. Prediction Speed B는 30분 전의 속도 데이터가 가장 큰 영향을 미치기 때문에 도로 속도가 급감하는 구간과 회복하는 구간 모두 뒤늦게 따라가는 흐름을 보인다. 또한, 실제 속도가 급격히 감소하는 구간은 실제 속도의 흐름과 다르게 완만한 흐름으로 예측하였다. 과거 데이터들을 학습하여 예측하기 때문에 일반적이지 않게 속도가 급격히 감소하는 경우, 그 흐름을 반영하려고 하지만 상대적으로 매우 완만하게 예측하여 예측 정확도가 감소한다.

[그림 9]는 이벤트 가중치를 적용한 예측 속도와 실

제 속도로 계산한 평균 RMSE(A) 값과 이벤트 가중치를 고려하지 않은 예측 속도와 실제 속도로 계산한 평균 RMSE(B)값을 나타낸다. 도로 속도 급감 구역이 발생한 두 구간을 구분하여 각 평균 RMSE 값을 계산하였다. Section1은 오전에 발생한 속도 변화 구간을 의미하고, Section2는 오후에 발생한 속도 변화 구간을 의미한다. 오전 구간의 A는 11.80으로 72.11인 B에 대비하여 RMSE 값이 60.31 감소하여 예측 오차를 크게 줄였으며, 오후 구간 또한 A는 14.67로 23.00인 B에 대비하여 RMSE 값이 8.33 감소하여 두 구간 모두 높은 정확도로 예측하였다.

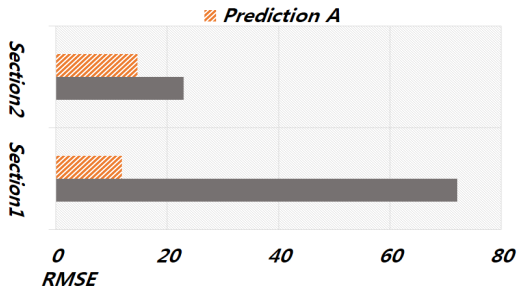


그림 9. 이벤트 가중치 반영에 따른 RMSE 비교

제안하는 기법에서 1차 속도 예측을 위한 학습 도구로 사용한 LSTM의 우수성을 입증하기 위해 RNN을 이용한 도로 속도 예측 기법과 성능 비교를 수행한다. RNN은 시간의 연결성을 고려하여 시계열 예측이 많이 활용되는 기계 학습 기법이다. 하지만 RNN은 시간 격차가 커지면 초기 데이터의 영향이 사라지는 기울기 손실 문제가 발생한다. 따라서 제안하는 기법에서는 기울기 손실 문제를 극복하기 위해 고안된 LSTM을 1차 속도 예측을 위한 학습 도구로 채택하였다. [그림 10]은 RNN을 이용한 예측 속도의 RMSE(RNN-P) 값과 LSTM을 이용한 예측 속도의 RMSE(LSTM-P) 값을 나타낸다. 2019년 9월 8일부터 9월 14일까지 하루 단위로 예측 속도의 RMSE 평균을 계산하였다. 월요일과 화요일을 제외한 모든 날의 LSTM-P로 예측한 속도의 RMSE 값이 낮으며, 평균적으로도 더 높은 정확도를 보임으로써 제안하는 기법의 우수성을 입증하였다.

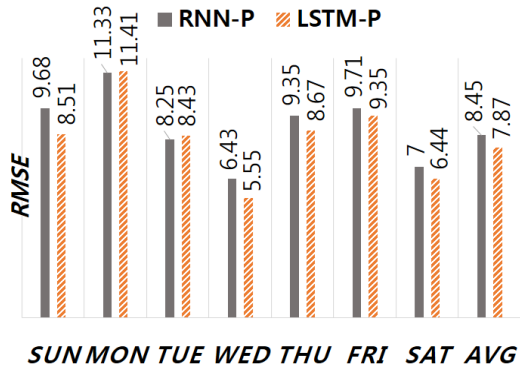


그림 10. RNN과 LSTM 비교

V. 결론

본 논문에서는 교통 돌발 상황을 고려한 도로 속도 예측 기법을 제안하였다. 제안 기법은 이웃 도로의 영향을 고려하기 위하여 이웃 도로들로부터 수집한 속도 데이터를 이용하여 학습을 통해 최적의 가중치를 갖는 LSTM 모델을 제안하였다. 이웃 도로들의 영향을 반영하여 1차 속도를 예측하고 교통 돌발 상황 발생 시 도로의 일반적인 흐름이 깨지면서 예측 오차가 커지는 문제를 보완하기 위해 이벤트 가중치를 적용하였다. 제안하는 기법은 한국도로공사와 같이 실시간 도로 상황을 제공하는 서비스나 속도 변화량 분석을 통해 사고 상황을 신속하게 알릴 수 있는 애플리케이션 등 다양한 교통 서비스 분야에 적용할 수 있다. 일반 도로의 경우 차량의 진출입이 구간이 다양하고 도로 속도에 영향을 미치는 요소가 다양하다. 제안 기법은 차량 진출입이 제한적인 고속 도로 환경에서 속도를 예측하는 기법으로 일반 도로에 적용하기 위해서는 속도에 영향을 미치는 다양한 요소들을 고려하여 할 필요가 있다. 향후 연구로는 일반 도로 환경에서 도로 속도 예측을 위한 연구를 진행하고 다양한 도로 환경에서 제안 기법의 유효성을 검증할 예정이다.

참고 문헌

- [1] 이창현, “지속 가능한 도시교통 관리방안 연구,” 서울특별시, 2014.
- [2] 최기주, “교통정보 현황 분석 및 정확성 개선방안 연구,” 교통정보플랫폼포럼, 2015.
- [3] 유영중, 문상호, 박성호, “도심 도로의 속도 예측을 위한 KNN 알고리즘 분석,” 예술인문사회융합멀티미디어논문지, 제7권, 제2호, pp.245-253, 2017.
- [4] 유영중, 조미경, “베이지안 네트워크를 이용한 단기 교통정보 예측 모델,” 한국정보통신학회 한국정보통신학회논문지, 제13권, 제4호, pp.765-773, 2009.
- [5] M. Fouladgar, M. Parchami, R. Elmasri, and A. Ghaderi, “Scalable deep traffic flow neural networks for urban traffic congestion prediction,” Proc. International Joint Conference on Neural Networks, pp.2251-2258, 2017.
- [6] 박송희, 최도진, 복경수, 유재수, “타겟 및 이웃 도로의 교통 정보를 이용한 속도 예측 기법,” 한국정보과학회 2019 한국컴퓨터종합학술대회 논문집, pp.142-144, 2019.
- [7] R. Jia, P. Jiang, L. Liu, L. Cui, and Y. Shi, “Data driven congestion trends prediction of urban transportation,” IEEE Internet of Things Journal, Vol.5, No.2, pp.581-591, 2018.
- [8] Z. Liu, Z. Li, K. Wu, and M. Li, “Urban traffic prediction from mobility data using deep learning,” IEEE Network, Vol.32, No.4, pp.40-46, 2018.
- [9] F. H. Tseng, J. H. Hsueh, C. W. Tseng, Y. T. Yang, H. C. Chao, and L. D. Chou, “Congestion prediction with big data for real-time highway traffic,” IEEE Access, Vol.6, pp.57311-57323, 2018.
- [10] Z. Duan, Y. Yang, K. Zhang, Y. N, and S. Bajgain, “Improved deep hybrid networks for urban traffic flow prediction using trajectory data,” IEEE Access, Vol.6, pp.31820-31827, 2018.
- [11] H. Z. Moayedi and M. A. Masnadi-Shirazi, “Arima model for network traffic prediction and anomaly detection,” Proc. International Symposium on Information Technology, pp.1-6, 2008.
- [12] C. Zhou and P. C. Nelson, “Predicting traffic congestion using recurrent neural networks,” Proc. World Congress on Intelligent Transport Systems, 2002.
- [13] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, Vol.9, No.8, pp.1735-1780, 1997.
- [14] F. V. Jensen, “An introduction to Bayesian networks,” UCL Press, 1996.
- [15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Communications of the ACM, Vol.60, No.6, pp.84-90, 2017.
- [16] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, “Recurrent neural network based language model,” Proc. Annual Conference of the International Speech Communication Association, pp.1045-14048, 2010.
- [17] W. Zaremba, I. Sutskever, and O. Vinyals, “Recurrent neural network regularization,” CoRR abs/1409.2329, 2014.
- [18] K. Chen, Y. Zhou, and F. Dai, “A LSTM-based method for stock returns prediction: A case study of China stock market,” Proc. IEEE International Conference on Big Data, pp.2823-2824, 2015.
- [19] F. Altché and A. de La Fortelle, “An LSTM network for highway trajectory prediction,” Proc. International Conference on Intelligent Transportation Systems, pp.353-359, 2017.
- [20] <https://pandas.pydata.org/pandas-docs/stable/>

저 자 소 개

박 송 희(Song-Hee Park)

준회원



- 2018년 2월 : 한국교통대학교 컴퓨터공학과(공학사)
- 2020년 2월 : 충북대학교 정보통신공학과(공학석사)

〈관심분야〉 : 기계 학습, 빅데이터 처리, 분산 처리, 데이터 분석

최 도 진(Do-Jin Choi)

정회원



- 2014년 2월 : 한국교통대학교 컴퓨터공학과(공학사)
- 2016년 2월 : 한국교통대학교 컴퓨터공학과(공학석사)
- 2020년 2월 : 충북대학교 정보통신공학과(공학박사)

〈관심분야〉 : 연속 질의 처리, 그래프 스트림, 빅데이터 처리 등

복 경 수(Kyoung-Soo Bok)

종신회원



- 1998년 2월 : 충북대학교 수학과(이학사)
- 2000년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2005년 8월 : 충북대학교 정보통신공학과(공학박사)
- 2005년 3월 ~ 2008년 2월 : 한국

과학기술원 정보전자연구소 Postdoc

- 2008년 3월 ~ 2011년 2월 : 가인정보기술 연구소 차장
 - 2011년 3월 ~ 2019년 8월 : 충북대학교 전자정보대학 정보통신공학부 초빙교수
 - 2019년 9월 ~ 현재 : 원광대학교 SW융합학과 조교수
- 〈관심분야〉 : 데이터베이스 시스템, 이동 객체 데이터베이스, 이동 P2P 네트워크, 소셜 네트워크 서비스, 빅데이터 처리 등

유 재 수(Jae-Soo Yoo)

종신회원



- 1989년 2월 : 전북대학교 컴퓨터공학과(공학사)
- 1991년 2월 : 한국과학기술원 전산학과(공학석사)
- 1995년 2월 : 한국과학기술원 전산학과(공학박사)
- 1995년 2월 ~ 1996년 8월 : 목포대학교 전산통계학과 전임강사
- 1996년 8월 ~ 현재 : 충북대학교 전자정보대학 정보통신공학부 정교수

〈관심분야〉 : 데이터베이스 시스템, 멀티미디어 데이터베이스, 센서 네트워크, 바이오 인포메틱스, 빅데이터 처리 등