효율적 환승을 위한 버스도착시간의 예측1)

변세정*, 이정훈*
*제주대학교 전산통계학과
rha4578@gmail.com, jhlee@jejunu.ac.kr

Prediction of Bus Arrival Time for Efficient Transit Planning

Sejung Byun*, Junghoon Lee*
*Dept. of Computer Science and Statistics, Jeju National University

요 약

본 논문에서는 제주시에서 오픈데이터로 공개한 버스탑승 기록을 기반으로 이용도가 높은 버스노선에 대해 특정정류장에서의 도착시간 예측모델을 구축한다. 버스들의 평균주행 속도, 운행시간대, 교통량 등을 입력으로 한 모델을 Sklearn을 이용하여 생성하고 MAE와 손실율 등의 성능을 분석한다.

1. 서론

지하철과 같이 높은 예측가능성을 갖는 교통수단이 보급되지 않고 공용버스 등이 핵심적인 역할을 담당하는 중소도시에서는 효율적인 환승이 대중교통의 편리성에 가장 중요한 요소가 된다. 그러나 버스의 운행은 교통 상황, 승객이동 패턴 등 다양한 요소에 의해 영향을 받기 때문에 특정 정류장에의 버스 도착 시간에 편차가 생긴다. 예측의 불확실성으로 인해, 버스간 배차 간격이 큰 경우에는 환승 계획을 작성하는데 큰 어려움이 있게 된다.

반면 최근, 특히 제주시지역에서, 버스들을 차량 Wifi를 통해 승객들에게 인터넷 서비스를 제공하면서 승객들의 승하차 데이터를 수집하고 있어서 이 데이터에 기반한 다양한 부가가치 창출이 가능하다 [1]. 해당 데이터들이 제주데이터허브를 통해 일반인들에게도 공개되고 있어서 이를 기반으로 한 버스 도착시간 예측 모델을 개발할 수 있다. 일반적으로 도착시간은 동일 구간의 이전 버스가 특정 정류장에 도착한 시간을 토대로 다음 정류장에 도착하는 시간을 예측하지만, 본 논문에서는 이와 아울러 이동시간에 영향을 주는 운행시간, 속도 등의 요소를 결합시킨다.

2. 예측 모델의 구축

먼저 제주데이터허브를 통해 취득한 버스 탑승 기록 데이터는 (그림 1)에서 보는 바와 같이 각 탑승마다의 승객정보(개인식별제외), 버스노선, 승하차 시각, 정류장, 도로노드 등의 필드들을 가지고 있으며 월별로 대략 66만개의 레코드들을 보유하고 있는데 본 논문에서는 2019년 3월 데이터를 사용한다.

	A	B	C	D	E	F	G
1	base_date	route_id	geton_datetime	geton_static	getoff_datetime	getoff_static	times
2	20180718	29990000	20180718205545	3271	20180718211623	150	21
3	20180718	29990000	20180718205605	3271	20180718211623	150	20
4	20180718	29990000	20180718205625	3271	20180718210639	150	10
5	20180718	23280000	20180718132805	3271	20180718133408	150	6
6	20180718	24190000	20180718155000	3271	20180718160408	150	14
7	20180718	24190000	20180718155013	3271	20180718160200	150	12
8	20180718	23280000	20180718114235	3271	20180718121805	150	36
9	20180718	23280000	20180718114239	3271	20180718121457	150	32
10	20180718	23000000	20180718152355	3271	20180718152844	150	5
11	20180718	23000000	20180718152359	3271	20180718154653	150	23
12	20180718	23000000	20180718152402	3271	20180718152822	150	4
13	20180718	24190000	20180718110739	3271	20180718112404	150	17
14	20180718	23000000	20180718213743	3271	20180718220843	150	31
15	20180718	23580000	20180718195012	3271	20180718195624	150	6
16	20180718	23580000	20180718131512	3271	20180718134030	150	25
17	20180718	30240000	20180718182614	3271	20180718184211	150	16
18	20180718	23580000	20180718131519	3271	20180718135113	150	36
19	20180718	23580000	20180718131520	3271	20180718133515	150	20
20	20180718	23580000	20180718131522	3271	20180718135351	150	38
21	20180718	23610000	20180718152053	3271	20180718152935	150	9

(그림 1) 데이터 추출 결과

(그림 2)는 본 논문에서 중점적으로 분석하는 노선 구간이며 이에 포함된 정류장들에서 버스들의 도착시간을 예측하는 모델을 구축한다. 그러나, 공개된 승하차 기록 레코드만으론 버스 노선, 정류장 아이디와 노선구간의 번호와 이들의 상세정보는 연계되어 있지 않다. 따라서 일부 관심 노선에 대해서는 수동적으로 데이터를 추출해야 한다. (그림 2)는 정류장이 3271, 150인 데이터에 대하여 추출한 경우이다. 3271은 제주시청->광양사거리 정류장이고, 150은 시외버스 터미널 정류장이다. 시간예측을 위한

본 연구는 R-WeSET사업의 지원을 받아 수행되었음.

교통량 데이터를 승하차 레코드와 정합하여야 하는데 차로를 중심으로 해당 구간을 나타내기 때문에 구간에 맞는 교차로를 찾고 해당 교차로의 노드, 도로 노드를 각각 대입하면서 검색한 결과 3개의 부구간으로 나뉜다.

(그림 2) 관심 구간

결과적으로 이를 종합하여 버스노선, 날짜, 걸린 시간, 출발 시각, 교통량, 속도 등을 입력으로, 이에 따른 도착시간을 출력으로 갖는 예측 모델을 구축한다. 이 과정에서 Python sklearn을 사용하여 기계학습을 수행하는데 Colab에 학습 데이터를 업로드하고 Colab에서 제공하는 기계학습 라이브러리를 활용한다[2]. 모델 구축에서 각 계층에 대해 걸리는 시간을 예측하여야 하기 때문에 (activation, classification) 조합으로 (sigmoid, softmax) 대신 (ReLU, tanh)을 선택하였다.

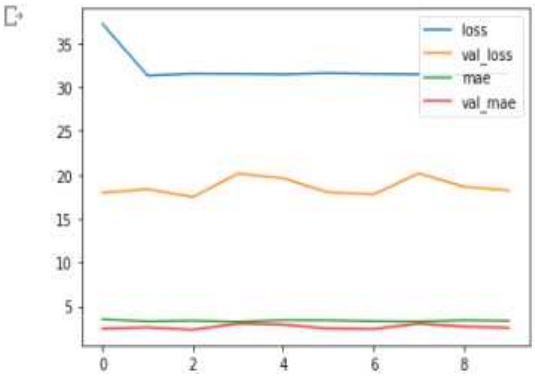
```
#4. 모델 학습
model.compile(optimizer="sgd", loss='mse', metrics=['mae'])
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 120)               600
_________________________________________________________________
dense_1 (Dense)              (None, 60)                7260
_________________________________________________________________
dense_2 (Dense)              (None, 15)                915
_________________________________________________________________
dense_3 (Dense)              (None, 1)                 16
=================================================================
Total params: 8,791
Trainable params: 8,791
Non-trainable params: 0
_________________________________________________________________
```

(그림 3) 학습 모델 환경 설정

모델의 세부구성은 (그림 3)에서 보는 바와 같이 손실함수를 평균제곱오차(MSE; Mean Square Error)로 두고, 이 함수의 값들이 최소가 되도록 하였다. 그 결과, 입·출력 레이어를 포함한 전체 레이어의 개수는 4개, SGD(Stochastic Gradient Decent) 옵티마이져를 선택한 후 epochs을 10으로 두어 학습시켰다.

생성된 예측모델에 대해 학습에 사용되지 않은 10%의 데이터를 모델에 입력하고 그 결과값과 실제값을 비교하여 의해 모델의 정확성을 평가한다. (그림 4)는 모델 학습결과에 있어서 전체 손실, MAE, 검증 손실, 검증 MAE를 보이고 있다. 샘플링된 데이터에 따라서 해당 평가손실, 평가 MAE 값이 달라진다. 학습결과 손실은 30 정도, MAE 값은 4~2 분 정도를 보이고 있다.



(그림 4) 예측 모델 평가

결과에 의하면 평균 9분 정도의 운행시간에 대해 약 1.5~0.5 분까지도 오차가 발생하므로 아직은 고정밀도의 예측 모델에 적용하기에는 부족한 면이 있다. 이를 개선하기 위해서는 1년 데이터로 확장하고 요일, 계절, 관광객 수 등의 인자를 추가적으로 고려하는 것이 바람직하다.

3. 결론

버스 교통량 예측의 정확성이 향상된다면, 복수의 버스 노선들을 조합하여 환승 지점을 고려한 경로 추천이 가능하다. 다익스트라 알고리즘을 용해서 각 노드를 정류장으로 하고 각 연결선을 노선 경로로 한 구조에서 각 데이터의 가중치를 예측시간으로 놓음으로 최단거리를 찾는 결과 환승 지점을 찾을 수 있게 된다. 또한, 해당하는 데이터를 공개함으로써 앞으로 교통데이터와 관련된 다양한 분야에 쓰일 수 있다.

참고문헌

[1] https://www.jejudatahub.net/
[2] https://scikit-learn.org/stable/