공공빅데이터 융합DB 기반 XAI를 활용한 지하철 내외부 혼잡도 예측 성능 개선 및 설명가능한 비즈니스 의사결정

Jaeheung Parka, Kyungwon Kima,*∗*

*aSchool of International Trade and Business, Incheon National University, Incheon, 22012, Republic of Korea*

# Abstract

*Background:* Korea has the highest suicide rate among Organisation for Economic Co- operation and Development (OECD) countries. Consequently, central and local governments and private organizations in Korea cooperate in promoting various suicide prevention projects to actively respond to suicide problems. Machine learning has been used to predict suicidal ideation in the fields of health and medicine but not from a social science perspective.

*Objective:* Since suicidal ideation is a major predictor of suicide attempts, being able to

anticipate and mitigate it helps prevent suicide. Therefore, this study presents a data-based analysis method for predicting suicidal thoughts quickly and effectively and suggests countermeasures against the causes of suicidal thoughts.

*Participants and Methods:* To predict early signs of suicidal ideation in children and adolescents, big data collected for approximately 4 years (from 2017 to 2020) from the Korea Youth Policy Institute (NYPI) were used. To accurately predict suicidal ideation, supervised ma- chine learning classification algorithms such as logistic regression, random forest, XGBoost, multilayer perceptron (MLP), and convolutional neural network (CNN) were used.

*Results:* Using CNN, suicidal ideation was predicted with an accuracy of approximately

90%. The logistic regression results showed that sadness and depression increased suicidal thoughts by more than 25 times, and anxiety, loneliness, and experience of abusive language increased suicidal thoughts by more than three times.

*Conclusions:* Machine learning and deep learning approaches have the potential to predict

and respond to suicidal thoughts in children, adolescents, and the general population, as well as help respond to the suicide crisis by preemptively identifying the cause.

*Keywords:* Subway Congestion, Explainable Prediction, Machine and Deep Learning, SHAP, Business Decision

*∗*Corresponding author: thekimk.kr@gmail.com

*ORCID(s):* 0000-0001-6530-8426 (K. Kim)

# Introduction

글로벌 시장조사업체 스태티스타의 조사(2023)[[1]](#footnote-1)에 따르면 전세계에서 대중교통 이용률이 가장 높은 나라는 대한민국이라고 한다. 대한민국은 관련 조사에서 ‘통근, 통학 시 대중교통을 이용한다’고 41%가 응답하였다. 특히 지하철은 대중교통 중 가장 높은 30% 수단분담률을 기록하고 있다.[[2]](#footnote-2) 이처럼 다수가 이용하는 대표적인 대중교통 수단인 지하철의 혼잡도를 파악하는 것은 운영 효율성과 편의성 측면 등에서 중요한 의미를 갖는다.

지하철 혼잡도를 파악하는 것은 지하철 이용 패턴과, 승객 흐름을 예측하는 데 중요한 정보를 제공한다. 이를 통해 운행 빈도, 열차 편성, 인력 배치 등 운영 전략을 효과적으로 수립할 수 있다. 또한, 혼잡도 파악은 안전성 확보와 승객 경험 개선에 핵심적 역할을 한다. 혼잡도가 과도하게 높을 경우 승객들의 이동이 제한되어 사고 위험이 증가하고, 응급 상황 발생 시 신속한 대응이 어려워질 수 있다. 따라서 혼잡도를 낮추기 위한 노력은 승객 안전성을 높이는 데 중요한 기여를 한다. 나아가 혼잡도 분석은 도시 계획 및 교통 정책 수립에 있어서도 중요한 참고 자료로 활용될 수 있다. 해당 분석을 통해 특정 구간 및 시간대의 수요에 대비한 장기적인 인프라 계획을 수립하고, 지속 가능한 도시 교통 시스템을 구축하는 데 기여할 수 있다.

이러한 지하철 혼잡도는 크게 두 가지로 분류하여 분석할 수 있다. 첫 번째는 객차 내 혼잡도이다. 서울교통공사에 따르면, 객차 내 혼잡도는 정원 대비 승차인원으로, 승차인과 좌석수가 일치할 경우 혼잡도 34%로 산정한다고 한다. 두 번째는 승강장 내 혼잡도이다. 본 연구에서는 승강장 혼잡도에 관한 예측을 진행하고자 한다. 이호, 최진경(2015)에서는 승강장 혼잡도를 승강장 면적 대비 승차대기인원으로 정의하고, 승차대기인원은 열차 탑승을 위해 대기하는 인원으로 정의하였다. 해당 연구에서 승차 대기인원을 산정하기 위해 열차출발정보와 승강장 누적진입인원을 활용한다. 현재 열차 승차를 위한 승강장 누적진입인원은 현재열차가 출발하기 전까지 승강장 누적진입인원에서 이전열차가 출발하기 전까지 승강장 누적진입인원의 차이다. 이상준 외 3명(2021)에서는 승강장 혼잡도를 승강장 내 승객수와 승객점유면적의 비율로 정의한다. 해당 연구에서는 승강장 내 승객이 실제 이용하는 대기 면적 개념을 정의하고 산정식으로 제안하였으며, 승차승객 대기면적계수를 제시하였다. 승차승객 대기면적계수의 경우, 상대식 승강장은 0.291, 섬식 승강장은 0.270으로 분석하였다. 이상준과 신성일(2020)에서는 승강장 동적 혼잡도를 다음과 같이 산출하였다. 1분 단위 승객 통행량과 승강장 통행량을 산출하고 승강장별 실용대기면적을 나누어 혼잡도를 추정하였다.

권효승 외 5명(2020)에서는 GRU 모델을 통해 지하철 구간 혼잡도를 예측하였다. 해당 연구는 동대문역사문화공원에서 을지로4가 사이의 혼잡도를 30분 단위로 예측하였고, 그 결과로 loss값 0.0431이 도출되었다. 이는 지하철 혼잡도를 GRU 기법으로 예측 모델을 만들었다는 점에서 의의가 있다. 장진영 외 2명(2023)에서는 서울교통공사에서 제공하는 혼잡도 정보를 가공하여 머신러닝 알고리즘으로 분석하였다. 혼잡도가 130% 이상일 경우 혼잡, 그 미만이면 비혼잡으로 분류하여 로지스틱 회귀, 의사결정나무, 랜덤 포레스트로 분석하였다. 해당 연구는 500m 이내 버스정류장 수, 역세권 건물 수 등 혼잡도 예측에 있어 새로운 지표를 사용했다는 점에서 의의가 있다. 김규진 외 3명(2024)에서는 다층 퍼셉트론(Multi-Layer Perceptron, MLP)을 통해 시간대별 혼잡도를 예측하였다. 해당 연구는 시간대별 특성과 기상 조건을 고려하여 지하철역 혼잡도를 예측한 점에서 의의가 있다.

기존 머신러닝과 딥러닝 모델은 예측값을 제공하지만, 그 예측에 대한 근거를 사람이 이해하기 어렵다는 한계를 가지고 있다. 이러한 한계를 해결하기 위해 설명가능한 인공지능(Explainable AI, XAI)이라는 개념이 등장하였다. 설명가능한 인공지능은 모델의 동작 방식과 최종 결과를 사람이 이해할 수 있는 방식으로 설명하는 것을 목표로 한다. 이를 통해 사용자는 모델이 어떤 특성과 패턴을 기반으로 결정을 내렸는지 알 수 있으며, 이로 인해 불확실성이 해소되어 신뢰성을 높일 수 있다. 설명가능한 인공지능은 단순히 모델의 성능을 넘어서, 인간과 인공지능 간의 상호작용을 향상시키고, 다양한 분야에서 AI의 책임 있는 사용을 가능하게 한다는 점에서 그 중요성이 부각되고 있다.

본 연구는 지하철 관련 공공 빅데이터를 최대한 통합하여 분석하는 것을 목표로 한다. 다양한 공공 데이터를 융합함으로써 지하철 혼잡도 예측의 정확도를 높이고자 하였다. 특히, 경제적 연관성을 반영하기 위해 경제 관련 데이터를 통합하여, 지하철 혼잡도와 경제적 요인 간의 상관성을 분석하였다. 또한, 본 연구에서는 실제 의사결정에 활용할 수 있을 만큼의 혼잡도 예측 성능을 목표로 하였다. 이를 통해 교통공사뿐만 아니라 여러 관계자들이 실제로 사용할 수 있는 도구로서 활용할 수 있는 예측 모델을 제안하고자 한다. 이러한 모델은 다양한 시간대와 노선, 그리고 여러 기상 상황에서도 혼잡도를 효과적으로 예측함으로써 효율적인 자원 배분과 운영 계획 수립을 지원할 수 있다.

아울러, 본 연구는 설명가능한 인공지능(Explainable AI, XAI)을 활용하여 모델의 예측 결과를 보다 쉽게 이해할 수 있도록 하였다. 설명가능한 인공지능을 통해 혼잡도 예측 모델의 결정 과정과 근거를 시각적으로 제공함으로써, 실제 비즈니스와 정책의 의사결정 과정에서 보다 투명하고 신뢰할 수 있는 의사결정을 지원한다. 또한, 본 연구에서는 고성능 PC 없이도 누구나 현장에서 쉽게 사용할 수 있는 기계학습 및 딥러닝 기반 비즈니스 애널리틱스 프레임워크을 제시하고자 한다. 이를 통해 누구나 현장에서도 손쉽게 모델을 적용하고 활용할 수 있도록 설계되었으며, 이는 실제 운영 환경에서의 실용성을 극대화하는 것을 목표로 한다.

# Methods

* 1. *Participants and Data Preprocessing*

[데이터 출처와 양 설명]

한국은 정보화 시대의 도래와 행정의 투명성 그리고 사회적 활용 효율성을 높이려는 노력으로 공공데이터를 수집하기 시작하였다. 이를 통해 국민들의 정보 접근성이 높아졌으며 데이터 기반 정책을 수립하고 민간에서도 창의적인 활용을 할 수 있게 되었다. 1990년대부터 대한민국 정부는 공공정보를 디지털화하려고 일부 데이터를 개방하기 시작하였으며, 2013년 “공공데이터의 이용 및 제공에 관한 법률”이 제정되면서 다양한 공공데이터가 개방되었다 (참고자료). 현재는 정부 뿐만 아니라 지방자치단체에서도 다양한 공공데이터를 생성 및 개방하고 있다. 향후 더욱 많은 공공데이터가 생성될 것이고 데이터 기반 의사결정이 강화될 것이기 때문에 이러한 공공데이터를 융합하여 사용하는 것이 필수적일 것이다. 따라서 본 연구에서는 기존 연구들에서 활용하지 않았던 공공데이터를 더욱 많이 융합하여 지하철 혼잡도의 예측 성능을 크게 개선하였다.

본 연구에서는 “공공 데이터 포털, 서울 열린 데이터 광장, 통계청 등” 총 8개의 포털에서 총 17개의 공공 데이터베이스(DB)를 통합하였다(표 1). 지하철의 역사정보와 사용 인원수에 관한 정보뿐만 아니라 지하철과 같은 주요 교통수단인 자동차의 등록대수, 주유소 판매가격, 그리고 경기와 관련된 지수, 실업률, 기준금리, 환율, 소비자물가, 지하철요금, 인구밀도 등의 정보들도 통합하여 데이터화 하였다. 각 데이터베이스의 수집 데이터의 기간은 2015년부터 2024년까지 다양하지만 공통적으로 추출될 수 있는 기간인 2021년부터 2023년까지 추출하였고, 데이터의 단위도 일단위로 통일하여 통합하였다.

표 1. 설명가능한 지하철 혼잡도 예측 성능 개선을 위해 활용한 공공 빅데이터 정보 및 변수활용 방향

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **데이터출처** | **데이터베이스(DB)명** | **기간** | **단위** | **추출변수** | **종속변수활용** |
| **공공데이터포털** | **한국천문연구원\_특일 정보** | 전체 | 일 | 연도, 월, 일, 요일, 공휴일, 대체공휴일 | - |
| **서울교통공사\_1\_8호선 역별 일별 시간대별 승객유형별 승하차인원** | 2015.01-2024.07 | 일 | 일별 승하차인원수, 우대권인원수, 청소년인원수 | 승강장혼잡도1(면적 대비 승하차인원) |
| **서울교통공사\_지하철혼잡도정보** | 2019-2024 | 분기 | 분기별 05시~23시 상하선 혼잡도(정원대비 승차인원) | - |
| **서울교통공사\_월별 환승유입인원** | 2019-2023 | 연 | 월별 환승유입인원수 | 승강장혼잡도2(면적 대비 승강장대기인원) |
| **서울교통공사\_역사운영 현황** | 2024.06.30 기준 | 연 | 역사운영현황(호선, 역명, 면적, 흥수, 승강장유형, 출입구수, 환승노선) | 승강장혼잡도1(면적 대비 승하차인원) |
| **서울교통공사\_수송순위** | 2019-2023 | 연 | 연도별 수송인원수 | 상위 30개역 필터링 및 종속변수 결측치 존재역 제외 |
| **서울열린데이터광장** | **서울시 지하철역 정보 검색 (역명)** | 2024.11.14 기준 | 연 | 역명 대응 역코드 | - |
| **서울시 역코드로 지하철역별 열차 시간표 정보 검색** | 2024.11.14 기준 | 연 | 역코드 대응 열차시간표, 시간대별 상하행 평균운행간격, 상하행 운행횟수 | 승강장혼잡도2(면적 대비 승강장대기인원) |
| **기상자료개방포털** | **종관기상관측(ASOS)** | 전체 | 시간 | 일시, 평균기온, 일강수량, 평균풍속, 평균상대습도, 일최심적설 | - |
| **Opinet** | **국내유가통계 > 주유소 > 평균판매가격** | 전체 | 일 | 일별 보통휘발유, 자동차용경유 가격 | - |
| **국토교통부 (시도별 자료)** | **국토교통부 > 자동차 등록 현황** | 전체 | 연 | 연도별 자동차 등록대수 | - |
| **통계청「산업활동동향」** | **기획재정부 > 경기종합지수** | 전체 | 월 | 월별 동행지수 순환변동치, 선행지수 순환변동치 | - |
| **통계청「경제활동인구조사」** | **기획재정부 > 취업자 수/실업률 추이** | 전체 | 월 | 월별 실업률, 청년실업률 | - |
| **통계청「장래인구추계 시도편 : 2022-2052」** | **통계청 > 지역별 인구 및 인구밀도** | 전체 | 연 | 연도별 서울/수도권 인구, 인구밀도 | - |
| **한국은행 경제통계시스템(ecos) -> 1.3 금리** | **금융위원회 > 시장금리 추이** | 전체 | 월 | 월별 기준금리 | - |
| **Investing.com** | **USD/KRW - 미국 달러 원** | 전체 | 일 | 일별 환율 | - |
| **한국 소비자물가지수(CPI)** | 전체 | 월 | 월별 소비자물가지수, CPI 대비 지하철요금 | - |

그리고 각 데이터베이스에서 추출할 수 있는 변수들을 생성하였고 필요에 따라서는 여러 개의 데이터베이스 변수들을 결합하여 생성하기도 하였다. 특히 종속변수인 혼잡도의 정의에 따라 여러 개의 데이터베이스를 결합하여 추출하여야 한다. 그리고 혼잡도가 낮은 경우보다 혼잡도가 높은 경우를 집중적으로 예측이 필요하기 때문에 수송인원수가 많은 상위 30개의 역을 필터링하고 종속변수로 활용하기에 결측치가 존재하는 경우를 제외하여 예측 대상역으로 선정하였다. 수집된 데이터의 양은 약 164만개의 샘플 데이터가 수집되었고 총 114개의 혼잡도 관련 변수가 추출되었다. 수집된 데이터의 지하철 노선은 1호선부터 8호선까지이며 2호선이 총 14개의 역이 포함되었고 37.01%의 가장 많은 샘플을 포함하고 있다. 다음으로 3호선이 총 8개의 역을 포함하며 20.19%의 두번째로 많은 샘플을 포함하고 있고, 6호선과 8호선이 가장 적은 1개의 역만을 포함하고 있다.

표 2. 통합된 공공 빅데이터 수송인원수 상위 30개 지하철 호선별 샘플수, 비율, 역명 정보

|  |  |  |  |
| --- | --- | --- | --- |
| **호선** | **샘플수** | **비율** | **역명** |
| **2** | 610,278 | 37.01% | 홍대입구, 시청, 신림, 을지로입구, 선릉, 역삼, 성수, 사당, 강남, 서울대입구, 신도림, 건대입구, 잠실, 합정 |
| **3** | 332,878 | 20.19% | 연신내, 신사, 종로3가, 양재, 압구정, 경복궁, 고속터미널, 안국 |
| **4** | 208,050 | 12.62% | 수유, 혜화, 명동, 사당, 서울역 |
| **1** | 166,436 | 10.09% | 종로3가, 서울역, 종각, 시청 |
| **7** | 124,830 | 7.57% | 가산디지털단지, 고속터미널, 건대입구 |
| **5** | 124,828 | 7.57% | 광화문, 여의도, 종로3가 |
| **6** | 41,610 | 2.52% | 합정 |
| **8** | 40,106 | 2.43% | 잠실 |

[종속변수 정의 및 비율]

본 연구에서 사용하는 종속변수는 승강장혼잡도로 “면적 대비 승하차인원”으로 정의된다. 총 약 164만개의 승강장 혼잡도의 평균값은 45.9131이며 최소 0.0101부터 최대 502.3289까지 분포되어 있다. 그림 1의 호선에 따른 분포를 비교해보면 가장 많은 역이 포함되어 있는 2호선의 혼잡도가 다른 호선들에 비해서 압도적으로 넓게 분포되어 있음을 알 수 있다. 이어서 3호선과 4호선이 2호선의 절반정도의 범위로 포진되어 있다. 또한 그림 1의 역에 따른 분포를 비교해보면 각 수치의 빈도에 따라 전체 혼잡도의 순위는 달라질 수 있지만, 전체 혼잡도를 기준으로 내림차순 했을 경우 강남역이 가장 넓은 혼잡도의 범위를 차지하고 있다. 다음으로 홍대입구와 잠실, 신림, 서울대입구 등 2호선 위주, 환승역 위주로 상위를 차지하고 있음을 알 수 있다. 그리고 승강장혼잡도를 제외한 나머지 113개의 변수들이 독립변수로 사용되어 승강장혼잡도를 예측하는 주요요인들로 활용된다.

|  |  |
| --- | --- |
|  |  |

그림 1. 종속변수인 승강장혼잡도의 호선별 그리고 역별 분포 비교

[독립변수 전처리]

본 연구는 공공데이터포털에서 제공하는 2021년부터 2023년 사이의 ‘서울교통공사\_1\_8호선 역별 일별 시간대별 승객유형별 승하차인원’을 중심으로 데이터를 가공하였다. 해당 데이터는 교통카드(선후불교통카드 및 1회용 교통카드)를 이용한 지하철 호선별 역별(1~9호선, 서울시 관할 운송기관에 한함) 시간대별 승하차인원을 나타내는 정보로 수송일자, 호선명, 역명, 승하차구분, 승객유형, 시간대별 승하차인원을 알 수 있다. 여기서 변수를 가공하여 시간대별 승하차인원의 합과 차, 시간대별 우대권인원수, 시간대별 청소년인원수 변수를 추가하였다. 그 후 공공데이터포털에서 제공하는 2021년부터 2023년 사이의 ‘서울교통공사\_지하철혼잡도정보’를 5시부터 23시까지 시간대별 상선 혼잡도와 하선 혼잡도로 분리하여 위 승하차인원(기준 데이터) 데이터에 병합하였다. 그 다음으로 공공데이터포털에서 제공하는 2021년부터 2023년까지의 ‘서울교통공사\_월별 환승유입인원’데이터를 역 및 월별 환승유입인원 변수를 기준 데이터에 병합하였다. 그 다음으로 공공데이터포털에서 제공하는 2024년 6월 30일 기준 서울교통공사의 1~8호선 역사운영 현황 데이터에서 역별(호선별) 면적, 출입구, 섬식여부, 환승노선 갯수를 추출해서 기준 데이터에 병합하였다. 이후 면적 변수와 승하차인원(합) 변수를 통해 승강장혼잡도(면적 대비 승하차인원) 변수를 만들었다. 기상자료개방포털에서 서울특별시 2021-2023 사이의 일시, 평균기온(°C), 일강수량(mm), 평균 풍속(m/s), 평균 상대습도(%), 일 최심적설(cm) 를 추출하여 기준 데이터에 붙였다. 날짜 정보의 경우, 위 데이터에서 수송일자를 활용하여 년도, 월, 일, 요일, 주 변수를 만들었고, holidayskr 라이브러리를 통해 공휴일 변수를 만들었다. 지하철요금 변수는 성인요금 기준으로 2023-10-01 이전에는 1250원, 이후는 1400원으로 적용하여 만들었다. USD-KRW 환율은 인베스팅닷컴에서 2021년부터 2023년까지의 데이터를 제공받아 기준 데이터에 붙였다. 결측치가 존재하는 날짜는 이전 날짜의 값으로 채웠다. 한국 CPI 데이터는 인베스팅닷컴에서 2021년부터 2023년까지의 데이터를 제공받아 기준 데이터에 붙였다. 여기에 추가로 CPI 대비 지하철 요금이라는 변수를 만들었다. 경우가격의 경우 오피넷에서 데이터를 다운받아 보통휘발유, 자동차용경우 일별 가격 데이터를 기준 데이터에 붙였다. 실업률 지표의 경우 통계청에서 제공하는 월별 실업률과 청년 실업률을 기준 데이터에 붙였다. 경기종합지수 지표 역시 통계청에서 제공하는 월별 동행지수 순환변동치와 월별 선행지수 순환변동치 데이터를 다운받아 기준 데이터에 붙였다. 인구 지표도 통계청에서 연도별 서울 인구와 서울 인구밀도, 수도권 인구, 수도권 인구밀도 데이터를 다운받아 기준 데이터에 붙였다. 자동차 등록대수는 국토교통부에서 제공하는 연도별 자동차 등록대수 데이터를 메인 데이터에 붙였다. 기준 금리는 한국은행 경제통계시스템(ecos)에서 제공하는 월별 기준 금리 데이터를 기준 데이터에 붙였다. 서울열린데이터광장에서 제공하는 서울시 역코드로 지하철역별 열차 시간표 정보 검색을 통해, 수송인원이 높은 30개 역 시간표 정보(상행, 하행 시간표/ 평일, 토요일, 공휴일(공휴일+일요일)로 구분)를 추출하였다. 이후 역별 및 시간대별, 상행\_평균운행간격, 하행\_평균운행간격, 상행\_운행횟수, 하행\_운행횟수 변수를 만들었다.

종속변수를 예측하기 위해 나머지 독립변수들을 사용하여 모델링을 하는데, 알고리즘이 이해할 수 있는 형태로 전처리 후에 반영해야 한다. 특히 머신러닝과 같은 인공지능 알고리즘을 사용하면 변수(질문)의 그룹핑 과정이 필요없이 각 질문마다 Loyalty에 어떤 영향을 미치는지 알 수 있는 장점이 있다. 물론 데이터 학습 과정에서 질문들의 상호작용(Interaction)도 반영이 되는건 물론이다. 단지 그러한 과정을 분석가가 굳이 할 필요가 없다. 이러한 장점을 반영하기 위해선 전처리 과정에서 불필요한 변수들을 삭제하는 것이 필수적이다. 그리고 문자로 된 응답값은 별도의 변수나 숫자로 변환하며 최종적으로 정리된 숫자들은 응닶값의 범위를 맞추기 위해 특정 범위로 스케일을 맞춘다(Scaling). 본 연구에서 활용한 데이터에서 새로운 종속변수를 만들기 위해 사용된 3개의 변수는 독립변수에 포함시키지 않기 위해 삭제되었다. 그리고 절반이상의 응답값이 없는 5개의 변수와 오리지 1개의 응답값만 포함하는 2개는 변수는 독립변수로 작동하기 어렵기 때문에 삭제되었다. 다음으로 문자값은 숫자변수로 변환하여 최종적으로 총 59개의 독립변수가 준비되었다. 그리고 독립변수마다 응답값의 범위가 차이가 있을 수 있는데, 응답의 범위가 넓으면 마치 중요한 변수인 것처럼 학습되는 것을 방지하고 특정 질문이 높은 중요도로 학습되는 왜곡을 줄이기 위해 0과 1사이의 범위로 맞추었다(Min-Max Scaling). 마지막으로 전체 330개의 samples를 8:2의 비율로 구분하여 Train and Test set으로 분리하였다. Train set은 모델링을 위해 학습에 사용되고 모델의 hyperparameters를 결정하기 위해 10-fold cross validation을 사용하여 AUC가 최대가 되는 방향으로 설정하였다. 그리고 Test set은 미래 데이터로 가정하여 학습된 모델이 미래에 얼마나 AI 스피커 Loyalty를 잘 예측하는지 평가하는데 사용된다. 기본적으로 고객의 Loyalty가 어떻게 설명되는지 독립변수의 가중치로 해석하는게 중요하더라도 그 해석이 미래 고객들의 Loyalty를 실제로 잘 예측하는 설명이어야 고객들에게도 정책적으로도 활용가치가 높을 것이다. 그러한 이유로 데이터를 분리하여 사용된다. 이해도를 높이기 위해 전체적인 전처리 과정을 아래 순서도로 제시한다 in Figure 1.

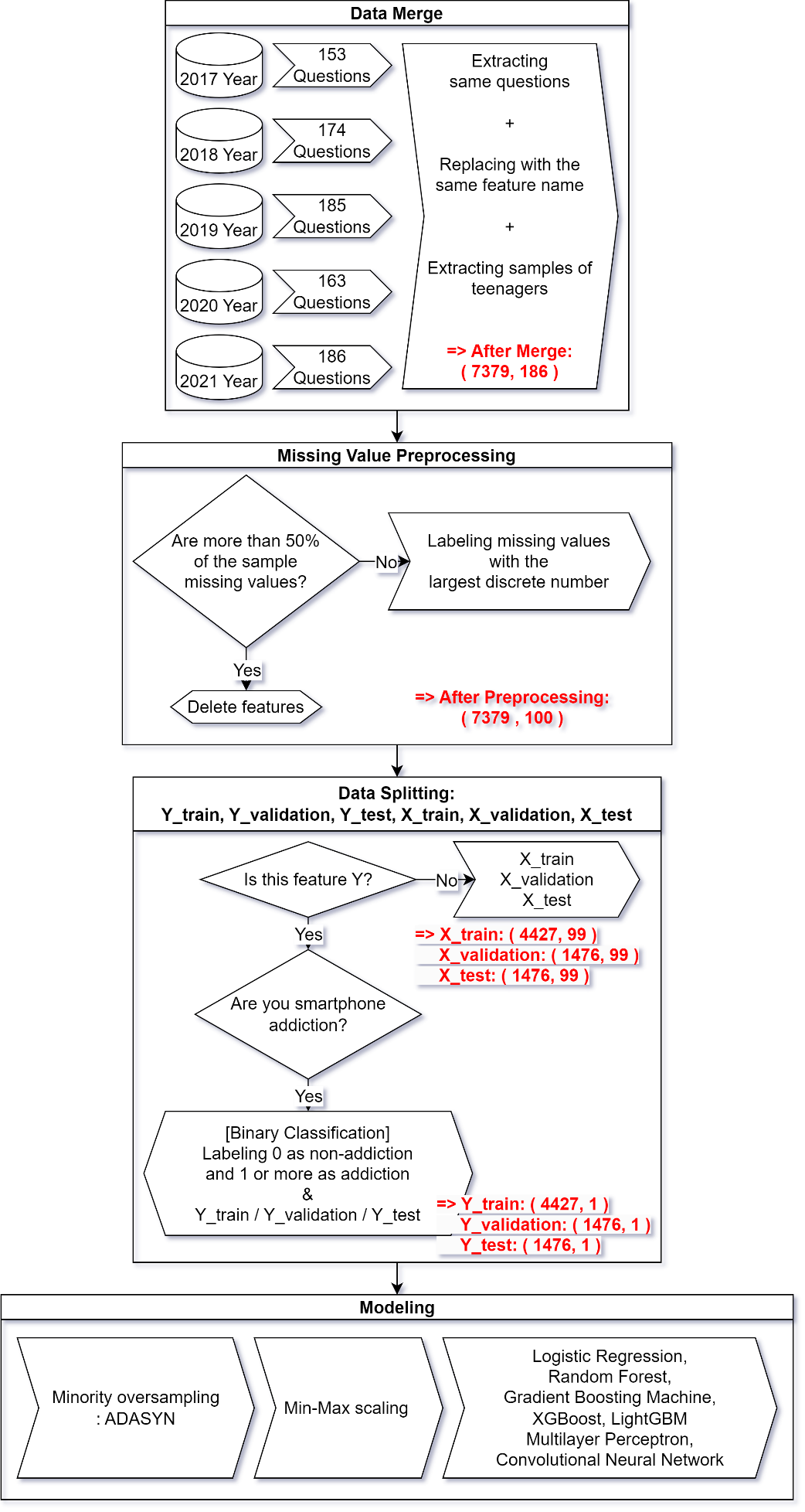


Figure 1. Data preprocessing for modeling.[실험 최종 확정시 그림 수정할 예정]

* 1. *Statistical Algorithm*

첫번째로 사용할 대표적인 regression algorithm은 linear regression이다. 이 모델은 statistical regression 알고리즘 중 하나로 전통적으로 다양한 분야에서 사용되는 기초 알고리즘이며 각 sample data에 대응되는 변수들이 종속변수에 영향을 주는 가중치가 곱해져서 예측값이 추정되는 개념이다. The basic form of this algorithm for 지하철 혼잡도 is as follows:

따라서 각 independent feature가 지하철 혼잡도에 얼마나 기여하는지 파악할 수 있지만 독립변수와 종속변수의 관계를 linear로 가정하기 때문에 정확성이 다른 머신러닝이나 딥러닝 대비 낮은 경향이 있다.

* 1. *Machine Learning Algorithm: Bagging*

Bagging은 여러 개의 샘플을 추출하여 모델링을 구성한 후 outputs의 투표로 최종 output의 label을 결정하는 방식이기에 bootstrap aggregating으로 불린다. 두번째 사용할 알고리즘은 이러한 bagging 알고리즘 중 대표적 알고리즘인 Random Forest(RF)다. 여러 개의 samples를 추출한 후 decision tree 알고리즘을 모델링으로 사용하여 여러 개의 outputs들의 투표를 통해 최종 output을 결정한다. 각 tree predictor의 그룹들로 의사결정하기 때문에, training을 random resampling하여 동일한 크기의 데이터로 분리하는 것이 필요하다. 그런 다음 각 sample에 대해 classification and regression tree (CART)를 사용하여 회귀 또는 분류문제를 예측하여 가장 많은 투표를 획득한 class를 최종 예측값으로 출력하는 ensemble model이다. 따라서 모든 samples를 병렬처리로 한꺼번에 모델링 할 수 있기 때문에 속도도 빠르다. 또한 RF는 training 데이터의 학습 성능이 우수한 편이고 noise data에도 성능이 크게 변하지 않는 robust algorithm이고, variable importance를 제공하기에 설명력도 가지고 있다 (Michie, Spiegelhalter et al. 1999). 하지만 분류 tree splitting 과정에서 각 변수들의 우선순위를 평균한 것으로 positive or negative와 같은 영향력의 방향성을 포함하지는 못하는 단점이 있다.

* 1. *Machine Learning Algorithm: Boosting*

Bagging처럼 Boosting도 classification and regression 문제를 해결하는데 모두 사용할 수 있는 supervised learning이다 (Chen and Guestrin 2016). Gradient boosting methods(GBM)를 기반으로 하고 있으며 Extreme gradient boosting (XGBoost) and lightGBM(LGBM), Catboost 등의 알고리즘으로 확장되며 다양한 기능이 추가되었다. 본 연구에서는 XGBoost and LGBM을 대표적인 boosting 알고리즘으로 스마트폰 의존도 예측에 활용한다. 이 알고리즘들 역시 CART를 기반으로 하고 있으며 전체 데이터를 학습할 때 잘 학습되지 않은 에러들을 가중치를 높여 재학습을 하면서 성능을 향상시키는 ensemble model이다. 메모리를 효율적으로 사용하거나 computation 속도를 높이기 위한 많은 기능들이 추가되었다. 따라서 iterative learning과 병렬처리 과정에서 성능이 높아질 수 있었다 (Alsubari, Deshmukh et al. 2021).



Figure 1. Comparison of bagging and boosting algorithms (Cha, Moon et al. 2021).

* 1. *Deep Learning Algorithm: MLP, CNN*

딥러닝은 머신러닝의 한 방법론으로, 인간의 두뇌에서 영감을 받아 개발된 방식으로 데이터에서의 일반적인 규칙을 학습하기 위해서 연속된 layer를 중첩하여 점진적으로 의미있는 규칙들을 배우는데 강점이 있는 새로운 방식이다. 이러한 구조를 본따 대표적 알고리즘으로 multilayer perceptron (MLP)가 있으며, 이미지나 시계열 등의 데이터를 학습 할 수 있도록 전처리 구조를 업데이트 한 convolutional neural network (CNN)과 recurrent neural network (RNN) 등으로 인간의 수준까지 성능이 향상 될 수 있었다. 이러한 neural network structure를 가진 알고리즘들 역시 간단하게 activation function만 변경함으로써 classification and regression 문제에 모두 적용가능한 universal approximator이다. 또한 수많은 가중치들을 한꺼번에 추정하기 위해서 feed-forward and backpropagation 과정을 사용하여 loss function을 최소화하는 gradient descent optimizer를 사용한다 (see Figure 2) (Viswavandya, Patel et al. 2021). 머신러닝과 마찬가지로 다양한 nonlinear patterns를 학습함으로써 인간이 인지하기 어려운 것들도 학습하는데 능하다.



Figure 2. Architecture of MLP and example of the training process.

본 연구에서는 시계열의 특성을 반영하진 않기 때문에 MLP and CNN 알고리즘을 활용하여 스마트폰 과의존을 예측할 것이다. 특히 CNN은 이미지 classification에 특화가 되어 있긴 하지만 feature를 요약하여 방식이기 때문에 regression 문제에도 적용이 가능하다. Feature를 요약하기 위해서convolution, pooling, and fully connected layers와 같은 blocks를 활용하여 인접한 feature values들의 관련성을 반영하여 새로운 feature를 genarating하는 것이다. 이러한 MLP and CNN 알고리즘은 여러 개의 neurons로 구성된 layer들을 여러 개 통화하게 되는데, 결국 low-level feature vector에서 점차 high-level feature vector로 변환된다 (Shustanov and Yakimov 2017). 따라서 다양한 인구통계나 응답들의 patterns가 요약될 것이고 마지막 output은 주어진 class에 속할 확률이 출력되어 classification 문제를 해결하게 된다.

* 1. *Model Explainability: SHapley Additive exPlanations (SHAP)*

머신러닝 및 딥러닝과 같은 인공지능 알고리즘은 발생가능한 변수들의 모든 상호작용들을 스스로 생성하여 학습하기 때문에 성능이 매우 높은 대신에 왜 그러한 결과가 발생한 것인지 심플하게 설명하기 어려운 블랙박스 알고리즘이다. 일부 머신러닝 알고리즘의 경우 종속변수의 분류에 도움이 된 순서대로 누적하여 feature importance를 출력하기도 하지만 방향성이 없고 선형기반의 해석은 현실과 다를 수 있기 때문에 주의를 기울여야 한다. 하지만 SHAP(Shapley Additive exPlanations)는 local interpretable model-agnostic explanations (LIME)과 shapley value를 연결한 이론으로, 이러한 한계를 보완하여 어떤 요인 또는 feature가 Loyalty 여부에 가장 높은 영향을 주는지 알 수 있다. LIME은 주어진 데이터 수치를 변화시킬 때 모델 예측값은 어떤 변화가 있는지 가중치를 계산하는 이론이다 (Molnar (2023)). 그리고 shapley values는 coalitional game theory를 기반으로 feature의 기여도를 계산하는 metric이다. 기본적으로 feature가 가질 수 있는 모든 coalitions 경우들을 만들고 feature value가 입력되었을 때 변화된 기여도의 평균을 계산한다. 하지만 연산의 효율성을 위해 랜덤 샘플들을 선택하여 기여도를 계산한다. 따라서 이 2가지 이론이 결합된 SHAP를 사용하여 예측값에 대해 각 feature가 기여하는 정도를 계산할 수 있고 블랙박스와 같은 인공지능 알고리즘을 설명하는 것이다. 기본적으로 선형 기반의 단순한 기울기로 해석하는 전통적인 statistical algorithms는 우리가 이해하기는 쉬울지언정 구조가 단순하여 정확도가 낮아서 미래 예측에는 활용되기 어렵다. 반면, 머신러닝 또는 딥러닝의 경우 구조가 매우 정교하지만 높은 정확성을 나타낸다. 따라서 높은 성능의 알고리즘을 사용하되 SHAP와 같은 기능들을 결합하여 설명 가능성을 확보하기 위한 model explainability의 중요성은 점점 높아지고 많은 연구들이 진행되고 있다.

* 1. *Evaluation Metrics*

지하철 혼잡도를 예측하고 설명의 신뢰성을 높이기 위해 6개의 regression metrics를 사용하여 모델링의 성능을 확인한다. 이들은 RMSE(Root Mean Squared Error), MSPE(Mean Squared Percentage Error), MAE(Mean Absolute Error), MAPE(Mean Absolute Percentage Error), MedAE(Median Absolute Error), MedAPE(Median Absolute Percentage Error)이다. 이러한 six evaluation metrics는 아래와 같이 계산된다:

모든 metrics는 예측 성능이 좋을수록 낮은 수치들이 나오도록 실제값()과 예측값()의 차이로 구성되어 있다. 따라서 6개의 metrics 모두 낮은 수치가 나오는 경우 지하철 혼잡도를 잘 예측하는 모델이라 볼 수 있다.

# Results

이번 섹션에서는 전처리 완료된 종속변수와 독립변수를 사용하여 모델에 데이터를 학습 지하철 혼잡도의 예측 결과를 제시한다. 실제 지하철 혼잡도 예측 성능이 높아야 모델이 설명하는 기여도 또는 관련성과 같은 설명 신뢰성이 확보될 수 있을 것이다. 선형회귀분석과 같은 전통적인 알고리즘을 사용하는 경우 지하철 혼잡도 예측을 위한 변수 기여는 쉽게 알수가 있지만 실제 그러한 변수들의 설명력으로 혼잡도를 예측할 경우 성능이 낮은 경향이 있다. 전통적인 사회과학에서 설명력에 집중하느라 그러한 설명이 미래에 얼마나 비즈니스적으로 신뢰할 수 있는 결과인지는 경시하는 경향이 있다. 따라서 머신러닝과 딥러닝으로 forecasting performance 결과를 우선적으로 확인하여 모델링의 성능을 확보하고 다음으로 블랙박스와 같은 알고리즘에 SHAP 알고리즘을 적용하여 신뢰할 수 있는 변수들의 설명력을 제공한다. 데이터 준비, 전처리, 모델링, 성능 검증 등의 모든 데이터분석 프로세스는 python 3.9.20 버전을 사용하였다. 그리고 Machine Learning 알고리즘들은 sklearn 1.5.2 버전의 라이브러리를 그리고 Deep Learning 알고리즘은 tensorflow 2.17.0 버전의 라이브러리를 사용하였다. 그리고 알고리즘의 예측 성능을 설명하기 위해 사용한 SHAP는 0.46.0 버전의 라이브러리를 사용하였다.

* 1. *Prediction Performance*

이번 section에서는 지하철 혼잡도의 모델링 성능을 확인하기 위해 머신러닝 및 딥러닝 알고리즘의 미래 예측 성능을 소개한다. 예측문제를 해결하기 위한 전통적인 알고리즘인 Linear Regression, 대표적인 머신러닝 알고리즘인 Random Forest, XGBoost, LightGBM, 그리고 CatBoost를 포함하여 총 5가지 머신러닝 알고리즘을 사용하였다. 그리고 대표적인 딥러닝 알고리즘인 MLP and CNN 총 2가지 알고리즘을 사용하였다. 전체 데이터 80%의 training set을 사용하여 모델링하고 hyperparameters를 조정하였다. 그리고 20% test set을 사용하여 미래 예측 성능을 확인함으로써 모델의 성능을 확보한다. 알고리즘을 성능을 공평하게 비교하기 위해 가급적 동일한 parameter values를 사용하였다 (표 1).

표 3. 지하철 혼잡도 모델링에 사용한 알고리즘의 하이퍼파라미터

|  |  |  |
| --- | --- | --- |
| **Algorithm** | **Parameter** | **Value** |
| **Logistic Regression** | Intercept | TRUE |
| Max Iteration | 100 |
| Random State | 0 |
| **Random Forest**  **XGBoost**  **LightGBM**  **CatBoost** | Samples | 100 |
| Criterion | Gini |
| Minimum Spitting Samples | 2 |
| Random State | 0 |
| **MLP** |  |  |
| **CNN** |  |  |

알고리즘에 데이터를 학습시킨 후, 지하철 혼잡도의 Test 예측 성능을 검증한다. 각 역마다의 모든 알고리즘의 검증지표 순위를 추정한 후 30개역의 순위 평균치(표 4)와 각 알고리즘마다 30개역의 검증지표 평균값(표 5)를 통해 보다 일반화된 알고리즘의 설명이 가능하다. 표 4를 보면 RMSE의 경우 MLP 알고리즘이 30개역 평균 순위가 1.31로 나타난다. 즉, 대부분의 경우에서 RMSE 지표는 MLP 알고리즘의 지하철 혼잡도 예측 순위가 1등으로 나타남을 의미한다. 그런데 RMSE를 제외한 나머지 5개 검증지표들에서는 Random Forest 알고리즘이 30개역 평균 순위가 1.15와 1.00으로 나타났다. 따라서 나머지 검증지표들에선 Random Forest 알고리즘이 거의 항상 1등을 했다는 것이다. 따라서 6개의 검증지표들 중 5개의 검증지표에서 상위순위를 차지한 알고리즘은 Random Forest다. 그리고 다음 순위를 차지한 알고리즘은 MLP다. 실제 알고리즘별로 30개역의 검증지표 수치를 평균해 보았더니(표 5), 1순위를 차지한 Random Forest 알고리즘이 평균치 기준인 MSPE에서 0.02%, MAPE에서 0.50%를 차지할만큼 1% 미만의 오차를 보이며 매우 정확하게 지하철 혼잡도를 예측하는 것으로 나타났다. 2순위를 차지한 MLP의 경우도 MSPE와 MAPE가 각각 0.34%와 1.83%를 나타낼 정도로 높은 정확도를 보이고 있다. 반면 전통적인 선형회귀분석 알고리즘은 순위에서도 최하위이며 실제 예측 오류는 MSPE가 144% 만큼 매우 부정확한 예측임을 알 수 있다. 따라서 이러한 정확도를 보이는 모델링에서 변수들의 기여도 또는 설명력을 신뢰하는 것도 어렵다고 판단된다. 따라서 지하철 혼잡도 예측에 사용한 공공 빅데이터 융합DB의 독립변수들 패턴은 단순히 선형회귀분석에서 잘 학습될 만큼 선형(Linear)패턴이 아니고 훨씬 복잡한 비선형(Non-linear) 패턴이 많이 포함되어 있음을 의미하고, 고성능의 PC 성능을 요구하는 고성능 딥러닝 알고리즘이 아니더라도 공공 빅데이터를 융합할 경우 매우 정교하고 높은 예측 모델링을 구축할 수 있음을 의미한다.

표 4. 수송인원수 상위 30개 역의 미래 예측 성능 검증지표 순위 평균

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Algorithm** | **RMSE** | **MSPE** | **MAE** | **MAPE** | **MedAE** | **MedAPE** |
| **Linear Regression** | 6.95 | 6.95 | 7.00 | 6.95 | 7.00 | 6.95 |
| **Random Forest** | 1.74 | **1.15** | **1.00** | **1.00** | **1.00** | **1.00** |
| **XGBoost** | 4.23 | 4.38 | 4.23 | 4.33 | 4.36 | 4.31 |
| **LightGBM** | 4.82 | 4.69 | 4.85 | 4.77 | 4.82 | 4.85 |
| **CatBoost** | 6.00 | 5.90 | 5.97 | 5.90 | 5.87 | 5.90 |
| **MLP** | **1.31** | 1.85 | 2.00 | 2.00 | 2.00 | 2.00 |
| **CNN** | 2.95 | 3.03 | 3.00 | 3.00 | 3.00 | 3.00 |

표 5. 알고리즘별 수송인원수 상위 30개 역의 미래 예측 성능 검증지표들 중 퍼센트 오차 평균

|  |  |  |  |
| --- | --- | --- | --- |
| **Ranking** | **MSPE** | **MAPE** | **MedAPE** |
| **Random Forest** | **0.02%** | **0.50%** | **17.47%** |
| **MLP** | 0.34% | 1.83% | 89.16% |
| **CNN** | 0.90% | 4.30% | 234.97% |
| **XGBoost** | 3.68% | 9.86% | 675.02% |
| **LightGBM** | 5.89% | 10.82% | 732.84% |
| **CatBoost** | 20.73% | 16.69% | 1052.98% |
| **Linear Regression** | 144.05% | 48.20% | 2865.35% |

일반화된 성능 이외에 실제 얼마나 지하철역의 혼잡도를 잘 예측하는지 확인하기 위해 미래 월별 혼잡도와 특정일 시간대별 혼잡도 예측 성능을 확인한다. 우선 그림 1의 오른쪽에 혼잡도 1위로 분포하고 있는 강남역과 가장 하위에 분포하고 있는 경복궁역을 대상으로 하였다. 가장 최근 개통된 신분당선이 환승되기도 하며 국내의 가장 많은 유동인구가 있기에 강남역을 대상으로 하는 것은 중요하며, 경복궁역 역시 대한민국의 역사적, 상징적, 그리고 문화적으로도 관광객이 많이 이용하는 곳이기 때문에 국내외 이용자들에겐 매우 관심이 있는 혼잡도 정보가 될 것이다. 두 역 모두 Random Forest 알고리즘이 가장 높은 성능을 보였고 비교를 위해 두번째로 높은 성능인 MLP 알고리즘과 함께 월별(그림 2) 그리고 시간대별(그림 3) 혼잡도 예측을 시각화하였다. 월별 미래 예측 성능에서도 강남역과 경복궁역 모두 Random Forest 알고리즘이 매우 실제값과 유사한 혼잡도를 예측하는 것을 확인할 수 있고, 시간대별 예측 성능은 더욱 정확하게 예측하는 모습을 보인다. 따라서 개별 지하철역에서도 미래 혼잡도를 예측하는데 매우 효과적인 방법임을 확인할 수 있다. 이제 매우 신뢰성 높은 모델을 구축하였으니 비즈니스 의사결정에 직접적으로 활용될 수 있는 신뢰성 있는 변수들의 기여 또는 설명력을 제공할 수 있다.

|  |  |
| --- | --- |
|  |  |
|  |  |

그림 2. 강남역과 경복궁역의 월별 Random Forest(1등)와 MLP(2등) 알고리즘 미래 예측 시각화

|  |  |
| --- | --- |
|  |  |
|  |  |

그림 3. 강남역과 경복궁역의 시간대별 Random Forest(1등)와 MLP(2등) 알고리즘 미래 예측 시각화

* 1. *Explainability of 지하철 혼잡도 Prediction*

머신러닝 그리고 딥러닝과 같은 인공지능 알고리즘은 복잡한 비선형 패턴과 변수들의 모든 상호작용 조합까지도 학습을 하여 모델을 만들기 때문에 전통적인 linear regression과 달리 예측에 기여하는 독립변수들의 기여도를 단순하게 파악하기 어렵다. 물론 머신러닝 알고리즘은 딥러닝과 달리 데이터의 학습 과정에서 예측 성능을 높이는데 기여한 feature importance를 제시하긴 하지만 순위들을 단순하게 누적할 뿐 positive or negative effect와 같은 방향성조차 제시하지 못한다. 하지만 SHAP explainer를 사용하면 지하철 혼잡도의 높은 미래 예측 성능이 왜 그러한지 독립변수들의 상대적 기여도로 평가할 수 있다. SHAP는 독립변수들의 수치들만 수집되면 실시간으로 지하철의 혼잡도가 얼마가 될 것인지 위 section에서 파악한 매우 높은 성능으로 예측해 낼 수 있다. 뿐만 아니라 독립변수들이 어떠한 방향으로 기여해서 그러한 높은 미래 예측 성능을 달성하였는지 설명해주기 때문에, 실제 지하철 교통 비즈니스에 활용되기도 용이하고 신뢰성도 높아 의사결정에 용이하게 활용될 수 있다. 그리고 이러한 실시간 혼잡도처럼 누적된 혼잡도 예측과 설명력들을 모두 결합하여 일반화된 정책으로 활용할 수 있는 지하철 혼잡도의 신뢰성 높은 설명력을 시각적으로 표현하기 때문에 이해하기 쉽다.

우선 실시간 지하철 혼잡도에 대한 설명력을 decision plot으로 시각화 할 수 있다. 가장 높은 성능을 보이는 Random Forest 알고리즘을 기준으로, 그림 4는 강남역(상단)과 경복궁역(하단)의 미래 특정일을 임의로 선택하여 혼잡도를 예측하고 어떻게 혼잡도 수치가 나오게 되었는지 변수들의 기여를 내림차순으로 보여준다. 예를 들어 강남역의 혼잡도가 높을 때 약 58점 정도의 수치가 예측되었는데 “하차인원, 출입구, 월평균\_환승유입인원수” 등은 예측 혼잡도를 높이는 변수들이고 “승차인원, 청소년인원수, 환승노선\_개수” 등은 예측 혼잡도를 낮추는 변수들이다. 그외 나머지 변수들은 혼잡도에 크게 기여하지 못하는 모습이다. 하지만 강남역 혼잡도가 낮을 때는 “승차인원”이 주된 혼잡도를 낮추는 변수이며 “환승유입인원수, 하차인원, 출입구, 월평균\_환승유입인원수, 환승노선\_개수, 하행\_평균운행간격” 등은 혼잡도를 높이는 변수들로 나타난다. 미래 특정일과 상황에 따라 변수들이 혼잡도를 높이거나 낮추는 영향정도와 방향은 얼마든지 달라질 수 있으며 실시간으로 높은 정확성으로 혼잡도를 예측할 뿐만 아니라 어떻게 그러한 수치가 나오게 되는지 의사결정 근거로 활용하기에도 매우 용이하다.

|  |  |
| --- | --- |
| (강남역 혼잡도 높을 때) | (강남역 혼잡도 낮을 때) |
| (경복궁역 혼잡도 높을 때) | (경복궁역 혼잡도 낮을 때) |

그림 4. 미래 특정 일 대상 강남역과 경복궁역의 혼잡도 예측과 변수영향 설명력 기반 의사결정

특정일 또는 시간대의 지하철 혼잡도에 대한 변수들의 기여를 확인하였다. 각 변수가 지하철 혼잡도에 일반화된 positive or negative를 보장하지도 않는다. 그러한 이유로 특정 변수가 지하철 혼잡도에 어떠한 방향성을 가지는지 일반화하는 것은 다소 위험할 수 있다. 정책적 근거를 위해 일시적으로 사용할 수는 있겠지만 정책 입안자들 또는 시민들에게 납득될만한 정보로 단정하기 어렵기 때문이다. 그렇다고 정책적 근거로 사용하지 말자는 것은 아니며, 높은 정확성의 실시간 혼잡도의 예측 설명력을 확인할 수 있었으니 이를 누적하여 표현한다면 충분히 일반화된 인싸이트로 재표현 할 수 있을 것이다. 단, positive or negative 와 같은 이분법적인 해석은 매우 조심해야 함을 위의 실시간 예측 결과를 통해 다시 한번 강조한다. 실제 독립변수와 종속변수의 관계가 이분법적 설명으로 명확할 정도로 선형적 관계이지도 않다. 만약 선형적 관계라면 선형회귀분석의 결과가 성능이 좋아야 하는데 전혀 그렇지도 않았기 때문이다. 따라서 본 연구에서는 매우 명확한 관계성이 나타나는 변수에 대해서만 조심스럽게 해석을 제안하면서 머신러닝 딥러닝 기반의 설명가능한 인공지능의 효과적인 활용법을 가이드 하는데 집중할 것이다.

각 샘플마다의 지하철 혼잡도 예측 설명력을 누적하여 일반화된 설명력을 확인할 것이다. 단, 본 연구에 활용한 샘플 수 기준의 일반화이기 때문에 샘플이 늘어나가나 줄어들 경우 변경될 수도 있음을 인지해야 한다. 아마도 데이터의 샘플이 매우 커진다면 향후 그 일반화 설명력을 더욱 신뢰할 수 있을 것임은 분명하다. 그림 5는 Random Forest 알고리즘으로 모든 미래 데이터의 지하철 혼잡도 예측을 위한 feature impacts를 시각화 한 것이다. 우선 세로축은 왼쪽과 오른쪽으로 2개가 있다. (왼쪽)세로축은 종속변수에 영향을 주는 feature importance 를 내림차순으로 정렬한 것이기 때문에 높은 곳에 위치한 변수들은 낮은 곳에 위치한 변수들보다 지하철 혼잡도 예측에 더욱 많이 기여한다. 즉, “하차인원, 승차인원, 환승노선\_개수, 출입구, 하행\_평균운행간격, 청소년인원수, 승하차인원차” 등의 순서로 변수들의 기여순위가 분석된다. 그리고 (오른쪽)세로축은 변수들의 값이 낮은 경우는 파란색 계열로 그리고 높은 경우는 빨간색 계열로 표시하여 변수들의 값에 대응되는 지하철 혼잡도의 예측값의 변화를 표시하였다. 마지막으로 가로축에 바로 지하철 혼잡도의 예측값을 반영하였는데, 예측에 대한 각 독립변수들의 average marginal contribution이다. 따라서 SHAP 값이 0보다 작으면 negative contribution을 0보다 크면 positive contribution을 의미한다. 각 시점마다 변수의 수치는 다양할 수 있고 각 수치별 지하철 혼잡도 or feature impact 방향도 다양할 수 있기 때문에, 각 변수들의 값과 대응되는 SHAP output 위치를 모두 누적하여 표현하면 각 변수의 값의 변화에 따른 지하철 혼잡도 기여 방향이 시각화 되는 것이다.

예를 들어 “하차인원과 승차인원” 변수는 값이 작을때(파란색계열) negative contribution에 많이 분포되어 있고 값이 커지면(빨간색계열) positive contribution에 많이 분포가 되어 있다. 따라서 해당 변수의 값이 높을수록 지하철 혼잡도가 높아지도록 영향을 준다고 해석할 수 있다. 반대로 “환승노선\_개수” 변수는 값이 작은 수치에서 커질수록 negative contribution에 많이 분포되기 때문에 지하철 혼잡도를 낮추는데 영향을 준다고 해석할 수 있다. 이를 정리하여 미래 예측 성능이 가장 높은 지하철 혼잡도에 positive correlation을 보이는 가장 중요한 특징은 "하차인원, 승차인원, 출입구, 청소년인원수, 승하차인원차, 우대권인원수, 환승유입인원수" 등이며, negative correlation을 보이는 특징은 "환승노선\_개수, xxx" 등으로 해석할 수 있다. 마찬가지로 나머지 변수들도 유사한 해석을 할 수 있지만 값의 변화와 contribution 방향이 명확한 정/역방향인 경우 그리고 그 contribution 변화의 폭이 넓은 경우만 정책적인 근거로 활용하길 추천한다. 그렇지 않은 애매한거나 좁은 범위의 변화를 억지로 일반화 하는 것은 왜곡된 해석을 줄 수 있으니 지양하는 것이 좋다.

|  |
| --- |
|  |

그림 5. 실시간 지하철 혼잡도 예측에 기여하는 변수들의 설명력 누적 기반 의사결정

# Conclusions

.

설명가능한 인공지능 알고리즘을 사용하면 실시간으로 지하철 혼잡도를 높은 성능으로 거의 정확하게 알수 있으며, 데이터의 샘플을 늘려 빅데이터에 가까울수록 정책적 비즈니스 사결정에 활용가능한 일반화된 변수들의 기여도고 파악할 수 있다. 특히 SHAP 모델에 의한 explainable feature importance는 기존 블랙박스와 같은 인공지능 알고리즘이 갖는 한계를 넘어서 비즈니스 활용에 대한 이해도를 훨씬 향상시키는 것이다. 이러한 미래 개별 예측력과 전체적인 일반화된 설명력으로 단순히 연구 수준을 넘어서 지하철 혼잡도와 관련된 비즈니스 의사결정에 기여할 수 있으다. 따라서 머신러닝 또는 딥러닝과 같은 간단한 인공지능 알고리즘을 활용하여 지하철 혼잡도를 빠르게 확인 할 수 있고 비즈니스 측면에서도 빠르고 시민들에게 효과적인 대응을 하는데 도움이 될 것이다. 본 연구에서는 "xxx, xxx, xxx” 등은 지하철 혼잡도를 높이는데 가장 중요하게 영향을 주는 것으로 나타났다. 이러한 결과는 다수의 선행연구에서 분석된 지하철 혼잡도에 영향을 주는 주요 변인들로 손꼽히는 xxx, xxx, xxx 등의 연구결과와 일치한다. 추가적으로 본 연구에서는 이전 연구에서 발견되지 않은 xxx 등도 지하철 혼잡도의 중요한 예측인자가 될 수 있음을 발견했다.

# Data Availability

The data can be provided on request.

# Declaration of Competing Interest

The authors declare no conflict of interest.

# CRediT authorship contribution statement

**Jaeheung Park:** Conceptualization, Software, Validation, Data Curation, Writing Original Draft, Preparation. **Kyungwon Kim:** Methodology, Preparation, Software, Writing Review and Editing, Visualization, Supervision.

# References

Alsubari, S. N., S. N. Deshmukh, M. H. Al-Adhaileh, F. W. Alsaade and T. H. H. Aldhyani (2021). "Development of Integrated Neural Network Model for Identification of Fake Reviews in E-Commerce Using Multidomain Datasets." Applied Bionics and Biomechanics **2021**: 5522574.

Cha, G.-W., H.-J. Moon and Y.-C. Kim (2021). "Comparison of random forest and gradient boosting machine models for predicting demolition waste based on small datasets and categorical variables." International Journal of Environmental Research and Public Health **18**(16): 8530.

Chen, T. and C. Guestrin (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, California, USA, Association for Computing Machinery**:** 785–794.

Feng, J., H. Xu, S. Mannor and S. Yan (2014). "Robust logistic regression and classification." Advances in neural information processing systems **27**.

Michie, D., D. Spiegelhalter and C. Taylor (1999). "Machine Learning, Neural and Statistical Classification." Technometrics **37**.

Molnar, C. (2023). Interpretable machine learning. A Guide for Making Black Box Models Explainable, Lulu. com.

Shustanov, A. and P. Yakimov (2017). "CNN Design for Real-Time Traffic Sign Recognition." Procedia Engineering **201**: 718-725.

Viswavandya, M., S. Patel and K. Sahoo (2021). "ANALYSIS AND COMPARISON OF MACHINE LEARNING APPROACHES FOR TRANSMISSION LINE FAULT PREDICTION IN POWER SYSTEMS." Journal of Research in Engineering and Applied Sciences **6**: 24-31.

1. https://www.statista.com/chart/30882/share-of-commuters-that-take-public-transport/ [↑](#footnote-ref-1)
2. https://news.seoul.go.kr/traffic/archives/289 [↑](#footnote-ref-2)