고성능 인공지능 알고리즘 활용 2025년도 KTX 수요예측

차명주a, 오영택b, 이승연b, 김경원a,*∗*

*a인천대학교 글로벌정경대학 무역학부*

*b한국철도공사 철도연구원*

# 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:* Demand Forecasting, Explainable Prediction, Machine and Deep Learning, SHAP, Business Decision

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

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

# Introduction

고속철도는 전 세계 교통 시스템에서 지속 가능한 발전의 핵심 축으로 자리 잡고 있다. 프랑스의 TGV, 일본의 신칸센, 중국의 고속철도와 같은 사례는 고속철도가 국가 경제와 사회적 연결성 강화에 미치는 긍정적 영향을 잘 보여준다. 이들 사례는 고속철도가 단순히 국가 내 이동 수단에 국한되지 않고, 경제적 연결성과 국제적 경쟁력을 강화하는 주요 인프라로 기능할 수 있음을 입증한다. 현재 25개국 이상이 고속철도를 도입하여 주요 교통수단으로 활용하고 있으며, 이러한 국가는 고속철도를 통해 물류와 인구의 이동성을 극대화하며 교통체계의 효율성과 환경적 지속 가능성을 동시에 달성하고 있다. 특히, 고속철도는 전기차보다 12배, 내연기관 차량보다 26배 낮은 CO2 배출량을 기록하며, 지속 가능한 교통수단의 대표 사례로 자리 잡고 있다(UIC, 2023).

글로벌 고속철도의 발전 속에서 KTX는 한국의 대표적인 고속철도 시스템으로 자리 잡았다. 2004년 개통 이후 교통 편리성과 시간 절약이라는 두 가지 핵심 가치를 통해 빠르게 성장해 왔다. 코로나19 팬데믹으로 인해 2020년과 2021년에는 수송 인원이 일시적으로 감소했으나, 이후 빠르게 회복하며 2022년 기준 약 7,500만 명의 이용객을 기록했다. 2024년에는 연간 이용객 수가 총 1억 1,658만 명으로 전년 대비 5.4% 증가하며 지속적인 성장을 이어갔다(국토교통부, 2024) 이와 같은 성장세는 KTX가 단순한 교통수단을 넘어, 국가 기반 교통망의 중심축으로 기능하고 있음을 보여준다.

최근 화제가 된 ‘수서역 KTX’ 논의에서도 알 수 있듯 이용객 수가 지속적으로 증가함에 따라 특정 시간대와 주요 노선에서 발생할 수 있는 혼잡 문제, 자원 배분의 불균형 등 운영상의 과제가 점차 복잡해지고 있다. 이러한 문제를 효과적으로 해결하기 위해서는 수요 변화를 사전에 예측하고 자원을 최적화하는 정밀한 운영 전략 수립이 필수적이다. 특히, KTX가 속한 교통 분야의 수요 예측은 운영 최적화뿐만 아니라 정책적 의사결정을 지원하는 중요한 도구로, 교통 인프라의 지속 가능한 발전에 기여할 수 있다. 국토교통부 역시 이러한 필요성을 인식하고, 교통 분야의 수요 예측과 연계된 미래 선도 기술 개발을 제안하고 있다(국토교통부, ‘국토교통 2050 미래기술 도출을 위한 조사분석 연구’, 2020).

고속철도 수요 예측은 주로 전통적인 통계적 접근에 기반한 시계열 분석 기법을 활용해 이루어져 왔다. 김관형 and 김한수(2011)는 개입 ARIMA 모형을 사용하여 경부고속철도 2단계 개통과 같은 정책적 개입이 수요에 미치는 영향을 분석하였다. 이 연구는 정책 변화에 따른 단기적 수요 변화를 효과적으로 포착했지만, 열차 운임, 운행시간, 서비스 품질 등 다양한 변수 간의 상호작용을 반영하지 못했고, 데이터의 선형적 구조를 가정하여 설명력에 한계를 드러냈다. Cha et al.(2019)은 다중 개입 계절형 ARIMA 모형을 활용하여 외부 개입 요인(예: 경부고속철도 2단계 개통, 호남고속철도 개통, 국가 감염병 발생)을 반영한 수요 예측을 수행하였다. 계절적 요인과 외부 충격을 함께 고려해 이전보다 개선된 결과를 도출했지만, 전통적인 시계열 분석 기법이 가진 구조적 한계에서 벗어나지 못했다. 이러한 방법론은 데이터가 선형적 관계를 따르며, 정상성(stationarity)을 만족해야 한다는 전제 조건을 가지므로 현실 세계의 복잡한 데이터 특성을 충분히 반영하지 못하고 정보 손실 가능성을 내포하고 있다. 결과적으로, ARIMA와 같은 시계열 분석 기법은 과거 데이터를 기반으로 수요 예측에 널리 활용되어 왔으나, 선형적 관계를 기반으로 한다는 점에서 현실 세계의 복잡하고 비선형적인 수요 변화를 효과적으로 예측하기에는 한계가 있다(Tan et al., 2009). 이러한 전통적 통계 기반 기법은 대규모 데이터 처리와 실시간 분석이 요구되는 현대 고속철도 수요 예측 문제에서 특히 두드러진 한계를 보인다.

머신러닝 및 딥러닝과 같은 고성능 인공지능 알고리즘은 데이터 내에 내재된 복잡한 패턴을 학습하고, 변수 간의 관계를 자동으로 탐지하며, 대규모 데이터를 처리하는 데 강점을 지닌다. 특히, 인공지능 기반 수요 예측 기법은 높은 예측 정확도와 계산 효율성을 제공하며, 비선형 데이터 패턴의 학습과 다양한 변수 간의 관계 탐지가 가능하다는 점에서 기존 통계적 접근법의 한계를 효과적으로 보완한다(Jeong and Lim, 2019). 또한 최근에는 머신러닝 및 딥러닝을 활용하여 고속철도 수요 예측에서 더 높은 정확도를 달성하고, 효율적인 운영 전략 수립에 기여할 가능성을 제시해왔다. 예를 들어, LSTM(Long Short-Term Memory)과 XGBoost를 적용한 연구에서는 비선형 데이터 패턴을 효과적으로 학습하고, 기존 통계 기반 모델보다 더 높은 예측 성능을 보임으로써 알고리즘의 실효성을 입증하였다(심진호 등, 2024). 하지만 인공지능 알고리즘의 발전에도 불구하고 고속철도 수요예측에 활용되는 사례나 연구는 아직 부족한 실정이다.

인공지능의 알고리즘의 높은 성능에도 불구하고 구조가 매우 복잡하여 왜 그러한 결과가 도출되었는지 설명하지 못하는 한계가 있다. 이를 “블랙박스” 이슈라고도 하며 의사결정 과정에서 예측 결과의 신뢰성을 낮추고, 정책 설계나 자원 배분과 같은 실제 활용에 제약을 초래할 수 있다. 이러한 한계는 특히, 교통망 운영과 같은 대규모 인프라에서 심각한 문제로 작용할 수 있다. 이와 같은 문제를 해결하기 위해 최근에는 설명 가능한 인공지능(XAI, Explainable AI)이 주목받고 있다. 설명 가능한 인공지능(XAI)는 인공지능 시스템이 수행하는 예측 및 의사결정 과정을 인간이 이해할 수 있도록 설명하는 기술로, AI 시스템의 행동과 상태를 명확히 전달하여 신뢰성을 높이는 것을 목표로 한다(Gunning et al., 2019). 이 기술은 금융, 의료, 제조 등 다양한 분야에서 활용 사례를 통해 그 중요성을 입증하고 있다. 예를 들어, 금융 분야에서는 SHAP(Shapley Additive Explanations)와 LIME(Local Interpretable Model-agnostic Explanations) 같은 도구를 활용하여 신용등급 평가와 대출 의사결정에서 예측 결과의 해석 가능성을 높이고, 투명한 의사결정을 지원한 사례가 있다(배재권, 2023). 의료 분야에서는 딥러닝 기반의 무릎 골관절염 진단 모델에 XAI를 적용하여 진단 근거를 명확히 제시함으로써 의료진의 신뢰를 확보하고 진단 정확도를 향상시킨 사례가 있다(Ahmed and Imran, 2024). 또한, 제조 분야에서는 XAI를 통해 수주량 변화의 주요 요인을 분석하고, 이를 기반으로 자원 배분 및 운영 최적화를 실현하여 비용 절감과 생산성 향상에 기여한 사례가 있다(정연수 and 윤철희, 2024).

이러한 연구사례들은 XAI가 단순히 예측 결과를 제공하는 데 그치지 않고, 그 결과를 해석하고 시각화함으로써 다양한 산업에서 실질적인 의사결정을 지원하는 도구로 자리 잡고 있음을 보여준다. XAI의 이러한 특성은 고속철도 수요 예측과 같은 대규모 교통 인프라 운영에서도 중요한 기여를 할 수 있다. 특히, XAI를 활용해 수요 예측 결과를 명확히 해석하고 시각화하여, KTX와 같은 교통망 운영에서 정책 설계와 자원 배분과 같은 의사결정을 더 신뢰도 높게 지원할 수 있다.

본 연구는 고성능 인공지능 알고리즘과 설명 가능한 인공지능(XAI, Explainable AI)을 활용하여 2025년도 KTX 수요를 정밀하게 예측하고, 예측 결과를 바탕으로 실질적인 비즈니스 및 정책적 의사결정을 지원하는 것을 주된 목적으로 한다. 이를 통해 KTX 운영 및 관리 효율성을 극대화하고, 지속 가능한 교통체계 구축에 기여하고자 한다. 본 연구의 주요 기여는 다음과 같다. 첫째, 최신 AI 알고리즘을 활용하여 KTX 수요 예측의 정확도를 대폭 향상시킨다. 기존의 통계 기반 모델이 가지는 선형적 한계를 극복하고, LSTM(Long Short-Term Memory), XGBoost 등 고성능 알고리즘을 통해 비선형적이고 복잡한 데이터 패턴을 학습함으로써, 철도 네트워크 확장 및 운영 효율화를 위한 정량적 근거를 제공한다. 이를 통해, KTX 수요 변화의 복잡한 양상을 효과적으로 예측하고, 미래 교통 계획 수립에 기여할 것이다. 둘째, XAI 기술을 통해 예측 결과의 신뢰성과 해석 가능성을 높인다. SHAP(Shapley Additive Explanations)와 LIME(Local Interpretable Model-agnostic Explanations) 등 XAI 도구를 활용하여 예측 과정과 결과를 시각화함으로써, 정책 설계 및 자원 배분 과정에서 신뢰도를 제공한다. 특히, 예측에 영향을 미치는 주요 변수(예: 시간대별 수요, 지역 간 이동량 등)를 분석함으로써, KTX 운영 전략을 보다 정교하게 설계할 수 있도록 돕는다. 셋째, 경량화된 AI 모델을 제안하여 실무 현장에서의 활용도를 강화한다. 고성능 컴퓨팅 환경 없이도 실행 가능한 경량화 모델을 설계함으로써, KTX 운영팀이나 정책 담당자가 현장에서 간편하게 수요 예측 결과를 확인하고 실시간으로 의사결정을 내릴 수 있도록 지원한다. 이를 통해, 데이터 기반 의사결정이 보다 효율적이고 접근 가능한 방식으로 이루어질 수 있다.

결론적으로, 본 연구는 KTX의 운영 효율성을 향상시키는 데 실질적인 기여를 할 뿐만 아니라, 다양한 교통 체계로의 확장 가능성을 제시함으로써 교통 전반의 지속 가능성과 효율성을 높이는 데 중요한 학문적, 실무적 시사점을 제공할 것으로 기대된다.

[논문의 구성]

# Methods

* 1. *Data Preprocessing*

본 연구에서는 경부선, 경전선, 동해선, 전라선, 호남선 총 5개의 주중과 주말, 그리고 전체 수요를 예측하는 것이 목적이다. 한국철도공사 철도연구원으로부터 제공받은 2015년 1월부터 2024년 3월까지 약 10년간의 “수송-운행일-주운행” 그리고 “시종착역별 열차운행” 정보가 담긴 데이터베이스를 결합하여 2025년 12월까지의 월별 KTX 수송수요를 예측하는데 활용하였다. 따라서 전체, 주중, 주말의 요일구분 변수와 열차종의 변수를 기준으로 수요예측에 도움이 될 수 있는 다양한 파생 변수들을 생성하였다. 첫째로, 과거의 수요가 현재 또는 미래의 수요에 영향을 줄 수 있기 때문에 과거 1개월~12개월 전의 수요를 “과거 승차인원수” 파생변수로 생성하였다. 둘째로, 버스나 자하철과 같은 대중교통과 달리 요일이나 이벤트에 따라 수요의 변화가 느리게 발생하기 때문에, 시계열데이터에서 각 월의 실제 날짜수, 주말수, 주중수, 공유일수, 명절 수 등의 “시간정보” 파생변수를 결합하였다. 셋째로, 대외적인 경제상황과 환경변화를 반영하기 위해서 한국의 주식 시장 지표와 소비자의 물가수준, 그리고 코로나 시기의 예방접종 인원수, 격리자수, 사망자수, 정부대응 지수 등의 “외부환경” 파생변수를 반영하여 정교함을 높였다. 마지막으로 한국철도공사에서 제공받은 공급좌석 정보, 열차 정보, 운행정보 등을 재계산하여 “좌석 및 운행정보” 변수로 반영하여 수요예측에 활용하였다.

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 |

본 연구에서 사용한 데이터는 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

표 1에 따르면 본 연구에서 총 17개의 공공 데이터베이스(DB)를 융합DB로 활용하였는데, “서울교통공사\_1\_8호선 역별 일별 시간대별 승객유형별 승하차인원” DB를 기준데이터로 사용하였다. 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. “서울교통공사\_지하철혼잡도정보” DB를 5시부터 23시까지 시간대별 상선 혼잡도와 하선 혼잡도로 분리하여 위 승하차인원(기준 데이터) 기준 데이터에 병합(Merge)하였다. 그 다음으로 "서울교통공사\_월별 환승유입인원" DB의 역 및 월별 환승유입인원 변수를 기준 데이터에 병합하였다. 그 후 2024년 6월 30일 기준 "서울교통공사\_역사운영 현황" DB에서 역별(호선별) 면적, 출입구, 섬식여부, 환승노선 갯수를 추출해서 기준 데이터에 병합하였다. 그리고 종속변수로 사용된 승강장혼잡도를 면적 변수와 승하차인원(합) 변수를 사용하여 생성하였다. "종관기상관측(ASOS)" DB에서는 서울특별시 2021-2023 사이의 일시, 평균기온(°C), 일강수량(mm), 평균 풍속(m/s), 평균 상대습도(%), 일 최심적설(cm) 를 추출하여 기준 데이터에 결합하였다. 날짜 정보의 경우, 위 데이터에서 수송일자를 활용하여 년도, 월, 일, 요일, 주 변수를 만들었고, holidayskr 라이브러리를 통해 공휴일 변수를 생성할 수 있었다.

그 외에 지하철요금 변수는 성인요금 기준으로 2023-10-01 이전에는 1250원, 이후는 1400원으로 적용하여 만들었다. USD-KRW 환율은 인베스팅닷컴에서 2021년부터 2023년까지의 데이터를 제공받아 기준 데이터에 결합하였으며, 결측치는 이전 날짜의 값으로 채웠다. "한국 소비자물가지수(CPI)"는 인베스팅닷컴에서 추출하여 기준 데이터에 붙였고 추가로 CPI 대비 지하철 요금이라는 변수를 생성하였다. "국내유가통계 > 주유소 > 평균판매가격"의 경우 오피넷에서 데이터를 다운받아 보통휘발유, 자동차용경우 일별 가격 데이터를 기준 데이터에 결합하였다. "기획재정부 > 취업자 수/실업률 추이"의 경우 통계청에서 제공하는 월별 실업률과 청년 실업률을 기준 데이터에 결합하였다. "기획재정부 > 경기종합지수" 역시 통계청에서 제공하는 월별 동행지수 순환변동치와 월별 선행지수 순환변동치 데이터를 다운받아 기준 데이터에 붙일 수 있었다. "통계청 > 지역별 인구 및 인구밀도"도 통계청에서 연도별 서울 인구와 서울 인구밀도, 수도권 인구, 수도권 인구밀도 데이터를 다운받아 기준 데이터에 결합하였다. "국토교통부 > 자동차 등록 현황"은 국토교통부에서 제공하는 연도별 자동차 등록대수 데이터를 결합하였다. "금융위원회 > 시장금리 추이"는 한국은행 경제통계시스템(ecos)에서 제공하는 월별 기준 금리 데이터를 기준 데이터에 결합하였다. 마지막으로 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 생성하였다.

생성된 공공빅데이터 기반 융합DB에서 종속변수를 예측하기 위해 나머지 113개의 독립변수들을 사용하여 모델링 할 때 알고리즘이 이해할 수 있는 형태로 전처리 후 반영해야 한다. 전처리 과정에서 불필요한 변수들을 삭제하기도 하고 문자로 된 응답값은 별도의 변수나 숫자로 변환하며 최종적으로 정리된 숫자들은 응닶값의 범위를 맞추기 위해 특정 범위로 스케일을 맞춘다(Scaling). 마지막으로 평균적으로 역마다 약 5.49만개 정도의 samples를 8:2의 비율로 구분하여 Train and Test set으로 분리하였다. Train은 모델링을 위한 학습에 사용되고 모델의 hyperparameters를 결정하기 위해 MSE???가 최대가 되는 방향으로 설정하였다. 그리고 Test는 미래 데이터로 가정하여 학습된 모델이 미래에 얼마나 지하철혼잡도를 잘 예측하는지 평가하는데 사용된다. 기본적으로 지하철 혼잡도가 어떤 원인들로 설명되더라도 그 해석이 미래 고객들이 이용할 지하철의 혼잡도를 실제로 잘 예측하는 설명이어야 고객들에게도 관련 의사결정자들에게도 정책적 활용가치가 높을 것이다. 이해도를 높이기 위해 전체적인 융합 DB 생성 과정을 제시한다 (그림 2).

그림 2. 데이터 모델링을 위한 전처리 과정

* 1. *Machine Learning Algorithm: Bagging and Boosting*

기본적으로 머신러닝 알고리즘은 예측 오차를 줄이는 방향으로 설계되었다. 오차는 bias and variance로 분리될 수 있는데, 예측의 안정성에 초점을 두어 variance를 줄이기 위해 샘플링 기법을 활용하는 Bagging과 성능에 초점을 두어 bias를 줄이기 위해 반복적인 모델링을 활용하는 Boosting으로 구분될 수 있다. Bagging은 여러 개의 샘플을 추출하여 각 샘플마다 모델링한 후 outputs의 투표로 최종 output의 label을 결정하는 방식이기에 bootstrap aggregating으로 불린다. 본 연구에 활용된 대표적 알고리즘은 Random Forest(RF)다. RF는 samples를 병렬처리로 한꺼번에 모델링 할 수 있기 때문에 속도도 빠르다며 training 데이터의 학습 성능이 우수한 편이고 noise data에도 성능이 크게 변하지 않는 robust algorithm이다 (Michie, Spiegelhalter, and Taylor 1999). 변수의 중요도를 제공하긴 하지만 각 샘플별 모델링 과정에서 각 변수들의 우선순위를 평균한 것으로 positive or negative와 같은 영향력의 방향성을 포함하지는 못하는 단점이 있다.

Bagging처럼 Boosting도 classification and regression 문제 모두 활용할 수 있는 supervised learning이다 (Chen and Guestrin 2016). Gradient boosting methods(GBM)를 기반으로 Extreme gradient boosting (XGBoost) and LightGBM, CatBoost 등의 알고리즘으로 확장되며 단점들이 개선되고 다양한 기능이 추가되었다. 본 연구에서는 XGBoost, LightGBM and CatBoost를 대표적인 boosting 알고리즘으로 활용하여 자하철 혼잡도 예측에 사용한다. 이 알고리즘들은 전체 데이터를 학습할 때 잘 학습되지 않은 에러들을 가중치를 높여 재학습을 하면서 성능을 향상시키는 방향으로 설계되었다. 메모리를 효율적으로 사용하거나 computation 속도를 높이기 위한 많은 기능들이 추가되었고, iterative learning과 병렬처리 과정에서 성능이 높아질 수 있었다 (Alsubari et al. 2021).



그림 3. 배깅(Bagging)과 부스팅(Boosting) 알고리즘의 구조 비교 (Cha, Moon, and Kim 2021).

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

딥러닝은 인공지능의 한 방법론으로, 인간의 두뇌 구조에서 영감을 받아 개발되었다. 데이터의 복잡한 패턴들을 학습하기 위해 연속된 layer를 중첩하여 변수들의 모든 상호작용을 포함하여 의미있는 규칙들을 학습해 내는데 강점이 있다. 이러한 구조의 기본이 되는 알고리즘으로 multilayer perceptron (MLP)가 있으며, 이미지나 시계열 등의 데이터도 학습해 낼 수 있도록 layer의 구조나 흐름을 개선하여 convolutional neural network (CNN)과 recurrent neural network (RNN) 등으로 확장되었다. neural network structure를 가진 모든 알고리즘들은 앞서 소개한 머신러닝의 알고리즘 처럼 activation function만 선택적으로 변경함으로써 classification and regression 문제 모두에 활용가능한 universal approximator이다. 모델의 구조가 복잡하기에 수많은 가중치들을 한꺼번에 추정해야 하는데 feed-forward and backpropagation 을 사용하여 loss function을 최소화하기 위해 gradient descent optimizer를 사용하였다 (see Figure 2) (Viswavandya, Patel, and Sahoo 2021). 머신러닝과 마찬가지로 다양한 nonlinear patterns를 학습함으로써 기존의 알고리즘보다 성능이 대폭 향상되어 인간이 인지하기 어려운 것들도 학습해내는데 유용하다.



그림 4. 대표적인 딥러닝 알고리즘인 MLP의 구조와 데이터 학습을 통한 수많은 가중치 추정 과정

본 연구에서는 MLP and CNN 알고리즘을 활용하여 지하철 혼잡도를 예측한다. 특히 CNN은 이미지 분류 특화 알고리즘이지만 내부 구조가 결국 feature를 요약하여 것이기 때문에 regression 문제에도 활용이 가능하다. Feature를 요약하기 위해 convolution, pooling, and fully connected layers를 활용하여 인접한 feature values들의 관련성에 더욱 초점을 두어 학습한다. 이러한 과정에서 여러 개의 neurons로 구성된 layer들을 계속적으로 통과하는데, 결국 low-level feature vector에서 점차 high-level feature vector로 변수들의 특성이 변환되어 학습된다 (Shustanov and Yakimov 2017). 따라서 다양한 변수들의 상호작용을 포함한 특징들이 학습될 것이고 마지막 output은 지하철 혼잡도 값이 출력되며 regression 문제를 해결한다.

* 1. *MLP 기반 알고리즘*

*2.4.1* NBEATS (2019)

N-BEATS(Neural Basis Expansion Analysis for Interpretable Times Series Forecasting) 모델은 벤치마크 통계 기반 모델보다 11%, 2018년 M4 대회 우승 모 델보다 3%의 예측 정확도를 높였다 [23]. 단변량 예측 모델인 N-BEATS는 블랙박 스로 알려진 딥러닝 모델과는 달리, Trend stack과 Seasonality stack을 사용하여 해석 가능하며, 시계열 예측 문제에서 딥러닝 구조를 통해 우수한 성능을 보인다는 특징이 있다 [24]. N-BEATS의 구조는 [그림 1]과 같이 여러 개의 basic block이 쌓여 하나의 stack이 되며, 여러 개의 stack이 모델을 구성한다

이 모델에서는 doubly residual stacking으로 불리는 변형 residual connection을 forecast와 backcast에 적용하여 gradient의 흐름을 투명하게 만들었다. 이 모델에는 generic architecture와 interpretable architecture가 있는데, basic block의 함수에 따라서 구분된다. interpretable architecture는 Trend stack과 Seasonality stack이 순서대로 쌓여있으며 basic block의 g b , g f함수가 단조 증가 함 수 또는 주기적 함수이다. Trend block의 함수는 단조 증가 함수이고 Seasonality block의 함수는 주기적 함수이다. Seasonality block의 g b , g f는 푸리에 급수를 사용하여 계절성을 모형에 반영한다. 그 결과, Seasonality block의 주기함수 출력값은 계절성을 보인다. Trend block의 g b , g f는 시간 t에 대해서 단조 증가 또는 단조 감소하는 함수이 다. 이를 통해 추세를 모형화하며, Trend block의 주기함수에서 생성된 출력값은 추 세를 보이게 된다.

*텍스트, 도표, 스크린샷, 평면도이(가) 표시된 사진

자동 생성된 설명*

그림 5. N-BEATS 구조

N-BEATS(Neural Basis Expansion Analysis for Interpretable Time Series Forecasting)는 단변량 시계열 예측을 위해 설계된 딥러닝 기반 알고리즘으로, 기존 통계적 기법과 하이브리드 모델보다 높은 예측 성능을 달성한 모델이다. N-BEATS는 Fully Connected(FC) Layer를 활용한 MLP 기반 구조로, 시계열 데이터의 패턴을 학습하고, 이를 바탕으로 과거 데이터를 복원(Backcast)하고 미래를 예측(Forecast)한다. 이 알고리즘은 여러 개의 **Basic Block**이 쌓여 **Stack**을 형성하며, 각 Stack이 데이터를 계층적으로 처리해 고차원적 특성을 학습하는 방식으로 작동한다. 특히, Stacks 간의 연결에는 **Doubly Residual Stacking**이라는 변형된 잔여 연결 방식을 적용하여 Gradient 흐름을 개선하고 학습의 안정성을 높이는 데 기여한다 (Oreshkin et al., 2020).

N-BEATS의 또 다른 주요 특징은 **Interpretable Architecture**를 통해 예측 결과의 해석 가능성을 제공한다는 점이다. 모델은 시계열 데이터를 **Trend Stack**과 **Seasonality Stack**으로 분리하여 학습하며, 각 Stack은 서로 다른 방식으로 데이터의 특성을 모델링한다. Trend Stack은 시간에 따른 데이터를 단조 증가 또는 감소 함수로 학습하여 데이터의 장기적인 변화를 포착한다. 반면, Seasonality Stack은 푸리에 급수(Fourier Series)를 활용해 주기적 변화를 학습하며, 계절성을 반영한 예측 결과를 제공한다. 이러한 구조는 모델의 결과를 단순히 수치로 제시하는 것이 아니라, 데이터의 주요 구성 요소를 시각적으로 이해할 수 있도록 돕는다.

N-BEATS는 **M4 대회**에서 기존의 통계적 방법과 딥러닝 하이브리드 모델을 능가하는 성능을 보였다. 벤치마크 통계 모델 대비 11%, 2018년 대회 우승 모델 대비 3% 더 높은 정확도를 기록하며, 딥러닝 기반 알고리즘이 시계열 예측 문제를 해결하는 데 있어 강력한 잠재력을 가지고 있음을 입증했다

*2.4.2* NBEATSx (2021)

NBEATSx(2021)는 NBEATS 알고리즘의 기본 구조를 확장하여 외생 변수(Exogenous Variables)를 통합한 모델이다. NBEATSx는 기존의 Backcast(과거 복원)와 Forecast(미래 예측) 모듈을 유지하면서, 외생 변수와 시계열 데이터 간의 관계를 학습해 더 정교한 결과를 도출한다. 이러한 구조는 외부 환경 요인을 함께 고려해 예측 정확도를 크게 향상시키며, 다양한 시계열 데이터 문제를 해결하는 데 적합하다.

*2.4.3* NHITS (2021)

NHITS (2021)는 NBEATS 구조를 발전시켜 **Hierarchical Interpolation** 기술을 도입한 시계열 예측 알고리즘이다. 이 모델은 시계열 데이터의 고주파수 및 저주파수 신호를 효과적으로 처리하며, 데이터의 세부 정보를 더욱 정교하게 학습할 수 있도록 설계되었다. Interpolation 기법을 활용하여 고해상도 시계열 데이터를 보완하고, 데이터의 미세한 패턴을 학습함으로써 높은 정밀도의 예측을 가능하게 한다. NHITS는 다양한 주파수 대역의 데이터를 처리할 수 있는 능력이 뛰어나며, 고해상도 시계열 데이터 분석에서 특히 우수한 성능을 발휘한다.

*2.4.4* TiDE (2023)

TiDE (2023)는 Transformer 구조와 딥러닝 기법을 결합한 최신 시계열 예측 알고리즘이다. 이 모델은 Attention 메커니즘을 활용한 Temporal Fusion Layer를 통해 시간적 변화와 상호작용을 효과적으로 학습하며, 과거와 미래 데이터 간의 상관성을 추론하는 데 초점을 맞춘다. TiDE는 복잡한 시계열 데이터의 다양한 패턴과 변동성을 유연하게 학습할 수 있는 강점을 가지고 있다. 특히, 데이터의 중요 시점을 선택적으로 학습하여 예측 성능을 극대화하며, 복잡한 시간적 상호작용을 처리하는 데 뛰어난 성능을 보여준다..

*2.4.5* DeepNPTS (2023)

DeepNPTS (2023)는 확률적 접근법(Probabilistic Approach)을 기반으로 한 시계열 예측 알고리즘으로, 예측 결과의 불확실성(Uncertainty)을 효과적으로 관리하는 데 중점을 둔다. 이 알고리즘은 Bayesian 접근법과 딥러닝의 강력한 학습 능력을 결합하여, 시계열 데이터의 패턴 학습과 불확실성 추정의 균형을 맞춘다. DeepNPTS는 불확실성을 모델링함으로써 예측 신뢰성을 높이고, 복잡한 시계열 데이터에 대한 신뢰성 있는 예측 결과를 제공한다.

* 1. *RNN 기반 알고리즘’*

*2.5.1 LSTM(1997)*

LSTM(Long Short-Term Memory, 1997)은 RNN(Recurrent Neural Network)의 한 종류로, 장기 의존성 문제(long-term dependency problem)를 해결하기 위해 설계된 모델이다. 셀 상태(cell state)와 게이트 구조(입력 게이트, 삭제 게이트, 출력 게이트)를 활용하여 중요한 정보를 선택적으로 저장하거나 삭제할 수 있어, 장기간의 시계열 패턴을 효과적으로 학습할 수 있다. 이러한 특성 덕분에 기상 예측, 주가 예측, 기계 상태 감지 등 다양한 시계열 예측 문제에 널리 사용된다.

*2.5.2 GRU(2014)*

GRU(Gated Recurrent Unit, 2014)는 Recurrent Neural Network(RNN)의 확장 모델로, LSTM(Long Short-Term Memory)의 경량화된 변형이다. LSTM이 장기 의존성 문제를 해결하기 위해 복잡한 게이트 구조(입력 게이트, 삭제 게이트, 출력 게이트)를 사용하는 반면, GRU는 단순화된 게이트 구조(업데이트 게이트와 리셋 게이트)를 채택하여 계산 효율성을 크게 향상시켰다. GRU는 적은 파라미터로도 효과적인 학습이 가능하며, 상대적으로 작은 데이터셋에서도 성능 저하 없이 빠른 학습 속도를 보인다. 이러한 특성으로 인해 GRU는 음성 인식, 번역, 금융 데이터 분석과 같은 다양한 시계열 예측 문제에서 자주 활용되고 있다. 또한, GRU는 단순한 구조 덕분에 하드웨어 자원에 제약이 있는 환경에서도 우수한 성능을 발휘한다.

*2.5.3 DilatedRNN(2017)*

Dilated Recurrent Neural Network(DilatedRNN, 2017)는 RNN(Recurrent Neural Network)의 시간적 한계를 극복하기 위해 설계된 모델이다. 전통적인 RNN 모델은 시간 순서를 따라 한 스텝씩 학습하는 방식으로, 장기적인 시간 의존성을 학습하는 데 한계가 있었다. DilatedRNN은 dilated(확장된) 연결을 도입하여 시간적으로 멀리 떨어진 정보를 효과적으로 학습할 수 있도록 설계되었다. Dilated 연결은 입력 데이터의 특정 시점 간 간격을 늘려 데이터를 처리함으로써 긴 시계열 패턴을 보다 효율적으로 학습할 수 있다. 이 모델은 특히 장기적인 시간 패턴을 분석해야 하는 문제에서 기존 RNN보다 뛰어난 성능을 보이며, 계산 효율성을 유지하면서 더 넓은 시간 범위를 처리할 수 있다는 장점이 있다. DilatedRNN은 시계열 데이터 분석, 시퀀스 모델링, 음성 신호 처리 등 다양한 분야에서 활용되고 있다.

*2.5.4 TCN(2018)*

Temporal Convolutional Network(TCN, 2018)는 시계열 데이터를 처리하기 위해 설계된 합성곱 신경망(Convolutional Neural Network) 기반 모델이다. TCN은 RNN 기반 모델과 달리 순차적으로 데이터를 처리하지 않고 병렬적으로 처리할 수 있어 계산 효율성이 높다. 또한, TCN은 dilated convolution을 활용하여 장기적인 시간 의존성을 학습하며, 패딩(padding) 기술을 통해 시계열 데이터의 순서를 유지하면서도 전체 시계열 범위에서 정보를 처리할 수 있다. TCN은 RNN 기반 모델에 비해 학습 속도가 빠르고, 기울기 소실(vanishing gradient) 문제를 효과적으로 해결할 수 있다는 장점이 있다. TCN은 다중 스텝 예측, 고해상도 시계열 데이터 처리, 시퀀스 모델링 등 다양한 응용에서 우수한 성능을 발휘하며, 특히 긴 시간 간격의 의존성을 학습해야 하는 문제에서 RNN보다 더 효과적이다.

*2.5.5 DeepAR(2019)*

DeepAR(2019)는 아마존(Amazon)에서 개발한 확률적 시계열 예측 모델로, RNN(Recurrent Neural Network)을 기반으로 각 개별 시계열 데이터의 확률 분포를 학습한다. 이 모델은 개별 시계열 간의 공통된 패턴을 학습하여, 조건부 확률 분포를 통해 다양한 시나리오에서 신뢰 구간을 포함한 예측을 수행한다. DeepAR은 개별 시계열 데이터뿐만 아니라 여러 시계열 데이터 간의 상관성을 활용할 수 있어, 다중 시계열 예측 문제에서 특히 효과적이다. 이를 통해 DeepAR은 기존 모델과 비교했을 때 높은 예측 정확도와 신뢰성을 제공하며, 전자상거래, 공급망 관리, 수요 예측 등 다양한 실세계 응용에서 널리 사용되고 있다. DeepAR은 미래 시점의 불확실성을 모델링하는 데 강점을 가지며, 실시간 운영 및 의사결정 과정에서도 유용하게 활용될 수 있다.

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

앞서 소개한 인공지능의 대표적인 알고리즘인 SHAP는 머신러닝과 딥러닝은 발생가능한 변수들의 모든 상호작용들을 스스로 생성하여 학습하기 때문에 인간의 수준을 뛰어넘는 성능 달성이 가능하게 하였다. 하지만 왜 그러한 결과가 발생한 것인지 쉽게 확인하거나 설명하기는 어려운 블랙박스 알고리즘이다. 일부 머신러닝 알고리즘이 변수들의 중요도(Feature Importance)를 출력하지만 방향성이 없기 때문에 해석에 주의를 기울여야 한다. 하지만 SHAP(Shapley Additive exPlanations)는 local interpretable model-agnostic explanations (LIME)과 shapley value를 연결한 이론으로, 이러한 블랙박스와 같은 알고리즘들의 한계를 보완하여 지하철 혼잡도 예측에 영향을 주는 변수들의 특징과 방향을 설명해준다. LIME은 주어진 데이터 값들을 변화시킬 때 모델 예측값의 변화와의 관련성을 가중치로 계산한다 (Molnar 2023). 그리고 shapley values는 coalitional game theory를 기반으로 변수들의 기여도를 계산하는 metric으로 생각할 수 있다. 따라서 변수들이 가질 수 있는 모든 coalitions 경우들을 생성한 후 실게 값들이 입력되었을 때 변화된 예측값의 기여도 평균을 계산한다. 하지만 연산량이 매우 많을 수 있기 때문에 랜덤 샘플링 기법을 활용 계산의 효율성을 높인다. 이 2가지 이론이 결합된 SHAP를 사용하여 지하철 혼잡도에 기여하는 변수들의 정도와 방향성을 알 수 있고 블랙박스와 같은 인공지능 알고리즘을 설명가능케 한다.

선형 기반의 단순한 기울기로 해석하는 전통적인 statistical algorithms는 우리가 직관적으로 이해하기는 쉽지만 샘플이 1개라도 변경되면 해석이 얼마든지 달라질 수 잇는 불안정성이 있고 실제 예측 성능도 낮아서 단순히 설명만 될 뿐 실제 비즈니스에 활용하기는 어렵고 조심해야 한다. 반면, 인공지능 알고리즘은 구조가 복잡하고 정교하지만 높은 정확성을 나타낸다. 따라서 인공지능 알고리즘을 사용하되 SHAP와 같은 설명가능한 기능들을 결합하며 설명력을 강화하기 위한 model explainability의 중요성은 점점 높아지고 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** | TensorFlow Random Seed | 0 |
| Number of Hidden Layers | 2 |
| Unit Numbers for Hidden Layers | (100, 100) |
| Activation for Hidden Layers | Relu |
| Activation for Output Layer | identity |
| Loss | Mean Squared Error |
| Optimizer | Adam |
| Epochs | 500 |
| **CNN** | TensorFlow Random Seed | 0 |
| Number of Convolution Layers | 1 |
| Unit Numbers for Convolution Layers | 64 |
| Number of Dense Layers | 2 |
| Kernel Size | 2 |
| Stride | 1 |
| Padding | Valid |
| Unit Numbers for Dense Layers | 64 |
| Activation for Hidden Layers | Relu |
| Activation for Output Layer | Linear |
| Loss | Mean Squared Error |
| Learning Rate | 0.001 |
| Optimizer | Adam |
| Batch Size | 32 |
| Epochs | 10 |

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

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

그림 6. 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 7. 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, 청소년인원수, 월평균\_환승유입인원수, 우대권인원수, 환승유입인원수, 섬식여부, 승하차인원차, 시간, 선행지수 순환변동치, 요일\_Saturday, 실업률, 요일\_Monday, 요일\_Tuesday" 등이며, negative correlation을 보이는 특징은 "환승노선\_개수, 하선 혼잡도, 상행\_평균운행간격, 기준금리, 달러환율, 주, 동행지수 순환변동치, 요일\_Sunday, 요일\_Friday, 요일\_Thursday, 지하철요금" 등으로 해석할 수 있다. 단순히 지하철 관련 변수들 뿐만 아니라 각 시간정보에 따른 혼잡도 기여, 그리고 경기상황이나 경제지표들의 영향력도 확인할 수 있으며 이것인 기존 연구들에서 발견하기 어려웠던 결과다. 총 17개의 DB 결합과 설명가능한 인공지능을 통해 확인할 수 있었다. 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.

|  |  |
| --- | --- |
| (a) 우선순위 1-10 | (b) 우선순위 11-20 |
| (c) 우선순위 21-30 | (d) 우선순위 31-40 |
| (e) 우선순위 41-50 | |

그림 8. 지하철 혼잡도 예측에 대한 변수들의 기여 설명력 기반 의사결정

# Conclusions

# References

Alsubari, Saleh Nagi, Sachin N. Deshmukh, Mosleh Hmoud Al-Adhaileh, Fawaz Waselalla Alsaade, and Theyazn 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, Gi-Wook, Hyeun-Jun Moon, and Young-Chan 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: 8530.

Chen, Tianqi, and Carlos Guestrin. 2016. "XGBoost: A Scalable Tree Boosting System." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–94. San Francisco, California, USA: Association for Computing Machinery.

Michie, D., D. Spiegelhalter, and Charles Taylor. 1999. 'Machine Learning, Neural and Statistical Classification', *Technometrics*, 37.

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

Shustanov, Alexander, and Pavel Yakimov. 2017. 'CNN Design for Real-Time Traffic Sign Recognition', *Procedia Engineering*, 201: 718-25.

Viswavandya, Meera, Shashwat Patel, and Kaushik 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.

Ahmed, Rafique, and Ali Shariq Imran. "Knee Osteoarthritis Analysis Using Deep Learning and XAI on X-rays." *IEEE Access*. 2024. doi:10.1109/ACCESS.2024.3400987.

Gunning, David, Mark Stefik, Jaesik Choi, Timothy Miller, Simone Stumpf, and Guang-Zhong Yang. 2019. "XAI - Explainable Artificial Intelligence." *Science Robotics* 4(37): eaay7120. <https://doi.org/10.1126/scirobotics.aay7120>.

Gao, X., & Lee, G. M. (2019). Moment-based rental prediction for bicycle-sharing transportation systems using a hybrid genetic algorithm and machine learning. Computers & Industrial Engineering, 128, 60-69.

International Union of Railways (UIC). 2023. *HIGH-SPEED RAIL, the right speed for our planet: Key Messages from the 11th UIC High-Speed Congress*. Paris: International Union of Railways.

KOSEN. 2018. "설명가능한 인공지능(XAI)의 기술 동향과 발전 방향." *KOSEN*. Accessed November 09, 2018. <https://kosen.kr/info/reports/REPORT_0000000001071>.

국토교통부. *국토교통 2050 미래기술 도출을 위한 조사분석 연구: 최종보고서*. 기술과가치, 2020.

김관형, 김한수. (2011). *개입 ARIMA 모형을 이용한 KTX 수요예측*. 한국철도학회 춘계학술대회 논문집, 1282-1289.

배재권. "설명가능한 인공지능(XAI) 방법론의 산업별 적용가능성에 관한 연구." 글로벌경영학회지 20, 2호 (2023): 195-208, 10.38115/asgba.2023.20.2.195.

은종환, and 황성수. "인공지능을 활용한 정책의사결정에 관한 탐색적 연구: 문제구조화 유형으로 살펴 본 성공과 실패 사례 분석." *정보화정책*, vol. 27, no. 4, 2020, pp. 55-56.

정연수, 윤철희. "XAI를 활용한 제조 공급망 데이터의 수주량 예측에 대한 연구." *정보기술학회논문지* 22, no. 8 (2024): 41-53. <https://doi.org/10.14801/jkiit.2024.22.8.41>.

Cha, D., Park, S., & Kim, H. (2019). *KTX Passenger Demand Forecast with Multiple Intervention Seasonal ARIMA Models*. Proceedings of the Korean Society for Railway Annual Conference, 44(1), 89-98.

심진호, 김태영, 박민재. (2024). *비선형 데이터 패턴 학습을 통한 고속철도 수송수요 예측 연구*. 한국철도학회 학술대회 초록집

박은경. (2024). *철도산업에서 AI 기반 예측 유지보수를 위한 사례 연구 및 시사점*. 한국산업정보학회 학술대회 논문집