

포트폴리오 - 원도현

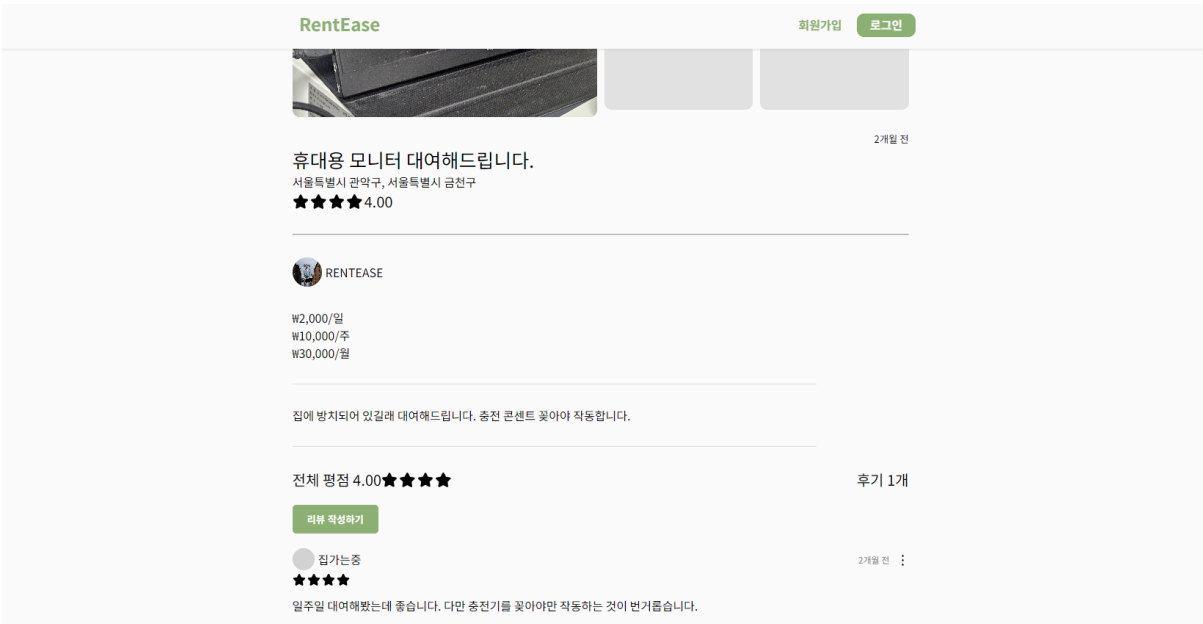
해당 포트폴리오는 이력서에 작성된 프로젝트의 상세 설명입니다. 이력서에 작성된 링크에서도 똑같은 내용을 확인할 수 있습니다.

- 1. [RentEase](#): 물품 대여 플랫폼
- 2. [TripWith](#): 지도 기반 여행 일정 관리 통합 서비스
- 3. [Coffee Price Prediction](#): 가격 예측 및 감성 분석 기반의 원두 가격 예측 시스템
- 4. [OTT Recommendation System](#): 사용자 메타데이터와 콘텐츠 임베딩 정보를 기반으로 개인화된 콘텐츠 및 요금제 추천 시스템

1. RentEase

개인 대 개인 렌탈 플랫폼

- 코드 링크: [1Dohyeon/RentEase: 개인 대 개인 렌탈 플랫폼 개발 프로젝트](#)
- 프로젝트 메인 이미지:



1.1. 프로젝트 개요

- 참가 인원: 개인 프로젝트
- 기간: 2024/06/18 ~ 2024/07/22 (약 1개월)
- 기술 스택: React, NestJS, MySQL

1.2. 프로젝트 배경(개발 동기)

현대인의 다양한 취미 활동을 위해서는 여러 장비가 필요합니다. 하지만 새로운 취미를 시작할 때마다 관련 장비를 구매하는 것은 상당한 경제적 부담이 되며, 보관을 위한 공간 확보도 쉽지 않습니다. 이러한 고민을 해결하고자 기존 업체 중심의 렌탈 서비스가 아닌, 개인 간 장비 대여 플랫폼을 기획하게 되었습니다.

RentEase는 이러한 비용/공간 문제를 해결하기 위해 개인 간 장비 대여 서비스를 제공합니다. 취미 생활에 필요한 장비들을 소유하지 않고도 합리적인 가격에 경험할 수 있도록 도와줍니다.

1.3. 프로젝트 설명

사용자는 자신의 물건들을 게시글에 올려 다른 사용자에게 빌려줄 수 있습니다. 상품 이미지, 렌탈 가격, 지역 등을 설정할 수 있습니다.

본인 프로필 지역 설정에 따라서 특정 지역 게시글들을 볼 수 있습니다. 또한 게시글을 통해 다른 사용자와 채팅을 시작하여 원활하게 거래할 수 있습니다.

1.4. 주요 구현 사항

- **사용자 인증:** NestJS + JWT 기반의 회원가입 및 로그인 API 개발
- **상품 등록/조회 기능:** 상품 CRUD API 설계 및 MySQL 연동
- **WebSocket 기반 실시간 채팅 기능** 구현으로 사용자 간 실시간 대화 지원
- **에러 핸들링 및 예외 처리** 로직 적용하여 안정적인 API 통신 구현
- **React 기반 UI 구성:** 사용자 경험을 고려한 상품 등록/조회/채팅 인터페이스 설계

NestJS 프레임워크를 공부 및 응용하기 위한 목표로 개발을 시작한 프로젝트입니다. 배포 단계까지는 진행하지 못했지만, http, socket 등 여러 기능을 접해보기 위해 진행한 프로젝트입니다.

1.5. 성과 및 배운 점

- 관계형 데이터베이스 설계
- NestJS 구조와 DI, 모듈화 설계 방식에 대한 이해
- RESTful API 설계 원칙 및 상태 관리, HTTP 통신 구조
- 실시간 통신(WebSocket) 구조 및 예외처리 기법 학습

2. TripWith

지도 기반 여행 일정 관리 통합 서비스

- 서버 코드 링크: [tripwith-dev/rest-nestjs](https://github.com/tripwith-dev/rest-nestjs)
- UI 코드 링크: [tripwith-dev/ui-react](https://github.com/tripwith-dev/ui-react)

2.1. 서비스 소개

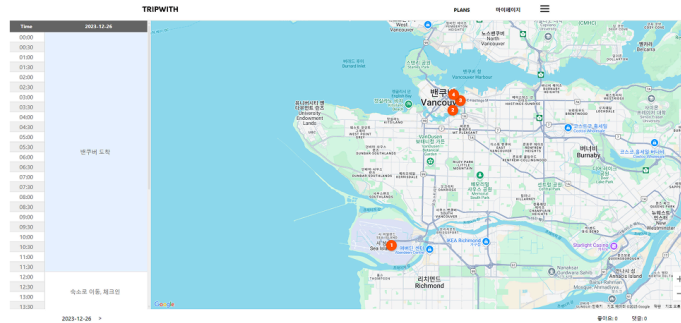
테이블 형태의 UI와 지도를 한 페이지에서 보며 계획을 세울 수 있는 플래너

- 프로젝트 제목: Tripwith
- 소요 기간: 약 3개월 (release-0.1.1)
- 개발 인원: 1인(+ 디자인 분야는 지인의 도움을 받음.)
- 기술 스택: NestJS, MySQL, React, Git, AWS EC

- 다른 사용자들의 다양한 (PUBLIC)플랜을 조회할 수 있으며, 이를 참고하여 계획을 세울 수 있다.

- 댓글 기능으로 서로의 플랜에 피드백을 남기며 소통 가능.(현재는 API만 개발 상태)

- 커뮤니티 공간을 통해 사용자는 여행 도중 혹은 이후에 여행 관련 소통을 할 수 있으며, 실시간으로 여행지에 대한 정보를 공유 가능.(현재는 API만 개발 상태)



TRIPWITH는 여행자를 위한 종합 플래닝 플랫폼입니다. 직관적인 테이블 형태의 인터페이스와 통합된 지도 시스템을 통해 효율적인 여행 계획 수립이 가능합니다.

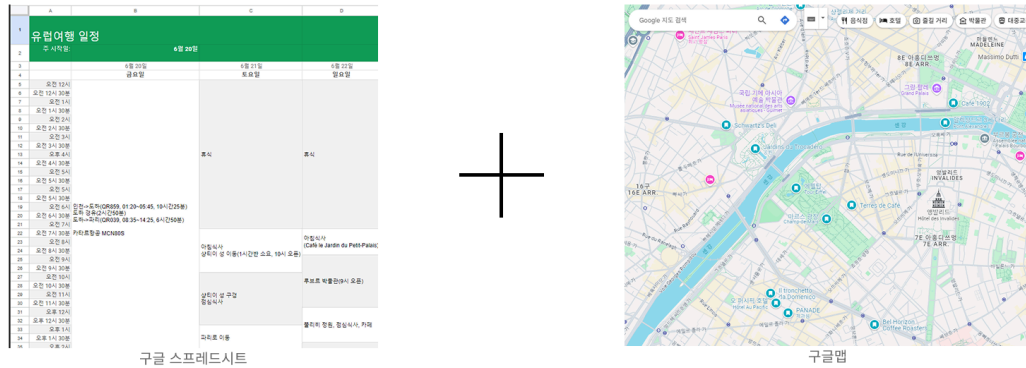
이용자는 자신만의 여행 계획을 커뮤니티와 공유하고, 다양한 피드백을 수렴할 수 있습니다. 또한 다른 여행자들의 계획을 참고하여 자신만의 특별한 여정을 구상할 수 있습니다.

아울러 실시간 커뮤니티 기능을 통해 여행 전후는 물론 여정 중에도 다른 여행자들과 소통할 수 있습니다. 이를 통해 더욱 풍요로운 여행 경험을 제공하고자 합니다.

2.2. 기획 배경

엑셀과 지도 창을 따로 띄워서 계획을 세우기 너무 불편해서 시작한 프로젝트

2. 프로젝트 배경



- 계획과 지도를 같이 볼 수 있는 서비스도 있지만, 원하는 형태의 UI가 아니었기에 다시 엑셀 또는 시트로 돌아가는 경우가 있었음.
- 따라서 단일 페이지에서 테이블 형태의 UI와 지도를 함께 볼 수 있는 웹 서비스를 개발하기로 결정.

여행 애호가로서 다양한 플래너 서비스를 경험하며 발견한 불편사항을 개선하고, 필수적인 기능을 보완하고자 본 프로젝트를 시작하게 되었습니다.

현재 시중의 플래너들은 테이블 형태의 계획 수립과 지도 확인을 동시에 할 수 없다는 한계가 있었습니다. 이에 **통합형 인터페이스**를 구현하여 사용자 경험을 향상시키고자 하였습니다.

또한 여행 계획에 대한 즉각적인 피드백 시스템을 도입하여, 더욱 완성도 높은 여정 설계가 가능하도록 하였습니다.

코로나 시대 이후 개별 여행이 증가하는 추세입니다. 독립적인 여행의 가치를 존중하면서도, 필요한 순간에는 여행자들과의 교류가 가능한 플랫폼의 필요성을 인식하였습니다.

특히 여행지의 최신 정보가 중요한 개별 여행자를 위해, 실시간 정보 교류가 가능한 커뮤니티 기능을 구현하여 여행의 질적 향상을 도모하고자 합니다.

2.3. 프로젝트 기능

2.3.1. 주요 기능 목록

1. 이메일 기반 로그인
2. 회원 프로필 관리 (프로필 이미지, 닉네임, 이름, 비밀번호)
3. 마이페이지 내 카테고리별 여행 계획 관리
4. 지도와 연동된 직관적인 일정 관리 인터페이스
5. 테이블 형식의 상세 일정 작성 기능

테이블 UI 설계 원칙

01:30	숙소 도착
02:00	
02:30	
03:00	

시간 단위는 30분을 기본으로 설정했습니다. 더 짧은 시간 단위를 적용할 경우 화면에 표시되는 칸이 과도하게 많아지는 문제가 발생합니다.

이러한 시간 설정은 여유로운 일정 계획을 자연스럽게 유도한다는 장점이 있습니다. 다만, 보다 유연한 시간 설정이 필요한 경우를 위해 기본 플래너 모드도 추가로 제공할 예정입니다.

2.3.2. 소통과 공유 기능

- 여행 계획 공개/비공개 설정
- 공유된 계획에 대한 반응 (좋아요, 댓글)
- 여행 관련 자유 게시판
- 태그 기반 게시물 분류
- 태그 검색 기능

2.4. 성과 및 배운점

- NestJS 기반 서버 설계 경험과 프론트-백 간 연동 실습
- 협업 기반 개발 (Git, 코드 리뷰, 이슈 관리) 경험
- 반복적인 코드 리팩토링을 통해 모듈화/객체화의 중요성을 체득하고 유지보수성을 고려한 구조 개선 경험
- 상태 관리 및 동적 UI 구성 실무 감각 향상

2.5. 참고 자료

- [한국관광 데이터랩: 개별여행 트렌드 분석](#)
- [한국관광 데이터랩: 여성 개별여행 증가 현황](#)

3. Coffee Price Prediction

커피 생두 가격 예측

- 링크: [MJU-Capstone-2025 repositories](#)

3.1. 프로젝트 개요

- 참가 인원: 6인 프로젝트
- 기간: 2025/03 ~ 2025/06 (4학년 1학기)

- 기술 스택: React, FastAPI, Pytorch
- 담당 파트: 데이터 엔지니어링, 모델 서빙, 풀스택

본 프로젝트는 커피 생두(Coffee Green Bean)의 가격을 예측하는 시계열 기반 머신러닝 모델을 개발하는 것을 목표로 합니다. 본 모델은 커피 원두 유통업체, 로스터리, 카페 프랜차이즈 운영자 등 커피 산업 종사자들이 시장의 변동성에 대응하고 전략적 의사결정을 내릴 수 있도록 지원하는 도구로 확장 가능합니다.

프로젝트는 주요 목표는 다음과 같습니다.

1. 다변량 시계열 모델 설계: 생두 가격에 영향을 미치는 기후 데이터(기온, 강수량), 환율(생산국 통화), 국제 유가 등 복합적인 글로벌 지표를 반영한 예측 모델 개발
2. 비전문가도 활용 가능한 인터페이스 개발: 업계 종사자가 직관적으로 사용할 수 있는 시각화 및 예측 결과 제공 시스템 구축
3. 모델 성능 고도화: 다양한 알고리즘 비교 및 최적화를 통한 예측 정확도 향상

본 프로젝트는 수집, 분석, 예측을 모두 실시간으로 자동화하는 것이 최종 목표였지만, 실시간 데이터 수집의 경우 api 비용 발생 이슈로 수집 가능한 범위의 데이터로 분석, 예측을 진행하였습니다. 데이터 가공과 모델 학습 부분은 자동화하였기에, 실시간 데이터 수집만 된다면 초기 목표로 확장 가능함하다.

3.2. 프로젝트 배경

생두는 대표적인 국제 원자재로서, 그 가격은 생산국의 기후 변화, 국제 정세, 물류 비용, 환율 등 다양한 외부 요인에 따라 높은 변동성을 보입니다. 이러한 가격 불안정성은 수입업체 및 유통업체의 경영 리스크를 증가시키며, 궁극적으로는 소비자 가격에 영향을 미칠 수 있습니다. 최근에는 이러한 생두 가격의 향후 흐름을 정밀하게 예측하고자 하는 산업적 수요가 높아지고 있습니다.

정확한 가격 예측은 수입업체의 계약 전략, 로스터리의 원재료 조달 계획, 카페 운영자의 가격 정책 등 실무 전반에 걸쳐 중요한 의사결정 기반이 됩니다. 또한 예측 모델이 제공하는 데이터는 소비자 가격 안정화에도 기여할 수 있습니다.

본 프로젝트는 머신러닝 기반의 정량 분석을 통해 업계 종사자에게 신뢰도 높은 예측 정보를 제공함으로써, 급변하는 시장 환경 속에서 실질적인 대응력을 높이는 데 기여하고자 합니다.

3.3. 프로젝트 설계 요약

3.3.1. 데이터 수집 및 가공(공동 구현)

3.3.1.1 데이터 수집

NASA 기후 데이터와 yfinance를 활용하여 기후 및 경제 데이터를 수집하였습니다. 이때 커피 재배 주요국인 브라질, 콜롬비아, 에티오피아의 기후 데이터만을 수집하였습니다. 기사 데이터는 Selenium을 활용하여 FAO, USDA, World Coffee Portal 등에서 커피 관련 뉴스를 수집하였습니다.

3.3.1.2 데이터 가공 및 데이터의 한계점

기후 데이터와 거시 경제 데이터를 하나의 테이블로 통합하는 작업을 수행하였습니다. 그러나 기후 데이터는 일별 단위인 반면, 거시 경제 데이터는 월별 또는 연도별로 구성되어 있어 해상도 차이의 문제가 있었습니다. 이러한 차이를 보완하기 위해, 월/연도 단위 데이터를 단순히 일별로 복제하여 직선적으로 표현하는 대신, **스플라인 보간(Spline interpolation)** 기법을 활용하여 보다 자연스러운 곡선 형태로 변환하였습니다.

입력 데이터의 품질 한계점으로는, 일부 NASA 기후 데이터 컬럼에서 최근 연도 정보가 누락된 현상이 관찰되었으며, NASA가 제공하는 위성 기반 기후 데이터는 전 지구 평균값을 기반으로 하기 때문에 실제 재배 환경에서의 미세한 지형이나 고도 차이를 충분히 반영하지 못하는 한계가 있었습니다.

3.3.2 기후/경제 데이터 활용: LSTM + Attention 기반 가격 예측 모델 구현(기획/리뷰 참여)

3.3.2.1 파생 변수 생성

기후/경제 데이터뿐 아니라 다음과 같이 파생 변수를 생성하였습니다.

- `Coffee_Price_Return` : 일간 수익률
- `abs_return` : 절대 수익률 크기
- `volatility_5d/10d` : 최근 5/10일간 변동성
- `momentum_1d/3d/5d` : 최근 1, 3, 5일간 순가격 상승폭
- `bollinger_width` : 상대 변동성
- `volatility_ratio` : 단기/중기 변동성 비율

또한 `재배지역명_PRECTOTCORR_harvets_mean` (재배지역별 비수확기 강수량 평균) 파생변수를 생성하였습니다. 이는 매 수확기 이전의 기후 상태를 반영하여 직전 비수확기 동안의 평균 강수량을 정적 변수로 저장하여 사용한 것입니다.

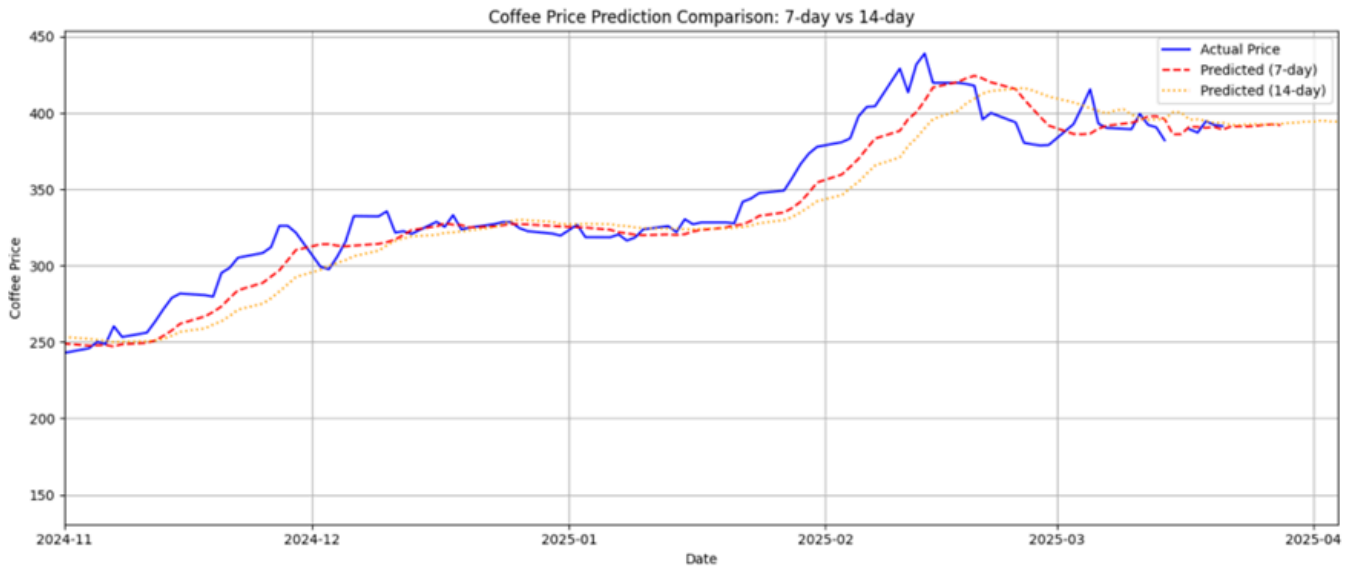
3.3.2.2 최종 모델 선정

최종 모델로는 **Attention 기반 LSTM 모델**을 선정하였습니다. 기존 LSTM 구조에 Attention 메커니즘을 결합함으로써, 시계열 데이터 내 예측에 중요한 정보를 보다 효과적으로 추출할 수 있도록 구성하였습니다. 기존 LSTM이 마지막 hidden state를 기반으로 예측을 수행하는 반면, Attention은 시퀀스 전반에서 중요도에 따라 가중치를 부여하여 가격 예측의 정밀도를 높이는 데 기여하였습니다.

입력 시퀀스는 기본적으로 예측일 이전 **100일치 데이터**로 설정하였으며, 각 입력 시퀀스에 대해 **7일 및 14일 예측**을 각각 수행하였습니다. 두 모델은 상호 보완적으로 활용되어, **단기 이상 징후에 대한 선제 대응**과 **중기적 추세 판단**을 모두 가능하게 하는 것을 목표로 하였습니다.

모델	목적	역할
7일 예측 모델	민감하게 반응하여 선제 대응	조기 경보 역할
14일 예측 모델	지속적인 흐름인지 여부 확인	추세 확정 역할

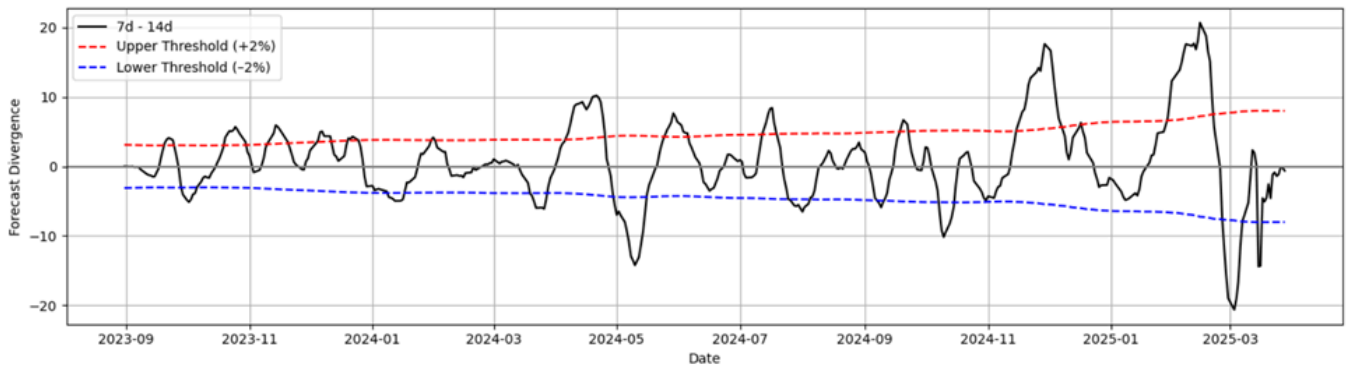
기존 목표는 장기적인 가격 흐름을 예측하는 것이었습니다. 실제로는 **상승·하락세는 어느 정도 예측 가능했지만, 급격한 변동성은** 모델이 제대로 반영하지 못하는 한계가 있었습니다. 이에 따라 논의를 통해 예측 기간을 7일과 14일로 분리하고, 두 모델을 **병렬적으로 운용**하는 전략으로 전환하였습니다.



3.3.2.3 예측 분기 감지

최근 4일 평균 대비 예측값이 각각 +0.7% 이상 상승할 경우 'Buy' 신호를 발생시키는 규칙 기반 전략을 설정하였습니다. 해당 신호는 실제 매매로 연결하지 않고 **검증 단계까지만 진행**하였습니다.

또한 두 예측(7일, 14일) 간 차이가 커질 때 예측 불일치 구간으로 판단하였습니다. 분기 신호는 확정된 매수/회피 판단이 아닌 보수적 접근 또는 판단 보류를 유도하는 경고성 보조 시그널입니다. 사용자가 판단하려는 시점이 임계값을 넘은 시점이라면 예측 결과에 대해 보수적인 판단을 진행합니다. 빨간 점선 초과 시 단기 예측이 과도하게 높다고 판단하며, 파란 점선 하회 시 단기 예측이 과도하게 낮다고 판단합니다.



Buy 신호가 발생한 시점에 매수하고 이를 일주일 보유했을 때의 누적 수익률이 어떻게 변하는지 시간 순서대로 시각화하면 다음과 같습니다.



3.3.2.4 모델 검증

- 14일 예측 모델 성능

RMSE	MAE
12.9053	8.9388

- 7일 예측 모델 성능

RMSE	MAE
9.1426	6.6435

3.3.3 감성 분석 모델 설계(리뷰 참여)

수집된 기사의 title 정보를 활용하여 키워드 기반 필터링으로 생두 가격과 연관된 문장을 선별하였습니다. 이후 FinBERT를 활용하여 각 문장의 감성을 분류하고 수치화된 점수를 날짜별로 집계하였습니다. 이렇게 생성된 감성 feature는 커피 가격 등의 정형 데이터와 결합하여 예측 모델의 입력 테이블로 활용하였습니다.

3.3.4 모델 서버: FastAPI(직접 구현)

위 3.2에서 개발한 모델을 서버에 서빙하였으며, 3.3 모델의 결과는 참고 지표로 조회하는 API를 구현하였습니다. 데이터베이스는 따로 사용하지 않고 모델의 결과를 CSV로 저장한 후, FastAPI 프레임워크를 활용하여 배포하였습니다.

모델의 결과는 시계열 LSTM 모델의 7일 결과, 14일 결과 두 가지를 반환합니다. `/prediction-dev` 엔드포인트를 통해 예측 결과를 조회할 수 있으며, 각 예측 결과는 날짜(Date), 예측 가격(Predicted_Price), 실제 가격(Actual_Price)을 포함합니다. 7일과 14일 예측 결과를 동시에 제공하여 다양한 기간의 예측 성능을 비교할 수 있도록 구성하였습니다.

감성 분석 모델 결과도 /news 엔드포인트를 통해 조회할 수 있습니다. 기사의 title, URL, date, 감성 분석 결과(rise, neutral, fall) 외에도 데이터를 같이 반환합니다. 시스템 UI에서는 언급한 특성들만 활용합니다.

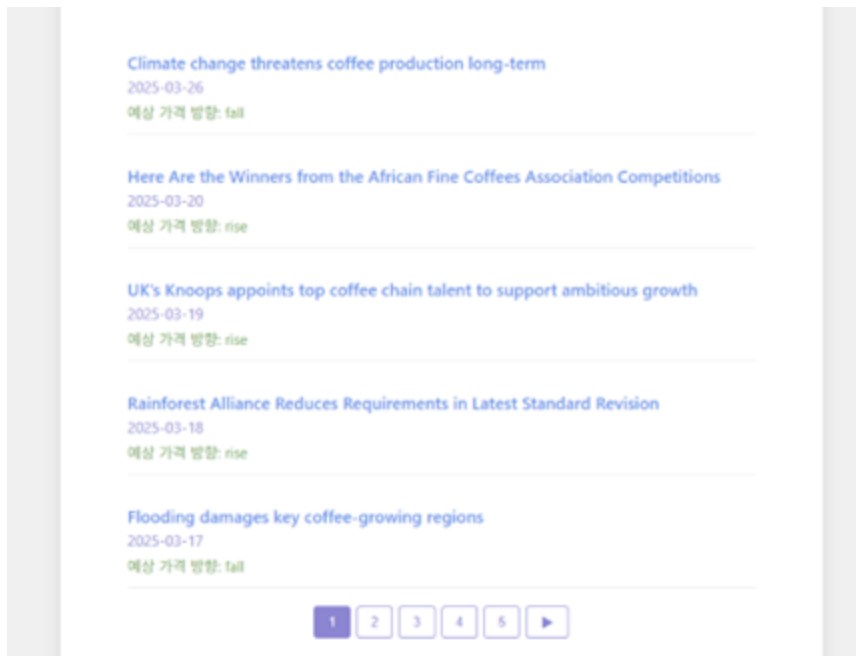
3.3.5 프로젝트 View(직접 구현)

UI 구축은 React 라이브러리를 활용하였습니다. 위 3.4 모델 서버의 응답을 활용하여 가격을 그래프로 시각화하였으며, 아래와 같이 보여줍니다.



2주~1년 버튼은 하나의 화면상 그래프의 최대 X축 비율을 조절하는 기능입니다. 실제 가격과 7일, 14일 예측 가격을 한눈에 비교할 수 있습니다. 구매 및 경고 신호는 신뢰도 검증만 진행하였으며 UI에는 구현하지 않았습니다. 검증을 통해 수익성을 확인하였으며, 그 결과는 3.2.3에서 확인할 수 있습니다.

감성 분석 모델의 결과는 UI에서 중립 감성을 제외하고 상승 및 하락 예측 결과만을 필터링하여 표시하였습니다. 각 뉴스 기사의 타이틀을 클릭하면 해당 기사의 원본 URL로 이동할 수 있도록 구성하였습니다. 전체 기사 목록은 최신순으로 정렬되어 있어 최근 시장 동향을 우선적으로 확인할 수 있도록 하였습니다.



3.4. 성과 및 배운 점

- 시계열 예측 모델 및 감성 분석 모델 개발 실습
- ML 모델을 실서비스에 통합하는 MLOps 개념 및 FastAPI 기반 서빙 경험
- 데이터 전처리 자동화와 배치 처리 개념 이해

4. OTT 추천 시스템

콘텐츠 1~8개와 해당 콘텐츠들을 방영하는 OTT들을 추천하는 시스템

- 링크: [2025-MJU-Social/RecommendModel_withFastAPI](#)

4.1. 프로젝트 개요

- 참가 인원: 3인 프로젝트
- 기간: 2025/05 ~ 2025/06 (약 6주)
- 기술 스택: React, FastAPI, Pytorch
- 담당 파트: 시스템 개발, 풀스택

오늘날 국내 OTT 시장은 수평, 수직적으로 확대되고 있으며, 소비자 입장에서는 이를 모두 구독하기 어렵다는 단점이 존재합니다. 이에 따라 사용자 취향에 맞는 콘텐츠를 추천하면서, 최대한 많은 추천 콘텐츠들을 사용자의 OTT 사용 시간 내에 시청할 수 있도록 효율적으로 OTT 구독을 관리할 수 있는 시스템을 설계하였습니다.

4.2. 프로젝트 설계

4.2.1 사용된 기술

시스템에 적용된 핵심 기술과 이론은 세 가지입니다.

첫 번째는 **자연어 처리 기술**입니다. Sentence Transformer라는 최신 NLP 모델을 사용하여 '스릴러', '로맨스', '액션'과 같은 장르명을 컴퓨터가 이해할 수 있는 384차원의 숫자 벡터로 변환하였습니다. 이 과정을 통해 '스릴러'와 '액션'의 유사도가 0.73으로 계산되는 등, 인간이 직관적으로 느끼는 장르 간 관계를 수치로 정확히 표현할 수 있게 하였습니다.

두 번째는 **유사도 계산 기술**입니다. 코사인 유사도를 기반으로 고차원 벡터 공간에서 두 벡터 간의 각도를 측정하여 유사성을 판단합니다. 이를 통해 장르뿐만 아니라 총 6가지 차원에서 각각 다른 가중치를 부여해 최종 추천 점수를 산출합니다.

세 번째는 **동적 프로그래밍 알고리즘 활용**입니다. 추천 콘텐츠를 특정 조건에 따라 필터링하는 과정에서 최대한 많은 콘텐츠를 추천할 수 있도록 이 알고리즘을 적용하였습니다.

4.2.2 데이터 수집

키노라이츠 웹사이트에서 Selenium 라이브러리를 활용하여 콘텐츠 데이터를 수집하였습니다. 연령대와 성별에 따라 다양한 콘텐츠를 확보하였으며, 해당 랭킹은 내부 인기도와 국내 미디어 트렌드 데이터를 종합하여 산정된 수치입니다.

세부 장르 데이터는 별도로 제공되지 않았기 때문에, 각 콘텐츠 타이틀을 위키백과 URL에 포함시켜 스크래핑을 통해 세부 장르, 출연 배우, 러닝타임 등의 정보를 수집하였습니다.

4.2.3 추천 시스템 종류

추천 시스템의 전체 흐름은 다음과 같습니다. 사용자가 연령대, 성별, 선호 장르, 주 시청 시간, 월 구독 예산, 좋아하는 콘텐츠 제목을 입력하면, 서버는 두 개의 추천 시스템에 해당 정보를 전달합니다.

첫 번째 시스템은 OTT만 추천하는 서비스입니다. 연령대, 성별, 좋아하는 콘텐츠 제목을 기반으로, 해당 콘텐츠와 유사한 콘텐츠가 존재하는 OTT를 추천합니다. 이 시스템은 참고용 지표로 활용됩니다.

두 번째 시스템은 콘텐츠와 OTT를 동시에 추천하는 서비스입니다. 연령대, 성별, 선호 장르, 주 시청 시간, 월 구독 예산 정보를 입력받아, 코사인 유사도를 기반으로 콘텐츠를 추천하고, 이를 시청할 수 있는 OTT를 함께 추천합니다.

4.2.4 콘텐츠 추천 프로세스

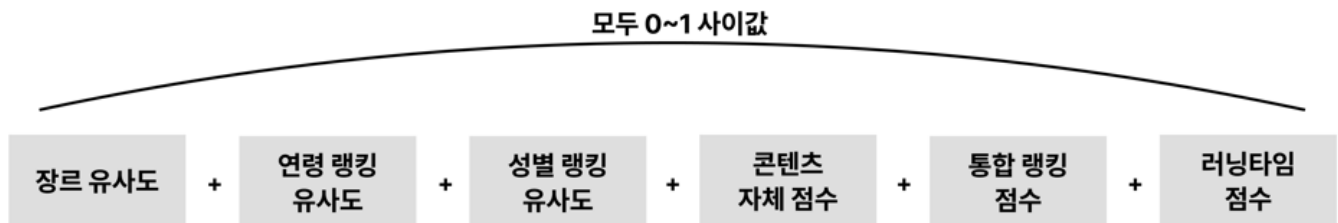
1. SentenceTransformer를 활용하여 장르 유사도를 파악합니다.
2. 성별 및 연령대별 콘텐츠 순위와 매치하여 점수를 부여합니다.



장르 정보만으로도 추천이 가능하지만, 추천 정확도를 높이기 위해 연령대 및 성별에 따른 타겟 매칭도를 추가로 고려하였습니다. 예를 들어, 20대 남성 사용자가 있다고 가정하면 다음과 같은 결과가 도출됩니다.

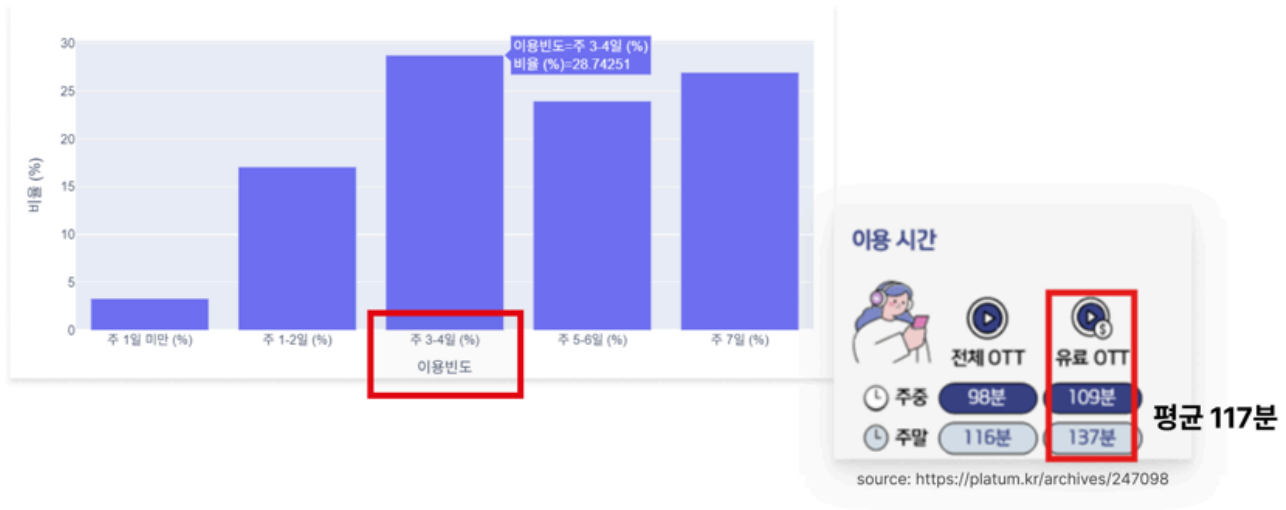
- 콘텐츠1: 30대 남성 대상 3위, 연령 페널티 0.8 → 최종 점수 5.72
- 콘텐츠2: 20대 남성 대상 21위, 연령 일치 1.0 → 최종 점수 3.12
- 콘텐츠3: 20대 여성 대상 1위, 성별 페널티 0.3 → 최종 점수 3.60

결국 추천 순위는 콘텐츠1 > 콘텐츠3 > 콘텐츠2로 결정됩니다.



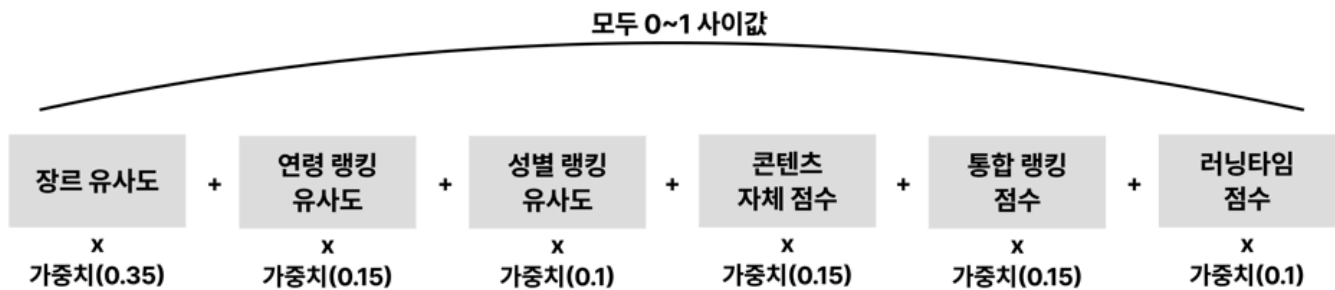
3. 러닝타임 점수를 계산합니다.

러닝타임이 길수록 점수를 낮추는 방식으로 설계하였습니다. 기준은 10시간이며, 시리즈 콘텐츠가 이 기준을 초과하면 점수가 감소합니다. 영화는 평균 2시간으로 대부분 1점을 부여합니다.



국내 평균 시청 패턴(주 5일 × 일 2시간 = 주 10시간)을 고려하여 기준을 설정하였습니다.

4. 총 6개 점수에 가중치를 부여해 통합 점수를 계산하고, 상위 콘텐츠를 추천합니다.



가중치는 주관적으로 설정하였지만, 변수별 상대적 중요도에 따라 다르게 적용하였습니다. 예를 들어 연령 유사도는 0.15, 성별 유사도는 0.1로 설정하였습니다.

연령 랭킹 유사도	성별 랭킹 유사도
× 가중치(0.15)	× 가중치(0.1)

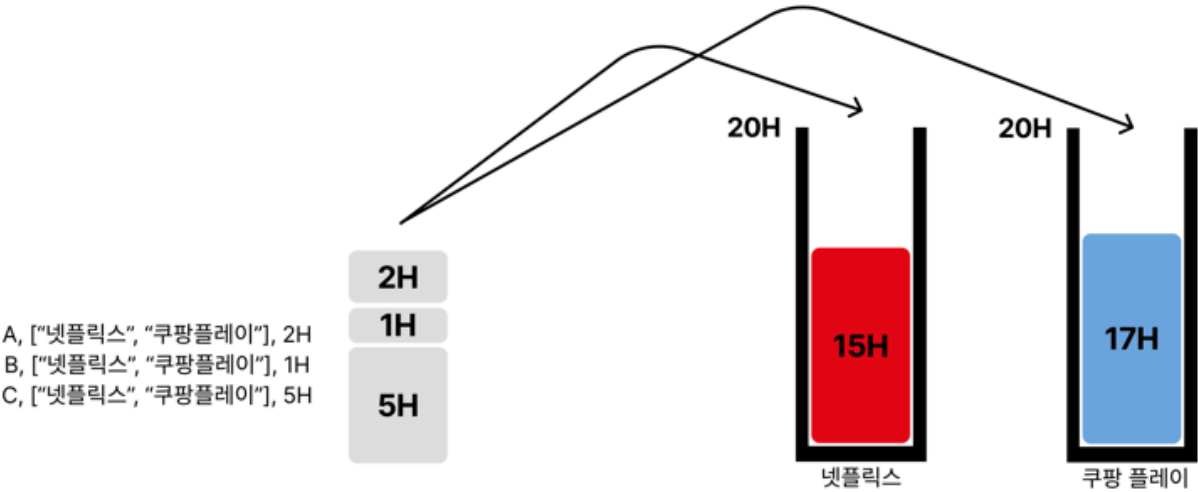
MovieID	Distinctive User Characteristics
423	only mean
118	artist(4), mean
15	artist(1), gender, mean
151	age(3)
216	only mean
210	age(4), writer(1), mean
9	doctor(3)
176	age(1), healthcare(1), mean
204	age(1), healthcare(1), mean
1	only mean
168	writer(1), mean
117	artist(1), mean
12	artist(3), mean
75	age(1), mean
105	only mean
202	artist(1), age(2), mean
202	salesman(2), mean
288	age(3), none(2), mean
172	age(3), mean
318	entertainment(1), mean

Gender는 1번만 유의미한 변수로 포함

이는 “KNN-based Recommender System” 논문 분석 결과를 기반으로 하였습니다. 성별보다 연령대가 더 유의미한 변수로 반복적으로 등장하였기 때문입니다.

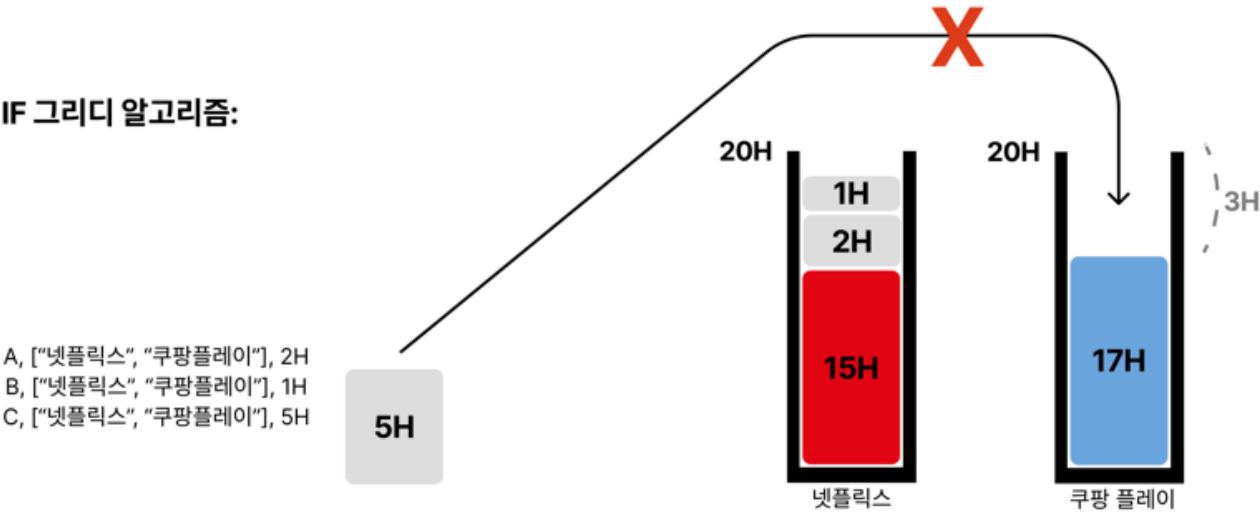
최대 8개의 콘텐츠를 추천하되, 사용자의 1주 OTT 시청 시간과 월 구독 플랫폼 수를 고려하여 추천 수는 유동적으로 결정됩니다.

5. 동적 프로그래밍을 기반으로 콘텐츠 필터링을 수행합니다.

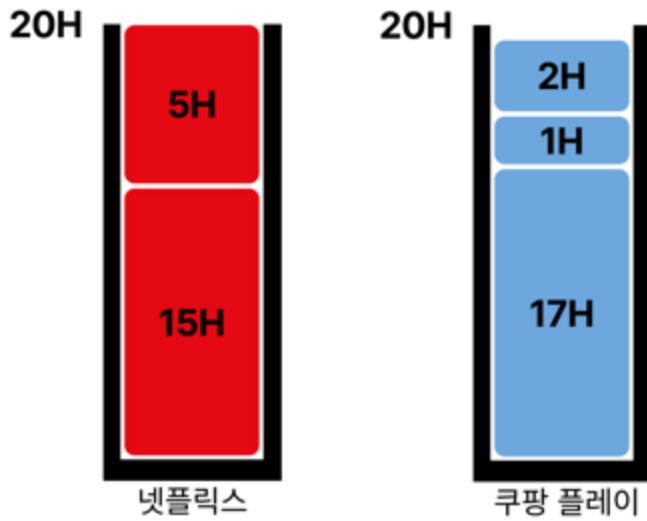


예를 들어 주간 시청 시간이 5시간이라면, 한 달 기준 총 20시간으로 계산하여 각 플랫폼별 시간 제한 내에서 가장 많은 콘텐츠를 추천할 수 있도록 동적 알고리즘을 적용합니다.

IF 그리디 알고리즘:



그리디 방식은 비효율적인 결과를 낼 수 있으나,



동적 프로그래밍은 가능한 모든 조합을 고려하여 최적의 콘텐츠 집합을 도출합니다.

4.2.5 사용 예제와 OTT 추천

연령대:

20대

성별:

남성

좋아하는 콘텐츠 (콤마로 구분):

오징어 게임

선호 장르 (콤마로 구분):

드라마, 예능

세부 장르 (콤마로 구분):

스릴러, 버라이어티

주간 시청 시간 (시간):

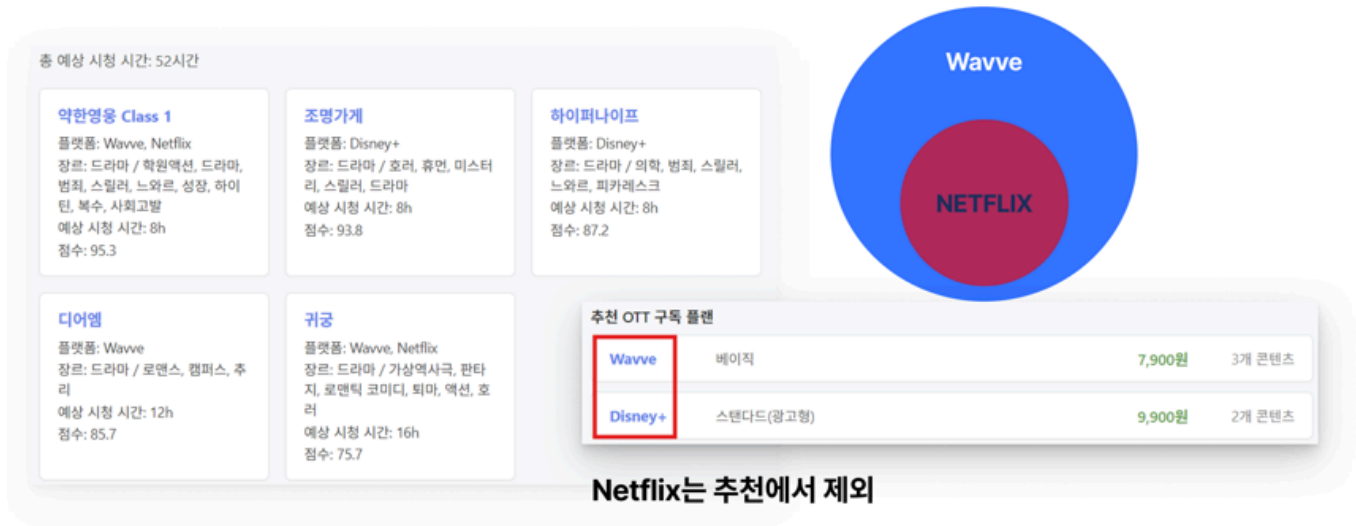
12

월 예산 (원):

10000

검색하기

위와 같이 사용자로부터 정보를 입력받으면, 아래와 같이 추천 콘텐츠와 해당 콘텐츠를 가장 저렴하게 시청할 수 있는 OTT를 함께 추천합니다.



사용자에게 콘텐츠를 반환하기 전, 추천된 OTT 간의 콘텐츠 집합을 비교하여 다른 플랫폼에서 대체 가능한 경우 중복 구독을 방지하는 **OTT 필터링 알고리즘**도 적용됩니다.

예를 들어 Netflix에서 추천된 콘텐츠들이 Wavve에서도 모두 시청 가능하다면, Netflix는 구독 추천 대상에서 제외됩니다.

4.3. 활용 한계점 및 확장성

현재 시스템은 사용자의 입력 정보에 기반하여 콘텐츠를 추천하고 있으나, 실제 시청 이력이나 평가 로그 등은 반영되지 않습니다. 이에 따라 개인화 수준에는 한계가 존재하며, 추천은 일반적인 선호 경향에 기반합니다.

향후 시청 로그와 평가 데이터를 축적하여 반영하게 된다면, 가중치를 모델이 직접 학습하게 하여 추천 정밀도를 향상시킬 수 있을 것으로 기대됩니다.

또한 추천 결과를 LLM과 연계하여 “이번 달엔 넷플릭스, 다음 달엔 티빙”처럼 자연어로 설명하는 컨설팅 형태로 확장할 수 있습니다. 이는 사용자가 점수나 리스트 대신 **직관적인 전략 제안**을 받을 수 있도록 도와줍니다.

나아가 생활 패턴이나 시청 환경까지 반영한 추천, 또는 구독 대비 만족도 분석을 통해 **자동 제안**하는 기능으로도 확장 가능하다고 판단하고 있습니다.

4.4. 성과 및 배운 점

- NLP 기반 콘텐츠 임베딩 및 벡터 연산을 활용한 추천 시스템 설계
- 다중 조건 최적화(다이나믹 알고리즘)를 실제 서비스 로직에 적용
- ML 모델 서빙과 프론트 통합을 통한 엔드 투 엔드 시스템 개발 경험

5. 참고 자료

1. 박성빈, 김정환, & 안석기. (2020). 뉴스 감성 분석을 이용한 딥러닝 기반 주가 예측에 대한 연구. 한국정보처리학회 춘계학술대회 논문집, 27(1), 241-244.

2. 조성호. (2021). 뉴스 콘텐츠의 오피니언 마이닝을 통한 매체별 주가상승 예측정확도 비교 연구. 정보과학 회논문지, 48(5), 533-541.
3. 김동진. (2019). 뉴스와 주가 빅데이터 감성분석을 통한 지능형 투자 모델. 정보처리학회논문지, 26(3), 157-166.
4. 김성주, 김진수, & 임상현. (2021). 주가지수 방향성 예측을 위한 주제지향 감성사전 구축 방안. 한국지능 정보시스템학회 학술대회논문집, 2021(하계), 332-375.
5. Ko-FinBERT-SC Github Repository. (2022). _Ko-FinBERT-SC: A Korean Financial Sentiment Classification Model Based on BERT.
6. Araci, D. (2019). _FinBERT: Financial Sentiment Analysis with Pre-trained Language Models.
7. Shen, Y., Zhang, P. K. (2024). _Financial Sentiment Analysis on News and Reports Using Large Language Models and FinBERT.
8. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. Proceedings of NAACL-HLT, 4171-4186.
9. Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. (2014). *SemEval-2014 Task 4: Aspect Based Sentiment Analysis*. Proceedings of the 8th International Workshop on Semantic Evaluation (*SemEval 2014*), 27-35.
10. _An Artificial Intelligence Approach to Prediction of Corn Yields under Extreme Weather Conditions Using Satellite and Meteorological Data
11. 신성호 외(2018). *LSTM 네트워크를 활용한 농산물 가격 예측 모델* A Prediction Model for Agricultural Products Price with LSTM Network, 416-429
12. 안재성. (n.d.). model_LSTM. Github. https://github.com/DSJosephAhn/model_LSTM
13. 이솔, 임민희, 이계민, 이현주, 김태홍, 조수민 (2022). 기후변화에 따른 커피생산량의 변화, 2022년 대한 지리학회 연례학술대회 발표 논문요약집, 232-233.