



# Wrap Up Report

MVTI ITS

테스트하러가기 cwjcsk프로필 로그아웃

MoVie + mbTI = MVTI  
주인공이 되어 영화에 몰입해보세요!

테스트 하러가기

ISTJ	ESTJ
ISFJ	ESFJ
ESFP	ISFP
ESTP	ISTP
INFJ	ENFJ
INFP	ENFP
INTP	ENTP
INTJ	ENTJ

인기있는 캐릭터 Top 10

나와 성격이 유사한 캐릭터 Top 10

작성자 : 추천시스템 8조

이나현\_T4146, 전해리\_T4191, 정의준\_T4200, 조원준\_T4211, 채민수\_T4217

## ▼ Table of Contents

- 1. 프로젝트 개요
  - 1-1. 문제 정의
  - 1-2. 해결책
    - A. 제공 서비스
    - B. 기대 효과
  - 2. 프로젝트 팀 구성 및 역할
  - 3. 프로젝트 수행 절차 및 방법
    - 3-1. 프로젝트 수행 방법
      - A. 활용 장비(도구)
      - B. 목표 설정
      - C. 프로젝트 일정
    - 3-2. 데이터
      - A. 데이터 개요
      - B. 데이터 수집
    - 3-3. Model
      - A. 모델 개요
      - B. 모델 선정 및 분석
      - C. 모델 파이프라인
      - D. 모델 성능 기록
  - 4. Product Serving
    - 4-1. 아키텍처
      - A. 전체 아키텍처
      - B. 웹 서비스 아키텍처
    - 4-2. CI/CD

#### 4-3. 사용자 요청 흐름도

##### 5. 협업

###### 5-1. PM Rotate

###### 5-2. Git을 통한 협업

###### 5-3. 협업 Tools

A. Notion을 이용한 데일리 스크럼

B. Gather.town을 활용한 실시간 소통

C. 스터디를 통한 프로젝트 이론 학습

##### 6. Review

###### 6-1. 어려움 극복

###### 6-2. 후속 개발

###### 6-3. 좋았던 점

###### 6-4. 아쉬웠던 점

#### ▼ Link

[인생캐릭터 찾으러가기](#)

[MVP 소개 영상](#)

[프로젝트 발표 영상](#)

[프로젝트 깃허브](#)

## 1. 프로젝트 개요

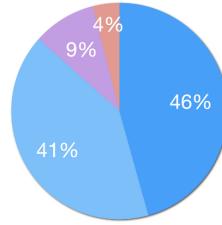
### 1-1. 문제 정의

- 요즘 수많은 OTT 서비스(ex. Netflix)에서 콘텐츠가 쏟아지나, 127명을 대상으로 조사한 결과 약 86%가 OTT를 이용하면서 볼 게 없다고 느낀 적이 있다고 응답

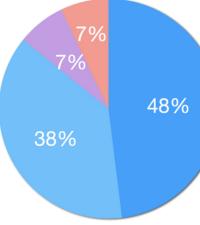
영화 고를 때 들 수 있는 생각들...



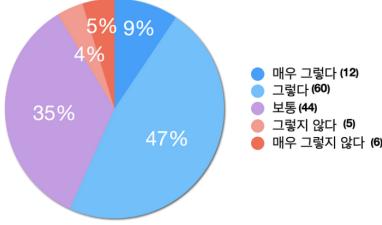
OTT 서비스(ex. Netflix)를 이용하면서 볼게 없다고 느낀 적이 있다.



서비스가 출시되면 사용할 의향이 있나요?



MBTI를 신뢰하시나요?



- 이러한 문제가 발생하는 이유

- 첫 번째는 왜 이러한 추천이 이뤄졌는지 인지하기 어려운 것
- 두 번째는 흥미 유발 요소가 부족

### 1-2. 해결책

#### A. 제공 서비스

##### 1. 나의 성격유형을 바탕으로 한, 캐릭터 추천 서비스

- 입력

사용자로부터 MBTI와 성격 관련 간단한 설문, 선호하는 영화 정보

- 추천

선호하는 영화정보 통해 추천된 영화에서, 나와 같거나 잘 맞는 성격 유형의 캐릭터를 추천

## 2. MBTI별 캐릭터의 인기순위 조회 서비스

### B. 기대 효과

- MBTI 와 같은 성격유형 기반 추천을 통해 **흥미를 유발**
- 성격유형을 근거로, 이 추천이 왜 이루어졌는지 **설명력 향상**
- 나와 같거나 잘 맞는 성격 유형의 캐릭터들을 통해 콘텐츠에 **더욱 높은 몰입을 경험**

## 2. 프로젝트 팀 구성 및 역할

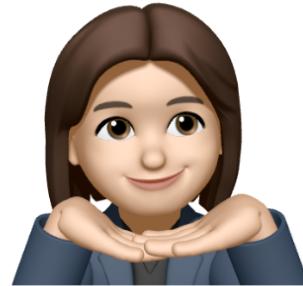
"IT 계의 방탄이 되자, ITS!"



조원준



정의준



이나현



채민수



전해리

- 웹 개발
- 모델 API 개발
- Airflow
- 배포

- 모델 설계 및 평가
- 데이터 수집
- Airflow

- 모델 평가지표
- 웹 개발
- CI / CD

- 데이터 수집 및 처리
- 모델 설계 및 분석
- 웹 사이트 데이터 분석

- 모델 설계 및 분석
- 데이터 처리
- 웹 개발

## 3. 프로젝트 수행 절차 및 방법

### 3-1. 프로젝트 수행 방법

#### A. 활용 장비(도구)

- 개발환경 : Vscode, Jupyter
- 개발 언어 : Python (Pytorch)
- GPU : V100
- 협업 : GitHub, MLflow
- 의사소통 : Slack, Notion, Zoom, Gather Town

#### B. 목표 설정

1. 실제 서비스의 비즈니스 모델 분석
2. 문제 해결을 위한 MVP(Minimum Viable Product)를 제작하여 배포하고 피드백 받기
3. 피드백을 바탕으로 추가 개발하여 배포 사이클 돌기
4. 사이트에 트래픽이 생기면 광고 붙여서 실제 수익화 해보기
5. 실제 현업에서와 같이 유동적인 데이터에 맞춰 완성도 있는 자동화 파이프라인 구축

## C. 프로젝트 일정

1주차	데이터 수집, MVP 소개영상 제작
2주차	데이터 처리, 모델 제작, MVP 소개영상 배포 및 설문
3주차	데이터 처리, 프론트엔드, 백엔드, 모델 성능 개선
4주차	1차 배포 및 피드백 반영
5주차	2차 배포 및 피드백 반영 → 3차 배포

## 3-2. 데이터

### A. 데이터 개요

No	데이터 분류	데이터 설명	수집 경로	활용
1	인터랙션 데이터	25M movie ratings 데이터 (162,000 users, 62,000 movies)	MovieLens	LightGCN, Annoy, Batch Train
2	영화 데이터	MovieLens에 없는 영화 데이터	Personality Database	Bert, TF-IDF
3		영화 정보 데이터	watchapedia	Bert, TF-IDF
4	캐릭터 데이터	캐릭터 MBTI 정보 데이터	Personality Database	유사도 기반 필터링
5		캐릭터 정보 데이터	chatGPT	캐릭터 소개

#### ▼ Personality Database란?



사람들의 투표를 통해 결정된 캐릭터의 MBTI 정보

- 캐릭터 MBTI 정보 데이터에 맞춰 **추가 콘텐츠 정보를 크롤링**
- 데이터 전처리
  - 투표 수가 적은 캐릭터는 제외**
  - 영화의 개봉년도와 평가 횟수를 바탕으로 필터링**

## B. 데이터 수집

데이터 분류	데이터 세부	수집 경로	수집 방법	사용 허가
인터랙션 데이터	약 16만 유저의 2500만 intercation 정보	MovieLens	공개 데이터	Non Commercial Use
영화 데이터	영화 제목, 제작연도, 장르, 이미지	Personality Database	크롤링	admin@pdb.co 메일을 통해 허락 받음
	영화 링크, 줄거리	watchapedia	크롤링	robots.txt 정책에서 유저 정보 제외한 모든 컨텐츠 크롤링 가능함을 확인함
캐릭터 데이터	캐릭터 MBTI, 등장한 영화제목, voting 수, 이미지	Personality Database	크롤링	admin@pdb.co 메일을 통해 허락 받음
	캐릭터 한글이름, 한글 설명	chatGPT	API 사용	OpenAI 사용 정책 확인

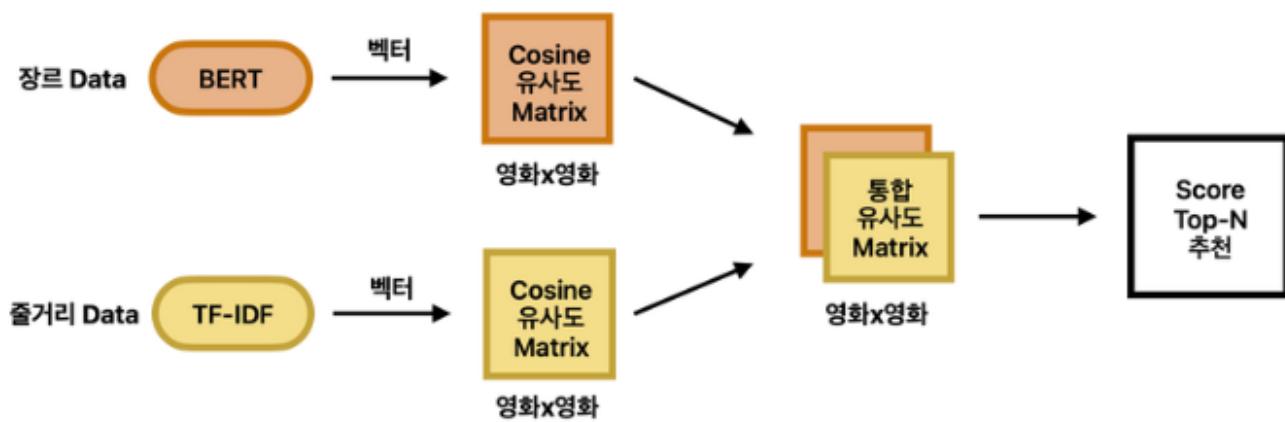
## 3-3. Model

### A. 모델 개요

No	예측 빈도	모델명	특징	사용 데이터	검증 방법	성능 (HitRate@10)
1	실시간	* BTS	장르, 줄거리 정보를 통해 영화간 유사도 측정 후 추천	Movie Side Information	유저 Random Sampling	0.49
2		* LEA	유저-영화 인터랙션 임베딩 벡터 기반의 annoy 추천	MovieLens 25M movie ratings + Real User's Interaction		0.54
3	배치	*ADER	Auto-Encoder기반 모델들을 양상별하여 Top-N 영화 추천			0.72

### B. 모델 선정 및 분석

#### ① BTS



- **BERT + TF-IDF + Cosine Similarity** (자체 제작)
- 한 영화의 장르, 줄거리 정보를 각각 BERT와 TF-IDF를 통해 벡터화
- 벡터 값을 통해 영화 간 **Cosine Similarity Matrix**를 생성
- 입력 받은 영화와 장르, 줄거리가 가장 유사한 영화를 순서대로 출력

## ② LEA

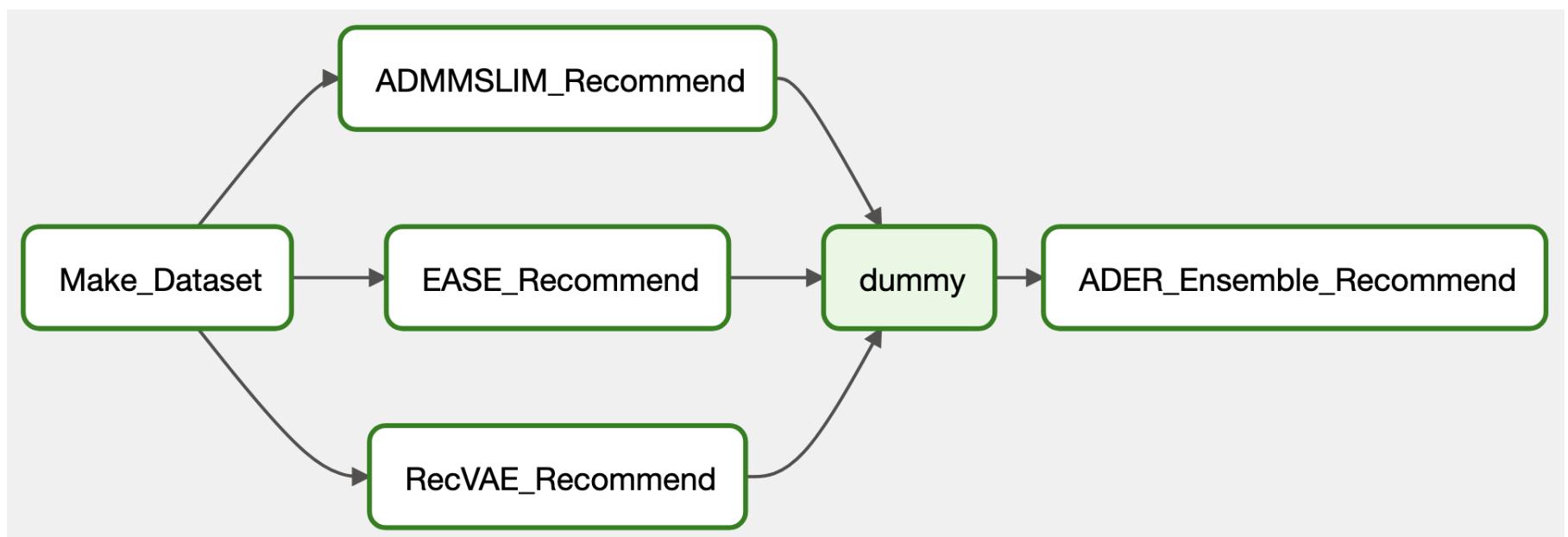


- **LightGCN Embedding + Annoy** (자체 제작)
- 영화와 유저 간의 관계를 **그래프 형태로 모델링** 후 임베딩 벡터 추출
  - 인접한 노드의 정보를 고려하여 임베딩 벡터를 학습하기 때문에, **유저와 영화의 관계를 잘 표현**
- 학습된 임베딩 벡터를 **Annoy**에 넣어 특정 벡터와 **유사한 벡터를 찾아 Top-N 추천**

## ③ ADER



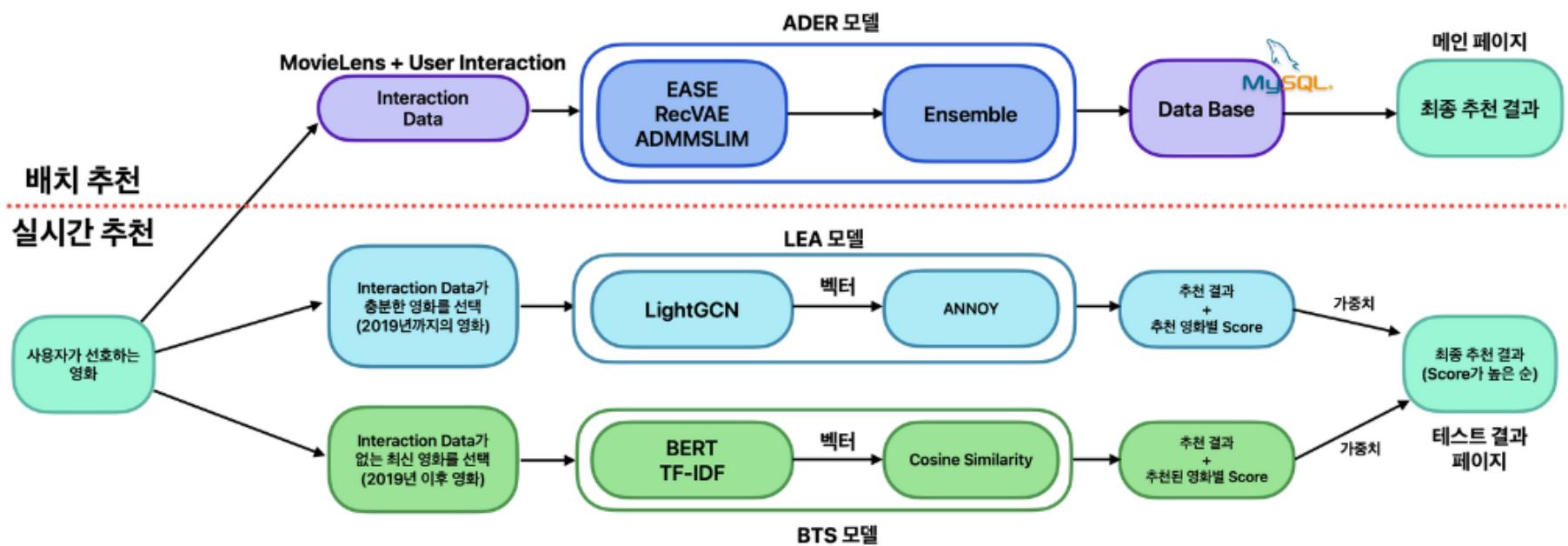
- **ADMMSLIM + EASE + RecVAE** (자체 제작)



- AutoEncoder 기반의 모델들은 **Sparse한 데이터**를 학습할 때 좋은 성능을 검증

- 양상블 하여 영화를 최종 추천

### C. 모델 파이프라인



- 배치 추천
  - MovieLens 인터랙션과 **Real User Interaction**을 학습 데이터로 사용
  - **Airflow**를 통해 매일 추천 결과를 갱신
- 실시간 추천
  - 유저의 성향과 테스트 결과를 바탕으로 학습된 모델을 통해 추천

### D. 모델 성능 기록

The screenshot shows the mlflow interface with the following details:

- Experiments:** A list of experiments including Default, ADMMSLIM, RecVAE, CDAE, LightGCN, and EASE. The EASE experiment is selected and highlighted with a red border.
- EASE Experiment Details:**
  - Description:** Track machine learning training runs in experiments. Learn more
  - Experiment ID:** 744697751960066772
  - Artifact Location:** mlflow-artifacts:/744697751960066772
  - Metrics Table:** A table showing performance metrics for three runs of the EASE model. The table is highlighted with a red border.

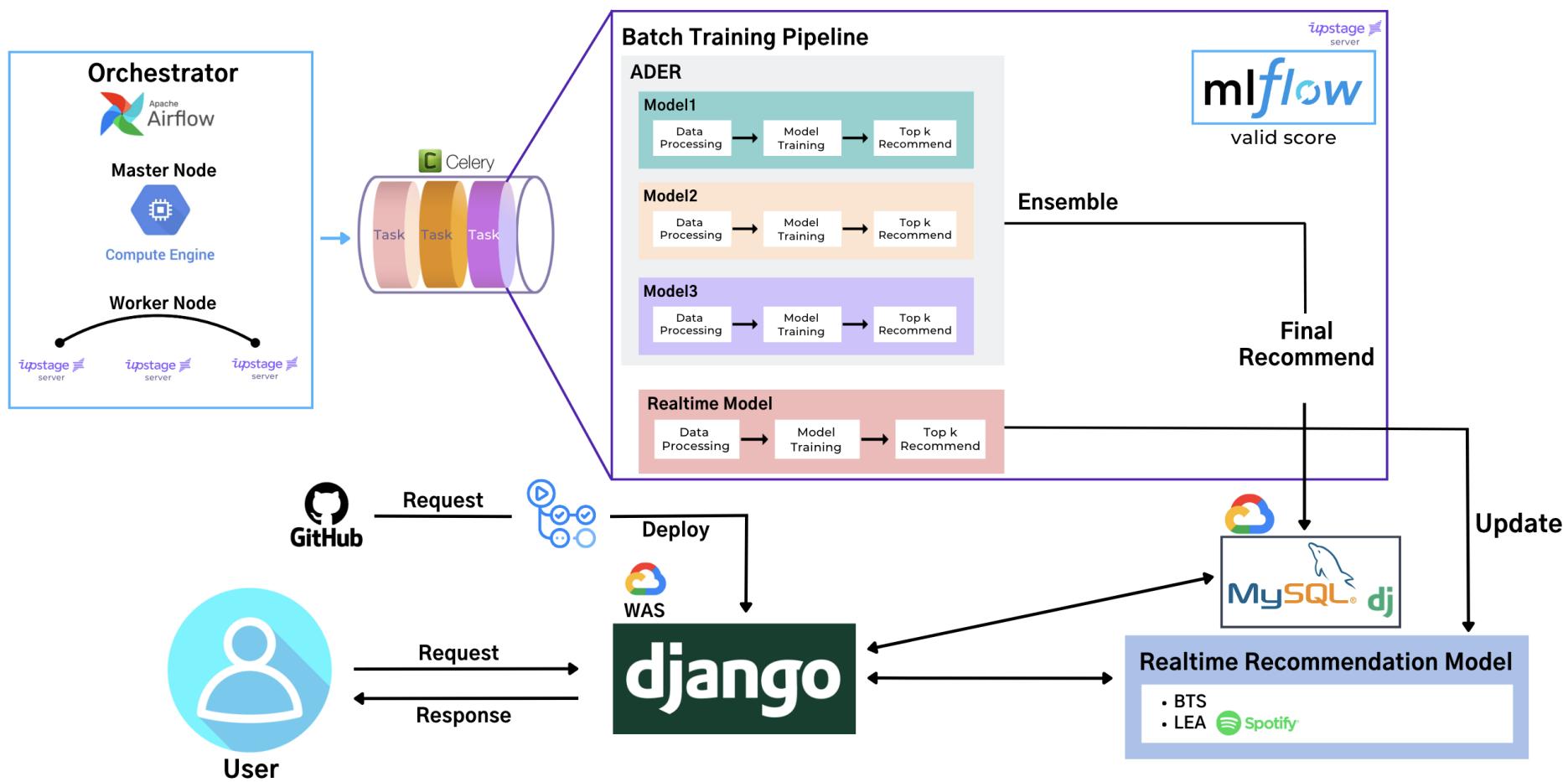
Metrics			
	Run Name	Created	Duration
hit.10	EASE	4 hours ago	56ms
ndcg.10	EASE	4 hours ago	56ms
recall.10	EASE	3 days ago	61ms

- 배치 추천
  - ADMMSLIM, RecVAE, EASE의 성능 기록
- 실시간 추천
  - LightGCN의 성능 기록

## 4. Product Serving

## 4-1. 아키텍처

### A. 전체 아키텍처



- **Batch Training Pipeline**

- 배치 추천 모델인 **ADER**와 실시간 추천 모델인 **LEA** 학습 → Valid Score는 mlflow에 기록

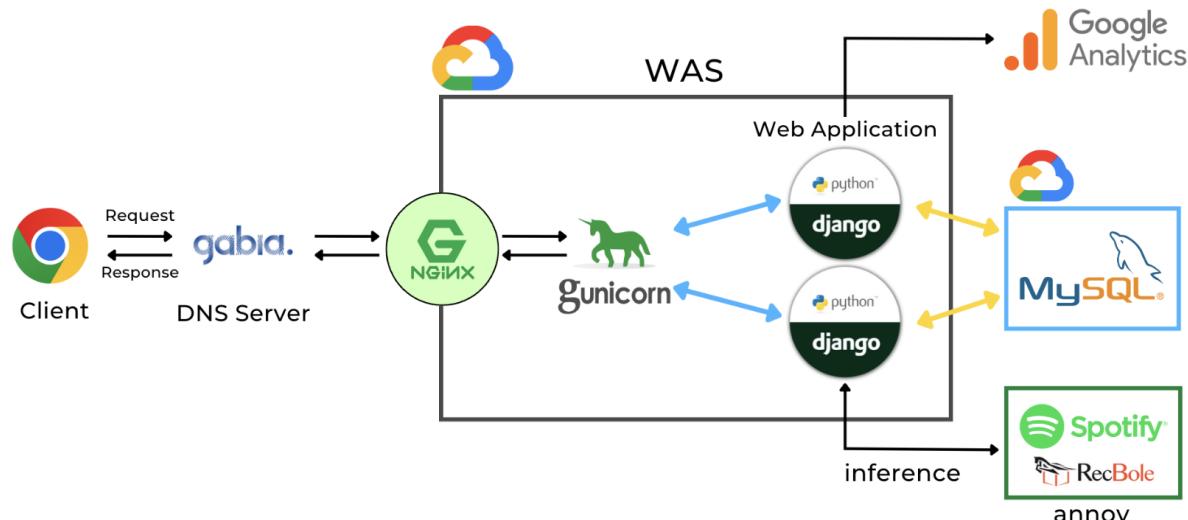
- **WAS**

- 사용자 요청이 왔을 때, Django를 통해 실시간 추천과 배치 추천이 진행

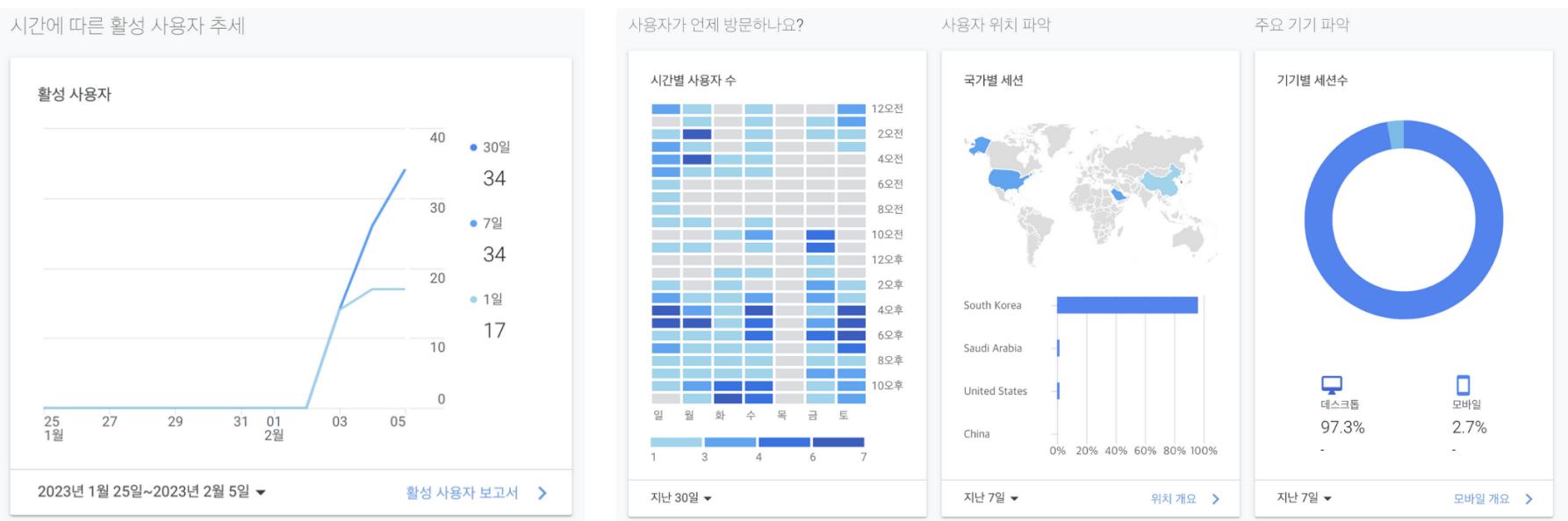
- **Orchestrator**

- Airflow를 이용해 aistage 서버를 최대한 활용

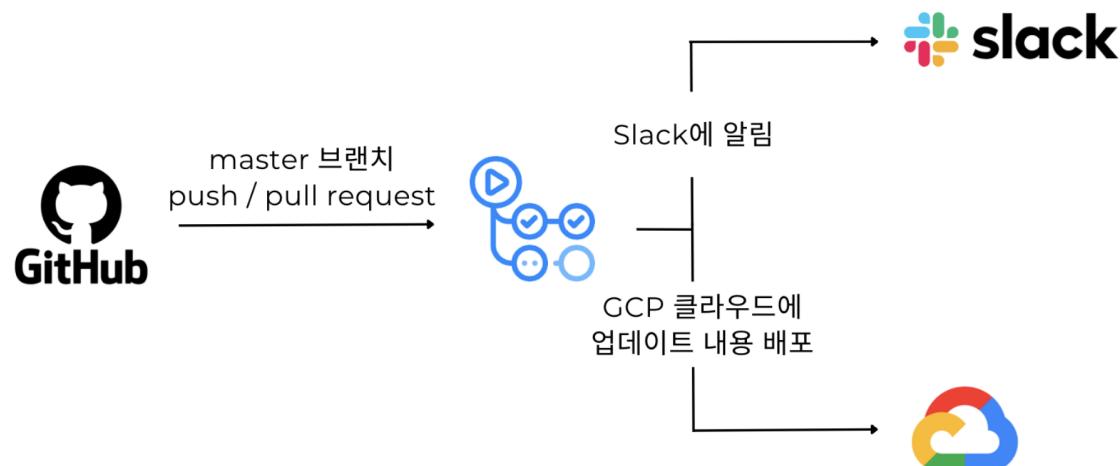
### B. 웹 서비스 아키텍처



- Django를 통해 구현, NginX와 Gunicorn을 통한 배포
- 구글 애널리틱스 연결 - 유저 데이터 분석

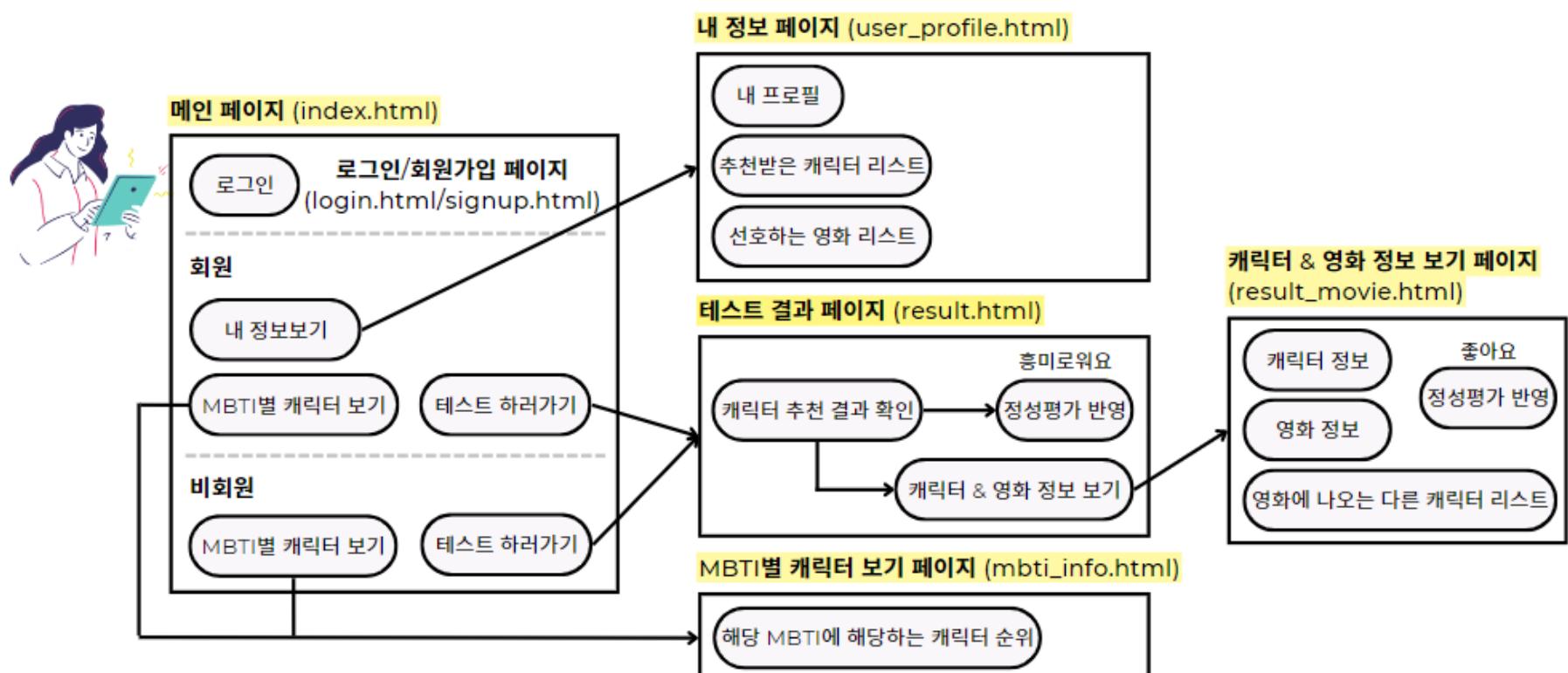


## 4-2. CI/CD



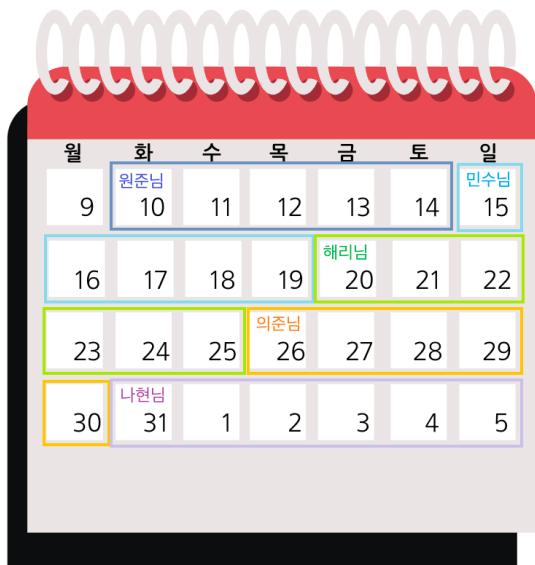
- master 브랜치에 push 또는 pull request 이벤트가 발생하면, 변경사항을 GCP 서버에 배포하고, 배포 성공여부를 slack에 알림

## 4-3. 사용자 요청 흐름도



## 5. 협업

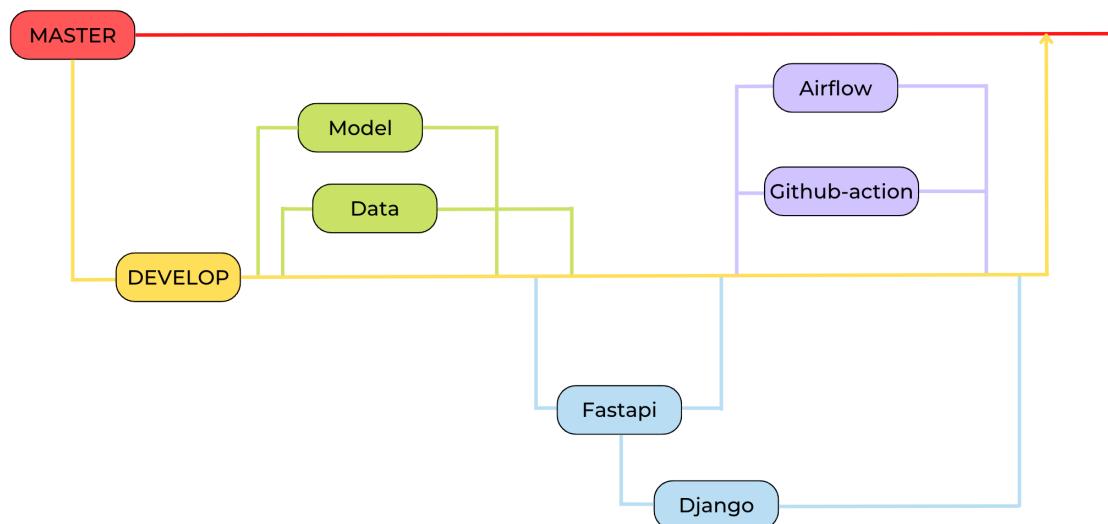
### 5-1. PM Rotate



- 사전에 협의한 기간마다 PM를 위임하여, 모든 구성원이 프로젝트를 리드
- 해당 기간 PM의 주도 하에, 개인의 목표를 수립하고 진행

### 5-2. Git을 통한 협업

→ 각 목적에 맞는 브랜치를 통해 체계적 버전관리



#### ▼ 깃 커밋 Convention



- 작성 방법 : [타입][이름] 간결하고 요약적인 서술

## ▼ 깃 Pull Request

↳ [Django] 전체적인 css, 캐릭터명 한국어로 수정	이나현	전해리	조원준
#50 by Inh31 was merged 11 minutes ago ↗ [Airflow]			
↳ [Airflow] 자동화 스크립트 제작	조원준		
#49 by netsus was merged 11 minutes ago ↗ [Airflow]			
↳ [Model] airflow 관련 start shell 작성, bts random user valid	조원준		
#48 by netsus was merged 11 minutes ago ↗ [Model]			
↳ [Django] MBTI x 애니어그램 성격 정보 추가	이나현	전해리	조원준
#47 by Inh31 was merged 9 hours ago ↗ [Django]			
↳ [Model] BTS 모델(자체 제작) 성능 평가	전해리	채민수	
#46 by chaeminsoo was merged 9 hours ago ↗ [Model]			
↳ [Airflow] gitignore 정리 및 github 파일 삭제	정의준	조원준	
#45 by netsus was merged 9 hours ago ↗ [Airflow]			
↳ [Django] MBTI별 인기순위 적용, 인덱스 페이지 유저 연결	이나현	전해리	조원준
#44 by jeonhaelee was merged yesterday ↗ [Django]			
↳ [Data] 캐릭터 정보 검색	조원준	채민수	
#43 by chaeminsoo was merged 2 days ago ↗ [Data]			
↳ [Airflow] UTC 시간, MLflow 추가, Real User interaction 추가	정의준	조원준	
#42 by uijoon was merged 2 days ago			
↳ [Django] 상세페이지 이미지 툴팁 추가, tmpuser 모델 수정, tmpuser 상세 페이지	이나현	전해리	정의준 조원준 채민수
#41 by Inh31 was merged 2 days ago			

## ▼ 깃 Issue

⌚ [Data] 해쉬태그 중복 이슈	bug	전해리	조원준
#37 by netsus was closed 4 days ago ↗ [Data]			
⌚ [Django] 유저 프로필 화면 에러	bug	이나현	조원준
#36 by netsus was closed last week ↗ [Django]			
⌚ [Model] Attribute Error 발생	bug	조원준	
#33 by netsus was closed last week ↗ [BaseLine]			
⌚ [Data] 데이터를 번역하는데 발생하는 ReadTimeout 에러 해결	bug	채민수	
#15 by chaeminsoo was closed 3 weeks ago			
⌚ [FastAPI] Pynecone Port 지정이 안되는 문제	bug	조원준	
#10 by netsus was closed 2 weeks ago ↗ [FastAPI]			
⌚ [FastAPI] Pynecone Watchpack Error	bug	조원준	
#9 by netsus was closed 2 weeks ago ↗ [FastAPI]			
⌚ [Issue template]	bug	조원준	
#1 by netsus was closed 3 weeks ago ↗ [BaseLine]			

## 5-3. 협업 Tools

### A. Notion을 이용한 데일리 스크럼

→ 데일리 기록을 바탕으로 매일 진행한 것, 어려운 점 등을 공유

## 트렐로

보드 보기 캘린더 보기 표 보기 타임라인

담당자 ▾

To Do	In Progress	[Mandatory]	[Data] Done	[협업] Done
MAB 적용 가능성? 해피 조	Airflow - Real User로 EASE 추천 해피 조 1월 28일	BTS 성능 검사 Minsu Chae 2월 7일	MBTI x 애니어그램 조합 성격 나현 이 해피 조 Minsu Chae 해리 전 의준 정 1월 31일	all scp send 해피 조 2월 7일
General 추천 rating 실험 해피 조	인덱스 페이지 디자인 해리 전 1월 29일 → 2월 4일	Batch Train용 MLFlow 해피 조 2월 5일	캐릭터 정보 크롤링 Minsu Chae 해피 조 해리 전 의준 정 나현 이 2월 4일	HTML 통합 템플릿 해피 조 1월 13일
+ 새로 만들기	2차 피드백중 에러 개선 2월 3일	Django - 백엔드 구조 변경 해피 조 2월 5일	캐릭터, 영화 프로세싱 → 인지도 있는 영화, 주연급 캐릭터 해피 조 2월 6일	Task 나누는 방법 및 인수인계 Flow 구체화 해피 조 1월 12일
	장고 post 요청 깨지는 오류 나현 이	깃허브 액션 배포 결과 슬랙 알림 나현 이 2월 6일	추천 캐릭터 가중치 캐릭터 DF Minsu Chae 1월 28일	Git에 Baseline 코드 옮겨두기 해피 조 1월 10일
	Airflow - LGCN 업데이트 해피 조 의준 정 2월 6일	MBTI별 인기순위 페이지 (최종) 해리 전 해피 조 2월 7일	MBTI, 애니어그램 특징 해리 전 1월 28일 → 1월 30일	MVP 영상 제작 1월 10일
	수집한 자료 사이트에 적용 Minsu Chae 2월 8일	2차 배포 에러 2월 1일	최종 데이터프레임 적용 해리 전 의준 정 1월 30일 → 1월 31일	+ 새로 만들기
	캐릭터 정보 크롤링데이터 정제 및 처리 해리 전 2월 7일 → 2월 8일	Django - 테스트 GCP - Airflow 추천 결과 공유 기능 나현 이 2월 5일	+ 최신 영화 movield 추가	
+ 새로 만들기				

## B. Gather.town을 활용한 실시간 소통

→ 서로 필요할 때 바로바로 소통하기 위해 게더타운을 도입



## C. 스터디를 통한 프로젝트 이론 학습

→ 프로젝트와 관련된 주제로 모델 구현 스터디, 면접 스터디를 진행하여 프로젝트에 필요한 이론들을 깊게 공부하고 공유

## 国旗 코딩 테스트 스터디 (금요일)

- 💻 이나현님의 연구일지
- 🦋 전해리님의 연구일지
- 😎 정의준님의 연구일지
- 🐤 조원준님의 연구일지
- 🎷 채민수님의 연구일지

## 윷 면접 스터디 (수요일)

- ▼ Product Serving에 사용되는 Tool 조사
  - 1. Docker (채민수)  
 Docker
  - 2. Google Cloud (이나현)  
 Google Cloud
  - 3. airflow (전해리)  
 Airflow
  - 4. BentoML(정의준)  
 BentoML
  - 5. k8s(조원준)  
 k8s
- ▼ 강의 내용 추가 학습(Product Serving)
  - 1. 프로젝트에 Airflow를 어떻게 도입할 수 있을까?  
 가만있는 프로젝트에 긴장감을 주는 Airflow
  - 2. HTML과 CSS  
 HTML과 CSS
  - 3. Github action  
 Github Action
  - 4. Pydantic  
 Pydantic
  - 5. Fast API - 기본 (채민수)  
 Fast API - 기본
- ▼ 장고, Firebase 등 웹개발
  - 1. Model, ORM  
 장고-Model, ORM
  - 2. 장고의 url 과 view  
 장고의 기본요소 (url과 view)
  - 3. MTV 패턴  
 버튼실습을 통한 장고 MTV패턴 이해하기
  - 4. 파이어 베이스란? (채민수)  
 파이어 베이스
  - 5. 장고에 Social Login 구현하기  
 장고에 소셜로그인 구현하기
- ▼ 배포 관련 자유주제
  - 1. gunicorn 이란?  
 Gunicorn
  - 2. Github 액션 간단 정리.. 어?? 겹친나요???? 😅  
 Github Actions
  - 3. nginx란?  
 Nginx란?
  - 4. GCP에서 airflow 맛보기  
 GCP에서 airflow 맛보기
  - 5. GCP App Engine을 통한 배포  
 GCP App Engine

## 모델 구현 스터디 (화요일)

### ▼ [Rebole 모델]

1. Recbole 모델 1  
(링크: [https://rebole.io/docs/user\\_guide/model\\_intro.html](https://rebole.io/docs/user_guide/model_intro.html))
  - ▶ 디버깅 위치
  - a. CDAE: 정의준
    - CDAE
  - b. ADMM-SLIM: 조원준
    - ADMM SLIM
2. Recbole 모델 2
  - a. SRGNN: 전해리
    - SRGNN
  - b. RecVAE: 이나현
    - RecVAE
  - c. NeuMF: 채민수
    - NeuMF
3. Recbole 모델 3
  - a. BPR: 정의준
    - BPR
  - b. Pop: 조원준
    - Popularity로 알아보는 Recbole의 추천 알고리즘
4. Recbole 모델 4
  - a. KSR : 전해리
    - KSR
  - b. FPMC : 이나현
    - FPMC
  - c. FOSSIL : 채민수

# 6. Review

## 6-1. 어려움 극복

### 🔥 데이터 수집의 어려움

- 캐릭터의 MBTI 정보 제공 사이트(Personality DataBase)에 데이터 사용 허가를 받음
- MBTI와 애니어그램 조합의 성격 정보는 OpenAI API를 사용
- 제목 뿐만 아니라 제작연도 등 다양한 정보를 이용하여 크롤링

### 🔥 콜드 스타트 문제

- 유저가 선호하는 영화를 3개 이상 선택하게 하여 이를 바탕으로 유사도 기반 추천
- MovieLens interaction 데이터 활용 → LEA 모델
- Contents Based 기반 추천 → BTS 모델

## 6-2. 후속 개발

### ⚙️ 지속적인 관리 및 업데이트

- Airflow를 활용한 ETL 파이프라인 구축

### ⚙️ A/B 테스트

- 유저 피드백에 따른 MAB(Multi-Armed Bandit) 알고리즘 도입
- A/B 테스트에 따른 Google Analytics 분석

### ⚙️ 쿠버네티스

- 유저 수가 많아지면 Load balancing을 통해 트래픽 분산

### 6-3. 좋았던 점

- 최소 기능 서비스 영상에 **폭발적인 관심**을 받은 점
- 1차 배포를 데드라인에 맞게 배포 성공하여, 피드백을 빠르게 받아 **개선 사이클을 돋**은 점
- 결과물의 퀄리티에 대해 **긍정적인 피드백**을 받은 점
- PM을 돌아가며 일주일 계획을 세우고, 데일리 스크럼을 진행하여 **공동의 목표**를 향해 나아간 점
- 단순히 프로젝트를 위한 서비스가 아닌, 실제 서비스를 개발하여 **배포까지 성공한** 점

### 6-4. 아쉬웠던 점

- 여러 출처의 데이터를 합치는 과정에서, **예외 처리**를 완벽하게 하지 못한 점
- **ETL 파이프라인**을 설계하지 못한 점
- **MAB 알고리즘**을 도입해보지 못한 점
- **쿠버네티스**를 사용해보지 못한 점