Final Presentation of Graduation Project 2

- Diary Emotion based Movie Recommendation System -

Yeji Gim 201814246
Inwoo Park 201512265
Soyeong Sohn 201711649
(Leader) Changhae Jung 201710304

CONTENTS

01 Project Introduction
02 Team Members and Roles
03 Model Description
04 App & Server Description
05 Scalability

01 Project Introduction

02. Team Members and Roles

03. Model Description

04. App & Server Description

05. Scalability

Project Background

- It is normal to suppress inner feelings nowadays.
  → Knowing how to understand and express the emotions is good for the mental health.

➡ Let users analyze the emotions in their diary so that they can understand the emotions of the day

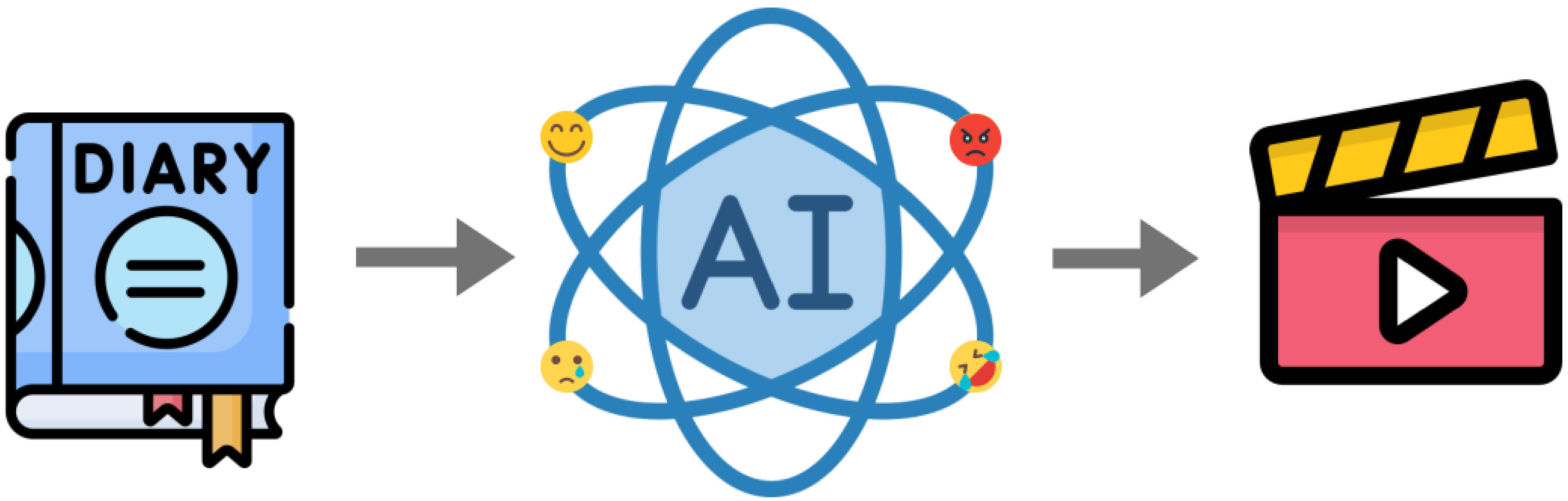
- There are some ways to express emotions such as writing a diary, watching movies, etc.
- Nowadays, OTT platforms can be found anywhere and more people spend their free time watching movies since there is no space and time constraints watching movies anymore.

➡ Let's recommend movies that fit the emotions of the day by analyzing the emotions in the diary!

01 Project Introduction
02. Team Members and Roles
03. Model Description
04. App & Server Description
05. Scalability

Our Topic

iOS App that Recommends Movies by Analyzing Emotions of the User's Diary



01 Project Introduction

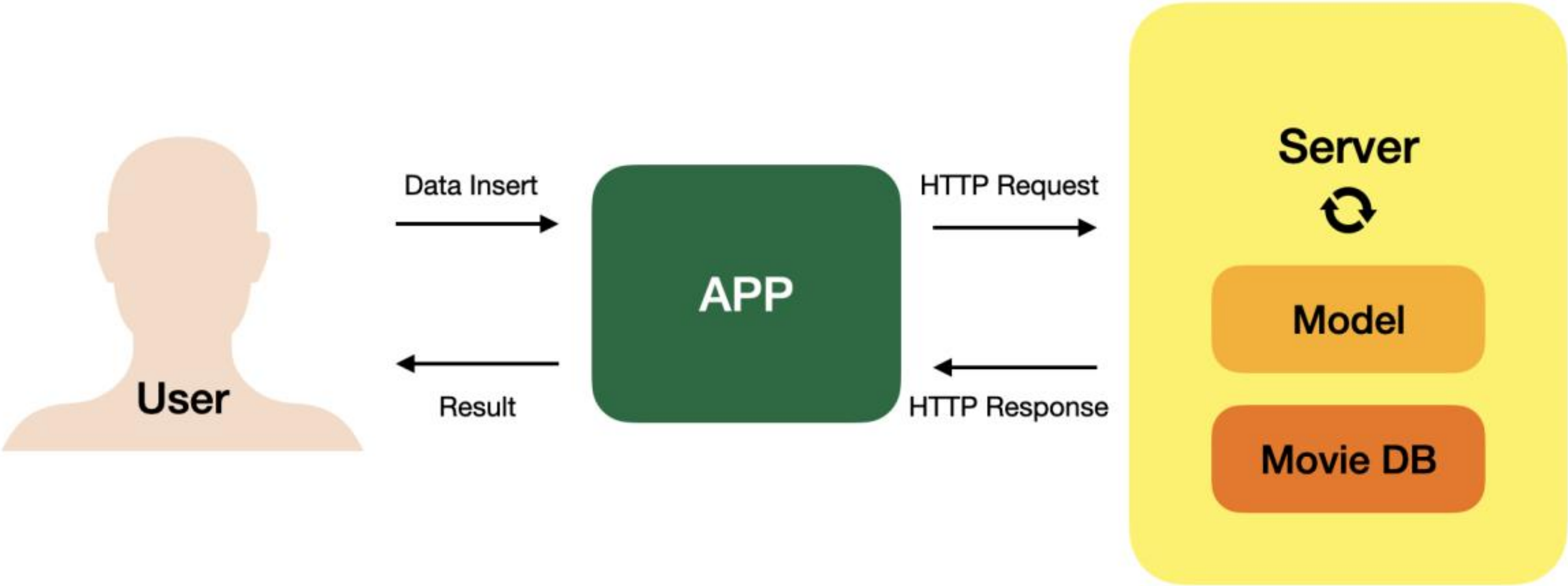
02. Team Members and Roles

03. Model Description

04. App & Server Description

05. Scalability

System Architecture



User

Data Insert

Result

APP

HTTP Request

HTTP Response

Server

Model

Movie DB

- When user writes a diary in the app, the pre-trained model returns the emotion vectors of the diary from the server
- Return recommended movies and emotion probabilities by calculating similarities between the emotion vector of diary and emotion vectors of movies

01. Project Introduction

02 Team Members and Roles

03. Model Description

04. App & Server Description

05. Scalability

Team Members and Roles

Yeji Gim

Model Design &
Data Labelling

Soyeong Sohn

Model Design &
Implementation

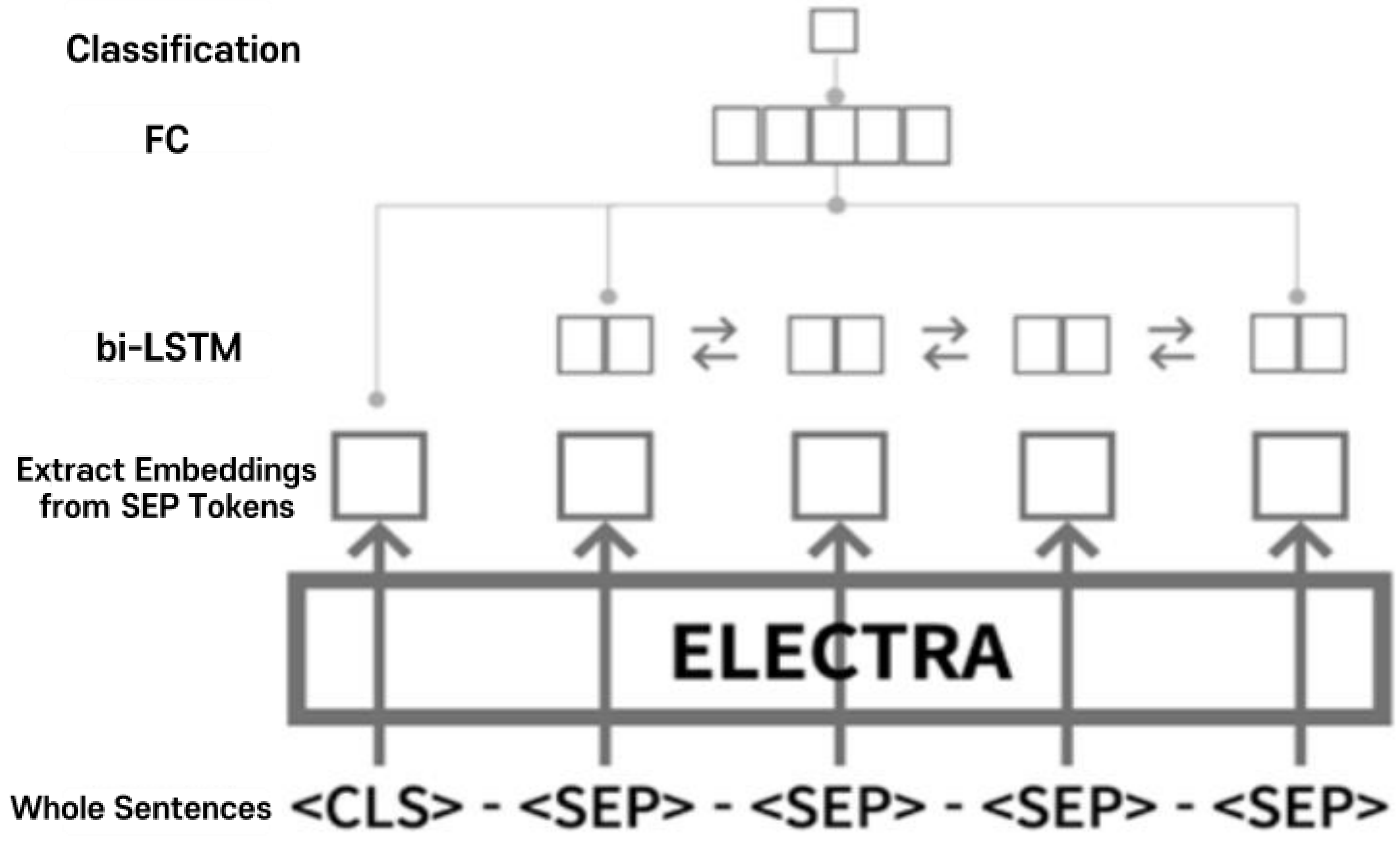
Inwoo Park

iOS App & Server
Development

Changhae Jung

Model Design &
Implementation

Emotion Analysis Model - ELECTRA + LSTM


Classification
FC
bi-LSTM
Extract Embeddings from SEP Tokens
ELECTRA
Whole Sentences
<CLS> - <SEP> - <SEP> - <SEP> - <SEP>

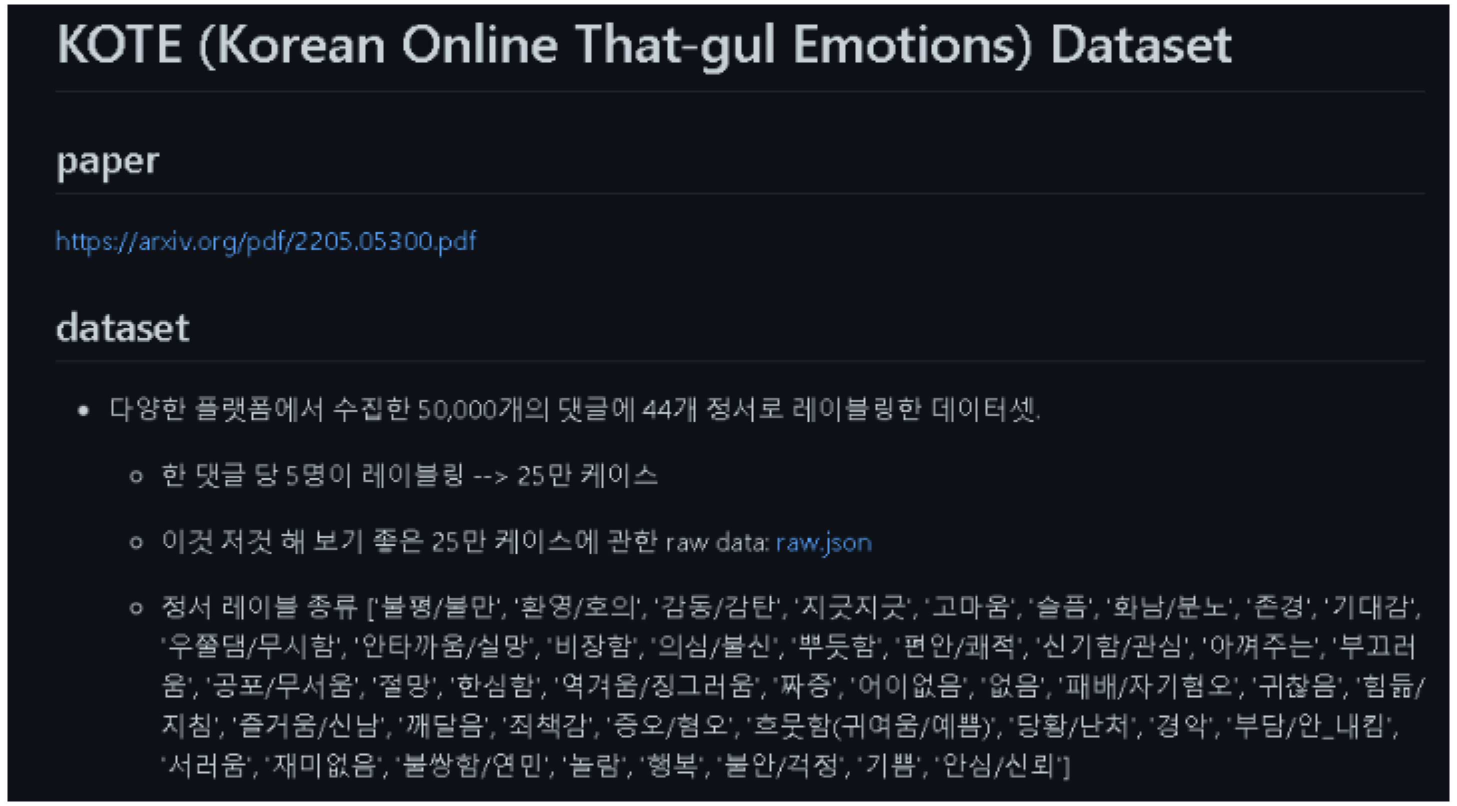
- ELECTRA: Pre-trained Language Model, which is one of the derived models of BERT
- Added bi-LSTM layer to reflect the flow of the diary
- Added <SEP> token between the sentences by using KSS(Korean Sentence Split), concatenate the embeddings from <SEP> tokens of the first sentence and the last sentence and put it in bi-LSTM layer
- Trained the model with the loss which is the sum of the loss from <CLS> token and the losses from <SEP> tokens of first sentence and last sentence

Emotion Analysis Model - Fine Tuning

- Due to the feature of diary written in the App, the vocabularies used to pre-train must be used frequently in daily life or online

→ Used kcELECTRA which is pre-trained with Korean online comments

- Used KOTE dataset, which multi-labeled sentences into 44 emotions, to classify the emotions
- Created silver label of each sentence which is retrieved by PLM, then made gold label by editing silver label by ourselves



KOTE (Korean Online That-gul Emotions) Dataset

paper

https://arxiv.org/pdf/2205.05300.pdf

dataset

- 다양한 플랫폼에서 수집한 50,000개의 댓글에 44개 정서로 레이블링한 데이터셋.
  - 한 댓글 당 5명이 레이블링 --> 25만 케이스
  - 이것 저것 해 보기 좋은 25만 케이스에 관한 raw data: raw.json
  - 정서 레이블 종류 ['불평/불만', '환영/호의', '감동/감탄', '지긋지긋', '고마움', '슬픔', '화남/분노', '존경', '기대감', '우쭐댐/무시함', '안타까움/실망', '비장함', '의심/불신', '뿌듯함', '편안/쾌적', '신기함/관심', '아껴주는', '부끄러움', '공포/무서움', '절망', '한심함', '역겨움/징그러움', '짜증', '어이없음', '없음', '패배/자기혐오', '귀찮음', '힘듦/지침', '즐거움/신남', '깨달음', '죄책감', '증오/혐오', '흐뭇함(귀여움/예쁨)', '당황/난처', '경악', '부담/안_내킴', '서러움', '재미없음', '불쌍함/연민', '놀람', '행복', '불안/걱정', '기쁨', '안심/신뢰']

01. Project Introduction
02. Team Members and Roles
03 Model Description
04. App & Server Description
05. Scalability

Emotion Analysis Model - Scheduler

Cosine Annealing Scheduler

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min})\left(1 + \cos\left(\frac{T_{cur}}{T_{\max}}\pi\right)\right)$$

- Scheduler for solving Local Minimum problem
- The Learning Rate drops to eta_min along the Cosine function, then rises to the initial Learning Rate

Emotion Analysis Model - Testing Result

모델	micro F1	차이
kcELECTRA	0.57	
kcELECTRA + bi-LSTM	0.62	+0.05
kcELECTRA + bi-LSTM + CosineAnnealingLR Scheduler	0.65	+0.03

- When kcELECTRA + bi-LSTM is applied, there was 5% improvement in performance compare to kcELECTRA only model
- Changing the scheduler to the Cosine Annealing LR scheduler added 3% improvement in performance
- 8% improvement in total

01. Project Introduction

02. Team Members and Roles

03 Model Description

04. App & Server Description

05. Scalability

Movie Recommendation - Cosine Similarities

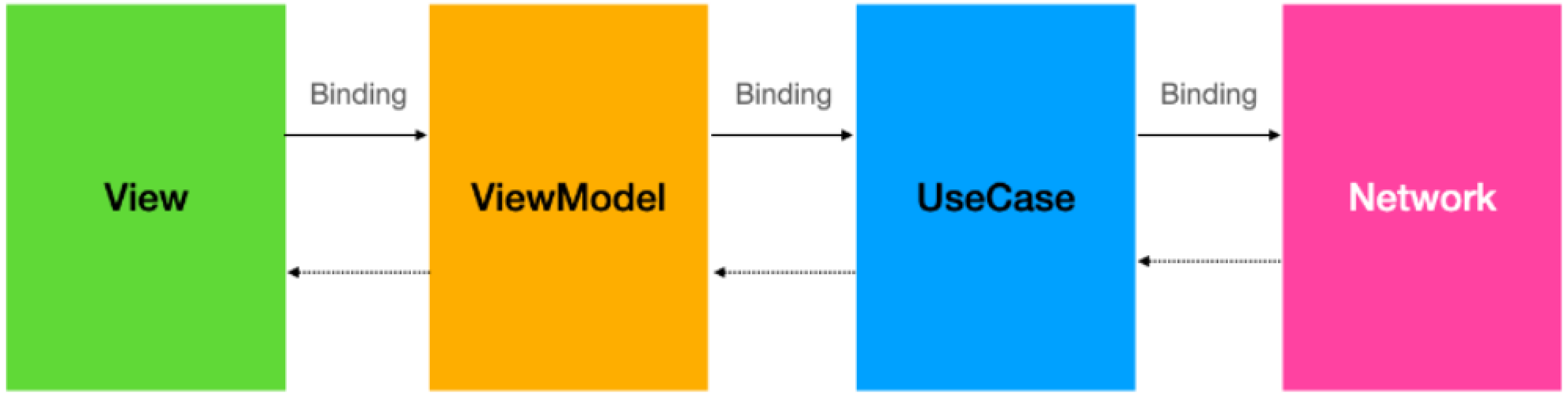
$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

- Metrics that compute the closeness between two vectors
- Compute similarities between the diary written by users and the plots of the movies

- Recommend 5 movies which are most similar to emotion of diary based on cosine similarity

01. Project Introduction
02. Team Members and Roles
03. Model Description
04 App & Server Description
05. Scalability

App & Server Description


View
Binding
ViewModel
Binding
UseCase
Binding
Network

App

- System based on MVVM architecture
- UseCase layer in the middle → reduce redundant code for the same task
- Data Binding in Combine → be able to receive data asynchronously

Server

- Implemented with Flask
- Check and export the data when the app requests the data
- The request consists of HTTP Reqeust in the form of REST API

01. Project Introduction
02. Team Members and Roles
03. Model Description
04. App & Server Description
05 Scalability

Scalability

Service Side

- Link to psychiatric counseling based on emotions derived from the diary
- Expand to recommendation not only movies but also contents such as TV series, book, music, etc.

Performance Side

- Recreating the dataset from scratch to improve accuracy
- Improvement in the recommendation system by reflecting user history such as a list of watched movies, review, etc.
- Experiment models with new approaches

Thank You