

THE FINAL PROJECT

나는 슬플 때 노래를 들어.. 

육미의세포들 

허수정(팀장) 서민정 이시현 정민주 정승욱



멋쟁이사자처럼
K-Digital Training AI스쿨

“Lr는 슬플 때
노래를 들어..”



사용자의 입력기반
감성분석을 활용한 음악 추천 서비스

TEAM 육미의세포들 🧠



허수정 (팀장)

감성분류 모델링, 웹 서비스, 코드 통합

정승욱

스크래핑, 전처리



서민정

스크래핑, 전처리

정민주

감성분류 모델링, 웹서비스, 추천서비스



이시현

감성분류 모델링, 웹서비스

프로젝트 주제와 목적

📌 주제

사용자의 입력 기반으로
감정분석을 활용한 음악 추천 서비스

📌 목적

사용자의 감정을 **분석**하여 음악 **추천**

✅ 감정에 따른 음악 추천

✅ 그날의 기분이나 감정을 해소하거나 치유



난 지금 미쳐가고있다
이 헤드폰에 내 모든 몸과
영혼을 맡겼다

음악만이 나라에서 허락하는
유일한 마약이니까

이게 바로 지금의 나다



협업방식



Github

기능별 브랜치 생성 협업



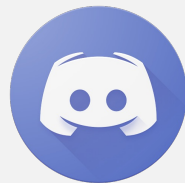
Google Drive

데이터 공유, 발표 자료



Notion

일일 진행사항, 레퍼런스 아카이브



Discord

회의, 협업

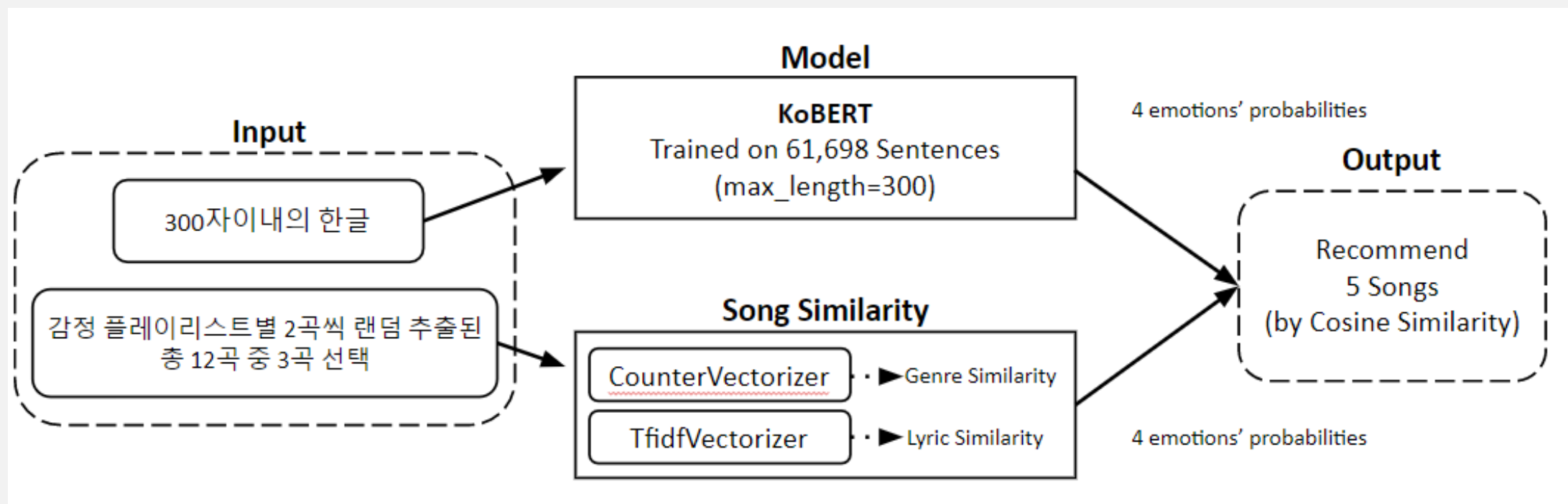
프로젝트 진행 방향

- 서비스 소개
- 활용 데이터 소개 & 데이터 전처리
- 감성분류 모델링 & 가사 감성 분류
- 추천 시스템 모델링
- 서비스 개발 - 장고
- 활용성 & 프로젝트 발전 방향

서비스 소개

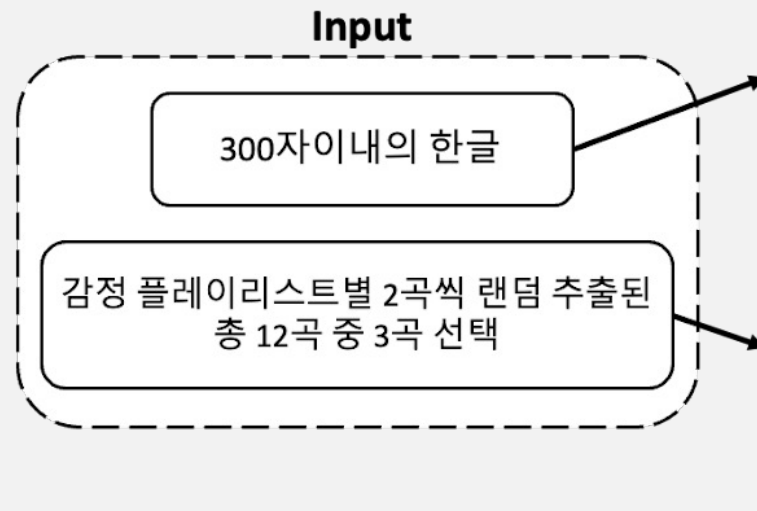


서비스 FLOW



서비스 이용가이드 1) Index

- 초기화면, 유저이름 입력



서비스 이용가이드 2) User Input

- 사용자가 입력창에 자신의 감정 표현 입력
- 300자내 한글 텍스트 입력 input
- 입력된 텍스트의 감정 파악, 감정 대시보드

TODAY 28 | TOTAL 243423

이시현의 미니홈피



오늘도 피곤한 하루를
끝낸 너에게
CHEERS!

뮤직 이즈 모티브이 리
앞..

오늘이 감정으로 노@#를 추천합니다

NO.1 이시현 (2022.05.31 13:07)



오늘의 기분을 300자 이내로 표현해 주세요.
마침표는 꼭!

제출하기

NO.2 반휘혈 (2002.08.19 13:48)



학교라는 감옥에 갇혀..

육미의
세포들
HOME

서비스 이용가이드 2.1) Emotion Dashboard

- 4개의 감정

['분노혐오', '놀람공포', '슬픔', 행복']

확률값, 학습된 Kober Model



서비스 이용가이드 3) User Music Selection

- **감정별 곡 리스트 큐레이션**

- 3곡씩 총 12곡

- **사용자가 3곡 선택**


- 사용자 seed 반영

- **[옵션]**

곡 리스트 새로고침 & 선택 초기화

TODAY 28 | TOTAL 243423

이시현의 미니홈피



오늘도 피곤한 하루를
끝낸 너에게
CHEERS!

뮤직 이즈 마하 이 러
앞..

듣고 싶은 노래를 3개 선택해주세요

☐ 오락실 - 한스 밴드

☐ Gone - 로제 (ROSÉ)

☒ 찐이야 - 영탁

☐ 피 땀 눈물 - 방탄소년단

☒ 그대와 춤을 - 예삿

☒ All The Way Up (K) - 박재범

☐ Martini Blue - DPR LIVE

☐ Candy Sugar Pop - ASTRO (아스트로)

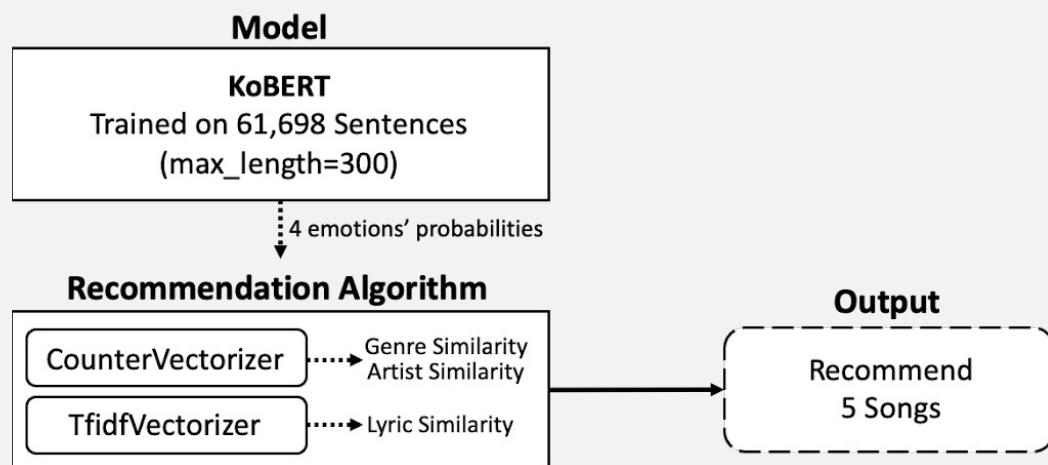
다시선택하기

제출하기

육미의
세포들
HOME

서비스 이용가이드 3) User Music Selection

- 사용자 감정 (KoBERT 분석 활용값)
 - + 사용자 곡 선택
 - + 감정 유사도
- 활용한 추천알고리즘으로 5개곡 추천



활용 데이터 소개 & 데이터 전처리



활용 데이터 소개

사용자의 감정 분류 모델링

- 한국어 감정 정보가 포함된 단발성 대화 데이터셋
- 한국어 감정 정보가 포함된 연속적 대화 데이터셋
- 감정 분류를 위한 대화 음성 데이터셋
- 네이버 영화 리뷰 데이터 셋
- 네이버 쇼핑 리뷰 데이터 셋
- 게임 스템 리뷰 데이터 셋

노래 추천, 가사 감성 분류

- 멜론 데이터 스크래핑
- 장르별 최신 500곡 총 4000 곡

데이터 전처리 - 감성분류 모델링용

▪ 최종 사용 데이터셋 : 7종 감정
분노 / 놀람 / 행복 / 슬픔 / 혐오 / 공포 / 중립

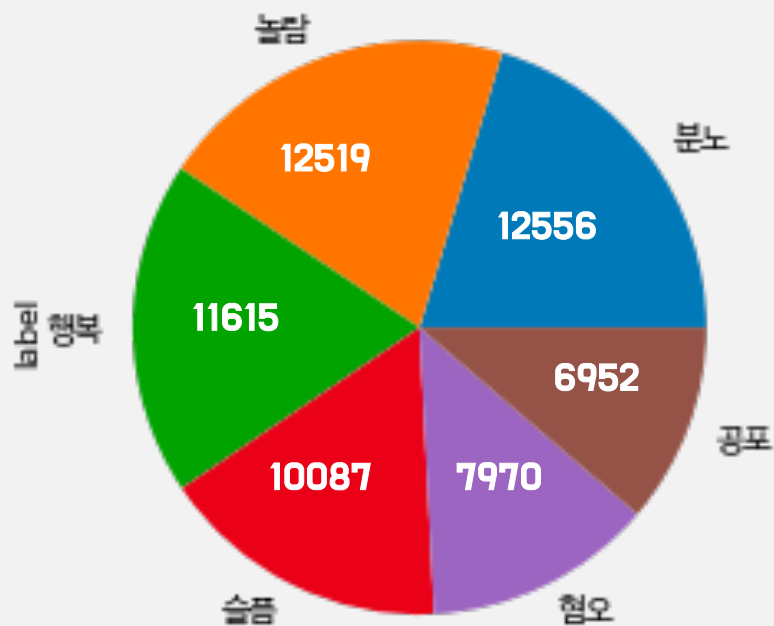
- ✓ 한국어 감정 정보가 포함된 단발성 대화 데이터셋
- ✓ 한국어 감정 정보가 포함된 연속적 대화 데이터셋
- ✓ 감정 분류를 위한 대화 음성 데이터셋

index	Sentence	label
0	아 진짜! 사무실에서 피지 말라니깐! 간접흡연이 얼마나 안좋은데!	분노
1	그럼 직접흡연하는 난 얼마나 안좋겠니? 안그래? 보면 꼭... 지 생각만 하고.	혐오
2	그걸 내가 어떻게 알아요!	분노
3	동사무소에서 인구조사 나왔니? 지금.	분노
4	나? ... 나보고 하라고?	놀람
...
61694	솔직히 예보 제대로 못하는 데 세금이라도 아끼게 그냥 폐지해라..	혐오
61695	재미가 없으니 망하지	혐오
61696	공장 도시락 비우생적임 아르바이트했는데 화장실가성 손도 안씻고 재료 담고 바닥 떨어...	혐오
61697	코딱지 만한 나라에서 지들끼리 피터지게 싸우는 센징 클래스 ㅈㅈㅈ	혐오
61698	와이프도 그렇고 댓글 다 볼텐데 이회재 좀 하차 하라고 전해주세요	혐오

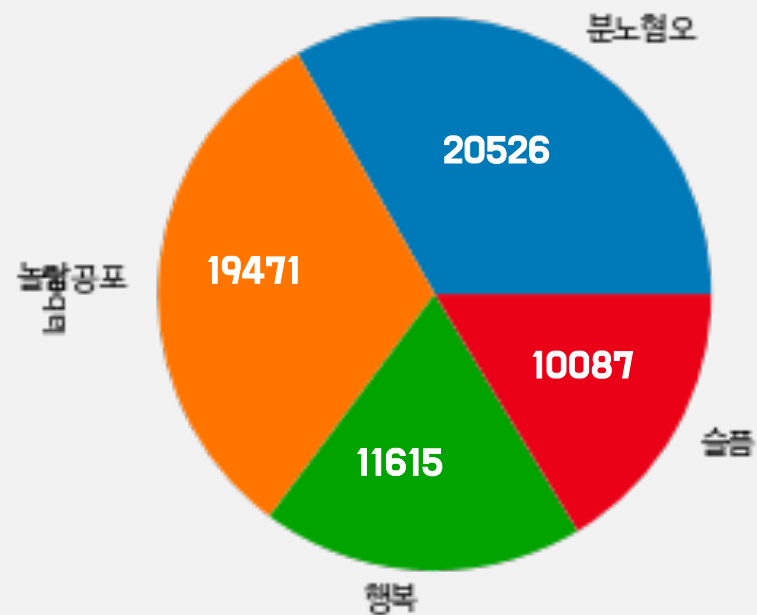
(61699.2)

데이터 전처리 - 감성분류 모델링용

- Ver 1. 6종 감정 : '중립' 제거
분노 / 놀라움 / 행복 / 슬픔 / 혐오 / 공포

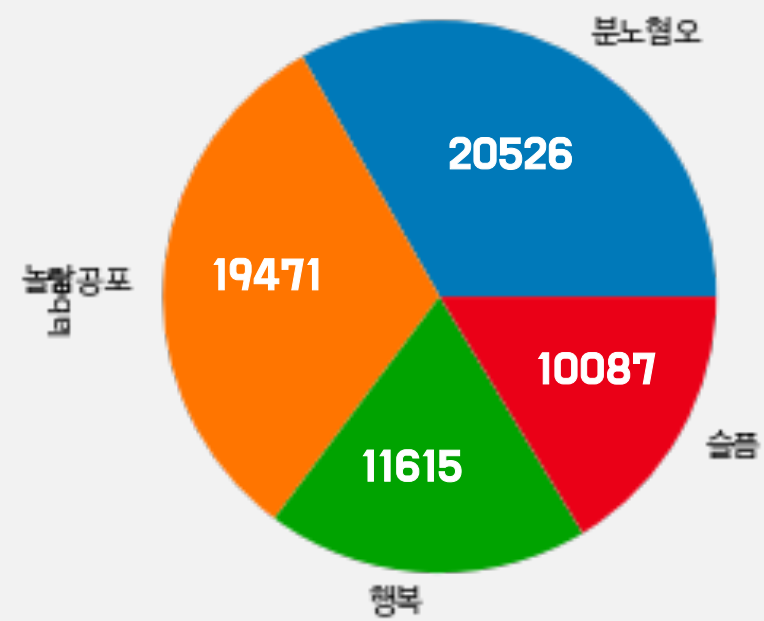


- Ver 2. 4종 감정 : '중립' 제거 및 병합
분노놀라움 / 행복 / 슬픔 / 혐오공포



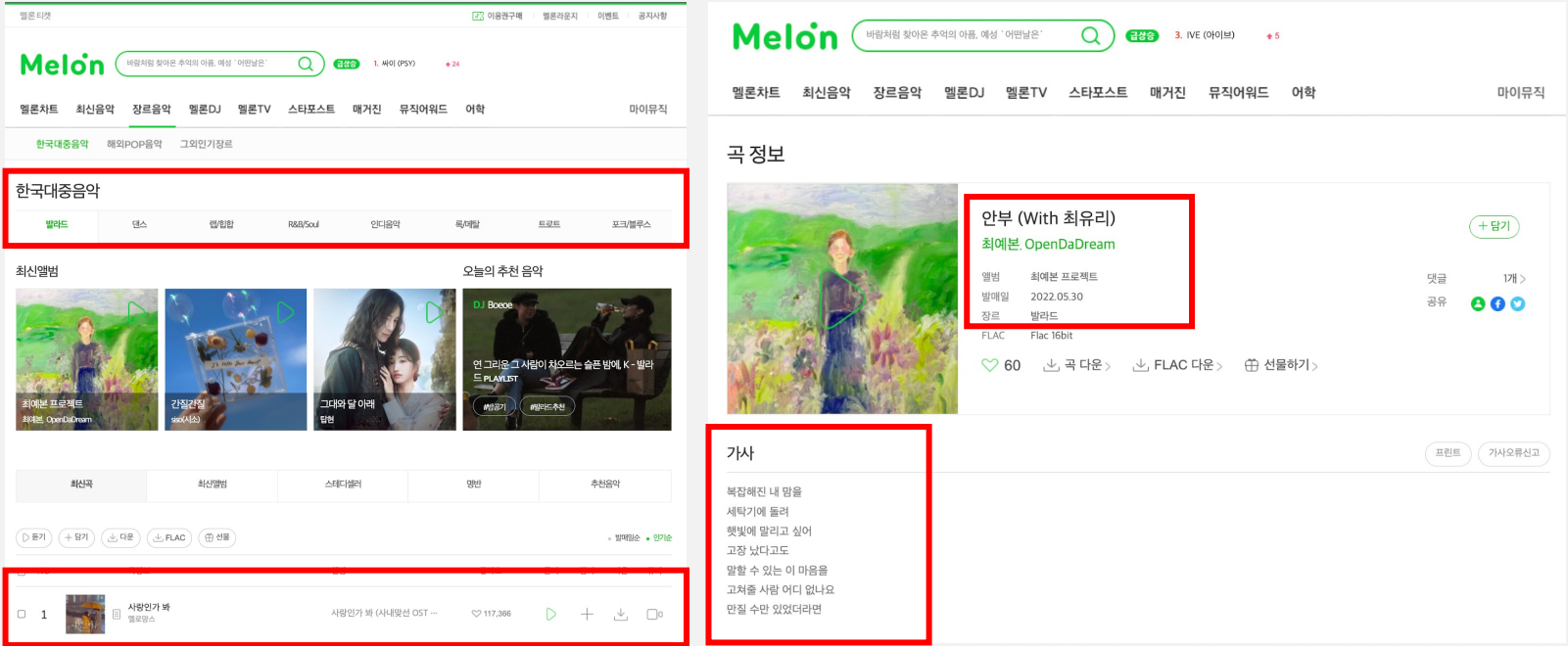
데이터 전처리 - 감성분류 모델링용

- 분노놀람 : 0
- 행복 : 1
- 슬픔 : 2
- 혐오공포 : 3



	Sentence	label
0	아 진짜! 사무실에서 피지 말라니깐! 간접흡연이 얼마나 안좋은데!	0
1	그럼 직접흡연하는 난 얼마나 안좋겠니? 안그래? 보면 꼭... 지 생각만 하고.	0
2	그걸 내가 어떻게 알아요!	0
3	동사무소에서 인구조사 나왔니? 지금.	0
4	나? ... 나보고 하라고?	1
...
61694	솔직히 예보 제대로 못하는 데 세금이라도 아끼게 그냥 폐지해라..	0
61695	재미가 없으니 망하지	0
61696	공장 도시락 비우생적임 아르바이트했는데 화장실가성 손도 안씻고 재료 담고 바닥 떨어...	0
61697	코딱지 만한 나라에서 지들끼리 피터지게 싸우는 센징 클래스 ㅈㅈㅈ	0
61698	와이프도 그렇고 댓글 다 볼텐데 이회재 좀 하차 하라고 전해주세요	0

데이터 전처리 - 멜론 가사 스크래핑



	Title	Artist	Date	Genre	Lyric_ver2
0	나의 X에게	경서	2022.04.24	발라드	<div class="lyric" id="d_video_summary"><!-- height:auto; 로 변경시, 확 우리 다시 만날까 </div>
1	취중고백	김민석 (멜로망스)	2021.12.19	발라드	<div class="lyric" id="d_video_summary"><!-- height:auto; 로 변경시, 확 뭐하고 있었니 </div>
2	사랑은 늘 도망가	임영웅	2021.10.11	발라드, 국내드라마	<div class="lyric" id="d_video_summary"><!-- height:auto; 로 변경시, 확 눈물이 난다 이 </div>
3	듣고 싶을까	MSG워너비(M.O.M)	2022.02.26	발라드	<div class="lyric" id="d_video_summary"><!-- height:auto; 로 변경시, 확 우리 함께 듣던 </div>

■ 멜론 한국 대중음악

- 장르별 최신곡 인기순 500곡
- 곡 고유 song_ID 추출

■ 멜론 곡 정보 : song_ID로 창 진입

■ DataFrame

- 곡명 Title, 가수명 Artist,

출시일 Date, 장르 Genre, 가사 Lyrics

- 성인 인증 필요 가사 제외

데이터 전처리 - 가사 전처리

■ 가사 Lyrics 스크래핑 전처리

- **html 태그들 삭제, 텍스트만 남기기**
- **문장 별로 element, 리스트에 저장**

가사 Lyrics 전처리 전 예시

<div class="lyric" id="d_video_summary"><!-- height:auto; 로 변경시, 확장됨 -->

길거리를 걷다 보면
사랑 노래만 흘러나와
나는 왜 저런 게 낫설까
난 한국말까지 서툴러
번역기도 전혀 도움 안 돼

</div>

📌 가사 Lyrics 전처리 코드

[illegible]

데이터 전처리 - 가사 전처리

문장 별로 element, 리스트에 저장

문장 별로

감성 분류 확률을 추출,

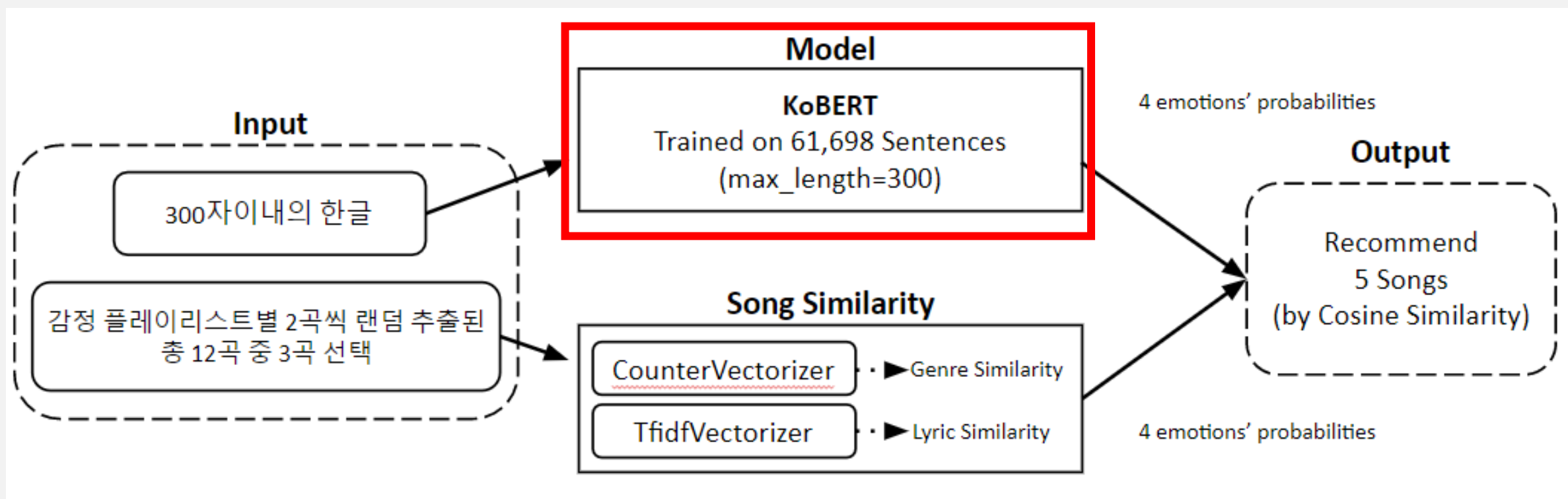
전체 가사의 감정 확률

도출해 내기 위함!

총 3821곡

	Title	SongID	Artist	Date	Genre	Lyrics
0	ON	32399830	방탄소년단	2020.02.21	랩/힙합	['I can't understand what people are sayin'", '어느 장단에 맞춰야 될지', '한 발자국 떴으면 한 발자국 커지는 shadow', '잠에서 눈을 뜬 여긴 또 어디']
1	어떤 날도, 어떤 말도	3534782	심규선 (Lucia)	2011.09.07	발라드, 인디음악	['함께 했던 많은 계절이', '봄에 눈이 녹듯 사라진다 해도', '아직 나는 너를 기억해', '세상 무엇보다 빛나던 모습을', '어떤 날도 어떤 말도', '']
2	너 아니면 안돼	34592758	먼데이 키즈 (Monday Kiz)	2022.02.03	발라드	['오늘도 내 기억을 따라 헤매다', '이 길 끝에서 서성이는 나', '다신 볼 수도 없는 네가 나를 붙잡아', '나는 또 이 길을 묻는다', '널 보고 싶다고', '']
3	사랑이었나봐	33158202	기리보이	2020.12.23	랩/힙합	['사소한 것들 땀에', '크게 다뤘었지', '하루 종일 말도 안 하고', '궁해 있었지', '사랑이었나 봐', '그땐 몰랐지만', '전부 다 지나고 나서', '깨달았어', '둘 중']
4	봄이야	34770461	로키드 (Lokid)	2022.03.21	R&B;/Soul, 인디음악	['서울은 좁아서', '어딜 가도 네가 따라와', '다시 돌아온 이 계절처럼', '또 너를 떠올려', '잘 해내고 있다 믿었지만', '넌 내게 불어와', '다시 흔들린 내 마음처럼', '']

감성분류 모델링 & 가사 감성 분류



KoBERT 모델링

```
1 df['Sentence'].str.len().sort_values(ascending=False)[:5]
```

41743	299
55368	298
61274	295
59693	261
9623	209

Name: Sentence, dtype: int64

```
1 print(len(tokenizer.tokenize(df['Sentence'][41743])))
2 print(len(tokenizer.tokenize(df['Sentence'][55368])))
```

75
292

➔ **max_length = 300**
텍스트 데이터를 토큰화 했을 때, 최대값 292

분석 결과

- ver 1. 6종 감정 : '중립' 제거
분노 / 놀람 / 행복 / 슬픔 / 혐오 / 공포

< Confusion Matrix >

	분노	놀람	행복	슬픔	혐오	공포
분노	2620	331	72	220	309	97
놀람	557	2152	173	216	139	239
행복	166	320	2406	214	67	36
슬픔	385	211	112	1668	60	293
혐오	940	259	73	127	754	88
공포	168	292	25	305	47	1184

< Classification Report >

	precision	recall	f1-score	support
분노	0.54	0.72	0.62	3649
놀람	0.60	0.62	0.61	3476
행복	0.84	0.75	0.79	3209
슬픔	0.61	0.61	0.61	2729
혐오	0.55	0.34	0.42	2241
공포	0.61	0.59	0.60	2021
accuracy			0.62	17325
macro avg	0.63	0.60	0.61	17325
weighted avg	0.63	0.62	0.62	17325

- ver 2. 4종 감정 : '중립' 제거 및 병합
분노놀람 / 행복 / 슬픔 / 혐오공포

< Confusion Matrix >

	분노놀람	행복	슬픔	혐오공포
분노놀람	3213	478	265	144
행복	519	2835	357	227
슬픔	263	330	1276	115
혐오공포	114	176	93	1935

< Classification Report >

	precision	recall	f1-score	support
분노혐오	0.78	0.78	0.78	4100
놀람공포	0.74	0.72	0.73	3938
슬픔	0.64	0.64	0.64	1984
행복	0.80	0.83	0.82	2318
accuracy			0.75	12340
macro avg	0.74	0.75	0.74	12340
weighted avg	0.75	0.75	0.75	12340

가사 감정 분류

학습된 KoBERT 모델 적용,
 노래 추천용 DB 제작

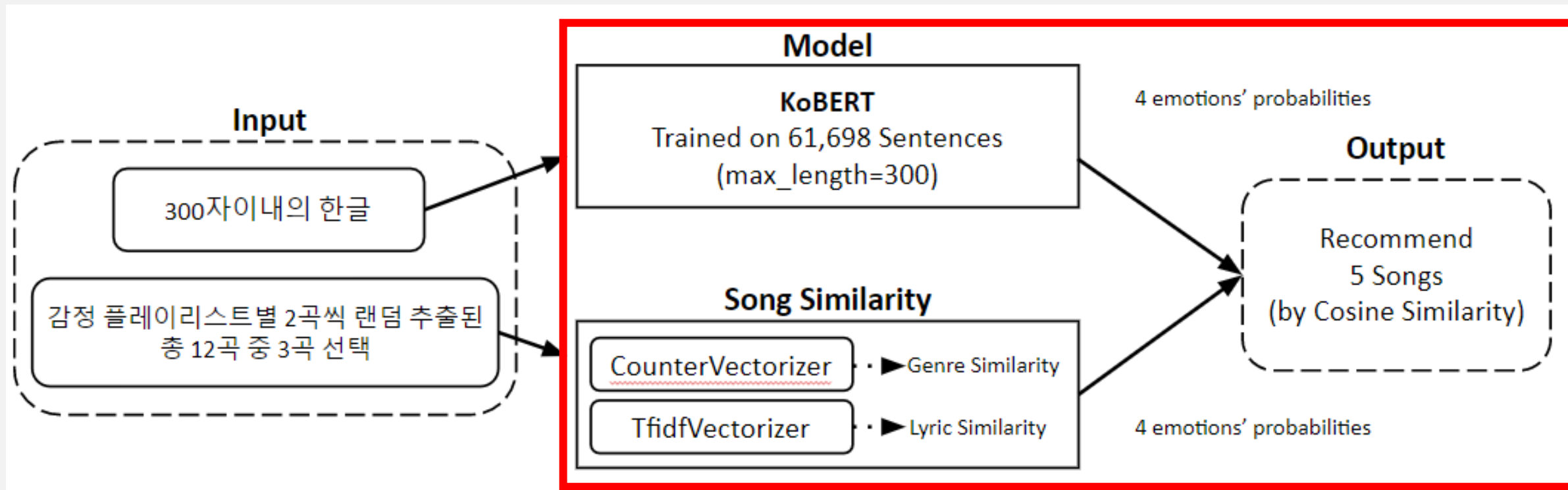
문장 별 감성 분류 확률 추출
 다 더한 후 정규화

= 전체 가사의 감성 확률

3821곡, 총 24 + 2시간 소요..
 (곡 당 20-40초 소요)

	Titl e	Son glD	Arti st	Date	Genre	Lyrics	Emotion	Pro ba	분노 혐오	놀람 공포	슬픔	행복
0	ON	323 998 30	방탄 소년 단	2020. 02.21	랩/힙 합	["I can't understand what people are sayin'", '어느 장단에 맞춰야 될지', '한 발자국 떴으면 한 발자국 커지는 shadow', '잠에서 눈을 뜬 여긴 또 어디']	행복	0.66	0.18	0.07	0.09	0.66
1	어떤 날도, 어떤 말도	353 478 2	심규 선 (Luc ia)	2011. 09.07	발라드, 인디음 악	['함께 했던 많은 계절이', '봄에 눈이 녹 듯 사라진다 해도', '아직 나는 너를 기억 해', '세상 무엇보다 빛나던 모습을', '어 떤 날도 어떤 말도', '']	슬픔	0.69	0.14	0.16	0.69	0.01
2	너 아니 면 안돼	345 927 58	먼데 이 키즈	2022. 02.03	발라드	['오늘도 내 기억을 따라 헤매다', '이 길 끝에서 서성이는 나', '다신 볼 수도 없는 네가 나를 붙잡아', '나는 또 이 길을 묻 는다', '널 보고 싶다고', '']	슬픔	0.37	0.35	0.06	0.37	0.22
3	사랑 이었 나봐	331 582 02	기리 보이	2020. 12.23	랩/힙 합	['사소한 것들 땀에', '크게 다뤘었지', '하 루 종일 말도 안 하고', '궁해 있었지', '사 랑이었나 봐', '그땐 몰랐지만', '전부 다 지나고 나서', '깨달았어', '둘 중']	놀람공포	0.73	0.21	0.73	0.05	0.0
4	봄이 야	347 704 61	로키 드	2022. 03.21	R&B;/ Soul, 인디음 악	['서울은 좁아서', '어딜 가도 네가 따라 와', '다시 돌아온 이 계절처럼', '또 너를 떠올려', '잘 해내고 있다 믿었지만', '넌 내게 불어와', '다시 흔들린 내 마음처럼', "]	놀람공포	0.6	0.16	0.6	0.18	0.06

추천 시스템 알고리즘



곡 유사도 계산

장르 유사도

- ✓ "Genre" 열을 Countvectorizer 적용
- ✓ Cosine_similarity 계산

가사 유사도

- ✓ 처음 스크래핑한 가사는 피어쓰기 처리가 잘 되어 있지 않아 유사도 값이 떨어짐
- ✓
 태그로 피어쓰기 처리를 진행한 가사로 TfidfVectorizer 적용
- ✓ Cosine_similarity 계산

- "랩/힙합" 코싸인 유사도 확인

```
1 genre_result.iloc[:,0].sort_values(ascending=False).iloc[400:500]
```

Genre	
랩/힙합	1.00000
랩/힙합	1.00000
랩/힙합	1.00000
랩/힙합	1.00000
랩/힙합	1.00000
...	
랩/힙합, 인디음악	0.57735
랩/힙합, 인디음악	0.57735
랩/힙합, 인디음악	0.57735
랩/힙합, 국내드라마	0.57735
랩/힙합, 인디음악	0.57735

Name: 랩/힙합, Length: 100, dtype: float64

```
1 song_simi_lyric.mean()
```

0.006770515289072341



```
1 song_simi_lyric.mean()
```

0.012733605365829086

-> 가사 피어쓰기 처리 후 코싸인 유사도 평균 값 상승

곡 유사도 계산

장르 유사도

- ✓ 장르 유사도 * 가중치 (0.3) + 가사 유사도 * 가중치 (1)
- ✓ 가중치는 임의로 설정

```
1 song_simi_co = (  
2     + song_simi_genre * 0.3 # 가중치 * 장르 유사도  
3     + song_simi_lyric * 1 # 가중치 * 가사 유사도  
4 )
```

```
1 song_simi_co_sorted_ind = song_simi_co.argsort()[::-1]  
2 song_simi_co_sorted_df = pd.DataFrame(song_simi_co_sorted_ind)
```

```
1 song_simi_co_sorted_df.head()
```

	0	1	2	3	4	5	6	7	8	9	...	3348	3349	3350	3351	3352	3353	3354	3355	3356	3357
0	0	2643	858	38	2862	1358	3088	2982	442	810	...	2919	3277	1978	1977	3323	1950	1938	2619	1928	1897
1	1	2524	304	429	479	1494	2468	3118	1805	1181	...	2154	2756	326	2150	2755	1791	333	1801	1812	2260
2	2	1712	1903	607	1633	1341	1326	848	1588	906	...	1970	357	1700	2804	1696	842	1454	709	1033	642
3	3	344	1742	1581	3344	2680	111	1169	2224	2048	...	2644	1014	1029	1017	2636	1020	2633	2632	1025	3357
4	4	1645	15	2176	2156	338	1398	3245	1585	1873	...	255	1579	2658	2366	2364	1305	873	1191	1949	2545

해당 인덱스와 유사도 값이 큰
인덱스 순으로 저장

사용자가 선택한 곡과
유사한 곡 추출 가능

유사도 기반 추천 시스템_곡

```
def find_simi_song(seed_song, top_n=10):
```

```
    song_df = pd.read_csv("song_total.csv", index_col=0)
    song_simi_co_sorted_df = pd.read_csv("song_similarity.csv", index_col=0)
    song_simi_co_sorted = np.array(song_simi_co_sorted_df)
```

```
    song_recom=pd.DataFrame(columns=song_df.columns)
```

```
    for seed in seed_song:
        title=song_df[song_df["Title"] == seed]
        title_index=title.index.values
        similar_indexes = song_simi_co_sorted[title_index,:top_n]
        similar_indexes = similar_indexes.reshape(-1)
        song_recom=pd.concat([song_recom,song_df.iloc[similar_indexes].iloc[1:,:]])
```

```
    return song_recom
```

```
1 seed_song=["취중고백", "New Face"]
```

```
1 song_recom=find_simi_song(seed_song,6)
2 del song_recom["SongID"]
3 song_recom.iloc[:, :3]
```

	Title	Artist	Genre
2393	사랑했었다 (Still love you)	이홍기 (FT아일랜드)	발라드
1668	못해준 게 많아서	잠골버스	발라드
1147	멜로를 좋아해	포맨 (4MEN)	발라드
423	미친 것처럼	V.O.S	발라드
1118	태양계	성시경	발라드
2416	YES or YES	TWICE (트와이스)	댄스
1322	사이렌 (Siren)	선미	댄스
1536	음오아예 (Um Oh Ah Yeh)	마마무 (Mamamoo)	댄스
1361	UGLY	2NE1	댄스
401	9INTRO	싸이 (PSY)	댄스



'취중고백', "New Face" 각각
가장 유사한 곡 순서대로 5개 추출

유사도 기반 추천 시스템_감정

```
def user_song_simi(song_recom, total_array):  
  
    song_emotion=np.array(song_recom[['분노혐오', '놀람공포', '슬픔', '행복']]))  
  
    total_array_list=list(total_array[2:-2].split())  
    total_array_list=list(map(float, total_array_list))  
    total_array=np.array([total_array_list])  
  
    song_user_cosine_index=cosine_similarity(total_array, song_emotion).argsort()[ :, :-1]  
    song_recom_index=song_user_cosine_index.reshape(-1)[:5]  
  
    return song_recom.iloc[song_recom_index]
```

유사한 곡들에 태깅된 4가지 감정 확률과 사용자가 입력한 문장의 4가지 감정 확률 값을 코사인 유사도 계산

 사용자의 감정과 가장 가까운 5곡 추천

유사도 기반 추천 시스템

```
1 total_array="[[0.06 0.20 0.13 0.61]]"
2
3 song_recomm_final=user_song_simi(song_recom,total_array)
4 song_recomm_final.iloc[:, :-2]
```

-> 순서대로 사용자 분노혐오, 놀람공포, 슬픔, 행복의 감정 확률값

	Title	Artist	Genre	Emotion	Probability	분노혐오	놀람공포	슬픔	행복
1361	UGLY	2NE1	댄스	행복	0.67	0.06	0.21	0.07	0.67
2416	YES or YES	TWICE (트와이스)	댄스	행복	0.76	0.12	0.06	0.06	0.76
1147	멜로를 좋아해	포맨 (4MEN)	발라드	행복	0.77	0.03	0.01	0.19	0.77
401	9INTRO	싸이 (PSY)	댄스	행복	0.95	0.01	0.03	0.01	0.95
2393	사랑했었다 (Still love you)	이홍기 (FT아일랜드)	발라드	놀람공포	0.52	0.04	0.52	0.12	0.32

코사인 유사도 값이 큰 순서로 추출

➡ 최종 추천 5곡

서비스 개발 Django



1. Index



1. Index

- index/index.html

```
<form method="post" action="{% url 'visitors_book:index' %}" class="post-form">
    {% csrf_token %}
    <p>이름: <input type="text" name="user_name" value=" ">

    <button type="submit" class="btn"> Today is...</button>
</form>
```



2. Visitors_book

TODAY 28 | TOTAL 243423

이시현의 미니홈피



오늘도 피곤한 하루를
끝낸 너에게
CHEERS!

뮤직 이즈 모티브이래
있..

오늘이 감정으로 노@#를 추천 ♪#드려요

NO.1 이시현 (2022.05.31 13:07)



오늘의 기분을 300자 이내로 표현해 주세요.
마침표는 꼭!

제출하기

NO.2 반휘혈 (2002.08.19 13:48)

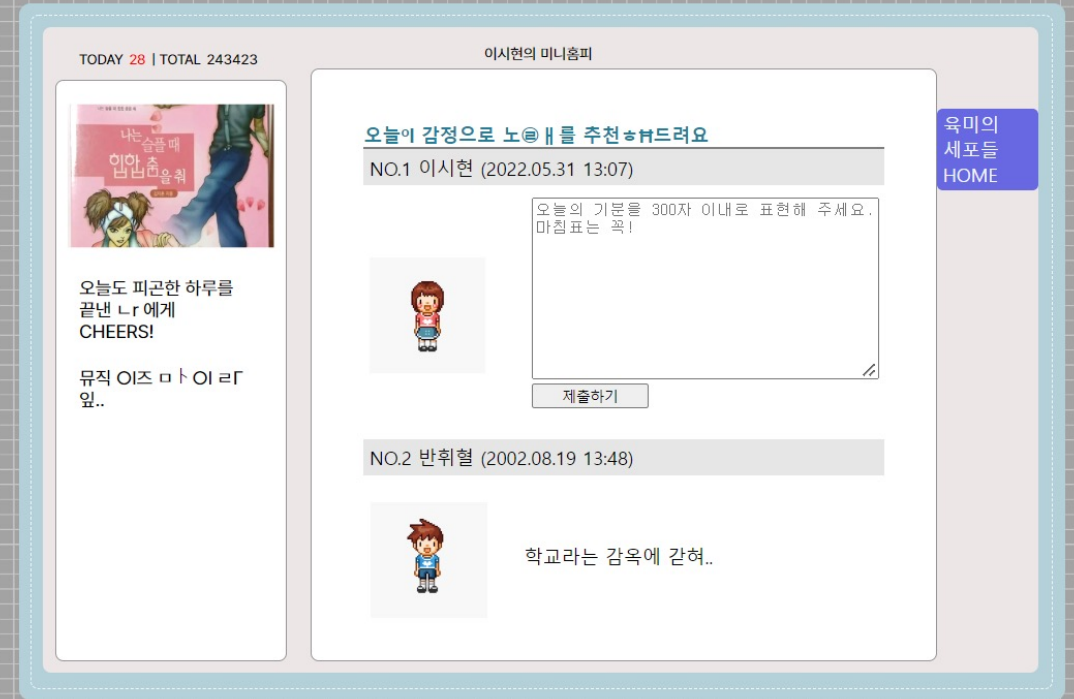


학교라는 감옥에 갇혀..

육미의
세포들
HOME

2. Visitors_book

- visitors_book/views.py :
유저 접속 시간 -> 방명록 작성 시간



```
def visitors_book(request):  
  
    current_time = timezone.now()  
    context = {'current_time':current_time}  
  
    return render(request, 'visitors_book/base.html', context)
```

2. Visitors_book

- visitors_book/base.html : 공란으로 제출하면 에러 메시지 뜨게 함 & 입력 형식 안내 메시지

```
{% if error_message %}
  <p><strong>{{ error_message }}</strong><p>
{% endif %}

<form action="{% url 'today_emotion' %}" method="POST" class="main-form">
  {% csrf_token %}
  <textarea name="target_sentence" cols="38" rows='10' style='font:12px;'
placeholder="오늘의 기분을 300자 이내로 표현해 주세요. 마침표는 꼭!"></textarea><br>
  <input type="submit" style="font-size: 12px; width: 100px;" value='감성 분석
하기' id='input_submit' ></input>
</form>
```

3. today_emoticon

TODAY 28 | TOTAL 243423

이시현의 미니홈피



오늘도 피곤한 하루를
끝낸 Lr에게
CHEERS!

뮤직 이즈 모티브이 러
앞..

오늘의 감성 분석 결과

오늘은

- 0.0%의 분노와 혐오
- 0.0%의 놀람과 공포
- 0.0%의 슬픔
- 99.0%의 행복

노래 추천받기

육미의
세포들
HOME

3. today_emoticon

- visitors_book/views.py - 유저 입력값 감정분석한 4개 감정별 확률값 & 공란 제출시 에러메시지

```
def today_emotion(request):
    try:
        target_sentence = request.POST['target_sentence']
        user_input = str(target_sentence)
        input_list = user_input.split(".")

        processed_input = []
        for sentence in input_list:
            output_string = re.sub(r'[^w\s]', '', sentence)
            processed_input.append(output_string)

        tokenizer = settings.TOKENIZER_KOBERT
        model = settings.MODEL_KOBERT

        total_array, top_pred_prob, top_pred_class, total_array_emotions =
predict_sentiment_func.predict_sentiment_user(processed_input, tokenizer, model)

        context = {'total_array': total_array, 'top_pred_prob': top_pred_prob,
'top_pred_class':top_pred_class, 'total_array_emotions': total_array_emotions}
    except:
        context = {"error_message": "방명록을 입력해주세요."}

    return render(request, 'visitors_book/today_emotion.html', context)
```

3. today_emoticon

- visitors_book/today_emotion.html
: total_array==유저 데이터 분석한 4개 감정 확률값, <input type=hidden' >으로 데이터 전송

```
<div class="miniroom">오늘의 감성 분석 결과</div>
<div class="miniroom-box">
  <form action="{% url 'music_choice:music_choice' %}" method="post">
    {% csrf_token %}
    {% for emotion, percent in total_array_emotions%}
      <h3>오늘은 {{percent}}의 {{ emotion }} 감정을 가지고 있어요</h3>
    {% endfor %}
    <input type='hidden' name="total_array" value="{{ total_array }}">
    <input type='submit' name='제출하기' value='노래 추천받기' style='width: 100px'>
  </form>
</div>
```


4. music_choice

TODAY 28 | TOTAL 243423

이시현의 미니홈피



오늘도 피곤한 하루를
끝낸 너에게
CHEERS!

뮤직 이즈 모티브이러
있..

듣고 싶은 노래를 3개 선택해주세요

☐ 오락실 - 한스 밴드

☐ Gone - 로제 (ROSÉ)

☒ 찐이야 - 영탁

☐ 피 땀 눈물 - 방탄소년단

☒ 그대와 춤을 - 예뵈

☒ All The Way Up (K) - 박재범

☐ Martini Blue - DPR LIVE

☐ Candy Sugar Pop - ASTRO (아스트로)

다시선택하기

제출하기

육미의
세포들
HOME

4. music_choice

- music_choice/views.py

1. KoBERT로 태깅한 감정별 플레이리스트

2. 감정별 2곡씩 랜덤 추출

3. (곡 제목, 가수) 튜플 리스트를 html로 전달

```
def music_choice(request):

    total_array = request.POST["total_array"]

    happiness = pd.read_csv("./data/happiness.csv",
encoding='utf-8', lineterminator='\n')
    anger = pd.read_csv("./data/anger.csv", encoding='utf-8',
lineterminator='\n')
    sadness = pd.read_csv("./data/sadness.csv", encoding='utf-8',
lineterminator='\n')
    surprise = pd.read_csv("./data/surprise.csv", encoding='utf-
8', lineterminator='\n')

    happiness = happiness[['Title', 'Artist']]
    anger = anger[['Title', 'Artist']]
    sadness = sadness[['Title', 'Artist']]
    surprise = surprise[['Title', 'Artist']]

    hap_2 = happiness.sample(n=2, replace=False)
    ang_2 = anger.sample(n=2, replace=False)
    sad_2 = sadness.sample(n=2, replace=False)
    sur_2 = surprise.sample(n=2, replace=False)

    random_songs_df = pd.concat([hap_2, ang_2, sad_2, sur_2])

    random_songs_list = []
    for row in random_songs_df.iterrows():
        random_songs_list.append((row[1][0], row[1][1]))

    context = {'random_songs_list': random_songs_list,
'total_array': total_array}

    return render(request, 'music_choice/music_choice.html',
context)
```

4. music_choice

- music_choice/music_choice.html : for문을 활용해 checkbox 생성 & 곡 리스트 새로고침 기능

```
{% for title, artist in random_songs_list %}
    <div class="form-check">
        <input class="form-check-input" type="checkbox" name="answer[]"
value="{{ title }}" id="flexCheckChecked{{ forloop.counter }}">
        <label class="form-check-label" for="flexCheckChecked{{ forloop.counter }}">
            {{ title }} - {{ artist }}
        </label>
    </div>
    <br>
{% endfor %}

<input type='hidden' name="total_array" value="{{ total_array }}">
<div style="width:90%; margin:10px auto;" >
    <div style="float: right; width:15%; box-sizing:border-box;">
        <button type="submit" class="btn btn-link" dfloat: right;>제출하기</button>
```

4.1 music recommendation

TODAY 28 | TOTAL 243423

이시현의 미니홈피



오늘도 피곤한 하루를
끝낸 너에게
CHEERS!

뮤직 이즈 마이트
있..

힐링 노래 추천 드려요!

- ▶ 짹짹 (Feat. 남승민) (Prod. 영탁) - 정동원
- ▶ KISS ME - DPR LIVE
- ▶ 방황 (Feat. 조지) - basecamp
- ▶ AUTOMATIC REMIX - 찬슬러
- ▶ 벚꽃웨딩 - 이지형

방명록 돌아가기

육미의
세포들
HOME

4.1 music recommendation

▪ music_choice/views.py

1. music_choice/music_choice.html 에서 선택한 checkbox 곡명 받기
2. 선택한 곡과 유사한 곡 추출
3. 추출한 곡과 사용자의 감정 유사도를 계산하여 최종 곡

데이터 html로 전달

+ 아무 곡도 선택하지 않을시 랜덤 추천

+ 추천 알고리즘은 코드는 앞에서 언급한 바와 같음

```
def recommendation(request):
    seed_song = request.POST.getlist("answer[]")

    song_recom =
    music_similarity.find_simi_song(seed_song,6)

    total_array = request.POST["total_array"]

    emotion_recom =
    music_similarity.user_song_simi(song_recom,total_array)

    recom_songs_list = []
    for row in emotion_recom.iterrows():
        recom_songs_list.append((row[1][0],row[1][2]))

    context = {"recom_songs_list":recom_songs_list}

    return render(request,
'music_choice/recommendation.html', context)
```

4.1 music recommendation

- `music_choice/recommendation.html`

전달받은 최종 곡 데이터를 반복문을 이용하여 노래 제목 - 가수의 형태로 출력

```
{% for title, artist in recom_songs_list %}

    &nbsp;
    {{ title }} - {{ artist }} <p>

{% endfor %}
```

활용성 & 프로젝트 발전 방향



활용성 & 프로젝트 발전 방향

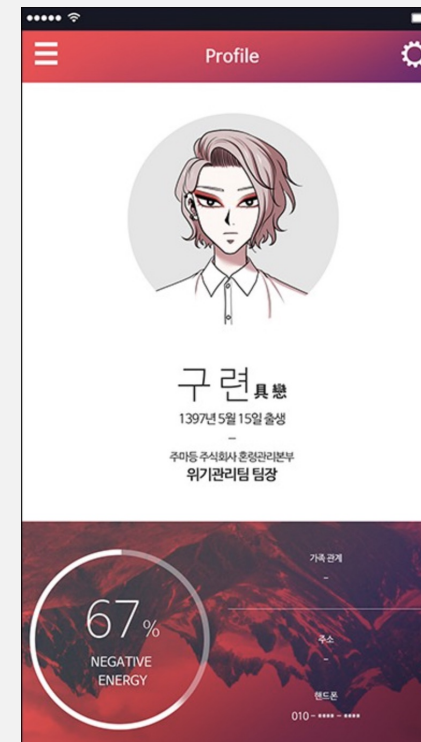
📌 향후 활용성 like 웹툰 '내일'

일상 및 가사 감정분류 데이터셋 구축 [플랫폼화]

- 누적 게시물, 노래 재생 목록 결과를 기반으로 평상시 감정을 수치화, 분석
- 최근 우울증이 의심되는 사용자를 조기에 발견하고 알려
- 치료 현장으로 이끌 수 있는 매체가 될 서비스

📌 프로젝트 발전 방향

- 사용자의 행동, 사용자와 유사한 타인의 행동 기반 협업 필터링 모델 추가반영
- 실시간 대시보드 및 모니터링 : 더욱 직관적인 유저 경험
- 이미지 저장 or sns 공유 기능 (감정 분석 결과, 노래 추천 결과)
- 감정 분류 모델 성능을 높여 정확한 감정 분석 결과 및 맞춤형 노래 추천 제공
- 곡 추천 시 음악 재생 기능 ✅



서비스 시연  by 시현

시현하다 

Thank you
감사합니다