

비건 레시피 추천 서비스

- 멀티캠퍼스 프로젝트 3
- 개발 기간
 - 8월 18일~ 9월 30일
- 개발 인원
 - 6명

요약

- 비건은 비건식을 제공하는 식당이 많지 않아 직접 요리하거나 기성품을 사먹는 의존도가 높습니다. 국내, 국외 비건 레시피 공유 플랫폼에서 추천 시스템을 제공하지 않는다는 사실을 알게되어, 추천시스템을 결합시킨 레시피 공유 서비스를 고안 및 제작하게 되었습니다
- 재료 데이터를 활용하여 레시피를 클러스터링과 CBF, 유저 데이터를 활용한 CF 서비스를 제작하였습니다

기능

* 제가 담당한 파트를 볼드체 표시(핑크색)하였습니다

클러스터링

- **재료 정보를 활용하여 레시피를 4개의 클러스터로 분류, 태그를 제공합니다**
- 로직

1. `download_recipes()` | `recommender_systems.py`

- 레시피 데이터를 `mysql`에 적재한 후, `pymysql` 모듈을 활용하여 데이터를 데이터프레임 형태로 불러옵니다

2. `C2_get_preprocessed_recipe()` | `recommender_systems.py`

- 재료 데이터의 복잡한 데이터구조 ([레시피:[재료 리스트]], [재료1,재료2,...,재료_n] : 재료_j는 str거나 dict인 구조))를 가공, 불용어 제거,stemming등의 전처리를 진행합니다

3. `C3_TF_IDF()` | `recommender_systems.py`

- 레시피에 대한 재료단어의 TF-IDF값을 계산하여, 빈도순 주요 단어 100개를 선정합니다.
- 주요 재료 단어에 대한 레시피의 TF-IDF값을 계산합니다

4. `visualize_silhouette()` | `recommender_systems.py`

- 클러스터 수를 바꾸면서 KNN 클러스터링을 시행하여 실루엣 계수를 측정합니다. 평균 실루엣 계수와 실루엣 모양을 고려하여 최적의 클러스터 수를 결정합니다

5. `make_clusters()` | `recommender_systems.py`

- 다소 레시피가 새로이 업데이트하더라도 각 클러스터에서 반복적으로 등장하는 재료들이 존재합니다. 그러한 재료들을 활용하여 자동으로 클러스터를 명명합니다

- 반복적으로 등장하는 재료들이 어느 지역,문화권에서 자주 활용되는지 분석하였고, 그 결과 각 클러스터에 지역명을 할당하였습니다
- `visualize_cluster_3d()` or `visualize_cluster()` | `recommender_systems.py`
 - 100차원 공간속 TF-IDF 벡터를 PCA 차원축소 후 2차원 or 3차원으로 클러스터의 분포를 시각화합니다

추천 시스템

• 재료 정보와 유저 아이템 정보를 활용하여 CBF를 제공합니다

- CBF 로직
 1. `recommend_by_algorithm()` | `views.py`
 - 로그인한 유저의 ID 정보를 받아 `recommender_sysmtems.py`의 입력값으로 줍니다
 - 5번까지 실행되면 해당 결과를 html 문서로 `render` 합니다
 2. `download_rating()` | `recommender_systems.py`
 - DB에서 레시피에 대한 유저의 평점 정보를 불러옵니다
 3. `make_CBF_model()` | `recommender_systems.py`
 - 레시피-재료 데이터를 document-word 관계로 대응, `doc2vec` 모듈을 적용시켜 각 레시피를 임베딩 벡터로 변환하는 모델을 생성합니다
 4. `CBF()` | `recommender_systems.py`
 - 유저의 선호 레시피(별점 4점 이상이라 정의함)들을 임베딩 벡터로 전환시켜 평균을 내고, 다른 레시피 벡터들과 코사인 유사도값을 계산하여, 가장 코사인 유사도값이 높은 레시피명 20개를 반환합니다
 5. `recommended_recipe_data_by_CBF()` | `recommender_systems.py`
 - `CBF()`에서 얻은 레시피명을 활용하여 레시피 데이터들을 DB에서 가져옵니다
- `metric_CBF()` | `recommender_systems.py`
 - 각 유저가 추천받은 레시피의 평균 코사인 유사도값을 계산하여 `lineplot`으로 시각화합니다
- 유저 아이템 정보를 활용하여 CF를 제공합니다

검색과 필터링

- 특정 재료가 들어간 레시피를 검색하는 기능을 제공합니다
- 검색이나 추천에서 특정 재료를 제외하거나, 특정 카테고리만의 레시피만 출력시키는 필터링 기능을 제공합니다
- 추천 필터링 로직

1. `main_login_q()` | `views.py`
 - 로그인한 유저의 ID 정보를 받아 `recommender_sysmtems.py`의 입력값으로 줍니다
 - Django의 Q 모듈을 활용하여 필터링된 레시피값을 받아 데이터프레임 형태로 저장합니다
2. `recommend_by_filtered_algorithm()` | `views.py` ,
 - `filtered_recipe_data_by_CBF()`, `filtered_recipe_data_by_CF()` | `recommender_systems.py`
 - 추천 레시피와 필터링된 레시피의 교집합인 레시피를 찾아 반환합니다

- 해당 결과를 html로 render 합니다
- 검색 필터링 로직
 1. `search_result_q | views.py`
 - Django의 Q 모듈을 활용하여 필터링된 레시피값을 받아 해당 결과를 html로 render 합니다

주요 레퍼런스

- 프로젝트 기획안
- WBS