졸업프로젝트 2차 발표

컴퓨터전공 권나현, 송지수

목차

1. 프로그램 소개
2. 프로그램 구조
3. 개발 계획 / 현황
4. Q&A

프로그램 소개

- 텍스트 분석(Word2Vec) 기반의 물품 추천 시스템
- Word2Vec
  - 입력된 코퍼스를 기반으로 단어들을 벡터로 표현하는 방법
  - 이를 이용하면 단어간 유사성을 구할 수 있으며, 문서를 군집화한 뒤에 결과를 이용하여 추천 시스템을 만들 수 있음

프로그램 구조

[Transaction data : (id, uri, new_query_string, referer)
Products Info : (prod_code, prod_type, prod_nm, ctry_cd, keyword)]

JOIN → prod_info : (id, prod_code, prod_type, prod_nm, ctry_cd, keyword)

INPUT → 단어 빈도수 CHECK MODULE → OUTPUT (id1 : {Word1 : 10, Word2 : 8, …},
id2 : {Word1 : 12, Word2 : 7, …}, …)

INPUT → 최빈 단어 FILTERING MODULE → OUTPUT (id1 : {Word1, Word2, …},
id2 : {Word4, Word6, …}, …)

INPUT
+ (prod_info) → 추천 상품 추출 MODULE → OUTPUT (id1 : {Prod1, Prod2, …},
id2 : {Prod4, Prod5, …}, …)

→ WEB에 표시

모듈 설명

단어 빈도수
CHECK MODULE

INPUT — prod_info : (id, prod_code, prod_type, prod_nm, ctry_cd, keyword)

OUTPUT — (id1 : {Word1 : 10, Word2 : 8, …}, id2 : {Word1 : 12, Word2 : 7, …}, …)

- prod_nm, keyword, (크롤링이 가능한 경우) 상품의 설명에 포함된 단어들을 세어, 각 사용자마다 확인한 상품들의 정보 중 어떤 단어들이 몇 번 나타나는지 구함
- 불용어는 제거, 명사만을 counting
- 불용어 제거에 KoNLPy 라이브러리를 사용 예정

모듈 설명

최빈 단어
FILTERING MODULE

INPUT

(id1 : {Word1 : 10, Word2 : 8, …},
id2 : {Word3 : 12, Word4 : 7, …}, …)

OUTPUT

(id1 : [{Word1, Word2, …}, {Word5, Word6, …}],
id2 : [{Word3, Word4, …}, {Word7, Word8, …}], …)

- 각 사용자마다 계산된 단어 출현 횟수를 참고해, 빈출도 상위 단어를 해당 사용자의 취향을 나타낼 수 있는 단어 리스트로 정함
- 단어들을 Word2Vec 알고리즘을 이용해, 서로 연관성이 있다고 판단되어 같은 주제로 묶일 수 있는 단어들을 클러스터링 하고 해당 단어끼리 따로 묶음
- 클러스터링 처리에는 gensim 라이브러리를 사용

모듈 설명

추천 상품 추출
MODULE

INPUT (id1 : [{Word1, Word2, …}],
id2 : [{Word3, Word4, …}], …) + (prod_info)

OUTPUT (id1 : {Prod1, Prod2, …},
id2 : {Prod4, Prod5, …}, …)

- 사용자의 취향을 나타낼 수 있는 단어 리스트와, 각 상품 정보와의 유사도를 측정하여 유사도가 높은 순서대로 추천 상품을 추출함
- 유사도 계산 방법으로는 Cosine Similarity를 이용할 예정

개발 계획 / 현황

- Transactions 데이터와 상품 정보 데이터는 연구실 MySQL 서버에 넣어놓고 개발을 진행할 예정
- 추후에 NoSQL (Redis, MongoDB) DB 환경이 필요할 경우, 서버 구축에 대한 추가 논의가 필요 (Local or Cloud or etc...)
- 25GB의 Transactions 데이터는 Python을 이용해 DB에 추가 예정 (현재 개발 중)

Q&A

감사합니다