**“카페레코”**

[워드 투 벡터를 이용해 제작한 추천 알고리즘 자연어 처리 프로젝트]

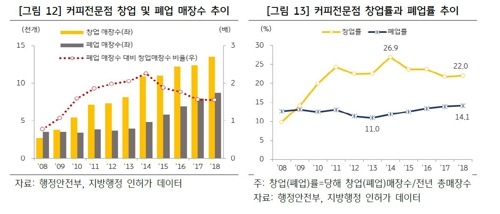
이찬호, 최명혜

요약

카페는 남녀노소 가리지 않고 누구에게나 친숙한 장소이며 높은 수요를 가지고 있다. 그에 맞춰 현재 전국에는 카페가 10만개 이상 존재하는 상황이다. 다양한 개성과 특징을 가진 카페들이 늘어나는 현대에 사용자가 가고 싶은, 원하는 취향의 카페를 찾기 쉽게 돕는 사이트가 있다면 편리하고 실용성이 있을 것이라고 판단하였다. “가고 싶은 타입의 카페를 쉽게 추천받고 싶다”라는 문제를 해결하기 위해서 가장 양이 많으며 다양한 리뷰를 수집할 수 있는 구글 리뷰에서 데이터를 수집하고, 문장들을 정제하여 카페의 키워드가 되는 단어들을 추출한다. 또, 그 단어들에 벡터 값을 부여해 카페를 이루는 단어들의 평균을 내거나 하는 등의 계산 과정을 거쳐 벡터 값을 기준으로 유사도를 비교하여 사용자에게 서비스를 제공한다.

키워드: 크롤링, 빅데이터, 자연어 처리, 워드투벡터

1. 서론



*그림 1) 커피전문점 창업 매장수 증가 추이*

최근 카페의 입지는 세대를 가리지 않고 즐겨 찾는 정도의 접근성을 가지고 있으며 그 수요만큼 수도 점점 늘어나고 있고 그만큼 다양한 개성의 카페들이 존재한다.

소상공인진흥공단 전국 카페 점포수의 기준에 따른 데이터 수집 상황에 따르면 2019년 기준으로 9만 8,779개의 카페가 운영되고 있고 코로나의 영향으로 그 기세가 줄어들 것이라 예상한 것과 다르게 “코로나가 시작된 후 다른 업종은 10%가량 폐업했지만 상대적으로 진입 장벽이 낮다고 느끼는 카페는 20% 이상 신규 창업이 늘었다”고 한다. [1][2]

이러한 카페의 인기는 쉽게 유추할 수 있다. 카페는 더 이상 단순히 커피와 같은 음료를 마시러 가는 곳이 아니라 그 외 다양한 활동을 목적으로 향하는 장소가 되었다. 잠시 밖에서 시간을 보내기 가장 쉽고 적합한 곳인 만큼 약속의 장소로 많이 이용하며 바쁘게 돌아가는 일상의 휴식처가 되어준다. 대학생들에게는 휴식이나 여가뿐만 아니라 공부를 하러 가기도 한다. 실제로 카페에서 공부를 한다는 신조어인 ‘카공’ 이라는 단어가 생겼을 정도로 카페는 우리의 일상과 밀접하게 관련이 되어있다.

리뷰 기능 또한 우리 팀이 집중한 요소이다. 사람들이 방문하고 남긴 리뷰들을 읽어보는 것만으로 그 장소의 환경과 분위기 등 많은 것을 쉽게 연상시킬 수 있다. 사람들이 카페를 선택할 때에 중요하게 여기는 요소 또한 다양하다고 한다. 커피의 맛부터 카페까지의 거리 등 여러 종류의 요소들이 카페를 선정하는데 영향을 끼칠 수 있다. 2017년 트렌드 모니터에서 실시한 설문조사[3]에 따르면 커피의 맛, 가격, 브랜드, 종류와 같은 커피에 관련된 요소들이나 카페까지의 거리, 카페의 분위기, 좌석의 안락함, 이벤트나 행사 등의 커피와 직접적인 관련이 없는 요소들 또한 카페 방문 시 고려사항으로 수집된 것을 살펴볼 수 있었다. 이러한 요소들에 관한 정보를 수집하기 위해 모든 카페를 직접 찾아갈 수는 없다. 하지만 사용자들이 남긴 리뷰에 이러한 내용들이 충분히 담겨있기 때문에 리뷰들을 수집하는 계기가 되었다.

이처럼 다양한 개성과 특징을 가진 카페들 사이에서 내가 원하는 카페를 쉽고 간편하게 추천 받을 수 있다면, 직접 방문하지 않고 여러 리뷰나 후기를 하나하나 확인하며 카페를 선택하는 번거로움이 줄어들어 편리하고 실용성 있을 것이라 생각했기 때문에 이러한 주제를 선정하였다.

1. 개발환경 및 시스템 설계 및 구현

*그림 2) 개발 환경*

2.1 정보 크롤링

‘카페레코’의 개발 환경의 정보를 모으는 과정에서는 Pycharm과 Google Colab으로, Python 환경에서 웹 어플리케이션 자동화와 테스트 포터블 프레임워크 Selenium과 웹드라이버를 사용하였다. 크롤링에 사용할 데이터는 소상공인진흥공단 상권 정보의 데이터를 사용해 불러와서 사용하였다. 카페의 정보들을 직접 수집하는 방식은 모든 카페를 수집하지 못하고 누락되는 경우가 생길 것을 염려해 공공 데이터를 이용하였다.

2.2 자연어 처리

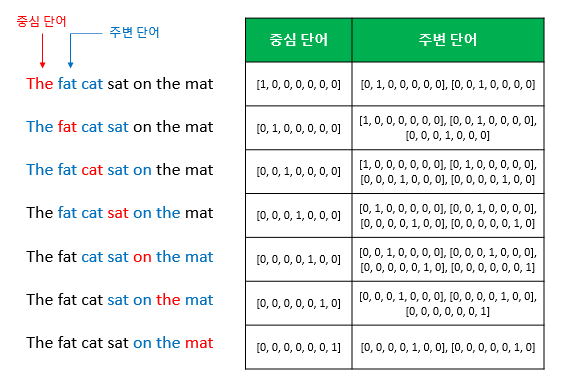
자연어 처리 과정은 Google Colab에서 Python을 이용해 진행되었다. 추출된 리뷰 텍스트들을 Konply의 한국어 처리 패키지 Okt를 통해 내용을 정제해 문장에서 한글 단어만 남겼다.[4] 패키지만으론 100% 온전하게 걸러내지 못하기 때문에 그 외의 사용이 어려운 단어들을 불용어로 선정해 추가로 제거하였다. 이후 남아있는 단어들에 벡터 값을 부여한다. 이때 벡터 값을 부여하기 위해 워드 투 벡터라는 알고리즘을 사용했다. Gensim 패키지에는 Word2Vec을 지원하고 있어 이 패키지를 사용하면 단어를 임베딩 벡터로 변환시킬 수 있다.

* + 1. 워드투벡터

워드투벡터는 워드 임베딩에 기반하여 단어 간의 유사도를 벡터화 하는, 단어의 의미를 수치화하여 유의미한 유사도를 계산할 수 있게 만드는 대표적인 알고리즘이다. 분산 표현(Distributed Representation)을 통해서 단어의 의미를 다차원 공간에 벡터화하고 이러한 표현 방법은 단어 벡터 간의 유의미한 유사도를 계산할 수 있게 한다.[5]

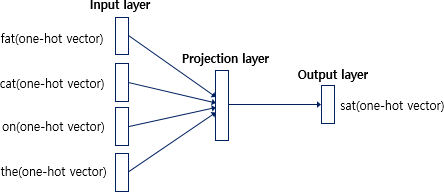
분산 표현은 저차원에 단어의 의미를 여러 차원에다가 분산하여 표현해 단어 벡터 간의 유사도를 계산한다 비슷한 문맥에서 등장하는 단어들은 의미 또한 비슷하다고 가정하고 진행한다. 이러한 가정을 분포 가설(distributional hypothesis)이라고 하는데 해당 내용을 가진 텍스트들을 벡터화 한다면 문장에 포함된 단어들은 유사한 벡터 값을 가진다. 예를 들어 *빵*이란 단어에 *맛있다, 달콤하다* 등의 단어가 주로 함께 등장하기 때문에 이러한 단어들은 비슷한 값을 갖는다.

워드투벡터의 학습 방식으로는 CBOW(Continuous Bag of Words) 와 Skip-Gram 두 방식이 가장 대표적이다. CBOW는 주변에 있는 단어들을 추측의 지표로 삼아 중간에 있는 단어들을 추정하는 방법이고 반대로, Skip-Gram은 중간에 있는 단어를 기준으로 주변 단어들을 예측한다. CBOW는 예측해야 하는 단어를 중심 단어(center word)로 정하고 주변 단어를 주변 단어(context word)라고 한다. 중심 단어 앞뒤로 몇 개의 단어를 사용할지 범위를 정하는데 이는 윈도우(window)라고 한다.



*그림 3) 윈도우 크기가 2일 때, 슬라이딩 윈도우 예문*

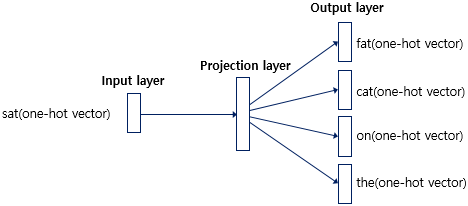
이 모델 학습의 목적은 주변의 단어들이 주어졌을 때 다음의 center 단어의 조건부 확률을 최대화 하는 것이다. 학습시킬 문장의 모든 단어들을 원-핫(one-hot encoding)방식으로 벡터화한다. 특정 위치에만 1이 들어가고 나머지에는 0이 들어가는 형식이다. 윈도우 크기를 정한 후 중심 단어와 주변 단어를 바꿔가며 학습을 위한 데이터셋들을 만들고 이러한 방식을 슬라이딩 윈도우(sliding window)라고 한다. 위 그림에서 중심 단어와 주변 단어의 선택에 따른 원-핫 벡터로 만드는 데이터셋을 확인할



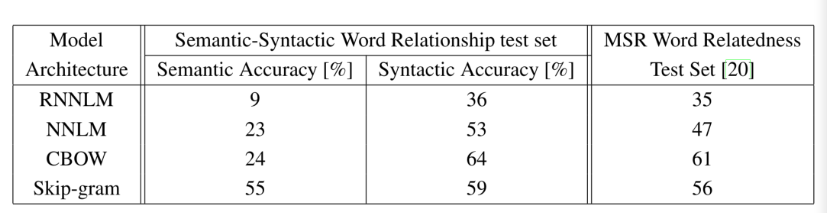
*그림 4) CBOW의 인공 신경망 도식화*

CBOW의 인공 신경망을 도식화하면 입력층(input layer)에 사용자가 지정한 윈도우 크기 범위 내 주변 단어들의 원-핫 벡터가 입력으로 들어가고 입력값이 투사층(projection layer)를 통과하면 출력층(output layer)에서 중간 단어를 예측한 벡터를 출력한다. 출력층은 단어 개수 만큼의 노드로 이루어져 있으며 활성화 함수로 소프트맥스를 사용한다. 그림에서 볼 수 있는 것 과 같이 CBOW는 은닉층이 다수인 깊은 인공신경망, 흔히 말하는 딥러닝 모델이 아닌 단일 은닉층만 있는 얕은 신경망(shallow neural network)모델이다.

Skip-Gram은 반대로 중심 단어에서 주변 단어를 예측한다. 중심 단어에 대해서 주변 단어를 예측하므로 투사층에서 벡터들의 평균을 구하는 과정이 없다. 성능 비교를 진행했을 때 전반적으로 Skip-gram이 CBOW보다 성능이 좋다고 알려져 있다.



*그림 5) SKIP-GRAM의 인공 신경망 도식화*



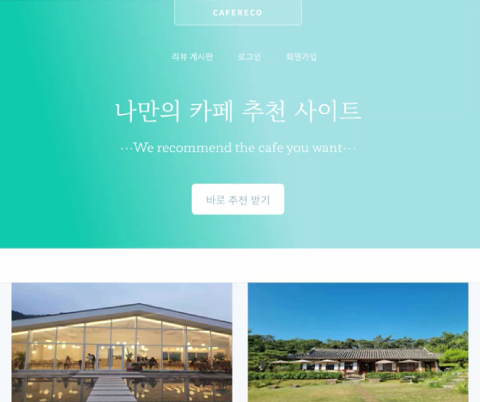
*그림 6) CBOW와 SKIP-GRAM 성능 비교*

2.3 데이터베이스

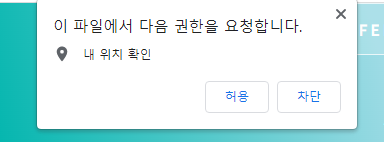
앞선 과정들을 거쳐 크롤링한 정보들에 자연어 처리를 거쳐 벡터 값을 담은 CSV 파일을 Mysql과 Phpmyinfo를 이용해 등재한다. 크로스 플랫폼 웹서버 Xampp를 이용해 Mysql과 Apache를 연동했다. 프론트엔드에서 사용할 수 있도록 PHP에서 서버 이름, 사용자 이름, 패스워드, DB 이름 등을 입력해 연동한다. 테이블 형태로 저장돼있는 DB를 불러오기 위해 해당하는 이름에 맞는 열을 불러와서 html을 호출하고 테이블 형태로 만들었다.

* 1. 프론트엔드

테이블 형태로 저장돼있는 DB를 불러오기 위해 해당하는 이름에 맞는 열을 불러와서 html을 호출하고 테이블 형태로 만든다. Pyscript를 이용해 사용자가 입력한 키워드를 벡터 값으로 변환한다. 변환된 값은 DB에 담은 값을 검색하는데 이용한다. 웹사이트에 접속하면 가장 먼저 보이는 메인 페이지에서 페이지로 넘어갔을 때 사용자의 위치 정보 수집을 허용하면 구글 API를 따로 받아와서 사용자의 위치 정보를 구글 지도에 표시하고, 이 정보는 검색에 앞서 사용자의 위치에 맞는 곳을 검색하는데 사용한다. 마지막으로 사용자가 원하는 키워드를 입력할 수 있는 검색 페이지를 표시해준다. 이 페이지에서는 사용자가 입력한 키워드에 대한 카페의 정보 결과 또한 출력된다.

*그림 7) 프론트엔드의 메인 화면 그림 8) 검색 화면*





*그림 9) Google API 사용 위치 받아오기*



*그림 10) 키워드 ‘빵’ 검색 결과*

1. 결론

가고 싶은 카페를 쉽게 고르고 싶다는 사소한 문제에서 시작해 제작한 카페레코는 이렇게 완성됐다. 카페레코를 잘 활용한다면 여행이나 출장 등으로 낯선 장소에 왔을 때 쉽고 간단하게 자신이 원하는 카페를 찾을 수 있을 것이다. 리뷰에서 정보를 모으고 정제하였기 때문에 리뷰나 후기 같은 문장으로 이루어진 정보가 존재하는 장소나 물건이라면 어떤 것이든 비슷한 방식으로 추천이 가능하다는 점에서 활용이 가능하다. 좀 더 업그레이드 하자고 하면 또한 모바일 지원이 된다면 미리 찾아보는 경우가 아니라 즉석에서, PC가 없는 상황에서도 추천 받을 수 있기 때문에 모바일 웹페이지를 구현하고 싶다. 또 일정 리뷰가 쌓여 카페의 테마가 될만한 키워드의 정보가 모일 때까지 꽤 많은 시간이 걸리기 때문에 이슈가 되지 못한 새로 생긴 카페와 같은 경우의 cold start 이슈에 대한 확실한 해결법을 제시하지 못한 것도 아쉽다는 점이다.

참고 자료:

<https://www.data.go.kr/data/15067614/fileData.do#/tab-layer-openapi>

(소상공인시장진흥공단 전국 카페 점포 수)

<https://www.sedaily.com/NewsView/263K07C3ZK>

(급증하는 커피 개인 사업자)

<https://catchmind-10.tistory.com/117>

(2017년 트렌드모니터 설문조사 결과)

<https://datascienceschool.net/03%20machine%20learning/03.01.02%20KoNLPy%20%ED%95%9C%EA%B5%AD%EC%96%B4%20%EC%B2%98%EB%A6%AC%20%ED%8C%A8%ED%82%A4%EC%A7%80.html>

(KoNLPy 한국어 처리 패키지)

<https://wikidocs.net/22660>

(Word2Vec)