DB 속 데이터 상호명 지점명 중분류 소분류 지번주소 도로명주소 신우편번호

+ 카페 키워드들, 평균 별점, 리뷰 개수(평균 별점 \* 리뷰 개수가 높은 순서일수록 상위로 갈 수 있게 하고 order by desc로 정렬) 유사도로 배점

현재 위치 파악 + 가는 길 연동

**UI, UX 위주로 추가**

DB에 카페 관련 단어들 키워드화해서 적재함

검색 단어 keyword로 받음

Keyword 벡터화 후 검색에 사용

SELECT keyword FROM café\_info WHERE

발표 자료

계획: 주제 선정 동기 + 나의 해결 방안 설명 방법 제시 + 내가 사용할 계획인 방법(실제로 구현 됐는지랑은 별개) 요게 핵심

결과: 실제 완성본

고민

PPT 재사용?

발표 각자 맡은 부분 자기가 읽기? OR 한명이 다 하기

실패한 부분 언급 방법

주제 선정 동기

카페는 남녀노소 연령 가리지 않고 누구에게나 친숙한 장소 + 높은 수요 -> 전국 카페 10만개 시대 -> 그만큼 다양한 개성을 가진 카페가 많기 때문에 내가 원하는 취향의 카페를 찾기 쉽게 돕는 사이트가 있다면 편리하고 실용성 있을 것이라고 생각했기 때문에 선정하게 되었다.

나의 해결 방안 설명과 방법 제시

구글 지도에서 카페들을 찾아보면 확인할 수 있는 리뷰들과 별점들을 이용한 빅데이터들을 취합해 유의미하게 사용해보자 -> 크롤링을 통해 앞서 말한 내용들을 자동화하여 수집 -> 자연어 처리를 통해 수집한 내용들을 가공하여 불순물들을 걸러내고 유의미한 내용들만 남기기 -> 처리 후 남은 내용들을 기반으로 본격적인 제작

내가 사용한 방법 및 계획(한거 아녀도 넣음)

Word2Vec을 이용해 검색 알고리즘을 만들자:

데이터들을 수집하고 wordcloud를 이용해 내용들을 정제해 문장들에서 유의미한 단어들을 추출하고 벡터화시켜 각 단어별의 연관성을 학습시켰다. mysql를 이용해 테이블에 정제한 데이터들을 집어 넣는다. 웹페이지 메인에 있는 검색창에 자기가 원하는 아무 내용이나 입력하면 마찬가지로 앞서 했던 일련의 동작처럼 문장에서 단어들만 남기고 불용어 처리해 유의미한 키워드만 남기고 DB에 등재된 카페들과 벡터의 유사도를 비교해 최대한 관련있는 카페들을 추천할 수 있게 함.

(기존에 사용한 문구)사용자가 이용하기 쉽게 원하는 키워드를 아무렇게나 검색해도 벡터화 된 값을 출력시킬 수 있게 하기 위해 wordcloud 알고리즘을 통해 입력한 내용과 최대한 유사한 키워드들을 갖고 있는 카페들을 출력할 수 있게 알고리즘을 완성시켰다.

깃허브를 통한 버전관리: 대충 넣기

1. 데이터 크롤링 단계

사용 Python 기반, Selenium과 Webdriver를 이용해 자동화처리

리뷰페이지로 이동하는 과정 – 자동화 단계는 세세한 동작까지 다 입력해줘야함 google.co.kr/maps를 실행하고 검색창에 입력하고 검색 버튼을 입력하는 등 모든 동작을 코드로 입력해야함

자동 스크롤 단계 – 구글 맵 리뷰는 처음부터 모든 리뷰를 보여주는 것이 아니라 스크롤 맨 밑으로 내리면 20개씩 리뷰를 추가로 보여주는 형태 + 추가로 리뷰의 길이에 따라 들쑥날쑥해지는 스크롤의 길이 - 기존의 스크롤 쭉 내리는 동작만으론 처음 20개만 보여주기 때문에 200개의 리뷰를 수집하기 위해선 스크롤을 내리는 동작을 추가로 19번 더 반복해야함

Csv 형태로 저장 – pandas를 이용해 테이블 형태로 데이터들을 저장하고 이를 csv 형태로 옮김

Pandas를 이용해 데이터프레임화 + csv 형태로 저장

1. 자연어 처리 단계

크롤링한 리뷰들 문장들 1차로 단어들만 남기고 제거 2차로 의미 있는 단어들만 남기기 위해 불용어 처리

Word2Vec에 이용한거 okt 같은 패키지 전부 적어줘~

1. 데이터베이스 만들기 단계

Mysql를 이용해 자연어 처리 단계를 거친 내용들 데이터베이스에 등재

Csv 형태로 저장했기 때문에 sql에 불러오기 방식으로 쉽게 등재가 가능함 phpmyinfo를 사용해 db들을 관리하였음

Xampp를 이용해 Mysql과 Apache 연동

1. 웹페이지 만들기 단계

HTML 기반으로 CSS와 JavaScript를 이용해 제작

PHP를 이용해 데이터베이스와 연동

Pyscript를 이용해 자연어 처리와 웹페이지 연동

피드백 받아서 추가한 부분(임의)

요구 받은건 GPS랑 필터링 옵션(필수 추가)

주제 선정 동기

카페는 남녀노소 연령 가리지 않고 누구에게나 친숙한 장소 + 높은 수요 -> 전국 카페 10만개 시대 -> 그만큼 다양한 개성을 가진 카페가 많기 때문에 내가 원하는 취향의 카페를 찾기 쉽게 돕는 사이트가 있다면 편리하고 실용성 있을 것이라고 생각했기 때문에 선정하게 되었다.

나의 해결 방안 설명과 방법 제시

구글 지도에서 카페들을 찾아보면 확인할 수 있는 리뷰들과 별점들을 이용한 빅데이터들을 취합해 유의미하게 사용해보자 -> 크롤링을 통해 앞서 말한 내용들을 자동화하여 수집 -> 자연어 처리를 통해 수집한 내용들을 가공하여 불순물들을 걸러내고 유의미한 내용들만 남기기 -> 처리 후 남은 내용들을 기반으로 본격적인 제작

내가 사용한 방법 및 계획(한거 아녀도 넣음)

Word2Vec을 이용해 검색 알고리즘을 만들자:

데이터들을 수집하고 wordcloud를 이용해 내용들을 정제해 문장들에서 유의미한 단어들을 추출하고 벡터화시켜 각 단어별의 연관성을 학습시켰다. mysql를 이용해 테이블에 정제한 데이터들을 집어 넣는다. 웹페이지 메인에 있는 검색창에 자기가 원하는 아무 내용이나 입력하면 마찬가지로 앞서 했던 일련의 동작처럼 문장에서 단어들만 남기고 불용어 처리해 유의미한 키워드만 남기고 DB에 등재된 카페들과 벡터의 유사도를 비교해 최대한 관련있는 카페들을 추천할 수 있게 함.

(기존에 사용한 문구)사용자가 이용하기 쉽게 원하는 키워드를 아무렇게나 검색해도 벡터화 된 값을 출력시킬 수 있게 하기 위해 wordcloud 알고리즘을 통해 입력한 내용과 최대한 유사한 키워드들을 갖고 있는 카페들을 출력할 수 있게 알고리즘을 완성시켰다.

+추가한 부분

크롤링한 데이터 (카페의 정보)를 자연어 처리 한다. 문장 형식으로 구성되어있는 리뷰에서 명사로 된 키워드만 필터링하고, 해당 단어들을 불용어 혹은 카페의 키워드로 선정한다. 불용어가 되는 기준은 여러 카페에서 공통으로 겹치는 단어, 혹은 단어 하나만으로 키워드가 될 수 없는 것으로 선정했다.

사용자가 입력한 키워드는 크기 200의 벡터값으로 즉석에서 바꾼다. 사전 훈련된 pre-trained word2vec embedding은 박규병님께서 공개한 모델을 사용한다.

깃허브를 통한 버전관리: 대충 넣기

1. 데이터 크롤링 단계

사용 Python 기반, Selenium과 Webdriver를 이용해 자동화처리

Pandas를 이용해 데이터프레임화 + csv 형태로 저장

2. 자연어 처리 단계

크롤링한 리뷰들 문장들 1차로 단어들만 남기고 제거 2차로 의미 있는 단어들만 남기기 위해 불용어 처리

Word2Vec에 이용한거 okt 같은 패키지 전부 적어줘~

+추가한 부분

파이썬은 3.7.9 버전으로 gensim은 3.8.3 버전으로

Word2Vec에 이용한 것

konlpy : 여기서 포함된 okt에서 morphs / nouns 를 사용해 문장을 형태소로 나누고 명사만 추출했다

gensim : 필터링 한 키워드를 벡터화 시키는데에 편리한 기능을 제공한다.

numpy : 코사인 유사도 함수 구할 때 가장 많이 사용함

- np.dot(), np.sqrt()등의 함수를 사용해 유사도를 구함

- 자연어 처리 과정 조금 더 자세하게

1 csv 데이터를 불러온다

2 종합된 데이터에서 리뷰 텍스트만 따로 추출한다

3 기존에 분리해둔 불용어 파일을 불러온다

4 리뷰를 긁어오며 생긴 빈 리뷰 칸을 없앤다

5 한글 리뷰만 남긴다

6 리뷰들을 토큰화 한다. 동시에 konlpy의 okt를 사용해 문장에서 명사만 남긴다

7 동시에 3에서 불러온 불용어 파일을 사용해 남은 명사 중 불용어는 지운다

8 리뷰 텍스트당 한 리스트를 가지고 있는 배열들을 다 합쳐준다

9 테마 분류 및 벡터값을 구한다

9-1 테마 분류

추출된 단어들에서 반복되는 특정 키워드들을 선정해서 추천할 인원, 시간대, 디저트, 분위기, 커피의 산미등을 판별한다

예시) 분위기, 풍경, 경치, 뷰, 감성 등의 키워드를 "분위기" 테마의 키워드로 선정, 얼마나 많이 언급되는지 그 수를 기록한다.

다른 카페들과 비교해 분위기 순으로 정렬할 수 있도록 한다.

예시2) 침묵, 사색, 책, 혼자 / 가족, 연인, 친구, 부모님 등의 키워드로 나눠서 "인원수" 테마의 키워드로 선정,

전자가 더 많이 언급 됐을 시 혼자가기 좋은 카페로, 후자가 더 많이 언급 됐을 시 다인이 가기 좋은 카페로 분류한다.

\* 기존에 사용할 방식은 아래에서와 마찬가지로 벡터 크기 비교로 진행하려고 했으나 의미있는 분류가 힘들어져 가능한 특정 키워드를

많이 선정하는 방식으로 변경했다

9-2 위에서 분류한 테마의 데이터를 따로 만들어 보관해둔다.

10 카페 별로 단어들의 벡터값이 크기 200으로 키워드 수 만큼 굉장히 많이 모여있는 것이 현재 모델의 상황인데, 이 벡터값들의 평균을 구해준다.

11 코사인 유사도 함수를 작성한다. 이는 카페와 카페의 유사도를 비교해 출력해준다.

11-1 코사인 유사도는 단지 카페와 카페 뿐만 비교할 수 있는것이 아니라, 애초에 카페의 벡터값이 카페의 키워드 단어들의 평균 벡터값이므로

카페와 키워드끼리의 비교도 가능하다. 이를 통해 유사도 비교를 진행한다.

12 유사도 비교하는 과정을 거쳐 검증을 끝냈다면 카페의 이름(1)과 카페의 테마 (9-2) 그리고 카페의 벡터값 (9)을 합쳐 csv 파일 형식으로 내보낸다.

3. 데이터베이스 만들기 단계

Mysql를 이용해 자연어 처리 단계를 거친 내용들 데이터베이스에 등재

Xampp를 이용해 Mysql과 Apache 연동

4. 웹페이지 만들기 단계

HTML 기반으로 CSS와 JavaScript를 이용해 제작

PHP를 이용해 데이터베이스와 연동

Pyscript를 이용해 자연어 처리와 웹페이지 연동

피드백 받아서 추가한 부분(임의)

요구 받은건 GPS랑 필터링 옵션(필수 추가)

캡스톤 최종 2차

검색어 바꿔서 다시 보여드리기

22일 검색어 바꿔서 발표 영상

25일까지 5분 이하 발표 영상

1. ~~크롤링 -> 자연어 처리 -> csv 출력 -> DB import 과정 자동화~~

실질적으로 구현 어려울 것 같다 -? 기존 작업이 간단해서 자동화하는게 더 복잡할 것으로 예상된다고

답변 쉽다고 하면 과정 자체를 보여드려야함

수집 과정, 명사

1. 사용자가 원하는 키워드 긍정 부정 분류 기준

단어 수식관계 쪽으로 접근 하면 안됨

키워드 그 자체가 이슈 리뷰는 문법에 맞게 쓰지 않는다

키워드 자체가 의미

컨셉이 간단 예시 위주로

부정적인 명사 사례 좋은 의미로도 나쁜 의미로도 사용되지만 그거 자체가 이슈 검색해서 나왔을 때

1. 명사만 추출하면 의미있는 분류 어려울 듯
2. ~~각 알고리즘 단계별 근거 자료 판단의 근거 뒷받침~~

그래프 가능하면 넣기

교집합에 들어가는 단어를 불용어 처리

* 남이 봤을 땐 의미 있어 보임
* 여러 카페에서 모두 나온 키워드라 불용어 처리했다고 표현

불용어 처리 기준 같은거 표현?

3개

~~발표할 때 이 부분만 집중적으로 보여드릴 예정~~

진행해야 될 부분

1. 명사의 긍정부정, 형용사, 동사의 추가 사용이 아닌 처음부터 명사만 사용하는게 목적이였고 왜 명사만 사용했는지, 해당 단어가 많이 언급된 것 자체가 그 카페를 설명할 수 있다. 이러한 내용의 좀 더 명쾌한 답변이 필요함.
2. DB에 키워드 넣어서 1대1 매칭, 이를 통해 시연 영상이 좀 더 그럴그럴 듯 나올 수 있게 만들어야 함. 맛이란 키워드로 검색한 이후 맛있는, 맛좋은, 군침이 도는 등으로 추가 검색해도 유사한 검색어가 나올 수 있게 확인시켜야 함.
3. 자동화를 안시켜도 괜찮은 이유에 대한 추가 설명, 어떤 식으로 진행되는지만 설명하고 해당 동작에 대한 결과물을 보여준게 적었던 것 같다
4. 가능하면 코랩에서 그래프 보여주고 이를 통해 설득력 높이기, 근데 어떤 내용의 그래프를 보여줘야 할지는 고민해봐야함.
5. HTML에 사용자 위치 받아오는 동작 영상에서 보여줘야함,

~~=============================================================~~

대본

교수님들의 심사 의견을 듣고 저희 팀은 저번 발표에 무엇을 사용했는지에 대한 부분에만 집중하고 이를 사용하면 무엇을 도출되고 확인할 수 있는 것이 무엇인지에 대한 내용을 확실하게 보여드리지 못했던 것 같아서 이에 대한 부분들을 집중적으로 보여드리려고 합니다. 또한 이 과정에서 말씀해주신 심사 의견에 답변드리도록 하겠습니다.

자동화가 필요 없는 이유

우선 저희의 과정을 간단하게 설명드리겠습니다. 이전에 설명드렸던 코드들을 그냥 단순 실행하면 크롤링이 진행되고 마찬가지로 자연어 처리 구간에서도 실행만 누르면 됩니다. 앞서 보여드린 동작들이 충분히 간단하다고 느꼈기 때문에 자동화를 하지 않는 방향으로 진행했습니다.

왜? -> 다음은 저희 팀 결과 자료 및 워드투벡터의 정확도의 이해를 돕기 위해 실제 카페들의 예시로 설명드리겠습니다.

다음은 저희 팀의 출력 결과를 구체적인 예시와 함께 설명드리려고 합니다. 현재 보이는 두 사진은 모두 일반적인 카페가 아닌 예전에 운영하다 중단된 폐공장을 리폼해 카페로 이용하고 있는 곳인데 특이한 컨셉으로 많은 인기를 끌고 있습니다. 왼쪽은 구글에서 캡쳐해온 별점과 구글에서 자체적으로 운영 중인 자주 나온 키워드들의 목록이고 오른쪽은 저희가 수집하고 가공한 결과입니다.

우선 조양방직의 검색 결과 볼거리, 공장, 소품, 감성 등의 키워드를 확인할 수 있습니다. 실제 저희 크롤링 결과에도 볼거리, 공장, 소품, 감성의 키워드를 포함해 사진, 분위기, 추억, 레트로 등의 키워드를 확인할 수 있습니다.

다음은 카페 어니언의 검색 결과입니다. 조양방직과 유사하게 인테리어, 공장, 분위기, 사진, 감성 등의 키워드가 도출되었습니다. 가장 눈에 띄는 차이점이라면 역시 가장 많이 출력된 키워드가 빵, 베이커리, 버터인데 이러한 키워드들은 카페 어니언에서 제과제빵 또한 취급한다는 점을 확인할 수 있습니다.

다음은 천안의 카페 이숲과 풍세커피입니다. 앞서 설명드린 카페와의 유사도 비교하는데 사용합니다.

이러한 부분에서 저희 팀은 명사만 사용해도 충분하다는 결론을 내렸습니다.저희 팀의 진행 방향은 처음부터 문장에서 명사만을 추출하고 이를 통해 내용을 분석하는 것을 목표로 잡고 진행하였습니다. 문장들을 말뭉치로 만들고 그 말뭉치에서 명사들만 분리해 키워드로 나온 이상 그 단어는 긍정과 부정을 떠나 카페와 관련된 이슈라고 파악할 수 있습니다. 앞뒤 내용을 확인해 긍정 부정을 확인하는 감성분석을 이용해 긍정적인 명사만 추려내는 것, 동사를 추가로 사용하는 방법 등도 고민했지만 긍정부정의 100% 정확한 감별도 어려울 뿐더러 어떤 방법이건 저희가 제공할 수 있는 자료의 판단은 오롯이 사용자의 몫이라고 느꼈습니다. 예를 들어 짜다 라는 키워드를 보고 누군가는 짠 음식을 선호하기 때문에 긍정적으로, 짠 음식을 싫어하기 때문에 부정적으로 받아들일 수 있습니다. 이처럼 저희는 키워드를 제시하고 이에 대한 판단에 대한 근거는 개인에 맡기고자 하였습니다. 키워드 자체의 자의적인 해석을 하는 것이 아닌 워드투벡터로 단어끼리의 연관성을 제공하는 것이 저희가 지향하던 방향입니다.

관련 움짤 or 영상들로 설명 추가