AI/data-Team

사용자 맞춤 카페 추천 시스템

Contents

Introduction

01

Motivation

다양한 카페 매장에서 공부 환경이 조성되고 있지만, 공부에 적합한 환경을 갖추었는지를 알기란 쉽지 않다. 따라서 현 위치에서 사용자 맞춤 스타일로 공부에 적합한 가까운 카페를 추천함으로써 학생들이 시간과 비용을 절약할 수 있도록 하는 시스템이 필요하다.

Objective

데이터베이스에 저장된 카페 이름, 평점, 평균가격, 공부환경점수, 카페 특색 및 리뷰 정보를 통해 사용자에게 적합한 카페를 추천 한다. 그리고 주기적으로 데이터베이스에 최신 정보를 추가로 저장하여 더 나은 추천을 할 수 있도록 한다.

System Flow

02

단순히 코사인 유사도 분류만 사용하지 않고, **감성분석을 사용해 걸러낸 긍정 리뷰로** 유사한 카페를 추천해 줌으로써 **모델의 정확도를 높였다.**

Web Scraping

카페의 정보 데이터 및
리뷰 데이터 수집

LSTM

시계열 데이터의 지도학습

RNN과 달리 긴 의존 기간을 필요로 하는 학습을 수행하는 능력을 가져, 리뷰의 감정분석에 사용될 수 있음

Sentiment Analysis

리뷰의 긍정 혹은 부정 도출

TF-IDF & Cosine-Similiarity

단어 임베딩 및 유사도 도출

- 의미가 있는 단어들로 특정 방향 벡터를 얻을 수 있음
- 카페에 대한 정보들이 얼마나 유사한지를 판별할 수 있음

Café Recommendation

리뷰 기반으로, 입력한 카페와
유사성이 높은 카페를 추천

Preprocessing1

리뷰데이터 전처리

***전처리**

라벨링, 라벨의 분포 균형화, 데이터 결측치 제거
중복샘플 제거, 불필요한 문자 제거
빈 데이터 제거, 리뷰 길이 통일(패딩60)
불용어 정의, 정수 인코딩, 희귀단어 빈도수 확인

Preprocessing2

리뷰에 대한 오버뷰와 컨셉 제작

***오버뷰**

긍정 리뷰 중 정확도 상위 5개 리뷰를 합친 데이터

***컨셉**

카페별 특색과 구조를 설명하는 데이터

Role designation

03

데이터 수집 파트

양지인천문학과, 조현진수학과, 문현정수학과
카페 정보 수집
리뷰 수집 및 라벨링

모델 학습 파트

정지호기계공학과, 장동훈소비자학과, 지수빈메카트로닉스공학과
카페 정보 수집
모델 학습 및 자료 정리

1주차	AI 1주차 스토리	22 days ago
2주차	AI 2주차 회의	22 days ago
3주차	3주차 전체회의 (백엔드 김재현님 기록)	22 days ago
4주차	Update 카공 4주차 진행방향.md	22 days ago
5주차	Item Based Collaborative Filtering	15 days ago
6주차	Delete 감성 분류(주석 추가).ipynb	7 days ago
7주차	데이터 스크래핑 통합	5 days ago

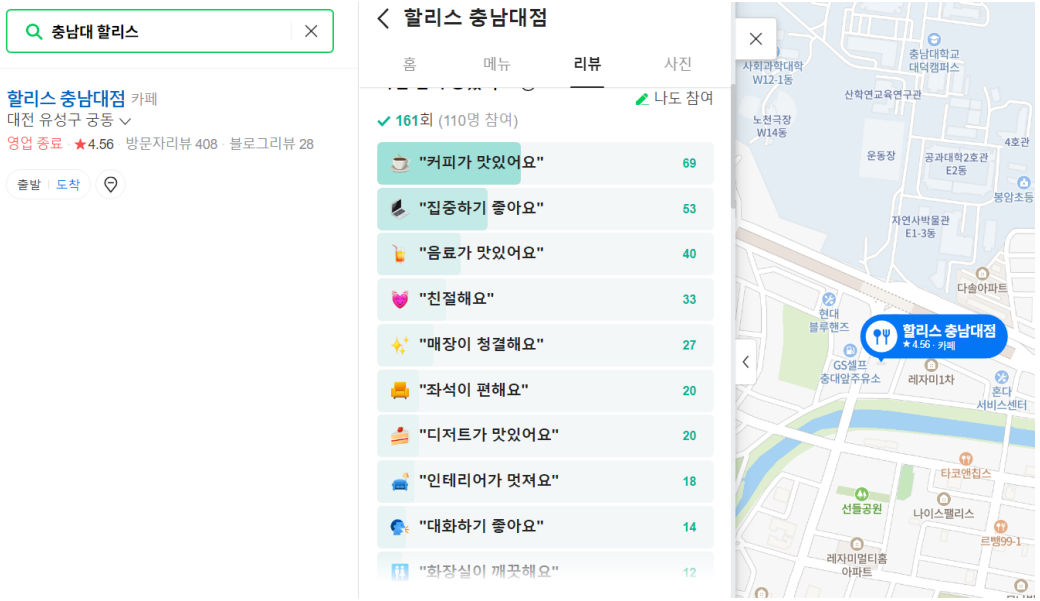
1주차 – 자료 조사
2주차 :: 개념설계
3주차 :: 백엔드+AI 전체 회의
4주차 :: 카페 데이터 수집
5주차 :: 시스템 구상 및 모델 선정
6주차 :: 전처리 및 모델 학습
7주차 :: 학습 및 자료 정리

Web Scraping

04

카페별 정보 데이터 수집

네이버 지도 리뷰 이용



위도 및 경도

평점 평균가격

카페명 주소

	카페	카페 도로명 주소	위도	경도	평균평점	평균가격	공부환경점수	매장이 청결해요	좌석이 편해요	집중하기 좋아요	카테고리 총 투표수
0	에이트(궁동)	34183 대전광역시 유성구 한밭대로 458 에이트	36.359845	127.349831	4.30	6600.000	0.015176	106	32	5	2438
1	커피인터뷰(궁동)	34136 대전광역시 유성구 한밭대로371번길 25-3 커피인터뷰	36.364536	127.341309	4.45	6176.000	0.026093	51	33	7	1533
2	소신(궁동)	34138 대전광역시 유성구 농대로15번길 7 소신	36.361609	127.352690	4.55	4471.000	0.007481	25	1	2	401
3	글로리데이즈(궁동)	34137 대전광역시 유성구 대학로 129 1층 글로리데이즈	36.361303	127.348133	4.59	5455.000	0.052721	39	16	15	588
4	마을(보드게임 만화 카페, 궁동)	34138 대전광역시 유성구 대학로151번길 38 마을	36.362349	127.350317	4.75	4521.000	0.236682	48	172	139	1314
...	...	...	...	...	...	...	...	...	...	...	...
80	스타벅스 대전구암DT점(구암동)	34181 대전광역시 유성구 월드컵대로289번길 6 스타벅스	36.350152	127.333093	4.50	5818.750	0.056907	193	65	52	2056
81	드롭탑 유성복합터미널DT점(구암동)	34176 대전광역시 유성구 유성대로680번길 1-11 드롭탑	36.354299	127.331011	4.72	4985.714	0.119926	57	40	25	542
82	투썸플레이스 대전(장대동)	34172 대전광역시 유성구 유성대로 756 1층 투썸플레이스	36.362675	127.351426	4.20	5770.513	0.118280	44	38	28	558
83	스타벅스 대전(장대동)	34165 대전광역시 유성구 유성대로 781 스타벅스	36.362675	127.351426	4.10	5818.750	0.064850	99	35	34	1064
84	뚜레쥬르 대전(장대동)	34171 대전광역시 유성구 문화원로 2 뚜레쥬르	36.361753	127.336818	4.30	4113.333	0.005277	35	1	1	379

청결+좌석+집중도 점수

Web Scraping

04

카페별 리뷰 데이터 수집

N 네이버 지도 리뷰 이용

카페별 리뷰 데이터

	id	review
1	미니미a	여행 일정들이 계속 꼬여 정말 지친 상황이었는데 마지막 코스를 스타벅스 송천점으로 결정한건 잘한 선택이었어요 파트너 성함은 모르겠지만 정말 친절하셨어요 스타벅스는 극히 일
2	Banking	언저 위에 토핑으로 올라가 있는 조콜릿부터 맛을 보았는데 특별할 거 없는 달달한 일반적인 초코 맛이었습니다 그 다음으로 위에 에스프레소 휘핑크림을 맛을 보았는데 초코 시럽이
3	나이삐	매장도 넓고 주차공간도 있어 좋아요! 당연히 커피도 맛있었어요
4	윤집사	매장도 넓고 직원분들도 친절합니다
5	온오프밀	새로나온 블랙글레이즈드라떼 고카페인이라 살짝 음청했지만 첫맛과 끝맛 달달구리하니 맛있어요~ 돌체블랙밀크티만 먹다가 한번 먹어보는데 괜찮네요~ 뭔한맛이있지만 한번쯤 맛
5	바별91	커피하면 스타벅인듯 하네요 오늘도 사람들이 많아요 굿
6	온오프밀	새로나온 블랙글레이즈드라떼 고카페인이라 살짝 음청했지만 첫맛과 끝맛 달달구리하니 맛있어요~ 돌 체블랙밀크티만 먹다가 한번 먹어보는데 괜찮...
7	Banking	커피가 빨리 나와 점심 시간에 가기 좋다
8	ypp****	오더받고 30분기다리는동안 순서대로 준비중이라고 하면서 한마디 없이 물량이없어서 준비가 안된다고하면 주문을 왜받는거고 확인도없이 기다리라고 하면 끝인 매장입니다 사과라
9	Banking	스벅에 들어 음료 한 잔 하기로 했다 스타벅스 제주청 글라임에이드의 맛이 궁금하기도 했고요?제주도에 서만 출시되는 건가 했는데 다행히 전주에서도 나왔네요 맛있습니다 **
10	hae****	생일쿠폰으로 출근길 음료마셨어용~ 친절하게 주문받아주시구 라떼는 넘 부드럽고 맛있어용~ 편리하게 픽업하니 넘 좋아용~
11	줄줄줄줄	여긴점장이시럽 소스 통형하는게 확실함
12	hhe****	스타벅스 커피 맛있어요????
13	작은천사33	역시 스벅은 번트앵..
14	피오나597	여유로운 시간 좋아요~~
15	리뷰리뷰뷰	스벅 케이크 맛있어요!!
16	홍분모드	오늘은 커피가 맛있네요. 충전할 콘센트가 없는 점은 좀 불편해요
17	연종32	종종 집근처라 방문하는데 직원분들이 항상 너무 친절해서 좋아요
18	그린조아97	리뷰 당첨 쿠폰이 있어 언제 갈까했는데 저녁 외식후 가까히 있어 시원하게 편히 매장에서 즐기고 왔어요~^^ 주말 가족단위로도 모임후로도 연인과 친구들과 아님 혼자서도 모두들
19	0000001033	커피마시기 편해요
20	나나노6277	위치가 좋아요
21	시리야75	별다방을 사랑하지않을수없는거다아시죠 가끔은사치로 즐길수있는 힐링장소
22	온오프밀	바쁜건 알겠는데 몇몇 직원분들은 손님한테 응대를 잘하는것보단 제품관리하느라 정신이없어보여 손님 인제가다 정신없을정도!! 방도그냥 집에와서 ...
23	이민호2342	맛있고 직원 분들 친절하시네요
24	KAKA2221	배송이 매우 빨라요
25	jjh****	아메리카노 좋아요
26	illust	블랙글레이즈드라떼 달고 맛있다

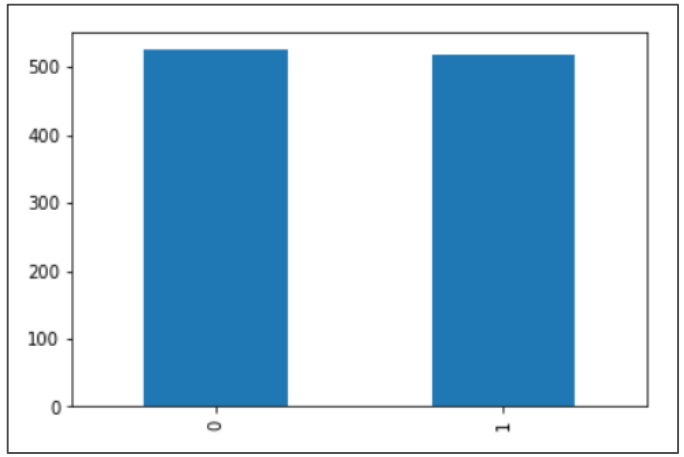
Preprocessing1

04

라벨링

	review	label
1121	왔다가 정말 스트레스만 적립하고 감.	0
811	주차 너무 편해요~	1
1172	층별로 분위기 다르고 공부하기도 좋은데 사람들이 많이 오면 와이파이가 너무 약해서 ...	0
534	진짜 맛있어요	1
514	분위기도 좋고, 사장님도 친절합니다.	1
...	...	...
1460	2층에 테이블간 간격이 넘좁아요	0
936	너무 시끄러워서 공부 집중이 안되네요	0
1338	이걸 보는 다른 손님들이 계시다면 차라리 편의점 커피를 드시길 추천합니다.	0
692	가게도 깨끗하고 분위기 좋아요	1
86	굿	1

라벨 분포 확인



불용어 정의

```
# 불용어 정의
stopwords = ['의','가','이','은','들','는','좀','잘','강','과','도','를','으로','자','에','와','한','하다', 'ㅠ', 'ㅠㅠ', 'ㅜㅜ', 'ㅋㅋ', 'ㅡㅡ', 'ㅠㅜ', 'ㄱ', 'ㄷㄷㄷ

X_train = []
for sentence in tqdm(train_data['review']):
    okt = Okt() # 형태소 분석기
    tokenized_sentence = okt.morphs(sentence, stem=True) # 토큰화
    stopwords_removed_sentence = [word for word in tokenized_sentence if not word in stopwords] # 불용어 제거
    X_train.append(stopwords_removed_sentence)

100%|██████████| 1037/1037 [00:07<00:00, 135.54it/s]
```

결측치 처리

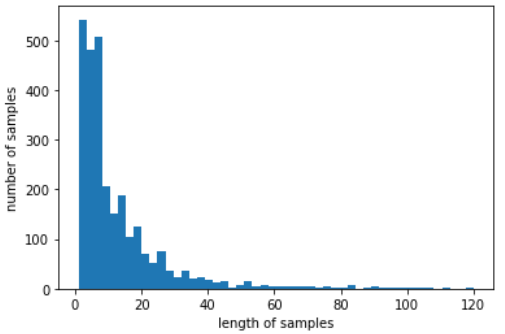
	review	label
27	NaN	1
92	NaN	1
313	NaN	1
186	NaN	1
161	NaN	1

정수 인코딩

{'좋다': 1, '너무': 2, '커피': 3, '없다': 4, '많다': 5, '있다': 6, '맛있다': 7, '공부
'조용하다': 15, '층': 16, '가격': 17, '사람': 18, '넓다': 19, '곳': 20, '음료': 21,
리': 29, '분': 30, '가다': 31, '되다': 32, '보다': 33, '비싸다': 34, '편하다': 35,
2, '에서': 43, '생각': 44, '공간': 45, '좌석': 46, '로': 47, '불편하다': 48, '것': 4
56, '그냥': 57, '깨끗하다': 58, '적': 59, '하고': 60, '받다': 61, '정말': 62, '고':
쁘다': 70, '주문': 71, '많이': 72, '쾌적하다': 73, '넘다': 74, '듣다': 75, '테이블':
3, '시간': 84, '앉다': 85, '느낌': 86, '여기': 87, '듯': 88, '인테리어': 89, '인': 9
들': 97, '괜찮다': 98, '아쉽다': 99, '다른': 100, '인지': 101, '안되다': 102, '엄청'
'디저트': 110, '뷰': 111, '에는': 112, '이라': 113, '추천': 114, '되어다': 115, '움'
화': 123, '이나': 124, '다양하다': 125, '좁다': 126, '거': 127, '응대': 128, '조금':
'저렴하다': 136, '하': 137, '가성': 138, '맡다': 139, '다시': 140, '같다': 141, '개'
8, '서': 149, '만들다': 150, '진짜': 151, '마다': 152, '사용': 153, '책': 154, '기':
1, '사진': 162, '아주': 163, '와이파이': 164, '주차장': 165, '그': 166, '주다': 167,
3, '케이크': 174, '알바': 175, '내': 176, '상태': 177, '때문': 178, '라떼': 179, '돼
6, '정도': 187, '특히': 188, '실내': 189, '전혀': 190, '새롭다': 191, '내부': 192,
다': 199, '걸': 200, '번': 201, '책상': 202, 'ㅎㅎ': 203, '제': 204, '청결하다': 205
'아이스': 212, '아메리카노': 213, '혼자': 214, '임': 215, '네': 216, '친구': 217, '나
'근처': 224, '시험': 225, '기간': 226, '가기': 227, '동네': 228, '정리': 229, '읽다'
'그리고': 237, '무더': 238, '지저분하다': 239, '힘들다': 240, '시원하다': 241, '따뜻
하다': 248, '이고': 249, '케익': 250, '말기': 251, '한잔': 252, '늦다': 253, '완전':
260, '이야기': 261, '불편': 262, '되게': 263, '빨리': 264, '적다': 265, '편안하다':
272, '크림': 273, '여유': 274, '수가': 275, '니': 276, '독서실': 277, '거의': 278,

리뷰 길이 확인

리뷰의 최대 길이 : 120
리뷰의 평균 길이 : 12.844309234073014



희귀 단어 처리

```
단어 집합(vocabulary)의 크기 : 4699
등장 빈도가 2번 이하인 희귀 단어의 수: 2891
단어 집합에서 희귀 단어의 비율: 61.523728452862315
전체 등장 빈도에서 희귀 단어 등장 빈도 비율: 9.017848088429165
```

=> 희귀 단어 등장 빈도 비율이 9% 밖에 안되므로 등장 빈도수가 2 이하인 단어들을 제거한다.

패딩

```
# 위의 분포 그래프를 봤을 때, max_len = 60이 적당할 것 같다.
max_len = 60
below_threshold_len(max_len, X_train)
```

전체 샘플 중 길이가 60 이하인 샘플의 비율: 97.74516821760916

Sentiment Analysis

04

LSTM을 이용한 감정분석 모델

```
from tensorflow.keras.layers import Embedding, Dense, LSTM
# Embedding: 벡터 공간에서 더 가까운 단어가 예상되도록 단어의 의미를 인코딩하는 실수 값 벡터의 형태로 텍스트 분석을 위한 단어 표현에 사용되는 용어
# Dense: 뉴런의 입력과 출력을 연결
# LSTM: RNN의 한 종류. 단순한 neural network layer 한 층 대신, 4개의 layer가  특별한 방식으로 서로 정보를 주고 받도록 되어 있다.
# LSTM -> 기울기 소멸 문제 방지

from tensorflow.keras.models import Sequential # 순차적으로 레이어 층 더해줌
from tensorflow.keras.models import load_model # 모델 불러오기 함수
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
# EarlyStopping: 과적합을 방지하기 위해 조기종료
# ModelCheckpoint: 학습 중인 모델 자동으로 저장하기

embedding_dim = 100 # 임베딩 벡터 차원은 100
hidden_units = 128 # 은닉 상태의 크기는 128

model = Sequential()
model.add(Embedding(vocab_size, embedding_dim))
model.add(LSTM(hidden_units))
model.add(Dense(1, activation='sigmoid')) # 활성화 함수로 시그모이드 함수 사용. 로지스틱 회귀를 사용해야 하므로
# 로지스틱 회귀분석에서는 종속변수가 0 또는 1이기 때문에  y=wx+b 을 이용해서 예측하는 것은 의미가 없다

es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=4) # 조기종료
mc = ModelCheckpoint('best_model.h5', monitor='val_acc', mode='max', verbose=1, save_best_only=True) # 자동 저장

model.compile(optimizer='rmsprop', loss='binary_crossentropy', metrics=['acc']) # 손실함수: binary_crossentropy. 이진 분류에 적합
history = model.fit(X_train, y_train, epochs=15, callbacks=[es, mc], batch_size=64, validation_split=0.2) # 검증데이터는 20% 사용
```

```
Epoch 9: val_acc improved from 0.87500 to 0.87981, saving model to best_model.h5
13/13 [==============================] - 1s 40ms/step - loss: 0.0620 - acc: 0.9807 - val_loss: 0.3566 - val_acc: 0.8798
Epoch 9: early stopping
```

```
loaded_model = load_model('best_model.h5') # 훈련 과정에서 검증 데이터의 정확도가 가장 높았을 때 저장된 모델인 'best_model.h5'를 로드
print("₩n 테스트 정확도: %.4f" % (loaded_model.evaluate(X_test, y_test)[1])) # 테스트 데이터에 대해서 정확도를 측정
```

```
11/11 [==============================] - 0s 7ms/step - loss: 0.3243 - acc: 0.8693

테스트 정확도: 0.8693
```

Train : Validation : Test = 7 : 1 : 2

Embedding dim=100

hidden unit size = 128

Activation = sigmoid & Loss = binary-crossentropy

Early stopping

train accuracy = 0.9807 & test accuracy = 0.8693

출력예시

```
sentiment_predict("가성비가 좋아요")
```

```
1/1 [==============================] - 0s 16ms/step
97.97% 확률로 긍정 리뷰입니다.
```

```
sentiment_predict('화장실이 청소가 잘 안되어있다')
```

```
1/1 [==============================] - 0s 14ms/step
99.86% 확률로 부정 리뷰입니다.
```

```
sentiment_predict('음료가 상큼하게 역겨워요... 뭔가 이상한 걸 넣었거나 뭔가 잘못 넣은듯 3모금 겨우 먹고 버렸어요..')
```

```
1/1 [==============================] - 0s 15ms/step
99.85% 확률로 부정 리뷰입니다.
```

Preprocessing2

04

Overview

긍정 리뷰의 정확도 상위 5개 결합

	cafe_name	overview
1	에이트	주차도 편하고 뷰가 엄청 좋아요 사진 찍기 좋은 장소들도 많고 테이블 간격도 넓어서 이야기하거나 공부하기도 좋은 것 같아요 재방문 의사 있습니당. 노트북 들고 와서 백색 소음 듣고
2	커피인터뷰	숲속에 온듯한 온실에서 커피 한잔 분위기 좋고 커피 맛있고 야외에서도 멋질 듯해요. 밤풍경도 궁금하네요 숲속에 온듯한 온실에서 커피 한잔 분위기 좋고 커피 맛있고 야외에서도 멋
3	소신	인테리어도 더 예뻐지고 분위기가 더좋아졌어요 시즌으로 나오는 무화과타르트는 감동입니다 일단 카페 분위기 너무 조용하고 좋아요 의자도 여러개고 인스타 갬성 이미 포장 다하셨
4	글로리데이즈	분위기도 좋고 탁 트인 공간이라서친구와 수다하기에도 좋고 넓은 책상에서 공부하기에도 좋아요 분위기 너무 좋고 야외 테이블이 있어서 요즘 같은 날씨에 딱이에요 콘센트 있는 자리
5	마을	여기 정말 자주오는데 올때마다 좋아요 공부하러 오기에도 보드게임 하러 오기에도 커피 마시러 오기에도 좋아요 조명도 많고 의자도 안락하고 사장님도 너무 친절하시고 음료도 너무
6	스타벅스 충남대정문점	공부하기 좋습니단 혼자 공부하러 가기엔 최고 하지만 번화가 위치해서 그런지 오가는 사람은 많아요 제가 다녀본 스벅 중엔 개인 자리가 제법 많은 편이라고 생각되어요 충남대 정문
7	시크릿	모임할 때 스터디룸을 사용합니다 이야기하기 좋은 프라이빗한 공간 같아요 혼자서 공부할 때도 스터디룸 사용해도 좋았습니다 사장님 너무 친절하시고 방도 따듯해요 음료수도 맛있
8	플라밍고 733	역시 여기 그냥 맛집으로 도장 찍었어요 꿀고구마 치케도 너무 맛있게 먹었어요 공부도 잘 되고 완전 강추 맛있어요 인기 많은 것 같아요 분위기 있어요 디저트 진짜 다양 ㅠㅠ이쁘고 노
9	투썸플레이스 충남대점	조용하게 공부나 과제하는 학생들이 많아요커피맛은 투썸 맛이겠죠 ㅎㅎ 투썸 오랜만에 갔는데 맛있네요 ㅎㅎ 대학가 근처라 그런지 과제나 공부하기 좋은 분위기 같습니다 매장이 넓
10	투썸플레이스 충남대점	조용하게 공부나 과제하는 학생들이 많아요커피맛은 투썸 맛이겠죠 ㅎㅎ 투썸 오랜만에 갔는데 맛있네요 ㅎㅎ 대학가 근처라 그런지 과제나 공부하기 좋은 분위기 같습니다 매장이 넓

+

Concept

카페를 설명하는 키워드 결합

없으면 공백으로 두었음

특색	구조
베이커리	넓은 홀(좌석이 30개 이상)
빵	지하
케이크	2층 이상
샌드위치	주차 가능
빙수	1인석
	스터디룸

=

Café review data for TF-IDF

카페 추천에 사용할 데이터

	cafe_name	overview	concept
0	데우스	분위기도 좋고 커피, 케이크도 맛있어요 또 오고 싶어용 디저트 맛있어요 커피는 아쉬...	빵 넓은 홀
1	스타벅스 합정점	직원분들이 정말 친절하고 좋아요 커피와 음료가 맛있고 직원분들이 친절하십니다 합정역...	샌드위치 넓은 홀 주차
2	보몽드	카페가 엄청 넓고 인테리어가 독특해요! 야외 정원도 너무 예뻐요. 음료도 맛있네용 ...	케이크 넓은 홀 주차

Café Recommendation

04

TF-IDF 및 Cosine Simuliarity를 이용한 카페추천

```
from sklearn.feature_extraction.text import TfidfVectorizer # Tf-idf vectorizer import
from sklearn.metrics.pairwise import cosine_similarity # cosine similarity import
```

```
tfidfvectorizer = TfidfVectorizer(analyzer='word', norm='l2') # 학습의 단위를 단어로 설정 + l2 정규화!
```

```
tfidf_matrix = tfidfvectorizer.fit_transform(CAFE_DATA["combined_features"])
# fit -> 데이터를 학습시키는 메서드
# transform -> 학습시킨 것을 적용하는 메서드
# fit_transform을 사용하는 것과 fit을 하고 transform을 하는거는 아주 약간의 미세한 성능 차이는 있지만 더 효율적임.
# transform 까지 거쳐 행렬이 생성됨.
```

```
tfidf_matrix.shape
```

```
(82, 1798)
```

```
cosine_sim = cosine_similarity(tfidf_matrix) # 생성된 행렬의 코사인 유사도를 계산함.
cosine_sim # array
```

```
def reco_top_similar_cafes(cafe_name, n=82):
  cafe_index = get_index_from_name(CAFE_DATA, cafe_name) # 위에서 만든 get_index_from_name 함수를 따로 사용
  similar_cafes =  enumerate(cosine_sim[cafe_index]) # cafe_index에 해당하는 행의 데이터를 읽어옴 (인덱스, 유사도값)
  sorted_similar_cafes = sorted(similar_cafes, key=lambda x:x[1], reverse=True) # 유사도값을 내림차순으로 (x[1]만 쓰면 오름차순이지만,

  ret_cafes = []
  i = 0
  for element in sorted_similar_cafes: # 유사도 값이 내림차순으로 정렬된 리스트를 맨 위부터 하나씩 찾는다
    name = get_name_from_index(CAFE_DATA, element[0])
    ret_cafes.append(name) # 카페 이름이 append됨.
    i=i+1
    if i >= n:
      break # 요청된 숫자만큼 찾는다. 여기서는 82번.
  return ret_cafes # 카페 이름들이 append 된 채 리턴이 된다. 밑에 예시를 참고하자.
```

	0	1	2	3	4	5	6	7
0	1.000000	0.025088	0.079185	0.034900	0.095500	0.022320	0.058767	0.040208
1	0.025088	1.000000	0.068597	0.091860	0.075194	0.000000	0.049028	0.023592
2	0.079185	0.068597	1.000000	0.077068	0.101388	0.001435	0.060560	0.082682
3	0.034900	0.091860	0.077068	1.000000	0.085485	0.004203	0.048656	0.044920
4	0.095500	0.075194	0.101388	0.085485	1.000000	0.036631	0.165554	0.025245
...	...	...	...	...	...	...	...	...
77	0.103573	0.039524	0.057287	0.048944	0.057628	0.063675	0.043567	0.045689
78	0.092489	0.058692	0.032825	0.031310	0.073727	0.028974	0.041773	0.030795
79	0.037580	0.009105	0.034124	0.070800	0.024427	0.039238	0.037738	0.009398
80	0.024939	0.011261	0.008639	0.038530	0.011376	0.050089	0.012797	0.052028
81	0.057425	0.023120	0.018736	0.052360	0.045445	0.013953	0.046014	0.070576

82 rows × 82 columns

normalization = 'l2'

analyzer = 'word'

TF-IDF로 벡터화

Cosine-Simuliarity로 유사도 검출

유사성이 높은 카페 추천

출력예시

```
print(reco_top_similar_cafes('스타벅스 대전도안DT점', 5))
```

```
['스타벅스 대전도안DT점', '에이트', '탐앤탐스 대전유성점', '스타벅스 대전구암DT점', '스타벅스 대전유성점']
```

```
print(reco_top_similar_cafes('에이트', 5))
```

```
['에이트', '스타벅스 대전도안DT점', '탐앤탐스 대전유성점', '파스쿠찌 유성온천역점', '할리스 유성온천점']
```

```
print(reco_top_similar_cafes('마을', 5))
```

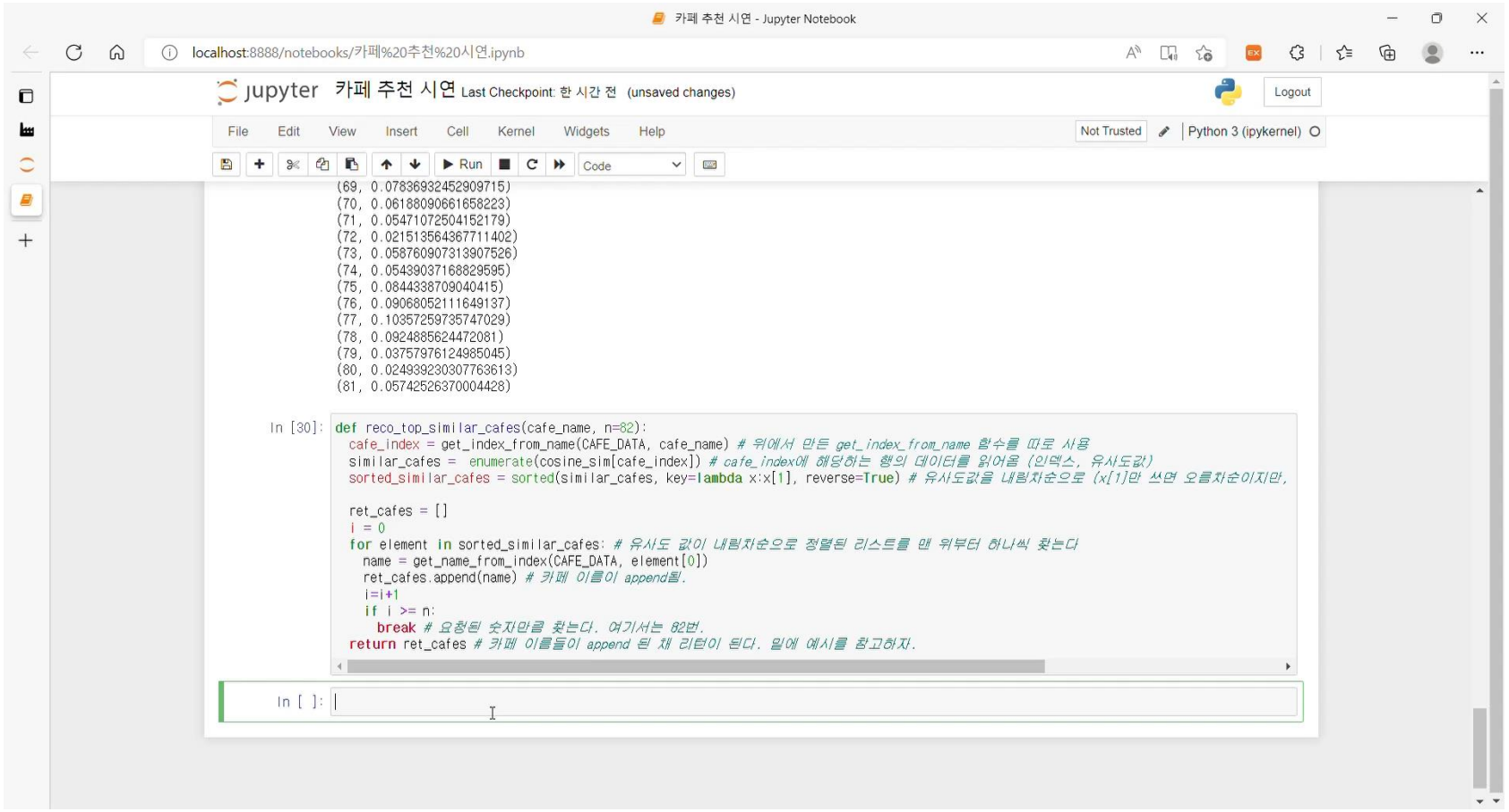
```
['마을', '시크릿', '이디야 충남대점', '소신', '할리스 유성온천점']
```

```
print(reco_top_similar_cafes('이디야 충남대점', 5))
```

```
['이디야 충남대점', '공차 충남대점', '카페소담', '투썸플레이스 충남대점', '이디야 대전유성온천역점']
```

Simulation

05



유사한 카페 추천

- 카페 이름이 input
- 추천 개수는 조정 가능