

전국 맛집 분포도 & 맛집 여행 트렌드

3조 사자들

SungJun Park JiEun Lee SangYeop Park

1. 프로젝트 배경

문제 발견

2. 프로젝트 과정

데이터 수집 및 분석

3. 프로젝트 결과

4. 활용방안

5.배운점 & 개선방안



문제 발견 : 수많은 맛집 정보 데이터! 어떻게 활용할 수 있을까?

사장님의 고민

“고객은 과연 어떤 내용을 보고
식당을 평가할까?”

“음식 종류별로 고객은 어떤 키워드에
중점을 두고
평가를 했을까?”

“우리집 식당 홍보 어떻게 해야할까?”

고객의 고민

‘맛있는 중식먹으러 맛집 투어 가고
싶은데...어디로 가야하나? 일일이
검색해서 지도에 표시할 수도 없고..’

여행 갈때마다 먹는 휴게소의
우동이 이제는 질린다! 지역
맛집으로!

‘손님많은 식당 말고, 지역에 리얼
맛집을 가고 싶은데..’



전국의 맛집들이 골고루 홍보될 수 있는 통로가 되었으면 좋겠다!

Framework

맛집 포털사이트 선정 ' 망고플레이트



리뷰 키워드 분석



‘한식’, ‘중식’, ‘일식’, ‘양식’ 키워드로
포털사이트에 검색을 하여, 음식 종류별
리뷰 내용 크롤링



해당 데이터로 wordcloud를 만들어서
음식 종류별 중심 키워드 확인

맛집 분포도 시각화



‘음식점 상호명’, ‘주소’, ‘지역명’을
크롤링



카카오 API를 활용하여 ‘주소’ 컬럼의
내용을 바탕으로 경도와 위도를 받아옴.



Folium을 통한 지역별 맛집 데이터 시각화

시각화 자료1

World Cloud 사용

- ✓ 키워드 분석 시각화
- ✓ 가장 많이 사용되는 용어 선별



시각화 자료2

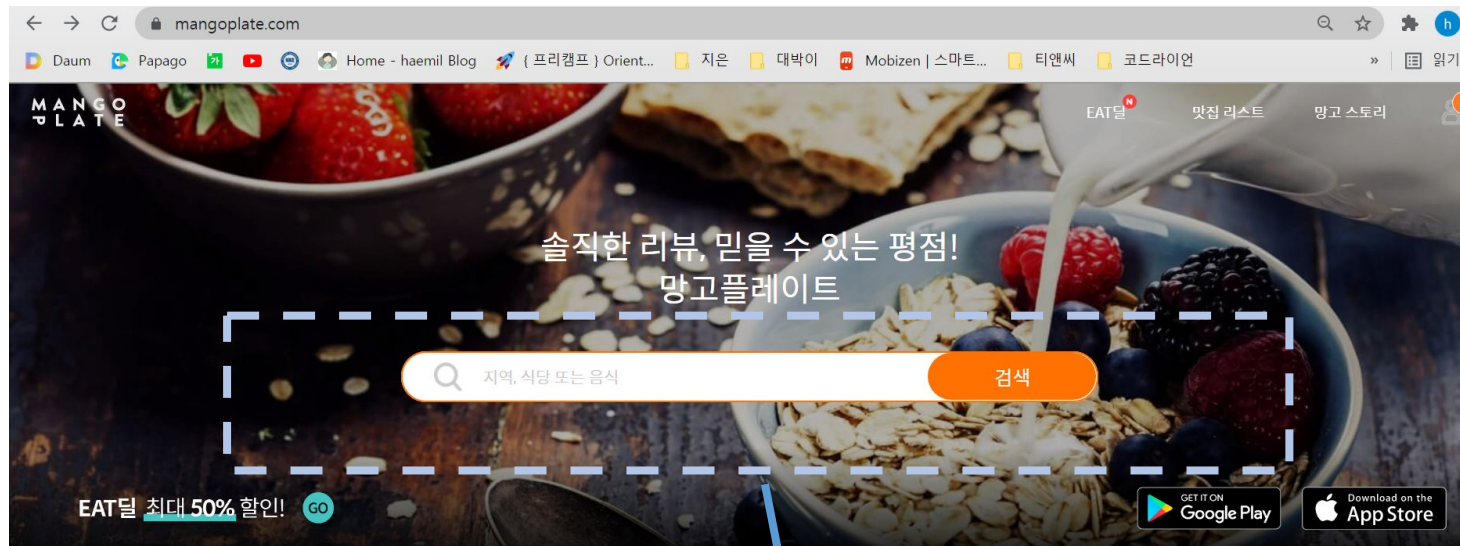
Folium Map 사용

- ✓ 맛집 위치 시각화
- ✓ 나를 위한 맛집 분포도 확인



프로젝트 과정 | 리뷰 키워드 분석

크롤링 - 키워드 자동 검색



홈 화면의 검색창에 '한식' 키워드 검색

```
In [4]: ▶ input_b = driver.find_element_by_xpath('//*[@id="main-search"]') #검색창
input_b.send_keys("한식") # 검색어
input_c=driver.find_element_by_xpath('/html/body/main/article/header/div/fieldset/input') #검색 버튼
input_c.click()
```


1~5페이지 크롤링


MANGO
PLATE

지역, 식당 또는 음식


E/

한식 맛집 인기 검색순위

필터



호반 4.8
삼청/인사 - 한정식 / 백반 / 정통 한식
123,262 52



맛짱조개 4.8
논현동 - 해산물 요리
59,256 38

...

1

2

3

4

5

6

7

8

9

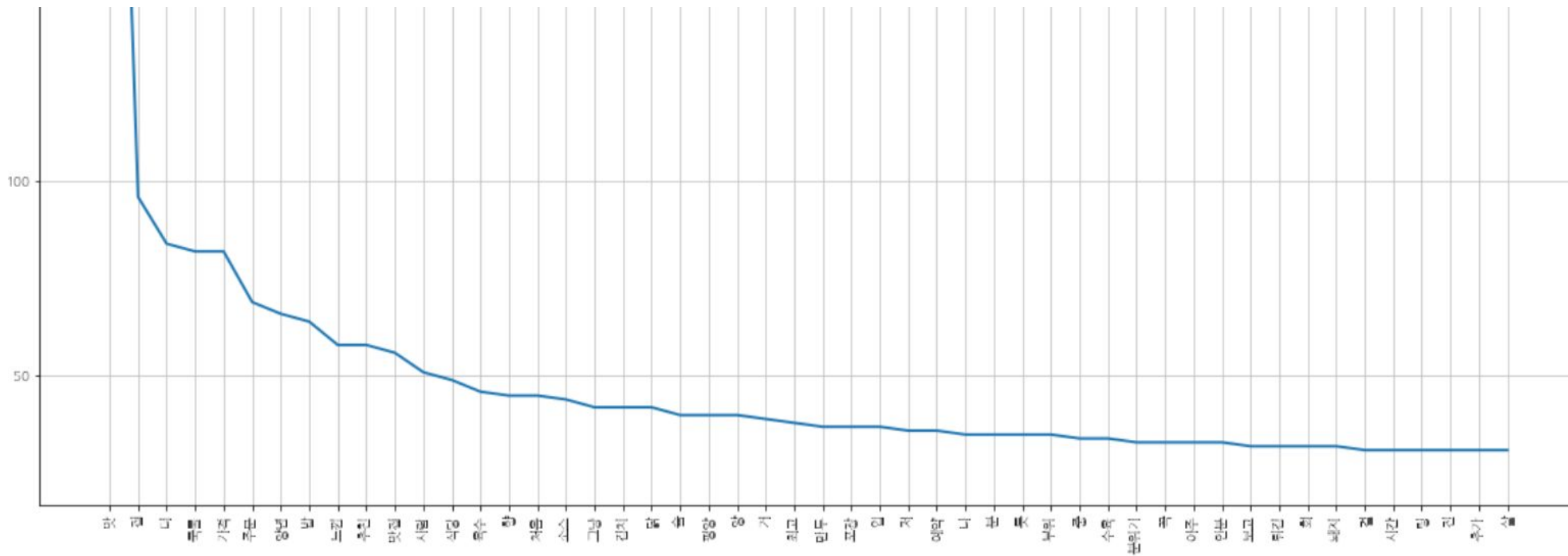
10

크롤링 방법:
한 페이지 20개 리스트 X
10 페이지

프로젝트 과정 | 리뷰 키워드 분석

Plot

```
[ ] new_ro2 = nltk.Text(new_ko, name="한식리뷰")  
plt.figure(figsize=(20,16))  
new_ro2.plot(50)
```



프로젝트 과정 | 맛집 분포도 시각화

```
def getLatLng(address):  
    app_key = 'KakaoAK ' + '[REDACTED]388b'  
    url = 'https://dapi.kakao.com/v2/local/search/address.json' # endpoint  
    params = {  
        'query' : address,  
        'page' : '',  
        'AddressSize' : ''  
    }  
    headers = {'Authorization' : app_key}  
    response = requests.get(url, headers=headers, params=params)  
    status_code = response.status_code  
    resp = response.json()  
  
    if resp['documents'] != []:  
        longitude = resp['documents'][0]['x'] # 경도  
        latitude = resp['documents'][0]['y'] # 위도  
        return longitude, latitude  
    else:  
        return (0, 0)
```

```
import pandas as pd  
train = pd.read_excel('./Hansik.xlsx')  
adr=train.iloc[:,2]  
adr[0]
```

'서울시 종로구 낙원동 85-7'

KaKao API이용 경도&위도 과정

- ✓ 위도&경도 함수 호출
- ✓ 카카오 api 가입후 App_key 받아오기
- ✓ 보유 맛집 데이터 호출
- ✓ 맛집 데이터 주소 불러오기!
- ✓ 카카오 함수에 주소 보내서 경도&위도 받기

프로젝트 과정 | 맛집 분포도 시각화

```
import pandas as pd
all_e=['Hansik.xlsx','IL.xlsx','jung.xlsx','yang.xlsx']
han_f=['불고기','게장','육회']
IL_f=['연어','초밥','우동']
jung_f=['탕수육','복음밥','양고기']
yang_f=['토마토','치즈','피자']
all_f=[han_f,IL_f,jung_f,yang_f]
fav=['han_f','IL_f','jung_f','yang_f']
adr_all=[] # 도시별 주소
for g in range(4):
    ex_fil=pd.read_excel(all_e[g])
    menu=ex_fil.iloc[:,11]
    i=0
    nam=[]
    adr=[]
    gyu=[]
    wedo=[]
    men=[]
    for s in menu:
        for mf in all_f[g]:
            if mf in s:
                nam.append(ex_fil.loc[i,'name'])
                adr.append(ex_fil.loc[i,'adr'])
                gyu.append(ex_fil.loc[i,'gyung'])
                wedo.append(ex_fil.loc[i,'wedo'])
                men.append(s)
            i+=1
    dat_r = {'name': nam, 'adr':adr, 'gyung':gyu, 'wedo':wedo, 'menu':men}
    dat = pd.DataFrame(dat_r)
    dat.to_excel((fav[g]+".xlsx"), encoding="utf-8", index = False)
    adr_r=[]
    for a in adr:
        s_p=a.split()
```

Folium Map 과정

- ✓ 맛집 위치 & 메뉴 호출
- ✓ 내가 좋아하는 음식 취향 작성
- ✓ 보유 맛집 메뉴와 취향 비교
- ✓ 취향에 부합하는 식당만 따로 저장!

프로젝트 과정 | 맛집 분포도 시각화

```
[4] fig=plt.figure(figsize=(15,5))
sns.set_style('whitegrid')

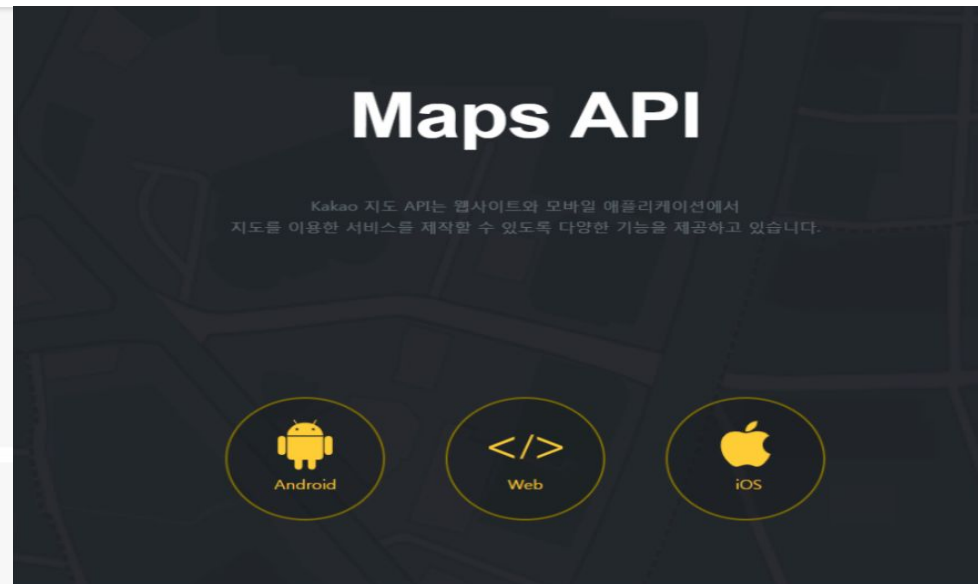
han1=fig.add_subplot(2,2,1)
han1.set_title('한식 도시')
han2=fig.add_subplot(2,2,2)
han2.set_title('중식 도시')
han3=fig.add_subplot(2,2,3)
han3.set_title('양식 도시')
han4=fig.add_subplot(2,2,4)
han4.set_title('일식 도시')

#count plot은 y축 없다 숫자확인하므로
sns.countplot(x=adr_all[0], ax=han1)
sns.countplot(x=adr_all[2], ax=han2)
sns.countplot(x=adr_all[3], ax=han3)
sns.countplot(x=adr_all[1], ax=han4)
```

```
import folium
from folium import Marker, Icon
import pandas as pd
import csv, time

han_u= pd.read_excel('./han_f.xlsx')
jung_u= pd.read_excel('./jung_f.xlsx')
yang_u= pd.read_excel('./yang_f.xlsx')
IL_u= pd.read_excel('./IL_f.xlsx')
yang_u.head(3)
```

	name	adr	gyung	wedo	menu
0	작은피자집	서울시 동작구 사당동 1029-37	126.980681	37.480528	:더블치즈트리플치즈하와이안페페로니반&반
1	매덕스피자	서울시 강남구 신사동 541-11 1F	127.023762	37.519077	:마르게리타피자치즈피자페퍼로니피자하와이안피자
2	엘리스리틀이태리	서울시 송파구 송파동 8 SANDONG빌딩	127.104847	37.508645	:카프레제봉골레푸타네스카새우버섯과루꼴라피자안심스테이크



Folium Map 작성을 위해서 엑셀 파일 호출
한식, 중식, 일식, 양식 맛집의 경도 위도 사용

프로젝트 과정 | 맛집 분포도 시각화

```
df_sample = IL_u[['name', 'gyung', 'wedo']]
# display(df_sample.head(3))
print('데이터 널값 확인')
display(df_sample.isnull().sum())
```

```
데이터 널값 확인
name      0
gyung     0
wedo      0
dtype: int64
```

```
loc=[33.37860539157057, 126.52645770837371]
map = folium.Map(loc, tiles='Stamen Terrain', zoom_start=11)
for i in range(len(han_u)):
    folium.Marker(location=[han_u.loc[i]['wedo'], han_u.loc[i]['gyung']],
                  tooltip=han_u.loc[i]['name'],
                  icon = folium.Icon(color='red', icon="info-sign")
                  ).add_to(map)
for i in range(len(jung_u)):
    folium.Marker(location=[jung_u.loc[i]['wedo'], jung_u.loc[i]['gyung']],
                  tooltip=jung_u.loc[i]['name'],
                  icon = folium.Icon(color='purple', icon="info-sign")
                  ).add_to(map)
for i in range(len(yang_u)):
    folium.Marker(location=[yang_u.loc[i]['wedo'], yang_u.loc[i]['gyung']],
                  tooltip=yang_u.loc[i]['name'],
                  icon = folium.Icon(color='blue', icon="info-sign")
                  ).add_to(map)
for i in range(len(IL_u)):
    folium.Marker(location=[IL_u.loc[i]['wedo'], IL_u.loc[i]['gyung']],
                  tooltip=IL_u.loc[i]['name'],
                  icon = folium.Icon(color='green', icon="info-sign")
                  ).add_to(map)
map.save('jeju.html')
map
```

Folium Map 작성을 위한 Null값 확인

한식, 중식, 일식, 양식 맛집 Icon_Mark 색으로 종류 나누기

프로젝트 결과 | 리뷰 키워드 분석

음식종류별 리뷰 키워드

한식



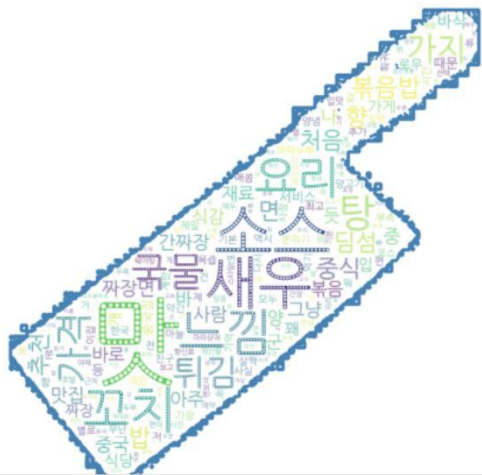
국물, 맛, 소스, 밥,
양념, 사람, 가격, 양, 향

일식



소스, 튀김, 맛, 스시,
밥, 다른, 느낌, 가격,
라멘, 마카, 우동, 향, 참치

중식



소스, 새우, 국물, 맛,
볶음, 꼬치, 탕, 튀김

양식



맛, 피자, 소스, 파스타,
치즈, 느낌, 버거, 향,
샐러드

프로젝트 결과 | 맛집 분포도 시각화

PSJ취향

- *한식: 불고기, 계장, 육회
- *중식: 탕수육, 볶음밥, 양고기
- *일식: 연어, 초밥, 우동
- *양식: 토마토, 치즈, 피자



한식



일식



중식



양식



프로젝트 결과 | 맛집 분포도 시각화

- ✓ 서울에 다수의 상위권 맛집 분포
- ✓ 전통 한식 맛집도 대부분 서울 분포



한식



일식



중식



양식



식당 홍보시 고객을 사로잡는 키워드를 선택



한국인의 입맛을 사로잡은 진한 **국물** 맛

30년 전통의 **양념**

가격도 착함

0000 **추천**

우리집만의 **소스**, 손님들이 반한 천연 **소스** 맛

엄마가 지어주신 **밥** 맛

맛으로 승부하는 집

활용 방안 | 리뷰 키워드 분석

오늘의 기분에 따라 추천해 주는 맛집

주제어 별로 맛집을 선택할 수 있는 방법에 쓰일 수 있음

고객의 리뷰를 분석하여, 분위기, 맛, 가격에 해당하는 키워드에 따른 맛집 추천이 가능함

EX) 분위기: #분위기 #럭셔리 #조명 #디자인 #매너 #친절함 등



식사류		10,000
제육볶음	(국내산)	8,000
김치찌개		6,000
돌솥비빔밥		7,000
비빔국수		5,000
냉면		5,000
볶음밥		2,000
공기밥		1,000

취향을 반영한 맛집 선정

좋은점:

- 1) 이미 선정된 맛집 리뷰에 취향 반영을 통한 실패할 확률 감소
- 2) 선택 사항 감소를 통한 선택장애 해결

개선점:

- 1) 맛집 분포가 너무 서울에 과하게 분포
- 2) 현재 상위 200위보다 더 많은 400위 까지 확보 필요!



배운점 & 개선 방안

1. 프로젝트 주제선정 & 목표

다양한 데이터 주제 & 문제 발견

2. 과정단위 기록!

데이터 수집 과정에서 배운점을 항상 기록!

3. 크롤링 과부하

확보한 자료& 데이터 항상 업로드!

4. 지역별 음식점 추가

음식점 메뉴&주소 추가

5. GUI 활용방안

선택 음식종류 한정

6. 음식 사이트 Hidden Html

숨겨진 Tag로 인한 크롤링 오류

