SNS 기반 제주 아라동 맛집 데이터 분석

전산통계학과 윤소원

목차

주제 선정 이유

끝 없는 뭐 먹을래?의 굴레에 갇히다.

블로그 리뷰에서 공통적으로 언급되는 키워드

과정 1 – SNS (블로그) 크롤링

키워드 검색 후
상위 검색 결과포스팅 선택

제목, 내용, 장소 등 크롤링

데이터 정제 (텍스트 정제)

“sns_posts.csv” 생성

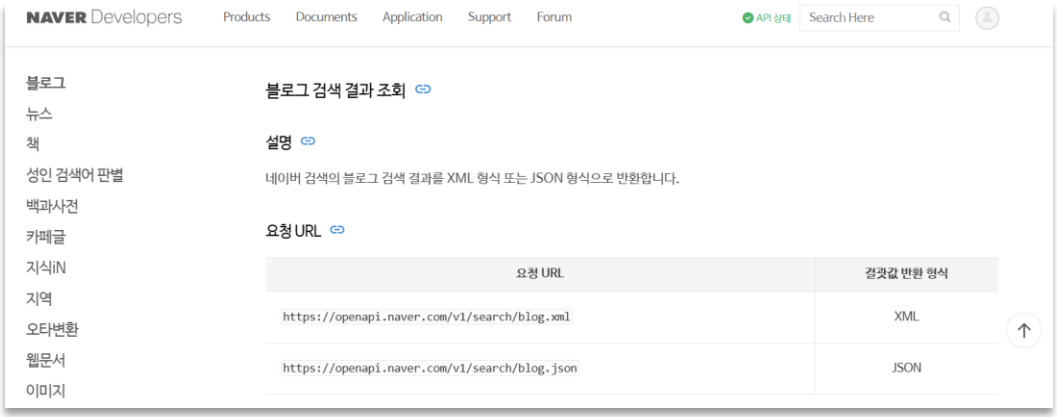
과정 1 – SNS (블로그) 크롤링

키워드 검색 후
상위 검색 결과포스팅 선택

제목, 내용, 장소 등 크롤링

데이터 정제 (텍스트 정제)

“sns_posts.csv” 생성



네이버 블로그 검색 API 이용 > 제목, URL 수집

```python
response_body = response.read()
json_response = response_body.decode('utf-8')
json_response=json.loads(json_response)
# print(json_response['items'][0]['title'])

titles=[]
links=[]

for i in range(display):
    titles.append(remove_html_tags(json_response['items'][i]['title']).strip())
    links.append(json_response['items'][i]['link'])
```

과정 1 – SNS (블로그) 크롤링

키워드 검색 후
상위 검색 결과포스팅 선택

제목, 내용, 장소 등 크롤링

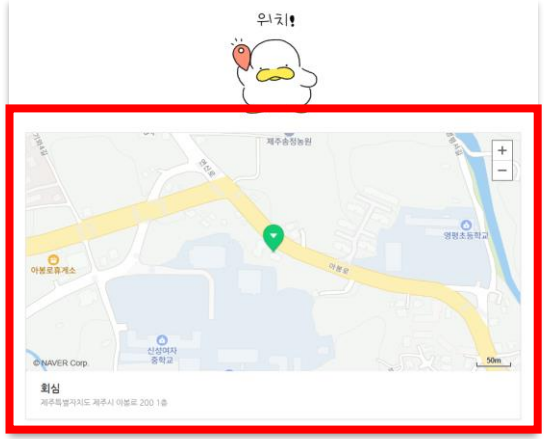
데이터 정제 (텍스트 정제)

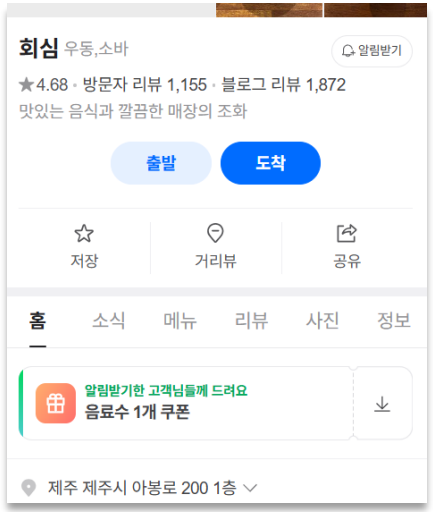
“sns_posts.csv” 생성

```
for i in range(display):
    url=links[i]
    driver = webdriver.Chrome()
    driver.get(url)
    time.sleep(1)

    driver._switch_to.frame('mainFrame')
    content=driver.find_element(By.CSS_SELECTOR,".se-main-container")
    # print(content.text)

    contents.append(remove_html_tags(content.text).strip())

    #해시태그 수집...
    hashtag=[]
    tag=driver.find_elements(By.CLASS_NAME, "wrap_tag")
    for i in tag:
        hashtag=i.text.strip().split("\n")
    txthashtag="".join(hashtag[1:])
    hashtags.append(txthashtag)
```

수집 데이터 : 제목, URL, 내용, 해시태그, 장소 이름, 주소

과정 1 – SNS (블로그) 크롤링





장소를 표기하는 방법이 다르다.

두 가지 모두 표기하는 경우 / 하나만 표기한 경우

```
#장소 수집
try:
    map_pic=driver.find_element(By.CSS_SELECTOR,"a.se-map-info")
    map_json = map_pic.get_attribute("data-linkdata")
    map_json = map_json.replace("&quot;", '"')
    map_json=json.loads(map_json)
    print(map_json.get('placeId'))
    href_value="https://map.naver.com/v5/entry/place/"+map_json.get('placeId')
    driver.get(href_value)
    time.sleep(2)
```

```
except Exception :
    try:
        link_element = driver.find_element(By.CSS_SELECTOR, ".location a")
        href_value = link_element.get_attribute("href")
        print(href_value)
        driver.get(href_value)
        time.sleep(2)

        driver._switch_to.frame('entryIframe')
```

과정 1 – SNS (블로그) 크롤링

키워드 검색 후
상위 검색 결과포스팅 선택

제목, 내용, 장소 등 크롤링

데이터 정제 (텍스트 정제)

“sns_posts.csv” 생성

```
#내용 정제...
df=pd.DataFrame({'titles':titles,'links':links,'location':location,'address':address,'hashtag':hashtags,'contents'
df['titles']=df['titles'].str.replace('[^가-힣A-Za-z0-9#\s]', '', regex=True)
df['contents']= df['contents'].str.replace('\n', ' ', regex=True)
df['contents']= df['contents'].str.replace('[^가-힣A-Za-z0-9#\s]', '', regex=True)
```

```
df.to_csv("sns_posts.csv",encoding="utf-8-sig",index=False)
```

수집된 데이터 정제

- HTML 태그 삭제
- 특수 문자 및 이모지 삭제
- 개행 문자 삭제

과정 2 – 키워드 분석

포스팅 내용 정제
(제외 단어 설정)

품사별 키워드 분석

데이터 시각화
(bar Chart, WordCloud)

과정 2 – 키워드 분석

포스팅 내용 정제
(제외 단어 설정)

품사별 키워드 분석

데이터 시각화
(bar Chart, WordCloud)

형태소 분석 > KoNLPy의 Okt 사용.

특징을 잘 나타내는 품사 선정 : 명사, 형용사.

일부 제외 단어 선정 후 제거.

```
for i in contest_list:
    okt_list=okt.pos(i,norm=True,stem=True)
    kk_noun.extend(i for i, j in okt_list if j in ['Noun'] and i not in['수', '곳','것','제주','아라동','제주시','맛집'])
    kk_abjective.extend(i for i, j in okt_list if j in ['Adjective']and i not in['이다','스럽다'])
```

```
Counter({'이': 16, '하나': 11, '코스': 9, '제수': 7, '에': 7, '너무': 7, '가': 7, '만배': 6, '한성식': 6, '로': 6, '으로': 6, '과': 6
, '은': 6, '는': 6, '성': 5, '도': 5, '층': 5, '2': 5, '나오다': 5, '좋다': 4, '음식': 4, '공간': 4, '되다': 4, '되어다': 4, '의': 4,
 'C': 4, '생선회': 4, '메뉴': 4, '맛집': 3, '에서': 3, '맛': 3, '맘': 3, '들다': 3, '보이다': 3, '있다': 3, '인': 3, '고급': 3, '구성
': 3, '갈비찜': 3, '옥돔': 3, '구이': 3, '등': 3, '따뜻하다': 3, '까지': 3, '한식': 2, '아라동': 2, '제주시': 2, '아연': 2, '4849': 2
, '소개': 2, '남편': 2, '날': 2, '아주': 2, '집': 2, '인데': 2, '요': 2, '정갈하다': 2, '프라이': 2, '빗': 2, '장소': 2, '#제주한식맛
집': 2, '#만배성한정식': 2, '시간': 2, '30분': 2, '00분': 2, '휴무': 2, '1': 2, '수': 2, '개인룸': 2, '꾸미다': 2, '진': 2, '만': 2,
'배성은': 2, '선택': 2, '39000원': 2, '갈치': 2, '조림': 2, '이상': 2, '불고기': 2, '새우': 2, '튀김': 2, '추가': 2, '를': 2, '특히':
 2, '부드럽다': 2, '이다': 2, '을': 2, '예약': 2, '준비': 2, '부터': 2, '느껴지다': 2, '시작': 2, '서빙': 2, '계속': 2, '전복죽': 2,
'잡채': 2, '그리고': 2, '바다': 2, '초밥': 2, '육회': 2, '먹다': 2, '와': 2, '로는': 2, '회': 2, '그때': 2, '맛있다': 2, '가마솥': 2,
 '주소': 1, '지난주': 1, '친구': 1, '들': 1, '모임': 1, '받다': 1, '음식점': 1, '특별하다': 1, '다녀오다': 1, '오늘': 1, '분위기': 1,
 '보다': 1, '한': 1, '상견례': 1, '고급스럽다': 1, '괜찮다': 1, '영업': 1, '11시': 1, '21시': 1, '브레이크': 1, '타임': 1, '15시': 1,
 '17시': 1, '정해진': 1, '없다': 1, '전화': 1, '확인': 1, '부탁': 1, '전화번호': 1, '0647126800': 1, '주차': 1, '시설': 1, '대형': 1,
```

과정 2 – 키워드 분석

포스팅 내용 정제
(제외 단어 설정)

품사별 키워드 분석

데이터 시각화
(bar Chart, WordCloud)

```python
#막대 그래프
fpath = "C:/Windows/Fonts/MALGUNSL.ttf"
font_name = fm.FontProperties(fname=fpath).get_name()
plt.rc('font', family=font_name)
plt.figure(figsize=(12, 7))
plt.xlabel("명사")
plt.ylabel("빈도")
xn=[]
yn=[]

for i,j in Counter(kk_noun).most_common(20):
    xn.append(i)
    yn.append(j)
plt.bar(xn,yn)
plt.xticks(rotation=45)
for i, v in enumerate(yn):
    plt.text(i, v, str(v),ha='center',va='bottom')
plt.savefig("pyplot_noun.png")
plt.show()
```

빈도 분석의 결과를 시각화.

pyplot 막대 그래프

wordcloud

```python
wc = WordCloud(font_path=fpath,background_color='white', width=800, height=400,colormap='seismic_r',
               max_words=100).generate_from_frequencies(Counter(kk_noun))
plt.imshow(wc, interpolation='bilinear')
plt.axis("off")
plt.savefig("WordCloud_noun.png")
plt.show()
```

과정 3 – 지도 시각화

가게 이름으로 지도 정보 조회

위도 및 경도 데이터 수집
음식점이 아닌 업종 제외

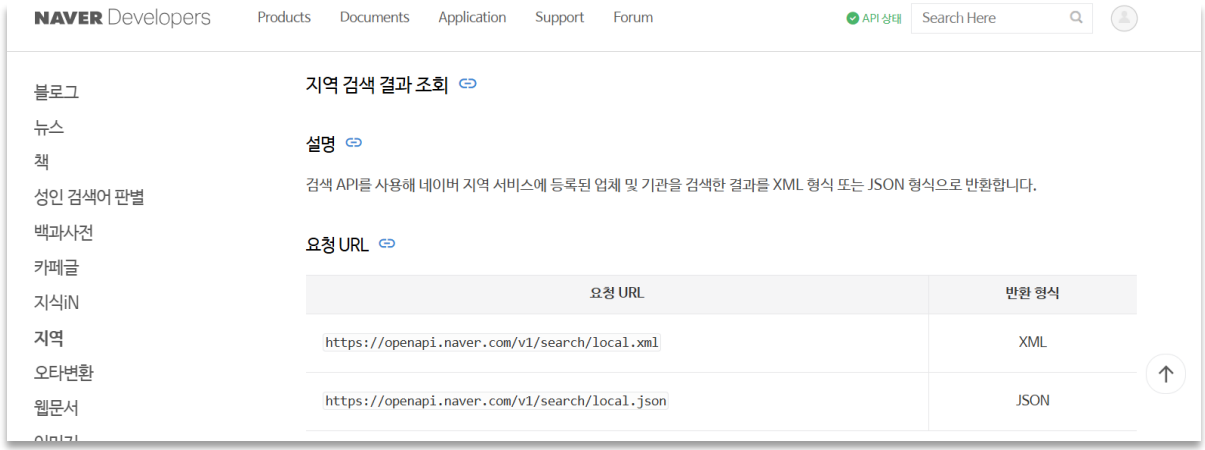
지도 시각화 (Folium)

과정 3 – 지도 시각화

가게 이름으로 지도 정보 조회

위도 및 경도 데이터 수집
음식점이 아닌 업종 제외

지도 시각화 (Folium)



네이버 API 이용 > 위도 및 경도 위치 반환.

```
if(rescode==200):
    response_body = response.read()
    json_response = response_body.decode('utf-8')
    json_response=json.loads(json_response)
    i=json_response['items'][0]['category']
    if i.find("한식")!=-1 or i.find("음식점")!=-1:
        x.append(int(json_response['items'][0]['mapx'])/10000000)
        y.append(int(json_response['items'][0]['mapy'])/10000000)
    else:
        x.append(-1)
        y.append(-1)
```

과정 3 – 지도 시각화

애프터네일 제주 아라동 페디

-> 네일 맛집, 염색 맛집, 헤어컷 맛집, ….

가게 이름으로 지도 정보 조회

위도 및 경도 데이터 수집
음식점이 아닌 업종 제외

지도 시각화 (Folium)

```
한식>장어,먹장어요리
음식점>양식
음식점>카페,디저트
한식>돼지고기구이
음식점>한식
쇼핑,유통>컴퓨터,모니터
음식점>한식
음식점>한식
여행,명소>도보코스
한식>육류,고기요리
음식점>일식>일식당
음식점>한식
한식>육류,고기요리
```

json_response['items'][0]['category']

업체/기관의 분류 정보

식당의 경우 음식점 혹은 한식

과정 3 – 지도 시각화

가게 이름으로 지도 정보 조회

위도 및 경도 데이터 수집
음식점이 아닌 업종 제외

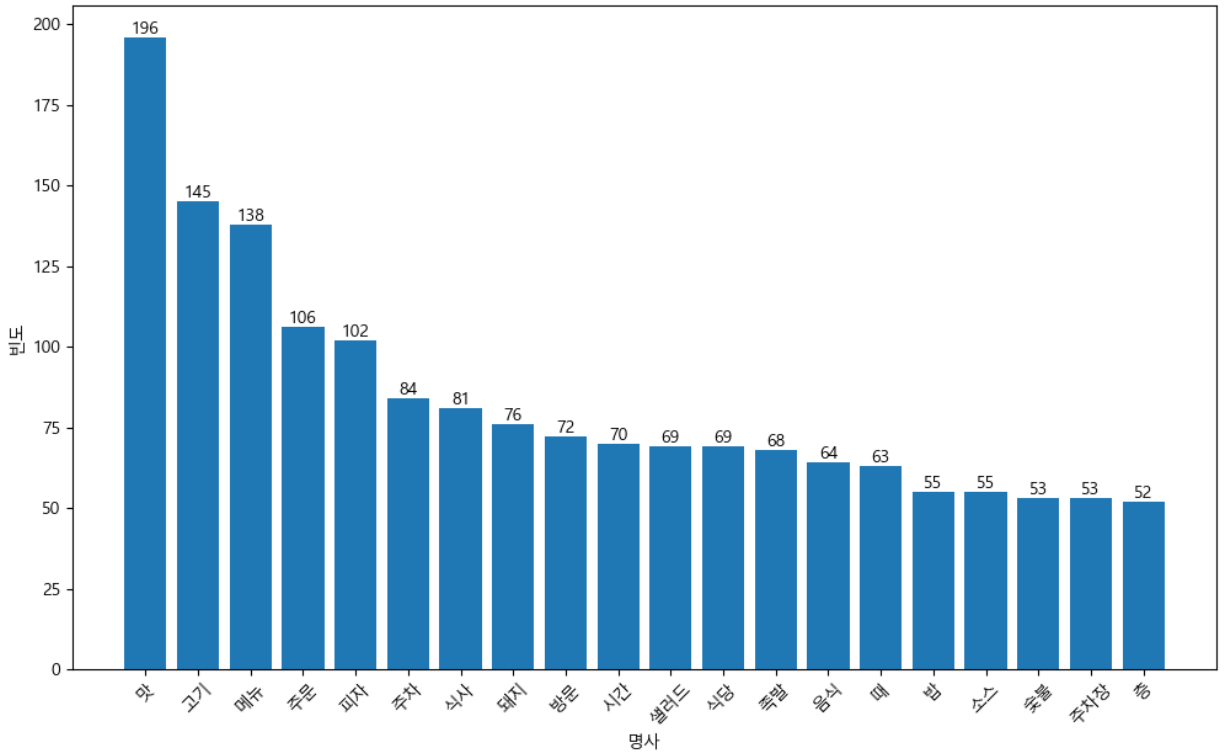
지도 시각화 (Folium)

Folium을 이용한 지도 시각화

- 언급된 장소 표시
- 언급 빈도 표시 (CircleMarker)
  많이 언급이 될 수록 큰 원

```
#counting
z=Counter(location)


map_ara = folium.Map(location=[y[0],x[0]],zoom_start=13)
for i,j,k in zip(x,y,location):
    if i!=-1:
        folium.Marker([float(j),float(i)],tooltip=k,popup=f"count:{z.get(k)}").add_to(map_ara)
        folium.CircleMarker([float(j),float(i)],radius = z.get(k) * 30,fill_color='skyblue').add_to(map_ara)

map_ara.save('hotplace_map.html')
```

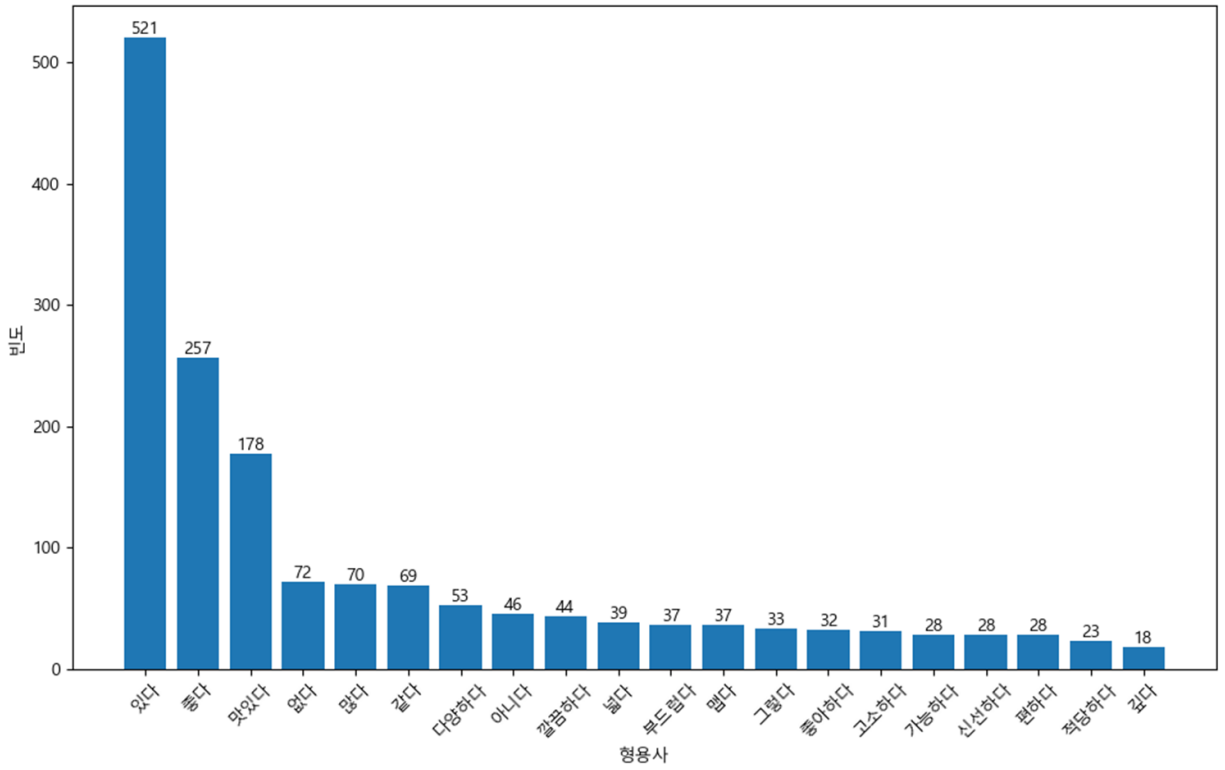
분석 결과



음식 : 피자, 돼지, 족발, 샐러드, 회, 닭갈비

시설 : 주차장, 공간, 주문, 방문, 식사

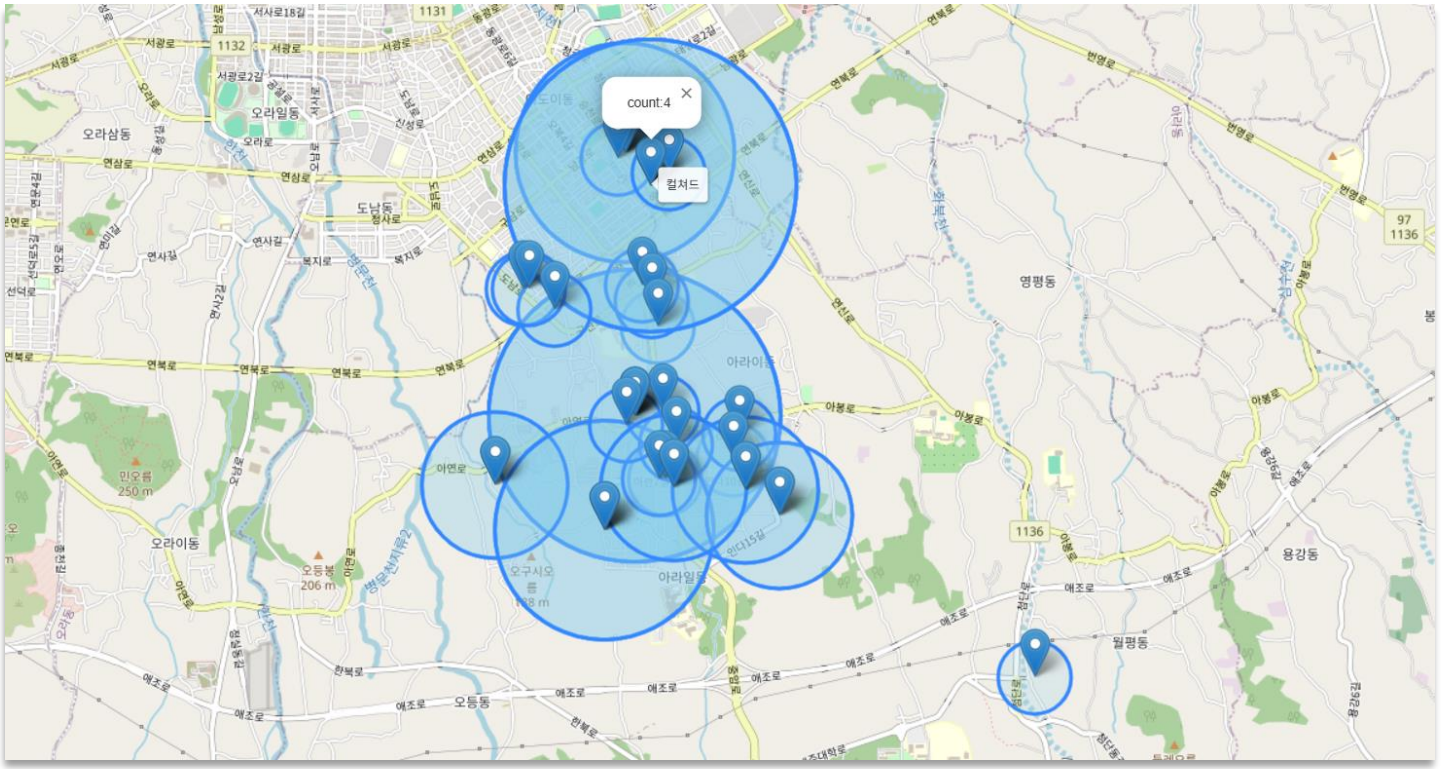
분석 결과



맛에 대한 표현 : 맛있다, 부드럽다, 맵다, 고소하다

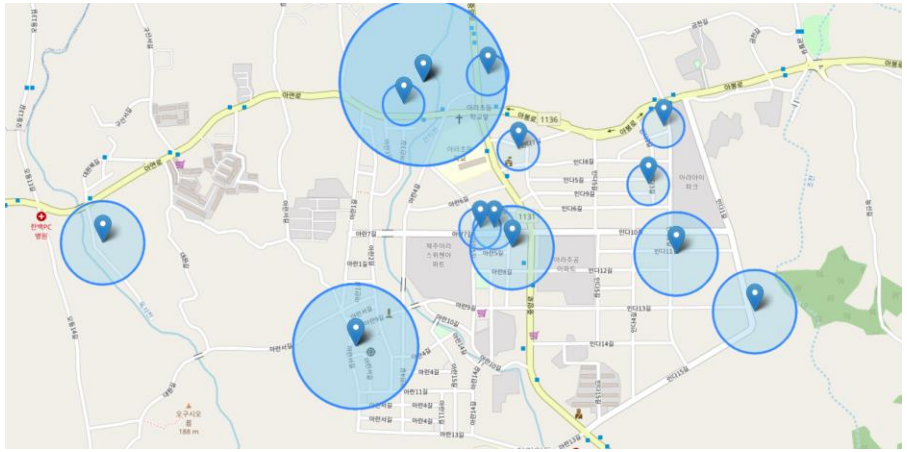
그 외 감상 : 좋다, 많다, 다양하다, 넓다

분석 결과

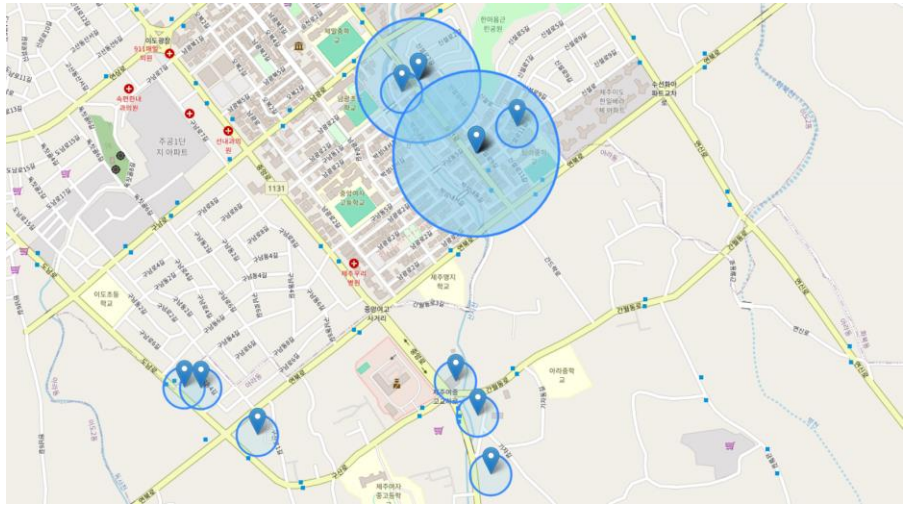


업체명	언급 횟수
광육	4회
컬처드	4회
코리아참숯불정육식당	3회
제주화도흑돼지	3회

분석 결과



<아라동>



<이도이동>

사람들이 생각하는 아라동?

실제 아라동 : 60% 이도이동 : 40%

> 사람들이 아라동이라고 인식하는 범위가 꽤 넓다.

향후 발전 방향

<과정 1 – SNS (블로그) 크롤링>

1. 장소를 찾아내는 방법에 대한 아쉬움.
2. 수집한 데이터를 활용하지 못함.
3. 더 많은 데이터?

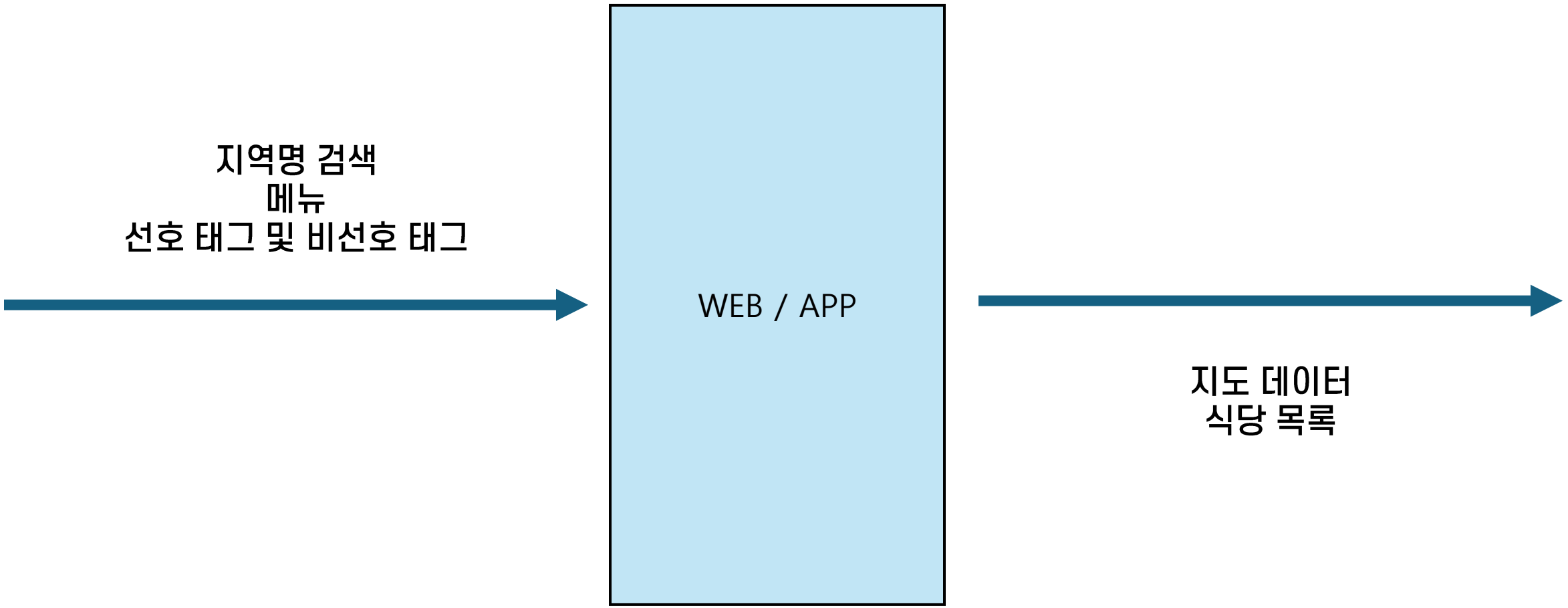
<과정 2 – 키워드 분석>

1. 제외 단어 선정 방법.
2. 빈도 분석 외의 다양한 데이터 분석 방법 선정.

<과정 3 – 지도 시각화>

1. 시각적 정보 추가.

향후 발전 방향




감사합니다!
https://github.com/YunYunYY/CrawlingNaverBlog2025