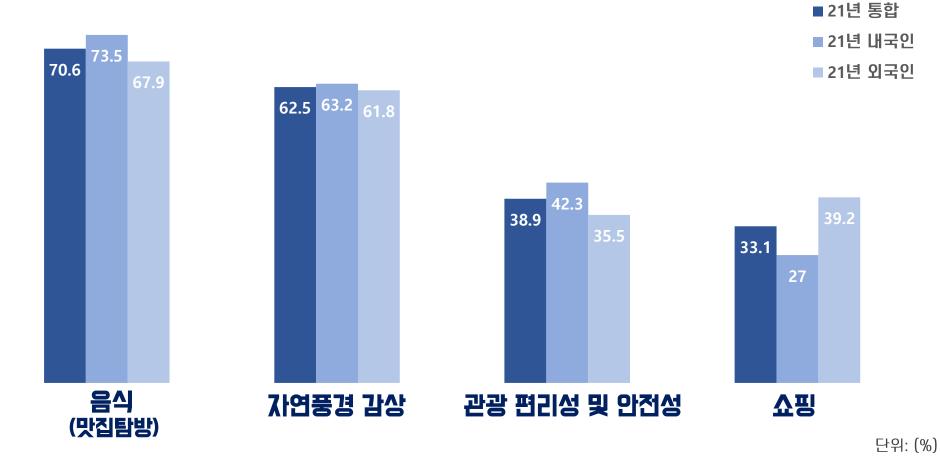


맛집 탐방 테마 선정 이유 -부산 관광지 목적



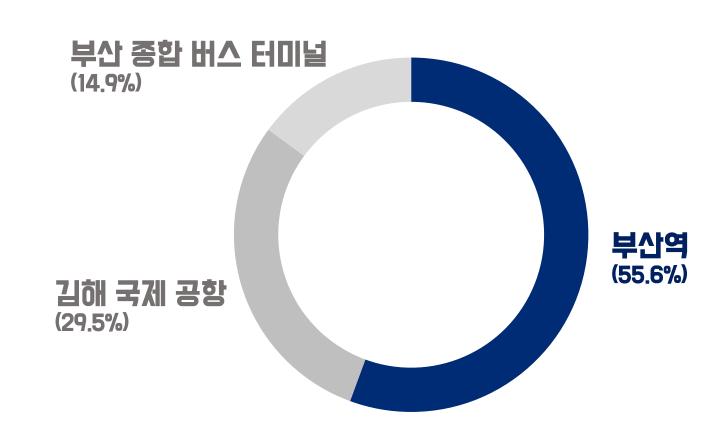
출처: 부산관광공사 2021 부산 방문객 관광 실태조사





1호선 선정 이유

- 관문 지역 방문 현황



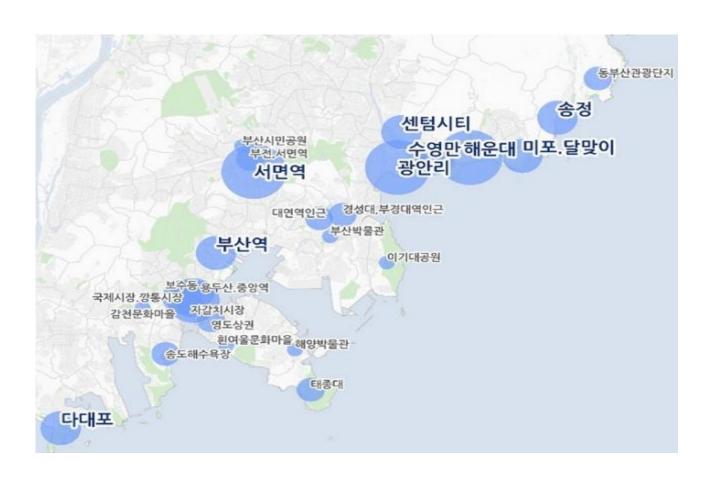


-주요 방문 지점 현황

주요 지접	20년 상반기	19년 상반기
서면	2,825,094	3,631,860 (2위)
해운대 해수욕장	2,462,509	3,451,231 (4위)
BIFF 광장 일원	1,971,205	3,826,210 (1위)
자갈치, 국제시장	1,813,437	3,581,784 (3위)
전포 카페거리	1,633,633	1,774,195 (5위)



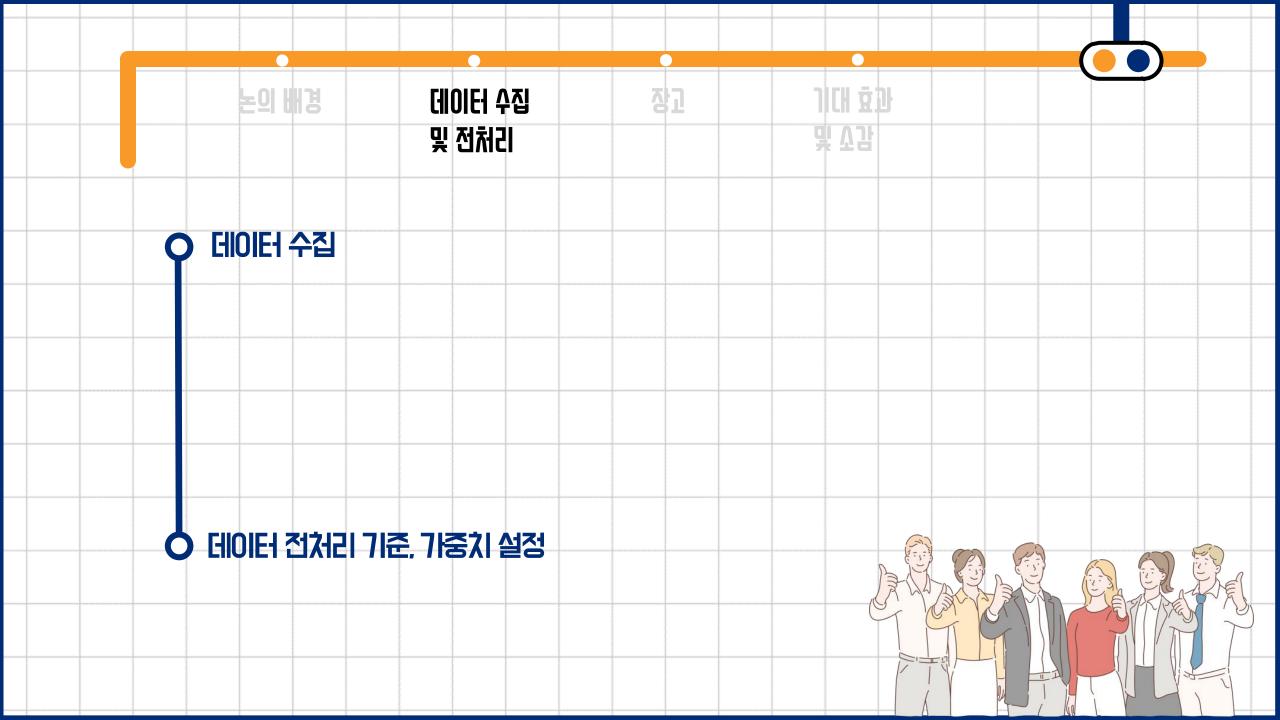
소외 지역 선정 이유



- 해운대, 서면, 남포 이외 방문 저조
- ㆍ 지역 간 양극화 심화

출처: 코레일

부산 주요 관광 방문지 그래픽







NAVER

kakao

DATA RANGO I KR

- 데이터 풍부
- 데이터 신뢰도가 셋 중에서 가장 낮음

• 상대적 데이터 부족

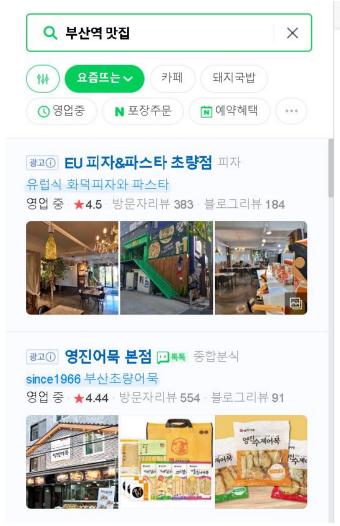
- 다방면 데이터 풍부
- 맛집 관련 데이터 부족

FIIDI

```
# 관련 모듈 임포트
import urllib
import time
from selenium import webdriver
from bs4 import BeautifulSoup
import pandas as pd
DRIVER_PATH = 'chromedriver.exe'
BASE_URL = 'https://map.naver.com/v5/search/'
def collectFamousStore(station):
   # 키워드 생성 후 URL로 변환
   keyword = station + "맛집"
   encText = urllib.parse.quote(keyword)
   url = BASE_URL + encText
   # selenium 브라우저로 해당 URL 접속
   browser = webdriver.Chrome(DRIVER_PATH)
   time.sleep(2)
   browser.get(url)
```

```
# 필요한 정보가 있는 iframe 페이지에 name 속성명을 통한 접근
browser.switch_to.frame('searchIframe')
time.sleep(2)
```





```
D 요소 콘솔 소스
메모리
                                               응용 프로그램
                                                              보안
                                                                      Lighthouse
                                                                                   CSS 개요 👗
                   ▼ <combined-search-list _ngcontent-cdg-c134 _nghost-cdg-c132 style="visibility: visible;">
                    ▼ <salt-search-list ngcontent-cdg-c132 nghost-cdg-c116 class="ng-star-inserted" style="visibility: visi
                      ble;">
                      ▼ <nm-external-frame-bridge _ngcontent-cdg-c116 style="height: 100%;">
                        ▼ <nm-iframe _nghost-cdg-c107 class="ng-star-inserted" style="height: 100%; pointer-events: auto;">
                          ▶<iframe _ngcontent-cdg-c107 src="about:blank" id="<mark>searchIframe</mark>" title="검색어 '부산역 맛집'에 대한
                           결과">...</iframe>
                         </nm-iframe>
                         <!--->
                         <!--->
                         <!--->
                       </nm-external-frame-bridge>
                      ▶ <salt-marker-list ngcontent-cdg-c116 nghost-cdg-c115 class="ng-star-inserted">...</salt-marker-list>
                       <!--->
                       <!--->
                       (!--->
                      </salt-search-list>
                      <!--->
                      <!--->
                      <!--->
                      <!--->
                    </combined-search-list>
                  </div>
                 ▶ <div ngcontent-cdg-c134 class="sub ng-star-inserted hidden">...</div>
                  <!--->
                  <!--->
                 </search-layout>
                 <!--->
               </app-base>
               <!--->
               <!--->
             </div>
           </shrinkable-layout>
           <div _ngcontent-cdg-c62 class="map_box">...</div>
```

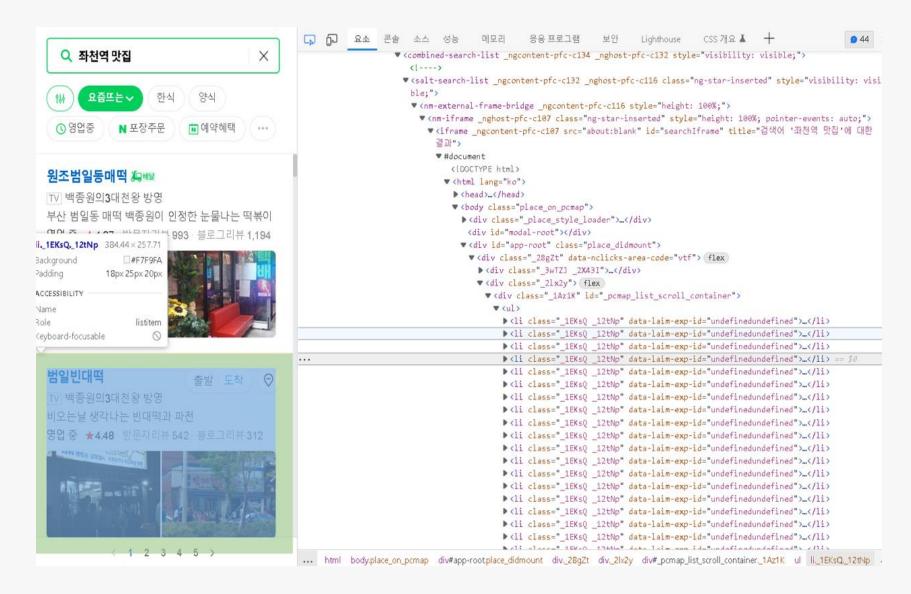
HOIEI 수집 방법

```
# 데이터를 담을 리스트와 각 항목에 대한 접근을 확인하기 위한 cnt 변수 생성 outList = [] cnt = 0

while True:

# 스크롤을 하면 추가되는 <Ii> 항목을 모두 가져오기 위한 div 스크롤 조작 container = browser.find_element_by_id('_pcmap_list_scroll_container') browser.execute_script("arguments[0].scrollBy(0, 3000)", container) time.sleep(2)
browser.execute_script("arguments[0].scrollBy(0, 4500)", container) time.sleep(2)
browser.execute_script("arguments[0].scrollBy(0, 5000)", container) time.sleep(2)
```









```
# 스크롤 된 페이지 소스를 bs4 객체로 변환
htmlStr = browser.page_source
bs = BeautifulSoup(htmlStr,'html.parser')

targetDiv = bs.select_one('div#_pcmap_list_scroll_container')
targets = targetDiv.select("li[class='_1EKsQ_12tNp']")

# 페이지당 50개인 <Ii>항목을 모두 가져왔는지 확인
print(len(targets))
```

```
for t in targets:
   # 접근 중인 항목 번호
   cnt += 1
   print(cnt)
   # 점포명
   name = t.select_one('span.place_bluelink').text
   # 별점 (별점이 없으면 None을 저장)
   score = t.select_one('em')
   if score:
       score = float(score.text)
   else:
       score = None
   # 썸네일 이미지
   img = t.select_one('div.cb7hz')
   if img:
       img = img['style'].split('"')[1]
   else:
       img = None
   # 영업 상태, 별점, 리뷰 등이 담겨있는 div 태그 텍스트
   info = t.select_one('div._17H46')
```

```
if info:
   info = info.text
   # 블로그리뷰 를 기준으로 split한 리스트의 길이가 1이면
   # 블로그리뷰에 대한 정보가 없는 것이므로 None을 저장
   if len(info.split('블로그리뷰'))== 1:
      blogNum = None
      if len(info.split('방문자리뷰'))== 1:
         visitNum = None
      else:
      # 리뷰 수가 1000 이상이면 ,로 구분되므로 제거
          info = info.split('방문자리뷰')
         visitNum = int(info[-1].replace(',',''))
   else:
      info = info.split('블로그리뷰')
      blogNum = int(info[-1].replace(',',''))
      # split된 텍스트 중 나머지 정보가 담긴 텍스트로 이동
      info = info[0]
      if len(info.split('방문자리뷰'))== 1:
         visitNum = None
      else:
          info = info.split('방문자리뷰')
         visitNum = int(info[-1].replace(',','))
```

```
else:
    blogNum = None
    visitNum = None

outList.append([name, score, img, visitNum, blogNum])
```

HIOIEI 수집 방법

break



탐색이 끝난 후 다음 페이지로 이동 # 이전 페이지, 5개의 페이지 번호, 다음 페이지 버튼으로 구성된 # 7개의 <a> 태그 중 다음 페이지 버튼인 마지막 <a> 태그 선택 nextBtn = browser.find_elements_by_css_selector('div._2ky45 a')[-1] # 해당 버튼의 class명 확인 isLastPage = nextBtn.get_attribute("class") print(isLastPage) # 클래스명이 "_2bgjK "이면 (공백 포함 주의 !!) # 마지막 페이지가 아니므로 다음 페이지로 이동 if isLastPage == "_2bgjK ": print("this is not last page.") nextBtn.click() time.sleep(2) # class명이 "_2bgjK _34ITS"이면 마지막 페이지이므로 반복문 종료 else:



```
# 엑셀에서 한글이 깨지지 않는 'utf-8-sig' 인코딩으로 csv 파일 추출
df = pd.DataFrame(outList,columns=['점포명','별점','이미지','방문자 리뷰 수', '블로그 리뷰 수'])
df.to_csv(f"{keyword}CSV.csv",encoding='utf-8-sig')

if __name__ == "__main__":
    collectFamousStore(input())

# stationList = ['범내골역','범일역','좌천역','부산진역','초량역','부산역','중앙역','남포역','자갈치역']

# 지하철역명 입력 후 collectFamousStore 함수로 전달

# for station in stationList:
    collectFamousStore(station)
```





데이터 전처리 기준

	점포명	별점	이미지	방문자 리뷰 수	블로그 리뷰 수
0	백화양곱창	4.31	https://search.p	274	1645
1	18번완당집	4.35	https://search.p	1859	1255
2	한월횟집	4.29	https://search.p	1176	1107
3	부산명물횟집	4.14		380	429
4	카마타케제면소 부산남포점	4.38		424	448
294	부영상회	4.43	https://search.p	50	13
295	동원양곱창12호	4.35		20	5
296	공차 롯데마트광복점	4.49		527	12
297	진주식당	4.27		60	2
298	우오가시 롯데백화점광복점	4.14		11	19
299	추억의 찐빵	4.26		173	

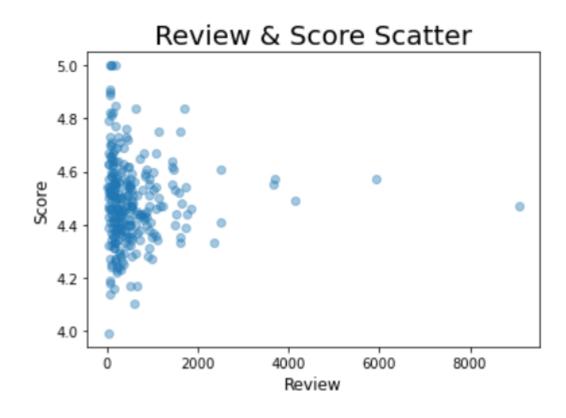
· 데이터 수집 결과

역 별로 300개인 데이터 확보

- 전처리 목표: 맛집 선정 유의미한 결과 확보
- · 별점 범위4~5절) + 리뷰 상[tbl(1점)+ 기본점수 4점
- · Review = 방문자 2 ዙ * 0.7 + 블로그 2 뷰 * 0.3







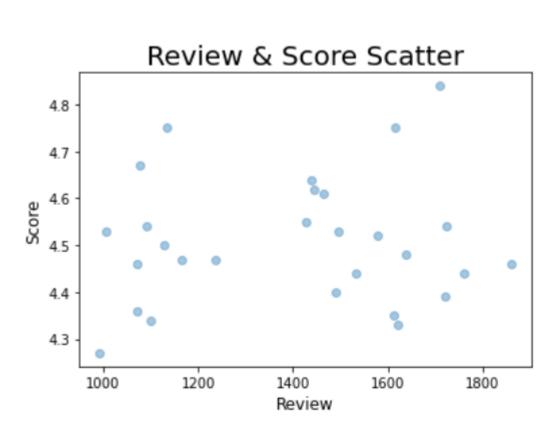
· 구분 기준(1) 〈법내골〉

2000 이상 - 1점

Review = 방문자리뷰 * 0.7 + 블로그리뷰 * 0.3

Score = 평점

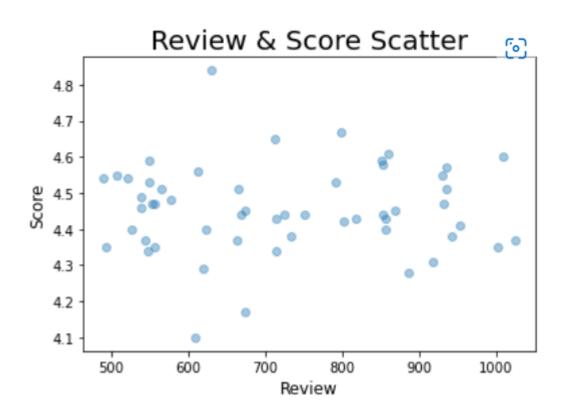




· 구분 기준(2) 〈법내골〉

1250 이상 - 0.9 점





· 구분 기준(3) 〈법내골〉

750 이상 - 0.8 점 600 이상 - 0.7 점





· 구분 기준(4) (조천역)

500 이상 - 0.6 점 400 이상 - 0.5 점 300 이상 - 0.4 점 200 이상 - 0.3 점 그 외 - 0.1 점





· 〈지갈치 역 Top 10〉

	점포명	별점	방문자 리뷰 수	로그 리뷰	수 점수
24	뚜벅스 남포	4.88	610	724	9.58
93	폴바셋 롯데백화점 광복점	4.54	4631	89	9.54
253	스타벅스 남포동역점	4.53	2969	128	9.53
84	모스버거 광복롯데점	4.49	2997	244	9.49
168	스타벅스 BIFF광장점	4.45	3811	104	9.45
21	옵스 광복롯데점	4.44	5469	331	9.44
90	스타벅스 국제시장점	4.46	1811	46	9.36
1	18번완당집	4.35	1859	1255	9.25
63	크리스피크림도넛 부산롯데광복점	4.42	1749	33	9.22
112	부산숯불갈비	4.38	965	300	9.18
290	홍가네양곱창	4.85	310	429	9.15

