중간 보고서

통합 리뷰를 기반으로 한 제품 안내 어플

Vol.1

제출일	2020. 01, 24	전공	컴퓨터공학과
과목	졸업작품 프로젝트	학번	2015722084 2015722083
담당교수	이기훈	이름	한승주 김성종

목 차

I. 개요

1. 배경 및 필요성

- 서로 다른 플랫폼과 웹사이트는 사용하는 연령대나, 나이 대 등이 달라 제품의 정보를 단 하나의 사이트만 보고는 신뢰성 있는 정보를 통한 구매가 불가능하다.

- 실제 국내 페이스북 유저 비율은 남성이 여성 대비 14% 많고, 인스타그램은 여성이 남성 대비 4% 많은 비율을 가지고 있다. 또한 페이스북은 연령대가 고른 반면, 인스타그램은 20~30대 비율이 상대적으로 높다.

- 페이스북, 인스타그램, 트위터와 같은 소셜미디어에 자주 노출되는 광고와 함께 실제 사용자가 올리는 리뷰, 그리고 제품 구매처인 쇼핑몰(G 마켓, 옥션 등), 제품 가격 비교 사이트(다나와, 네이버 비교 쇼핑 등)은 여러 존재하지만, 이렇게 많은 웹사이트와 플랫폼에 퍼져 있는 사용자의 구매 후기를 통합적으로 파악할 수 있는 곳은 없다.

2. 목적

- **정보의 신뢰성**

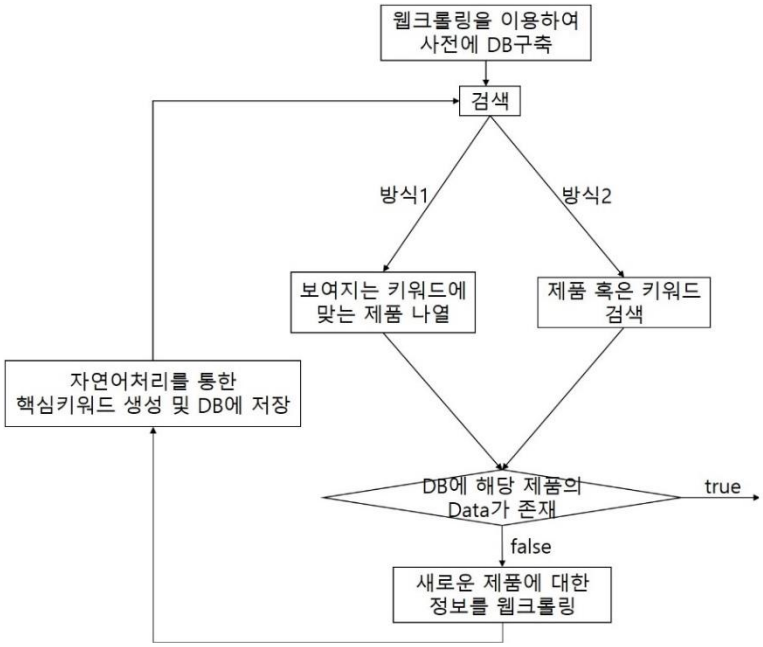
  여러 플랫폼에 퍼져 있는 제품의 리뷰를 중요 키워드를 중심으로 데이터베이스를 구축하면 이렇게 만들어진 데이터베이스를 통해 키워드만으로 사용자가 원하는 조건의 제품을 소개하고 찾아볼 수 있다. 긍정적 리뷰와 부정적 리뷰를 다 같이 한번에 보여줌으로써 제품의 장점뿐만 아니라 단점을 파악하고 같은 제품군을 비교 구매하고 싶어 하는 사용자에게 편의성을 제공할 수 있다.

- **접근의 용이성**

  이런 결과를 어플리케이션(혹은 웹사이트)으로 추가적인 구현을 하면 사용자가 언제 어디서든 쉽게 접근하고 사용할 수 있다.

3. 설계내용

ㄱ. Flow Chart



ㄴ. 개념 설계

1) 플랫폼의 웹사이트 코드 분석 및 크롤링(Beautiful soup)

- DB 를 구축하기 위해 해당 웹사이트 접속
- 제품 정보와 사용자 및 구매자의 리뷰가 담긴 웹사이트 코드 분석
- BS4를 이용해 페이지 데이터 호출
- 제품의 기본 정보(이름, 가격, 리뷰 등) tag 를 찾아 추출
- 각 사이트와 페이지별로 링크를 재귀적으로 검색하여 데이터를 추출

2) 크롤링된 정보를 이용한 DB 구축(Mysql)

- MySQL 서버에 접속하여 데이터베이스 생성
- cursor 를 추출하여 execute 메서드로 SQL 을 실행, 테이블 생성
- Execute 메서드에 데이터를 계속 확장

3) Kkma, Hannanum 을 이용한 KoNLP(키워드 생성)

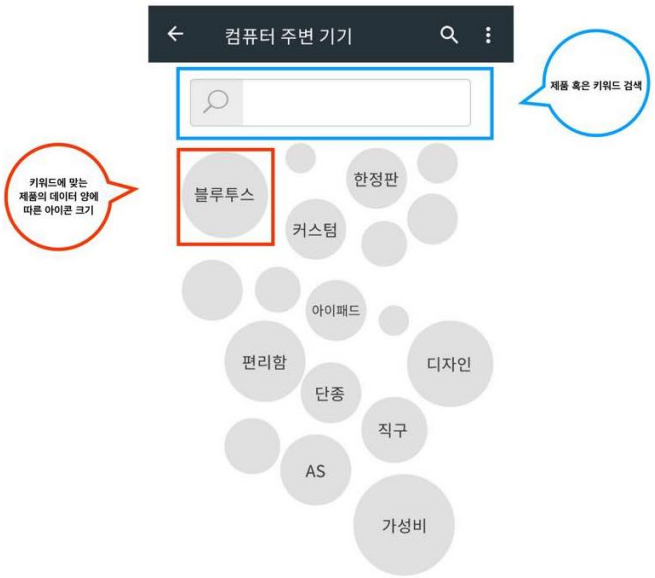
- Kkma 나 Hannanum 모듈을 이용하여, 해당 모듈에 맞추어 입력된 문자열에서 키워드로 표현할 품사 추출

- 가장 빈도수가 높은 단어(키워드로 설정할 단어)를 DB 에 저장

4) Android UI 제작

- 자신이 구매하고자 하는 제품의 리뷰를 보기 위한 제품의 검색창과 위에서 나타낸 키워드 중 전체 제품에서 가장 많은 비중을 차지하는 몇 개의 키워드를 다음과 같이 제품 검색창 아래에 사용자가 보기 편하도록 UI 로 구현

- 검색 시, 제품 검색 및 키워드(조건)를 검색할 수 있게 구현



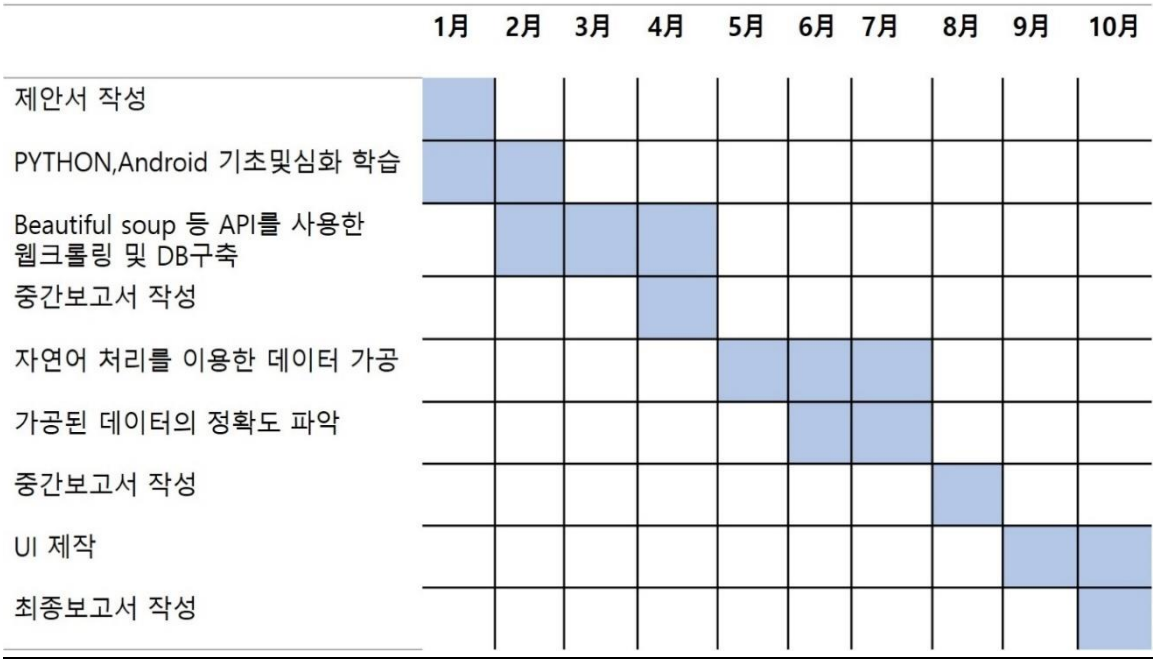
- 제품 검색 시, 제품의 가격 정보와 리뷰의 중요 키워드를 노출

- 실제 리뷰를 사이트 별로 노출

- 어플 자체 리뷰를 남겨 사용자가 어플을 더 다양하게 사용하도록 유도

4. 과제 수행

1. 수행 일정


1月
2月
3月
4月
5月
6月
7月
8月
9月
10月
제안서 작성
PYTHON,Android 기초및심화 학습
Beautiful soup 등 API를 사용한
웹크롤링 및 DB구축
중간보고서 작성
자연어 처리를 이용한 데이터 가공
가공된 데이터의 정확도 파악
중간보고서 작성
UI 제작
최종보고서 작성

2. 웹 크롤링

ㄱ. 트위터

트위터의 정식 API인 tweepy가 존재는 하나 단점이 최근 7일간의 트윗만을 가져올 수 있기 때문에, 리뷰분석을 하는데에 있어서 한계가 존재한다. 그래서 사용한 API는 getoldtweet3(https://github.com/Jefferson-Henrique/GetOldTweets-python)이며, 해당 API는 기간을 설정하여 트윗을 수집할 수 있다. 기간은 2019년 1년동안 '로지텍'이라는 검색어가 들어간 트윗을 수집해보았다.

In [6]:
```
try:
    import GetOldTweets3 as got
except:
    !pip install GetOldTweets3
    import GetOldTweets3 as got
```

In [7]:
```
import datetime

days_range = []

start = datetime.datetime.strptime("2019-01-01", "%Y-%m-%d")
end = datetime.datetime.strptime("2019-12-31", "%Y-%m-%d")
date_generated = [start + datetime.timedelta(days=x) for x in range(0, (end-start).days)]

for date in date_generated:
    days_range.append(date.strftime("%Y-%m-%d"))

print("=== 설정된 트윗 수집 기간은 {} 에서 {} 까지 입니다 ===".format(days_range[0], days_range[-1]))
print("=== 총 {}일 간의 데이터 수집 ===".format(len(days_range)))
```

```
=== 설정된 트윗 수집 기간은 2019-01-01 에서 2019-12-30 까지 입니다 ===
=== 총 364일 간의 데이터 수집 ===
```

In [8]:
```
import time

# 수집 기간 맞추기
start_date = days_range[0]
end_date = (datetime.datetime.strptime(days_range[-1], "%Y-%m-%d")
            + datetime.timedelta(days=1)).strftime("%Y-%m-%d") # setUntil이 끝을 포함하지 않으므로, day + 1

# 트윗 수집 기준 정의
tweetCriteria = got.manager.TweetCriteria().setQuerySearch('로지텍')₩
                                           .setSince(start_date)₩
                                           .setUntil(end_date)₩
                                           .setMaxTweets(-1)

# 수집 with GetOldTweet3
print("Collecting data start.. from {} to {}".format(days_range[0], days_range[-1]))
start_time = time.time()

tweet = got.manager.TweetManager.getTweets(tweetCriteria)

print("Collecting data end.. {0:0.2f} Minutes".format((time.time() - start_time)/60))
print("=== Total num of tweets is {} ===".format(len(tweet)))
```

```
Collecting data start.. from 2019-01-01 to 2019-12-30
Collecting data end.. 15.82 Minutes
=== Total num of tweets is 7521 ===
```

가져온 트윗에서 사용할 정보는 유저, 트윗 개시 날짜, 트윗 링크, 내용이며 해당 내용을 csv 파일로 저장한다.

In [10]:
```
from random import uniform
from tqdm import tqdm_notebook

# initialize
tweet_list = []

for index in tqdm_notebook(tweet):

    # 메타데이터 목록
    username = index.username
    link = index.permalink
    content = index.text
    tweet_date = index.date.strftime("%Y-%m-%d")
    tweet_time = index.date.strftime("%H:%M:%S")


    # 결과 합치기
    info_list = [tweet_date, tweet_time, username, content, link]
    tweet_list.append(info_list)

    # 휴식
    time.sleep(uniform(1,2))
```

```
HBox(children=(IntProgress(value=0, max=7521), HTML(value='')))
```

In [12]:
```
# 파일 저장하기

import pandas as pd

twitter_df = pd.DataFrame(tweet_list,
                          columns = ["date", "time", "user_name", "text", "link"])

# csv 파일 만들기
twitter_df.to_csv("sample_twitter_data_{}_to_{}.csv".format(days_range[0], days_range[-1]), index=False)
print("=== {} tweets are successfully saved ===".format(len(tweet_list)))
```

```
=== 7521 tweets are successfully saved ===
```

저장한 파일을 확인하면

In [13]:
```
# 파일 확인하기

df_tweet = pd.read_csv('sample_twitter_data_{}_to_{}.csv'.format(days_range[0], days_range[-1]))
df_tweet.head(10) # 위에서 10개만 출력
```

Out[13]:

	date	time	user_name	text	link
0	2019-12-30	18:37:18	s_NEGEV_s	와 근데 로지텍 마우스 G903이 HERO 센서 달고 오니까 배터리가 거의 닳지를 ...	https://twitter.com/s_NEGEV_s/status/121171791...
1	2019-12-30	16:29:14	mookbini	로지텍 빠순이라 키보드도 마우스도 헤드셋도 로지텍으로 하고싶었지만 만만치않았다구한ㄴ다	https://twitter.com/mookbini/status/1211685685...
2	2019-12-30	16:09:43	KeepGoingMasiro	지금 게임용으로 쓰고 있는 키보드.. 사실 엄청 오래 되기는 했는데.. 아직도 잘 ...	https://twitter.com/KeepGoingMasiro/status/121...
3	2019-12-30	14:03:32	RyuZU_Seed	음 그리고 키보드는 COX CK450 마우스는 로지텍 G102 정도면.. 80 살짝...	https://twitter.com/RyuZU_Seed/status/12116490...
4	2019-12-30	11:34:39	nabislife	마우스 로지텍 mx 버티컬 쓰고있음	https://twitter.com/nabislife/status/121161155...
5	2019-12-30	10:37:24	lulul_jd	안녕하세요.. 멍청하게 자기 아이디도 모르는 놈.. 팔로 해주셔서 감사해요... 로...	https://twitter.com/lulul_jd/status/1211597147...
6	2019-12-30	09:18:35	sinrisoung	킹직히 유로트럭 최고세팅은 이거아님? 트리플모니터 하는 비용보다 vr저렴하게 업어오...	https://twitter.com/sinrisoung/status/12115773...
7	2019-12-30	09:12:54	wannabecoolman	로지텍 블투키보드 고장났어.. 미쳤나... 내손에 남아나는게 없다는 뜻이니.....	https://twitter.com/wannabecoolman/status/1211...
8	2019-12-30	09:01:08	kkibaek	로지텍 키즈투고, 이거 사지 마세요. '블루투스 키보드'의 본질을 놓친 키보드	https://twitter.com/kkibaek/status/12115729177...
9	2019-12-30	09:00:22	binu_4_lvlz	로지텍 keys to go 키보드 타자 느낌이 좋아서 계속 치고 있음 ㅋㅋㅋㅋㅋ	https://twitter.com/binu_4_lvlz/status/1211572...

이러한 결과를 얻을 수 있다.

ㄴ. 네이버

실제로 많은 사용자들이 많이 보는 리뷰는 네이버일 것이다. 따라서 네이버 블로그의 검색어에 따른 결과 크롤링을 해보면 제목과 블로그 링크를 가져올 수 있다. 무한정으로 많은 양의 검색결과를 가져올 수 없는데 이는 좋은 검색결과를 위해 네이버가 1000건의 검색결과만을 보여주고 있기 때문이다. 하지만 이것만으로도 충분히 키워드를 추출한다던지, 제품 한 개를 분석함에 있어서 부족함이 보이지는 않는다.

In [30]:
```
import requests
import pandas as pd
from bs4 import BeautifulSoup
from collections import OrderedDict
from itertools import count

def mycrawler(input_search):
    url='https://search.naver.com/search.naver'
    post_dict=OrderedDict()

    for page in count(1):
        params={
            'query': input_search,
            'where': 'post',
            'start': (page-1)*10+1,
            'date_from': 20191231,
            'date_to' : 20200220,
        }
        print(params)
        response = requests.get(url, params=params)
        html=response.text

        soup=BeautifulSoup(html, 'html.parser')

        title_list=soup.select('.sh_blog_title')

        for tag in title_list:
            if tag['href'] in post_dict:
                return post_dict

            print(tag.text, tag['href'])
            post_dict[tag['href']]=tag.text
    return post_dict
```

In [31]:
```
result=mycrawler('로지텍')
print(len(result))
```

```
로지텍 MX KEYS 무선키보드 : 매력적이고 편리한 팬타그래프... https://blog.naver.com/purplecrom?Redirect=Log&logNo=221678278985
로지텍 G900 마우스 스위치 교체 도전 https://blog.naver.com/soundbross?Redirect=Log&logNo=221731982821
맥북 마우스 추천! 로지텍m350 (디자인중심) https://blog.naver.com/sevenlove__?Redirect=Log&logNo=221709979670
무선 게이밍 키보드 로지텍 G613 블루투스까지 품다 https://neces2.blog.me/221465140806
로지텍 G410 기계식 키보드 수리 - 스위치 불량 https://blog.naver.com/azrama?Redirect=Log&logNo=221599052278
무선 게이밍 마우스 로지텍 G304으로 즐긴 로스트아크 http://isaac.pe.kr/221444525789
{'query': '로지텍', 'where': 'post', 'start': 981, 'date_from': 20191231, 'date_to': 20200220}
로지텍G PRO GAMING MOUSE... 증상으로 로지텍마우스수리... http://cardin.co.kr/221531572598
게이밍 스피커 로지텍 G560 후기 http://mnteye.com/221665338654
블루투스 마우스! 로지텍 MX 버티컬 후기 http://blingyue.com/221460503384
게이밍 헤드셋 추천 로지텍 G933S http://gomdora.com/221608392732
로지텍 k380 블루투스 키보드 오프라인 파는곳 / 연결방법 https://blog.naver.com/sini0222?Redirect=Log&logNo=221811762513
[IT주변기기구입]무선마우스 로지텍 M171 (Wireless... https://blog.naver.com/dazzling_jun?Redirect=Log&logNo=221722749598
태블릿 무선 키보드 '로지텍 K380' https://blog.naver.com/xhdlalr45?Redirect=Log&logNo=221687643672
게이밍 키보드 즉각적인 반응을 보여준 로지텍 G512 택타일... https://blog.naver.com/dogslife78?Redirect=Log&logNo=221584762334
로지텍 마우스 럭키박스 개봉기 / 소소하고 확실한 도박 https://blog.naver.com/pinkbomi?Redirect=Log&logNo=221701602578
로지텍 MX MASTER 더블클릭이 잘 안되는 증상으로 마우스수리... http://cardin.co.kr/221559555784
{'query': '로지텍', 'where': 'post', 'start': 991, 'date_from': 20191231, 'date_to': 20200220}
990
```

위는 글에서 글의 제목과 링크 주소를 가져온 스크래핑의 결과인데, 해당 링크를 타서 본문을 가져올 수 있지만, 해당 기능은 다음인 인스타그램에서 구현해보고자 한다.

ㄷ. 인스타그램

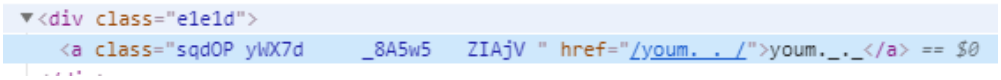
네이버 블로그가 긴 리뷰, 트위터가 짧은 리뷰라고 하면 인스타그램은 해쉬태그라는 자신만의 키워드를 표현하는 기능이 있지만, 실제로 분석함에 있어서는 해쉬태그를 따로 볼 것이 아니라 전체적인 리뷰로 보고 트위터와 유사한 짧은 리뷰라 판단하여 전체적인 분석을 하고자 한다.

```
def InstagramUrlFromKeyword(browser, keyword, numofpage):
    keyword_url_encode = quote(keyword)  # 한글인식

    url = 'https://www.instagram.com/explore/tags/' + keyword_url_encode + '/?hl=ko'

    browser.get(url)

    arr_href = []

    body = browser.find_element_by_tag_name('body')

    for i in range(numofpage):
        body.send_keys(Keys.PAGE_DOWN)

        time.sleep(1)

    time.sleep(3)

    post = browser.find_elements_by_class_name('v1Nh3')

    for j in post:
        href_str = j.find_element_by_css_selector('a').get_attribute('href')

        arr_href.append(href_str)  # append 추가시키는거

    return arr_href
```

인스타그램에서 어떤 키워드로 검색어를 하면 해당 키워드인 한글이나 영어는 컴퓨터가 해석하기 난해하므로, 인코딩을 통해 해당 검색어에 맞추어 주소변환이 가능하다. 또한,

여러 글의 본문을 찾기 위해서 해당 함수를 사용하게 되는데, 인스타그램 웹의 소스코드에는 v1Nh3 class 밑에 href를 통해 모든 글의 주소를 가져올 수 있어서 해당 글의 ref를 가져오기 위한 정의이다.

```
def IdHashTagFromInstagram(browser, url):
    browser.get(url)

    insta_id = ""

    hash_data = ""

    wait = WebDriverWait(browser, 20)

    wait.until(EC.presence_of_element_located((By.CLASS_NAME, "e1e1d")))  # e1e1d는 아이디가 적혀있는 소스코드

    id_href = browser.find_elements_by_class_name('e1e1d')

    insta_id = id_href[0].find_element_by_css_selector('a').text  # id_href[0] 첫번째 있는 a 찾기

    wait.until(EC.presence_of_element_located((By.CLASS_NAME, "C4VMK")))  # 댓글들

    href = browser.find_elements_by_class_name('C4VMK')

    total_hash_text = []
```



아이디를 찾는 부분으로, eleId 클래스 영역에 href로 text만 뽑아내면 해당 부분이 인스타그램 id이므로 이를 추출한다.

```
for i in range(0, len(href)):  # 댓글가져와서 하나씩 끝까지 보는 거 len 몇개 개수

    hash_text = href[i].find_element_by_css_selector('span').text

    total_hash_text.append(hash_text)

    image_src = ''

try:

    image_temp = browser.find_element_by_class_name('KL4Bh').find_element_by_css_selector('img')  # 이미지 찾기

    image_list = image_temp.get_attribute('srcset')  # srcset이란 속성을 가지고 있는 애를 가져와라

    temp = image_list.split(',')  # ,로 구분해서 temp로 가져와라

    for i in temp:

        if '1080w' in i:  # 사진의 많은 url중에서 1080w있는 문자열 찾기

            image_src = i.split(' ')[0]  # url 1080w이 있는 링크에서 1080w를 떼고 공백 앞의 정보를 가져오기

except:

    image_src = ''  # 동영상이면(이미지가 아니면) 빈칸으로 둬라

    pass

return insta_id, image_src, total_hash_text
```

해당 코드는 for문을 이용하여 참조할 reference가 있는 동안 해당 ref의 본문을 축적하는 부분과 본문에 덧붙인 이미지의 src를 찾는 부분으로 구성이 되어있다.

```
▼<span class title="수정됨"> == $0
  "저한테 로지텍 핑크 k380이 있지만"
  <br>
  "늘 화이트색상의 키보드를 가지고 싶다는 생각이 마음 저 구석에 있는데, 드디어 저도 화이
  트 키보드와 마우스를 하나 더 가지게 되었어요!"
  <br>
  <br>
  "제 책상과도 아주 찰떡인 k580과 m350"
  <br>
```

본문의 부분으로 댓글또한 위와같이 span의 영역에 글이 작성되어 있다.

```
▼<div class="KL4Bh" style="padding-bottom: 100%;">
  <img alt="이미지: 화면, 실내" class="FFVAD" decoding="auto" sizes=
  "327px" srcset="https://scontent-ssn1-1.cdninstagram.com/v/t51.2885-
  15/sh0.08/e35/s…nc_ohc=ofaGvbRHTCAAX9xuG3v&oh=3651489…&oe=5E8B19FD
   640w, https://scontent-ssn1-1.cdninstagram.com/v/t51.2885-15/sh0.08/
  e35/s…nc_ohc=ofaGvbRHTCAAX9xuG3v&oh=7f6cbd4…&oe=5E7814FD 750w, https:/
  /scontent-ssn1-1.cdninstagram.com/v/t51.2885-15/e35/s1080x108…
  nc_ohc=ofaGvbRHTCAAX9xuG3v&oh=a27c787…&oe=5E8A724A 1080w" src="https:/
  /scontent-ssn1-1.cdninstagram.com/v/t51.2885-15/e35/s1080x108…
  nc_ohc=ofaGvbRHTCAAX9xuG3v&oh=a27c787…&oe=5E8A724A" style="object-fit:
  cover;">
</div>
```

Img_src또한 KL4Bh의 영역에 img를 find하여 찾을 수 있다.

```
browser = webdriver.Chrome('C:\chromedriver.exe')

keyword = input("검색어를 입력하세요 : ")

num_of_pages = 2

arr = InstagramUrlFromKeyword(browser, keyword, num_of_pages)

insta_df = pd.DataFrame(columns=['Insta ID', 'Image Src', 'Content'])

for url in arr:
```

```
    try:

        insta_id, image_src, hash_data = IdHashTagFromInstagram(browser, url)

        char = re.compile('[^[0-9a-zㄱ-ㅣ-가-힣!#?]')  # 재정비
        """
        정규식을 두 번 이상 사용한다면, 모듈의 match, search 함수는 효율적이지 않다.
        매번 match 혹은 search를 수행할 때마다, 정규식을 분석해서 처리하기 때문이다.
        효과적인 처리 방법은 정규식을 내부 표현식으로 일단 변환하고, 그것을 계속 활용하는 것이다.
        compile 함수가 정규식을 내부 표현식으로 변환하여 정규식 객체를 리턴한다.
        """
        hash_data_str = ""

        for data in hash_data:
            hash_data_str = hash_data_str + data

        hash_data_str = char.sub("", hash_data_str)  # ""를 hash_data_str으로 바꿔주기

        dic_insta = {"Insta ID": insta_id, "Image Src": image_src, "Content": hash_data_str}

        temp_df = pd.DataFrame(dic_insta, index=[0])  # index=0은 dic을 temp로 바꾸는데 에러가 나지 않도록 하는 것

        insta_df = insta_df.append(temp_df, ignore_index=True)

    except:

        print(sys.exc_info()[0])

        pass

nsta_df.to_csv('insta_temp.csv', mode='w', encoding='euc-kr')
```

실제로 동작하는 부분은 검색어를 입력받아서 위의 두가지 함수를 활용하여 인스타그램의 주소를 가져와서 해당 주소의 해쉬태그를 데이터프레임에 쌓고 이를 csv로 저장하는 부분이다.

	Insta ID	Image Src	Content
0	youm._._	https://scc	저한테로지텍핑크k380이있지만늘화이트색상의키보드를가지고싶다는생각이마음저구석에있는데드디어저도화이트키보드와마우스를하나더가지게되었어요!제책상과도아주칠떡인k580과m350제가키
1	picn2k	https://scc	마우스없이는노트북못쓰는일인항상저렴이마우스쓰다가새로장만한마우스!그립감도한상인데다크필드기능으로패드없이도유리대리석에사용가능하니진짜가장편해매번잡지깔고쓰는거너무불편했는데
2	kkomtokki	https://scc	라이연으로채워본책상#라이언데스크매트#라이언마우스패드#허브그리고요즘쓰고있는#로지텍#무선키보드와#무선마우스디자인이깔끔해서예쁜데사용감도좋아서계속쓰는중#kakao#akaofriends#카
3	jjun_ji	https://scc	2년만에우리오빠이렇게진심담아웃는거첨본다ㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋ평생잘해!!!사랑해예비남편#럽스타그램#부부스타그램#서프라이즈#이벤트#프로포즈#신혼집#아침프로포즈#꽃길#로지텍#로지
4	so_ye0n_0	https://scc	새로산키보드로일기쓰고오늘하루마무리#오늘하루도수고했어#애플#아이패드#아이패드프로#아이패드프로3세대#아이패드다꾸#애플팬슬#애플팬슬2세대#앱등이#앱등이의길#로지텍#로지텍k380#
5	sweetp_nu	https://scc	기계육심많은살암이라마우스와키보드만해도몇개씩갖고있음요즘은새로나온#제닉스#버티컬마우스쓰고있는데기존에쓰던#로지텍#mxmaster2s보다확실히더가볍다!로지텍쓰다가제닉스들면순간나도
6	lotx__	https://scc	#로지텍과#프라이탁ㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋ누구손이지
7	dndbajrwk	https://scc	#나를위한소비주로컴터로일하는나를위해손목보호마우스새로구매vv#버티컬마우스괜찮은거같아서로지텍껄로추가구매!손목보호패드도귀여운짱구로햇키보드는사는김에#지름신강림앞으로도열심히
8	mazect	https://scc	요즘내가사진작업할때주로사용하는키보드깔끔한디자인에부드러운키감이참좋다-동그랗게생긴녀석은크라운이라고하는데저다이얼을돌려자주사용하는기능들을간편하게사용할수있다!생생한사용장단
9	habi_snap	https://scc	아이패드미니를위한키보드두둥!점심먹고옆에이마트애플샵가서냉큼사왔다국민키보드크크크크좋구먼이건회사에두고맥이랑도연결해서써야지교교교#로지텍#380#국민키보드#블루투스키보드
10	best_t_soc	https://scc	아침에일어나서고시문쓰고우체국갔다가이제서야제대로#착석일단내해설서문제부터!#착석인증#스터디플래너#플래너#모트모트플래너#study#공스타그램#공부스타그램#공부인증#유아임용#유아임
11	davidthegi	https://scc	와내가이마우스를쓴지5년만에처음으로빼터리없다고붙들어옴빼터리도점살때들어있던거계속쓴건데여육시#로지텍#510빼터리라이프죽이네이번에빼터리갈아끼면또어나덜5년갈까나??ㅎㅎㅎ
12	aaanong	https://scc	새세팅완료!새로운작업실새로운사람들새로운환경과새로운일상안수호어린이집입학-7#브라보텍디파이b40#라이젠2700x#로지텍#lg4k
13	best_t_soc	https://scc	계획이틀어지는바람에올클못했다오늘갑자기동기부여가팍팍내일더열심히달려보자#내일은해설서문제푸는날#장마비예상#스터디플래너#플래너#모트모트플래너#study#공스타그램#공부스타그램#:
14	koo_9_che	https://scc	컴퓨터도착이제피시방안가요#게이밍컴퓨터#로지텍#키보드#마우스#헤드셋#오버워치#배틀그라운드#ram16g#geforce10606gb#ssd240g#i56600#영상속노래는#여자친구#교차로형키보드는다시사
15	flowermin	https://scc	바지락술찜면겨울시즌한정이래요진짜바지락이든라면이라니!#gs25편의점#gs25추천#바지락술찜면#진짜바지락#신기#신기방기#겨울#시즌한정#리미티드에디션#삼양라면#라면의원조#라면끝판왕
16	j.stagram1	https://scc	#꿀자료가수북하다고좋은거아니지?알맹이가알찬게중요하지!!!저도처음엔자료가수두룩그게나만알기쉽겠다는생각이들면서매번자료정리를하고필수요소만핵심만추려서이해하기쉽게자료를만드는중-
17	no.1_assac	https://scc	httpwwwassacomcom주문정보20021456141감랑고객님제품가격1787110원제작담당홍석호동일사양바로구매하기httpwwwassacomcomshopproductautomakesystemmn2abog33제품정보[#인텔코어
18	no.1_assac	https://scc	httpwwwassacomcom주문정보20021635691이병고객님제품가격981700원제작담당김정훈동일사양바로구매하기httpwwwassacomcomshopproductautomakesystemmnmymg33제품정보[#라이젠53
19	computer	https://scc	매장방문구매하여주신최석고객님로지텍ireless게이밍마우스httpcafenavercombusancomputerplaza부산컴퓨터프라자는게이밍기어체험형매장입니다#로지텍#무선마우스#지프로무선##마우스#부산
20	computer	https://scc	매장방문구매하여주신유욱고객님로지텍ireless게이밍마우스httpcafenavercombusancomputerplaza부산컴퓨터프라자는게이밍기어체험형매장입니다#로지텍#무선마우스#지프로무선##마우스#부산

라는 결과를 얻을 수 있다. 여기서, 활용한 것은 저번 2주차 동안의 목표였던 링크를 타서 본문을 가져오는 것과, 검색어를 입력하여 해당 검색어에 맞는 리뷰를 가져오는 것을 사용해보았으며, 추가적으로 csv파일로 저장하는 것까지 구현해보았다.

해당 주차를 크롤링하면서 생각한 부분은 페이스북은 리뷰를 분석하기에는 자료가 너무 적어서 의미가 없다는 점과 크롤링된 결과에서 필요한 키워드를 뽑아내는 것을 연구해야 할 것 같다. 따라서 자연어처리를 실습하는 것을 계획하고 있으며, 어떠한 DB를 사용할 지까지 고민해보고자 한다. 또한, 최근 1년 사이의 리뷰를 확인 기간을 정하고자 하는데 이는 리뉴얼된 제품 즉, 최신의 정보도 업데이트가 가능할 것이라고 판단하여 해당 기간으로 리뷰를 분석하고자 한다. 해당 기간 설정이 필요한 이유는 리뉴얼된 제품의 경우, 과거의 제품과는 다른 제품력을 보여주기 때문이다.

3. DB 구축

(진행도가 있을 때 추가 작성예정)

4. 단어 사용 빈도 추출

(진행도가 있을 때 추가 작성예정)

5. UI 제작

(진행도가 있을 때 추가 작성예정)

III. 과제 평가

1. 개선방안

- Request를 사용했을 때보다 selenium을 사용했을 때 동작에 있어 시간이 다소 오래걸리는 것을 확인했는데, html을 파싱하는데에 있어서 다소 문제가 있어서 selenium을 사용했던 것인데 이 파싱하는 부분을 바꿔서 코드를 수정해보고자 한다. 또한, 딕셔너리로 return을 받을 때 나중에 데이터를 사용할때는 반점과 괄호로 구분이 되어서 크게 상관은 없지만, 직접적인 view에 있어서는 다소 불편함이 있어서 중간중간 '₩n'가 삽입이 가능한지 확인해 보고자 한다.
- 이번 같은 경우에는 상품정보는 외국 몰에서, 리뷰는 영화로 대체를 했는데, 국내 쇼핑몰의 경우 오른쪽클릭이 안되어서 html소스를 못보는줄 알았는데, F12를 사용해서 할 수 있는 것을 확인하고 추가로 해보고자 한다. 특히, 이 경우 위에서처럼 제품정보를 얻으면서 링크를 구할 수 있고, 그 링크로부터 리뷰를 뽑아내는 것까지 해보고자 한다.
- 자연어처리 konlpy설치하는데에 있어서 pip upgrade 등의 시작부터 문제가 있는 듯하여 해결하고자 한다..

2. 기대효과

ㄱ. 기업적 측면

- 제품에 대한 긍정적인 리뷰를 많이 보이는 제품의 경우, 구매자로 하여금 다양한 웹사이트의 공통적인 리뷰를 확인가능함으로써 리뷰의 신뢰성이 높아져 구매력이 증가할 것이다.

- 해당 제품의 실제 이용고객이 제품을 사용하고 실제로 좋은 제품이라면 회사의 신뢰성이 증가하여 차기 출시될 제품의 신규 구매력도 증가할 것이다.

ㄴ. 사용자 측면

- 제품에 대한 리뷰를 알아보기 위해 번거롭게 여러 플랫폼을 일일이 찾아 검색하지 않아도 한번에 파악할 수 있다.

- 리뷰에서 자주 언급된 단어를 통해 중요 키워드를 산출해내기 때문에 신뢰가 가는 제품인지 쉽게 파악할 수 있다.

- 비슷한 조건의 다른 좋은 선택지가 없는지 한 눈에 알아볼 수 있다.

참고문헌

- 파이썬을 활용한 클로러 개발과 스크레이핑 입문, 2019, 카토 카츠야, 요코야마 유우키, 위키북스
- 파이썬 데이터 수집 자동화 한방에 끝내기 한입에 웹크롤링, 2018, 김경록, 서영덕, 비제이퍼블릭
- 파이썬을 이용한 웹크롤링과 스크레이핑, 2018, 카토 코타, 위키북스
- 파이썬을 이용한 머신러닝, 딥러닝 실전 개발 입문, 2019, 쿠지라 히코우즈쿠에, 위키북스
- Web Scraping with Python, 2019, 라이언미첼, 한빛미디어
- https://www.crummy.com/software/BeautifulSoup/bs4/doc.ko/