# -\*- coding: utf-8 -\*-

# from konlpy.tag import Okt

from ckonlpy.tag import Twitter

import pickle

from collections import Counter

import requests

import csv

import os

import re

from bs4 import BeautifulSoup

import pandas as pd

from datetime import datetime, date

from dateutil.relativedelta import relativedelta

from wordcloud import WordCloud

import warnings

import time

warnings.filterwarnings("ignore")

now = datetime.now()

todaydate = now.strftime('%Y-%m-%d')

todaydate = todaydate.replace('-', '.')

checkdate = (now - relativedelta(days=1)).strftime('%Y-%m-%d')

checkdate = checkdate.replace('-', '.')

fl = open('fixed\_korean\_stopwords.csv', 'r', encoding='UTF8')

files = csv.reader(fl)

stopwords = []

for file in files:

stopwords.extend(file)

stopwords = set(stopwords)

# item = ['대웅제약', '한미약품', '유한양행', '종근당', '우루사', '임팩타민', '베아제']

item = ['대웅제약', '한미약품', '유한양행', '종근당']

max = 67

address = 'C:/Users/User/Desktop/wordcloud/ver4/data/'

def kpa\_crawler(maxpage, query):

# print("<%s>" % query)

page = 1

total = []

while page < maxpage:

url = 'https://www.kpanews.co.kr/article/list.asp?page=' + str(page) + '&search\_word=' + query

req = requests.get(url)

cont = req.content

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

for urls in soup.select("div.lst\_article1 > ul > li > a"):

try:

news\_detail = get\_article('https://www.kpanews.co.kr/article/'+urls['href'], query)

total.append(news\_detail)

except Exception as e:

# print(e)

time.sleep(5)

continue

page += 1

print('%d / %d ' % (page, maxpage))

return total

def get\_article(n\_url, it):

news\_detail = ['', '', '', '', '']

breq = requests.get(n\_url)

bsoup = BeautifulSoup(breq.content, 'html.parser')

news\_detail[0] = '약사공론'

news\_detail[1] = n\_url

title = bsoup.select('div.inr-c2 > p.h1')[0].get\_text().replace('\n', '')

try:

title = title.encode('latin-1').decode('cp949','ignore')

except Exception as e:

None

news\_detail[2] = title

text = bsoup.select("div.bbs\_cont.inr-c2 > div")[0].get\_text()

text = re.sub(r"\[[^)]\*\]", '', text).replace('\t', '').replace('\n', '').strip()

try:

text = text.encode('latin-1').decode('cp949','ignore')

except Exception as e:

None

news\_detail[3] = text

date = bsoup.select('p.t1.t\_date > span')[0].get\_text()[:10].replace('-', '.')

news\_detail[4] = date

return news\_detail

def excel\_make(it, results):

df = pd.DataFrame(results, columns=['press', 'link', 'title', 'article', 'date'])

results = df.drop\_duplicates('link', keep='last')

try:

results.to\_csv('%s%s\_kparesult.csv' % (address, it), index=False, encoding='cp949')

except:

results.to\_csv('%s%s\_kparesult.csv' % (address, it), index=False)

def today(query):

# print("kpa <%s>" % query)

page = 1

check = False

total = []

while page < 2:

url = 'https://www.kpanews.co.kr/article/list.asp?page=' + str(page) + '&search\_word=' + query

req = requests.get(url)

cont = req.content

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

for urls in soup.select("div.lst\_article1 > ul > li > a"):

try:

news\_detail = get\_article('https://www.kpanews.co.kr/article/'+urls['href'], query)

if news\_detail[4] not in [todaydate, checkdate]:

check = True

break

# print(news\_detail)

total.append(news\_detail)

except Exception as e:

# print(e)

time.sleep(5)

continue

page += 1

if check:

break

return total

def addnews(it):

try:

df = pd.read\_csv('%s%s\_kparesult.csv' % (address, it),encoding='cp949')

except:

df = pd.read\_csv('%s%s\_kparesult.csv' % (address, it))

results = pd.DataFrame(today(it), columns=['press', 'link', 'title', 'article', 'date'])

total\_results = pd.concat([results, df])

total\_results = total\_results.drop\_duplicates('link', keep='last')

try:

total\_results.to\_csv('%s%s\_kparesult.csv' % (address, it), index=False, encoding='cp949')

except:

total\_results.to\_csv('%s%s\_kparesult.csv' % (address, it), index=False)

def crawl():

for it in item:

filename = '%s%s\_kparesult.csv' % (address, it)

if os.path.isfile(filename):

if (datetime.fromtimestamp(os.path.getmtime('%s%s\_naverresult.csv' % (address, it))).strftime('%G%m%d')) == date.today().strftime('%G%m%d'):

continue

addnews(it)

else:

data = kpa\_crawler(max, it)

excel\_make(it, data)

crawl()