Module 2: Reddit & Bing Search APIs

In this demo I will demonstrate how to utilize the reddit API and Bing Search to pull news articles and posts as a source of external data.

First, I will show how to create a Reddit personal use script for accessing the Reddit API. This will require having a reddit account, if you don't have one, follow along using the provided excel file.

Then, we will all create a university account on Azure, and then create a Bing Search resource to access the Bing Search API.

Use this link to create a personal use script for the Reddit API Click Here

Load in Dependcies, pip install praw

```
!pip install praw
import praw

from datetime import datetime
from datetime import date

import pandas as pd
import re
import string
from google.colab import userdata
```

```
Requirement already satisfied: praw in /usr/local/lib/python3.10/dist-packages (7.7.1)
Requirement already satisfied: prawcore<3,>=2.1 in /usr/local/lib/python3.10/dist-packag
Requirement already satisfied: update-checker>=0.18 in /usr/local/lib/python3.10/dist-pa
Requirement already satisfied: websocket-client>=0.54.0 in /usr/local/lib/python3.10/dis
Requirement already satisfied: requests<3.0,>=2.6.0 in /usr/local/lib/python3.10/dist-pa
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dis
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-pack
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-pack
```

Specify Reddit credentials and subreddits to be scraped

```
# Create a Reddit instance
reddit = praw.Reddit(client_id='4aygbvUFqWfGAllqJemgvQ',
                     client_secret=userdata.get('client_secret'),
                     user_agent='reddit_app/v1')


# not a secure way to store credentials, consider using a separate file, creating environmer


# Specify the subreddit names you want to retrieve posts from
subreddit_names = ['powsurf', 'gxor', 'exmormon', 'datascience']
```

Pull in selected Post Attributes, store and convert to dataframe

```
# create an empty post_attributes list
post_attributes = []

for subreddit_name in subreddit_names:
    subreddit = reddit.subreddit(subreddit_name) # set subreddits
    posts = subreddit.top(time_filter='month', limit=20) # set post parameters

    for post in posts: # pull in the following post attributes
        post_attributes.append({
            'Title': post.title,
            'Content': post.selftext,
            'URL': post.url,
            'Date': datetime.utcfromtimestamp(post.created_utc).strftime('%Y-%m-%d'),
            'Provider': subreddit_name
        })

df_red = pd.DataFrame(post_attributes) # create dataframe
df_red['All_Text'] = df_red['Title'] + ' ' + df_red['Content'] # create all_text column
df_red['Provider'] = 'r/' + df_red['Provider'] # create provider column
df_red.head()
```

```
WARNING:praw:It appears that you are using PRAW in an asynchronous environment.
It is strongly recommended to use Async PRAW: https://asyncpraw.readthedocs.io.
See https://praw.readthedocs.io/en/latest/getting_started/multiple_instances.html#discor

WARNING:praw:It appears that you are using PRAW in an asynchronous environment.
It is strongly recommended to use Async PRAW: https://asyncpraw.readthedocs.io.
See https://praw.readthedocs.io/en/latest/getting_started/multiple_instances.html#discor

WARNING:praw:It appears that you are using PRAW in an asynchronous environment.
It is strongly recommended to use Async PRAW: https://asyncpraw.readthedocs.io.
See https://praw.readthedocs.io/en/latest/getting_started/multiple_instances.html#discor

WARNING:praw:It appears that you are using PRAW in an asynchronous environment.
It is strongly recommended to use Async PRAW: https://asyncpraw.readthedocs.io.
See https://praw.readthedocs.io/en/latest/getting_started/multiple_instances.html#discor
```

	Title	Content	URL	Date	Provider	All_
0	Got out for a little urban powsurf in my city		https://v.redd.it/m0yfs73e52bc1	2024-01-07	r/powsurf	Gc for a u pov in my
1	POW TIME		https://v.redd.it/rpn97sgteucc1	2024-01-16	r/powsurf	F T
2	Getting some Midwest	Found this community and picked up	https://v.redd.it/y4t90y3hjtdc1	2024-01-21	r/powsurf	Ge s Mid turr

```
# df_red = pd.read_excel('Reddit_posts.xlsx')
# df_red.head()
```

Clean Data Function (not in template??)

```
# from text corpora notebook
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.probability import FreqDist
nltk.download('punkt')
nltk.download('stopwords')

def clean_text(text):
    # cleaned_text = BeautifulSoup(text, 'html.parser').get_text()
    if not isinstance(text, str):
        # Convert non-string types to string
        text = str(text)
    text = re.sub(r'[^a-zA-Z0-9\s]', '', text)
    text = text.lower() # lowercase
    text = text.strip() # remove extra whitespaces

    stop_words = set(stopwords.words("english")) # bring in stopwords
    tokens = nltk.word_tokenize(text) # tokenize
    cleaned_tokens = [token for token in tokens if token not in stop_words] # remove stopwor

    cleaned_text = ' '.join(cleaned_tokens) #rejoin tokens
    print(cleaned_text)
    return cleaned_text

df_red['Clean_All'] = df_red['All_Text'].apply(clean_text)
df_red.head()
```

```
fun
got high im mormon backrooms
time come last shelf broke 2015 kept secret wife 2017 mixed faith marriage since deci
mom started crying today might drink coffee work starbucks told love lattes guess ass
ironic
dad wants fail school decision made literally 8 years old love dad stop even say
random man walmart left church 19 years ago started new college wanted get coffee mak
hits way close home
first like divorces must final
florida man book mormon saw rexjw thought itd fun
book mormon translation
craziest mormon belief ever heard dad seriously believes eve sinnedher clit inside va
915am unexpectedly heard knock missionary apartment door opened find mission presider
wyoming hemorrhaging missionaries ward council bishop asked keep mind support missior
greatest data science achievement
5 years rdatascience salaries broken yoe degree
anyone know good titanic datasets ive looking datasets related titanic particularly w
realized dont know python thinking fairly good work ds people work python masters lec
pre screening assessments getting insane data scientist industry applied job data sci
market tough us even recession guy masters 2 years work experience suffer much find j
fall analyticsengineering spectrum
official 2023 end year salary sharing thread official thread sharing current salaries
give worst read good quantify impactsavings resume tried much yes real savingsand muc
nonstupid people work edit thank insights learned many people totally fine things bre
kind data scientist demand 2024 looking insights skills learn order become data scier
ds actually dying ive heard multiple sentiments reddit irl ds dying field replaced ml
planning quit joined one big 4 8 months ago thought would good role data science posi
imposter syndrome data analyticsscience common im m27 currently senior data analyst p
data scientist ml engineer interview expectation 2024 interview process new graduate
hard truth artificial intelligence healthcare clinical effectiveness everything flash
normal spend day something doesnt work recently started coop large retail organizatic
love pythonam good dont understand sql hi need get better sql ive leetcode 15 months
probability reference book data science professionals hi id like rehash understanding
update half year first data science job title started first data science job seven mc
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

	Title	Content	URL	Date	Provider	A
0	Got out for a little urban powsurf in my city		https://v.redd.it/m0yfs73e52bc1	2024-01-07	r/powsurf	f i
1	POW TIME		https://v.redd.it/rpn97sgteucc1	2024-01-16	r/powsurf	
2	Getting some Midwest turns in!	Found this community and picked up a board a f...	https://v.redd.it/y4t90y3hjtdc1	2024-01-21	r/powsurf	

Filter dataframe to external URLs

```
filtered_df = df_red[~(df_red['URL'].str.startswith('https://www.reddit.com') | df_red['URL'
filtered_df.head()
```

	Title	Content	URL	Date	Provider	All_Text	Clean_
0	Got out for a little urban powsurf in my city		https://v.redd.it/m0yfs73e52bc1	2024-01-07	r/powsurf	Got out for a little urban powsurf in my city	got ur pow
1	POW TIME		https://v.redd.it/rpn97sgteucc1	2024-01-16	r/powsurf	POW TIME	pow t
2	Getting some Midwest turns in!	Found this community and picked up a board a f...	https://v.redd.it/y4t90y3hjtdc1	2024-01-21	r/powsurf	Getting some Midwest turns in! Found this	get midv tu fo commu picked

```
filtered_df['URL'][21]
```

```
'https://i.redd.it/rfzcnhlthtbc1.jpeg'
```

What is the purpose of filtering out internal URLs? When might you want to use straight reddit posts vs reddit posts linking to external sources?

In order to find news via reddit. If you looked at internal URLs they would not lead to news, rather just more reddit posts.

Bing Search API

Next, we will use Microsoft Azure to create a Bing Search Resource to Access the Bing Search API. Why not Google? Google got rid of their Google News Search API so Bing is what we've got!

Being by going to the Azure Portal and creating a university account, then we will walk through the steps of creating a Bing Search resource and storing our secret keys within an environment variable.

```
import json
import os
from pprint import pprint
import requests
```

Set up Bing Credentials & Specify Search Query

```
# Set subscription key and endpoint variables
subscription_key = userdata.get('bing_secret')
endpoint = 'https://api.bing.microsoft.com/v7.0/news/search'

# Query term(s) to search for
query = "snowboard"
```

Set Bing Search Parameters

```
mkt = 'en-US'
params = { 'q': query,                                                  # Search query strir
           'mkt': mkt,                                                 # Only searching US cu
          'freshness': 'week',                                          # set freshness
          'count': 100}                                                 # max articles returne
headers = { 'Ocp-Apim-Subscription-Key': subscription_key}
```

Call to API, store info in JSON

```
# Call the API
try:
    response = requests.get(endpoint, headers=headers, params=params) # fill in parameters
    response.raise_for_status()


    json_data = response.json()


    with open('response.json', 'w', encoding='utf-8') as json_file:
        json.dump(json_data, json_file, ensure_ascii=False, indent=4)

    print("JSON response saved to response.json.")

except Exception as ex:
    raise ex
```

```
 JSON response saved to response.json.
```

Extract JSON response store in dataframe