**Experiment 3:**

**Topic: Select the social media platforms of your choice ( Twitter, Facebook, LinkedIn, Youtube,Web blogs etc), connect to and capture social media data for business.**

**Group no. 51**

**Data source: Reddit API wrapper**

**Dataset link:** [**https://github.com/Hiten-Dusseja/SMA\_Repo/blob/main/exp2\_gh/legal\_advice\_india\_all.json**](https://github.com/Hiten-Dusseja/SMA_Repo/blob/main/exp2_gh/legal_advice_india_all.json)

**Code:**

import praw

import json

from datetime import datetime

import re

reddit = praw.Reddit(

    client\_id="YkjnPbh7fE-NVhEMap86pw",

    client\_secret="kg0xkGmXuQmQZpo48cW00OkQ4nSxvQ",

    user\_agent="LegalAdviceIndiaBot by YOUR\_USERNAME"

)

subreddit = reddit.subreddit("LegalAdviceIndia")

def extract\_location(text):

    # Example list of Indian cities; expand this list as needed

    locations = ["Delhi", "Mumbai", "Bangalore", "Chennai", "Hyderabad", "Kolkata", "Pune", "Noida", "Gurgaon"]

    for location in locations:

        if location.lower() in text.lower():

            return location

    return "Unknown"

data = []

limit = 2000  # Number of posts to fetch

batch\_size = 100  # PRAW allows a maximum of 100 posts per request

count = 0

after = None  # Used to paginate backward

print(f"Fetching up to {limit} posts from r/LegalAdviceIndia...\n")

while count < limit:

    # Fetch a batch of posts

    posts = subreddit.new(limit=batch\_size, params={"after": after})

    batch\_data = []

    for post in posts:

        count += 1

        post\_data = {

            "title": post.title,

            "author": str(post.author),

            "url": post.url,

            "score": post.score,

            "created\_utc": post.created\_utc,

            "created\_date": datetime.utcfromtimestamp(post.created\_utc).strftime('%Y-%m-%d %H:%M:%S'),

            "num\_comments": post.num\_comments,

            "selftext": post.selftext,

            "id": post.id,

            "subreddit": str(post.subreddit),

            "location": extract\_location(post.selftext + " " + post.title),  # Combine title and selftext

        }

        batch\_data.append(post\_data)

        # Stop if we reach the limit

        if count >= limit:

            break

    data.extend(batch\_data)

    if batch\_data:

        after = batch\_data[-1]["id"]

    else:

        break

    print(f"Fetched {len(batch\_data)} posts. Total so far: {count}.\n")

file\_name = "legal\_advice\_india\_all.json"

with open(file\_name, "w", encoding="utf-8") as json\_file:

    json.dump(data, json\_file, ensure\_ascii=False, indent=4)

print(f"Data saved to '{file\_name}' with {len(data)} entries.")