Experiment No. 03

Semester	B.E. Semester VIII – Computer Engineering
Subject	Social Media Analysis Lab
Subject Professor In-charge	Prof. Amit Aylani
Academic Year	2024-25

Student Name	Deep Salunkhe
Roll Number	21102A0014

Title: Monitor the online reputation of a specific brand by collecting social media mentions, analyzing the sentiment of these mentions, and identifying the top influencers discussing the brand.

Description:

The primary goal of this experiment is to monitor the online reputation of a specific brand by collecting social media mentions, analyzing their sentiment, and identifying key influencers discussing the brand. This will help in understanding public perception, addressing negative feedback, and leveraging positive discussions for brand growth.

Methodology:

1. **Data Collection:**
   - Use web scraping and APIs (such as Twitter/X API, Facebook Graph API, and Reddit API) to collect social media mentions related to the brand.
   - Gather data including text, timestamps, user details, and engagement metrics (likes, shares, retweets, comments).
2. **Preprocessing:**
   - Remove stop words, emojis, special characters, and unnecessary metadata.
   - Standardize text by converting to lowercase and applying lemmatization or stemming.
3. **Sentiment Analysis:**
   - Apply NLP techniques to classify mentions as positive, negative, or neutral.
   - Use pre-trained models like VADER (for short social texts) or fine-tuned transformer models (BERT, RoBERTa) for sentiment classification.
4. **Influencer Identification:**
   - Rank users based on engagement metrics (follower count, retweets, mentions, replies).
   - Use network analysis to identify key opinion leaders discussing the brand.
5. **Insights and Visualization:**
   - Generate sentiment trend graphs to track reputation over time.
   - Create a network graph of influencers and their reach.

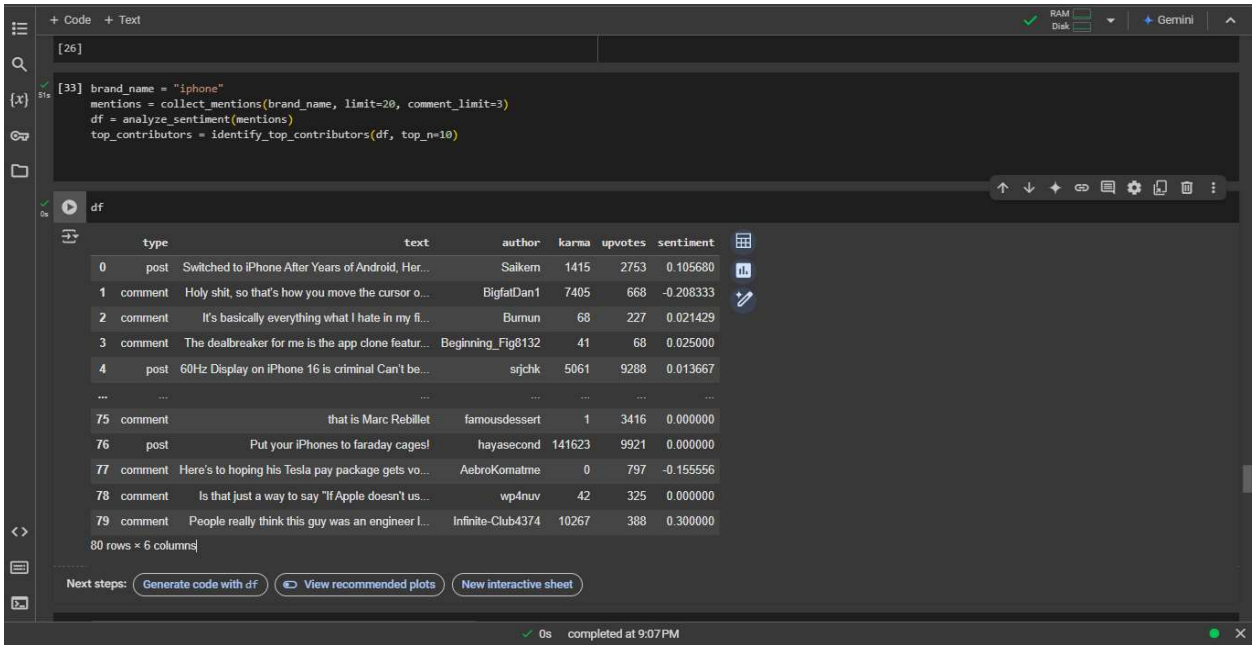
- Identify emerging patterns and topics associated with the brand.

Program Code:

```
!pip install asyncpraw

import praw
from textblob import TextBlob
import pandas as pd

CLIENT_ID = 'wR4s22ZsHO85tg5kvqpx7g'
CLIENT_SECRET = 'Ko7OcgyNlmVjupa-OlDaHbTmCwpURA'
USER_AGENT = 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36'

reddit = praw.Reddit(
  client_id=CLIENT_ID,
  client_secret=CLIENT_SECRET,
  user_agent=USER_AGENT,
  check_for_async=False
)

def collect_mentions(brand_name, limit=10, comment_limit=5):
  mentions = []
  for submission in reddit.subreddit("all").search(brand_name,
limit=limit):
    author_karma = getattr(submission.author, 'link_karma', 0)

    mentions.append({
      'type': 'post',
      'text': submission.title + " " + submission.selftext,
      'author': getattr(submission.author, 'name', 'Deleted'),
      'karma': author_karma,
      'upvotes': submission.score
    })

    submission.comments.replace_more(limit=0)
    for comment in submission.comments[:comment_limit]:
      mentions.append({
        'type': 'comment',
        'text': comment.body,
        'author': getattr(comment.author, 'name', 'Deleted'),
        'karma': getattr(comment.author, 'link_karma', 0),
        'upvotes': comment.score
      })
```

```
  return mentions

def analyze_sentiment(mentions):
  data = [{
    'type': mention['type'],
    'text': mention['text'],
    'author': mention['author'],
    'karma': mention['karma'],
    'upvotes': mention['upvotes'],
    'sentiment': TextBlob(mention['text']).sentiment.polarity
  } for mention in mentions]
  return pd.DataFrame(data)

def identify_top_contributors(df, top_n=5):
  return (df.groupby('author')
      .agg({'karma': 'max', 'upvotes': 'sum', 'sentiment': 'mean'})
      .reset_index()
      .sort_values(by=['upvotes', 'karma'], ascending=[False, False])
      .head(top_n))



brand_name = "iphone"
mentions = collect_mentions(brand_name, limit=20, comment_limit=3)
df = analyze_sentiment(mentions)
top_contributors = identify_top_contributors(df, top_n=10)


df

top_contributors[['author', 'karma', 'sentiment', 'upvotes']]


print("Average Sentiment Score:", df['sentiment'].mean())
```

Output:



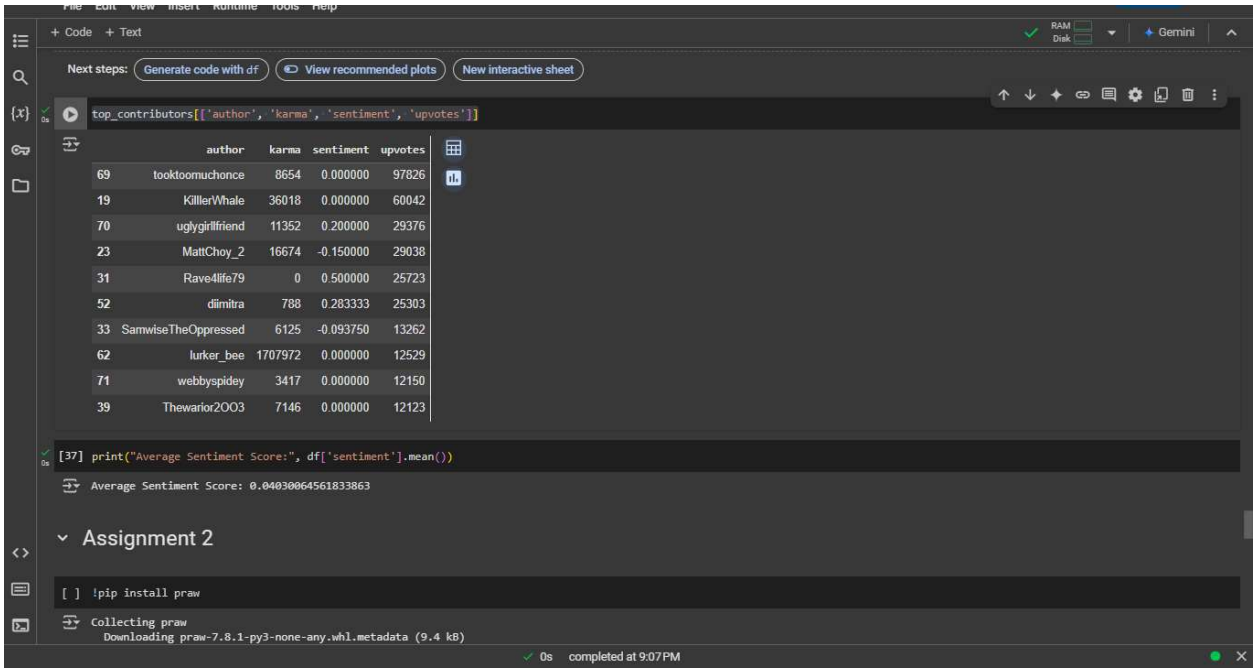
```
[26]

[33] brand_name = "iphone"
     mentions = collect_mentions(brand_name, limit=20, comment_limit=3)
     df = analyze_sentiment(mentions)
     top_contributors = identify_top_contributors(df, top_n=10)

df
```

	type	text	author	karma	upvotes	sentiment
0	post	Switched to iPhone After Years of Android, Her...	Saikern	1415	2753	0.105680
1	comment	Holy shit, so that's how you move the cursor o...	BigfatDan1	7405	668	-0.208333
2	comment	It's basically everything what I hate in my fi...	Burnun	68	227	0.021429
3	comment	The dealbreaker for me is the app clone featur...	Beginning_Fig8132	41	68	0.025000
4	post	60Hz Display on iPhone 16 is criminal Can't be...	srjchk	5061	9288	0.013667
...	...	...	...	...	...	...
75	comment	that is Marc Rebillet	famousdessert	1	3416	0.000000
76	post	Put your iPhones to faraday cages!	hayasecond	141623	9921	0.000000
77	comment	Here's to hoping his Tesla pay package gets vo...	AebroKomatme	0	797	-0.155556
78	comment	Is that just a way to say "If Apple doesn't us...	wp4nuv	42	325	0.000000
79	comment	People really think this guy was an engineer l...	Infinite-Club4374	10267	388	0.300000

80 rows × 6 columns

Next steps: Generate code with df | View recommended plots | New interactive sheet

0s completed at 9:07 PM



Next steps: Generate code with df | View recommended plots | New interactive sheet

```
top_contributors[['author', 'karma', 'sentiment', 'upvotes']]
```

	author	karma	sentiment	upvotes
69	tooktoomuchonce	8654	0.000000	97826
19	KilllerWhale	36018	0.000000	60042
70	uglygirllfriend	11352	0.200000	29376
23	MattChoy_2	16674	-0.150000	29038
31	Rave4life79	0	0.500000	25723
52	diimitra	788	0.283333	25303
33	SamwiseTheOppressed	6125	-0.093750	13262
62	lurker_bee	1707972	0.000000	12529
71	webbyspidey	3417	0.000000	12150
39	Thewarior2OO3	7146	0.000000	12123

```
[37] print("Average Sentiment Score:", df['sentiment'].mean())

Average Sentiment Score: 0.04030064561833863
```

Assignment 2

```
[ ] !pip install praw

Collecting praw
  Downloading praw-7.8.1-py3-none-any.whl.metadata (9.4 kB)
```

0s completed at 9:07 PM

Conclusion:

This experiment enables the brand to monitor its online reputation by collecting social media mentions and analyzing sentiment. By leveraging sentiment analysis, the brand can track public perception over time and address both positive and negative feedback. Identifying top influencers helps the brand engage with key opinion leaders, amplifying positive discussions and managing potential risks. The insights gained from this process allow for data-driven decision-making, improved brand image, and better consumer engagement. Continuous monitoring ensures timely responses to trends, fostering a stronger connection with the audience.