**A Program to request Twitter data**

def get\_tweets(api=None, screen\_name=None):

timeline = api.GetUserTimeline(screen\_name=screen\_name, count=200)

earliest\_tweet = min(timeline, key=lambda x: x.id).id

print("getting tweets before:", earliest\_tweet)

while True:

tweets = api.GetUserTimeline(

screen\_name=screen\_name, max\_id=earliest\_tweet, count=200

)

new\_earliest = min(tweets, key=lambda x: x.id).id

if not tweets or new\_earliest == earliest\_tweet:

break

else:

earliest\_tweet = new\_earliest

print("getting tweets before:", earliest\_tweet)

timeline += tweets

return timeline

if \_\_name\_\_ == "\_\_main\_\_":

api = twitter.Api(

CONSUMER\_KEY, CONSUMER\_SECRET, ACCESS\_TOKEN\_KEY, ACCESS\_TOKEN\_SECRET

)

screen\_name = sys.argv[1]

print(screen\_name)

timeline = get\_tweets(api=api, screen\_name=screen\_name)

with open('examples/timeline.json', 'w+') as f:

for tweet in timeline:

f.write(json.dumps(tweet.\_json))

f.write('\n')

**Use of Data from Twitter API**

I can use data from Twitter to compose tweets, read profiles, and access my follower’s data. I can also use high volume of tweets on specific subjects in different locations.

**Analysis of Twitter data**

In recent years, there has been an increase in interest in the field of Sentiment Analysis. Sentiment analysis includes what customer want (likes and dislikes). Programmers can use inbuilt sentiment analyser in R to calculate the presence of different emotions by checking corresponding valence in the text.

**Libraries needed for Twitter data**

Programmer can use the tweepy python library. This library makes it easy to connect and fetch data. This library can be installed by using the command *pip install tweepy, pip install -r requirements.txt*, and *pip install python-twitter*.

For the setup, programmer installs *python setup.py build and python setup.py install*