### **Setup for Using MongoDB (NoSQL) in the System**

#### **1. Install MongoDB**

#### **2. Install pymongo**

pip install pymongo

#### **3. Load Tweets Data into MongoDB**

import pandas as pd

from pymongo import MongoClient

df = pd.read\_csv('correct\_twitter\_201904.tsv', sep='\t')

client = MongoClient('mongodb://localhost:27017/')

db = client['twitter\_db']

collection = db['tweets']

data\_dict = df.to\_dict(orient='records')

collection.insert\_many(data\_dict)

print("Data inserted successfully into MongoDB")

### **4. Querying MongoDB using Python**

Now that the tweets are stored in MongoDB, you can update the Python functions to query the data from MongoDB instead of directly from a CSV/TSV file.

We will use the **pymongo** library to query the MongoDB collection.

### **Modified Python Functions to Query MongoDB**

Here are the updated functions to query tweets from MongoDB based on the term you're searching for:

#### **Tweets Per Day Containing the Term:**

from pymongo import MongoClient

from datetime import datetime

from collections import Counter

client = MongoClient('mongodb://localhost:27017/') # Replace with your MongoDB URI if using Atlas

db = client['twitter\_db']

collection = db['tweets']

def tweets\_per\_day(term):

query = {"text": {"$regex": term, "$options": "i"}}

tweets = collection.find(query, {"created\_at": 1})

tweet\_dates = [tweet['created\_at'][:10] for tweet in tweets] tweet\_count = Counter(tweet\_dates)

result = tweets\_per\_day("music")

print(result)

#### **Count Unique Users Posting the Term:**

def unique\_users\_posting\_term(term):

query = {"text": {"$regex": term, "$options": "i"}}

tweets = collection.find(query, {"author\_id": 1})

unique\_users = set(tweet['author\_id'] for tweet in tweets)

return len(unique\_users)

result = unique\_users\_posting\_term("music")

print(result)

#### **Average Likes for Tweets Containing the Term:**

def average\_likes\_for\_term(term):

query = {"text": {"$regex": term, "$options": "i"}}

tweets = collection.find(query, {"like\_count": 1})

total\_likes = 0

tweet\_count = 0

for tweet in tweets:

if 'like\_count' in tweet:

total\_likes += tweet['like\_count']

tweet\_count += 1

if tweet\_count == 0:

return 0 # Avoid division by zero

return total\_likes / tweet\_count

result = average\_likes\_for\_term("music")

print(f"Average likes: {result:.2f}")

### **5. Run the Updated Code**

Once you’ve updated the functions to work with MongoDB, you can run the code in the same way as before.

**Start MongoDB**  
sudo service mongod start

**Run the Python Script**:  
python twitter\_analysis\_mongo.py

You should see similar outputs but now powered by MongoDB queries instead of reading from a CSV file.