Streaming Twitter Data to MySQL Database

0.0.1 Streaming Twitter Data to MySQL Database

Twitter is one of the largest social media nowadays. And there are tons of information that can be collected if approached correctly. Unlike scraping other websites where hard-coding is necessary, Twitter provided Twitter API, making data scraping much more straightforward. This notebook searched Twitter containing specific keywords and stream them directly to the MySQL database.

[ ]:
```
import mysql.connector # no module named mysql: solution: conda install␣
 ↪mysql-connector-python
from mysql.connector import Error
import tweepy
import json
import re
```

0.0.2 Step 1

Connect to MySQL server and create a database named 'twitterdb'

[ ]:
```
mydb = mysql.connector.connect(
  host="localhost",
  user="root",
  password="your sql server passcode"
)
mycursor = mydb.cursor()
mycursor.execute("CREATE DATABASE IF NOT EXISTS twitterdb")
```

0.0.3 Step 2

Create a table in 'twitterdb' database to store streamed data.

Before creating a table in the database, it is essential to know that Twitter summarized all information related to a specific tweet into a JSON file. The JSON file is a list of key: value pairs that contain information about every field of that tweet, such as the tweet content, the time it was created, the user who posted that tweet, etc. (click the link for Twitter JSON example: https://www.sitepoint.com/twitter-json-example/).

I am only interested in twitterid, username, userid, description, created_at, tweet, place, and location in this project. Therefore, the table 'Tag' in the ''twitterdb' database will be created accordingly.

[ ]:
```
mydb = mysql.connector.connect(
  host="localhost",
  user="root",
  password="your sql server passcode",
  database="twitterdb"
)
cursor = mydb.cursor()
cursor.execute("CREATE TABLE Tag (twitterid BIGINT, username VARCHAR(15),userid␣
↪BIGINT, description VARCHAR(280), created_at VARCHAR(50), tweet VARCHAR(280),␣
↪place VARCHAR(30), location VARCHAR(30))")
```

0.0.4 Step 3

The following blocks define functions/class about extract tweets using Twitter API and extract info from JSON file then stream to MySQL database.

1. A function to insert twitter into database.

[ ]:
```
def connect(twitterid, username, userid, description, created_at, tweet, place,␣
↪location):
    """
    connect to MySQL database and insert twitter data
    """
    try:
        con = mysql.connector.connect(host = 'localhost',
        database='twitterdb', user='root', password = 'your sql server␣
↪passcode', charset = 'utf8')

        if con.is_connected():
            """
            Insert twitter data
            """
            cursor = con.cursor()
            # twitter, golf
            query = "INSERT INTO Tag (twitterid, username, userid, description,␣
↪created_at, tweet, place, location) VALUES (%s, %s, %s, %s, %s, %s, %s, %s)"
            cursor.execute(query, (twitterid, username, userid, description,␣
↪created_at, tweet, place, location))
            con.commit()

    except Error as e:
        print(e)

    cursor.close()
    con.close()

    return
```

2. A function to remove emojis

```
[ ]: def remove_emojis(data):
         emoj = re.compile("["
             u"\U0001F600-\U0001F64F"  # emoticons
             u"\U0001F300-\U0001F5FF"  # symbols & pictographs
             u"\U0001F680-\U0001F6FF"  # transport & map symbols
             u"\U0001F1E0-\U0001F1FF"  # flags (iOS)
             u"\U00002500-\U00002BEF"  # chinese char
             u"\U00002702-\U000027B0"
             u"\U00002702-\U000027B0"
             u"\U000024C2-\U0001F251"
             u"\U0001f926-\U0001f937"
             u"\U00010000-\U0010ffff"
             u"\u2640-\u2642"
             u"\u2600-\u2B55"
             u"\u200d"
             u"\u23cf"
             u"\u23e9"
             u"\u231a"
             u"\ufe0f"  # dingbats
             u"\u3030"
                           "]+", re.UNICODE)
         return re.sub(emoj, '', data)
```

3. Extract streamed tweets information then load to MySQL database.

```
[ ]: # Tweepy class to access Twitter API
     class Streamlistener(tweepy.StreamListener):

         def on_connect(self):
             print("You are connected to the Twitter API")

         def on_error(self):
             if status_code != 200:
                 print("error found")
                 # returning false disconnects the stream
                 return False

         def on_data(self,data):
             try:
                 raw_data = json.loads(data)

                 if 'text' in raw_data:

                     twitterid = raw_data['id']
                     username = remove_emojis(str(raw_data['user']['screen_name']))
                     userid = raw_data['user']['id']
```

```
                description = remove_emojis(str(raw_data['user']['description']))
                created_at = raw_data['created_at']
                tweet = remove_emojis(str(raw_data['text']))

                if raw_data['place'] is not None:
                    place = remove_emojis(str(raw_data['place']['country']))
                    #print(place)
                else:
                    place = None

                location = remove_emojis(str(raw_data['user']['location']))

                #insert data just collected into MySQL database
                connect(twitterid, username, userid, description, created_at,␣
↪tweet, place, location)
        except Error as e:
            print(e)
```

0.0.5 Step 4 Implement

API keys and access tokens To access Twitter data through Twitter API, a Twitter Developer account (https://developer.twitter.com) is needed. Once you got the developer account and created at least one App, **API key, API secret key, Access Token, Access token secret** could be found by: 'your app' –> 'Details' –> 'Keys and tokens'.

[ ]:
```
auth = tweepy.OAuthHandler("API key",
                           "API secret key")
auth.set_access_token("Access Token",
                      "Access token secret")
api = tweepy.API(auth, wait_on_rate_limit=True)
```

Stream Twitter to MySQL Once a tweet contains one of ['python', 'java', 'c++' ,'AnimalCrossing', 'PokenmonGo'], it will be collected and streamed to Tag tabel Immediately.

[ ]:
```
listener = Streamlistener(api = api)
stream = tweepy.Stream(auth, listener = listener)

track = ['python', 'java', 'c++' ,'AnimalCrossing', 'PokenmonGo']

# choose what we want to filter by
stream.filter(track = track, languages = ['en'])
```

[ ]:
```

```

[ ]:
```

```