2020-0701 IST652 Scripting for Data Analysis HW # 2

Sathish Kumar Rajendiran

Prasad Kulkarni

2020-08-21

Table of Contents

[Task Summary – Semi Structured Data Processing 2](#_Toc49189110)

[Data set creation using Twitter API 3](#_Toc49189111)

[Data Exploration 4](#_Toc49189112)

[Data Transformation 9](#_Toc49189113)

[Data Questions & Visualizations 11](#_Toc49189114)

[Summary 15](#_Toc49189115)

[Conclusion 16](#_Toc49189124)

[Contributions 16](#_Toc49189125)

# **Task Summary – Semi Structured Data Processing**

Write a program that reads JSON formatted data (ex. Twitter Streaming data) from MongoDB collection. Analyze the data as lists of JSON structures and load them into Pandas dataframes for further analysis. Tasks includes (but not limited to),

1. Dataset Creation using Twitter API
   1. Setup developer account
   2. Authentication
   3. Define list of search keywords
2. Data Exploration
   1. Authentication into MongoDB & Objects creation
   2. Tweepy stream data into MongoDB
   3. Data Description
   4. Analyze JSON formatted collection
      1. MongoDB Compass
      2. Python
   5. Pandas dataframes for processed data
3. Data Transformation
   1. Data type conversion
   2. Derive calendar items
   3. Remove redundant or unnecessary columns
   4. Remove special characters
   5. Re-arrange columns
   6. Write to file
4. Data Questions & Visualization
   1. Analyze twitter trends
      1. Top 10 users by followers
      2. Ratio of percentage of Retweets vs Actual tweets
      3. Top 10 retweets
      4. Number of Tweets by day, hour of day, by month
   2. Word cloud
5. Summary
   1. What’s the story(stories) in this data?
6. Conclusion
7. Contribution
   1. Task distribution

# **Data set creation using Twitter API**

In order to perform tweets analysis, first step is to setup a developer account with Twitter. It involves the following steps,

1. Setup developer account
2. Create developer account via <https://developer.twitter.com/en>
3. Complete access request form by submitting a justification on why access being requested. Sample quote below,

“Master’s in applied data science student of Prof. xxx xxx from Syracuse university is exploring semi structured data to analyze twitter feeds. Hence, access to twitter API is requested.”

1. Once developer account is activated, create an app and generate authentication token. This includes 4 key values,
   1. Consumer Key
   2. Consumer Secret
   3. Access Secret
   4. Access token

A screenshot of a cell phone

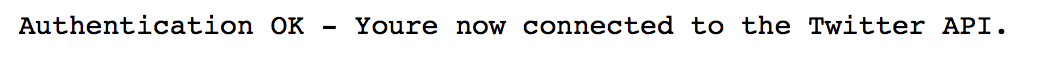
Description automatically generated

A screenshot of a cell phone

Description automatically generated

1. Authentication

Lets, authenticate through Jupyter Notebook using the Tweepy library from python and TwitterAPI keys. Credentials are read through an excel file (standard secret management) in the code.



1. Define list of search keywords

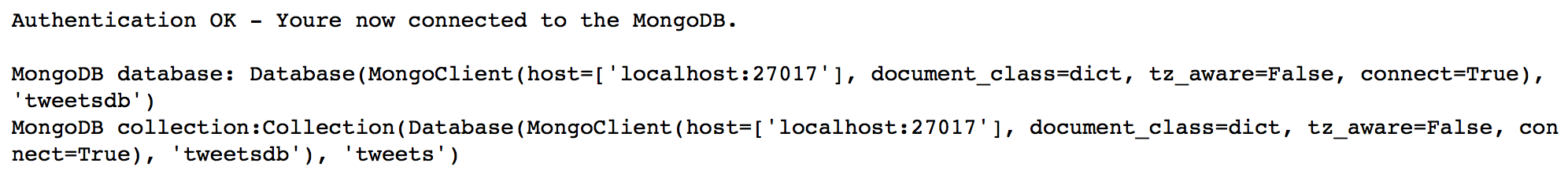
Let’s define a list of keywords as “'#LA’, ‘#LosAngeles’, ‘#LAtraffic', '#LAFD', '#LASTREETCLOSURE'” to fetch tweets. Other parameters including language as “en”

# **Data Exploration**

Next step is to stream twitter data into Mongo DB – a NoSQL database for semi structured (documents) data. Data exploration include the following steps,

1. Authentication into MongoDB & Objects creation

Initialize connection to Mongo DB local instance running on port number 27017. In addition, create “tweetsdb” and “tweets” collection object as well. This process will establish connection and create the databases objects if doesn’t exist.



1. Tweepy stream data into MongoDB

Now that, Connections to Twitter API and Mongo DB are established; next step is to setup streaming listener and start collecting tweets into Mongo DB.

This step includes, connecting to twitter feed and collects JSON formatted data and inserts into MongoDB collection “tweets”

A screenshot of a cell phone

Description automatically generated

1. Analyze JSON formatted collection

Let’s review the tweets from MongoDB collection. It has over 277k tweets collected.

* + 1. ipython Jupyter Notebook - First tweet from the collection as below,

{'\_id': ObjectId('5f39fe371a3bc61df03bfac4'),

'created\_at': 'Mon Aug 17 03:49:06 +0000 2020',

'id': 1295205989478731781,

'id\_str': '1295205989478731781',

'text': 'Hurricane Awareness: Zephyr Insurance \n\nREAD MORE: [https://t.co/SEhxK4bwyi\n\n#Accidents](https://t.co/SEhxK4bwyi/n/n#Accidents) #Claims #DisasterMitigation… <https://t.co/ew7ylyeHCz',>

'display\_text\_range': [0, 140],

'source': '<a href="https://www.blog.iammarketingmedia.com" rel="nofollow">IAMBLOG2TWITTER</a>',

'truncated': True,

'in\_reply\_to\_status\_id': None,

'in\_reply\_to\_status\_id\_str': None,

'in\_reply\_to\_user\_id': None,

'in\_reply\_to\_user\_id\_str': None,

'in\_reply\_to\_screen\_name': None,

'user': {'id': 226310002,

'id\_str': '226310002',

'name': 'IAM Platform',

'screen\_name': 'IAM\_\_Network',

'location': 'Worldwide',

'url': 'https://www.iammarketingmedia.com',

'description': 'Curation | Tools | Tips | Services\n\nIAM Platform powers IAM Network:\n\nGO: [http://bit.ly/2Ywsbg8\n\nBlog](http://bit.ly/2Ywsbg8/n/nBlog) | Social | Podcast | Code Trove',

'translator\_type': 'none',

'protected': False,

'verified': False,

'followers\_count': 18016,

'friends\_count': 14938,

'listed\_count': 3290,

'favourites\_count': 65467,

'statuses\_count': 778665,

'created\_at': 'Mon Dec 13 21:24:29 +0000 2010',

'utc\_offset': None,

'time\_zone': None,

'geo\_enabled': False,

'lang': None,

'contributors\_enabled': False,

'is\_translator': False,

'profile\_background\_color': '94D487',

'profile\_background\_image\_url': 'http://abs.twimg.com/images/themes/theme1/bg.png',

'profile\_background\_image\_url\_https': 'https://abs.twimg.com/images/themes/theme1/bg.png',

'profile\_background\_tile': False,

'profile\_link\_color': '3366CC',

'profile\_sidebar\_border\_color': 'FFFFFF',

'profile\_sidebar\_fill\_color': 'DDEEF6',

'profile\_text\_color': '333333',

'profile\_use\_background\_image': True,

'profile\_image\_url': 'http://pbs.twimg.com/profile\_images/701708113653669888/Nzm67hhC\_normal.png',

'profile\_image\_url\_https': 'https://pbs.twimg.com/profile\_images/701708113653669888/Nzm67hhC\_normal.png',

'profile\_banner\_url': 'https://pbs.twimg.com/profile\_banners/226310002/1584072260',

'default\_profile': False,

'default\_profile\_image': False,

'following': None,

'follow\_request\_sent': None,

'notifications': None},

'geo': None,

'coordinates': None,

'place': None,

'contributors': None,

'is\_quote\_status': False,

'extended\_tweet': {'full\_text': 'Hurricane Awareness: Zephyr Insurance \n\nREAD MORE: [https://t.co/SEhxK4bwyi\n\n#Accidents](https://t.co/SEhxK4bwyi/n/n#Accidents) #Claims #DisasterMitigation #Insurance #InsuranceTechnology #InsurTech #Points #RiskMitigation #Technology~ <https://t.co/kCrl2YxHfK',>

'display\_text\_range': [0, 194],

'entities': {'hashtags': [{'text': 'Accidents', 'indices': [76, 86]},

{'text': 'Claims', 'indices': [87, 94]},

{'text': 'DisasterMitigation', 'indices': [95, 114]},

{'text': 'Insurance', 'indices': [115, 125]},

{'text': 'InsuranceTechnology', 'indices': [126, 146]},

{'text': 'InsurTech', 'indices': [147, 157]},

{'text': 'Points', 'indices': [158, 165]},

{'text': 'RiskMitigation', 'indices': [166, 181]},

{'text': 'Technology', 'indices': [182, 193]}],

'urls': [{'url': 'https://t.co/SEhxK4bwyi',

'expanded\_url': 'https://blog.iammarketingmedia.com/hurricane-awareness-zephyr-insurance/?utm\_campaign=twitter&utm\_medium=twitter&utm\_source=twitter',

'display\_url': 'blog.iammarketingmedia.com/hurricane-awar…',

'indices': [51, 74]}],

'user\_mentions': [],

'symbols': [],

'media': [{'id': 1295205987209621505,

'id\_str': '1295205987209621505',

'indices': [195, 218],

'media\_url': 'http://pbs.twimg.com/media/Efl--6qWsAEpwkT.jpg',

'media\_url\_https': 'https://pbs.twimg.com/media/Efl--6qWsAEpwkT.jpg',

'url': 'https://t.co/kCrl2YxHfK',

'display\_url': 'pic.twitter.com/kCrl2YxHfK',

'expanded\_url': 'https://twitter.com/IAM\_\_Network/status/1295205989478731781/photo/1',

'type': 'photo',

'sizes': {'small': {'w': 448, 'h': 252, 'resize': 'fit'},

'thumb': {'w': 150, 'h': 150, 'resize': 'crop'},

'medium': {'w': 448, 'h': 252, 'resize': 'fit'},

'large': {'w': 448, 'h': 252, 'resize': 'fit'}}}]},

'extended\_entities': {'media': [{'id': 1295205987209621505,

'id\_str': '1295205987209621505',

'indices': [195, 218],

'media\_url': 'http://pbs.twimg.com/media/Efl--6qWsAEpwkT.jpg',

'media\_url\_https': 'https://pbs.twimg.com/media/Efl--6qWsAEpwkT.jpg',

'url': 'https://t.co/kCrl2YxHfK',

'display\_url': 'pic.twitter.com/kCrl2YxHfK',

'expanded\_url': 'https://twitter.com/IAM\_\_Network/status/1295205989478731781/photo/1',

'type': 'photo',

'sizes': {'small': {'w': 448, 'h': 252, 'resize': 'fit'},

'thumb': {'w': 150, 'h': 150, 'resize': 'crop'},

'medium': {'w': 448, 'h': 252, 'resize': 'fit'},

'large': {'w': 448, 'h': 252, 'resize': 'fit'}}}]}},

'quote\_count': 0,

'reply\_count': 0,

'retweet\_count': 0,

'favorite\_count': 0,

'entities': {'hashtags': [{'text': 'Accidents', 'indices': [76, 86]},

{'text': 'Claims', 'indices': [87, 94]},

{'text': 'DisasterMitigation', 'indices': [95, 114]}],

'urls': [{'url': 'https://t.co/SEhxK4bwyi',

'expanded\_url': 'https://blog.iammarketingmedia.com/hurricane-awareness-zephyr-insurance/?utm\_campaign=twitter&utm\_medium=twitter&utm\_source=twitter',

'display\_url': 'blog.iammarketingmedia.com/hurricane-awar…',

'indices': [51, 74]},

{'url': 'https://t.co/ew7ylyeHCz',

'expanded\_url': 'https://twitter.com/i/web/status/1295205989478731781',

'display\_url': 'twitter.com/i/web/status/1…',

'indices': [116, 139]}],

'user\_mentions': [],

'symbols': []},

'favorited': False,

'retweeted': False,

'possibly\_sensitive': False,

'filter\_level': 'low',

'lang': 'en',

'timestamp\_ms': '1597636146312'}

* + 1. MongoDB Compass – view the collection from MongoDB UI Interface (Compass). This interface provides better view of the JSON formatted data in both tabular and JSON format.

A screenshot of a cell phone

Description automatically generated A screenshot of text

Description automatically generated

1. Data Description: Find the data description of the fields that are in scope here

A screenshot of text

Description automatically generated

1. Pandas dataframes for processed data

Now, let’s collect few selected fields into a list and load it into pandas dataframes for further analysis. Fields in scope are id, id\_str, username, source, followers\_count, retweets\_count, coordinates, place, full\_name, text, profile\_background\_color and possibly\_sensitive are collected through tw\_list [] as below. Later, data from tw\_list [] is loaded into “tweetsDF” pandas dataframe.

tweetsDF.info displays more information about the dataframe including Number of rows as in index range, missing values, column name, datatype and total memory allocated.

A screenshot of a cell phone

Description automatically generated

tweetsDF.head() displays top 5 rows from the dataframe as in rows and columns.

A screenshot of a social media post

Description automatically generated

# **Data Transformation**

Preprocessing complete. Next step is to prepare the data for further analysis. It involves, cleaning the data, data type conversion, removal of unwanted columns, removal of special characters, creation of new columns based on calendar date time, re-arranging the column sequence and exporting it to csv file.

1. Data type conversion & Derive calendar items

Let’s convert “datets” field to datetime datatype from defaulted “object” datatype and create additional columns from this field such as Date, year, Month, Monthday, weekday, monthname, hour, minute and seconds for tweets analysis at different calendar date/time schedules.

A screenshot of a cell phone

Description automatically generated

1. Remove special characters

Next step is to cleanup “source” column by removing all the special characters and retain only the platform detail into a new column.

A screenshot of a cell phone

Description automatically generated

1. Remove redundant or unnecessary columns

Columns “id\_str”, “source”, “retweets” are removed as no longer needed as they seem redundant.

1. Preprocessing API is used to clean Text and userName fields with special characters for tokenization.
2. Re-arrange the columns

Let’s rearrange the columns for ease of analysis.

A screenshot of a social media post

Description automatically generated

1. Write to file – export the file into CSV for validation

A screenshot of a computer

Description automatically generated

# **Data Questions & Visualizations**

Let’s explore twitter feeds further; For the analysis only focus on text containing “las” string.

A screenshot of a cell phone

Description automatically generated

1. Analyze twitter trends
   1. Top 10 users by followers – from the above dataframe lets find out top 10 users by their number of followers.

A screenshot of a cell phone

Description automatically generated

* 1. Find min, max, average number of followers

A screenshot of a cell phone

Description automatically generated

* 1. Ratio of percentage of Retweets vs Actual tweets

From the revised dataframe, lets find out ratio of retweets vs direct tweets

A picture containing table

Description automatically generated

* 1. Top 10 tweets by platform

A screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generated

A close up of a card

Description automatically generated

* 1. Number of Tweets by day, hour of day, by month

A screenshot of a social media post

Description automatically generated

A close up of a map

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A picture containing boat, different, group, filled

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Word cloud – Lastly, a Wordcloud on twitter feeds to conclude the trending words on the key words including #latraffic, #lapd, #lafd, #losangeles city – designed in a butterfly shape. It also excluded few common stop words like “https”,” rt”,” hi”,” promo”,” thank” etc.

**A close up of a piece of paper

Description automatically generated**

# **Summary**

After extracting data via Twitter API, following observations were made.

* Data is in JSON format.
* Tweepy streaming API pulled more than 270k tweets
* MongoDB datastore used to collect all the tweets
* Pandas dataframes used for data transformation, processing and answering data questions with simple plots (bar, line graph and pie chart)
* Simple Word cloud representation made in butterfly shape
* Analysis including
  + tweets filtered on keywords such as #losangeles, #latraffic, #lapd, #lafd etc.
  + 45% of the tweets were re-tweets
  + 38% of tweets were posted through Instagram followed by iPhone with 22%
  + Monday 6 AM showed the least number of tweets where as 8PM on Monday had the highest tweets traffic. Monday remained a busy day with more tweets by day, followed by Sunday.
  + CareersInGovernment user remained top with number of active followers; followed by LAFD talk
  + Average followers count is ~3000
  + Wordcloud highlighted some of the top words as “outpouring”, “heavy hearts”, “support”, “jose perez” , “fire station” etc. – highlighting the outpouring support from users on twitter in support of “Jose Perez- a fire fighter who had lost his life due to COVID 19”

<https://abc30.com/coronavirus-deaths-firefighter-death-covid-19/6338926/>

# **Conclusion**

Overall, an amazing exercise to explore the power of twitter streaming api, how to use MongoDB to store and view JSON- Semi structured data collections effortlessly, Power of Pandas dataframe to perform data cleaning/transformation, analysis tasks with visualization options including plots/graphs/charts to answer any meaningful data questions.

Lastly, with over 1.3 billion users – twitter feeds can be considered as gold mine of data that can be used effectively to get insights of the general public on any topic that is trending. Based on the public’s reaction to topics/products/promotional events – a success of marketing campaigns can be determined and even improvised based on target audience pulse.

# **Contributions**

* Project Topic: Sathish Kumar Rajendiran
* Twitter Streaming: Sathish Kumar Rajendiran
* MongoDB & JSON Analysis: Prasad Kulkarni, Sathish Kumar Rajendiran
* Data Cleansing and Formatting: Prasad Kulkarni, Sathish Kumar Rajendiran
* Twitter Trend Analysis: Prasad Kulkarni
* Wordcloud: Sathish Kumar Rajendiran
* Summary & Conclusion: Prasad Kulkarni, Sathish Kumar Rajendiran