CSE4/587 Data-intensive Computing Spring 2017

LAB1: DATA CLIENTS AND INFORMATION SERVERS: B. RAMAMURTHY

OVERVIEW:

The hands-on practical learning components of the course comprises two types of activities: labs covering one or two knowledge units (skills, competencies) of data-intensive computing and a single term project serving as a capstone covering the entire data pipeline. This document describes Lab1: Data Clients and Information Servers that deals primarily with data collection.

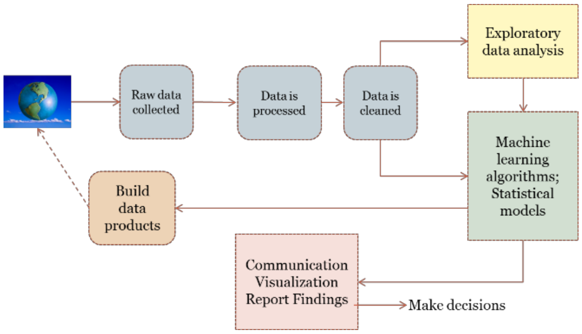
GOALS:

- Install a work environment for carrying out various activities of the data science process as described in [1] (your text book), repeated here for completeness.
- Understand and implement Application Programming Interface (API) based programmatic data collection from popular (/public) data sources (Data clients).
- Process the data collected for extracting basic information.
- Serve the information extracted through simple applications and visualization (Information servers).

OBJECTIVES:

The lab goals will be accomplished through these specific objectives:

1. Install Jupyter [2] notebook environment and within it R language [3] kernel. See the instructions here [4].
2. Familiarize yourself with Jupyter Notebook environment and R language.
3. Collect data by querying Twitter REST API [5]. You will have to get a developer account on twitter and also get the credentials for your application (the twitter client) that you will be writing.
4. Process data using twitteR [6] library package of R.
5. All the query inputs from the user (for data collection, summarization and visualization) will be specified by Jupyter widgets.
6. Summarize (simple print statement) information extracted as answers for specific queries.
7. Visualize geo spatial information extracted from the tweets using geo-map libraries of R: ggplot2, ggmap, maps,and maptools [7]. Maps and geo codes are supported by Google map API.


Raw data collected
Data is processed
Data is cleaned
Exploratory data analysis
Machine learning algorithms; Statistical models
Build data products
Communication Visualization Report Findings
Make decisions

LAB DESCRIPTION:

Introduction: An important and critical phase of the data-science process is data collection. Several organizations including the federal government (data.gov) have their data available to the public for various purposes. Social network applications such as Twitter and Facebook collect enormous amount of data contributed by their numerous and prolific user. For other businesses such as Amazon and NYTimes data is a significant and valuable byproduct of their main business. Nowadays everybody has data. Most of these data generator businesses make subset of their data available for use by registered users for free. Some of them as downloadable data files (.csv, .xlsx) as a database (.db, .db3). Sometimes the data that needs to be collected is not in a specific format but is available as a web page content. In this case, typically a web crawler is used to crawl the web (pages) and scrap the data from these web pages and extract the information needed. Data generating organizations have realized the need to share at least a subset of their data with users interested in developing applications. Entire data sets are sold as products.

What is an API? Why is this so important? A standard, secure and programmatic access to data is provided through an Application Programming Interface (API). We are in an API economy [ ]. An API offers a method for one or two way communication among software (as well as hardware) components as long as they carry the right credentials. These credentials for authentication for programmatic access is defined by another standard OAuth (Open Authentication) delegation protocol [8].

Twitter Application Development: We will develop applications that are “data clients” for twitter data. Twitter supports many APIs: REST API, Search API, Streaming API, Firehouse API, and Ads API. We will use Search API that is a part of the REST API.

Preparation: Here are the preliminary requirements for the lab. Time needed: 1 or 2 hours (Day 1)

1. Work environment: You will working on Jupyter with R kernel. Install Jupyter and R kernel as instructed by the handout [. This will be our “Learning Environment”. Later on we will explore “Development Environment” in RStudio; (Production Environment” in a robust programming language such as Java or C++).
2. Create an account on twitter as a user as well as a developer. In the developer site, the tab MyApps is of particular interest. (After you create a twitter developer account,) you will click on MyApps to create a new app called Lab1. Fill in the required fields **as per the instructions given there.** Once you submit you should be able to get the OAuth credentials that had four parts: Customer API key, Customer API secret, Access Token Key and Access Token Secret. All are needed for programmatically working with Twitter. (Yes, you can auto-tweet, if you know what I mean ;-)
3. R community has created a package for working with Twitter data called “twitteR”. Read the vignette by Jeff Gentry [10] about the package he contributed.

LAB 1: WHAT TO DO?

1. (10 points) Learning Jupyter, R and twitteR: All these can be achieved by one activity: working with twitteR package library vignette. Type in the R language instructions for each example discussed: try it with different names and twitter hash tags. I tried it with "#datascience" for topics and for person I used "elonmusk". You don't try the same, you try some other hashtag and person of interest to you. Time needed: 2-3 hours (Day 2)

2. (20 points) Problem 1: Study response to an event.
   We are interested in finding out how the nation reacted to the Super Bowl. We are NOT interested in sentiment analysis. We are interested in sheer number of tweets on a topic that is associated with Super Bowl LI. You have to choose a good topic. Understand the Search API that we are using for can give you only limited number of tweets per day and also only a sampling of the all the tweets. You will collect at least 20000 tweets (Hmm…How could we categorize them?? We will learn later.). Group them by geo-location as in Google maps API (one more API) and plot them on the map of USA. If you plot the location every tweet then there will be too many points on the map. You can plot all the tweets at a given location (say a city or state) by a single blob, the size of the blob representing the density of tweets. You may need some R programming here.
   Input: Search word or hash tag related to super bowl. Data client processing: Obtain and group tweets by location. Output: plot the groups by size on a map of USA for visual understanding of the response to an event. (Days 3, 4)
   Issue 1: Of course, there is an issue with location meta-data. This is not available (N/A) if the user does hide his/her location. This is often the case nowadays with most of us. Many celebrities are especially conscious about this. They don't want people knowing their locations for obvious reasons. Then how can we get "set of locations"?
   Here is a verified approach using function of twitteR
   1. Convert search result tweets into dataframe
   2. Lookup screen name from this dataframe
   3. From Screen names get user info and convert into dataframe
   4. Keep only users with location info
   5. Get the geo code of the locations from this dataframe
   6. Hints on TwitteR functions you may need: twListToDF, lookupUsers, geocode

   (5 points) Generalize: Once this is completed, generalize the solution for any twitter search hashtag. We should be able to reproduce the results for any event or happening of your choice. Add a text input as a widget so that user can interact with your program with their choice of search word for the twitter search API. (Day 5)

3. (10 points) Problem 2: Summarizing trending topics about a location (place).
   When we are visiting places (say, for an interview or other official visits) you may want to about topics trending in that place. Instead of reading newspapers and online news, you want just a quick summary. You want to put use your twitter "data client" application development experience. You use the twitteR libraries "trends" function to retrieve 10 top things trending about the place and summarize it appropriately as a complete message (print out).

Input: Location specified either as geo-location or by its name Output: A message listing the topics trending about the place. (Day 6)

(5 points) Generalize: Once above problem is completed generalize your solution for any location. Add a text input widget for inputting the location. (Day 7)

4. Bundle all the work in two different notebooks and submit it. Make sure your document your application development.
5. You have to work on your own. This is an individual lab. You will get an F for the course if you plagiarize or copy somebody else's work or share your work with somebody.

DUE DATE: 2/18/2017 BY MIDNIGHT. ONLINE SUBMISSION.

REFERENCES:

[1] C. ONeil and R. Schutt, Doing Data Sceince, ISBN:978-1-4493-5865-5. Oreilly Media, Doing Data Sceince, http://shop.oreilly.com/product/0636920028529.do

[2] Jupyter. http://jupyter.org/, last viewed 2017.

[3] The R Language. https://cran.r-project.org/, last viewed 2017.

[4] Jupyter-R Kernel Installation instruction. Class handout. CSE 4/587 Spring 2017.

[5] Twitter API. Twitter Developer https://dev.twitter.com/, last viewed 2017.

[6] TwitteR package. https://cran.r-project.org/web/packages/twitteR/twitteR.pdf, last viewed 2017.

[7] D. Kahle and H. Wickham. ggmap: Spatial Visualization with ggplot2. The R Journal Vol. 5/1, June 2013, https://journal.r-project.org/archive/2013-1/kahle-wickham.pdf.

[8] OAuth2.0. OAuth2.0: https://oauth.net/2/, last viewed 2017.

[9] API Economy, http://www.ibm.com/cloud-computing/learn-more/hybrid-integration/api-economy/, last viewed 2017.

[10] J. Gentry. TwitteR Vignette: A Twitter Client for R. http://geoffjentry.hexdump.org/twitteR.pdf, last viewed 2017.