Instant Visualization of Twitter Data with Data Dashboard

DS-GA 1006 Capstone Project

Meihao Chen & Yitong Wang
Center for Data Science, New York University

Contact Information:
Center for Data Science
New York University

Abstract

Dahlia is an instant visualization application specially designed for analysis of Twitter API data. It is written mostly in JavaScript and PHP and runs in a web server (XAMPP) environment. Dahlia provides robust and reproducible data analysis, and interlinks with existing visualization packages. Analytical features are based on two main ideas:

- captured datasets can be relatively large for HTML to handle and thus requires preprocessing;
- the resulting selections of data can be analyzed in various ways, mainly by instant visualization on a webpage.

Introduction

As Twitter has become a global and real-time mirror of what is happening in the world at any given moment, being able to access tweet information opens up a world of possibilities: from monitoring mentions of your brand or event (think of it as real-time market research) to performing demographic analysis to using Twitter activity as research material, uses of this feature are endless. With an online dashboard that combines all this with the analysis and visualization capabilities, you can execute super advanced analysis on events that are of you interests.

Main Objectives

1. Visualize the massive amount of twitter data (tweets) in a timely fashion;
2. The final deliverable is ideal for fast information extraction;
3. Preprocessing modules are reproducible in Python
4. To be utilized by a variety of industries and technical professionals as well as professionals without programming skills;
5. Especially suitable for professionals without programming skills.

Dataset: Twitter Data

We utilize 2 datasets for designing purposes:

- Hillarys Presidential Announcement (892.7 MB)
- 2014 Oscar (749.8 MB)

Follow table shows some statistics about these 2 datasets, and listed fields as well as their interaction will be visualized in dashboard.

Table 1: Dataset Fields Statistics

	Hillary.tar	Oscar.tar
Unique tweets	1592531	1440256
Hashtags	42889	62230
Geolocations	570	754
User mention	98484	172775
Sources	4164	2044
Unique Tweets	84863	137742
Words	110843	129648
Unique Users	687919	960104

For future data acquisition, user should acquire data before utilizing Dahlia since we donnot have build-in function to connect with Twitter API. On the one hand there exsits excellent app for Twitter API query (e.g. DMI-TCAT). On the other hand, such task is computationally expensive, and requires workload such large that itself can become a meaningful project.

Methodology

In this project we used various methods of visualization to optimize the user experience and information extraction. The visualization allows for user interaction, and also accounts for many interactions between fields.

The entire data visualization part of this project is written in JavaScript and PHP. We utilized various external resources, including D3js.org, NVD3.org, d3-cloud, d3-slider, intro-d3, jsPdf, Twitter Bootstrap, etc. As we go through every step of this project, we gladly acquired knowledge about JavaScript, and it libraries. This whole project is a fantastic learning experience for us.

Overview

Pie chart and stacked area charts are most useful when we intend to explore the overall structure of the dataset. In this part of the dash board, we provide pie charts and stacked area charts on different fields, and visualize on most mentioned items in such field.

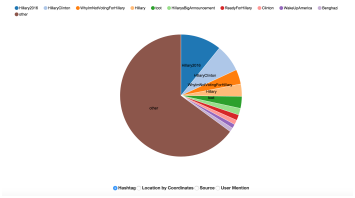

Figure 1: Pie Chart of Top 10 Hashtags

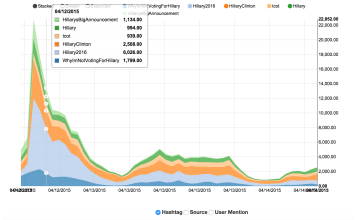

Figure 2: Stacked Area Chart of Top 5 Hashtags

Tweet Content Understanding

In order for users to get a better idea about the actual content of their Twitter dataset, we provide several visualization methods to describe the tweets from different perspectives.

Figure 3: Top 10 Popular Tweets

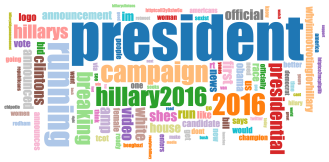

Figure 4: Word Cloud

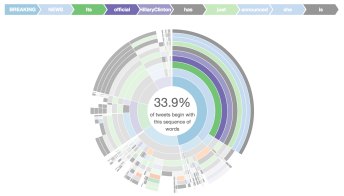

Figure 5: Word Sequence

Item Interaction

Interaction between different fields can bring significant insight about user behavior. Interactions among Twitter users can also be interesting in that it will demonstrate information influence on social network.

We found it most interesting to look at interactions between hashtags and geolocation. By look at out dashboard you will have a clear understanding of people's opinions or focuses in different states. Also, our hierarchical bundling graph will show the most important tweet-retweet relations, which illustrates how does information flows within dataset analyzed.

Other Functions

We also designed many other functions to improve user experience, such as tour function, export to pdf function, and show reel animation.

Results

As stated before, we want to design an application that is especially suitable for professionals without programming skills to visualize Twitter data, our final product is very easy to install and use.

- Easy installation: we have automated installation for Mac OS X, and will test on other systems in the future;
- Easy to use: after installation is completed, user will only need to go to web server directory and run './run.sh' in command line.

Following is a screen shot of our data dashboard. You are welcome to play with it on our laptop or find it on out github!

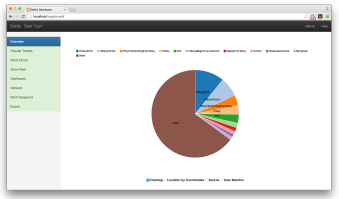

Figure 6: Dahlia Dashboard

Conclusions

- Memory handling is one of the most important tasks, as memory explosion will greatly impair user experience. In this task we chose python modules to preprocess dataset, and visulization functions will read well-formatted data from disk;
- Mapreduce might not be the most suitable method to preprocess data for this task: when reducing data, a large part of dataset may be lost due to improper exception handling, and usually requires looping over the dataset several times, which is computationally inefficient;
- Cross-field visualization is not just an extension of single field visualization. It requires much more work on behavior handling and does provide good insight about the dataset.

Forthcoming Research

- Data Acquisition: Connection with Twitter API, database management, and query function that allows for customized data capture; refine captured data: search queries, exclusions, date range, etc.

Acknowledgements

We want to thank Prof. Pablo Barber and Prof. Claudio T. Silva for their guidance. We also want to give our special thanks to Bowen Yu for his kind help on introducing us to the world of visualization.