Instant Visualization of Twitter API Data

Team Dahlia:
Meihao Chen, Yitong Wang

Our Objectives

- Visualize the massive amount of twitter data (tweets) in a timely fashion
- Ideal for fast information extraction
- Reproducible in Python
- To be utilized by a variety of industries and technical/non-technical professionals
- Especially suitable for **non-technicals**

Twitter Dashboard: Why Bother?

- **Graphics VS. Table**
- **Lack of Visualization Tools for Non-Tech**
  - **Unable to handle large dataset**
  - **Require some knowledge about coding**

- ✓ **Most straightforward, convenient way for non-tech specialists**

Dataset: Twitter API Data

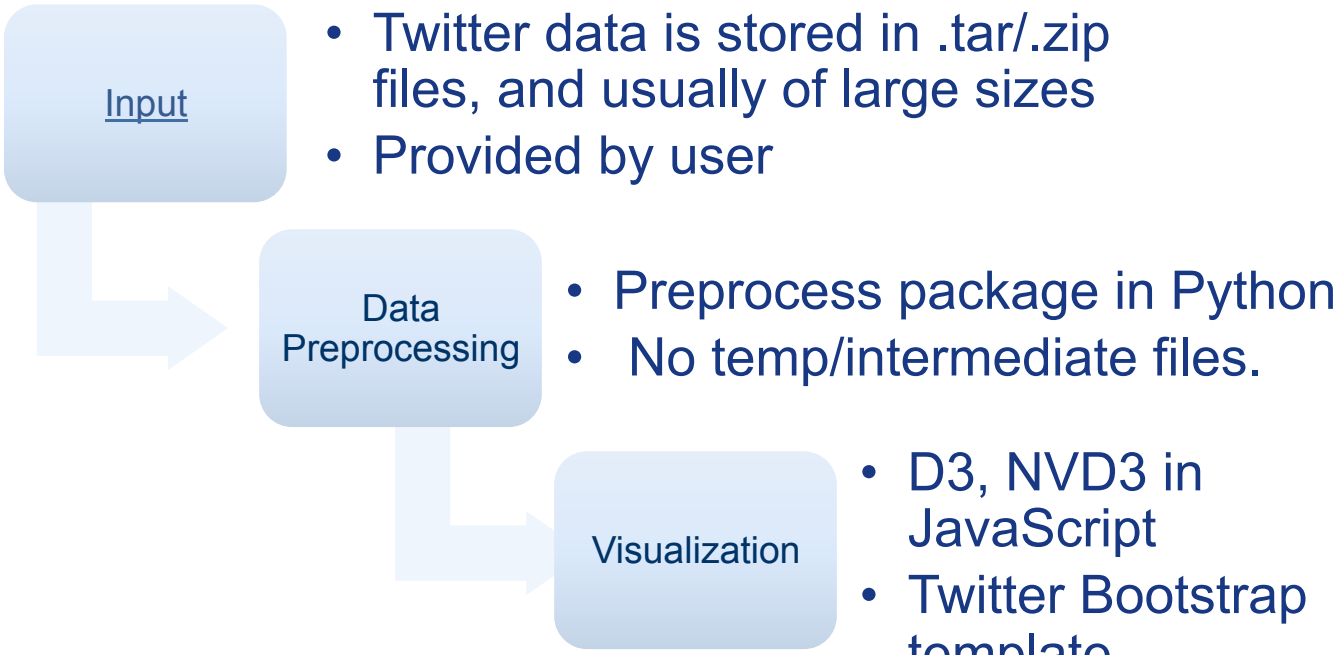
- **2 datasets (Provided by Pablo)**
  - **Hillary's Presidential Announcement 892.7 MB**
  - **2014 Oscar 749.8 MB**

	Hillary.tar	Oscar.tar
No. of Instances	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

Dataset: Twitter API Data

- **Future Data Acquisition:**
  - **User should acquire data before utilizing Dahlia:**
    - **Existing app for Twitter API query: DMI-TCAT**
    - **Computationally expensive**
- **Data Processing:**
  - **Field selection and writing files in python VS MapReduce**
  - **Running Time: ~10 min for 1 Gb**

Pipeline: How do we get there?



- Twitter data is stored in .tar/.zip files, and usually of large sizes
- Provided by user

- Preprocess package in Python
- No temp/intermediate files.

- D3, NVD3 in JavaScript
- Twitter Bootstrap template

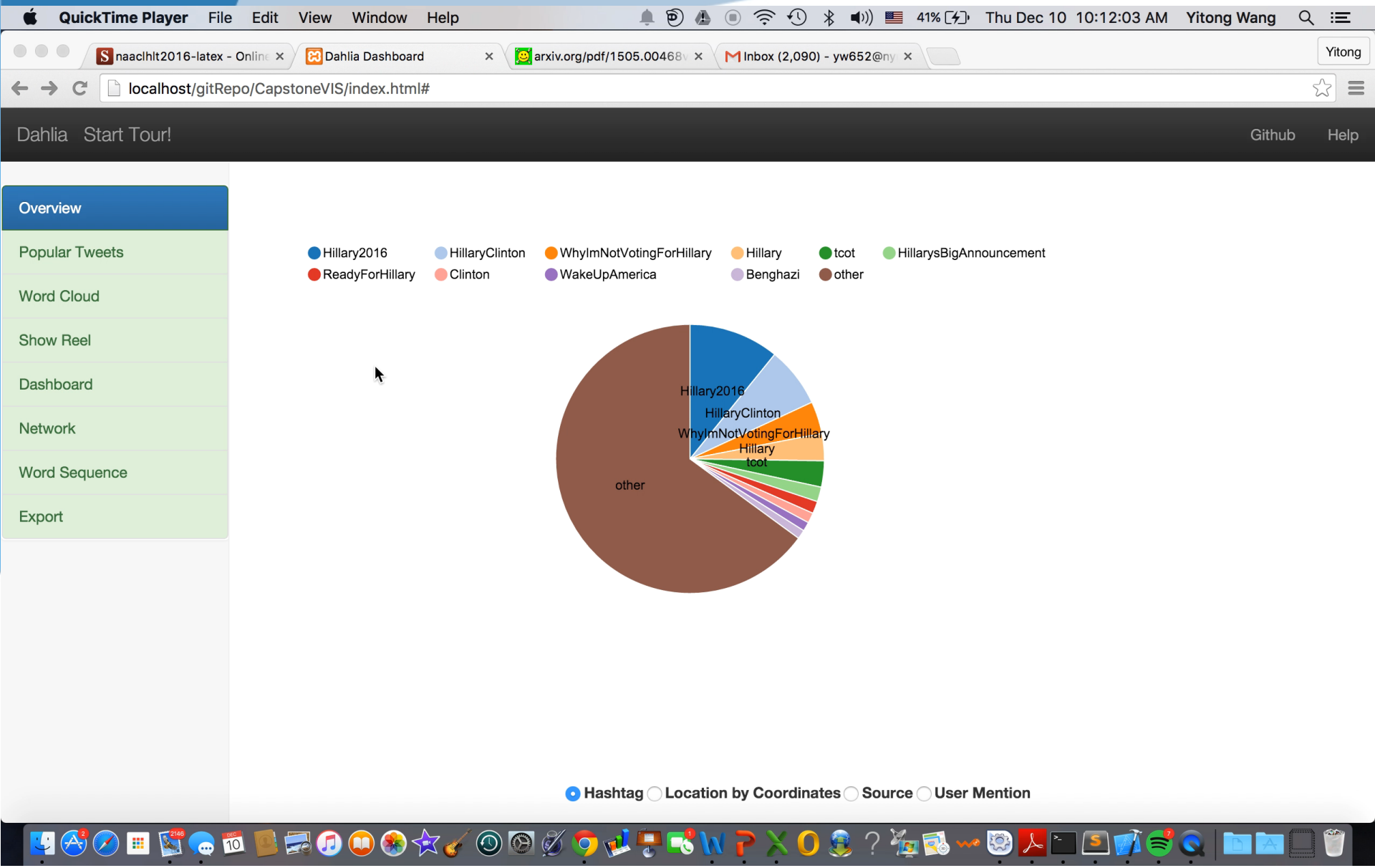
Methodology: Visualization

- **Overview:** Pie Charts and Stacked Area Charts
- **Tweet Content:**
  - Table of popular tweets
  - Word Sequence
- **Keywords:** Word Cloud
- **Trends Animation:** Show Reel
- **Interaction:**
  - **Fields:** Dashboard on location and hashtag
  - **User:** Hierarchical Bundling Graph

Product: How does it work?

- Dahlia
  - Proc
    - proc.py — Preliminary process, utility functions
    - main.py
    - deploy.sh — Automated install
    - ./Proc_d3 — Process for visualization, output goes to Vis/data
  - Vis
    - ./bower_components
    - ./css
    - ./data — Data directly used for visualization
    - ./js
    - Index.html — Visualization result in html
  - run.sh — Trigger data processing

Hillary-Demo



Oscars-Demo

Chrome File Edit View History Bookmarks People Window Help 53% Thu Dec 10 10:26:41 AM Yitong Wang

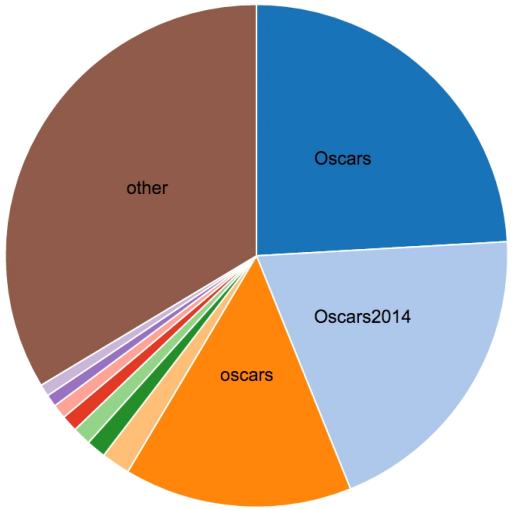
naaclhlt2016-latex - Onlin | Dahlia Dashboard | arxiv.org/pdf/1505.00468 | Inbox (2,090) - yw652@ny | Yitong

localhost/gitRepo/CapstoneVIS/index_o.html#

Dahlia Start Tour! Github Help

- Overview
- Popular Tweets
- Word Cloud
- Show Reel
- Dashboard
- Network
- Word Sequence
- Export

Oscars Oscars2014 oscars Oscar OscarsForVenezuela SOSVenezuela ERedCarpet Gravity Blessed oscars2014 other



Hashtag Location by Coordinates Source User Mention

Product: Getting Started

- Easy Installation:
  - Automated install for Mac OS X
  - Will test on other systems in the future
- Easy to Use:
  - Go to yours web server directory
  - **./run.sh** in command line

Future Work

- Data Acquisition: Connection with Twitter API
- Refine Captured Data: search queries, exclusions, date range, etc.)
- More User Interaction

Acknowledgement

- **Pablo Barberá**
- New York University

- **Claudio T. Silva**
- New York University

- **Bowen Yu**
- Ph.D. Candidate in Computer Science,
- Computer Science and Engineering Department,
- New York University Tandon School of Engineering

Thank You

- **Our Wiki**: https://github.com/NYU-CDS-Capstone-Project/dahlia/wiki
- **Our Demo:**
- http://dahliallc.github.io/#

External resources

- **D3js.org** https://d3js.org
- **D3-cloud** https://github.com/jasondavies/d3-cloud
- **D3-slider** http://thematicmapping.org/playground/d3/d3.slider/
- **jsPDF** https://github.com/MrRio/jsPDF
- **Introjs-D3** https://github.com/anmolkoul/introjs-D3
- **Twitter bootstrap** http://getbootstrap.com
- **NVD3** http://nvd3.org/