
Storytelling with Streaming Data

HW3-Distribution and API

YU WANG UNI: yw2783 - March 9, 2016

Purpose

This homework is based on homework 2 and the same main idea about presidential election. The project consumes streaming data from Twitter API and parse tweets for presidential election. In other words, we parse tweets whose content is about the election and identify different parties: Hillary Clinton or Donald Trump. The following list is what I did for the assignment:

- Tracked distribution — track tweets about Hillary, Trump and represent the Bernoulli distribution by a pie chart.
- API — An API is built to return the rate, entropy and distribution
Visit <http://localhost:5000/rate> to get rate
Sample return {"rate": 57.0}
Visit <http://localhost:5000/entropy> to get entropy
Sample return {"entropy": 0.56, "time": "2016-03-09 16:40:11"}
Visit <http://localhost:5000> to get distribution
Sample return {"rate": 59, "clinton": 0.71, "trump": 0.29}
- Extended the alert system
Based on observation, we set the following condition to trigger an alert. As either the alert would be too often or too rare if we change the percentage to lower or higher.
*if current_entropy - previous_entropy > 2% * previous_entropy*
then alert
- Built a webpage that display the current distributions stored in Redis by querying the API
A histogram chart and a pie chart will be presented on the webpage to represent entropy and data distribution. The graph is updated every 10s by querying API.

This document will demonstrate how the distribution, entropy and rate will be presented (with demo graph), what is the structure of source code, and how to run the application. The following sections will elaborate the information respectively.

Data Visualization

The project will present distribution, entropy and rates both in graph and absolute numbers.

1. Three text box updating in real time to present number of each value.
2. Entropy Histogram updating every 10s in 90s time slot
3. Distribution Pie chart updating every 10s

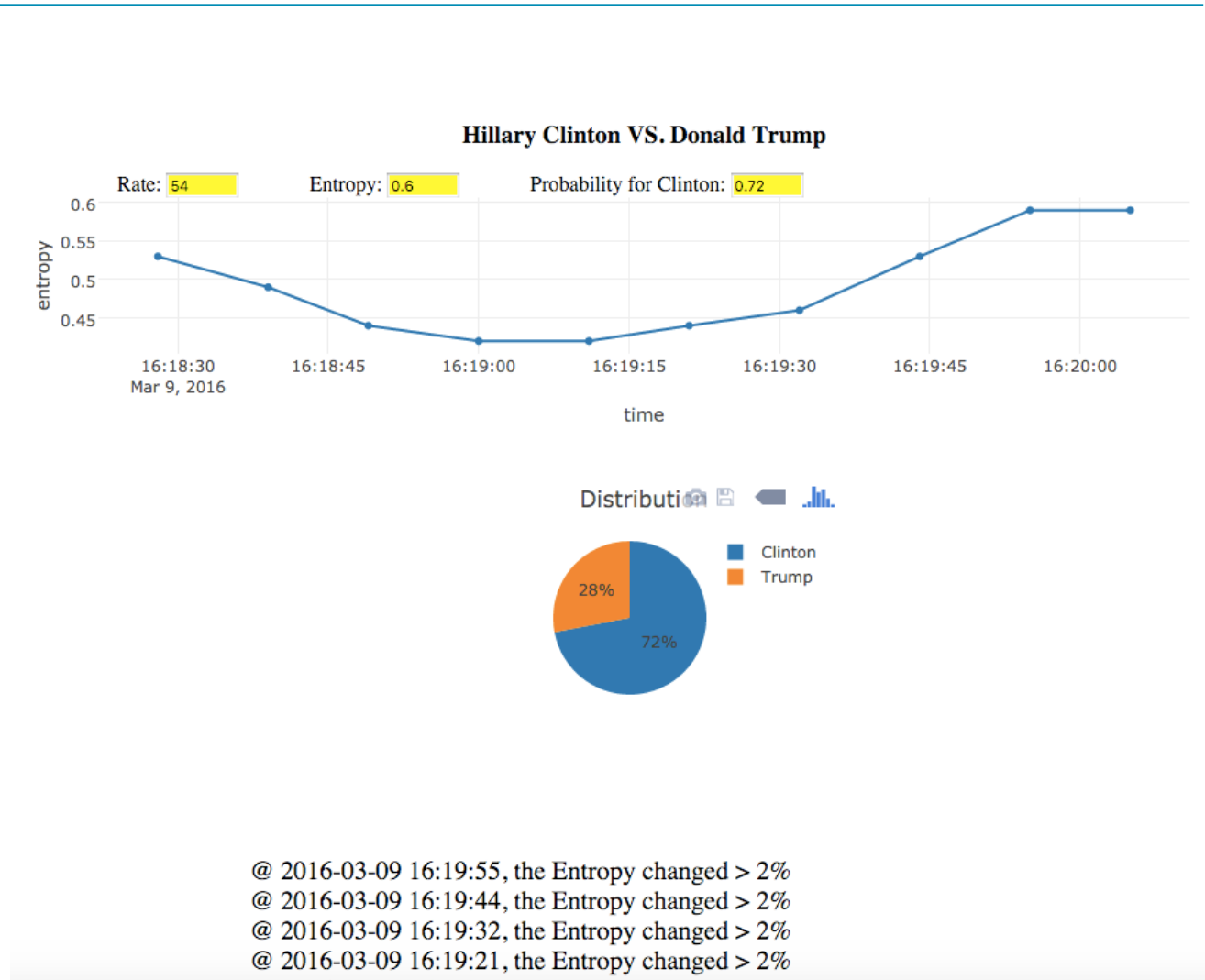


fig 1. data visualization

The graph shows the entropy trend in 90s time slot. Once the entropy change satisfies the condition:

*if $current_entropy - previous_entropy > 2\% * previous_entropy$*

An alert will be appeared on the webpage as well as a message. (figure 2)

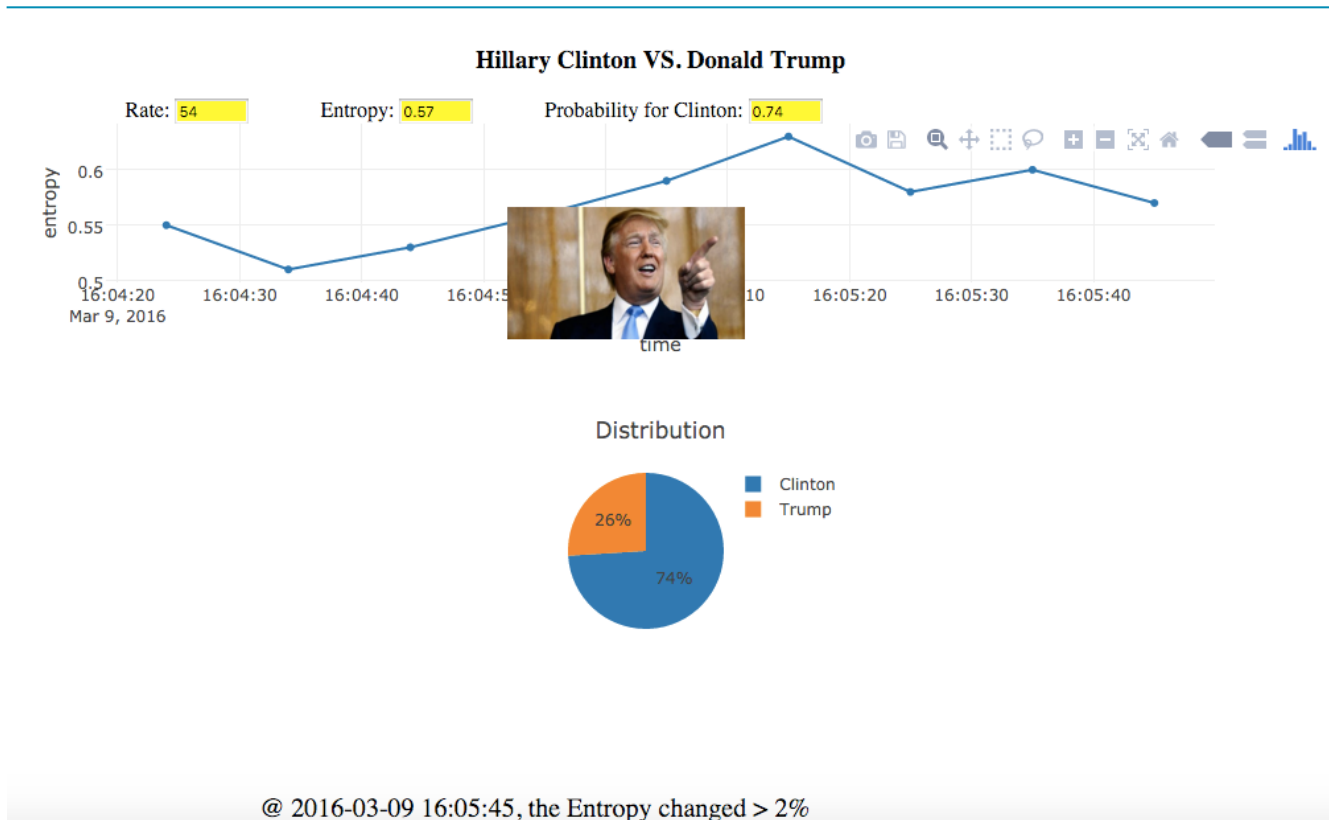


fig.2 Alert Scenario

How to Run

1. Run Redis database server
#redis-server
2. Run the flask server
#python server.py
3. Get tweets data
python get_data.py
4. Open the webpage in browser by link:
<http://localhost:5000/index>

Source Code

- server.py
 - Run flask server
 - Build histogram
- get_data.py
 - Fetch the twitter streaming data
 - Filter tweets about "Hillary Clinton" and "Donald Trumps"
- script.js
 - Query API to get entropy, rate and distribution
 - Draw histogram and Pie graph on the frontage
- index.html

-
- Web front-end page to demonstrate the result and graph

Note: The functionality of source code is also explained in the comments of each file.