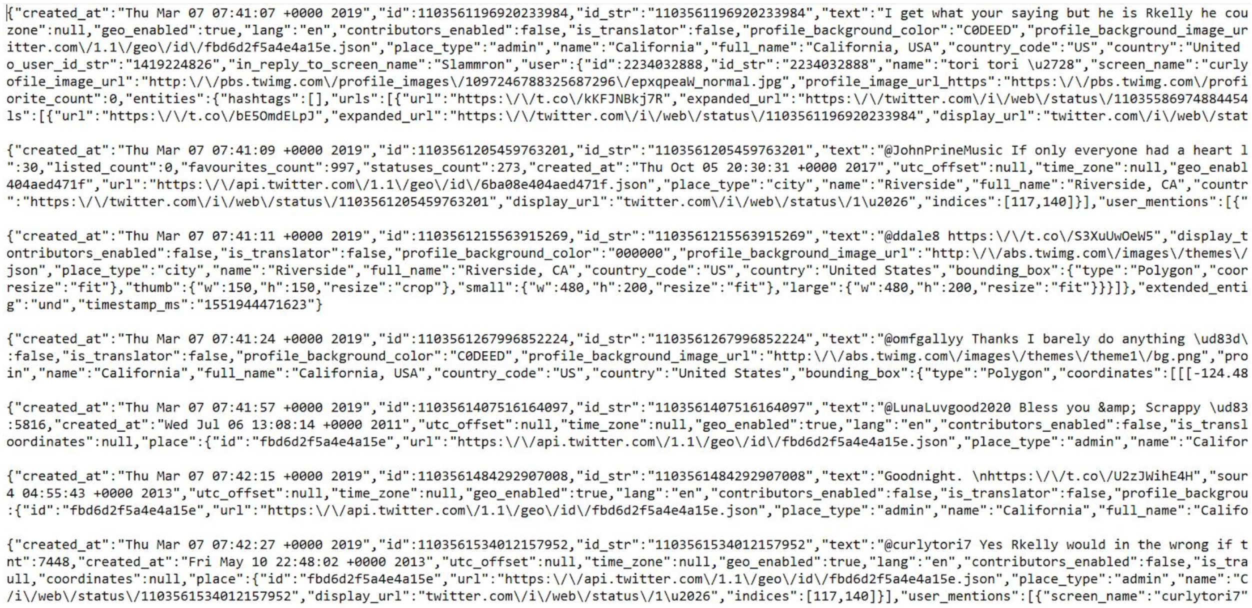
CS172 Final Project Report

Thomas Lee
Patrick Nguyen
Alex Shin
Waheed Sharifi

Crawler

Thomas Lee - Pair programming, ran twitter_streaming.py for a week, researched Twitter Streaming API and Tweepy
Patrick Nguyen - Pair programming, researched Twitter Streaming API and Tweepy

Using the Tweepy library/documentation, we grabbed the Consumer API key, the Consumer API secret key, the Access token, and the Access token secret. We stored them into variables for use in the authentication/connection to the Twitter Streaming API. We filtered the stream by the locations filter, using this geolocation bounding box argument: `[-117.498779,33.884097,-117.250900,34.015104]`. (A box roughly from Jurupa Valley in the top left, to Moreno Valley in the bottom right.) Then it would print anything that came through the stream. So we ran it in the terminal by using `python twitter_streaming.py > twitter_data.txt` to output the printing to a file we could later use.
This is what the twitter_data.txt would look like (each one came in the Tweet JSON format):



```
{"created_at":"Thu Mar 07 07:41:07 +0000 2019","id":1103561196920233984,"id_str":"1103561196920233984","text":"I get what your saying but he is Rkelly he cou
zone":null,"geo_enabled":true,"lang":"en","contributors_enabled":false,"is_translator":false,"profile_background_color":"C0DEED","profile_background_image_ur
itter.com\/1.1\/geo\/id\/fbd6d2f5a4e4a15e.json","place_type":"admin","name":"California","full_name":"California, USA","country_code":"US","country":"United
o_user_id_str":"1419224826","in_reply_to_screen_name":"Slammron","user":{"id":2234032888,"id_str":"2234032888","name":"tori tori \u2728","screen_name":"curly
ofile_image_url":"http:\/\/pbs.twimg.com\/profile_images\/1097246788325687296\/epxqpeaW_normal.jpg","profile_image_url_https":"https:\/\/pbs.twimg.com\/profi
orite_count":0,"entities":{"hashtags":[],"urls":[{"url":"https:\/\/t.co\/kKFJNBkj7R","expanded_url":"https:\/\/twitter.com\/i\/web\/status\/1103558697488454
ls":[{"url":"https:\/\/t.co\/bE5OmdELpJ","expanded_url":"https:\/\/twitter.com\/i\/web\/status\/1103561196920233984","display_url":"twitter.com\/i\/web\/stat

{"created_at":"Thu Mar 07 07:41:09 +0000 2019","id":1103561205459763201,"id_str":"1103561205459763201","text":"@JohnPrineMusic If only everyone had a heart l
":30,"listed_count":0,"favourites_count":997,"statuses_count":273,"created_at":"Thu Oct 05 20:30:31 +0000 2017","utc_offset":null,"time_zone":null,"geo_enabl
404aed471f","url":"https:\/\/api.twitter.com\/1.1\/geo\/id\/6ba08e404aed471f.json","place_type":"city","name":"Riverside","full_name":"Riverside, CA","countr
":"https:\/\/twitter.com\/i\/web\/status\/1103561205459763201","display_url":"twitter.com\/i\/web\/status\/1\u2026","indices":[117,140]}],"user_mentions":[{"

{"created_at":"Thu Mar 07 07:41:11 +0000 2019","id":1103561215563915269,"id_str":"1103561215563915269","text":"@ddale8 https:\/\/t.co\/S3XuUwOeW5","display_t
ontributors_enabled":false,"is_translator":false,"profile_background_color":"000000","profile_background_image_url":"http:\/\/abs.twimg.com\/images\/themes\/
json","place_type":"city","name":"Riverside","full_name":"Riverside, CA","country_code":"US","country":"United States","bounding_box":{"type":"Polygon","coor
resize":"fit"},"thumb":{"w":150,"h":150,"resize":"crop"},"small":{"w":480,"h":200,"resize":"fit"},"large":{"w":480,"h":200,"resize":"fit"}}}]},"extended_enti
g":"und","timestamp_ms":"1551944471623"}

{"created_at":"Thu Mar 07 07:41:24 +0000 2019","id":1103561267996852224,"id_str":"1103561267996852224","text":"@omfgallyy Thanks I barely do anything \ud83d\
:false,"is_translator":false,"profile_background_color":"C0DEED","profile_background_image_url":"http:\/\/abs.twimg.com\/images\/themes\/theme1\/bg.png","pro
in","name":"California","full_name":"California, USA","country_code":"US","country":"United States","bounding_box":{"type":"Polygon","coordinates":[[[-124.48

{"created_at":"Thu Mar 07 07:41:57 +0000 2019","id":1103561407516164097,"id_str":"1103561407516164097","text":"@LunaLuvgood2020 Bless you &amp; Scrappy \ud83
:5816,"created_at":"Wed Jul 06 13:08:14 +0000 2011","utc_offset":null,"time_zone":null,"geo_enabled":true,"lang":"en","contributors_enabled":false,"is_transl
oordinates":null,"place":{"id":"fbd6d2f5a4e4a15e","url":"https:\/\/api.twitter.com\/1.1\/geo\/id\/fbd6d2f5a4e4a15e.json","place_type":"admin","name":"Califor

{"created_at":"Thu Mar 07 07:42:15 +0000 2019","id":1103561484292907008,"id_str":"1103561484292907008","text":"Goodnight. \nhttps:\/\/t.co\/U2zJWihE4H","sour
4 04:55:43 +0000 2013","utc_offset":null,"time_zone":null,"geo_enabled":true,"lang":"en","contributors_enabled":false,"is_translator":false,"profile_backgrou
:{"id":"fbd6d2f5a4e4a15e","url":"https:\/\/api.twitter.com\/1.1\/geo\/id\/fbd6d2f5a4e4a15e.json","place_type":"admin","name":"California","full_name":"Califo

{"created_at":"Thu Mar 07 07:42:27 +0000 2019","id":1103561534012157952,"id_str":"1103561534012157952","text":"@curlytori7 Yes Rkelly would in the wrong if t
nt":7448,"created_at":"Fri May 10 22:48:02 +0000 2013","utc_offset":null,"time_zone":null,"geo_enabled":true,"lang":"en","contributors_enabled":false,"is_tra
ull,"coordinates":null,"place":{"id":"fbd6d2f5a4e4a15e","url":"https:\/\/api.twitter.com\/1.1\/geo\/id\/fbd6d2f5a4e4a15e.json","place_type":"admin","name":"C
/i\/web\/status\/1103561534012157952","display_url":"twitter.com\/i\/web\/status\/1\u2026","indices":[117,140]}],"user_mentions":[{"screen_name":"curlytori7"
```

A limitation of this method is that we only received a small portion of the overall tweets coming through the Twitter Streaming API.

Indexer

Thomas Lee - pair programming, running ElasticSearch, writing the ElasticSearch queries, writing the elasticindex.sh and elasticquery.sh scripts
Patrick Nguyen - pair programming, researching ElasticSearch
Alex Shin - researching ElasticSearch

We used ElasticSearch to index our collected tweets. Using the ElasticSearch Query DSL, we wrote queries that could search for any field inside the tweets. We settled on one that would search the bodies of the tweets, using the “query” and “match” fields in the DSL.

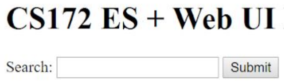
Limitations include: Our elasticindex.sh script splits loading the tweets files (there are 12 separate files containing all of our tweets) one by one, because for some reason our windows terminal could not handle loading one very large file containing all the tweets at once.

To run, start ElasticSearch. (Depends on what version and where you downloaded ElasticSearch on your computer.) After ElasticSearch is started, run the elasticindex.sh script. This will load all of our tweets into ElasticSearch. After this, you could return tweets one of two ways. The first way is covered here, the second way is covered in the extensions. Elasticquery.sh is a script that will ask you to type a query term, and will return tweets based on relevance in the tweet body. (This is returned in JSON format, not very user-friendly.)

Extension: Web UI

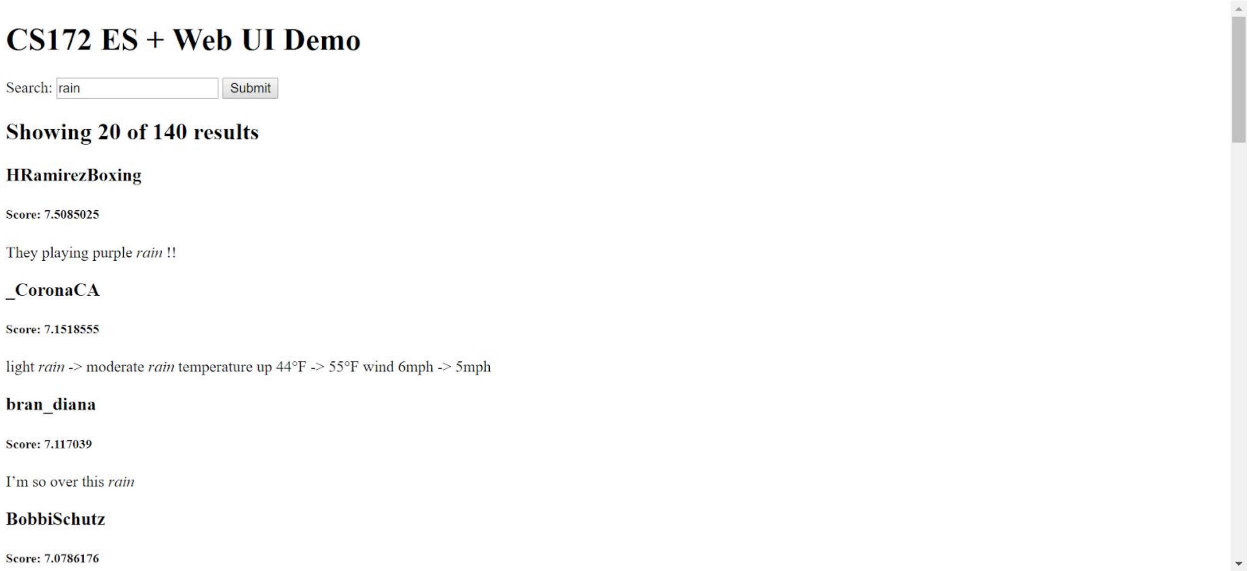
Thomas Lee - pair programming, wrote index.html, researched html, XMLHttpRequest(), javascript debugging, CORS complications
Patrick Nguyen - pair programming, researched html, XMLHttpRequest(), javascript debugging, CORS complications
Alex Shin - pair programming, added to & altered index.html, XMLHttpRequest() / CORS
Waheed Sharifi - pair programming, added to & altered index.html, XMLHttpRequest() / CORS

We created a web UI that includes a search bar, which takes in a user's search term/query. We wrote a javascript function that uses XMLHttpRequest() that runs a query and returns results from our ElasticSearch server. This displays the results in a much more user-friendly way, with the Screen Name, ElasticSearch query score, and the tweet body snippet, with the search term highlighted (further covered in the next extension). We also used XMLHttpRequest() to display a map generated by the Google Maps API (covered in last extension) and placed it on the left hand side next to the results. In addition, we enclosed the returned results in a scrollable text box to ensure we can see the map at all times.



CS172 ES + Web UI Demo

Search:



CS172 ES + Web UI Demo

Search: rain

Showing 20 of 140 results

HRamirezBoxing

Score: 7.5085025

They playing purple *rain* !!

_CoronaCA

Score: 7.1518555

light *rain* -> moderate *rain* temperature up 44°F -> 55°F wind 6mph -> 5mph

bran_diana

Score: 7.117039

I'm so over this *rain*

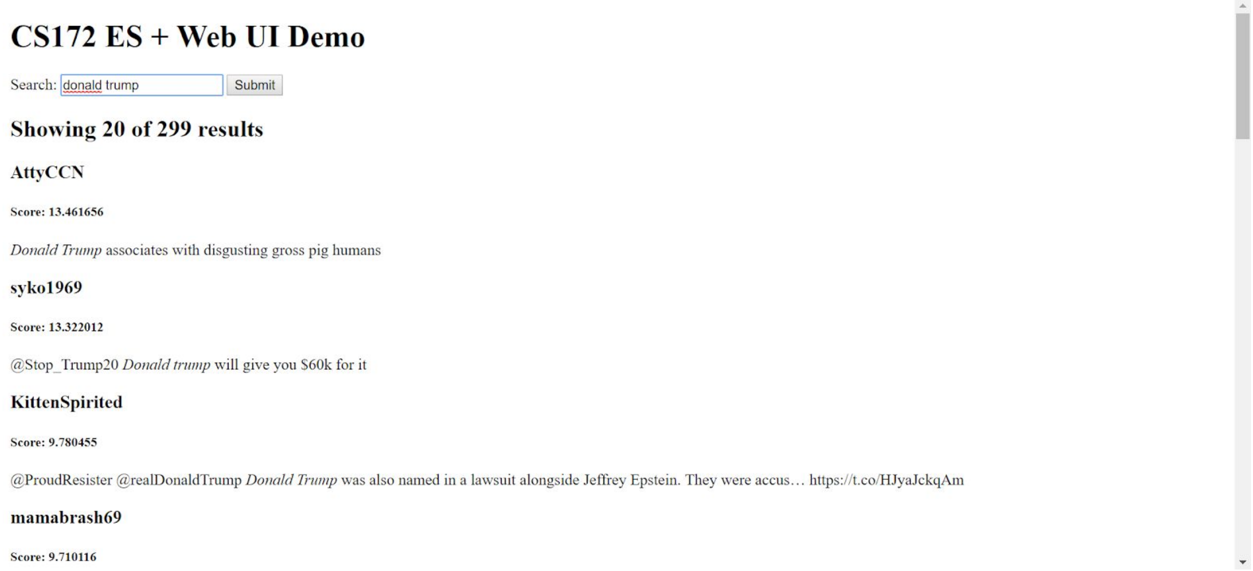
BobbiSchutz

Score: 7.0786176

Extension: Snippet

Thomas Lee - pair programming, wrote query/researched ElasticSearch Query DSL highlight function
Patrick Nguyen - pair programming, wrote query/researched ElasticSearch Query DSL highlight function

We created snippets that would get the score and text fields of the tweet. We created a javascript function that pulls those fields from the JSON file every time a query for a term would be searched. For that term, it would retrieve the tweet most relevant to the search then output the fields above alongside the userID.



CS172 ES + Web UI Demo

Search: donald trump Submit

Showing 20 of 299 results

AttyCCN

Score: 13.461656

Donald Trump associates with disgusting gross pig humans

syko1969

Score: 13.322012

@Stop_Trump20 *Donald trump* will give you $60k for it

KittenSpirited

Score: 9.780455

@ProudResister @realDonaldTrump *Donald Trump* was also named in a lawsuit alongside Jeffrey Epstein. They were accus… https://t.co/HJyaJckqAm

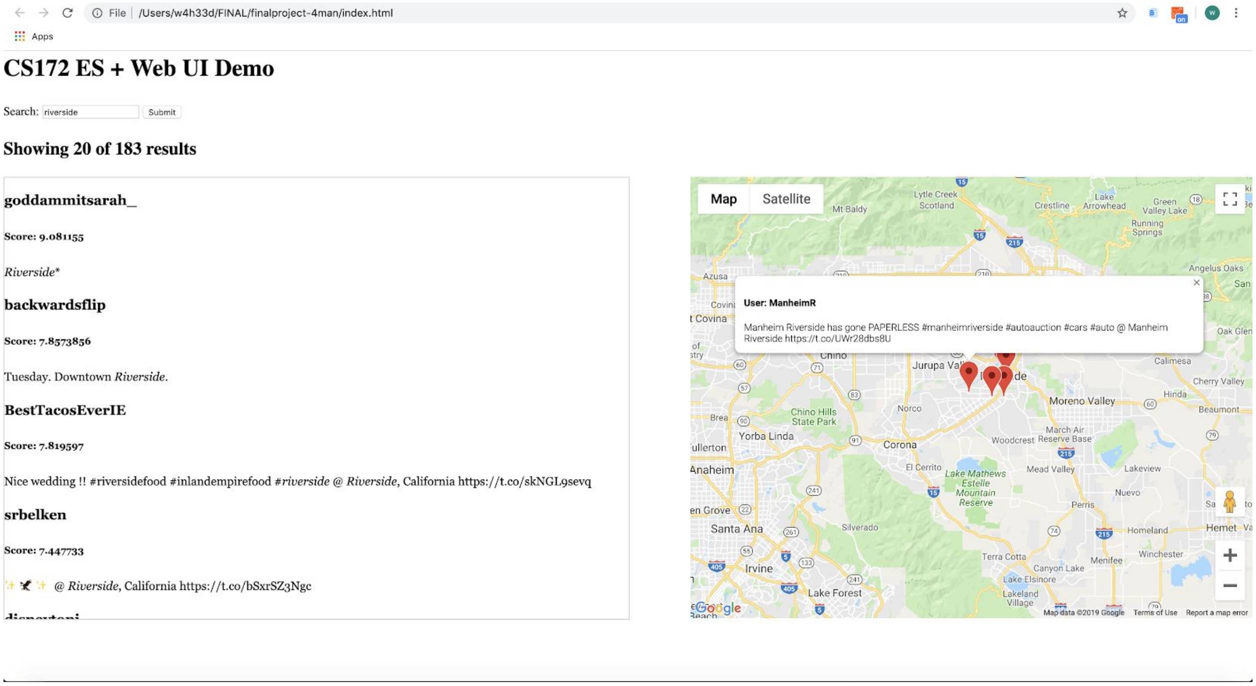
mamabrash69

Score: 9.710116

Extension: Plot Tweets on Map

Alex Shin - pair programming, Google Maps API
Waheed Sharifi - pair programming, Google Maps API

We added a map to our website UI that displays the location of tweets that have geolocation enabled from our searched results. The map centers around the first tweet, but is zoomed out so you are able to see other tweets in the area. When you click on one of the markers on the map, a text box will pop up with the user's name and tweet that they posted. Unfortunately because most people do not enable geolocations and we do not have an enormous amount of data, some searches do not have any tweets to mark on the map.



CS172 ES + Web UI Demo

Search: riverside Submit

Showing 20 of 183 results

goddammitsarah_

Score: 9.081155

*Riverside**

backwardsflip

Score: 7.8573856

Tuesday. Downtown *Riverside*.

BestTacosEverIE

Score: 7.819597

Nice wedding !! #riversidefood #inlandempirefood #*riverside* @ *Riverside*, California https://t.co/skNGL9sevq

srbelken

Score: 7.447733

✨ 🦅 ✨ @ *Riverside*, California https://t.co/bSxrSZ3Ngc

User: ManheimR

Manheim Riverside has gone PAPERLESS #manheimriverside #autoauction #cars #auto @ Manheim Riverside https://t.co/UWr28dbs8U