{"created_at": "Thu Apr 06 15:24:15 +0000 2017", "id_str": "850006245121695744", "text": "1\/ Today we\u201
9re sharing our vision for the future of the Twitter API platform!\nhttps:\/\/t.co\/XweGngmxlP", "user": {"
id": 2244994945, "name": "Twitter Dev", "screen_name": "TwitterDev", "location": "Internet", "url": "https:
\/\/dev.twitter.com\/", "description": "Your official source for Twitter Platform news, updates & events.
Need technical help? Visit https:\/\/twittercommunity.com\/ \u228\ufe0f #TapIntoTwitter"}, "place": { },
"entities": { "hashtags": [], "urls": [{ "url": "https:\/\/t.co\/XweGngmxlP", "unwound": { "url": "https:
\/\/car6.tr @r.com\/cards\/18cg3wg04h\/3xolc", "title": "Building the Ture of the Twitter API
Platfor]} [], "trt"@r/f@r.gar6.tr @r.com_at"@r.gar6.tr @r.com/cards\/18cg3wg04h\/3xolc", "title": "Building the Ture of the Twitter API
N//car6.tr @r.com\/cards\/18cg3wg04h\/3xolc", "title": "Building the Ture of ficial source foor
Twitter Platform news, updates & events. Need technical help? Visit https:\/\/twittercommunity.com\/ \u228
\u160f #TapIntoTwitter"}, "place": { }, "entities": { "hashtags": [], "urls": [{ "url": "https:\/\/twittercommunity.com\/ \u228
\u160f #TapIntoTwitter"}, "place": { }, "entities": { "hashtags": [], "urls": [{ "url": "https:\/\/twittercommunity.com\/ \u228
\u160f #TapIntoTwitter"}, "place": { }, "entities": { "hashtags": [], "urls": [{ "url": "https:\/\/txot
//xweGngmxlP", "uwound": [] } } { "url": "https:\/\/txot
//xweGngmxlP", "uwound": [] } } { "urls": "Twitter Dev", "isoreed_at": "Thu Apr 06
15:24:15 +0000 2017", "id_str": "S0006245121695744", "text": "1\/ Today we\u2019re sharing our vision for
the future of the Twitter API Platform"} }], "user": "I'data": "I'data": "Twitter Dev", "screen_name": "Twitter Dev", "url": "https:\/\/dev.twitter.com\/",
"Today we\u2019re sharing our vision for
the

Creating a geoFence

Create a geoFence and apply it to a collection of Twitter data

A small percentage of Tweets contain geocoordinates. The geocoordinates present in <u>Tweet</u> <u>JSON</u> represent the geographic location of this Tweet as reported by the user or client application. The coordinates are formatted as geoJSON (longitude, latitude). You can filter Tweets in your collection by location by using a geoFence. In this tutorial, you will learn how to create a geoFence for Clark County, Nevada, and then filter Tweets in the <u>1 October Twitter</u> <u>Data Collection</u> through that geoFence using <u>Twarc</u>. The resulting file will contain Tweets from the 1 October Collection that were sent in Clark County.

Difficulty level: Intermediate

Prerequisite(s)

- Tweet ISON
- Command Line
- <u>Collection Design</u>
- <u>Collection with Twarc</u>
- <u>Collection Documentation</u>
- Collection Ethics
- Cleaning Your Data
- <u>Collection Analysis</u>

Tutorial Key

- Command Line arguments will be displayed in this format
- 🞉 The party popper emoji signals the end of each set of instructions 🎉

Lesson objectives

- Create a geoJSON fence
- Learn how to filter for tweets with coordinates
- Apply a geoFence to a collection of Twitter data

Key Terms

- Terminal OS X Command Line
 - A text interface for your computer. Terminal receives commands, and then passes those commands on to the computer's operating system to run.
- Twarc
 - A command line tool and python library
- JSON JavaScript Object Notation
 - A minimal, human-readable format for structuring data. Twitter data is in JSON format.
- Hydrate Twarc Command
 - Reads a file of tweet identifiers and write out the tweet JSON for them using Twitter's <u>status/lookup API</u>.
- Shapely
 - A Python package for the manipulation and analysis of planar geometric objects

Table of Contents

| 1 |
|----|
| 2 |
| 2 |
| 2 |
| 3 |
| 8 |
| 8 |
| 10 |
| 11 |
| 12 |
| 13 |
| |

Introducing "Place" and "Coordinates"

In the first tutorial in this series, we learned that JSON is based on name-value pairs. Each name is an attribute that is paired with an associated value. There are two common attributes that deal with location in Tweet JSON. The first is "place", the second is "coordinates". It is

important to know the difference between the two attributes, because they are two completely different ways of representing location. The following definitions are <u>provided by Twitter</u>, and outline the exact definitions and uses of both the "place" and "coordinates" attributes.

<u>Place</u>

Places are specific, named locations with corresponding geo coordinates. When users decide to assign a location to their Tweet, they are presented with a list of candidate Twitter Places. When using the API to post a Tweet, a Twitter Place can be attached by specifying a place_id when posting the Tweet. Tweets associated with Places are not necessarily issued from that location but could also potentially be about that location.

<u>Coordinates</u>

The coordinates object is only present when the Tweet is assigned an *exact* location. If an exact location is provided, the coordinates object will provide a [longitue, latitude] array with the geographical coordinates, and a Twitter Place that corresponds to that location will be assigned.

Download the 1 October Twitter Data Collection

UNLV Access: If you are a UNLV faculty, student, or staff member, you can access the private, <u>full collection</u>.

Step 1: Download the collection

1. Start by visiting the UNLV library site to access the <u>private, full dataset</u>. Then download the zipped folder, 'oct1_postdedupe.zip'.

Index of /private

| | Name | Last modified | Size Description |
|----------|--------------------------|-------------------|------------------|
| P | arent Directory | | |
| | et1 24hour intervals.zip | 27-Sep-2018 12:57 | 15G |
| <u>o</u> | <u>ct1 images/</u> | 04-Oct-2018 22:28 | - |
| | ct1_postdedupe.zip | 27-Sep-2018 22:49 | 34G |
| re | adme.txt | 25-Oct-2018 16:07 | 2.7K |

Apache/2.2.15 (CentOS) Server at smedia.library.unlv.edu Port 80

- 2. Unzip the 'oct1_postdedupe' folder. The file we will be working with today is '20171003_20171008_postdedupe_chronological.json'.
- 3. Create a folder titled '1 October Twitter Collection'.



4. Open Terminal and move the '**20171003_20171008_postdedupe_chronological.json**' file to the '1 October Twitter Collection' folder you just created using the move command.



5. Navigate to your '1 October Twitter Collection' directory using the change directory command.



6. Rename the '20171003_20171008_postdedupe_chronological.json' file to '1_october_tweets.json' using the move command.



🎉 Great job! You successfully downloaded the 1 October Twitter Collection! 🎉

Public Access: All other users may access the <u>public set of tweet identifiers</u> by clicking here. Follow the instructions below to download the tweet IDs and then 'rehydrate' them using Twarc. You can learn how to rehydrate Twitter data in the <u>fourth tutorial</u> in this series.

It is important to note that Twitter limits users to 900 API status/lookup requests every 15 minutes. Each request can hydrate up to 100 Tweet IDs using the statuses/lookup REST API call. This means that every 15 minutes you will only be able to hydrate 90,000 tweets.

900 requests x 100 tweets = 90,000 tweets/15 min = 360,000 tweets/hour

This means that the full dataset of 14,108,104 tweets will take approximately 39 hours to hydrate.

Important: Rehydrating the tweet IDs will most likely not return the full dataset. Tweets and accounts that have been deleted since the collection period and tweets from now-private accounts will not be returned.

Step 1: Download the collection

1. Create a folder on your Desktop titled '1 October Twitter Collection'.



- 2. Visit the UNLV Library site to access the <u>public tweet IDs</u>. Save the IDs as a text file to your '1 October Twitter Collection' folder you created on your Desktop.
- 3. Change the name of the text file to 'tweet_ids_1_october'.

Great job! You successfully downloaded the 1 October Twitter Collection tweet IDs!

Step 2: Use Twarc to 'hydrate' the collection

- 1. Open the Terminal application (Located in the applications folder)
- Change directories by starting the command cd and then dragging your '1 October Twitter Collection' folder from your Desktop into Terminal. Hit return to complete the command once you have dragged the folder into terminal successfully.
 **Tip: Make sure to leave a space between the command cd and the filepath.*
- 3. Make sure you are in the right directory by entering the command pwd. You should be in your '1 October Twitter Collection' directory.

4. Hydrate your dataset by entering the following command:

twarc hydrate tweet_ids_1_october.txt > 1_october_tweets.json

Nice work! You now have your tweets in JSON format ready to go in your '1 October Twitter Collection' folder. You can always check your folder to confirm your .jsonl file is there.

Getting Started

Important: If you have already created a '1 October Twitter Collection' folder, you can skip this step.

Before creating a geoJSON file and a geoFence, go ahead and set up the master folder you will be saving your data to. Follow the steps below.

- 1. Open Terminal and navigate to your Desktop using the change directory command cd
- 2. Create a new directory and name it '1 October Twitter Collection' using the make directory command mkdir
- 3. Navigate to your new '1 October Twitter Collection' directory using the change directory command **cd**

🎉 Congrats! You're ready to start the tutorial. 🎉

Create a geoJSON Fence

To create a geoJSON Fence of Clark County, you will need a shapefile. This data is available through the U.S. Census Bureau. Follow the steps below to download the shapefile for Clark County, Nevada in 2017.

- You can download shapefiles for counties in the United States in 2017 by visiting <u>census.gov</u>. For the purposes of this tutorial, we have removed the Clark County shapefile which you can <u>download here</u>.
- 2. Once you have downloaded the data for Clark County, you can convert your files into geoJSON using <u>mapshaper</u>.

a. Select all files from the 'ClarkCountyNV' folder and upload them to mapshaper.



b. Select the 'Simplify' option and make sure to select 'prevent shape removal' before selecting 'Apply'.



c. Select 'Export' and then select the 'geoJSON' option before selecting the final 'Export' button.



🎉 Great job! You have successfully created a geoFence. 🎉

Prepare a geoJSON Fence

Some of the name-value pairs in your 'ClarkCountyNV.json' geoJSON file are not compatible with <u>Shapely</u>, which is a Python package for the manipulation and analysis of planar geometric objects. Follow the steps below to remove the non-compatible name-value pairs.

- 1. Go to <u>geojson.io</u>
- 2. Open your 'ClarkCountyNV.json' geoJSON file



3. Remove the following from the geoJSON file and then save as a new geoJSON file named 'clark_county_fence.geojson' to your '1 October Twitter Collection' directory.



| "METDIVFP": "", |
|----------------------------|
| "FUNCSTAT": "A", |
| "ALAND": 20439281450, |
| "AWATER": 438376844, |
| "INTPTLAT": "+36.2142575", |
| "INTPTLON": "-115.0138120" |
| } |



🎉 🕼 🞉 🞉 🎉 🎉 🎉 🎉 🎉 🎉

Install Shapely

In order to use Twarc to filter tweets using a geoFence, we will need to install <u>Shapely</u>. Open up Terminal and enter the following command to do so.

pip install Shapely



🎉 You did it! Go ahead and move on to the next section. 🎉

Filter For Tweets With Coordinates

Before we filter tweets using our geoFence, we are going to first narrow by tweets that contain coordinates. Follow the steps below to do so.

- 1. Open Terminal and navigate to your '1 October Twitter Collection' folder
- 2. Use Twarc to filter for tweets in the original collection that contain coordinates.

python ~/git/twarc/utils/geofilter.py 1_october_tweets.json --yes-coordinates > 1_october_yes_coordinates.json

Note: If you are working with the private collection of 1 October Twitter Data, the '1_october_yes_coordinates.json' file should contain 10,489 tweets.

🎉 Look at you go! You just filtered the <u>entire collection</u> for tweets with coordinates. 🎉

Use Twarc to Filter Tweets by a geoFence

You can use the geoFence of Clark County you just created to filter Tweets in the 1 October Twitter Data Collection. If you experience any issues, go ahead and <u>download this geoFence</u> of Clark County.

- 1. Open Terminal and navigate to '1 October Twitter Collection' folder
- 2. Use Twarc to filter your Tweets that contain coordinates within the geoFence of Clark County.

python ~/git/twarc/utils/geofilter.py 1_october_yes_coordinates.json --fence clarkcountyfence.geojson > 1_oct_clark_county.json

Note: If you are working with the private collection of 1 October Twitter Data, the '1_oct_clark_county.json' file should contain 5,792 tweets.

🎉 Awesome job! You just completed the tutorial. 🎉