COVID-19 Commodity Mapping Project

Royce Phillip Jr. and Jeong Huh

Motivation

COVID-19 is the worst pandemic in the past 100 years. It has made millions of people sick and as of May of 2020, more than 300,000 died from COVID-19 related symptoms. It has brought numerous life-changing experiences, such as stay-at-home orders, shortage of hospital beds and medical supplies and so forth. However, one of the issues that everyone around the world is facing is shortage of commodities such as certain foods, hygiene and personal protection products such as sanitizer, disinfectant wipes, mask and isopropyl alcohol. A lot of stockpiling occurred with panic purchasing and it has become difficult to locate and acquire certain items. Based on these facts, our team wanted to see if it is possible to locate these commodities with information available to everyone online and map essential items so it can help people get access to them as easy as possible.

Approach

Twitter

Twitter comes vital in this project since people will notify other twitter users of their findings on a broad based social media platform and it is easy to transfer the knowledge by retweeting. Also businesses who have their own twitter account will want to advertise their stocked commodity items for the general public so that they will come and not only buy the commodities but also buy other merchandise and products from their store. We wanted to collect many tweets using an API from people reporting stocked items from their local stores or from businesses themselves advertising commodity restocking.

Google Maps

After gathering tweets from twitter, we wanted to map the commodity locations on Google Maps since it’s the most popular and widely used map platform for people to look up locations and directions.

Process Flow

The main process flow can be generalized as these four steps. This can be applied to any kind of emergency response that requires crowd sourcing and location mapping.

1. **Emergency**

   This can be any kind of emergency such as forest fire, flood, pandemic that people will tweet about enmass and any kind response that is necessary to look up on a map. A good example can be hurricane Harvey in Texas (2017) or California wildfire (2019) where many people used social media platforms to share the news and urgency for help.

2. **Twitter, or any social media, data collection**

   As mentioned above, social media is a powerful platform to gather information about emergencies. Twitter is particularly well-suited for this situation due to its pervasiveness and it's speed to crowdsource information.

3. **Data filter**

   Not all information collected from social media can be transferred to useful information right out of the box. There are a lot of irrelevant social media feeds that can flood the data collection stage. Proper data cleaning and filtering is necessary. And in this particular project, geo location of the twees is the most critical piece of information to make the mapping function possible.

4. **Mapping**

   The mapping process involves packaging the data together for commodities, the store/business where they are stocked and using their GPS location to map it properly on Google Maps. Google Maps makes it relatively simple to use Javascript to perform mapping on their cloud services.

Data Collection Using Twitter API

A resource that was needed in the collection of data was using Twitter's own API. First we needed to sign up for a Twitter account. Then after we opened up a Twitter account we had to ask permission for an API. After filling out all the appropriate information they then give you four keys for you to use their API. You get an 'API key', 'API secret key', 'Access token', and 'Access token secret.' We used the free tier of Twitter's developer API which limits you to 100 tweets per request and a maximum of 250 requests per month. This is a great option to get a proof of concept, but if you were looking to grab more tweets it would be recommended to get one of the upgraded tiers.

The data collection was done by using a Python library called Tweepy. We used this library to help us collect tweets with certain topics, for example 'clorox wipes.' The files come in a large JSON file. After doing some more research on how to use tweepy we created a function that would easily turn the JSON file into a more readable format. We then turn that into a Pandas dataframe so that we could begin cleaning and learning more about the dataset. The Twitter API gives you a lot of information and we felt that not all of it was needed. We pulled information we felt would be of best use. The tweets, locations, name of account, geo location. We felt that these would be the best sources to give us the best information.

Data Analysis

The initial data analysis showed what could be some issues. Due to the fact that Twitter is getting large amounts of tweets and most of it is not informative towards commodities being available. We had to look over tweets to see what qualities of tweets were more useful than others. Some other issues we noticed were businesses that had available commodities, but weren't sending out descriptive tweets. This made it all more important to find the qualities that would distinguish the important tweets from the irrelevant ones.

We had to clean the tweets because of all the characters that are widely used on the platform. Removing "@" and "#" were the main characters, but also escape characters. We used regex and beautifulsoup to remove unnecessary characters from our text. We also removed stopwords like '"the", "me", and "I". We felt some of these wouldn't be helpful in finding the best descriptors of tweets. Once we had cleaned up the tweets we were able to use some word vectorizers to see what words showed up the most often. We used both CountVectorizer and TFIDFVectorizer to compare how they performed. We went with CountVectorizer because it was picking words that we felt was better fit to find the proper tweets. Then we ran a DBSCAN on the CountVectorized words. We used that to help us discover tweets that would be of use. Also to help us choose which coordinates that were best to use. Even though we did use this model it didn't perform as well as we hoped and we still had to manually read some tweets to see which ones were more helpful in finding available commodities.

Data Filter and Coordinate Generator

The data filtering process was done by Python libraries, Pandas and Numpy. The first step required to import the data into a DataFrame and clean up the dataset such as removing missing data and selecting data with GPS location. We considered location as one of the most important filtering criteria since without which, mapping becomes impossible. More specifically, GPS location was the key to connecting the commodities to a location on a mpa. There were many tweets that had generic location information such as, Los Angeles, California. However, these were left out due to the lack of specific GPS location. The GPS coordinates were buried in the ‘place’ column in the twitter data under the tag called ‘bounding box’. This gave the four corners of the GPS location of the person/business tweeting out the relevant information. It was necessary to pull out these coordinates from a long string of characters since it was difficult to read and feed this information one by one for hundreds of tweets to Google Maps. A python code was written to automatically parse the location string and extract the GPS coordinates. Unfortunately, trying to figure out whether the tweets was about actual commodities availability or just complaining about the lack of essential items needed human intervention to figure out what the tweet actually meant. So we had to read the tweets text in the DataFrame and hand pick hundreds of tweets that were conveying the necessary information about the commodity availability.

Google Maps

In order to do any kind of mapping process using Google Maps platform, one needs to make a developer’s account in Google Clouds, register a payment method and generate an API key. It is possible to just drop pins on the map manually, but if one would want to input all the necessary data and generate the map in an automatic way, it is impossible without a key. The commodity name, business name and GPS location was mapped on Google Maps using a Python library called gmaps. Although gmaps is an easy to use library for Python programmers, Jupyter notebook is necessary to look up the actual location of the commodities. Therefore, we borrowed a Javascript code from Google Maps Help website to use it for mapping onto Google Maps without any special libraries for anyone who has access to the internet to see and locate the items that they are interested in.

Summary

In the past few years, there have been several emergencies in the United States, such as massive hurricanes, wildfires and COVID-19 pandemic. In those situations, critical resources can make a real difference in one's life and also the location of those resources. We have used twitter as the main information gathering tool to locate available commodities during CVOID-19 pandemic and Google Map as the main delivery platform to share the information. We successfully opened a Twitter Developers account and generated an API key to download and collect thousands of tweets. Also wrote a code to screen and parse relevant data pertaining to commodities and filter and collect corresponding GPS locations for the location of commodities. In order to use Google Maps, it was also required to open a Developers account in Google Clouds and generate an API key. Gmaps Python library was used to map the location and relevant information on Google Maps and finally Javascript was used to open the Google Maps with commodities of interest online so anyone interested can look it up with the link. Overall the project showcased the feasibility of a proof-of-concept prototype of an emergency mapping API.

Future Work

When we started this project, we were hoping to see a commodity availability map of Austin, Texas. It did not take a long time to realize that goal was going to be difficult to achieve. The main reason was due to the scarcity of relevant tweets. The free tier API key from Twitter only allows you to download a limited number of tweets and only 7 prior days from the time of download. Knowing that only 3% of the tweets had GPS location attached to them, we would have needed tens of thousands of tweets to get enough data to generate a local commodity map. If this project had a budget behind it, I believe the data availability problem should have been easily overcome. However, with a more urgent emergency such as wildfire or hurricanes, I can envision having many more tweets being generated at a short amount of time in a relatively smaller geographical region. In such cases, I believe it would serve its purpose even with the limited API key from twitter.

The fact that a human had to go through the tweets and read the texts to filter out the essential ones also needs improvements. With more time and manpower, it would be possible to write a Natural Language Processing code to parse through the tweets and select the ones that were relevant to the commodities in stock. With a more urgent emergency, this would be essential to the success of the emergency mapping functionality for fast turnaround time.