Analysis of Carmen

Accuracy of geolocation via Carmen
as compared to tweets with known geo tags

Sonal Sannigrahi

July 25, 2019

1 Methodical Approach

To justify the use of the Carmen library in geolocating tweets, we must first address in fact how accurate is Carmen? To do so, we collect a large enough sample of tweets, which have geolocation enabled, and compute the difference between the real location given by the geo tags of the tweet and the one computed by using the Carmen library.

Each tweet in its JSON format has several attributes, some of which will be of interest to us in order to create the sample. According to the Twitter API, for each tweet we have the following relevant attributes: **"geo"** and **"coordinates"**. For tweets with no geo tags, these two attributes are set to null. Using this information we can filter a real time stream of tweets to create a smaller sample of those with known locations.

Once we filter the incoming sample, we can create a copy through which we test the accuracy of the geolocation. For all of the tweets in this filtered sample we create a new copy in which we set the "geo" and "coordinates" attributes to null. In this way, the geolocation done by Carmen will be through dictionary search and mapping rather than the true location provided by Twitter.

2 Implementation and Results

The above method was implemented in Python and the following files were created: three JSON files which were the all the tweets collected, the tweets with geo location enabled, those tweets geo located via carmen respectively and lastly, a .csv file which noted the tweet id, the city as geolocated by carmen, the city as identified by the known coordinates, similarly for the county and state, and lastly the distance between the geolocated data and the real location by using the coordinates of both.

Upon generating a sample of real time tweets, the data retrieved was as follows:

Total number of tweets with geo tags streamed: 9,708
Percentage accuracy at the city level: 18.33 %
Percentage accuracy at the county level: 30.15 %
Percentage accuracy at the state level: 82.00 %
Average Haversine distance error: 385.6 km
Average Euclidean distance error: 370.9 km

3 Further analysis of Error distribution

To understand the error distribution better, below is a plot of the Error against the tweets represented by numbers signaling each data point. In the density plot below, we observe that close to 60 % of the data is geolocated with an error in distance in an acceptable error range. Out of the remaining 40 % we notice that there are few data points (much less than 1 %) are geolocated with very high error (larger than 2000 km).

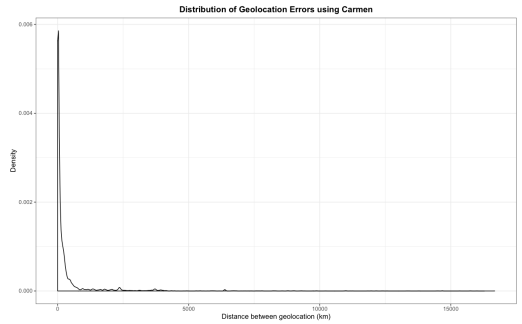

Figure 1: Complete Density Plot

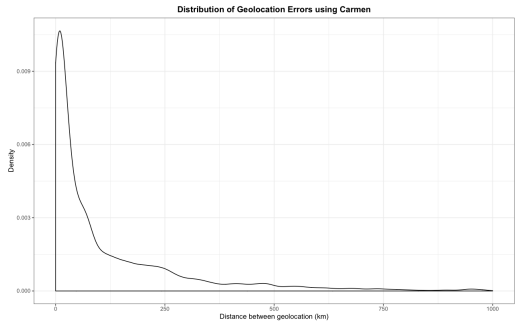

Figure 2: Density Plot zoomed for 1000 entries

However, these don't make up a large portion of the data and thus mostly all of the data is geolocated with acceptably low errors.

4 Note on computing distance between the two points

4.1 Haversine Distance

In order to compute the distance using the latitude and longitude information, the **Haversine formula** was used. This computes the distance in the **as the crow flies** method, i.e the great circle distance. An important assumption made about the formula is the spherical shape of the Earth. This might lead to significant errors when considering distances towards the poles, however in our sample we consider the continental United States and thus there is no significant error produced. The formula proceeds as follows

The haversine of an angle is given by: $hav(\theta) = sin^2(\frac{\theta}{2})$.
The haversine of a central angle is defined as: $hav(\frac{d}{r})$. The central angle for a sphere is given below.

The haversine is further calculated by:

$$hav(\frac{d}{r}) = hav(\phi_2 - \phi_1) + cos(\phi_1)cos(\phi_2)hav(\lambda_2 - \lambda_1) = h$$

Here, r is the radius of the Earth (6371 km), d is the required quantity which is the distance between the two points, ϕ_1 and ϕ_2 are the latitude of the two points, and λ_1 and λ_2 are the longitude of the two points. Now, solving for d we get:

$$d = rhav^{-1}(h) = 2rsin^{-1}\sqrt{sin^2(\frac{\phi_2 - \phi_1}{2}) + cos(\phi_1)cos(\phi_2)sin^2(\frac{\lambda_2 - \lambda_1}{2})}$$

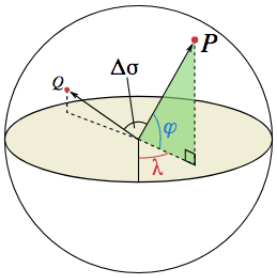

Figure 3: Central Angle of a Sphere

4.2 Euclidean Distance

In addition to the great-circle distance as calculated by the Haversine formula, the final results also include the calculation of a Euclidean distance. Considering two points on the Earth as 3-D points given by:

$$A = (x_1, y_1, z_1) \text{ and } B = (x_2, y_2, z_2)$$

We calculate the **Euclidean Distance** by the following formula:

$$Dist^2 = \sum_{n=1}^{3} (A_i - B_i)^2$$

This gives the **ordinary straight line** distance between two point i.e the distance through the Earth. This is commonly called the **as the worm digs** distance.

In order to apply this method with latitude and longitude information, we must first convert latitude and longitude into valid coordinates. This was done by standard spherical coordinates:

$$x = Rsin(\theta)cos(\phi), y = Rsin(\theta)sin(\phi), z = Rcos(\theta)$$

Here, θ represents the latitude, ϕ represents the longitude, and R is the radius of the Earth in km.

5 Final Remarks

Looking at the final results, we make some crucial remarks regarding the accuracy of Carmen as a geolocation tool. Based n the accuracy percentages presented above, it is safe to say that Carmen is sufficiently accurate at the State level and resultingly, the Country level.

When considering counties or cities however, Carmen doesn't seem to be a good geolocater. However, coupling the dictionary matching that Carmen does along with the known geo tags that Twitter provides a roughly 30 % accuracy rate (at the county level) is sufficiently close. Further note that these accuracy percentages check for exact matches in the names of counties by reverse geocoding the latitude and longitude data. In effect, some neighbouring places are deemed inaccurate. For example, considering a county along the border of New York and another county along the border of New Jersey whilst being 2.5 km apart are said to be inaccurate based on this method. Given this, we still can't deem Carmen accurate enough at the city level as a 18% match rate is not justifiably low.

Concluding, we have tested the accuracy of Carmen and come to the finding that Carmen can be safely used to geolocate Tweets down to the county level accurately (considering the continental United States).