OpenStreetMap Data Case Study

Ken Musante, DAND, October 2017

I. Problems encountered:
   a. Needed Geographic Area which was at least 50 mb. Consequently, I selected the SF Bay Area even though I am not that familiar with the area.
   b. Did not fully understand what I needed to do to complete problem set: This involved multiple reads and reviews of the Forums as well as the 1 on 1 help
   c. 1 on 1 help was a new concept: It took me a while to understand how to get information to experts on the 1 on 1 desk
   d. Size of the file was massive. I tried several times to manipulate the original OSM file and was very frustrated and wasted a lot of time. I appreciate the 'iterparse' command more than you know.
   e. Understanding how/why to apply the Lessons to the data. Some of the lessons were not communicated in a way that allowed me to easily understand how to apply it to a new file. This took a lot of additional work and time.
   f. Unable to use Jupyter Notebook to manipulate the csv files. I tried and failed several times before understanding that an easier way was to just manipulate them directly in sqlite3.

II. OverView of Data
   a. Jupyter Notebook is attached along with the coding for the steps described below. I was very pleased to get additional practice in Jupyter Notebook. I found it easy and helpful to use that tool to document work and pleased I could apply prior lessons to this project. While I wont repeat the code found in the Jupyter Notebook, here are the steps I undertook.
      i. First step was to create several different sample file sizes for easy manipulation. This was helpful as I could see file in Sublime
      ii. My next step was to count tags and put in Dictionary so I could understand file sizes
      iii. Next I queried specific fields so I could get an idea of the users(dictionary), location names, restaurant types(dictionary), web URLS and cities (dictionary)
      iv. I then updated the street names such that the last word in the street names was consistent (as we did in the class lesson). Many of the street names had non standard endings so this program was very helpful/useful. I also updated postal codes such that all postal codes were 5 digits (truncated the 9 digts) and out-sorted any non digits.
      v. I then moved the data into 5 csv files:
         1. NODES_PATH = "nodes.csv"
         2. NODE_TAGS_PATH = "nodes_tags.csv"

3. WAYS_PATH = "ways.csv"
4. WAY_NODES_PATH = "ways_nodes.csv"
5. WAY_TAGS_PATH = "ways_tags.csv"

vi. I used the Jupyter Notebook to view the csv files

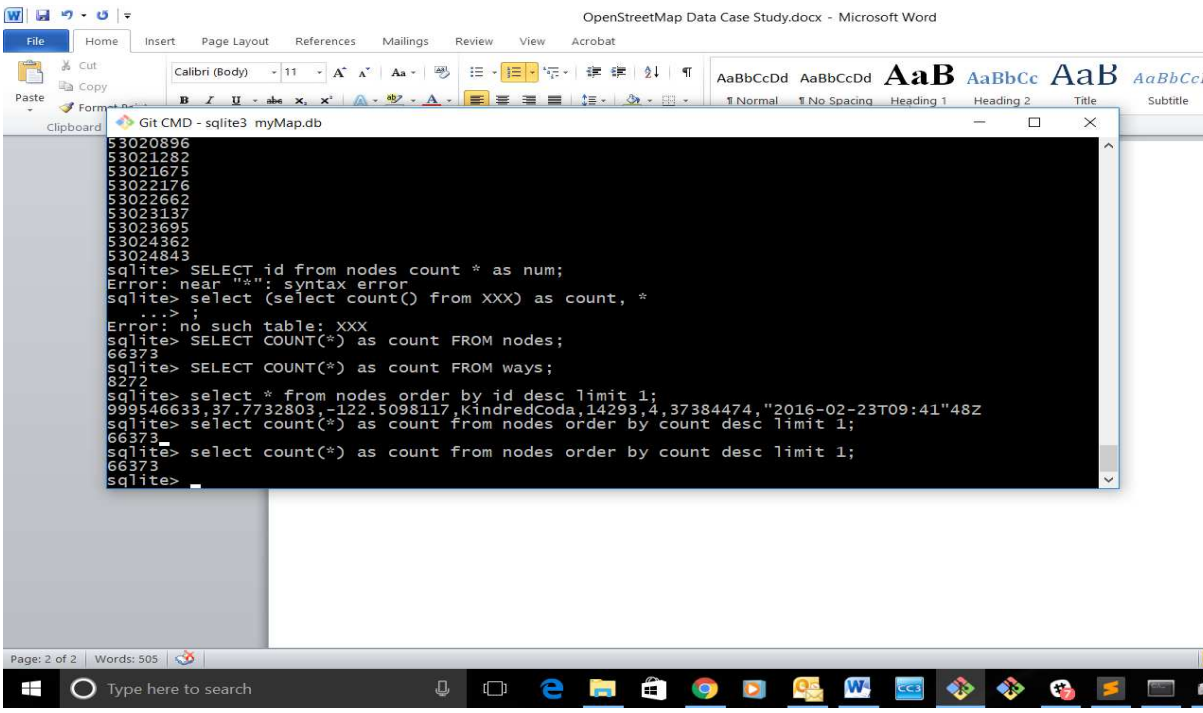
b. I attempted to use Jupyter Notebook to set up the Tables. I failed and found it much easier to set them up in sqlite3. Running queries directly in sqlite3 was easy. Here are my queries and response:

i. Size of the file: I'll provide a couple of responses to ensure I understand your query-

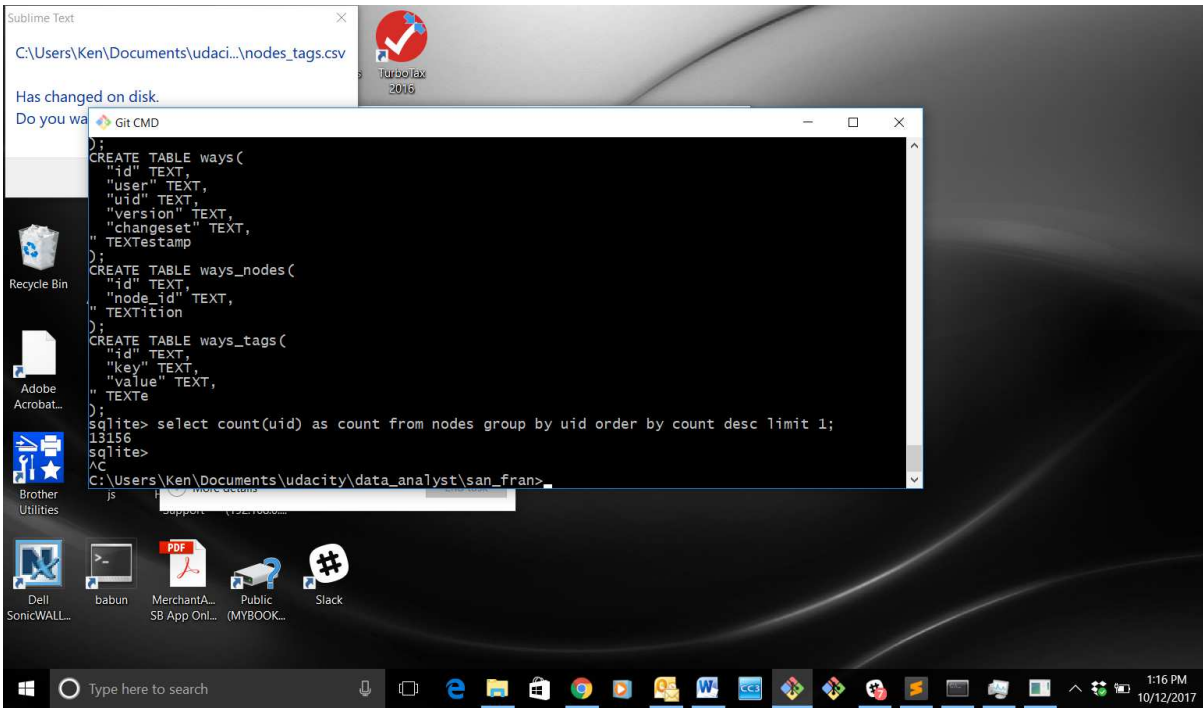
1. San Francisco.osm-1.38 GB
2. San Francisco Data Base-10.1 MB
3. Samplek100.osm-13.6 MB
4. Nodes.csv-5.5 MB
5. Nodes_tags.csv-1.0 MB
6. Ways.csv-97 KB
7. Ways_nodes.csv-1.9 MB
8. Ways_tags.csv-600 KB

This could also mean the number of rows. I found 2 ways to do this. Both solutions resulted in 66,373 rows:

ii.

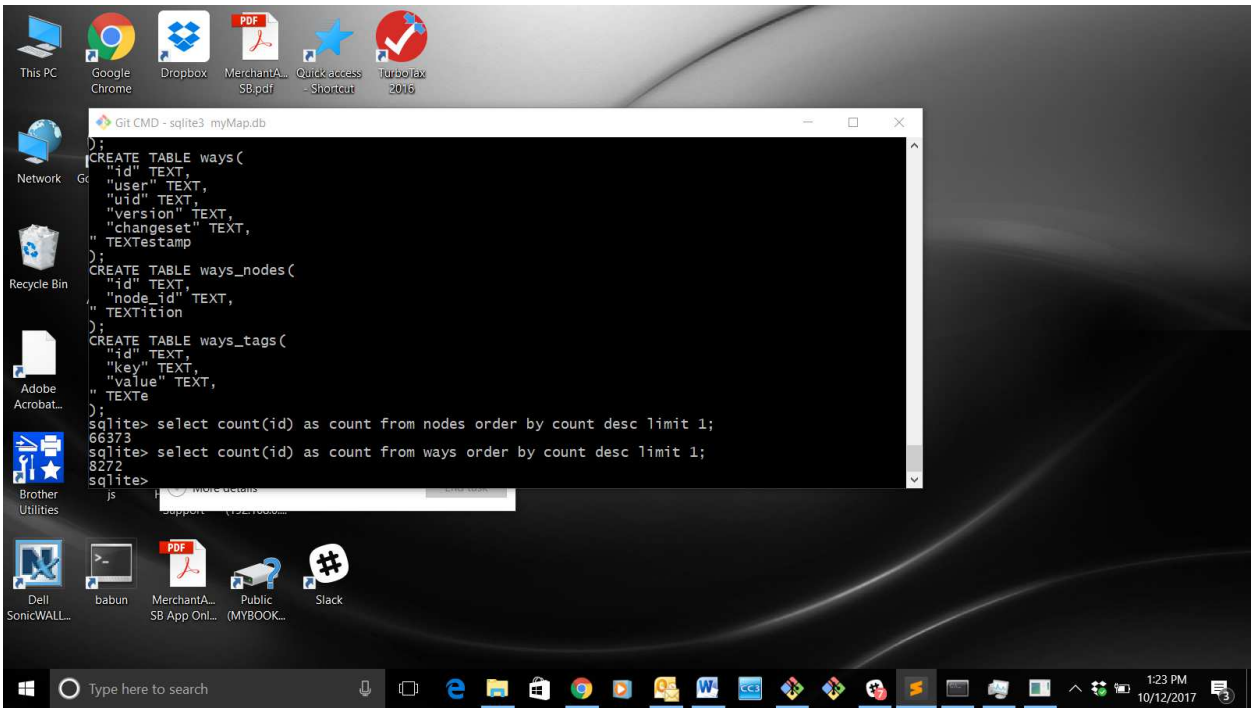


Number of unique users: 13,156

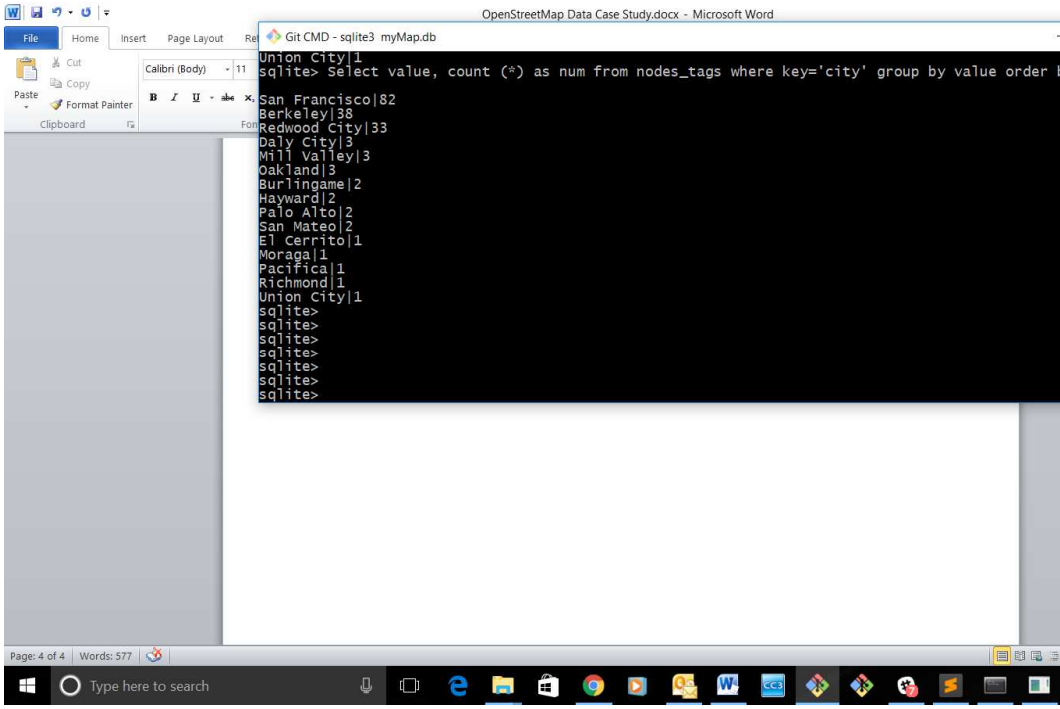


iii. Number of nodes and ways:

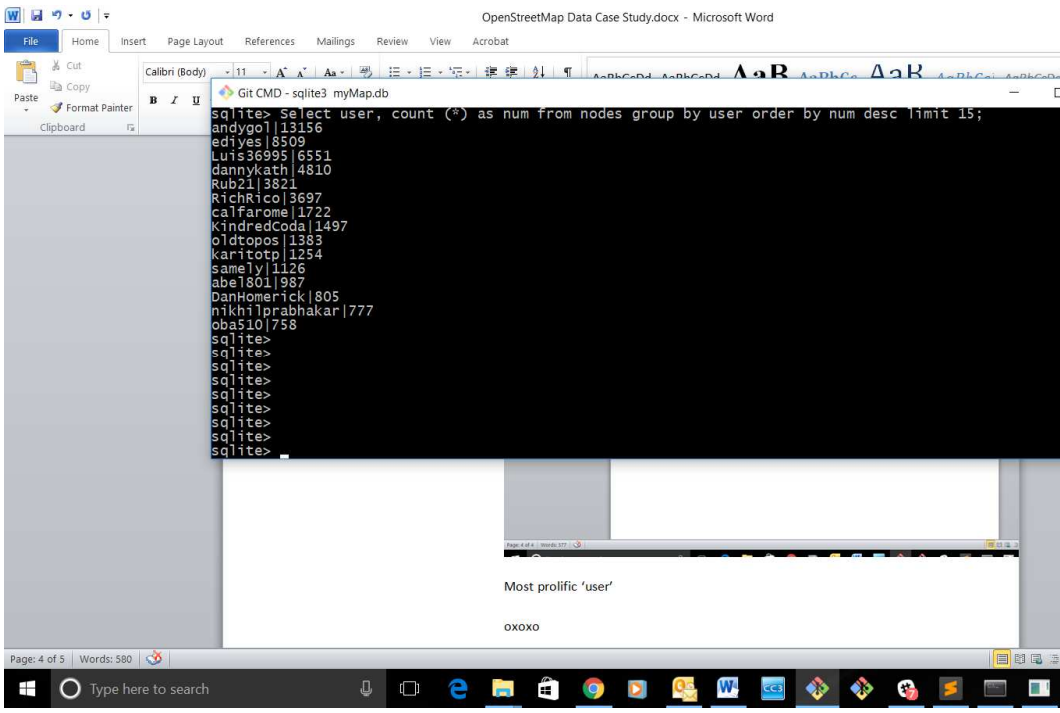
1. 66,373 node ids
2. 8,72 way ids



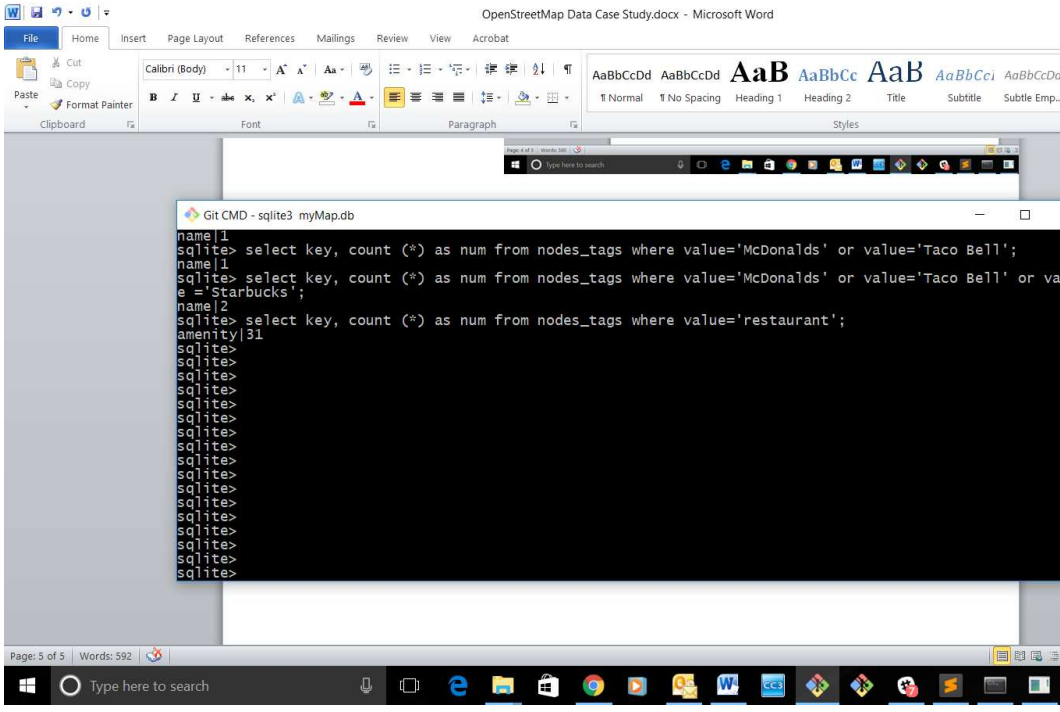
iv. Cities and frequency of appearance:



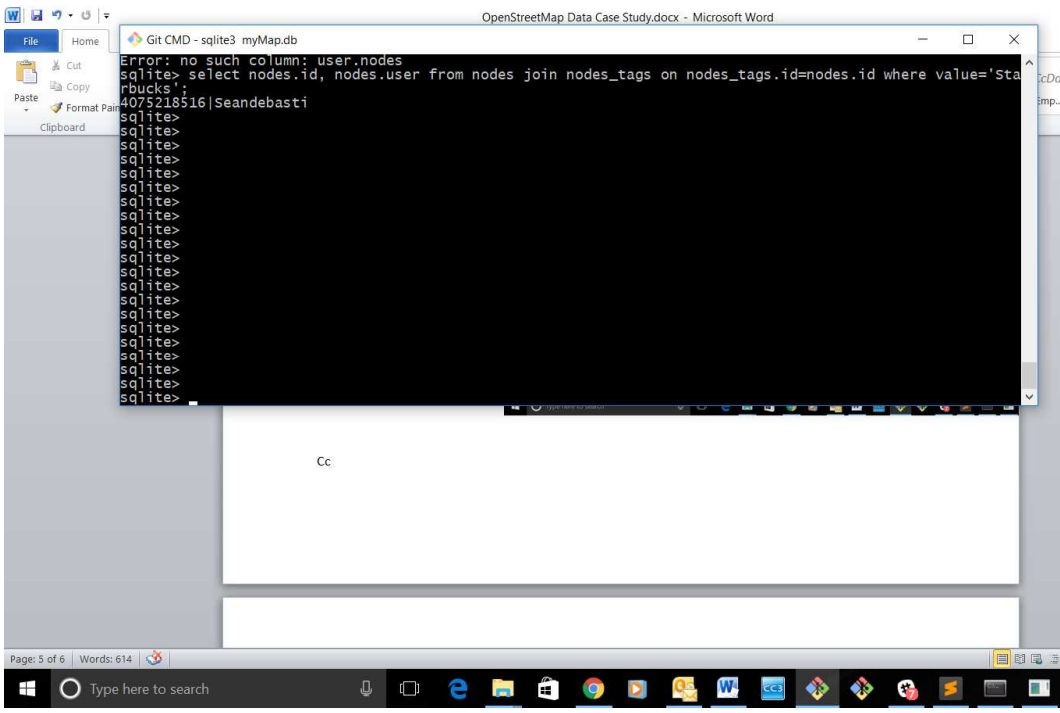
Most prolific ‘user’



We had only 1 Taco Bell and 0 Mcdonalds and 1 Starbucks. The solution below indicates there were 2 when I asked for Taco Bell, McDonalds or Starbucks and our sample had 31 restaurants.



And the Starbucks entry was made by Seandebasti



III. Conclusion

a. The biggest issue in working with the OpenStreets database is that it was so large it was difficult to work with unless I took a smaller sample. While this was helpful, I wish I had a more powerful PC to be able to run queries on the entire dataset. Certainly working with the data once it was in the CSV files was quick and easy to manipulate. It also might have been useful to divide up the csv files by City so even though I could not look at an entire data set I could look at an entire city.