OpenStreetMap Project Report
Data Wrangling with MongoDB

Name: Sheethal Mohan

Map Area: Raleigh, North Carolina

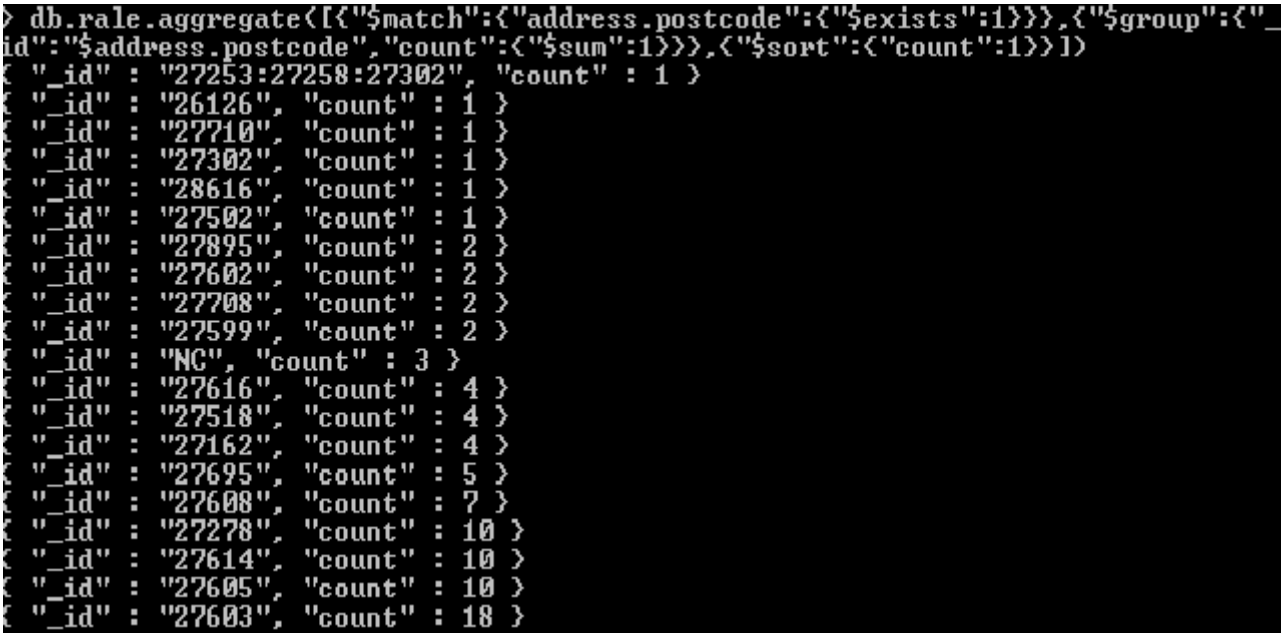
https://s3.amazonaws.com/metro-extracts.mapzen.com/raleigh_north-carolina.osm.bz2

The dataset of my hometown Thalassery, Kerala, India was very small(less than 50 MB). So, I decided to take a place which I have visited with my family. I choose Raleigh, North Carolina, since I went there during my last vacation and was there for over a month. I am still familiar with the places in Raleigh, so I thought it would be interesting to work with this dataset.

After extracting the compressed version of my osm file, I passed it my python files: mapparser.py, tags.py, users.py, audit.py and data.py.

1. **Problems encountered in the map**

- I ran it against my audit.py code and corrected the over-abbreviated street names.
  For example there where street names like,
  (a) E Rosemary St.
  (b) W. Pettigrew St.
  Buck Jones Rd
  Using the update function of audit.py, I modified these street names by iterating over each word of these street names. After the modifications all the street names were rid of these abbreviations. For example, these turned to
  (a) East Rosemary Street
  (b) W. Pettigrew Street
  (c) Buck Jones Road
  I went on to find any discrepancies in the JSON file uploaded into MongoDB. So I performed a queries on the postal code. The output was:



```
Type "it" for more
> it
{ "_id" : "27610", "count" : 18 }
{ "_id" : "27604", "count" : 23 }
{ "_id" : "27607", "count" : 27 }
{ "_id" : "27613", "count" : 31 }
{ "_id" : "27617", "count" : 33 }
{ "_id" : "27703", "count" : 50 }
{ "_id" : "27516", "count" : 51 }
{ "_id" : "27704", "count" : 55 }
{ "_id" : "27713", "count" : 55 }
{ "_id" : "27560", "count" : 72 }
{ "_id" : "27517", "count" : 73 }
{ "_id" : "27707", "count" : 80 }
{ "_id" : "27601", "count" : 85 }
{ "_id" : "27606", "count" : 94 }
{ "_id" : "27513", "count" : 98 }
{ "_id" : "27511", "count" : 116 }
{ "_id" : "27514", "count" : 175 }
{ "_id" : "27510", "count" : 272 }
{ "_id" : "27615", "count" : 362 }
{ "_id" : "27510", "count" : 272 }
{ "_id" : "27615", "count" : 362 }
{ "_id" : "27705", "count" : 504 }
Type "it" for more
> it
{ "_id" : "27701", "count" : 672 }
{ "_id" : "27519", "count" : 868 }
{ "_id" : "27609", "count" : 1150 }
{ "_id" : "27612", "count" : 1746 }
> _
```

I found a problem in the result of the above query. The range of postal codes seemed to vary a lot according to me, which I didn't find it to be correct, considering a particular region. So, I googled the postal code for Raleigh, NC. I found the link below very useful.

http://www.city-data.com/zipmaps/Raleigh-North-Carolina.html

From this link, I understood that all the Raleigh, NC postal codes begin with "276" and the postal codes adjacent to the city boundary are 27587. There are several codes that doesn't belong to the Raleigh city (postal code: 27519, with a count of 868, is for Cary, North Carolina) and includes invalid postal codes entries like NC, 27253:27258:27302. I was curious to know to which all are the other cities that have been listed in this dataset. Finding it out manually, for each postal code, would have been a tedious process. So, I decided to run a query.

```
> db.rale.aggregate([{"$match":{"address.city":{"$exists":1}}},{"$group":{"_id":
"$address.city","count":{"$sum":1}}},{"$sort":{"count":1}}])
{ "_id" : "durham", "count" : 1 }
{ "_id" : "Chapel Hill, NC", "count" : 1 }
{ "_id" : "cary", "count" : 1 }
{ "_id" : "Apex", "count" : 1 }
{ "_id" : "Wake Forest", "count" : 2 }
{ "_id" : "chapel Hill", "count" : 2 }
{ "_id" : "raleigh", "count" : 2 }
{ "_id" : "Morrisville", "count" : 98 }
{ "_id" : "Chapel Hill", "count" : 211 }
{ "_id" : "Carrboro", "count" : 270 }
{ "_id" : "Raleigh", "count" : 764 }
{ "_id" : "Durham", "count" : 1242 }
{ "_id" : "Cary", "count" : 1706 }
```

That query gave me the result above.

Since majority of the postal codes were close to 276xx, I got this intuition that the remaining postal codes could be the surrounding cities of this metro. This led me to search for Raleigh Metropolitan Area. That led me to the Research Triangle. I got to study about that from the link below.

http://en.wikipedia.org/wiki/Research_Triangle

From the above link, we get to know that the Research Triangle, commonly referred to as simply "The Triangle", is a region in the Piedmont of North Carolina in the United States, anchored by North Carolina State University, Duke University, University of North Carolina at Chapel Hill, and the cities of Raleigh and Durham and the towns of Cary and Chapel Hill. The eight-county region, officially named the Raleigh–Durham–Chapel Hill CSA, comprises the Raleigh and Durham–Chapel Hill metropolitan. So, we can come to a conclusion that this dataset is for the Research Triangle.

- After running mapparser.py, my result is as follows:
  defaultdict(<type 'int'>, {'node': 2520066, 'nd': 2779231,
  'bounds': 1, 'member': 7548, 'tag': 810504, 'relation': 710,
  'way': 210721, 'osm': 1})
  I decided to run queries to check if these values were correct.

```
> db.rale.find({"type":"node"}).count()
2520066
>
```

  Yes, the result of the query did match with that of my mapparser.py output.

Overview of the Data

File size

raleigh_north-carolina.osm → 494 MB
raleigh_north-carolina.osm.json → 559 MB

MongoDB queries

Total number of documents

```
> db.rale.find().count()
2730787
> _
```

Total number of unique users

```
> db.rale.distinct("created.user").length
662
>
```

Top 10 contributing users

```
> db.rale.aggregate([{"$group":{"_id":"$created.user","count":{"$sum":1}}},{"$so
rt":{"count":-1}},{"$limit":10}])
{ "_id" : "jumbanho", "count" : 2143028 }
{ "_id" : "woodpeck_fixbot", "count" : 129075 }
{ "_id" : "yotann", "count" : 67842 }
{ "_id" : "JMDeMai", "count" : 62936 }
{ "_id" : "runbananas", "count" : 43494 }
{ "_id" : "sandhill", "count" : 26199 }
{ "_id" : "FIM", "count" : 22759 }
{ "_id" : "dufekin", "count" : 20678 }
{ "_id" : "TIGERcnl", "count" : 20503 }
{ "_id" : "MikeInRaleigh", "count" : 15435 }
>
```

Number of users who have made only one post

```
> db.rale.aggregate([{"$group":{"_id":"$created.user","count":{"$sum":1}}},{"$gr
oup":{"_id":"$count","num_users":{"$sum":1}}},{"$sort":{"_id":1}},{"$limit":1}])

{ "_id" : 1, "num_users" : 128 }
>
```

Additional ideas about users

- The Top user(jumbanho) contibution is: 78.48%
- Top 3 users contribution is: 83.45%
- Top 10 users contribution is: 93.45%

Additional MongoDB Queries

Top 10 amenity types

```
> db.rale.aggregate([{"$match":{"amenity":{"$exists":1}}},{"$group":{"_id":"$ame
nity","count":{"$sum":1}}},{"$sort":{"count":-1}},{"$limit":10}])
{ "_id" : "parking", "count" : 1841 }
{ "_id" : "place_of_worship", "count" : 543 }
{ "_id" : "bicycle_parking", "count" : 510 }
{ "_id" : "restaurant", "count" : 491 }
{ "_id" : "fire_hydrant", "count" : 326 }
{ "_id" : "fast_food", "count" : 248 }
{ "_id" : "school", "count" : 226 }
{ "_id" : "fuel", "count" : 200 }
{ "_id" : "bench", "count" : 123 }
{ "_id" : "bank", "count" : 110 }
>
```

Top 5 fast food restaurants

```
> db.rale.aggregate([{"$match":{"amenity":"fast_food"}}, {"$group":{"_id":"$name
","count":{"$sum":1}}}, {"$sort":{"count":-1}},{"$limit":5}])
{ "_id" : "McDonald's", "count" : 21 }
{ "_id" : "Subway", "count" : 16 }
{ "_id" : "Burger King", "count" : 11 }
{ "_id" : "Wendy's", "count" : 10 }
{ "_id" : "Bojangles", "count" : 8 }
>
```

Top 5 café shops

```
> db.rale.aggregate([{"$match":{"amenity":{"$exists":1},"amenity":"cafe"}},{"$gr
oup":{"_id":"$name","count":{"$sum":1}}},{"$sort":{"count":-1}},{"$limit":5}])
{ "_id" : "Starbucks", "count" : 16 }
{ "_id" : null, "count" : 4 }
{ "_id" : "Caribou Coffee", "count" : 3 }
{ "_id" : "Joe Van Gogh", "count" : 3 }
{ "_id" : "Dunkin Donuts", "count" : 2 }
>
```

Total number of café shops

```
> db.rale.find({"amenity":"cafe"}).count()
93
>
```

Top 3 religious groups

```
> db.rale.aggregate([{"$match":{"amenity":{"$exists":1},"amenity":"place_of_wors
hip"}},{"$group":{"_id":"$religion","count":{"$sum":1}}},{"$sort":{"count":-1}},
{"$limit":5}])
{ "_id" : "christian", "count" : 480 }
{ "_id" : null, "count" : 48 }
{ "_id" : "jewish", "count" : 5 }
{ "_id" : "muslim", "count" : 4 }
{ "_id" : "unitarian_universalist", "count" : 2 }
>
```

Top 10 cuisines

```
> db.rale.aggregate([{"$match":{"amenity":{"$exists":1},"amenity":"restaurant"}}
,{"$group":{"_id":"$cuisine","count":{"$sum":1}}},{"$sort":{"count":-1}},{"$limi
t":8}])
{ "_id" : null, "count" : 185 }
{ "_id" : "mexican", "count" : 33 }
{ "_id" : "american", "count" : 31 }
{ "_id" : "pizza", "count" : 31 }
{ "_id" : "italian", "count" : 26 }
{ "_id" : "burger", "count" : 23 }
{ "_id" : "chinese", "count" : 21 }
{ "_id" : "regional", "count" : 14 }
>
```

Unfortunately several types of cuisines are not documented.

I noticed that there were several streets, landmarks and buildings named after royalty like “King”, “Queen”, “Prince”, and “Princess”.

```
> db.rale.distinct("name",{"name":{"$in":[/^King\s/,/^Queen\s/,/^Prince\s/,/^Pri
ncess\s/]}})
[
        "King Pond",
        "King Pond Dam",
        "King James Road",
        "King William Court",
        "Prince Street",
        "King Charles Road",
        "King Street",
        "Princess Anne Drive",
        "King Mountain Court",
        "Queen Street",
        "King William Rd",
        "King James Court",
        "Queen Elizabeth Drive",
        "Prince George Lane",
        "King Cross Court",
        "King Charles Lane",
        "King Lawrence Road",
        "King Richard Road",
        "Queen Annes Drive",
        "Princess Anne Road",
        "King William Road",
        "Prince Oliver Place",
        "Prince Farm Road"
]
>
```

We can see below that these street names have names of royalty.

```
> db.rale.distinct("address.street",{"address.street":{"$in":[/^King\s/,/^Queen\
s/,/^Prince\s/,/^Princess\s/,/^Royal\s/]}})
[
        "Royal Sunset Drive",
        "Royal Drive",
        "King James Road",
        "King William Court",
        "Prince Street",
        "King Charles Road",
        "King Street",
        "Princess Anne Drive",
        "King Mountain Court",
        "Queen Street",
        "Royal Pines Dr",
        "King William Rd",
        "King James Court",
        "Royal Anne Lane",
        "Queen Elizabeth Drive",
        "Prince George Lane",
        "King Cross Court",
        "King Charles Lane",
        "King Lawrence Road",
        "Royal Oaks Drive",
        "King Richard Road",
        "Queen Annes Drive",
        "Royal Street",
        "Royal Club Drive",
        "Princess Anne Road",
        "King William Road",
        "Royal Wood Court",
        "Prince Farm Road"
]
>
```

Other ideas about the dataset

One of the main problem we are facing over here is the accuracy of the dataset. It seems to me that most of the users have submitted the information through computers. Rather than that, if we had used mobile phones, we could have used its GPS feature to accurately submit the information regarding the respective location.

I believe that the data structure is flexible enough to incorporate a vast multitude of user generated quantitative and qualitative data beyond that of simply defining a virtual map. I believe that extending this open source project to include data such as user reviews of establishments, subjective areas of what bound a good and bad neighbourhood, housing price data, school reviews, walkability, quality of mass transit, and on would form a solid foundation of robust recommender systems. These recommender systems could aid users in anything from finding a new home or apartment to helping a user decide where to spend a weekend afternoon.

Unfortunately, it appears that, at least for the area of the world that I analysed, the mapping data is far too incomplete to be able to implement such recommender systems. I believe that the OpenStreetMap project would greatly benefit from visualizing data on content generation within their maps. For example, a heat map layer could be overlaid on the map showing how frequently or how recently certain regions of the map have been updated. These map layers could help guide users towards areas of the map that need attention in order to help more fully complete the data set.

Conclusion

While working with the Raleigh, I found that most of the data contributed by the users to be not standardised. This dataset was meant for the Raleigh city, but, it seems to be for the Research Area. By using my audit.py and data.py, I hope I have cleaned my dataset well and I hope that many benefit with these findings and cleaner data.