DataWrangling-Project

February 19, 2018

1 OpenStreetMap

1.0.1 Map Area:

```
Atlanta, Georgia, US

*  https://www.openstreetmap.org

Atlanta is my hometown so I have chosen this map area. I am very much interested to
```

1.0.2 Problems Encountered in the Map:

Initially I have downloaded Atlanta_georgia.osm, due to large file size I have taken a small sample size file to insert into database. Upon analyzing the data I have encountered following problems which are as follows:

```
* Street names were represented in various ways on secondary "k" tags("addr:street"
    Bldv: 1
    blvd: 1
    Blvd: 103
    Blvd.: 2
    Boulevard: 654
* Postalcodes of Georgia state starts from 30002 to 39901 but there was one zipcode
  few codes that does not belong to Georgia.
    80083: 9(Ukraine)
    58502: 1(North Dakota)

* Repitition of state name in secondary tags in "v" tag field.
   <tag k="is_in" v="Gwinnett,Georgia,Ga.,GA,USA" />

* To find the route, users have used Tiger GPS data hence the secondary tags "k" ha
    <tag k="tiger:cfcc" v="A41" />
    <tag k="tiger:county" v="Chambers, AL" />
    <tag k="tiger:reviewed" v="no" />
    <tag k="tiger:zip_left" v="36863" />
    <tag k="tiger:name_base" v="8th" />
    <tag k="tiger:name_type" v="St" />
```

```
<tag k="tiger:name_direction_prefix" v="N" />
```

1.0.3 Auditing Street Names:

Street names that were not spelled correctly is changed to street names that was entered by users most frequently. For example, Trl has lesser count than Trail so the value in secondary tag field is changed to Trail. This example is taken from a sample file, primary tag is "way" and secondary is child of "way" which is "tag". Trl: 2303 Trail: 3370

Output:
id,key,value,type 35625538,street,Rampley Trail,addr

1.0.4 Postal codes from atlanta_georgia.osm

After analyzing the data, I have to make sure whether the postal codes are from Georgia or not. Few of them are not from Georgia and some codes were inconsistent. Sample file that contains postal codes from Georgia in OSM are as follows:

Output: 30012: 1 30013: 2 30016: 1 30022: 4 30033: 1 30034-4894: 1 30035: 1 30035-2355: 1 30058: 1 30062: 1 30066: 1 30067: 1 30068: 1 30075: 12 30076: 10 30080: 1 30082: 1 30083: 1 30094: 4 30101: 1 30102: 4 30107: 3 30114: 13 30115: 6 30127: 1 30132: 8 30134: 3 30141: 5 30143: 1 30153: 1 30157: 8 30180: 1 30183: 1 30184: 1 30188: 16 30189: 7 30213: 11 30268: 3 30291: 7 30296: 2 30305: 4 30306: 4 30307: 3 30308: 3 30310: 8 30311: 9 30312: 2 30314: 7 30315: 7 30316: 4 30318: 7 30319: 1 30322: 1 30324: 2 30326: 1 30327: 8 30328: 5 30331: 13 30337: 2 30342: 8 30344: 9 30349: 12 30350: 3 30354: 4 30601: 1 30605: 1

1.0.5 Overview of dataset

This section describes the size, number and type of tags that are present in the dataset as well as queries for number of nodes, ways and unique users in the dataset.

Size of dataset 8594608179 -rw-r–r–@ 1 spoonepa staff 2.4G Feb 6 12:12 atlanta_georgia.osm

Number and type of tags in atlanta_georgia.osm {'bounds': 1, 'member': 46223, 'nd': 13537857, 'node': 11990171, 'osm': 1, 'relation': 5447, 'tag': 6371820, 'way': 884325}

Queries for number of nodes, ways and unique users in the dataset.

Number of nodes: sqlite> select count(*) from nodes; 11990171

Number of nodes_tags: sqlite> select count(*) from nodes_tags; 1821590

Number of ways: sqlite> select count(*) from ways; 884325

Number of ways_nodes: sqlite> select count(*) from ways_nodes; 13537858

Number of ways_tags: sqlite> select count(*) from ways_tags; 4517827

Unique users in the dataset: sqlite> select count(distinct uid) as uniqueusers from (select nodes.uid from nodes union select ways.uid from ways); 2695

1.0.6 Additional ideas about the dataset

- Based on the postal codes from OSM, we can make changes to the data so that the postal codes will be five digit and the postal codes belong to Georgia. For example, GA 30601: 1 (GA can be ignored). This process might result to couple of problems:

1. If there are additional character or number in the postal code we can remove it but that might result to loss of data. For example, 300313: 2 It is a six digit number and if we remove last digit from the postal code, still the postal code belongs to Georgia but due to typo mistakes (if the postal code 30033 is required by the user) this process might result to data loss.
2. If there is a postal code that does not belong to Georgia and if we remove the postalcode, which leads to loss of data.

- Auditing the dataset by editing the highway exits that were merged into addr:street values. For example, 92: 31 400: 2
  Both 92 and 400 are highways.

1.0.7 Additional data exploration

Top ten amenities: sqlite> select value, count() *from nodes_tags . . . > where key='amenity' . . . > group by value . . . > order by count*() . . . > desc limit 10;

place_of_worship,3982 grave_yard,2044 school,2018 restaurant,1081 fast_food,579 bench,374 fuel,354 fire_station,242 bicycle_parking,230 atm,224

Top ten nodes in a way: sqlite> select id, count() *from ways_nodes . . . > group by id . . . > order by count*() . . . > desc limit 10;

34828730,1998 33122855,1997 33124405,1997 33164538,1997 34427163,1997 34427183,1997 34427189,1997 34427215,1997 34427221,1997 34427256,1997

Minimum(10) nodes in a way: sqlite> select id, count() *from ways_nodes . . . > group by id . . . > order by count*() . . . > asc limit 10;

6253195,1 6254292,1 6255836,1 6256703,1 6258172,1 6286544,1 6289220,1 6290019,1 7950156,1 7952046,1

Average number of nodes in a way: sqlite> select avg(sum) from (select id,count(*) as sum from ways_nodes group by id);

15.3086734982348

1.0.8 Conclusion

Case study of OSM of atlanta_georgia gave me an experience of how the data wrangling process is done on original dataset. I believe more cleaning can be done on dataset and given as an input to OpenStreetMap.org