P3: Wrangle OpenStreetMap Data

Map Area

Coquitlam, BC, Canada

https://www.openstreetmap.org/relation/2221139#map=12/49.2857/-122.7568

This map is of Coquitlam, British Columbia in Canada. It is my current place of residence and I am interested to get to know the place more as I have only moved here 3 months ago.

Problems Encountered

The below are the few problems encountered while working with the data.

- Abbreviated street names such as '3420 Bell Ave' and 'Bonsor St.'.
- Inconsistencies in the way postal codes were represented.
- Inconsistencies in which province we represented.
- Two 'k' values with problematic characters were found.

Solutions to the Problems Encountered

- Abbreviated Street Names:

The two abbreviations that were found to be in need of correcting were 'Ave' and 'St.' as can be seen from the set below.

Faulty_names:
'Ave': {'3420 Bell Ave', 'MacPherson Ave', 'Royal Oak Ave'}
'St.': {'Bonsor St.'}

These abbreviated street names were corrected using the below update_name function that was called within the shape_element function. This ensures that the abbreviations are corrected before generating the csv files.

```
def update_name(name, mapping): #used for fixing abbreviated street names
    word = name.split()[-1]
    for i in mapping:
        if word == i:
            new_name = name.rsplit(' ',1)[0] +' '+ mapping[i]
    return new_name
```

- Postal Codes

Some inconsistencies were found with the way postal codes were presented in the data. Canadian postal codes are typically consisted of 6 characters but with a space after the first 3 (e.g. V3B 3J5). However, some postal codes were entered without the space in between (e.g. V3B3J5). For consistency purposes, those that had 6 characters but did not contain the space in between has the space added with the below function.

```
def update_postcode(postcode): #used for fixing postal codes
   if len(postcode) == 6:
      new_postcode = postcode[:3] + ' ' + postcode[3:]
   else:
      new_postcode = postcode
   return new_postcode
```

As with the abbreviated street names, update_postcode was called in the shape_element function.

- Province

Province name for all data in the osm file should be British Columbia or commonly abbreviated as BC. For consistency purposes, all province value is being changed to BC within the shape_element function.

- Problematic Characters

Two occurrences of problematic characters were found in the 'k' values of 'tag' tags. These two elements with problem characters will be disregarded. Their values are:

```
<tag k="hov.minimum" v="2" />
<tag k="Mullti Flooring" v="showroom" />
```

Data Overview

File Sizes

coquitlam.osm	88 MB
coquitlam.db	63 MB
nodes.csv	33 MB
nodes_tags.csv	2 MB
ways.csv	3 MB
ways_tags.csv	5 MB
ways_nodes.csv	10 MB

Number of nodes

```
sqlite> SELECT COUNT(*) FROM nodes;
```

401270

Number of nodes_tags

```
sqlite> SELECT COUNT(*) FROM nodes_tags;
```

60797

Number of ways

```
sqlite> SELECT COUNT(*) FROM ways;
```

49396

Number of ways_nodes

```
sqlite> SELECT COUNT(*) FROM ways_nodes;
```

465267

Number of ways_tags

```
sqlite> SELECT COUNT(*) FROM ways_tags;
```

138020

Number of unique users

```
sqlite> SELECT COUNT(*) FROM (SELECT uid FROM nodes UNION SELECT uid FROM ways);
```

457

Top 10 users

```
sqlite> SELECT user, COUNT(*) as num
   ...> FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways)
   ...> GROUP BY user
   ...> ORDER BY num DESC
```

```
   ...> LIMIT 10;
```

```
Marcott|91965
rbrtwhite|78055
z-dude|53526
mbiker_imports_and_more|42096
pnorman|38663
mattropolis|24203
mbiker|13072
geoffengland|12185
pnorman_mechanical|9550
MetVanRider123acme|8774
```

With the total contributions by users as:

```
sqlite> SELECT COUNT(*) as num FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways);
```

```
450666
```

We can see that the top 3 users contributed nearly 50% of the data and the top 10 users contributing over 80% of the data.

Number of users with only 1 entry

```
sqlite> SELECT COUNT(*) FROM (SELECT user, COUNT(*) as num
   ...> FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways)
   ...> GROUP BY user
   ...> HAVING num=1);
```

```
83
```

Top 10 amenities

```
sqlite> SELECT value, COUNT(*) as num FROM nodes_tags
   ...> WHERE key='amenity'
   ...> GROUP BY value
   ...> ORDER BY num DESC
   ...> LIMIT 10;
```

```
restaurant|201
bench|116
fast_food|105
waste_basket|93
parking|90
cafe|75
bank|57
toilets|57
```

post_box|41
fuel|40

Top 10 cuisines

```
sqlite> SELECT nodes_tags.value, COUNT(*) as num FROM nodes_tags JOIN
   ...> (SELECT DISTINCT(id) FROM nodes_tags WHERE value='restaurant') i ON nodes_tags.id=i.id
   ...> WHERE nodes_tags.key='cuisine'
   ...> GROUP BY nodes_tags.value
   ...> ORDER BY num DESC
   ...> LIMIT 10;
```

chinese|20
japanese|20
sushi|13
pizza|8
korean|7
sandwich|7
indian|6
vietnamese|6
greek|5
american|4

Since the top most frequent amenity was that of the restaurant, I was curious as to the cuisines available. The results were not surprising since there has been huge influx of immigrants to BC which brought about great diversity to the region.

Top 10 fast food

```
sqlite> SELECT nodes_tags.value, COUNT(*) as num FROM nodes_tags JOIN
   ...> (SELECT id FROM nodes_tags WHERE value='fast_food') i ON nodes_tags.id=i.id
   ...> WHERE nodes_tags.key='name'
   ...> GROUP BY nodes_tags.value
   ...> ORDER BY num DESC
   ...> LIMIT 10;
```

Subway|16
A&W|5
KFC|5
McDonald's|5
Tim Hortons|5
Quiznos|4
Panago|3
Church's Chicken|2
Dairy Queen|2
Fatburger|2

Top 10 café

```
sqlite> SELECT nodes_tags.value, COUNT(*) as num FROM nodes_tags JOIN
   ...> (SELECT id FROM nodes_tags WHERE value='cafe') i ON nodes_tags.id=i.id
   ...> WHERE nodes_tags.key='name'
   ...> GROUP BY nodes_tags.value
   ...> ORDER BY num DESC
   ...> LIMIT 10;
```

```
Starbucks|19
Starbucks Coffee|7
Tim Hortons|4
Renaissance|2
Tim Horton's|2
Amelia Cafe|1
Anny's|1
BG Urban Cafe|1
Blenz|1
Blenz Coffee|1
```

Having moved from the prairies province of Saskatchewan, I was used to seeing Tim Hortons café around most corners. However, I was surprised that Tim Hortons are not that common in Coquitlam. The results confirmed my suspicion that Starbucks are more popular in Coquitlam.

Top 10 banks

```
sqlite> SELECT nodes_tags.value, COUNT(*) as num FROM nodes_tags JOIN
   ...> (SELECT id FROM nodes_tags WHERE value='bank') i ON nodes_tags.id=i.id
   ...> WHERE nodes_tags.key='name'
   ...> GROUP BY nodes_tags.value
   ...> ORDER BY num DESC
   ...> LIMIT 10;
```

```
TD Canada Trust|14
Scotiabank|9
CIBC|4
RBC|4
Vancity|4
HSBC|3
Bank of Montreal|2
TD Bank|2
BMO|1
BMO Bank of Montreal|1
```

Additional Ideas

As can be seen from the results of top 10 café and banks, abbreviations used by different users can caused the same entities to be separated. This can result in inaccurate analysis of the data. One example would be that of Starbucks' market share in the area. If we only took the top most result of 19 Starbucks, we would say that 25% out of all café in Coquitlam are Starbucks. However, the real figure should be 26 (19+7) since "Starbucks Coffee" is the same as Starbucks. This would mean that out of the 75 café in the area, 35% are Starbucks. Therefore, these abbreviations have the potential to cause a 10% difference in the analysis. Further cleaning of the data needs to be carried out in order to ensure accurate analysis is presented.

In order to carry out cleaning of abbreviated names, an audit should be carried out for all nodes and keys tags where k values = "name". We can then see a list of abbreviations and decide on which name to clean. For example, "BMO", "Bank of Montreal" and "BMO Bank of Montreal" can all be standardized to "Bank of Montreal".

Another foreseeable problem would be the categorization of certain entities in different categories. For example, Tim Hortons appears in both fast food as well as café. This presents us a question of whether the 5 Tim Hortons in fast food are part of the six that are showing up in café. This part of the cleaning would be more difficult as it affects the accuracy of the data.

Conclusion

It seems that further cleaning of the data is needed in order to get better and more accurate results for our analysis.