Cleaning and Importing OpenStreetMap Data

July 14, 2018

1 Map Area

Data for Southern Singapore was downloaded using the Overpass API with the following query.

```
(
   node(1.2768,103.7866,1.3785,103.8687);
   <;
);
out meta;
```

2 Problems Encountered in the Map

After sampling the original openstreetmap data file with `sampler.py`, I noted that the `addr:street` tags contained values that had to be interpreted in context of local naming conventions. It is common to have Singapore addresses end with a number e.g. Ang Mo Kio Avenue 4, so I used regular expression to extract the last word that does not contain a number.

Auditing the full dataset using `audit.py` revealed the following obvious problems with `addr:city`, `addr:postcode` and `addr:street` tags.

- Abbreviated street names (*"Lornie Rd"*)
- All lowercase letters for street names (*"jalan kubor"*)
- All lowercase letters for city name (*"singapore"*)
- City names with invalid characters (*"Singaporezs"*)
- Postcodes with only 5 digits (*"18984"*)
- Postcodes with a leading 'S' (*"S118556"*)
- Postcodes with whitespaces (*"437 437"*)

The following problems with street names, city names and postcodes were cleaned in `db_make.py` before the `.csv` files are made.

2.1 Street Names

I corrected street names before creating the `.csv` files, using `.strip()` to remove leading and end spaces, and `.capwords` to capitalize the first letter of each word.

I also used the following mapping dictionary to replace abbreviated street names.

```
mapping_street = { "St": "Street",
                   "Ave" : "Avenue",
                   "Avebue" : "Avenue",
                   "Dr" : "Drive",
                   "Rd" : "Road",
                 }
```

2.2 City Names

I removed leading and ending spaces with `.strip()`, and invalid white spaces with `.replace()`. City names were replaced with "Singapore" as long as it contains a substring `singapore`.

2.3 Postal Codes

I removed leading and ending spaces with `.strip()`, and invalid white spaces with `.replace()`. I also corrected the postal codes with leading 'S' characters.

Singapore's postcodes are 6 digit long but my dataset contained many invalid 5 digit postcodes. After checking on streetdirectory.com, it's apparent that the error is due to the removal of the leading '0' in these 5 digit postcodes. The below table shows an excerpt of the postcode mapping from OpenStreetMap and Street Directory.

Street Name	OpenStreetMap Postcode	Street Directory Postcode
Amoy Street	49965	049665
Bayfront Avenue	18957	018957
Cecil Street	69544	069544
Chulia Street	49513	049513
Eu Tong Sen Street	59815	059815
Kadayanallur Street	69184	069184
Marina Boulevard	18983	018983
Marina Gardens Drive	18953	018983
Marina View	18960	018960
Market Street	48942	048942
New Bridge Road	59443	059443
New Market Road	50032	050032
Smith Street	50336	050336
Park Road	59108	059108
Pickering Street	48659	048659
Raffles Avenue	39802	039802

3 Import Data Into SQL Database

I created a sql script `data_wrangling_schema.py` that created the `nodes`, `ndoes_tags`, `ways`, `ways_tags` and `ways_nodes` table with the suggested schema. I then used sqlite3 command line to execute the following commands.

```
> sqlite3 osm.db
```

```
sqlite3> .read data_wrangling_schema.sql
sqlite3> .mode csv
sqlite3> .import nodes.csv nodes
sqlite3> .import ways.csv ways
sqlite3> .import nodes_tags.csv nodes_tags
sqlite3> .import ways_tags.csv ways_tags
sqlite3> .import ways_nodes.csv ways_nodes
```

4 Data Overview

This section contains basic statistics about the dataset and the SQL queries used to gather them.

```
rawdata.osm ......... 77.8 MB
osm.db .............. 42.0 MB
nodes.csv ........... 25.8 MB
nodes_tags.csv ...... 1.77 MB
ways.csv ............ 3.26 MB
ways_tags.csv ....... 5.78 MB
ways_nodes.cv ........10.0 MB
```

4.1 Number of Nodes and Tags

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

```
321151
```

```
SELECT count(*) FROM nodes_tags; `
```

```
47136
```

4.2 Number of Ways and Tags

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

```
55718
```

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

```
172793
```

4.3 Number of Ways Nodes

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

```
420662
```

4.4 Number of Unique Users

```
SELECT COUNT(DISTINCT(jnodeway.uid)) FROM
(SELECT uid FROM nodes UNION ALL SELECT uid FROM ways) jnodeway;
```

```
1045
```

4.5 Top 10 Users

```
SELECT jnodeway.user, COUNT(*) AS num FROM
(SELECT user FROM nodes UNION ALL SELECT user FROM ways) jnodeway
GROUP BY jnodeway.user
ORDER BY num desc
LIMIT 10;
```

```
JaLooNz|89255
Luis36995|28745
Aditya Anggun|20219
happy-camper|17784
Evandering|14255
Mapintosh|10031
yurasi|9894
rene78|9371
Paul McCormack|9077
nikhilprabhakar|9004
```

4.6 Number of Users Appearing Only Once (having 1 post)

```
SELECT COUNT(*)
FROM
(SELECT jnodeway.user, count(*) AS NUM
FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways) jnodeway
GROUP BY jnodeway.user
HAVING num=1)
```

```
373
```

5 Additional Ideas

5.1 User Contribution

- User `JaLooNz` alone contributed 23.7% of the tags.
- The 2nd ranked user `Luis36995` contributed 7.6% of tags.
- 35.7% of users only contributed to 1 post

This is not surprising as we can see from user `JaLooNz`'s blog that he/she had set out to improve the OpenStreetMap coverage of Southern Singapore after observing the holes in its coverage. A way to encourage more users to contribute is through gamification (as suggested in the Sample Project) where users will gain points from adding to a map area. Top ranking contributors could be display on OpenStreetMap's website for better recognition.

5.2 Additional Data Cleaning

While auditing data for the `addr:city` tag, I observed values `563455`, `Ang Mo Kio` and `Holland Village`, which are invalid city names. `563455` is a postcode, and `Ang Mo Kio` and `Holland Village` are street addresses. It is most likely that the addresses were misentered/ shifted across

the other addr tags. Cleaning up these data entries will require looking at all the addr tags associated with a node or way and making sure that they are entered in the correct heading, which will also require some contextual understanding of the addresses in Singapore. I did not clean up these nodes and ways but could do so in the future by coding a section that look at addr values in its entirety.

5.3 Additional Data Exploration

5.3.1 Top 10 Amenities

Singapore is known to be a food and shopping haven. Southern Singapore is a popular tourist destination with links to other islands of Singapore and Indonesia. It is not a surprise then to find that restaurant is the top amenity in this area.

```
SELECT jnodeway_tags.value, COUNT(*) AS num FROM
(SELECT value, key FROM nodes_tags UNION ALL SELECT value, key FROM ways_tags) jnodeway_tags
WHERE jnodeway_tags.key ='amenity'
GROUP BY jnodeway_tags.value
ORDER BY num DESC
LIMIT 10;
```

```
restaurant|1252
parking|626
cafe|308
atm|276
taxi|200
place_of_worship|173
parking_entrance|170
swimming_pool|165
bar|142
school|138
```

5.3.2 Number of Religions

Singapore is a multi-racial, multi-lingual and multi-religious society. Here I investigated the top places of worship, grouped by religion in Southern Singapore. The top 4 values match the top 4 religions found in Singapore, i.e. Christianity, Buddhism, Islam, and Hinduism. It would be interesting to compare the proportion found here in Southern Singapore to that of the rest in Singapore.

```
SELECT all_tags.value, count(*) as num FROM
(
SELECT nodes_tags.value, nodes_tags.key
FROM nodes_tags JOIN (SELECT DISTINCT(id) FROM nodes_tags WHERE value='place_of_worship') nt
ON nodes_tags.id=nt.id
UNION ALL

SELECT ways_tags.value, ways_tags.key
FROM ways_tags JOIN (SELECT DISTINCT(id) FROM ways_tags WHERE value='place_of_worship') wt
```

```
ON ways_tags.id=wt.id
) all_tags

WHERE all_tags.key = 'religion'
GROUP BY all_tags.value
ORDER BY num DESC
LIMIT 10;
```

```
christian|67
buddhist|35
muslim|24
hindu|9
taoist|8
jewish|2
sikh|2
```

5.3.3 Most popular cuisines

As mentioned, Singapore is famous for its shopping and food options. I wanted to explore the cuisines available in Southern Singapore and found that Chinese and Japanese food are the most popular. This is not surprising as the Chinese make up the majority of the population, while Japanese cuisine has been popular in the country for a long time. There is an increasing popularity of Korean food, so I am not surprised to see that on the list. I am surprised at seeing French food on the list as it is a relatively more expensive and exclusive option. That said, there has been a rise in accessible French restaurants such as Poulet in Singapore, which might explain for it being on the top 10 list.

```
SELECT all_tags.value, count(*) as num FROM
(
SELECT nodes_tags.value, nodes_tags.key
FROM nodes_tags JOIN (SELECT DISTINCT(id) FROM nodes_tags WHERE value='restaurant') nt
ON nodes_tags.id=nt.id
UNION ALL

SELECT ways_tags.value, ways_tags.key
FROM ways_tags JOIN (SELECT DISTINCT(id) FROM ways_tags WHERE value='restaurant') wt
ON ways_tags.id=wt.id
) all_tags
```

```
chinese|130
japanese|63
indian|43
korean|42
italian|34
thai|25
asian|23
pizza|23
seafood|15
french|12
```

6 Conclusion

Most of the trends observed are as expected. However, there is a need to increase user input in populating information for Southern Singapore. I also noted that the data is well-organized with some minor instances of incorrect values, as observed for city names. This could be corrected with a more holistic cleaning approach.

A side note that validating the elements in this 77.8MB `rawdata.osm` file took about 8 hours on my machine. I had initially wanted to use data for the entire mainland island of Singapore but decided not to follow through with this approach due to the long processing time.

7 Resources referred to

- Sample Project