Project: Wrangle OpenStreetMap Data

Map Area

Pune, Maharashtra, India

https://mapzen.com/data/metro-extracts/metro/pune_india/

This map belong to one of the fastest growing city Pune in the state of Maharashtra, India. My hometown is 250 km away from Pune. The common language that is spoken in state of Maharashtra is Marathi, so I expect most of street names, local addresses are related to language Marathi. Marathi is my mother tongue. So, understanding the map area will not be much difficult. So, I take this opportunity to contribute to project of OpenStreetMap.org

Problems Encountered in the Map

I downloaded the metro extract of the Pune city. The size of the file is 293 MB. I created the sample OSM file of 9.3 MB from given code, to audit and clean the data. I created some audit scripts to check the problems associated with the data. Some of the problems associated with the data are listed below.

- Inconsistent postal codes. Indian postal codes are of six digit. Most of the postal codes do not have space between digits, some postal codes have space after the three digits.

  ```
  <tag k="addr:postcode" v="411 021" />
  ```

- Regional names for street types and ways. These are Marathi names for roads but written in English. *("Path", "Marg")*
- Non standard phone numbers (Phone numbers starting with zero instead of country code) *(“02041206727”)*

In this project street types, way names and phone numbers have been cleaned up, the data base is created in the sqlite from csv files and queried using sqlite3 engine.

Regional Names for Street Types and Ways

At the beginning I used code developed in exercise stage. When I ran the audit street types code. I found some of the problems associated with street type. The names in Marathi are translated as is. The example is given below.

```
<tag k = "name" v = "Bhagwan Tatyasaheb Kawade Marg" />
<tag k= "name:mr" v = "भगवान तात्यासाहेब कवडे मार्ग" />
```

In Marathi "Marg" means "Road" but we can see that, it is as is written in English. The below written function cleans names for streets in the nodes and ways tags.

```
def update_name(name, mapping):
    ''' update street names '''
    if name.split()[-1] in mapping.keys():
        ChangedWord = mapping[name.split()[-1]]
        name = name.split()[:-1]
        name.append(ChangedWord)
        name = " ".join(name)
    return name
```

Phone Numbers

Some phone numbers start with '0'. Ideally the phone number should start with plus sign and country code. The following is the extract of the document on wiki page of openstreetmap which specifies the structure of the phone number.

```
def update_phone_number(phone_number):
    ''' update phone numbers '''
    if phone_number[0] == '0':
        new_number = "+91" + phone_number[1:]
        return new_number
    else:
        return phone_number
```

	name
0	nodes
1	nodes_tags
2	ways
3	ways_tags

	name
4	ways_nodes

Size of the Files

audit_clean_phone_numbers.py --> 1.58 kB
audit_clean_street_types.py --> 1.92 kB
audit_clean_way_names.py --> 1.89 kB
create_db.sql --> 1.87 kB
data.py --> 8.83 kB
database_query.ipynb --> 22.57 kB
database_query.md --> 12.18 kB
nodes.csv --> 113.35 MB
nodes_tags.csv --> 512.25 kB
openstreet.db --> 228.12 MB
pune_india.osm --> 293.96 MB
sample.osm --> 9.3 MB
schema.py --> 2.56 kB
ways.csv --> 15.73 MB
ways_nodes.csv --> 39.13 MB
ways_tags.csv --> 9.06 MB

Number of Nodes

```
# number of nodes

pd.read_sql_query("SELECT COUNT(*) FROM nodes;", conn)
```

	COUNT(*)
0	1418342

Number of Ways

```
pd.read_sql_query("SELECT COUNT(*) FROM ways;" ,conn)
```

	COUNT(*)
0	270302

Number of unique users

```
pd.read_sql_query("SELECT COUNT(DISTINCT(e.uid))FROM (SELECT uid FROM nodes UNION ALL S
```

	COUNT(DISTINCT(e.uid))
0	690

Additional Ideas

1. **Phone Numbers with space or dash between country code, area code and local number** As mentioned earlier the phone number should have following format.

   ```
    "phone" = "number" where the number should be in international (ITU-T E.164) format
   phone = +<country code> <area code> <local number>, following the ITU-T E.123 and
   the DIN 5008 pattern phone = +<country code>-<area code>-<local number>, following
   the RFC 3966/NANP pattern)
   ```

   I have addressed only the country code issue i.e. phone numbers begining with zero have been replaced by plus sign and country code. In fact the country code, area code and local numbers should be seperated by either "dash" or "space". This will completely address the phone numbers issue.

2. **Differentiation between mobile numbers and landline numbers** In India mobile numbers do not have area codes. So to represent mobile numbers there has to be some different strategy. Mobile numbers have 10 digits, it might be represented like this +91-123-456-7890 or +91 123 456 7890

3. **Inconsistency in postal codes** Indian postal codes have six digits. There is particular format provided in documentation on how to represent postal codes. But Indian case

since it has six digits it makes sense to have either space or dash after three digitis. It will increase the readability.

Additional Data Exploration

Top 10 appearing amenities

```
pd.read_sql_query("SELECT value, COUNT(*) as num \
FROM nodes_tags \
WHERE key='amenity' \
GROUP BY value \
ORDER BY num DESC \
LIMIT 10", conn)
```

	value	num
0	restaurant	240
1	bank	175
2	atm	139
3	place_of_worship	120
4	cafe	75
5	fast_food	70
6	hospital	53
7	fuel	45
8	school	40
9	police	31

Biggest Religion

```
pd.read_sql_query("SELECT nodes_tags.value, COUNT(*) as num \
FROM nodes_tags \
    JOIN (SELECT DISTINCT(id) FROM nodes_tags WHERE value = 'place_of_worship') i \
    ON nodes_tags.id = i.id \
WHERE nodes_tags.key = 'religion' \
GROUP BY nodes_tags.value \
ORDER BY num DESC \
LIMIT 3; ", conn)
```

	value	num
0	hindu	76
1	muslim	10
2	christian	4

No surprise here. Almost 80% of Indian population has religion of Hindu. We can see that Hindu and Muslim are two bigger religions in city of Pune, even this statistic is applicable at national level as well.

```
pd.read_sql_query("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 5;", conn)
```

	value	num
0	indian	46
1	vegetarian	13
2	pizza	10
3	regional	8
4	international	5

Conclusion

The street types and phone numbers have cleaned as part of this project requirement. As mentioned in additional ideas, lot of cleaning is still left. To bring improvements faster to openstreet map project, I think people have to start using it. The local businesses, communities should be encouraged to add their details to the map data.

Benefits of Suggestions

1. If local businesses start entering their the information the map data will be more accurate and updated.

2. It will be easy for others find out necessary about information about locating businesses or addresses.

Anticipated Problems in Implementations

1. Due to availability and popularity of Google maps. People will be hesitant to use other maps. Bringing people to OpenStreetMap will itself be big challenge.

2. Making local [e.g. Language] specific user interfaces so that adding information about local businesses and addresses will be easy.

References

1. https://mapzen.com/data/metro-extracts/metro/pune_india/
2. http://wiki.openstreetmap.org/wiki/Key:phone
3. http://wiki.openstreetmap.org/wiki/Names
4. https://gist.github.com/carlward/54ec1c91b62a5f911c42#file-sample_project-md
5. https://www.dataquest.io/blog/python-pandas-databases/
6. https://discussions.udacity.com/t/project-suggestions-for-improvements/196037