OpenStreetMap Data Case Study

Map Area

Greenwich, Stamford, Norwalk, Bridgeport, CT, United States

- https://www.openstreetmap.org/relation/3306055#map=11/41.0794/-73.455
- https://mapzen.com/data/metro-extracts/your-extracts/b76db9fc4894

I live in Stamford, so I am interested to see what I could find from the database. Stamford is not a big city, so three more cities that near to stamford are included in the database. However, other smaller city in this region cannot be removed when I choose the area, so these small towns also in the data.

The Problems Encountered in the Map

After downloading the data from Map Zen and running the data using data.py, I got five .csv files, and I found the most common problems in the files.

- Street Names and Postal Codes are not standardized

For example, St. is used for Street, Ave. is used for Avenue, post codes has many formats including five digits like 06901, or followed by four more digits like 06901-1602, or start with the state like CT 06901, so I add two functions, one named 'update_name' to standardized these abbreviated names, one named 'update_postcode' to standardized postal codes. And then I ran the data.py again to get new five .csv files.

The code of the function 'update_name':

```
def update_name(name, mapping):
    if name in mapping:
        name = mapping[name]
    else:
        return name
```

The code of the function 'update_postcode':

```
def update_postcode(code):
    postcode = re.findall(r'(\d{5})', code)[0]
    return postcode
```

Import .csv Files to Database

import .csv files using the following code template in the command:

```
sqlite3 new.db
sqlite> CREATE TABLE myTable()
sqlite> .mode csv
sqlite> .import newFile.csv myTable
```

And then type:

```
sqlite> .tables
```

get the result:

nodes	nodes_tags	ways	ways_nodes	ways_tags

It mean all five tables are created successfully.

Data Overview and Some Statistics

In this section I will give the files' size and use sql queries to get some statistics.

File Sizes

```
GSNB.osm ......... 72.3 MB
GSNB.db .......... 39.8 MB
nodes.csv ............. 26.9 MB
nodes_tags.csv ........ 988 KB
ways.csv .............. 2.3 MB
ways_tags.csv ......... 8.8 MB
ways_nodes.cv ......... 6.3 MB
```

Number of unique users

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

```
445
```

Number of nodes

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

```
320886
```

Number of ways

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

```
37590
```

TOP 3 Cafe

```
SELECT nodes_tags.value, COUNT(*) as num
FROM nodes_tags JOIN (SELECT DISTINCT(id) FROM nodes_tags WHERE value='cafe') AS a
ON nodes_tags.id=a.id
```

```
GROUP BY nodes_tags.value
ORDER BY num DESC
LIMIT 3;
```

name	num
cafe	35
Dunkin' Donuts	14
donut	8

Number of School

```
SELECT COUNT(*) as num
FROM nodes_tags
WHERE value='school'
GROUP BY value;
```

```
180
```

Some Data Mistake and Improvement Advice

First, sort the TOP 10 cities,

```
SELECT tags.value, COUNT(*) as count
FROM (SELECT * FROM nodes_tags UNION ALL
      SELECT * FROM ways_tags) tags
WHERE tags.key = 'city'
GROUP BY tags.value
ORDER BY count DESC
LIMIT 10;
```

and the edited result:

City	count
Fairfield	246
Bridgeport	232
Norwalk	85
Stamford	79
Westport	37
Greenwich	35
Darien	28

"Cos Cob"	23
"Old Field"	18
Wilton	10

Then, sort the TOP 10 postal codes,

```
SELECT tags.value, COUNT(*) as count
FROM (SELECT * FROM nodes_tags
      UNION ALL
      SELECT * FROM ways_tags) tags
WHERE tags.key='postcode'
GROUP BY tags.value
ORDER BY count DESC
LIMIT 10;
```

the edited result:

Postal Code	count
06824	65
06902	47
06880	36
06825	33
06855	32
06807	26
06820	24
06851	24
06830	23
11733	18

From these two result, I found some problems. First, the city ' Fairfield' appears most times(246), however, as I know, ' Fairfield' is not a city but a county, all cities I chose belongs to this Fairfield. Clearly, the OpenStreetMap tags 'Fairfield' incorrectly. And then the tenth frequent postal code is 11733, but the postal codes of the region I chose start with '068' or '069'. Again, data has the mistake.

OpenStreetMap has the huge contribution to open data as a collaborative project to create a free editable map of the world, it still can do some improvement, for example, when the volunteers upload their data, the website should verify the authenticity and accuracy of the data, what's more, maybe the website can set a system to give the users some rewards and create the leaderboard, I think that can motivate the users to provide the real data. However, some problems maybe followed if this advice is implemented, for example, it will cost more. As a free website, I think OpenStreetMap has the limited budget, rewards or leaderboard may not have enough financial support.