OpenStreetMap Project Data Wrangling with MongoDB

Chris Eldredge

Map Area: San Francisco, CA, United States
https://www.openstreetmap.org/relation/111968
https://mapzen.com/data/metro-extracts/#san-francisco-california

1. Problems Encountered

I observed two inconsistencies in the OpenStreetMap data for the San Francisco area:

- Inconsistent abbreviations in address street names (e.g. St. vs. St vs. Street). I applied the update_name function developed in Lesson 6 to standardize the street names.
- Some zipcodes (postcodes) were not the correct length (e.g. “9412”). United States zipcodes must contain a minimum of 5 digits, however I noticed some invalid postcodes in the San Francisco OSM file with fewer than 5 digits. Since there was not a clear way of determining the correct zipcode in these cases, I chose to exclude invalid zipcodes before loading the data into MongoDB.

2. Data Overview

Basic statistics about the dataset.

File sizes

sample-san-francisco_california.osm ----> 92 MB
sample-san-francisco_california.osm.json ----> 104 MB

# Count documents

```
> db.sfosm.find().count()
497122
```

# Count nodes

```
> db.sfosm.find({"type":"node"}).count()
446690
```

# Count ways

```
        > db.sfosm.find({"type":"way"}).count()
        50420

# Count unique users
        print len(db.sfosm.distinct("created.user"))
        1278

# Top 10 amenities for San Francisco area
        > db.sfosm.aggregate([{"$match":{"amenity":{"$exists":1}}}, {"$group":{"_id":"$amenity",
        "count":{"$sum":1}}}, {"$sort":{"count":-1}},{"$limit":10}])

        {u'_id': u'parking', u'count': 391}
        {u'_id': u'restaurant', u'count': 267}
        {u'_id': u'school', u'count': 122}
        {u'_id': u'place_of_worship', u'count': 108}
        {u'_id': u'bench', u'count': 100}
        {u'_id': u'cafe', u'count': 75}
        {u'_id': u'post_box', u'count': 62}
        {u'_id': u'fast_food', u'count': 56}
        {u'_id': u'drinking_water', u'count': 49}
        {u'_id': u'bicycle_parking', u'count': 49}
```

3. Additional Ideas

Automated validation

From reviewing the OpenStreetMap data for the San Francisco area it occurred to me that because location data tend to follow strict conventions (e.g. postcodes of at least a length of 5 digits in the United States), would it be possible to alert the user when attempting to input invalid data? E.g. “That entry only contained 4 digits, but postcodes in the United States must contain at least 5 digits. Please enter a valid input.” While this kind of automated validation would be helpful in ensuring accurate data, it could be challenging to build and maintain such a system across multiple geographies, each with varying address formats and customs. Even within the same area, there can be multiple correct ways of writing an address. That said, if designed correctly and maintained, this kind of system could help improve the accuracy of the OpenStreetMap project.

Community and business outreach

Reviewing the top amenities in the San Francisco area, I was slightly surprised to see “parking” as the most common amenity. Given the difficulty of finding parking in a large city, I could see this type of facility being common. However, perhaps further analysis could be completed to see how these counts of facilities align with other official data sources. Are certain categories of buildings under or overrepresented in the OpenStreetMap data for San Francisco? If so, perhaps targeted outreach could be done to community and business organizations to have underrepresented groups input and update listings in the OpenStreetMap project. As part of the community outreach, it would make sense to clearly communicate the benefit to local organizations of clearer listings on OpenStreetMap. Accurate listing in OSM, and applications that use location data from OSM, would help new clients and customers find these organizations. However, a possible problem with implementing this kind of program would be finding someone to complete this kind of analysis for free. Perhaps a creative solution could be found, such as working with a university student interested in gaining experience and partnering with a local journalist to publicize the findings.

Conclusion

For the most part I found the OpenStreetMap data for San Francisco to be complete and well structured. There were a few areas of inconsistent and invalid data that I chose to clean before transforming the XML into JSON and loading it into MongoDB. Additionally, as I worked through the project, it occurred to me that it may make sense for the OpenStreetMap project to consider adding automated validation rules and performing additional analysis to identify opportunities for community outreach to improve the quality of the location data. However, there would be some challenges with maintenance and finding resources that would need to be worked out.