San Francisco OpenStreetMap Data Case Study

Map Area

San Francisco, CA

- OpenStreetMap link (https://www.openstreetmap.org/relation/111968)
- Direct link to my file (https://drive.google.com/open?id=0B2BGHnr9cnONSEJYd3FTSEQ2TU0)

This is a map of my foggy home city. It also includes bits of the surrounding areas such as Berkeley and Daly City. I'm mostly interested in the shop and amenities queries... like how many ice cream shops are in San Francisco.

Problems Encountered in Map Data

After a very long auditing process on the large XML document, here are the major problems I noticed:

- Inconsistent street name abbreviations (ST, St., St for Street)
- Incomplete entries ("Telegraph" for "Telegraph Avenue")
- Inconsistent zip codes (CA 94116)
- Inconsistent shop, amenity, and cusiine values (typos, singulars/plurals)
- Many shop and cuisine values are referring to the same thing (Hair Salon, hair salon, hair_salon, hairdresser)
- Values in every audited category had inconsistent capitalizaions

Problems in Street Names

To audit street names, I first compiled a list of expected street names. Then I checked the last word of every street name to see if it was in the list. If there were many entries that aren't in the list, then I must have missed it. Otherwise, I made a note on problems to clean later.

Variations of "Alameda de las Pulgas" were found through a regex search.

```
alameda_regex = re.compile(r”(alameda\sde\sla\s)(pulgas)?”, re.I)
```

To clean the street names, I either replaced the entire street name or replaced parts of the string with the correct abbreviations.

Problems in Amenity, Shop, and Cuisine Values

The main problem with these values is many of them are two values that refer to the same thing. Some are named vaguely or different such as "dimsum" and "dim_sum" or variations of singular and plural versions of the word. I also made notes of typos and removed subcategories to further simplify categorizing this data.

I made dictionaries to replace all the "bad" values to have a more consistent query for the same values. Some bad values that are mislabeled are removed.

Data Overview

File Sizes

```
4.0K    README.txt
8.0K    audit_files.py
 20K    clean_and_prep_files.py
4.0K    csv_to_db.py
4.0K    data_wrangling_schema.sql
502M    map_data.db
387M    nodes.csv
317M    nodes.db
6.9M    nodes_tags.csv
4.0K    presentation_notes.txt
4.0K    project03.md
1001M    san_francisco.osm
4.0K    schema.pyc
336K    sf_sunset_district.osm
 34M    ways.csv
133M    ways_nodes.csv
118M    ways_nodes.db
 40M    ways_tags.csv
```

Total Nodes

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

4840499

Total Ways

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

580988

Number of Unique Contributors

```
sqlite> SELECT COUNT(DISTINCT(e.uid))
   ...> FROM (SELECT uid FROM nodes UNION ALL SELECT uid FROM ways) e;
```

2293

Total Amenities

```
 sqlite> SELECT key, COUNT(*)
   ...> from nodes_tags
   ...> where key="amenity";
```

amenity|248

Total Shops

```
sqlite> SELECT key, COUNT(*)
   ...> from nodes_tags
   ...> where key="shop";
```

shop|4091

Total Cuisines

```
sqlite> SELECT key, COUNT(*)
   ...> from nodes_tags
   ...> where key="cuisine";
```

cuisine|2128

Most Popular Cuisines

```
sqlite> SELECT value, COUNT(value) as total
   ...> FROM nodes_tags
   ...> WHERE key="cuisine"
   ...> GROUP BY value
   ...> ORDER BY total desc
   ...> LIMIT 8;
```

```
coffee|224
mexican|197
pizza|157
chinese|147
japanese|126
burgers|115
italian|115
sandwiches|106
```

Additional Ideas

Percentage of one way streets

```
sqlite> SELECT value, COUNT(*)
...> FROM ways_tags
...> WHERE key="oneway" AND value="yes";
```

yes|17432

One of the reasons it seems difficult to drive in San Francisco is the number of one way streets. So how many do we actually have? Only makes 3%! It definitely feels like more when we commute in certain areas of the city.

How many Starbucks can there be?

```
sqlite> SELECT value, COUNT(*)
...> FROM nodes_tags
...> WHERE value="Starbucks" OR value="Starbucks Coffee"
...> ;
```

Starbucks|115

This is much less than I expected. Again, it seems like there's a Starbucks on every street because that's how quickly we run out of coffee now.

How about ice cream shops?

```
sqlite> SELECT value, COUNT(*)
 ...> FROM nodes_tags
 ...> WHERE value="ice_cream";
```

ice_cream|72

One major problem with this query is it queries amenities, shops, and cuisine categories. The categorization is unclear, and some shops may be double or even triple counted which is too bad. I hoped we'd have more ice cream shops than Starbucks in my foggy home city.

Discussion

First, my process is very obviously incomplete. Although I think I did a thorough job cleaning things such as typos and similar categorizations, there are many points I had to either remove or simply not use due to lack of reliable references. I would like to further audit and clean relation tags in order to determine how difficult it may be to drive in San Francisco. Density of one way streets is one reason driving in San Francisco is difficult. No left turns, stuck in relation tags, are another reason.

Furthermore, it is clear from my querires, particularly about ice cream shops, that categorization of nodes needs to be more specific. I would also like to add a "district" key, so I can analyze things such as shop density in different areas of San Francisco.

While there are good arguments for both narrower and broader categorizations for amenities, shops, and cuisines, this ultimately requires local knowledge to handle. In addition, my script completely removed subcategories. I can improve on my analysis by figuring out how to handle subcategories when they matter.