OpenStreetMap Data Case Study

Map Area

Minneapolis, MN from https://www.openstreetmap.org/#map=5/51.500/-0.100

I chose Minneapolis because that's where I currently live and have enjoyed living here for 27 years outside of undergrad. However, I'm from Green Bay and will always be a Packer fan! There isn't the city export on OpenStreetMap anymore, so I had to use the approximate latitude and longitude boundaries.

Tags and Counts

Using the provided mapparser.py here are the different types of tags and their counts:

- osm: 1
- note: 1
- meta: 1
- bounds: 1
- node: 1190824
- tag: 709053
- way: 175548
- nd: 1427622
- relation: 987
- member: 31249

Tag Issues

Using the provided tags.py the osm file was checked for problem characters and lower case issues:

- lower: 389904
- lower_colon: 309486
- problemchars: 0
- other: 9663

Auditing the data – Cleaning the street names

Over abbreviation of the street names was quite the issue with the data, so I used the provided audit.py to update the abbreviations to the full words. I added North, South, East, West, Northeast, Northwest, Southeast, Southwest, Highway, and Circle to the expected list, and adjusted the mapping list to be more inclusive. To further clean the data a more extensive list of possible street names could be added, as well searching for more than the last word, and county roads or highways that ended in a number.

Here are a few examples of the cleaned street names:

- Arbour Ave => Arbour Avenue
- N 2nd St => N 2nd Street
- Harry Davis Ln => Harry Davis Lane
- Cleveland Avenue N => Cleveland Avenue North
- Marshall Street NE => Marshall Street Northeast

- Minnetonka Blvd => Minnetonka Boulevard

Here are a few examples that were not updated because the expected list wasn't more extensive or they end in numbers:

- 4th Street East, #120 => 4th Street East, #120
- MN-36 => MN-36
- East River Rd/Pkwy => East River Rd/Pkwy
- Washington Ave. N. => Washington Ave. N.

Then there are street names which turned out just fine but were listed because they were not in the excepted list:

- Charlton Ridge => Charlton Ridge
- Alpine Pass => Alpine Pass
- Tyrol Crest => Tyrol Crest
- Evergreen Knolls => Evergreen Knolls

Data Query and Overview

File sizes:

- MPLS.osm: 267.7MB
- nodes.csv: 96.3MB
- nodes_tags.csv: 7.6MB
- ways.csv: 10.0MB
- ways_nodes.csv: 34.3MB
- ways_tags.csv: 17.0MB
- MPLS.db: 140.3MB

I used a query_db.py created from looking at the SQL sample project and others to find some other pieces of information:

- Number of nodes: 1190824
- Number of ways: 175548
- Number of unique users: 1170
- Top contributing users: [('Mulad', 363169), ('stucki1', 191859), ('Omnific', 110829), ('iandees', 100431), ('woodpeck_fixbot', 55176), ('DavidF', 49305), ('houston_mapper1', 36825), ('rhardy', 35227), ('neuhausr', 34958), ('sota767', 28796)]
- Number of users contributing once: 228
- Common Amenities: ('restaurant', 504)
- Biggest religion: ('christian', 38)
- Popular cuisines: ('pizza', 31)

I'm not surprised by the number of restaurants and that it is the most common amenity. I often find it difficult to decide on a place to eat because of the variety. However, I was surprised by the fact that pizza is the most popular cuisine. Minneapolis has so many steakhouses with its proximity to farm country and being in the upper Midwest.

I was not surprised by Christianity being the most common religion. If the data was available and we broke it down even further I'm sure we would find the great majority to be Lutheran.

It's interesting that one user, Mulad, has 30% of the contributions to the OpenStreetMap export of Minneapolis. The top 10 contributors account for 84.5% of the nodes, which leaves 1160 making up the other 15.5%.

Conclusion

The human factor for errors or abbreviations is evident in the data set. While it is nice to go back and clean the data, it may be more useful to ensure proper spelling is input in the first place.

It would also be nice to have more information in terms of amenities and type of restaurants as the queries only returned one of each. Google and Facebook often ask users for updates depending on where they've been and what experiences they've encountered (yes, they are tracking us!); while OpenStreetMap isn't as popular it could certainly benefit from this type of input.

As mentioned above, having more extensive expected street name and mapping lists would improve the data further. This would require several iterations.

Files

audit.py – cleaned street names

data.py – file provided by Udacity for converting osm file to csv files

data_wrangling_schema.py – created database and tables using specified schema

mapparser.py – showed the tags in the XML and the count of each tag

MPLS.db – database created of Minneapolis, MN

MPLS.osm – export from OpenStreetMap

nodes.csv – csv of all the nodes in the MPLS.osm file

nodes_tags.csv – csv of tags of nodes

schema.py – schema provided by Udacity

tags.py – file provided by Udacity to find common errors in the MPLS.osm file

ways.csv – csv of all the ways in the MPLS.osm file

ways_nodes – csv of nodes of ways

ways_tags – csv of tags of ways

readme – this file