OpenStreetMaps Case Study – Cleveland Ohio

By Chris Burkhart

Map Area

Cleveland, Ohio, United States

https://www.openstreetmap.org/relation/182130#map=11/41.4980/-81.7060
https://mapzen.com/data/metro-extracts/metro/cleveland_ohio/

Growing up in north central Ohio, Cleveland is one of the largest cities around. I've never lived there, but have visited the area frequently and feel this will be a likely good candidate for potential job openings. As a large metropolitan area, I am interested to find out if the size of the city affects the reliability of the data.

Issues With the Data:

Tiger:name_type vs. Addr:Street – Two different tags representing what should be the same information but drastically different number of entries for the two.

addr:street way_tags = 3541
Tiger:name_type way_tags = 57078

Street Names – Inconsistent abbreviations for street names (Road, Rd, Rd., and rd)

Cities – Many cities included in the file, not just Cleveland, and several misspellings as well. Below is a sample from way_tags for 'addr:city'

'Clevalnd': 1,
'Cleveland': 353,
'Cleveland Heights': 353,
'Cleveland OH': 1,
'Clevland': 3,

Zip Codes – Most zip codes appear to be in the correct 5 digit format for US cities. Some however, contain the additional 4 digit identifier and several have OH or Ohio as the postcode.

'44203': 6,
'44203-4914': 1,
'44212': 2,
'44214': 2,

Fixing the Issues

Tiger data – Upon further investigation, the Tiger data was imported from a data set created by the US Census Bureau. This data was uploaded to OpenStreetMaps almost 10 years ago, and the data is not considered overly reliable for map-making purposes, so OSM has included a 'tiger:reviewed' tag for each data point. Reviewing this information showed that only 3073 of the 71,264 total 'tiger' data points have been reviewed. Due to the potential unreliability of this data I will be ignoring the tiger keys in my analysis

Street Names – Most issues with the 'addr:street' tags involved using multiple abbreviations for the same street type. These were fixed while creating the csv files, and prior to upload to SQL.

Cities – Data was cleaned and edited using python prior to uploading csv files to upload into SQL.

Zip Codes – The Ohio and OH entries were replaced with '00' to help make them easily identifiable in the database for manual correction, and all other zip codes were formatted to the 5-digit length.

Database Fixes

Street named Paula – During the cleaning stage, I encountered an issue with one street being named Paula, with no street type attached. Using the database, I found the ID of this entry, and cross-referenced it with the rest of the table to determine the road type.

A simple google search for Middleburg Heights Junior High Schools, shows it's address is on Paula Drive. The housenumber and postcode for the school match the information from OpenStreetMaps as

```
sqlite> select * from ways_tags where id=480858666;
480858666|housename|Middleburg Heights Junior High School|addr
480858666|housenumber|7000|addr
480858666|postcode|44130|addr
480858666|street|Paula Drive|addr
480858666|amenity|school|regular
480858666|area|yes|regular
480858666|ele|259|regular
480858666|county_id|035|gnis
480858666|created|07/12/1979|gnis
480858666|feature_id|1043294|gnis
480858666|state_id|39|gnis
480858666|name|Middleburg Heights Junior High School|regular
```

well.

```
sqlite> update ways_tags set value='Paula Drive' where id=480858666 and key='street';
```

```
sqlite> .open osm_project_cleveland
sqlite> select * from ways_tags where id=480858666;
480858666|housename|Middleburg Heights Junior High School|addr
480858666|housenumber|7000|addr
480858666|postcode|44130|addr
480858666|street|Paula|addr
480858666|amenity|school|regular
480858666|area|yes|regular
480858666|ele|259|regular
480858666|county_id|035|gnis
480858666|created|07/12/1979|gnis
480858666|feature_id|1043294|gnis
480858666|state_id|39|gnis
480858666|name|Middleburg Heights Junior High School|regular
```

Fixing Zip Codes – The zip codes that were not in a regular number format were changed to '00' for ease of identification within the database. Once in the database it was a simple matter of identifying the associated id's and then manually adjusting the zip codes as needed.

```
sqlite> select tags.id from (select * from ways_tags
   ...> union all select * from nodes_tags) tags
   ...> where tags.value='00' and tags.key='postcode';
237600110
498855658
498855659
498855743
498855745
```

Here are the results for the first id in the list.

```
sqlite> select * from (select * from ways_tags
   ...> union all select * from nodes_tags) tags
   ...> where tags.id = 237600110;
237600110|name|Seton Catholic School|regular
237600110|amenity|school|regular
237600110|building|yes|regular
237600110|city|Hudson|addr
237600110|street|Stow Road|addr
237600110|postcode|00|addr
237600110|housenumber|6923|addr
```

After a quick google search, Seton Catholic School is located at 6923 Stow Road, Hudson, OH 44236. The id for this tag was only in the ways_tags table.

```
sqlite> update ways_tags set value=44236 where id=237600110 and key='postcode';
```

```
sqlite> select * from ways_tags where id=237600110;
237600110|name|Seton Catholic School|regular
237600110|amenity|school|regular
237600110|building|yes|regular
237600110|city|Hudson|addr
237600110|street|Stow Road|addr
237600110|postcode|44236|addr
237600110|housenumber|6923|addr
```

Database Investigation

File Sizes

cleveland_ohio.osm.....650 MB
osm_project_cleveland.db.... 300 MB
nodes.csv..... 151 MB
nodes_tags.csv..... 4.69 MB
ways.csv..... 11.4 MB
ways_nodes.csv..... 50.8 MB
ways_tags.csv..... 35.0 MB

Number of ways

```
sqlite> select count(*) from ways;
count(*)
----------
197195
```

Number of nodes

```
sqlite> select count(*) from nodes;
count(*)
----------
1856783
```

Individual Users

```
sqlite> select count(users.uid)
   ...> from (select uid from nodes union select uid from ways) users;
count(users.uid)
---------------
1346
```

Top 10 Cities – Since my initial cleaning showed this dataset includes entries from a variety of locations let's take a look at the top 10 cities.

```
sqlite> 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;
```

The results show that Cleveland itself isn't even the most common city in this dataset.

value	count
Avon Lake	924
Cleveland	840
Cleveland Heights	770
Hudson	484
Akron	462
Kent	371
Lyndhurst	342
Euclid	212
Chardon	176
Ashtabula	174

I adjusted the column width on this table, as the initial column width was set to 10 and Cleveland and Cleveland Heights both appeared on this table as Cleveland. Giving a wider column showed the full names for these two separate cities.

Land usage

```
sqlite> select tags.value, count(*) count
   ...> from (select * from ways_tags union all select * from nodes_tags) tags
   ...> where tags.key='landuse'
   ...> group by tags.value
   ...> order by count desc
   ...> limit 10;
value                     count
------------------------  ----------
grass                     2048
residential               463
cemetery                  231
retail                    218
meadow                    196
forest                    192
reservoir                 191
farmland                  171
farm                      138
industrial                119
```

This look at land usage was a bit surprising. I have been to Cleveland many times, and never noticed that much grass in the city.

```
sqlite> select ways_tags.value, count(*) as count
   ...> from ways_tags join (select distinct(id) from ways_tags
   ...> where value='grass') grass
   ...> on ways_tags.id = grass.id
   ...> group by ways_tags.value
   ...> order by count desc
   ...> limit 10 offset 1;
value                count
-------------------  ----------
pitch                164
tee                  98
no                   94
golf                 93
green                93
yes                  61
fairway              58
Bing                 56
soccer               40
american_football    37
```

Based on these results it appears most of the grass is used for sports. I left off the top result, as it would simply include the grass values themselves, and not provide useful information about the uses of those areas. Golf appears to be the biggest use of grass in the city, with tee, golf, green, and fairway all likely referring to golf courses. The other big areas of grass seem to be sports fields with pitch, soccer and american football all in the top 10.

Conclusion

The biggest area for improvement discovered in this process was the need to clean and verify the Tiger data entries. It appears the process is underway, but there is a lot of work to do. Based on a brief scan there are over 67,000 data entries that would need to be verified to make this work complete. One issue that will make this difficult is the way the entries are organized. The Tiger data splits an address into more entries than the regular addr tags have, so comparing the two for matches would be extremely difficult programatically. The only way to really verify the Tiger information in the dataset is accurate would be to walk and drive around the city to manually verify each entry.

Another area that could use improvement is simply the method used to extract the data for these datasets. It was a huge file that included a large number of entries that are really not relevant to a Cleveland Ohio dataset. Perhaps identifying a better way to limit the area around a city included in these datasets would help limit this issue in the future.

Sources:

https://discussions.udacity.com/t/final-project-issue-with-update-street-type-and-update-postcode-functions/185747/6 – Udacity forum thread on issues with getting final csv files to take updates to street names. Was having similar issues, and noticed the suggestion to just compare the osm file with the mapping dictionary instead of the expected dictionary first. This fixed the issue I was running into with my updates.

http://wiki.openstreetmap.org/wiki/TIGER – OpenStreetMap Wikipedia page discussing the Tiger tags.

https://www.berea.k12.oh.us/Domain/106 – School website for Middleburg Heights Junior High confirming address as on Paula Drive.