OpenStreetMap Data Case Study

-Tom Nececkas

Map Area

Vermont, United States

- https://www.openstreetmap.org/relation/177415
- http://download.geofabrik.de/north-america/us/vermont.html

Living in Vermont feels so different than the suburban and urban areas where I lived previously. Even just a few minutes drive outside its biggest city, the Vermont landscape rapidly morphs into evergreen forests. Not too much farther and you can find dirt roads, some of which will take you to the next town, and others of which will fade into walking paths or disappear entirely.

That's part of the magic of living in Vermont -- driving in reverse on a winding, potholed dirt road with branches sliding along the car windows, because it's too narrow to turn around and the road had been worth exploring.

I wanted to look at the OpenStreetMap ("OSM") data for Vermont to see if its backwoods character was reflected in OSM's data. Vermont is in the United States, but even so, I wondered whether the OSM project might be heavily focused on urban areas, leaving holes in the data for rural areas like Vermont.

To be clear, the below text doesn't contain a full-blown analysis. But by explaining my interest in the Vermont data extract, I hoped to give context for why I wanted to clean and prepare this particular dataset for further analysis, and also for why I chose certain queries (SQL) in the initial analysis.

Problems in the Map Data

Before querying the data, I needed to process the XML data into CSV files and then SQL databases, and before that I wanted to audit and clean some problems.

To audit the data, I used the functions in the audit.py file to look for problems. I noticed some issues with the following that I wanted to change:

- Street names
- Postal codes

Street Names

To audit street names, I ran the audit_street_names function in the audit.py file to sort street names by their last word. I wanted to remove abbreviations to increase uniformity, by for instance changing 'Main St' to 'Main Street'.

I used the update_street_names function (see further below) in the data.py file to swap abbreviations for matching words in a dictionary.

However, ensuring uniformity became more difficult when cleaning numbered routes. Below are a few examples of how routes were named:

```
VT-14
VT 132
VT Route 12
State Route 15
Route 139
```

I decided to use the 'VT Route 12' format, but that presented some challenges.

While I could simply change 'VT-' to 'VT Route', I could not safely change 'VT' to 'VT Route'. The latter would correctly change 'VT 132' to 'VT Route 132', but it would also change 'VT Route 12' to 'VT Route Route 12'. Similarly, changing 'State' to 'VT' in 'State Route 15' works, but it doesn't work if the street name is 'State Street'.

For these reasons, I needed to write a function that sometimes looked ahead at the next word before making any changes:

```
NUMBERS = re.compile(r'\d+')

def update_street_name(name, mapping):
    words = re.split('VT-|US-| ', name)
    for w in range(len(words)):
        # substitutes abbreviation for word in mapping (see above dictionary)
        if words[w] in mapping:
            words[w] = mapping[words[w]]
        # if word is 'VT' or 'Vermont' and it's followed directly
        # by a 'word' starting with numbers (i.e. VT 100), then
        # it's changed to 'VT Route'
        if words[w] == "VT":
            if NUMBERS.search(words[w + 1]):
                words[w] = "VT Route"
        # changes 'State' to 'VT' if followed by 'Route'
        # this avoids changing 'State' in 'State Street'
        if words[w] == "State":
            if words[w + 1] == "Route":
                words[w] = "VT"
    name = " ".join(words)
    name = name.strip(' ,')
    return name
```

That function improved uniformity somewhat, but there was still a problem. I could change 'Route 139' into 'VT Route 139', but I wasn't sure that was accurate. Instead of a state route, 'Route 139' might be a county route or even a US route (the OSM data contained a 'US Route 2'). Without more research, I didn't know which is correct, so I opted not to alter any street names with the format

'Route 139'.

On a different note, I also noticed a number of street names that began with 'Rue':

```
SELECT value
FROM ways_tags
WHERE key = "street"
AND VALUE LIKE "Rue%"
LIMIT 2;
```

```
Rue Railroad
Rue Notre-Dame Ouest
```

'Rue' is French for 'street', and Vermont shares a border with French-speaking Quebec, Canada. It's possible some streets retain their French names when crossing into the U.S. side of towns that straddle the border. Of course, it's also possible that the data is completely erroneous.

To verify that the data points are at least close to Vermont, I ran a query to identify the maximum and minimum latitudes for streets that start with the name Rue.

```
SELECT min(n.lat), max(n.lat)
FROM nodes as n
JOIN nodes_tags AS nt ON n.id = nt.id
WHERE nt.key = "street"
AND nt.value LIKE "Rue%";
```

Minimum latitude: 45.0056916 degrees Maximum latitude: 45.0175017 degrees

This is roughly the latitude of Vermont's northern border, so this data might plausibly belong in the Vermont extract. However, it's also possible that the map data includes slightly more than it should. Since I wasn't sure, I kept the French street names and accompanying data in the dataset.

Postal Codes

To audit postal codes, I ran the audit_postcodes function in the audit.py file to identify postcodes that were not exactly 5 digits (e.g. '05460'). Below are a few examples:

```
05460-9998
VT 05143
J0B 3E0
```

When processing the data, I could use python's regular expression module and its .sub() function to remove any non-digits, and then I could slice the result to include only the first 5 digits. That made sense for zip codes like '05460-9998' or 'VT 05143'.

For zip codes like 'J0B 3E0', I was unsure that was the right approach.

I googled the zip code to confirm my suspicion that this was the format used for Canadian postcodes. This confirmed that at least some of the Vermont OSM extract is actually Canadian data.

However, that doesn't mean that all high-level XML elements for locations in Canada will necessarily include a Canadian postal code tag. If postal codes aren't a perfect indicator, they might be useful in figuring out a better indicator or combination of indicators.

For that reason, I kept the elements with Canadian postal codes in the data, and used regular expressions to do so.

```
CANADA_POSTCODE = re.compile(r'^J\S\S ')

def update_postcode(postcode):
    if CANADA_POSTCODE.search(postcode):
        postcode = postcode
    else:
        # remove any character or whitespace that isn't a number
        postcode = re.sub('\D', '', postcode)
        # take only 5 leftmost digits
        postcode = postcode[:5]

    return(postcode)
```

I used queries to do an initial check of whether Candian postal codes might be sufficient to identify all Candian data.

Number of ways with street names starting with 'Rue':

```
SELECT count(distinct(id)) as num
FROM ways_tags
WHERE key = "street"
AND value LIKE "Rue%";
```

26

Number of ways with Canadian postal codes:

```
SELECT count(distinct(id)) as num
FROM ways_tags
WHERE key = "postcode"
AND VALUE LIKE "J%";
```

14

More elements have street name tags starting with "Rue" than tags for Canadian postal codes. So the Vermont extract likely contains data points in Canada which cannot be identified by a postal code tag, or at least we can't rule out that possibility.

Overview of Data

After auditing and processing the data, I wanted to get a basic overview of file sizes and basic statitistics for the dataset.

File sizes

```
vermont.osm ....... 449 MB
osm_vermont.db .... 240 MB
nodes.csv ......... 184 MB
nodes_tags.csv ...... 4 MB
ways.csv ............ 8 MB
ways_tags.csv ...... 23 MB
ways_nodes.csv ..... 55 MB
```

Number of nodes

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

2,181,287

Number of ways

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

133,083

Number of unique users

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

1,101

Exploration of Data

Having cleaned the data and obtained some basic statistics, I was ready to do a basic analysis of the Vermont extract. My main question: Does the Vermont OSM data capture details about Vermont's backwoods roads and paths?

The most obvious way to check whether the OSM data captures Vermont's back roads is to look at the road surfaces.

Common road surfaces

```
sqlite> SELECT value, count(*) as num
FROM ways_tags
WHERE key = "surface"
GROUP BY value
ORDER BY num DESC
LIMIT 5;
```

```
asphalt..... 11,403
gravel....... 5,642
unpaved...... 2,334
dirt......... 1,576
paved.......... 458
```

For the type of road surface, I didn't have precise expectations. However, I expected that the number of unpaved roads would be comparable to the number of unpaved roads. The counts are comparable, since 'paved' includes the more specific 'asphalt,' and 'unpaved' includes th more specific 'gravel' and 'dirt.'

But since I don't know how many dirt roads there are in Vermont, the query results were not the most useful result.

I decided to look at tourism spots to judge whether the OSM data might be weighted more heavily towards urban areas.

Most numerous tourism spots

```
SELECT value, COUNT(*) as num
FROM nodes_tags
WHERE key = "tourism"
GROUP BY value
ORDER BY num DESC
LIMIT 5;
```

```
viewpoint..... 161
information... 148
camp_site...... 60
hotel.......... 52
museum......... 51
```

That viewpoints and campsites outnumber hotels and museums matches my expectations for Vermont. However, I was somewhat surprised that there were only nine more camp sites than museums.

I googled the number of campsites in Vermont, and found that the Vermont Campground Association provides details for over 100 campsites. Assuming the Vermont Campground

Association's data is accurate, and there are no differences between how they and OSM define campsites, this would seem to suggest that the Vermont OSM data is missing at least some data about rural places.

In the future, it would be useful to check which campsites are not included in the Vermont OSM data, and to find other tags and methods for assessing the completeness of the Vermont OSM data.

At present, though, I wanted to know more about who was collecting the data about Vermont's more out-of-the way places. I decided to look at contributors based on the type of data they were contributing.

Top Contributors for Dirt Roads versus Asphalt Roads

Top Contributors of Ways Tagged as Dirt Roads:

```
SELECT w.user, count(*) AS num
FROM ways AS w
JOIN ways_tags AS wt ON w.id = wt.id
WHERE wt.key = "surface"
AND wt.value = "dirt"
GROUP BY w.user
ORDER BY num DESC
LIMIT 5;
```

```
Adam Franco.......... 942
ZekeFarwell.......... 156
dpawlyk.............. 42
dufekin.............. 31
Martin868............ 29
```

Top Contributors of Ways Tagged as Asphalt Roads:

```
Adam Franco........ 6,576
maxerickson.......... 676
bot-mode............. 656
EdSS................. 206
user_599436.......... 189
```

I was surprised that the top contributors mostly don't overlap, aside from the go-getter Adam Franco. I wanted to see if the contributors for dirt roads were solely focused on more rural areas. I ran the following query to see whether the main contributors for dirt roads (excluding Adam Franco) had contributed to ways associated with more developed areas, meaning those with a paved or asphalt surface.

```
SELECT w.user, count(*) AS num
FROM ways AS w
JOIN ways_tags AS wt ON w.id = wt.id
```

```
WHERE (w.user = "ZekeFarwell" OR w.user = "dpawlyk" OR w.user = "dufekin"
OR w.user = "Martin868")
AND wt.key = "surface"
AND (wt.value = "asphalt" OR wt.value = "paved")
GROUP BY w.user
ORDER BY num DESC
```

```
Martin868..... 147
dufekin....... 95
ZekeFarwell..... 56
dpawlyk....... 17
```

So it doesn't look like there are contributors solely focused on mapping more rural areas, but it does seem that some contributors may be more conscious about including more rural areas.

Reflection

It's likely that Vermont's data is incomplete, but Vermont's back roads certainly have not been ignored. There's information about a large number of dirt roads and campsites, even if there are likely campsites and possibly dirt roads that haven't been included.

In terms of ideas for improving the dataset, I think it might make sense to partner with outdoor organizations. There are already plenty of organizations who are interested in more rural areas for recreation, such as hiking and biking groups. It might be possible to persuade some group members to get involved in contributing to the OSM project.

The most obvious benefit of recruiting members of outdoor organizations to contribute to the OSM data is that the data would be more complete for rural areas. A somewhat less obvious benefit would be that it'd be publicizing the existence and benefits of the OSM data to more people that aren't computer programmers.

That potential benefit carries with it a challenge. The OSM data is written in XML and interacting with computer code can be intimidating for people. Even though the OSM project has an interface that allows users to enter data by clicking on the map, the OSM wiki page on contributing map data suggests it could be useful for contributors to know about elements and tags. I could imagine that might start to seem like a lot to learn, particularly for a contributor who's only interested in contributing to one type of map data (e.g. information on campsites). One way to deal with that is to provide organizations with very specific instructions geared towards the exact type of information they would be contributing.

The other potential challenges relate to the effort required to contact organizations and persuade members to get involved. That's going to be a challenge with almost every volunteer project. And it's a challenge that can be overcome, as evidenced by the number of contributors and amount of map data involved with the OSM project already.