OpenStreetMap Sample Project

Data Wrangling with MongoDB

Xiaopeng Hu

November 2, 2015

Content

0. General Information:

- About the Dataset:

  Map Area: Harrisburg, PA, United States

  Link: https://s3.amazonaws.com/metro-extracts.mapzen.com/harrisburg_pennsylvania.osm.bz2

- Objective:

  Audit and clean the data set, converting it from XML to JSON format.

- References:

  Link: https://wiki.openstreetmap.org/wiki/Main_Page

1. Problems Encountered in the Map

A: Overview of the dataset:

The Harrisburg area map dataset was downloaded from MAP ZEN as the sample dataset for this project. The size of this file is around 67M so it is pretty difficult to quickly brows it with Notepad to get general ideas about the data. Therefore several Python codes were employed to collect the general ideas before taking actions to audit and clean the dataset.

With code1, a list of all / attributes were extracted. The result is listed in supporting material.

Eight element tags were found. *,<osm>* and *<bounds>* only appear once. By checking file head, it is apparent that *<osm /osm>* is for the root element, while element of *<bounds />* defines the boundary covered by this map dataset. The means of other six tags are straightforward. For the eighteen attributes, '*generator', 'maxlat', 'maxlon', 'minlat*' and '*minlon'* appear once. By checking the file head, it turns out that all these five are for *<bounds />* element. For left attributes, e.g., '*changeset'* appears 356369 times. According to *wiki.openstreetmap*, every record includes one '*changeset*' attribute, so there are totally 356369 records in this dataset. As for *'timestamp'* and '*version'*, they were counted one more times than 'changeset' because of their appearance in the *<bounds />* element.

B: General scheme of the dataset:

By referring with *wike.openstreetmap*, a general scheme of this dataset could be obtained as followings:

```
<osm
   <bounds k='#' />
   <node  k='#'  />
   <way   k='#'  />
      <nd   k='#'       />
      <tag  k='#'  v='#'>
   <relation   k='#'  />
      <member    k='#'       />
      <tag  k='#'  v='#'>
</osm>
```

C: Analysis of attributes in the dataset:

With code2, a list of all key values of attributes was extracted. For clarity，this long list was included in the supporting materials. It turns out that in second level tags, besides *"addr:###"* which need to be reformed to a dictionary type, there are more, such as *'gnis:', 'is_in:', 'name:', 'source:'* and *'tiger:'*. The important ZIPCODE is actually buried under 'tiger'. According to *wiki.openstreetmap*, *'tiger'* means *Topologically Integrated Geographic Encoding and Referencing system (TIGER)* data, and *'gnis'* means *USGS Geographic Names Information System (GNIS)*. Both are important data source of *openstreetmap project*, therefore it is better to keep all of them in the output JSON file.

D: A general rule for reforming records in the dataset

To produce a suitable JSON file for MongoDB, a general rule set was set to reform the records:

- *All attributes of "node", "way" and "relation" should be turned into regular key/value pairs*
- *For any second level tag, "k" values in format of "####:$$$$" should be added to a nested dictionary with "####" as main key, "$$$$" as sub-key, and "V" value as value of this sub-key*
- *Zipcode is included in a single array, with "zip" as key and value from "tiger_left_zip"*

Exception:

- *Attributes in the CREATED array should be added under a key "created"*
- *Attributes for latitude and longitude should be added to a "pos" array, the values inside "pos" array are floats*
- *All attributes of "member" in "relation" should be added into a "member" array*

This rule set was performed with code3.

E: Abbreviated street names and inconsistent phone numbers.

People used to use abbreviate in street name and different phone number styles. With code4, it turns out that this data set contains a few abbreviated street names such as *Dr, Rd., St. Blvd.* but with code5, phone numbers are indeed in many different styles. The output list was included in the supporting materials. Abbreviated street names can be easily corrected to full name with codes obtained from course. For phone numbers, it is fortunate to find out that they don't contain any extensions, so a simple code can be used to correct them. The correction procedures were included in code3.

2. Data Overview

Basic statistics about the dataset (MongoDB queries are included in code6):

harrisburg_pennsylvania.osm ..... 67 MB

harrisburg_pennsylvania.json ..... 91 MB

Number of documents 356369

Number of nodes 313907

Number of ways 42271

Number of unique users..... 292

Top 1 contributing user ..... '_id': 'cbley', 'count': 133202

Number of users appearing only once (having 1 post) 57

Zip code in Harrisburg ..... `[None, '17070', '17011', '17339', '17319', '17070; 17319', '17112', '17036', '17113', '17034', '17057', '17109', '17111', '17110', '17104', '17103', '17028', '17020', '17053', '17050', '17025', '17055', '17093', '17043', '17013']`

Top 10 amenity .....
```
[{'_id': 'parking', 'count': 698},
 {'_id': 'restaurant', 'count': 167},
 {'_id': 'school', 'count': 162},
 {'_id': 'place_of_worship', 'count': 119},
 {'_id': 'fast_food', 'count': 89},
 {'_id': 'fuel', 'count': 67},
 {'_id': 'bank', 'count': 39},
 {'_id': 'university', 'count': 33},
 {'_id': 'grave_yard', 'count': 28},
 {'_id': 'post_box', 'count': 27}]
```

Unique Phone Numbers .....seeing supporting materials

3. Additional Ideas

The number of nodes is overwhelming (~88%, ~12% for way). Since most of nodes actually only provide coordination for 'refs' used by 'way', the 'created' array of node is not needed at all on map drawing and querying purposes. Because MongoDB needs lots

storage for data index, the actual files for MongoDB have a total size of 208M, twice of the original JSON file. If we remove the 'created' array from nodes, we might significantly reduce the working file size, and may improve query speed (fewer indexes).

There is no official .xsd Schema exists. It may bring flexibility to developers, but it also may bring troubles in data consistency. A major problem occurred here is how do identify fields requiring correction. An official schema with both immutable key elements and expendable elements could be a solution to limit/exclude incorrect records. This schema should be maintained and upgraded monthly or yearly, so developers will share a common guide of data collection and at the same time, bring new features to the dataset. The new coming data can be easily merged and new features, if acceptable to most users, can be updated to both immutable key elements. The whole dataset then can be released as stable version and beta version.

4. Conclusion

The dataset of Harrisburg is relatively clean. Besides incorrect records found here, there might be more. A workable and efficient method is needed to audit the dataset. On the other hand, although the data were contributed by many users, it is noticeable that a few users contributed a huge part of all data. The reason could be that these dominant users just imported data from other dataset (such as TIGER) with programs. Dirty data, if there any, are likely contributed manually form individual users. An official .xsd Schema could significantly limit dirty data and reduce the effort for data audition.

5. Code

Seeing acctahed ipython notebook file.

6. Supporting materials

Output of Code1:

```
[{'bounds': 1,
  'member': 1932,
  'nd': 384529,
  'node': 313907,
  'osm': 1,
  'relation': 191,
  'tag': 147259,
  'way': 42271},
 {'changeset': 356369,
  'generator': 1,
  'id': 356369,
  'k': 147259,
  'lat': 313907,
  'lon': 313907,
  'maxlat': 1,
  'maxlon': 1,
  'minlat': 1,
  'minlon': 1,
  'ref': 386461,
  'role': 1932,
  'timestamp': 356370,
  'type': 1932,
  'uid': 356369,
  'user': 356369,
```

```
  'v': 147259,
  'version': 356370}]
```

Output of Code2:

```
{'Address': 1,
 'Agency': 1,
 'FID': 23,
 'FIXME': 60,
 'FIXME:old_ref': 1,
 'Hist_Distr': 1,
 'ISO3166-1': 1,
 'ISO3166-1:alpha2': 1,
 'ISO3166-1:alpha3': 1,
 'ISO3166-1:numeric': 1,
 'ISO3166-2': 1,
 'NAME': 9,
 'NHS': 149,
 'abbreviation': 1,
 'access': 2751,
 'addr:city': 405,
 'addr:country': 3,
 'addr:housename': 16,
 'addr:housenumber': 438,
 'addr:interpolation': 7,
 'addr:postcode': 334,
 'addr:state': 215,
 'addr:street': 405,
 'admin_level': 102,
 'aeroway': 61,
 'alt_name': 7,
 'alt_name:vi': 1,
 'amenity': 1650,
 'area': 80,
 'atm': 16,
 'attribution': 136,
 'authoritative': 9,
 'barrier': 54,
 'bicycle': 339,
 'bicycle_parking': 1,
 'board_type': 1,
 'boat': 1,
 'border_type': 17,
 'boundary': 103,
 'brand': 5,
 'bridge': 510,
 'bridge:structure': 7,
 'bridge:support': 8,
 'building': 25268,
 'building:levels': 19,
 'bus': 1,
 'cables': 7,
 'capacity': 16,
 'capacity:disabled': 1,
 'capital': 1,
 'cave:access': 1,
 'cave:difficulty': 1,
 'census:population': 12,
 'center_turn_lane': 1,
 'clopin:id': 9,
 'clopin:route': 9,
 'code': 1,
 'collection_times': 9,
 'colour': 2,
 'covered': 3,
 'craft': 1,
 'created_by': 1324,
 'crossing': 54,
 'cuisine': 146,
 'cutting': 5,
```

```
'cycle_network': 3,
'cycleway': 33,
'cycleway:right': 1,
'delivery': 2,
'denomination': 44,
'designation': 7,
'destination': 74,
'destination:ref': 59,
'destination:ref:to': 9,
'destination:street': 8,
'destination:street:to': 3,
'diocese': 1,
'direction': 9,
'dispensing': 5,
'drive_through': 2,
'education': 1,
'ele': 525,
'electrified': 142,
'email': 4,
'embankment': 17,
'emergency': 6,
'exit_to': 15,
'fee': 4,
'fixme': 5,
'flag': 1,
'foot': 369,
'footway': 65,
'frequency': 34,
'from': 2,
'fuel:diesel': 1,
'fuel:lpg': 1,
'gauge': 196,
'generator:source': 1,
'gnis:Class': 197,
'gnis:County': 197,
'gnis:County_num': 197,
'gnis:ST_alpha': 197,
'gnis:ST_num': 197,
'gnis:county_id': 286,
'gnis:county_name': 42,
'gnis:created': 295,
'gnis:feature_id': 331,
'gnis:feature_type': 9,
'gnis:id': 197,
'gnis:import_uuid': 33,
'gnis:reviewed': 33,
'gnis:state_id': 286,
'golf': 19,
'guage': 4,
'hgv': 559,
'hgv:national_network': 142,
'highway': 15473,
'historic': 8,
'history': 1,
'hoops': 1,
'horse': 279,
'hours': 1,
'iata': 2,
'icao': 2,
'ident': 1,
'import_uuid': 197,
'information': 1,
'internet_access': 3,
'is_in': 265,
'is_in:continent': 1,
'is_in:country': 33,
'is_in:country_code': 32,
'is_in:iso_3166_2': 32,
'is_in:state': 48,
'is_in:state_code': 33,
'junction': 18,
```

```
 'junction:ref': 7,
 'landuse': 250,
 'lanes': 1018,
 'lanes:backward': 3,
 'lanes:extra': 2,
 'lanes:forward': 6,
 'layer': 509,
 'lcn': 45,
 'lcn_ref': 1,
 'leisure': 339,
 'lengths:right': 2,
 'level': 1,
 'lit': 5,
 'loc_name': 2,
 'location': 1,
 'man_made': 27,
 'maxspeed': 585,
 'memorial': 2,
 'microbrewery': 2,
 'military': 1,
 'modifier': 1,
 'motor_vehicle': 18,
 'name': 10786,
 'name:ab': 1,
 'name:ace': 1,
 'name:af': 2,
 'name:als': 1,
 'name:am': 1,
 'name:an': 1,
 'name:ang': 1,
 'name:ar': 2,
 'name:arc': 2,
 'name:arz': 1,
 'name:as': 1,
 'name:ast': 1,
 'name:av': 1,
 'name:ay': 1,
 'name:az': 2,
 'name:ba': 1,
 'name:bar': 1,
 'name:bat-smg': 1,
 'name:bcl': 1,
 'name:be': 1,
 'name:be-x-old': 1,
 'name:bg': 2,
 'name:bi': 1,
 'name:bm': 1,
 'name:bn': 2,
 'name:bo': 1,
 'name:bpy': 1,
 'name:br': 1,
 'name:bs': 1,
 'name:bxr': 1,
 'name:ca': 2,
 'name:cbk-zam': 1,
 'name:cdo': 1,
 'name:ce': 1,
 'name:ceb': 1,
 'name:chr': 1,
 'name:chy': 1,
```


Output of Code4:


```
{'260': {'Carlisle Pike #260'},
 '550': {'Carlisle Pike #550'},
 'Blvd': {'High Pointe Blvd'},
 'Bypass': {'Camp Hill Bypass'},
 'Circle': {'Brenneman Circle'},
 'Dr': {'Mapleton Dr'},
 'Miller': {'Miller'},
```

```
 'Pike': {'Carlisle Pike', 'Gettysburg Pike'},
 'Rd.': {'Williams Grove Rd.'},
 'Rear)': {'West Main Street (Rear)'},
 'St': {'Paxton St'},
 'Streets': {'12th & Herr Streets',
             '15th & Vernon Streets',
             '18th & Walnut Streets',
             '19th & Forster Streets',
             '3rd and Division Streets',
             'Norwood & Holly Streets',
             'Penn & Sayford Streets',
             'Summit & King Streets'},
 'Terrace': {'Whitetail Terrace'},
 'Way': {'Market Plaza Way'}}
```

Part Output of Code5:

```
['717-975-0940',
 '717-975-0940',
 '+1 717 697 4641',
 '+1 717 697 4641',
 '(717) 566-0455',
 '(717) 566-0455',
 '+1 717 957 0188',
 '+1 717 957 0188',
 '+1 717 9018277',
 '+1 717 9018277',
 '(717) 541-1669',
 '(717) 541-1669',
 '(717) 545-4709',
 '(717) 545-4709',
 '(717) 540-4606',
 '(717) 540-4606',
 '(717) 657-3203',
 '(717) 657-3203',
 '(717) 652-6700',
 '(717) 652-6700',
 '(717) 566-8799',
 '(717) 566-8799',
 '+717 558 4150',
 '+717 558 4150',
 '(717) 652-5340',
 '(717) 652-5340',
 '(717) 657-5600',
 '(717) 657-5600',
 '717-233-7358',
 '717-233-7358',
 '(717) 236-1680',
 '(717) 236-1680',
 '717-232-7374',
 '717-232-7374',
 '+1 717 5455930',
 '+1 717 5455930',
 '717.795.1700',
 '717.795.1700',
 '(717) 657-9252',
 '(717) 657-9252',
 '+1 717 7379464',
 '+1 717 7379464',
 '+1 717 9209464',
 '+1 717 9209464',
 '717-620-9420',
 '717-620-9420',
 '(717) 652-1415',
 '(717) 652-1415',
 '(717) 545-6424',
 '(717) 545-6424',
 '(717) 545-4254',
 '(717) 545-4254',
```

```
'(717) 671-6663',
'(717) 671-6663',
'(717) 652-7601',
'(717) 652-7601',
'(717) 652-4900',
'(717) 652-4900',
'(717) 541-8790',
'(717) 541-8790',
'(717) 652-8378',
'(717) 652-8378',
'(717) 697-2142',
'(717) 697-2142',
'(717) 695-4888',
'(717) 695-4888',
'(717) 635-8991',
'(717) 635-8991',
'(717) 236-3626',
'(717) 236-3626',
'(717) 412-7415',
'(717) 412-7415',
'+1 717 795 9200',
'+1 717 795 9200',
'+1 717 761 1803',
'+1 717 761 1803',
'+1 717 774 7503',
'+1 717 774 7503',
'+1 717 737 0395',
'+1 717 737 0395',
'+1 717 730 2090',
'+1 717 730 2090',
'+1 717 763 0430',
'+1 717 763 0430',
'717-957-4481',
'717-957-4481',
'717-265-9448',
'717-265-9448',
'717-737-5344',
'717-737-5344',
'717-761-4530',
'717-761-4530',
'717-233-6551',
'717-233-6551',
'717-957-2030',
'717-957-2030',
'+1 717 730 4401',
'+1 717 730 4401',
'+1 717 7309464',
'+1 717 7309464',
'+17177287482',
'+17177287482',
'+1 800 237 7288',
'+1 800 237 7288',
'+1 717 409 3102',
'+1 717 409 3102',
'717-761-1865',
'717-761-1865',
'717-737-4513',
'717-737-4513',
'(717) 796-6585',
'(717) 796-6585',
'+1 717 737 6560',
'+1 717 737 6560',
'+1 717 737 3550',
'+1 717 737 3550',
'+1 717 761 6040',
'+1 717 761 6040',
'+1 717 761 5121',
'+1 717 761 5121',
'+1 717 731 1095',
'+1 717 731 1095',
'+1 717 257 1270',
```

```
'+1 717 257 1270',
'+1 717-732-2623',
'+1 717-732-2623',
'717-975-0940',
'+1 717 697 4641',
'(717) 566-0455',
'+1 717 957 0188',
'+1 717 9018277',
'(717) 541-1669',
'(717) 545-4709',
'(717) 540-4606',
'(717) 657-3203',
'(717) 652-6700',
'(717) 566-8799',
'+717 558 4150',
'(717) 652-5340',
'(717) 657-5600',
'717-233-7358',
'(717) 236-1680',
'717-232-7374',
'+1 717 5455930',
'717.795.1700',
'(717) 657-9252',
'+1 717 7379464',
'+1 717 9209464',
'717-620-9420',
'(717) 652-1415',
'(717) 545-6424',
'(717) 545-4254',
'(717) 671-6663',
'(717) 652-7601',
'(717) 652-4900',
'(717) 541-8790',
'(717) 652-8378',
'(717) 697-2142',
'(717) 695-4888',
'(717) 635-8991',
'(717) 236-3626',
'(717) 412-7415',
'+1 717 795 9200',
'+1 717 761 1803',
'+1 717 774 7503',
'+1 717 737 0395',
'+1 717 730 2090',
'+1 717 763 0430',
'717-957-4481',
'717-265-9448',
'717-737-5344',
'717-761-4530',
'717-233-6551',
'717-957-2030',
'+1 717 730 4401',
'+1 717 7309464',
'+17177287482',
'+1 800 237 7288',
'+1 717 409 3102',
'717-761-1865',
'717-737-4513',
'(717) 796-6585',
'+1 717 737 6560',
'+1 717 737 3550',
'+1 717 761 6040',
'+1 717 761 5121',
'+1 717 731 1095',
'+1 717 257 1270',
'+1 717-732-2623']
```

Last Output of Code6:

```
All phone numbers
[{'_id': '7177322623'},
 {'_id': '7177616040'},
 {'_id': '7177966585'},
 {'_id': '7177374513'},
 {'_id': '7177611865'},
 {'_id': '7177373550'},
 {'_id': '7174093102'},
 {'_id': '7177615121'},
 {'_id': '8002377288'},
 {'_id': '7172363626'},
 {'_id': '7177287482'},
 {'_id': '7177614530'},
 {'_id': '7179574481'},
 {'_id': '7177747503'},
 {'_id': '7177375344'},
 {'_id': '7177611803'},
 {'_id': '7177630430'},
 {'_id': '7175456424'},
 {'_id': '7174127415'},
 {'_id': '7177309464'},
 {'_id': '7176527601'},
 {'_id': '7176954888'},
 {'_id': '7177370395'},
 {'_id': '7176524900'},
 {'_id': '7179570188'},
 {'_id': '7179018277'},
 {'_id': '7177951700'},
 {'_id': '7172336551'},
 {'_id': '7177379464'},
 {'_id': '7176358991'},
 {'_id': '7175454254'},
 {'_id': '7172571270'},
 {'_id': '7176209420'},
 {'_id': '7176974641'},
 {'_id': '7177311095'},
 {'_id': '7175454709'},
 {'_id': '7175668799'},
 {'_id': '7179209464'},
 {'_id': '7176716663'},
 {'_id': '7176579252'},
 {'_id': '7177376560'},
 {'_id': '7175660455'},
 {'_id': '7176521415'},
 {'_id': '7172327374'},
 {'_id': '7175455930'},
 {'_id': '7176528378'},
 {'_id': '7176972142'},
 {'_id': '7172337358'},
 {'_id': '7176575600'},
 {'_id': '7177302090'},
 {'_id': '7179750940'},
 {'_id': '7176525340'},
 {'_id': '7177959200'},
 {'_id': '7177304401'},
 {'_id': '7175584150'},
 {'_id': '7176526700'},
 {'_id': '7176573203'},
 {'_id': '7172659448'},
 {'_id': '7172361680'},
 {'_id': '7175404606'},
 {'_id': '7179572030'},
 {'_id': '7175418790'},
 {'_id': '7175411669'}]
```