OpenStreetMap Project
Data Wrangling with MongoDB

Yang Bo

Map Area: New Delhi, India

1. Problems Encountered in the Map

After initially downloading a small sample size of the New Delhi, India and running it against the audit.py and tag.py file, I noticed many main problems with the data, which I will discuss in the following order:

Hindi-transliteration street name

Once the data was running against the audit.py, it revealed Hindi-transliteration street name like Marg or Wali (In Hindi, it means road) is quite prevalent in the Delhi map. It is not a mistake, but still need to be noticed in the later analysis
'Wali': set(['Gali Chandi Wali'])
'Marg': set(['Amrita Shergil Marg', 'Choudhary Hukum Chand Marg', 'Indraprastha Marg', 'Kasturba Gandhi Marg', 'Vinay Marg', 'Rao Tula Ram Marg', 'Ramakrishna Ashram Marg', 'Mahatma Gandhi Marg', 'Nyaya Marg', 'Prithviraj Marg', 'Bahadur Shah Zafar Marg', 'August Kranti Marg', 'Chaudhary Dalip Singh Marg', 'Tees January Marg', 'Bahadur shah Zafar Marg', 'Africa Avenue Marg', 'Aurobindo Marg', 'Benito Juarez Marg']

Street Names including the district name

Some street names including the district name, I simply delete the district name.

‘Area': set(['Rama Road, Industrial Area']),
'Gate': set(['Lothian Road, Kashmere Gate'])
'Main)': set(["Lawyer's Street, Green Park (Main)"]),

Wrong Label of Street Names

Wrong “Market” label of street names , I simply delete the wrong “Market” labels.
'Market': set(['Defence Colony Market', 'Khan Market', 'Naoroji Nagar Market', 'Sujan Singh Park, Subramania Bharti Marg,Behind Khan Market'])})

Wrong Postcode

Some address have wrong postcode, most of them are have wrong length as below, so I write two function to audit and clean it.

```
 "address": {
    "city": "New Delhi",
    "housename": "N5 South Extension Part 1",
    "postcode": "1100049"
  },
"address": {
    "city": "New Delhi",
    "street": "Hailey Road",
    "housenumber": "12",
    "postcode": "110 001"
  },
 "address": {
    "housenumber": "4, Block B, IP Estate",
    "street": "Indraprastha Marg",
    "housename": "School of Planning and Architecture",
    "postcode": "1100002"
  },
invalid = 0
```

```
def audit(osmfile):
    osm_file = open(osmfile, "r")
    street_types = defaultdict(set)
    postal_code = defaultdict(int)
    for event, elem in ET.iterparse(osm_file, events=("start",)):

        if elem.tag == "node" or elem.tag == "way":
            for tag in elem.iter("tag"):
                if is_street_name(tag):
                    audit_street_type(street_types, tag.attrib['v'])
                if is_postal_code(tag):
                    audit_postal_code(postal_code,tag.attrib['v'])
    osm_file.close()
    return [postal_code, street_types]

new_postal_code = 110001


def audit_postal_code(invalid_postal_codes,postal_code):
#checks if postal code have right length
    if len(postal_code) != 6:
        invalid_postal_codes[postal_code] += 1



def update_postal_code(postal_code):
#checks if postal code have right length, if not replaces with 110001 default
    if len(postal_code) != 6:

        return new_postal_code
```

2. Data Overview

This section contains basic statistics about the dataset and the MongoDB queries used to gather them.

File sizes

map2.osm ..... 53.9 MB
map2.osm.json ....79.2 MB

Number of documents

```
> db.map2.find().count()
286134
```

Number of nodes

```
> db.map2.find({"type":"node"}).count()
243114
```

Number of ways

```
> db.map2.find({"type":"way"}).count()
43018
```

Number of unique users

```
> db.map2.distinct('created.user').length
351
```

Top 1 contributing user

```
> db.map2.aggregate([{
...                 $group:{
...                     '_id':'$created.user'
...                    , 'count':{$sum:1}
...                  }
...               }, {
...                 $sort:{'count':-1}
...               }, {
...                 $limit:1
...               }])

{ "_id" : "saikumar", "count" : 41282 }
```

Number of users appearing only once (having 1 post)

```
> db.map2.aggregate([{
...                     $group:{
...                         '_id':'$created.user'
...                        , 'count':{$sum:1}
...                     }
...                   },{
...                     $group:{
```

```
...                         '_id':'$count'
...                       , 'num_users':{$sum:1}
...                     }
...                 },{
...                     $sort:{
...                         '_id':1
...                     }
...                 },{
...                     $limit:1
...                 }])
{ "_id" : 1, "num_users" : 72 }
```

So, 72 users contributed to only one post.

3. Additional Ideas

Contributor statistics

The contributions of users seems not as skewed as in the charlotte examples. Here are some user percentage statistics:

Top user contribution percentage (“saikumar”) - 14.4%
Combined top 2 users' contribution (“saikumar” and “bindhu”) - 23.2%

```
{ "_id" : "saikumar", "count" : 41282 }
{ "_id" : "bindhu", "count" : 24994 }
```

Additional data exploration using MongoDB queries

```
# Top 10 appearing amenities

> db.map2.aggregate([{
...                     $match:{
...                         'amenity':{$exists:1}
...                         }
```

```
...                 },{
...                     $group:{
...                        '_id':'$amenity'
...                      , 'count':{$sum:1}
...                        }
...                 },{
...                     $sort:{'count':-1}
...                 },{
...                     $limit:5
...                 }])
{ "_id" : "school", "count" : 117 }
{ "_id" : "place_of_worship", "count" : 89 }
{ "_id" : "parking", "count" : 66 }
{ "_id" : "restaurant", "count" : 49 }
{ "_id" : "embassy", "count" : 40 }
{ "_id" : "fast_food", "count" : 35 }
{ "_id" : "atm", "count" : 30 }
{ "_id" : "hospital", "count" : 28 }
{ "_id" : "fuel", "count" : 28 }
{ "_id" : "public_building", "count" : 26 }
```

Indian people attached a lot of importance to education. So school is the top 1 amenities is not that surprise. Since religion really hold a very important position in Indian people's daily life, worship place ranking No.2 is also very reasonable. The map I choose including the embassy area --Chanakyapuri. That area is also the headquarters of many indian government departments, state-owned buildings and so on.

Biggest religion (slightly surprise here)

```
> db.map2.aggregate([{
...                     $match:{
...                        "amenity":{$exists:1}
...                      , "amenity":"place_of_worship"
...                     }
...                   },{
...                     $group:{
```

```
...                     "_id":"$religion"
...                   , "count":{$sum:1}
...                   }
...                 },{
...                   $sort:{"count":-1}
...                 },{
...                   $limit:5
...                 }])
{ "_id" : null, "count" : 24 }
{ "_id" : "muslim", "count" : 23 }
{ "_id" : "hindu", "count" : 19 }
{ "_id" : "christian", "count" : 11 }
{ "_id" : "sikh", "count" : 9 }
```

It is slightly surprise that muslim amenities is a little more than hindu's, since hindu is the largest religion in India. However, the area I chose did includes a very big Jama Masjid and muslim community. So it is also possible. Christian is also become more and more popular in India, especially in the big cities. This statistics reflects this trend.

Most popular cuisines（No surprise here）

```
> db.map2.aggregate([{
...                   $match:{
...                     "amenity":{$exists:1}
...                   , "amenity":"restaurant"
...                     }
...                 },{
...                   $group:{
...                     "_id":"$cuisine"
...                   , "count":{$sum:1}
...                   }
...                 },{
...                   $sort:{"count":-1}
...                 },{
...                   $limit:5
...                 }])
{ "_id" : null, "count" : 29 }
```

{ "_id" : "indian", "count" : 3 }
{ "_id" : "chinese", "count" : 2 }
{ "_id" : "sandwich", "count" : 2 }
{ "_id" : "North_Indian", "count" : 2 }

No doubt that indian cuisine is the most popular one in India. Chinese ranked No.2 is a little surprised. However, the number of it is same with sandwich and north_indian. So it is also reasonable.

Conclusion

After this review of the data it's obvious that the Delhi area is incomplete, though I believe it has been well cleaned for the purposes of this exercise.
The problems includes some common problems like not accurate enough as well as some very specific problem for India like the Hindi-transliteration.
It interests me to notice that the Google map is much more accurate than OpenStreetMap, so it is possible to cleaned data of OpenStreetMap.org under its help.Also some background knowledge is also very important.
We could use the google maps api to retrieve data from google maps and improve the data of OSM. It has a Google Maps JavaScript API. however, it has a limitation of 25000 calls limited(After 90 days over 25000 calls, you have to pay for the extra calls.) The API Policy's change has make people doubt about Google Maps reliability. I don't think Google Maps can always be more reliable OSM, but its overall data quality is definitely better than OSM. How to keep the volunteers engaging in the data collection, auditing and cleaning is really a challenge for OSM.

In the future, I also think a data analyst shall more cooperate with other experts who have specialised knowledge background. Or maybe the expert shall learn some important data analysis techniques.