Wrangle OpenStreetMap Data

Ada Lee

October 8, 2015

Map Area: Shanghai, China

https://www.openstreetmap.org/relation/913067
shanghai_china.osm.bz2

1. Problems encountered in the map

1.1 Ignore tag k="type"

I notice for some ways, there are sub tag called 'type', I don't add it since it would override {'type':'way'}, this would make counting number of ways difficult. Below is an example.

```
<way id="143624326" ... uid="253683" user="sinopitt">...
        ...
        <tag k="type" v="public"/>
        ...
</way>
```

1.2 Audit street name

Some Street name is using abbreviation or has misspell, for example, "Rd." is abbreviation for "Road", and some names have spelling error, for example, "Rode" should be "Road", I use a dictionary `street_mapping` to map from error or abbreviated name to correct name. See my auditing results for street in `auditing_street.txt`.

```
street_mapping = { u'Rd\.': u'Road', u'Rd': u'Road', u'Rode': u'Road',
u'Roaf': u'Road', u'Ave\.': u'Avenue', u'avenue': u'Avenue' }
```

1.3 Audit city name

There are many spellings for one city, or the content of city field is not city, but about town name, or district name. I use regular expression to find all problematic names and change them to

correct name. Below is mapping dictionary `city_mapping` that I created. See my auditing results for cities in `auditing_city.txt`.

```
city_mapping = { u'杭州': u'Hangzhou', u'[h]an\w\whou': u'Hangzhou',
u'上海': u'Shanghai', u'shanghai': u'Shanghai', u'.+Shanghai':
u'Shanghai', u'Shanghai.+': u'Shanghai', u'松江': u'Shanghai', u'闵行':
u'Shanghai', u'金山区': u'Shanghai', u'友谊路街道': u'Shanghai', u'枫泾
镇': u'Shanghai', u'Anting': u'Shanghai', u'临港新城': u'Shanghai', u'浦
江漕河泾高科技园区': u'Shanghai', u'苏州': u'Suzhou', u'Suzhou.+':
u'Suzhou', u'Kun[Ss]han': u'Kunshan', u'昆山市': u'Kunshan', u'宁波':
u'Ningbo', u'泰兴': u'泰兴市', u'新市镇': u'Huzhou', u'湖州市': u'Huzhou',
u'嘉兴': u'Jiaxing', u'jiaxing': u'Jiaxing', u'无锡': u'Wuxi', u'新埭镇':
u'平湖市', u'南京': u'南京市', u'Nantong': u'Nantong', u'南通':
u'Nantong', u'Qidong': u'Nantong', u'fenghua': u'Fenghua' }
```

Now there are only 23 cities, and they all are correct different city names.

```
>db.shanghai.distinct("address.city")
#["Shanghai", "Hangzhou", "Kunshan", "Ningbo", "Nantong", "Suzhou",
"Shaoxing", "泰兴市", "Huzhou", "张家港市", "溧阳市", "Hanzghou", "南京市",
"Wuxi", "平湖市", "Zhoushan", "Jiaxing", "扬州", "如皋", "镇江",
"Fenghua", "靖江", "太仓市"]
```

1.4 Audit postcode

Postcode should only consists of 6 numbers, I find some post codes consist of 6 numbers plus some Chinese, for this kind of incorrect post codes, I only need to remove Chinese, and some postcode has only 5 numbers, this kind of postcode need to correct them manually.

- regular expression to specify correct post code: `u'^[0-9]{6}$'`
- regular expression to extract numbers form incorrect post code: `u'[0-9]+'`

There are only two postcode errors that we could correct automatically, we can find it in `auditing_postcode.txt`:

```
Audit postcode: "201315 上海" --> "201315"
Audit postcode: "201315 上海" --> "201315"
```

There are also 3 postcode errors that we could not correct automatically, we can find it in `auditing_postcode.txt`

```
Could not correct postcode "21351", please investigate manually.
Could not correct postcode "20032", please investigate manually.
Could not correct postcode "20032", please investigate manually.
```

We need to investigate in mongodb

```
## investigate problematic postcodes
>db.shanghai.find({"address.postcode":{"$in":["21351", "20032"]}},
{"address":1,"_id":0})
#{ "address" : { "city" : "溧阳市", "street" : "竹箦镇", "housenumber" :
"南旺村", "postcode" : "21351" } }
#{ "address" : { "city" : "Shanghai", "street" : "中山南二路",
"housenumber" : "699", "postcode" : "20032" } }
#{ "address" : { "city" : "Shanghai", "street" : "龙腾大道",
"housenumber" : "2555", "postcode" : "20032" } }

## correct postcodes
>db.shanghai.update({"address.postcode":"21351"}, {"$set":
{"address.postcode":"213351"}})
#WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>db.shanghai.update({"address.postcode":"20032","address.street":"中山
南二路"},{"$set":{"address.postcode":"200032"}})
#WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>db.shanghai.update({"address.postcode":"20032","address.street":"龙腾
大道"},{"$set":{"address.postcode":"200232"}})
#WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

1.5 Incorrect field name: postal_code

There was one item has postal_code field, should be "address.postcode"

```
>db.shanghai.find({"postal_code": {$exists:1}},{"postal_code":
1,"address":1,"_id":0,"id":1})
#{ "postal_code" : "310014", "id" : "30067595" }
```

```
## update, rename "postal_code" to "address.postcode"
>db.shanghai.update({"postal_code": {$exists:1}},{"$rename":
{"postal_code":"address.postcode"}})
#WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

## check updated correctly
>db.shanghai.find({"id":"30067595"},{"postal_code":1,"address":
1,"_id":0,"id":1})
#{ "id" : "30067595", "address" : { "postcode" : "310014" } }
```

2. Overview of the data

2.1 File sizes

shanghai_china.osm …….. 445 MB
shanghai_china.osm.json ……. 513 MB

2.2 MongoDB Queries

```
## Number of documents
>db.shanghai.count()
#2353972

## Number of nodes
>db.shanghai.find({"type":"node"}).count()
#2107126

## Number of ways
>db.shanghai.find({"type":"way"}).count()
#246846

## Number of unique users
>db.shanghai.distinct("created.user").length
#1352

## Top 1 contributing user
>db.shanghai.aggregate([{"$group":{"_id":"$created.user","count":
{"$sum":1}}},{"$project":{"_id":0,"user":"$_id","count":1}},{"$sort":
```

```
{"count":-1}},{"$limit":1}])
#{ "count" : 179330, "user" : "Chen Jia" }

## Number of users appearing only once (having 1 post)
db.shanghai.aggregate([{"$group":{"_id":"$created.user","count":
{"$sum":1}}},{"$match":{"count":1}},{"$group":
{"_id":"users_appearing_only_once", "count":{"$sum":1}}}])
#{ "_id" : "users_appearing_only_once", "count" : 221 }
```

3. Other ideas about the datasets

3.1 Number of ways or nodes created each year

```
>db.shanghai.aggregate([{"$project":{"year":{"$substr":
["$created.timestamp",0,4]},"_id":0}},{"$group":{_id:"$year",count:
{$sum:1}}},{"$project":{"year":"$_id","_id":0,"count":1}},{ "$sort":
{"year":1}}])
#{ "count" : 84067, "year" : "2007" }
#{ "count" : 19699, "year" : "2008" }
#{ "count" : 40109, "year" : "2009" }
#{ "count" : 86208, "year" : "2010" }
#{ "count" : 179180, "year" : "2011" }
#{ "count" : 379451, "year" : "2012" }
#{ "count" : 346814, "year" : "2013" }
#{ "count" : 561990, "year" : "2014" }
#{ "count" : 656454, "year" : "2015" }
```

We can see that map data uploaded by user in a year has a trend to increase over time.

Incomplete address

There are 1342 addresses only having pure numeric house number and not having street field, house number without street is not complete.

```
>db.shanghai.find({"address.housenumber":{"$exists":1,"$regex":/
^[0-9]+$/},"address.street": {"$exists":0}},{"address":1,"_id":0})
#{ "address" : { "housenumber" : "119" } }
#{ "address" : { "housenumber" : "479" } }
#{ "address" : { "housenumber" : "1" } }
```

```
#...
#{ "address" : { "housenumber" : "109" } }
#{ "address" : { "housenumber" : "115" } }
#{ "address" : { "housenumber" : "329" } }
#{ "address" : { "housenumber" : "345" } }
#Type "it" for more

## how many imcomplete address
>db.shanghai.find({"address.housenumber":{"$exists":1,"$regex":/
^[0-9]+$/},"address.street": {"$exists":0}},{"address":1,"_id":
0}).count()
#1342
```

4. Conclusion

After investigating map data in Shanghai, we find that there are many error messages, for example for one city there are many name variations, and the content of city sometimes is not city but town name or address. Another example is street, there are many misspell or abbreviations. Postcode is not all correct as well. What's more, some address only consist of house number, and street message is missing. In summary, this map contains some error, lack some standard and miss some information.

I have some suggestion to improve this dataset. Firstly, for some field like city, do not let user to fill this field, but give them choices to check. Secondly, for some field like postcode, we need to valid content, for example, postcode is China must only consist of 6 numbers. For some filed like house number, users should not be allowed to fill them before they fill street message.

5. Postscript

There are many dirty work during this project, I need to look all streets to come out `street_mapping`, also need to look at all cities to generate `city_mapping`, for some incorrect post code, I need to search online by address to find correct post codes. Also dataset for Shanghai consists of many Chinese, finally I find unicode would be good for this. Investigating data is a really time-consuming work and sometimes we go to wrong direction which would kill many of our times. Analyzing data also need to be very patient and do many dirty work.