# OpenStreetMap Sample Project

# Data Wrangling with MongoDB

*Wei Zhang*

Map Area: Shanghai, China
*https://mapzen.com/data/metro-extracts/metro/shanghai_china/*

References：
http://blog.csdn.net/qq_21970857/article/details/46532221

## 1.Problem Encountered in the Map

Using audit.py to the information of Shanghai, China with an output txt document containing problem data. I noticed five main problems, which I will discuss in the following order:

- Unicode Problem: \u767d\u6c99\u6cc9
- Chinese Translation: 南山路 89 号
- Over-abbreviated Street Names & Invalid Names: Aomen Rd, هوم لند یگ
- Postal Codes: “310200”,”2100080”,”22005”; House Numbers: 56 anxilu

### Unicode Problem

The Unicode string will not be displayed properly. So we should import sys for auditing. If using them, we should decode first before process

```
import sys
reload(sys)
sys.setdefaultencoding('utf-8')
```

## Chinese translation

In this part, I use pypinyin to solve this problem. In [address, name, user], each of them contains Chinese, so I translate them into pinyin and this will maintain most of their meanings.

```
def update_chinese(name):
    new_name = ""
    uni_str = name.decode('utf-8')
    pinyinlist = pypinyin.pinyin(uni_str,style=pypinyin.NORMAL)
    for i in pinyinlist:
        new_name = new_name + i[0]
    return new_name
```

## Over-abbreviated Street Names & Invalid Names

In this part, using Mapping will solve most of the problems. However, I found lots of information are contained in the string, so I search for the typical words including ‘road’,’lu’, etc. some other style is wrong, so return False.

```
mapping = {
            "Jie":"Street",
            "Rd":"Road","rd":"Road","Rd,":"Road","Rd.":"Road",
            "road":"Road","raod":"Road","Raod":"Road","Rode":"Road",
            "avenue":"Avenue","Ave.":"Avenue","Dadao":"Avenue",
            "Hwy.":"Highway",
            "lu":"Road","Lu":"Road",
            "Gonglu":"Road","Gong lu":"Road",
            "Xiang":"Alley"
            }
```

## Postal Codes & House Numbers

Postal Codes in Shanghai Region is ‘200000’ with 6 digits. If the capital number is 3 or the length is not 6, then it is Zhejiang region, return false. House Numbers should not contain Chinese or English, so remove then or return false

## 2.Data Overview

Using **json_extract.py** will output a standard JSON document. Using **mongodb_import.py** will import a shanghai_map dataset contains collections of places. In this part, I will do some statistics using Mongodb:
Shanghai_China.osm ......... 817.3 MB
Shanghai_China.osm.json .... 1.17 GB

Number of Documents:

```
db.places.find().count()
4337894
```

Number of Node and ways:

```
db.places.find({"type":"way"}).count()      468518
db.places.find({"type":"node"}).count()     3869062
```

Top 3 postcode list ( Because the postcode is not good enough for analyse, so the web should encourage people to enter in correct postcode or do not leave the entrance blank):

```
db.places.aggregate( {"$match":{"address.postcode":{"$exists":1}}},
                     {"$group":{"_id":"$address.postcode",
                                "count":{"$sum":1}}},
                      {"$sort":{"count":-1}},
                     {"$limit":10})

[{u'_id': u'214121', u'count': 28},
 {u'_id': u'215600', u'count': 21},
 {u'_id': u'212003', u'count': 14}]
```

## 3.Additional Ideas

Data Problems Analysis:

a) Type limitation suggestion:

From this message, Chen Jia is the most contributor. However, as we count before, the valid information for node and ways are 468518 and 3869062, so the contribution is big, but the accuracy is pretty low, or in other words, it is not English style rule for typing in. *If the problem solved, the data will be available for English-style analyst*

Top 3 Contribution list of users:

```
db.places.aggregate([{"$group":{"_id":"$created.user",
                                "count":{"$sum":1}}},
                     {"$sort":{"count":-1}},
                     {"$limit":10}])
```

```
[{u'_id': u'Chen Jia', u'count': 710158},
 {u'_id': u'Austin Zhu', u'count': 238380},
 {u'_id': u'aighes', u'count': 193391}]
```

So, In order to avoid the different between Chinese user and foreigners, rules should be clear for the users.
In ‘addr: street’ : type in characters only + involved given street type; all a new type just for shanghai, the cross between two roads
In ‘addr:postcode’: region code can be generated automatically if the street name is written in a good form.
After this method, the dataset will be much clear for foreigners to use not only for local data analyst, but for the global users. Although, the characteristic of Chinese road will lose, for data analyst, the method seems much more accurate rather than leave it.

b) More type of amenity are list:

Find Top 10 popular amenity type in shanghai:

```
db.places.aggregate([{"$match":{"amenity":{"$exists":1}}},
                     {"$group":{"_id":"$amenity",
                                "count":{"$sum":1}}},
                     {"$sort":{"count":-1}},
                     {"$limit":10}])
```

```
[{u'_id': u'bicycle_rental', u'count': 2619},
 {u'_id': u'parking', u'count': 1540},
 {u'_id': u'school', u'count': 1517},
 {u'_id': u'restaurant', u'count': 1281}, …
```

That is not surprise because the increasing of OFO, MOBIKE sharing bicycles. But I’m wondering whether we need so many bicycle_rental places then parking space or schools? Or the data is just that enough for the area. *So wish them to specify the name of the amenity as well unless with no name.* But this is OK for now, because user may only need the type of amenity.

## Conclusion

Shanghai_China.osm contains lots of information in Chinese style. As mentioned before in 3.a, rather than spend lots of time cleaning the data, I suggest OpenStreetMap to have rules for uses to type in the information of street. Rather than require extra time for cleaning optimizing in order to get a correct and perfect dataset because of the differences between Chinese and English, this method will improve the readability. So The future work should be set up rules for completing the information of the map.