OpenStreetMap Sample Project Data Wrangling with MongoDB

我选择的地图链接是： https://mapzen.com/data/metro-extracts/your-extracts/60feed1af51d

下载的具体 OSM 是地址是：
https://s3.amazonaws.com/mapzen.odes/ex_mRA8Tb7wXg7FR1aGCbfkLDeeDwgrJ.osm.bz2

1. 地图中遇到的问题

在最初下载上海地区的小样本并运行临时 data_explore.py 文件后，我注意到了数据的三个主要问题，我将按以下顺序讨论：

- 过分简化的街道名称 如 [Gongji Rd.(E.)]
- 中文路名里面有其他内容 如 [泰安路 120 弄]
- city 名混乱，有些 addr:city 是"上海",有些是"上海市"，有些是"浦东新区"
- 邮政编码错误，上海没有 20 开头的右边，如 [314211]

过分简化的街道名称(另外部分英文名称中也包含了街道名称)

我的方案是，对字符串进行处理，将一些简化的字符串还原为正常的。如 Gongji Rd.(E.) ---> Gongji Road(E.) Bao'an Hwy. ---> Bao'an Highway 另外部分 Node 和 Wat 英文名称中也包含了街道名称，这个也要对应地做调整

中文路名里面有其他内容

我的方案是，字符串做判断，发现在"路"后面，只要是数字，就直接截取"路"这个字及以前的部分

city 名混乱

整个数据集中，有 city 的不多，所以我打算再导入 Mongo 的时候，不考虑这个字段，绝大部分都说是"上海"，所以这个字段没有多大意义。

邮政编码错误

我的方案是，判断是不是 6 位数字，如果不是，直接过滤掉 如果是，再判断前 2 位是不是 20 开头的，如果不是，则过滤掉。

在此根据以上几点策略，将数据做基本的数据清理后，导入到 MongoDB

导入 MongoDB 的数据格式是：

```
"id": "2406124091",
"type: "node",
"visible":"true",
"created": {
          "version":"2",
          "changeset":"17206049",
          "timestamp":"2013-08-03T16:43:42Z",
          "user":"linuxUser16",
          "uid":"1219059"
        },
"pos": [41.9757030, -87.6921867],
"address": {
          "housenumber": "5157",
          "postcode": "201203",
          "street": "肇家浜路"
          "street:en": "Zhao Jia Bang Road"
        },
"amenity": "restaurant",
"cuisine": "mexican",
"name": "星巴克咖啡",
"name:en": "StarBucks Coffee"
"phone": "1 (773)-271-5176"
}
```

使用以下命令新建 mongodb 数据库

```
> use openstreetmap
switched to db openstreetmap
```

建立号以后，执行代码 data_wrangling.py 清洗并在 mongodb 中导入所有的数据

2. 数据概述

本节包含有关数据集和用于收集这些数据集的 MongoDB 查询的基本统计信息。

OSM 文件大小:
ex_shanghai.osm ..... 253 MB
ex_shanghai.osm.json ..... 386 MB

文档数

```
> db.arachnid.find().count()
1355351
```

节点(node)和道路(way)的数量

```
>  db.arachnid.find({"type":"node"}).count()
1189350
>  db.arachnid.find({"type":"way"}).count()
166001
```

创建数据的用户数(created.user)的数量

```
>  db.arachnid.distinct("created.user").length
1649
```

创建数据量最多的前 30 个用户是:

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

{ "_id" : "aighes", "count" : 121454 }
{ "_id" : "zzcolin", "count" : 82635 }
{ "_id" : "xiaotu", "count" : 81101 }
{ "_id" : "Koalberry", "count" : 73348 }
{ "_id" : "Xylem", "count" : 70182 }
{ "_id" : "duxxa", "count" : 67542 }
{ "_id" : "yangfl", "count" : 61903 }
{ "_id" : "alberth2", "count" : 45361 }
{ "_id" : "Austin Zhu", "count" : 44606 }
{ "_id" : "HWST", "count" : 41550 }
```

可以大致看出，贡献数据的人并没有出现 1 到 2 个贡献的了大部分数据的情况。 在此我代表 openstreetmap,对以上几位贡献数据最多的用户表示真心地感谢。

创建数据量最多的前十个用户创建的数据总共有：

```
> db.arachnid.aggregate(
    [{"$group":{"_id":"$created.user",
                "count":{"$sum":1}}},
    {"$sort":{"count":-1}},
    {"$limit":30},
    {"$group":{"_id":null,
               "total":{"$sum":"$count"}}}])

{ "_id" : null, "total" : 997873 }
```

可以看出，贡献数据最多的前 10 个人，贡献的数据是 997873，占总数 1355351 的 73.6%。 数据说明了少部分用户贡献了大部分数据。

只创建过 1 条到 10 条数据的用户有哪些:

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

{ "_id" : 1, "num_users" : 408 }
{ "_id" : 2, "num_users" : 152 }
{ "_id" : 3, "num_users" : 83 }
{ "_id" : 4, "num_users" : 45 }
{ "_id" : 5, "num_users" : 81 }
{ "_id" : 6, "num_users" : 41 }
{ "_id" : 7, "num_users" : 41 }
{ "_id" : 8, "num_users" : 27 }
{ "_id" : 9, "num_users" : 24 }
{ "_id" : 10, "num_users" : 20 }
```

从以上数据可以看出，很大部分用户只贡献了很少的数据。

只创建过 20 条数据一下的用户有哪些:

```
> db.arachnid.aggregate(
    [{"$group":{"_id":"$created.user",
                "count":{"$sum":1}}},
    {"$group":{"_id":"$count",
               "num_users":{"$sum":1}}},
    {"$sort":{"_id":1}},
    {"$limit":10},
```

```
    {"$group":{"_id":null,
               "total":{"$sum":"$num_users"}}}])
```

```
{ "_id" : null, "total" : 922 }
```

从数据可以看出，贡献了 20 条数据以及一下的用户数 922，总共贡献数据的用户数是 1649，占比 55.9。 也就是说超过了一半的用户，只贡献了很少的数据。

3. 其他发现

前面探索了基本信息，后面我们探索一下数据中还隐藏了哪些其他的信息。

前 10 名出现的设施

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

```
{ "_id" : "restaurant", "count" : 932 }
{ "_id" : "parking", "count" : 691 }
{ "_id" : "school", "count" : 578 }
{ "_id" : "bank", "count" : 384 }
{ "_id" : "cafe", "count" : 312 }
{ "_id" : "toilets", "count" : 299 }
{ "_id" : "fast_food", "count" : 240 }
{ "_id" : "bicycle_rental", "count" : 209 }
{ "_id" : "fuel", "count" : 155 }
{ "_id" : "hospital", "count" : 151 }
```

根据以上可以看出，最多的分别是 餐厅，停车场，学习，公园，还有咖啡厅

最多的宗教场所

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

```
{ "_id" : "christian", "count" : 29 }
{ "_id" : "buddhist", "count" : 20 }
{ "_id" : null, "count" : 18 }
{ "_id" : "muslim", "count" : 5 }
{ "_id" : "taoist", "count" : 3 }
{ "_id" : "confucian", "count" : 2 }
{ "_id" : "jewish", "count" : 1 }
```

说明上海的宗教场所并不多，最多也是 基督教 和 佛教。不过基督教的居然多余佛教，也是意外。

最受欢迎的美食

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

{ "_id" : null, "count" : 647 }
{ "_id" : "chinese", "count" : 99 }
{ "_id" : "japanese", "count" : 15 }
{ "_id" : "burger", "count" : 14 }
{ "_id" : "italian", "count" : 13 }
{ "_id" : "noodles", "count" : 10 }
{ "_id" : "asian", "count" : 9 }
{ "_id" : "american", "count" : 9 }
{ "_id" : "international", "count" : 7 }
{ "_id" : "pizza", "count" : 7 }
```

从数据可以看出，中国口味的餐厅占据大多数，除去中国口味的餐厅，最多就是日本菜，汉堡 和 意大利口味

具有高速公路点的数量

```
> db.arachnid.aggregate(
    [{"$match":{"highway":{"$exists":1}}},
    {"$group":{"_id":"$type",
               "count":{"$sum":1}}},
    {"$sort":{"count":-1}}])

{ "_id" : "way", "count" : 90279 }
{ "_id" : "node", "count" : 6842 }
```

具有高速公路属性的具体内容前 10 位分别是

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

{ "_id" : "way", "count" : 90279 }
{ "_id" : "node", "count" : 6842 }

{ "_id" : "service", "count" : 16790 }
{ "_id" : "residential", "count" : 13321 }
{ "_id" : "tertiary", "count" : 11205 }
{ "_id" : "unclassified", "count" : 10633 }
{ "_id" : "primary", "count" : 7867 }
{ "_id" : "secondary", "count" : 7025 }
{ "_id" : "footway", "count" : 5413 }
{ "_id" : "motorway", "count" : 4167 }
{ "_id" : "motorway_link", "count" : 3879 }
{ "_id" : "traffic_signals", "count" : 2575 }
```

说明有高速公路属性的节点，服务区，餐馆和第三产业是最多的

4. 建议与预期

建议：采用一些心里奖励机制让用户更多地贡献数据

从前面用户数的情况看，用户贡献并不是非常积极的。 贡献数据最多的前 10 个人，贡献的数据是 997873，占总数 1355351 的 73.6%。 再次建议 openstreetmap 可以采用一些心里奖励的机制来鼓励用户更多地贡献数据，例如搞个排行版，给些徽章等等方式。让贡献数据的人更容易让大家指导。

好处：

- 提高用户活跃程度，让已经贡献数据的用户更愿意贡献更多数据
- 让更多人来贡献数据

预期：

- 把数据量从 130w+的数据，至少提高一倍的数据量
- 让数据的字段更完善，如餐馆中大量数据是没有记录口味的

5. 总结

在对这些数据进行审查之后，上海地区的数据显然是不完整的，我认为上海的数据应该远远不止这些，但我相信已有的数据已经被清理干净了。

参考资料

OpenStreetMap: http://www.openstreetmap.org

Charlotte area Report https://s3.cn-north-1.amazonaws.com.cn/static-documents/nd002/SampleDataWranglingProject_en.pdf