Wrangle OpenStreetMapData with MongoDB

November 4, 2015

1. Problems Encountered in the Map

After initially downloading a small sample size of the Utrecht area and running it against a provisional data.py file, I noticed three main problems with the data, which I will discuss in the following order:

- "Incorrect" postal codes. Utrecht area zip codes all begin with "35" however a large portion of all documented zip codes were outside this region.

```
In [16]: from pymongo import MongoClient
         client = MongoClient("mongodb://localhost:27017")
         db = client.test
         top_pc = db.utrecht.aggregate([
                 {"$match":{"address.postcode":{"$exists":1}}},
                 {"$group":{"_id":"$address.postcode", "count":{"$sum":1}}},
                 {"$sort":{"count":-1}},
                 {"$limit":3 }])
         for pc in top_pc:
             print pc
```

```
{u'count': 508, u'_id': u'3706AA'}
{u'count': 352, u'_id': u'3621VC'}
{u'count': 294, u'_id': u'3513EW'}
```

```
In [17]: top_cities = db.utrecht.aggregate([
                 {"$match":{"address.city":{"$exists":1}}},
                 {"$group":{"_id":"$address.city", "count":{"$sum":1}}},
                 {"$sort":{"count":-1}},
                 {"$limit":3 }])
         for city in top_cities:
             print city
```

```
{u'count': 144886, u'_id': u'Utrecht'}
{u'count': 30927, u'_id': u'Nieuwegein'}
{u'count': 26400, u'_id': u'Zeist'}
```

```
In [16]:
```

```
Out[16]: 464132
```

2. Overview of the Data

Number of nodes

```
In [18]: db.utrecht.find().count()
```

```
Out[18]: 3582231
```

Number of nodes

```
In [19]: db.utrecht.find({"type":"node"}).count()
```

```
Out[19]: 3118099
```

```
In [ ]: Number of ways
```

```
In [20]: db.utrecht.find({"type":"way"}).count()
```

```
Out[20]: 464132
```

Unique users

```
In [21]: len(db.utrecht.distinct("created.user"))
```

```
Out[21]: 833
```

Top users

```
In [22]: top_users = db.utrecht.aggregate([
                {"$match":{"created.user":{"$exists":1}}},
                {"$group":{"_id":"$created.user", "count":{"$sum":1}}},
                {"$sort":{"count":-1}},
                {"$limit":3 }])
         for user in top_users:
             print user
```

```
{u'count': 720208, u'_id': u'Gertjan Idema_BAG'}
{u'count': 475096, u'_id': u'3dShapes'}
{u'count': 474095, u'_id': u'PeeWee32_BAG'}
```

Top amenities

```
In [23]: top_amenities = db.utrecht.aggregate([
                {"$match":{"amenity":{"$exists":1}}},
                {"$group":{"_id":"$amenity", "count":{"$sum":1}}},
                {"$sort":{"count":-1}},
                {"$limit":3 }])
         for amenity in top_amenities:
             print amenity
```

```
{u'count': 1353, u'_id': u'parking'}
{u'count': 949, u'_id': u'bench'}
{u'count': 403, u'_id': u'restaurant'}
```

```
In [ ]: 3. Other ideas about the datasets
```

```
In [25]: db.utrecht.aggregate([{"$match":{"amenity":{"$exists":1}}}, {"$group":{"_id":"$amenity",
         "count":{"$sum":1}}}, {"$sort":{"count":-1}}, {"$limit":10}])
```

```
Out[25]: <pymongo.command_cursor.CommandCursor at 0x7f673c882f90>
```

```
In [ ]:
```