OpenStreetMap Project

Sam Morrow

1. Problems Encountered in the Map

I initially downloaded a section of Northern Ireland, from openstreetmap.org, and opened it with an iPython Notebook. I loaded the output data into a local MongoDB instance, and began to have a look around. During my analysis I noticed the following:

- Incorrect & mixed case postcodes
- Missing space in postcode
- Letters in house number
- Lists in house number

1.1 The first query that lead me to discover anomalies in the postcode address field was:

```
# Find and instances of postcodes where they start with capital letter and then have one or more lowercase letters

> db.data.find({"address.postcode": { $regex: /^[A-Z][a-z]+/} })

excerpts:
      "postcode" : "Bt26 6fh"
      "postcode" : "BtT40"
```

I was looking for lowercase characters and I found plenty of entries with Bt, instead of BT. The most interesting anomaly I found with this query was entries with Btt, which is an Invalid Postcode in the UK, there can be a maximum of 2 letters before a number.

Next I went looking for postcodes to find any that were missing spaces:

```
# Check for postcodes that are all alphanumeric chars from start to finish

> db.data.find({"address.postcode": { $regex: /^\w+$/} })
```

```
excerpts:
      "postcode" : "BT222FF" # no space
      "postcode" : "BT40" # missing second half
```

My suspicions were confirmed, there were a number of postcodes missing spaces. This is not so much invalid, as inconsistent. It is normal to leave a space, as the two sections (normally) denote different things (i.e. area, street). In order to maintain consistency I opted for fixing missing spaces in my dataset.
Next I set out to find house numbers that contained letters, to see if I could make them all actual integers:

```
# Check for house numbers that contain alphabetical characters

> db.data.find({"address.housenumber": { $regex: /[a-zA-Z]/} })

excerpts:
      "housenumber" : "14a"
      "housenumber" : "FFC4"
```

I had entirely forgotten about flats, where the address is often identified with a letter (i.e. 6a) after looking up the documentation, it seems that there is not enough consensus for buildings with multiple house numbers, but for flats there is a key addr:flats which should be used for lists of flats

```
# Check for house numbers that contain space characters

> db.data.find({"address.housenumber": { $regex: /\s/} })

excerpts:
      "housenumber" : "12 & 14"
      "housenumber" : "205A & B"
      "housenumber" : "A4 A5"
      "housenumber" : "1B; 1C; 1D; 1E" # A list… hmmm
```

This is a complex issue, and there are letters allowed in house numbers in certain places (i.e. Russia). I think however, that it would most useful to split all flats out and list them in the addr:flats tag, and try to keep the house number field down to a single number, unless it is actually a range of numbers.

2. Data Overview

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

File sizes

```
BelfastAndSurrounding.osm ......... 124 MB
BelfastAndSurrounding.osm.json .... 150 MB
```

Number of documents

```
> db.data.find().count()
634198
```

Number of nodes

```
> db.data.find({"type":"node"}).count()
579090
```

Number of ways

```
> db.data.find({"type":"way"}).count()
55098
```

Number of unique users

```
> db.data.distinct("created.user").length
610
```

Top five most mentioned cities

```
> db.data.aggregate([
      {"$match":{"address.city":{"$exists":1}}},
      {"$group":{"_id":"$address.city", "count":{"$sum":1}}},
      {"$sort":{"count": -1}},
      {"$limit": 5}
])

[{ "_id" : "Belfast", "count" : 258 },
{ "_id" : "Crossgar", "count" : 29 },
{ "_id" : "Lurgan", "count" : 11 },
{ "_id" : "Larne", "count" : 10 },
{ "_id" : "Jordanstown", "count" : 7 }]
```

Top 1 contributing user

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

[ { "_id" : "Stephen_Co_Antrim", "count" : 207738 } ]
```

Most Common Postcode

```
{"_id":"$address.postcode", "count":{"$sum":1}}},
{"$sort":{"count":-1}}, {"$limit":1}])

[{ "_id" : "BT30 9PP", "count" : 32 }]
```

Number of users appearing only once (having 1 post)

```
> db.data.aggregate([{"$group":{"_id":"$created.user",
"count":{"$sum":1}}}, {"$group":{"_id":"$count",
"num_users":{"$sum":1}}}, {"$sort":{"_id":1}}, {"$limit":1}])
[ { "_id" : 1, "num_users" : 93 } ]
# “_id” represents postcount
```

3. Additional Ideas

3.1 Use Location to Verify Data

A suggestion I think would be interesting, is that Open Street Maps asks to use location data from your mobile phone whenever you are on the website, and can ask you to verify the details of the building you are in, when it detects you are in a building for which it has data, or to ask you if you would like to add the place you are if it has no information.

This might need to be an opt-in setting for logged in users, but enthusiasts might like the ability to help add / validate certain data on a casual basis. This would hopefully help manual map editors to make more contributions.

3.1 Additional Exploration of the Data

Most Popular Cuisine

```
db.data.aggregate([{"$match":{"cuisine": {"$ne": null},
"amenity":{"$exists":1}, "amenity":"restaurant"}},
{"$group":{"_id":"$cuisine", "count":{"$sum":1}}},
{"$sort":{"count":-1}}, {"$limit":3}])
```

```
[{ "_id" : "pizza", "count" : 7 },
{ "_id" : "italian", "count" : 7 },
{ "_id" : "regional", "count" : 6 }]
```

5 Most Common Amenity Types in Area

```
db.data.aggregate([{"$match":{"amenity":{"$exists":1}}},
{"$group":{"_id":"$amenity", "count":{"$sum": 1}}},
{"$sort":{"count":-1}}, {"$limit":5}])

[{ "_id" : "parking", "count" : 945 },
{ "_id" : "place_of_worship", "count" : 330 },
{ "_id" : "school", "count" : 181 },
{ "_id" : "fast_food", "count" : 141 },
{ "_id" : "restaurant", "count" : 121 }]
```

3 Most Common Take Away Restaurants

```
> db.data.aggregate([{"$match":{"cuisine": {"$ne": null},
"amenity":{"$exists":1}, "amenity":"fast_food"}},
{"$group":{"_id":"$cuisine", "count":{"$sum":1}}},
{"$sort":{"count":-1}}, {"$limit":3}])

[{ "_id" : "fish_and_chips", "count" : 26 },
{ "_id" : "burger", "count" : 19 },
{ "_id" : "chinese", "count" : 19 }]
```

3.3 Conclusion

During this investigation, I have seen large amounts of data that has not been correctly / consistently formatted, but that a lot of it can be fixed through automated parsing and correction. The bigger issue I feel is that there is clearly lots of missing data. My stats on the the most common cities mentioned shows that only Belfast has any significant number of addresses mentioned, and as it is a capital city, even 258 is a low number. There is clearly a lot of work to be done to complete the map, and probably other sources will need to be mined for additional data, if NI is to have a more complete map.

References:

- https://en.wikipedia.org/wiki/Postcodes_in_the_United_Kingdom#Formatting
- http://stackoverflow.com/questions/164979/uk-postcode-regex-comprehensive/17507615#17507615
- http://wiki.openstreetmap.org/wiki/Key:addr:flats
- http://wiki.openstreetmap.org/wiki/Addresses#Buildings_with_multiple_house_numbers