OpenStreetMap Sample Project Data Wrangling with MongoDB

Martín Pons Martínez

Map area: Palma de Mallorca and surroundings. Balearic Islands. Spain
http://www.openstreetmap.org/export#map=11/39.5369/2.6182

1. Problems encountered in the map

Encoding problems

While scanning the most frequent tags of the file one by one, with different helping functions, I first realized problems with the encoding, since Spanish and Catalan languages include some special characters ("´", "ç", etc). Data in the file is encoded as utf-8, while Python returns unicode encoding. Decode method was used to get an appropiate mapping. For example, the word 'camí' (way. Note the accent) is printed as 'cam\xc3\xad' in Python, while the same world is printed as 'cam\xed' when loaded from the xml file. This results in the mappings and the functions associathed with them, not working as expected. I solved this issue with the 'decode' method for string class, i.e "camí ".decode("utf-8")

Use of two official languages

Being The Balearic Islands a bilingual region, different users tag elements with one of the two official languages. For example, "Street", translates as "Calle" in Spanish and "Carrer" in Catalan. I decided to unify all street name types to Catalan. This can be seen specifically, in the 'name' tag and the 'addr:street' tag.

Lowercases

The word to distinguish the street type is placed at the beggining of the street name in Spanish and Catalan (as in 'Calle Niceto Alcalá Zamora': Niceto Alcalá Zamora Street or 'Plaza Pedro Garau': Pedro Garau Square). Therefore, this initial word, must begin with a capital letter. The problem is that some streets (squares, etc.) begin with a lowercase, instead of an uppercase. All the first initials were converted into uppercase.

Inverted house numbers

House number format can be a number, or a number followed by a letter. Apart from having again the some uppercase/lowercase mixture issues, the main problem here consisted in the inversion of some house numbers. Instead of having the number first, and then the letter, it was the other way around: first the letter and then the number. A helpler function, which inverted the string accordingly, was built to deal with this issue.

Ambiguous denomination for the city of Palma de Mallorca

The denominations ‘Palma’ and ‘Palma de Mallorca’ are used for both locals and foreigners to refer to the name of the city. Therefore, both denominations were encountered when inspecting the ‘addr:city’ field (apart from, as usual, the lowercase/uppercase issue). All city denomination instances in the ‘addr:city’ field, were set to ‘Palma’

Typos in postcodes

The postcode for the region of Balearic Islands is a 5 figure number beginning in ‘07’. A very few instances had typos, like, for example, the inversion of the first numbers of the sequences (‘70’).

All these problems were solved using proper mapping and helpler updating functions.

No address in most nodes

As I ran some queries in Mongo, I realised the lower number in diferent types of amenities than I expected. The problem is, there is no address tag associated for most nodes that should have one.

```
db.wrangling.find({"amenity": {"$exists":1}, "address":{"$exists":1}}).count()
```

473

```
db.wrangling.find({"amenity": {"$exists":1}, "address":{"$exists":0}}).count()
```

2982

2. Data overview

Basic statistics

- **File sizes**
  palma_y_alrededores: 62 Mb
  cleaned.json: 63 Mb

- **Number of documents**
  db.wrangling.find().count()
  665294

- **Numer of nodes**
  nodes_query = {"type":"node"}
  db.wrangling.find(nodes_query).count()
  593294

- **Number of ways**
  ways_query = {"type":"way"}
  db.wrangling.find(ways_query).count()
  71807

- **Number of unique users**
  len(db.wrangling.distinct("created.user"))

Additional statistics and exploration

- **Different types of restaurants**

```
db.wrangling.aggregate([{"$match":{"amenity":{"$regex":"restaurantcafe|bar|fast_food"}},
{"$group": {"_id":"$amenity", "count":{"$sum":1}}},
{"$sort":{"count":-1}}])
```

```
{u'_id': u'restaurant', u'count': 328}
{u'_id': u'cafe', u'count': 144}
{u'_id': u'bar', u'count': 137}
{u'_id': u'fast_food', u'count': 63}
```

- **Restaurants distribution by postcode**

```
db.wrangling.aggregate([{"$match":{"amenity":{"$regex":"restaurant|cafe|bar|fast_food"}
}}, {"$group":{"_id":"$address.postcode", "count":{"$sum":1}}},
{"$sort":{"count":-1}}])
```

```
{u'_id': None, u'count': 535}
{u'_id': u'07180', u'count': 52}
{u'_id': u'07160', u'count': 34}
{u'_id': u'07320', u'count': 24}
{u'_id': u'07181', u'count': 7}
{u'_id': u'07014', u'count': 7}
{u'_id': u'07002', u'count': 2}
{u'_id': u'07012', u'count': 2}
{u'_id': u'07003', u'count': 2}
{u'_id': u'07610', u'count': 1}
{u'_id': u'07006', u'count': 1}
{u'_id': u'07122', u'count': 1}
{u'_id': u'07140', u'count': 1}
{u'_id': u'07157', u'count': 1}
{u'_id': u'07015', u'count': 1}
{u'_id': u'07009', u'count': 1}
```

Here we have some consequences of not having complete information for most of the nodes: the postcode with most restaurants is ‘None’, and the next one is 07180, corresponding to the entire town of Santa Ponsa, smaller than Palma, but relatively important due to tourism.

- **How many schools and kindergardens are in the different cities belonging to the map area**

```
db.wrangling.aggregate([{"$match":{"amenity":{"$regex":"kindergarten|school"}}},
{"$group": {"_id": "$address.city",  "count":{"$sum":1}}},
{"$sort":{"count":-1}}])
```

```
{u'_id': None, u'count': 150}
{u'_id': u'Palma', u'count': 9}
{u'_id': u'Portals Nous', u'count': 1}
{u'_id': u'Peguera', u'count': 1}
{u'_id': u'Santa Maria del Cam\xed', u'count': 1}
```

Again, suffeing from the lack of information.

3. Additional ideas

Complement urban plan analysis with visualizations

Despite the City Hall may have public data resources at its disposal (data that can be used for urban planning) the internet provides additional resources like openstreetmap.

A simple analysis sometimes is enough: it gives useful information to the City officials. However, sometimes the patterns, the signals, are not so obvious, and a good visualization can be a great help. Thus, public officials can make use of the information available in openstreetmap to complement an analysis for urban planning.

Officials may extract data from the map at regular periods (annually, for example) and visualize it (postcode might be a good geographical reference). Then complement the city planning analysis (when not being the main tool for urban planning itself) analyzing this visualization and how it changes over time. More concretely, the visualization can be used for:

- **Anticipate problems, spot long-term trends and make useful questions**

Example: an examination of the map reveals that in a concrete district the number of restaurants and other amenities are decreasing, why? Is it all about the economic cycle or maybe there is something more, like an increasing in crime, for instance. Something that the City should be involved in? Is this affecting the surrounding districts in some way? Data available in openstreetmap can be used to answer this question.

(measure: variation of the relative number of restaurants, cafes, etc in the district)

- **Asses the outcome of a public plan**

Example: The public officials developed a subsidies plan to promote the creation of a new industrial conglomerate in a certain district. How it turned out? Has an industrial conglomerate actually been created? To what extent? Are there negative effects in other existing industrial conglomerates? If this is the case, what is the cause? Have been the closest industrial districts the most affected ones (geographical cause)? That is, it has not been an overall increase in the industrial structure of the city, but just a change in location.

(measure: relative number of industrial buildings in the district at one moment and overtime. Comparison with other industrial districts)

In order for this plan to be affordable and executable, an automation of the updates is mandatory (or private citizens, owners, have to have the right incentives, like publicity, to update the map) as well as a standard format for tag updating. An outdated map or not-normalized data can lead to misleading conclusions and, consequently wrong public plans.

4. Conclussions

The possibilities for data analysis of this kind of special data, might seem almost limitless, but in order to have some reliable data for even the most simple analysis, a minimum amout of data is necessary. The data extracted from this map is clearly incomplete, actually, most of it is incomplete. An effort has to be carried out to add more information to the map.