OpenStreetMap Sample Project
Data Wrangling with MongoDB

Laurent BRINGUIER – January 2015 Cohort

Map Area: Tokyo, Japan
https://www.openstreetmap.org/export#map=13/35.6175/139.6815

Chapter 0. Lost in Translation …

Sunday night… a clear warm spring Sunday night in Tokyo.
Tired of watching the sakura's flower dancing in the wind of this early spring night , I found a refuge in my favorite Jazz bar.
Park Hyatt Hotel, 52 floor , New york Bar unplugged session. The best answer to Sunday Blues …

As I was listening to the band I noticed a man, his nose in his whisky glass. He stared at me with eyes more melancholic than a dog expecting a caress after having peed in your slippers.

I recognized him without a doubt, it was BOB HARRIS, an aging American action movie star.

After a few minutes of hesitation , I came to him offering a second glass of Whisky and a partner to discuss.

Bob explained to me that he came to Tokyo to film an advertisement for Suntory whisky. The origin of his melancholic state was a woman named Charlotte.

He was leaving tomorrow and tonight was the only chance to see her again. He didn't know where she was, his only clues were a Map area she left on Openstreetmap and some whisky soaked memories of the address name and place where she was going tonight.

After seeing the picture of Charlotte on Bob's phone , I decided to help him find her !

The Map was in Japanese, Bob had no idea about how the Japanese address system was working and of course she had no Mobile phone…

Chapter 1. Where the streets have no name … *(Problems Encountered in the Map)*

Me : "Bob , it s your Lucky Day , I m a Udacity Data scientist student , I'll help you !!"
Bob : "A Uda.. what ? And you know something about Japanese address and Map ? "
Me : "Nevermind ! Well, let me briefly explain to you how it works : Firstly, in Japan, most of the streets have no name.."
Bob : "You must be kidding ! How can I find a building ?"
Me : "Well, it is not that complicated.. but not that simple too !"

The Japanese addressing system is based on areas, subdivided from big to small. There are different types of sub-divisions in different areas, but in the case of Tokyo, you will find :
1 - The prefecture : *to* (都), (capital), for Tokyo

2 - The City : *ku* (区) : Tokyo is divided into *ku* (区) generally called "cities" or "wards".
3 - The Town : *chou* (町). Wards are divided into *chou* (町) (though sometimes the name doesn't include the word *chou*).
4 - The District Number – block number-building number : Sometimes the *chou* are divided into *choume* (丁目), which are numbered divisions of a *chou*. Then the blocks are numbered and, at the lowest level, the building has a number.
We have two main problems with our Openstreetmap :

1 - The translation from Kanji to roman letter so that Bob can read the address and maybe recognize it.
2 - Be sure that the address makes sense.

After checking the data, we figured out that we can mainly focus on two fields:

- The "addr:full" field containing the entire address : <tag k="addr:full" v="東京都目黒区緑が丘 1-1-1"/>
- The "addr:postcode" field containing the postal code :<tag k="addr:postcode" v="152-0034"/>

Here is the strategy we are going to follow to clean the data and make it ready to use :

We have downloaded on the Japan post website an excel file containing all the Postal code with their Prefecture,City, town in English and Japanese Kanji.
We extract from that file only the prefecture and city potentially covered by the map and build a csv file which will be our source for mapping.

No	postal_code	town	city	prefecture	town_kanji	city_kanji	prefecture_kanji	full
36907	100-0001	Chiyoda	Chiyoda-ku	Tokyo	千代田	千代田区	東京都	東京都千代田区千代田
36908	100-0002	Kokyogaien	Chiyoda-ku	Tokyo	皇居外苑	千代田区	東京都	東京都千代田区皇居外苑
36909	100-0003	Hitotsubashi(1chome)	Chiyoda-ku	Tokyo	一ツ橋（１丁目	千代田区	東京都	東京都千代田区一ツ橋（１丁目）
36910	100-0004	(tsuginobiruonozoku), Ote-machi	Chiyoda-ku	Tokyo	大手町（次のビ	千代田区	東京都	東京都千代田区大手町（次のビルを除く）

For each "addr:full" in the map we will do the following steps :

For example with <tag k="addr:full" v="東京都千代田区千代田 1-1-1"/>

Check if it contains one of the "full" value of our source file.

No	postal_code	town	city	prefecture	town_kanji	city_kanji	prefecture_kanji	full
36907	100-0001	Chiyoda	Chiyoda-ku	Tokyo	千代田	千代田区	東京都	東京都千代田区千代田

If yes , add new fields :

- Prefecture_eng : with the name of the Prefecture => "Tokyo"
- City_eng : with the name of the ward or city => " Chyoda-ku"
- Town_eng : with the name of the town => "Chiyoda"
- Block : with the rest of the string => "1-1-1"
- Expected postal code => "100-0001"

We can also compare the Postal code from :<tag k="addr:postcode" v="100-0001"/> and the expected postal code field and check if they match or not.

We will separate the matching and non matching record and correct them afterwards if possible.

 # Number of node having a full address :

```
> db.MEGURO.find({"address.full" :{"$exists":1}}).count()
543
```

We can work on these nodes which have a full address. Note: only 543 on 402 289 nodes (only 0,135 % !!)

 # Number of node having a postal code AND a full address :

```
> db.MEGURO.find( { "$and": [ {"address.postcode" :{"$exists":1}}, {"address.full" :{"$exists":1}} ] } ).count()
430
```

For these specific nodes, we can check if the translation make sense by seeing if the expected postal code and the postal code matches.

```
> db.MEGURO.find( { "$and": [ {"address.Expected_Code" :{"$exists":1}} ] } ).count()
472
```

We could map and translate 472 address on 543 (mapping success ratio : 86,9 %)

Before Cleaning / Mapping	After Cleaning / Mapping
address.full: **u'東京都 港区 高輪 3-25-20'** address.postcode: u'108-0074'	address.full: **u'Tokyo Minato-ku Takanawa 3-25-20'** address.postcode: **u'108-0074'** address.Expected_Code: **u'108-0074'** address.Prefecture_eng: u'Tokyo' address.City_eng: u'Minato-ku' address.Town_eng: u'Takanawa' address.Block: u'3-25-20'

Main Cause of Errors when Mapping failed :

東京都 is missing at the beginning
Mistake between the city and town. Town not in the right city most of the time
Kanji Mistake

As we saw before , 430 records have an Address full and a postal code. After the cleaning we found that only 386 records have an expected Postal code.

```
> db.MEGURO.find( { "$and": [ {"address.postcode" :{"$exists":1}}, {"address.Expected_Code" :{"$exists":1}} ] } ).count()
386
```

On these 386 records only 359 records are found with an expected postal code and postal code matching, meaning that the name of the address and postal code make sense.

Cause of Errors when Postal Code Mapping failed :

Mismatch example Record	Explanation	Number of errors
Expected Postal code: 156-0055 Postal code : 1560055	Missing the “-“ , so format problem. We can check and correct this with a regular expression.	1 case
Expected Postal code: 156-0053 Postal code : 154-0015	Non matching value, can be human mistake in the postal code or Address. No way to automatically sort it out	27 cases

These results highlights different things :

1- A dramatically low number of nodes have an address
2- Postal code are most of the time pretty clean and make sense But we can see around 5 % of human error or mistake
3- Having one field to put the entire address can become very confusing and increase error rate. Splitting in different sub field like town, city can help the user not to make mistake.
4- The use of a mapping/lookup file to check the address and make translation is a great help to validate the data.

At the end of the process, we have :

543	Records with full Address
472	Records that we can map with the postal code reference file
386	Records that we can map with the postal code reference file and have a original postal code to check
359	Records that we can be confident make sense and show real address

Chapter 2 : A Huge City ...*(Data Overview)*

Me : "you know Bob, Tokyo is Huge and even this map covers a big area. Searching for Charlotte is going to take time…"

Bob, looking deeper and deeper in his glass of whisky : " You, French people are always exaggerating, can't be that big !"

Me : "Let me give you some numbers…"

The files size are :

```
Meguro.osm ..... 94 MB
Meguro.osm.json ..... 105 MB
```

The number of documents :

```
>db.MEGURO.find().count()
478257
```

The number of nodes in the map :

```
> db.MEGURO.find({"type":"node"}).count()
402289
```

Number of ways

```
> db.MEGURO.find({"type":"way"}).count()
75957
```

Number of unique users

```
> len(db.MEGURO.distinct("created.user"))
497
```

Top 1 contributing user

```
db.MEGURO.aggregate([{"$group":{"_id":"$created.user", "count":{"$sum":1}}}, {"$sort":{"count":-1}}, {"$limit":1}])
 [ {u'ok': 1.0, u'result': [{u'_id': u'kurauchi', u'count': 68349}]}
```

Number of users appearing only once (having 1 post)

```
db.MEGURO.aggregate([{"$group":{"_id":"$created.user", "count":{"$sum":1}}}, {"$group":{"_id":"$count",
"num_users":{"$sum":1}}}, {"$sort":{"_id":1}}, {"$limit":1}])
{u'ok': 1.0, u'result': [{u'_id': 1, u'num_users': 99}]}
# "_id" represents postcount
```

Chapter 3 : Looking for Charlotte… *(Additional Ideas)*

Me, thinking to myself : " Is this MAP really reliable ? Where do these data come from ? I must find out.. "

Source statistics and sources suggestion:

Comparing to the total number of nodes, the number of nodes with a full address field is dramatically low.
If we have a look at the sources of address information for the nodes, here is what we find:

```
db.MEGURO.aggregate([{'$match': {"address.full":{'$exists': 1}}},{"$group":{"_id":"$source", "count":{"$sum":1}}},
{"$sort":{"count":-1}}, {"$limit":10}])

{u'ok': 1.0,
 u'result': [{u'_id': u'Bing 2010; (URL)', u'count': 137},
  {u'_id': u'KSJ2; (URL) 2014', u'count': 72},
  {u'_id': u'(URL)', u'count': 59},
  {u'_id': None, u'count': 52},
  {u'_id': u'KSJ2; Bing 2010; (URL) 2014', u'count': 40},
  {u'_id': u'KSJ2; (URL) 2013', u'count': 32},
  {u'_id': u'Bing 2010; (URL) 2012', u'count': 27},
  {u'_id': u'KSJ2', u'count': 19},
  {u'_id': u'survey', u'count': 15},
  {u'_id': u'(URL) 2012', u'count': 8}]}
```

These results show us some interesting facts:

- Around 85 % (461 on 543) of the records having an address comes from the TOP 10 sources.
- The rules of standardization for the source field are not really respected. (sometimes we find the year, sometimes the URL, or None value, etc..)
- In our Top 10 sources: some data are pretty old

To dig deeper, we had a look at all the sources together, and here is what we found:

- Only 33 % of the data were acquired between 2015 and 2013 (so last 2,5 years) , 35,5 % are older than 2,5 years (25% is 5 years old !!) and 31,5% has no date. When you know how fast Tokyo construction moves, you can tell that these data are very too old to be accurate.
- BING and KSJ2 are the 2 main sources (68%) of the data.

Source \ Year	2015	2014	2013	2012	2011	2010	2009	2008	2007	Unknown	Total per source	% per source
Bing		2		29		139			16	7	193	35,54%
KSJ2		118	35							27	180	33,15%
URL			1	8						59	68	12,52%
None										52	52	9,58%
Survey	8	9	7							21	45	8,29%
Knowledge										4	4	0,74%
Pushpin for iOS										1	1	0,18%
Total per year	8	129	43	37	0	139	0	0	16	171	543	
% per year	1,47%	23,76%	7,92%	6,81%	0,00%	25,60%	0,00%	0,00%	2,95%	31,49%		

- **BING :** Bing Aerial Imagery analysis
- **KSJ2 : "National-Land Numerical Information based on JPGIS"** is geo data provided by National Land Information Division, a section of *National and Regional Policy Bureau(NRPB), Ministry of Land, Infrastructure, Transport and Tourism (MLIT), Japan*

These 2 sources seems reliable, but having a bigger number of sources will help to cross validate data, and cover more nodes.
In order to be reliable the data should be updated on a regular basis and a special field should tag the date of update. More than 2,5 years is not acceptable for a Map in a very active urban area.

Recently, the Japanese government is promoting the Open Data initiative, in which the government widely discloses public data in machine-readable formats and allows secondary use of the public data for profit-making or other purposes. This initiative has the goals of improving people's lives and stimulating corporate activities, thereby contributing to social and economic development of Japan.

The following site provides a catalog of Data from different institution of the government (including the ministry of Land or economy or education …):

http://www.data.go.jp/?lang=english

It would be interesting to search in this data Catalog which dataset can help improve the openstreetmap for Tokyo.

Looking for Charlotte… (Additional data exploration using MongoDB queries)

Me : "Do you have any idea where to start ? what did she say ?"
Bob : "MMmmm…. She was joining friend to drink or party or something like that .. "
Me : "Ok let's see the Amenities info !"

Let's have a look at the Top 100 appearing amenities , and see where we can have fun !!

```
db.MEGURO.aggregate([{"$match":{"amenity":{"$exists":1}}}, {"$group":{"_id":"$amenity","count":{"$sum":1}}},
{"$sort":{"count":-1}}, {"$limit":100}])

{u'_id': u'pub', u'count': 223},
{u'_id': u'bar', u'count': 75},
{u'_id': u'nightclub', u'count': 6},
{u'_id': u'biergarten', u'count': 4},
{u'_id': u'casino', u'count': 3}
,{u'_id': u'karaoke_bok', u'count': 3},
{u'_id': u'karaoke_box', u'count': 2},
{u'_id': u'karaoke', u'count': 2},
```

Me : "Karaoke Maybe ? or Bar, Pub ?"
Bob : "Now I remember, it was a Café , yes a Café .. "
Me : "Ok let's see where are the café ! Any town sounds familiar ?"

```
db.MEGURO.aggregate([{'$match': {"amenity":{'$eq': u'cafe'}}},{'$match': {"address.Town_eng":{'$exists':
1}}},{"$project":{ "_id": 0, "address.Town_eng" : 1,"address.City_eng" : 1,}}])


[{u'address': {u'City_eng': u'Meguro-ku',  u'Town_eng': u'Ookayama'}},
 {u'address': {u'City_eng': u'Shinagawa-ku', u'Town_eng': u'Higashioi'}},
 {u'address': {u'City_eng': u'Ota-ku', u'Town_eng': u'Omorikita'}},
 {u'address': {u'City_eng': u'Setagaya-ku', u'Town_eng': u'Gotokuji'}},
 {u'address': {u'City_eng': u'Setagaya-ku', u'Town_eng': u'Miyasaka'}},
 {u'address': {u'City_eng': u'Setagaya-ku', u'Town_eng': u'Sangenjiya'}}]}
```

Bob : “Yes !! ah Yes !! Ookayama !! that was Ookayama !! .. ”
Me : “Rigth ! let’s check the info !”

```
db.MEGURO.aggregate([{'$match': {"amenity":{'$eq': u'cafe'}}},{'$match': {"address.Town_eng":{'$eq': u'Ookayama'}}},{"$project":{ "_id": 0, "address.Town_eng" : 1,"address.full" : 1,}}])

{u'ok': 1.0, u'result': [{u'address': {u'Town_eng': u'Ookayama',    u'full': u'Tokyo Meguro-ku Ookayama 2-12-1'}}]}
```

Me : “BINGO ! Here you are : Tokyo Meguro-ku Ookayama 2-12-1!!”

Bob : “Oh my god ! You did it !! ”

Me : “ Go! Go ! run ! Get a Taxi and give him the Address !”

Bob : “You … You and your Uda..Sciences .. You are marvelous !! ”

As the Band was playing:
“ I want to run, I want to hide, I want to tear down the walls, That hold me inside ,
I want to reach out, And touch the flame, Where the streets have no name..”

He left… running with a big smile ...forgetting to pay his 5 glasses of Whisky …

Conclusion :

Creating and updating Map is not an easy job… especially in Japan. As we saw, many obstacle exists :
The language, the address system, the reliability of source and the number of different sources.
But if we really want to find a way to succeed in this kind of project, then taking time to know the culture, study the language, find new data sources and walk around the city is mandatory … And if you can also have fun and enjoy your time than it sounds like a good recipe for Happiness.
Cleaning and wrangling Data is a key to create great Map and be able to find your way or to be found.
Because …

Everyone wants to be found…

Sources

- **Japan Post (2015)**
  [online] Available at: http://www.post.japanpost.jp/english/index.html [Accessed 10 Apr. 2015].
- **How does the Japanese addressing system work? (2015).**
  Available at: http://www.sljfaq.org/afaq/addresses.html [Accessed 20 Apr. 2015].
- **Japanese addresses: No street names. Block numbers. (2015). [Blog] *Derek Sivers*.**
  Available at: https://sivers.org/jadr [Accessed 20 Apr. 2015].
- **Wikipedia, (2015). Lost_in_Translation**
  [online] Available at: http://en.wikipedia.org/wiki/Lost_in_Translation_(film) [Accessed 30 Apr. 2015].
- **DATA GO JP. (2015).**
  Available at: http://www.data.go.jp/?lang=english [Accessed 20 Apr. 2015].

All characters appearing in this work are fictitious. Any resemblance to real persons, living or dead, is purely coincidental. ☺

About the Map and Me :

I m French and leaving in Tokyo since 2013. My Japanese level can still be considered as a beginner and I had many hard times to find places where I needed to go within Tokyo. The area of Tokyo that I chose includes my house.
I decided to write this project as a story related to Lost in translation in order to make it more fun to read and first of all because I enjoyed it !
Hope it made you smile.
Laurent.

https://www.openstreetmap.org/export#map=13/35.6175/139.6815