**OpenStreetMap Project**

Content

[1. Map area 2](#_Toc501553752)

[2. Issues in the map 3](#_Toc501553753)

[3. Data Overview 4](#_Toc501553754)

[4. Conclusion 6](#_Toc501553755)

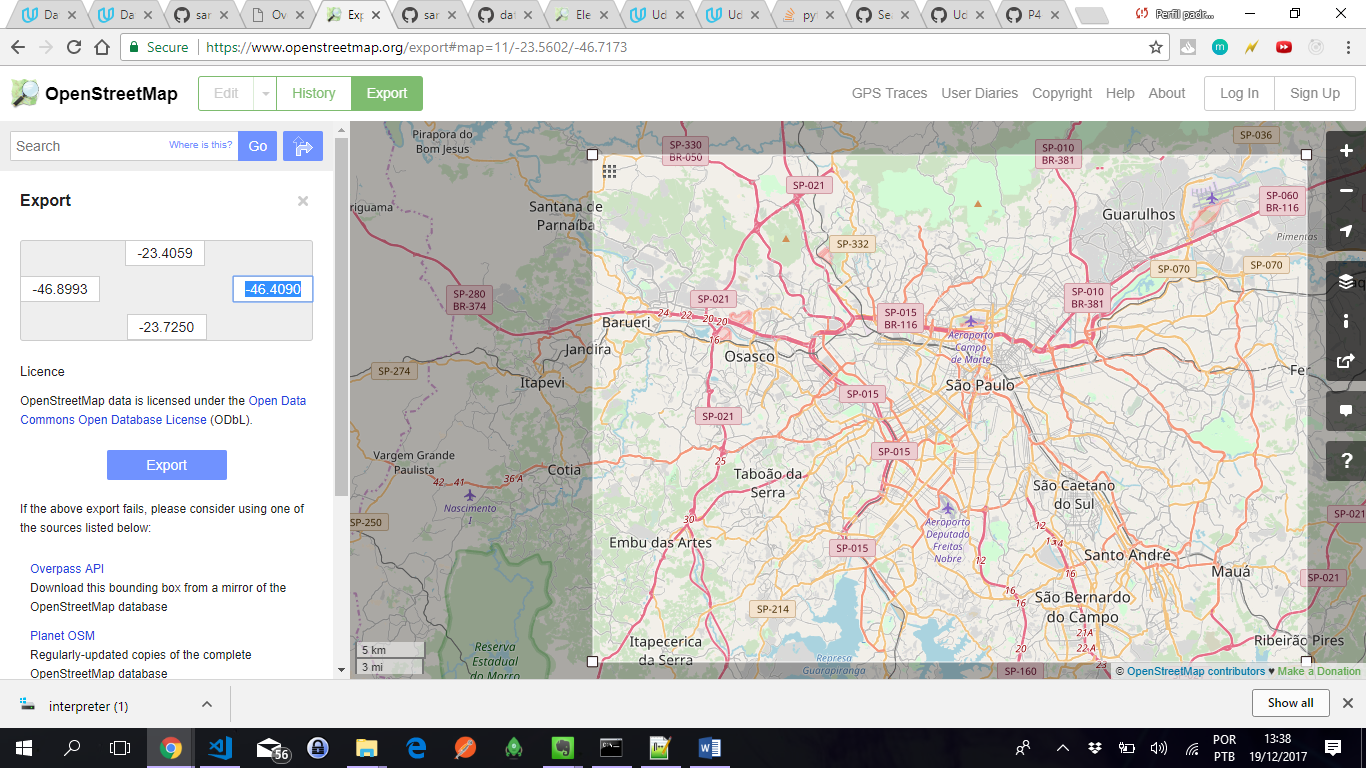
[5. References 6](#_Toc501553756)

# Map area

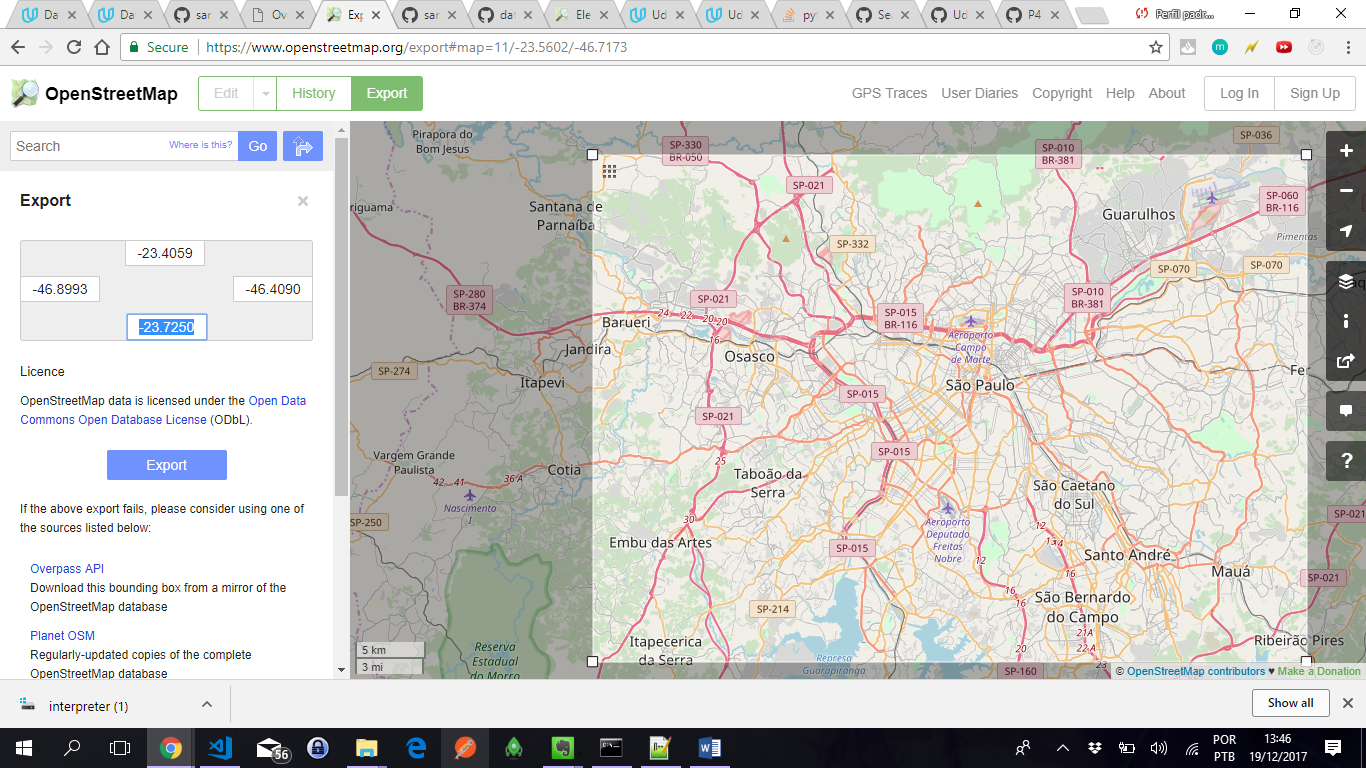
São Paulo, SP, Brazil.

São Paulo central area was the sample for this analyze and its coordinates is minimum latitude: -23.5595, minimum longitude: -46.6534, maximum latitude: -23.5375, maximun longitude: -46.6202

Size file:  39293 KB.



São Paulo complete map has this coordinates: minimum latitude: -23.725, minimum longitude: -46.8993, maximum latitude: -23.4059, maximum longitude: -46.4090.



The city where I live does not have enough content to analyze, and then, I decide for São Paulo because there were stores enough for a good understanding, and analyzing.

# Issues in the map

This section contain information about the issues found during the analyze.

Based on the sample file, it was found these issues:

* Overabbreviated and misspelled name:

The map has 46923 streets and 98 errors, less than 1% of errors. Some of them was abbreviation error, and it was fixed mapping the errors and their correct value.

MAPSTREET = {

"Av":"Avenida",

"Estr":"Estrada",

"Pç":"Praça",

"Al":"Alameda",

"Alamedas":"Alameda",

"Rue":"Rua",

"Rúa":"Rua",

"R":"Rua"

}

After the fixes, the errors felt down to 65.

* Postal code inconsistence

The audit method returned postcode did not follow the Brazilian pattern: 5 numbers plus dash plus 3 numbers

Some of the postcode returned for the audit method as error was possible to fix. The number of digits was correct but without the dash. For cases like these, it was developed a method to fix it. Follow an example for this issue:

  <node id="3063117943" lat="-23.5568747" lon="-46.6498269" version="1" timestamp="2014-09-06T14:07:36Z" changeset="25267529" uid="2315838" user="Ana Paula1608">

    ...

    <tag k="addr:postcode" v="01330000"/>

    ...

  </node>

Follow the routine to fix these cases. When it returns None, the value is not insert in the csv file.

if is\_postalcode(kvalue):

if auditory\_postalcode(value) == False:

value = fix\_postalcode(value)

if value == None:

continue

def fix\_postalcode(value):

"""Return the postal code when it is possible to fix or None when it is not"""

if len(value) == 8:

nvalue = value[0:5] +'-'+value[5:]

# verify if the new code is valid

if auditory\_postalcode(nvalue):

return nvalue

else:

return None

else:

return None

# Data Overview

This section contains information about the files, basic statistics about the dataset and queries used to gather them.

File size:

sample.osm: 392MB

SaoPaulo.osm: 849MB

Nodes.csv 314MB

nodes\_tags 11MB

Ways.csv 32MB

ways\_nodes.csv 122MB

ways\_tags.csv 49MB

Number of nodes select count(\*) from nodes; 3653282

Number of ways select count(\*) from ways; 529372

Number of unique users: 1865

select count(distinct(uid)) from (select uid from nodes union select uid usway from ways);

Number of traffic\_signals: 6618

select (count(\*) + qtd) As qtdtrafficsignals from nodes nd, (select count(\*) as qtd from ways wy inner join ways\_tags wt on wy.id = wt.id where wt.VALUE = 'traffic\_signals') inner join nodes\_tags nt on nd.id = nt.id where nt.VALUE = 'traffic\_signals';

Number of positions you can cycle: 2591

select (count(\*) + qtd) As qtdcicleplace from nodes nd, (select count(\*) as qtd from ways wy inner join ways\_tags wt on wy.id = wt.id where wt.value = 'yes' and wt.key='cicle') inner join nodes\_tags nt on nd.id = nt.id where nt.VALUE = 'yes' and nt.key='cicle';

Number of nodes with tags: 131783

SELECT COUNT(DISTINCT(ID)) FROM NODES\_TAGS;

Most 5 popular cuisine

SELECT tags.value AS CUISINE, COUNT(\*) as count FROM (SELECT \* FROM nodes\_tags where nodes\_tags.key='cuisine' UNION ALL SELECT \* FROM ways\_tags where ways\_tags.key='cuisine') tags WHERE tags.key='cuisine' GROUP BY tags.value ORDER BY count DESC limit 5;

CUISINE | count

regional|254

burger|172

pizza|136

japanese|62

sandwich|56

The top 10 contributors

SELECT USER,COUNT(\*) FROM NODES GROUP BY USER ORDER BY COUNT(\*) DESC LIMIT 10

Bonix-Mapper|2454573

Bonix-Importer|434301

AjBelnuovo|135497

cxs|73426

O Fim|72681

ygorre|46031

MCPicoli|33776

naoliv|28221

D4RK-L3G10N|28119

patodiez|24726

# Conclusion

There are 3653282 nodes in the XML file, and there are 131783 nodes with tags, which represents 3.60% of nodes with tags.

There are 1865 different users contributed to the openstreetmap. The same user Bonix-Mapper inserted approximately 67% (2454573) of these insertions.

I consider this map is incomplete due to the number of tags added to the map. São Paulo is a city with a huge quantity of restaurants, bars, streets, etc. I think the number of tags must be higher than 3.60% and to increase the number more people should contribute to this map.

# References

<https://github.com/SamMorrowDrums/Udacity-OpenStreetmap/blob/master/submission.pdf>

<https://github.com/ollkorrect/P4-Streetmap-Data-Wrangling/blob/master/Report.pdf>

<https://gist.github.com/carlward/54ec1c91b62a5f911c42#file-sample_project-md>