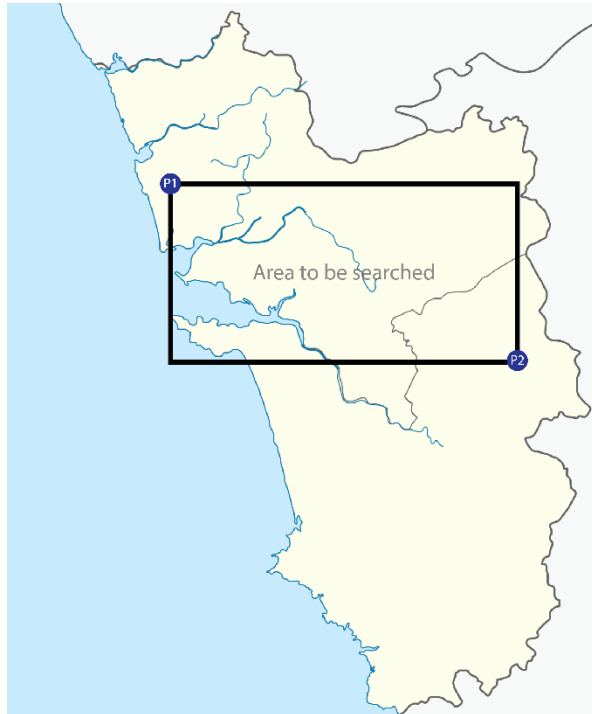


Insert the two opposite coordinates of a rectangular area into points.csv as

P1	Latitude(P1)	Longitude(P1)
P2	Latitude(P2)	Longitude(P2)

Point P1 and P2 are as shown below.

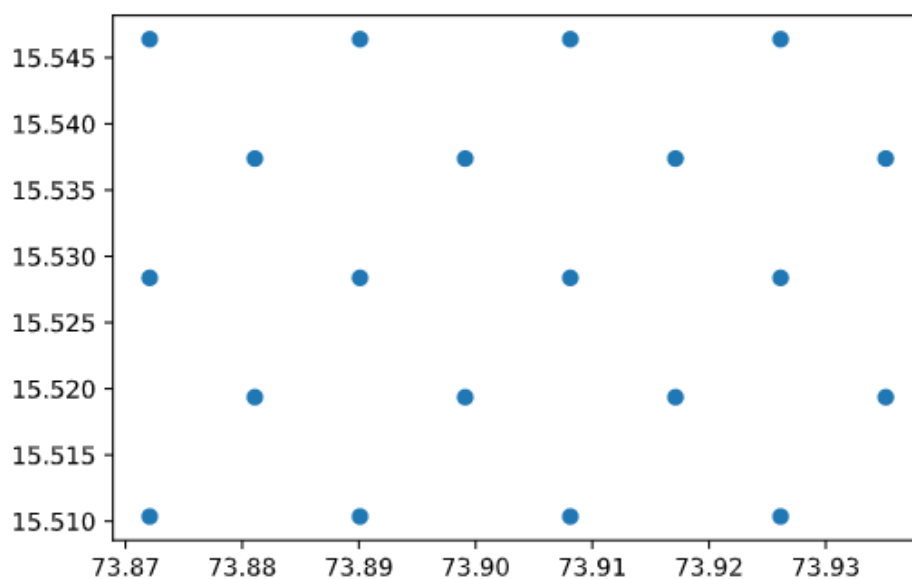


\*image isn't scaled

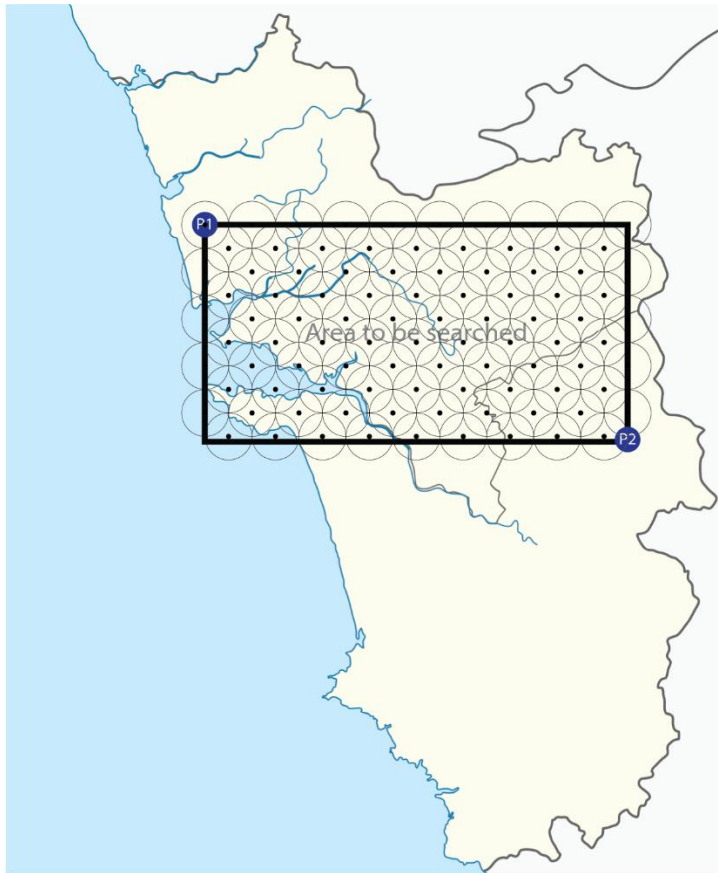
It seems that you cannot get the data properly beyond the radius of approximately 1 km.

I am going to stick with the data from a region enclosed by a circle of radius 1 km around it's center.

So first, I will generate multiple circles using *map\_points* situated in *map\_splitter.py* such that the complete area is covered.



So, I am trying to capture the data by making repetitive calls to the API, such that it can cover the whole area as following



\*image isn't scaled

Thus, covering the complete area.

As you can see a small area requires a lot of map points and so I don't recommend doing this over a larger area, as it will make a lot of API calls this increasing the traffic as well as making the whole process a bit slow.

The entity points are stored in the '**entity types.csv**' file. By default it will read all the points of interests and their entity codes and extract the information.

It stores all the json files into the '**json files**' directory according to entity type and it will save all the output data to the '**output**' directory with filenames as '**[entity type].txt**'

The file '**output/1.count**' stores the counts of all the Entities found