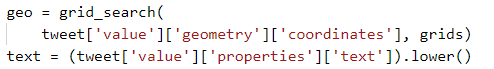
I will briefly explain this project from the main part of the program, how to implement it, how to parallelize it, and performance analysis.

1. Main part of the Program:
2. Accept the Twitter file: Use json.load() to read as python dict.
3. Extract the coordinates and text parts: 

Calculate area information from Latitude and longitude, also convert text to lowercase.

1. Calculate the score according to the text:



First, we calculate the scores of all the phrases and delete the phrases from the text (because the phrase may contain words), then calculate the scores of all the words and add them to get the result.

1. Add it to the corresponding area:

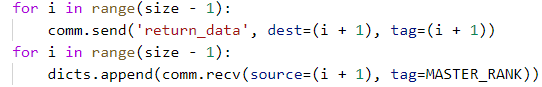
In step b, we have obtained the area information, now we add the text score to the corresponding area and add one to the total number of regional tweets.

1. Parallelize step:

We use MPI (message passing interface) to exchanging message between multiple nodes across distributed memory.

We will assign ranks to parallel programs on different cores according to the preset number of cores.

Different cores will determine the data they need to process according to their unique rank and total core number (the data is not duplicated with other cores) Only processed when the text number corresponds to the rank.

Only the first rank of them will be the master, it will ask other cores called slaves to return the result to it and summarizing them,so our program can be parallel.