Task 3 Extra Credit Name: Xixuan Wang (xw2579)

Statistics about the results of original task 1:
(After running 10 times)

CPU time spent (ms)=116520
CPU time spent (ms)=102610
CPU time spent (ms)=91040
CPU time spent (ms)=70520
CPU time spent (ms)=88310
CPU time spent (ms)=64090
CPU time spent (ms)=103410
CPU time spent (ms)=112990
CPU time spent (ms)=90630
CPU time spent (ms)=87490

The strategy I tried to optimize my map reduce program is to use the map-side join instead of reduce-side join. The reason is that reduce-side join is relatively less efficient because there's a lot of overhead of disk operations in the middle. By using the map-side join, such kind of overhead can be eliminated.

However, after trying this, I found the strategy doesn't fit to task 1 because there are two separate input files, and there could be different working nodes that are specified by the HDFS configuration. In other words, each "map.py" is not able to collect both data sets, which leads to failure of joining the two tables of data.

Although the strategy doesn't work for this assignment, it indeed works well if one "map.py" is able to handle with two data sets at the same time within the same memory space.

Therefore, to make it work for this assignment, from my point of view, there are two ways. First, assemble the two data sets to be only one. Second, change the relevant running configuration to enable two separate data sets to be fed into one mapper. There might be some other ways as well, but the overall idea is the same: make the map-side join successful.