Implementation of Page Rank Algorithm with Embeddings for Wikipedia

TASK 1:

Convert an input file to an adjacency list using Map Reduce

Mapper 1:

For every line in the standard input, the following operations are performed:

1) strip, split functions used to isolate source and destination nodes

2) Convert them to integer value and handle Value Error in case of one

3)Output using print in the following format: source_node, dest_node

In-built Modules Used: sys

Reducer 1:

1) Read source and destination nodes from standard input

2) Keep track of current source node using the variable cur_src

3) A list maintains all destination nodes for the cur_src and is updated for every new source

4)This cur_src and list are separated by the delimiter $ while printing

5)Every cur_src is written to the v file in the format cur_src,1

In-built Modules used: sys

TASK 2:

Iteratively calculate and update page ranks until convergence

Mapper 2:

1) v file and page-embeddings file are loaded into memory

2) Source node and intermediate rank are read from the v file and are added/updated in a dictionary named rank

3) Source node and its corresponding destination nodes are read from standard input

4) For each pair of source and destination, similarity and contribution (partial contribution ideally) are calculated according to the given formulae by calling the user-defined functions cosine_similarity() and contribution()

5) Destination node, corresponding source node, corresponding individual/partial contribution are printed in the respective order

6) If any node does not act as the destination in the given network, it is found using the rank dictionary and this node, 0, 0 are printed

In-built modules used: sys, json

Reducer 2:

1) Destination node, source node and contribution are read in the same order from standard input

2) A variable named cur_dest is used to keep track of the current destination node in question

3) All the partial contributions are summed up for a given destination node

4) Final rank is calculated as (0.15 + 0.85*sum_of contributions)

5) The current destination, cur_dest, and its final rank with a precision of 2 decimal points are printed

6) This process is repeated for every destination node

In-built modules used: sys