1-to-N Strongest Path Problem

January 18, 2016

1 Problem Description

Let $G(V, E)$ be an undirected graph where V is a set of nodes and E a set of edges. Each edge $e \in E$ is associated with a weight w, $0 < w < 1$. We define the *strength* of a path $(n_1, n_2, \ldots, n_k)$ as $w_1 \times w2 \times \ldots \times w_{k-1}$, where w_i is the weight of the edge between n_i and n_{i+1}.

Given a node n (called source node) and a set of nodes $\{n_1, \ldots, n_k\}$ (called target nodes), you are required to write a program to find the strongest path (a path with the largest strength) from the source node to each of the target nodes.

2 Inputs and output

Your program should take three parameters:

Graph File. A text file that contains the data for the graph G. Each line of the file corresponds to an edge, with the format

```
node1 node2 weight
```

`node1` and `node2` are the names (of type string) of the nodes. For example, one line may be
`Alice Bob 0.667`

Source and Target File. A text file that lists names of the source node and the target nodes. The format of the file is as follows:

```
source_node

target_node1
target_node2
...
target_nodek
```

That is, the first line is the name of source node, the second line is empty,

and each of the remaining lines is the name of a target node. Here, we assume that the number of target nodes is much smaller than the total number of nodes in the graph.

Output File. A text file that contains the result of your program. Each line corresponds to the strongest path from the source node to a target node, with the following format:

```
source_node target_node_1 path_weight:  source_node w1 node1 w2
node2 ...wk target_node
```

Here `path_weight` is the weight of the strongest path from the source node to the target node. `w1` is the edge weight between the source node and `node1`, `w2` is the edge weight between `node1` and `node2`, etc. If there are multiple strongest paths between the source and a target, just output one of them.

In summary, to run your program, it would be like:

```
your_program graph.txt source_target.txt output.txt
```

3 Submission and Evaluation Criteria

A graph can be very large, with tens of thousands of nodes and millions of edges. Your program will be evaluated not only on its correctness but also on its performance. You may choose the programming language, design your own data structures, and assume that your program will be run in a multi-core system. Further, please make sure that: (1) your code is appropriately commented, as we will also evaluate the clarity of your code; and (2) you are using defensive programming so that any unexpected error (for example input formatting errors) is handled.

When submitting your program, please also include a `README.txt` file with detailed instructions regarding system requirements, compiling (if needed) and execution.