

Homework 1

Chunwei Yan

2012 年 10 月 26 日

1 大体思路

原题中限制了直接获取全局图的能力，而是只单次提供每一个节点及其 neighbour 的信息。如此，想到从微观角度进行计算。即，每次只计算一个节点的 pagerank 值，并且不断在磁盘中进行覆盖重写。全局利用广度优先的方式在 neighbour 中遍历开来。当然，也需要考虑 BFS 对孤立集合的问题。每次 BFS 终止时，随机从未计算的节点里面选取一个节点，继续计算用 BFS 进行计算。初始时，同样为每一个节点分配 $\frac{1}{N}$ 的初始值，然后按照以下公式计算

$$P_i = k \sum_{P_j \text{ Neighbours}} \frac{P_j}{L_j} + (1 - k) \frac{1}{N}$$

2 算法解释

就像一滴水落到水面，刚开始只是一点有很大的起伏，之后水面会溅起涟漪，不停震荡，最终又会恢复到平面。那多余的一滴水在不停震荡之后，自然会平稳地分布到平面。用初始时从一点（可能也可以从多点）计算来模拟一滴水掉到水平面，改进的 BFS 来模拟涟漪的扩展，用不停迭代代表震荡，而最终平静之后，就是各点的 pagerank 值的稳定。我们尝试用不停的微观的微调来模拟宏观的调整。

3 一些注意点

1. 注意磁盘与内存中内容的同步
2. BFS 终止时，需要从此轮未计算过的节点中选取一个，继续 BFS
3. 每一次计算的节点，都需要是此轮未计算过的节点，或许在磁盘中需要保持一个标记