**电子科技大学计算机科学与工程学院**

**标 准 实 验 报 告**

**（实验）课程名称 数据结构与算法**

**电子科技大学教务处制表**

**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名：陶浩轩 学 号：2023080902011 指导教师：陈端兵**

1. **实验室名称：**

学知三组团 110139

**二、实验项目名称：**

重要节点组挖掘

**三、实验原理：**

本实验基于复杂网络理论，利用网络中节点的度以及节点之间的连接关系，寻找对信息传播影响最大的节点组。主要原理如下：

* **网络模型构建**: 将用户之间的关系抽象为网络，每个用户作为一个节点，用户之间的联系作为边。
* **节点重要性评估**:
  + **度排序**: 节点的度越大，其连接的节点越多，对信息传播的影响力也越大。
  + **基于投票策略**: 节点根据其邻居节点的投票能力进行排序，投票能力高的节点更有可能被选中作为种子节点，从而启动信息传播。本实验中，投票能力采用 PageRank 算法计算，PageRank 算法通过分析节点之间的链接关系，评估节点在网络中的重要性。
* **信息传播模型**: 采用SIR模型模拟信息在网络中的传播过程，评估不同节点组作为种子节点时的信息传播效果。

**四、实验目的：**

* 探索在有限资源下，如何选择一组节点进行营销推广，以达到最佳效果。
* 比较度排序策略和基于投票策略在节点组选择上的优劣。
* 分析不同初始种子节点数量和感染率对信息传播效果的影响。

**五、实验内容：**

1. 网络构建: 利用邻接表构建用户关系网络。
2. 节点重要性排序:
   * 实现度排序算法。
   * 实现基于投票策略的节点排序算法，包括投票能力的计算和更新机制。
3. 信息传播模拟:
   * 基于SIR模型，模拟信息在网络中的传播过程。
   * 记录不同种子节点组合下的感染规模，作为评价标准。
4. 结果对比与分析:
   * 对比度排序策略和基于投票策略在感染规模上的差异。
   * 分析不同初始种子节点数量和感染率对信息传播效果的影响。

**六、实验器材（设备、元器件）：**

IDE: IntelliJ IDEA 2024.1

Java: 17.0.11 2024-04-16 LTS

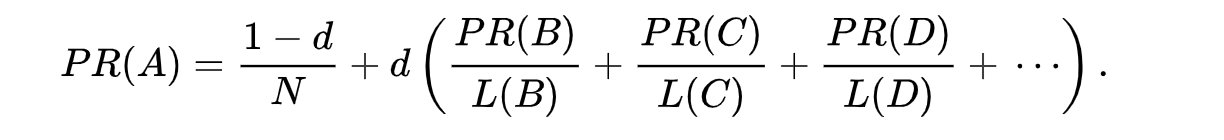
OS: Windows 10

**七、实验步骤：**

1. 根据选择的数据集构建 GraphReader 类，使用Scanner读取文件，支持读取节点与边的数量，通过邻接表构建图

2. 构建 PageRank 类，在初始化过程中读取节点数N，以 int[] 数组存储各个节点的 ID 与出度（故而一个节点还有一个属性：idx下标，用于检索其 ID 与出度），用 HashMap构建两张邻接表：二者的键都是节点的 ID，其中，graphOut 的值为一个 List<Integer>，存储其指向的节点的 ID；graphIn 的值为一个 int[]，存储指向此节点的节点的idx下标，这是为了方便在计算 PR 时快速获得投票节点。对于出度为 0 的节点，则令其指向其他所有节点，以分摊其PR值，防止“吞噬”PR

3. 采用下面的公式循环计算各个节点的 PR（默认循环500次），设置 d = 0.85



4. 建立记录 PageRankResult以存储，处理PageRank的结果

5. 建立SIRModel类，构造函数的参数为Map<Integer, int[]> graph，但与前面的graphIn不同的是，int[]存储的是ID而不是idx，在SIRModel里面没有下标的概念。在这里，我将每个节点抽象为一个Person，每个 Person 都有其id和status。Status是一个enum，有Susceptible, Infected, Recovered三种状态。在开始模拟时,所有人的状态都是Susceptible，我们需要传入一个含有初始感染者id的数组（如果没有，则随机选择），将这些人的状态设置为 Infected。然后开始循环：遍历当前的所有已感染者，感染其邻居，邻居按照感染概率将其状态调整至 Infected；当遍历完后，这些感染者按照概率将状态按照恢复率（在此实验中必然恢复）调整至 Recovered，随后不再参与模型。接着开始下一轮的感染，直到没有感染者为止。最终统计所有曾经被感染的人，与总人数之比为感染率。

6.在Main里面批量读取数据集，用Page Rank，按度选择，随机选择三种策略选择重要节点，并将结果代入SIR模型进行测试，打印测试结果

**八、实验数据及结果分析：**

我收集了以下数据：

* <https://snap.stanford.edu/data/ca-AstroPh.html>
* <https://snap.stanford.edu/data/ca-CondMat.html>
* <https://snap.stanford.edu/data/ca-GrQc.html>
* <https://snap.stanford.edu/data/ca-HepPh.html>
* <https://snap.stanford.edu/data/ca-HepTh.html>

通过程序分析，得到了这五个数据集 PR 最大的15个节点，以及基于 SIR 模型，各个选择策略（Page Rank，最大入度，随机选择）的测试结果：

在这个简易的SIR模型上，以感染率0.4，恢复率1，初始感染人数为11模拟时：

Test test\CA-HepTh.txt

Page Rank Result: Largest 15 PR:

ID PR In Degree

1 1441 0.0008750 65

2 19615 0.0007829 60

3 44262 0.0007756 51

4 23420 0.0007628 53

5 30744 0.0007615 56

6 63113 0.0006697 59

7 63697 0.0006433 38

8 14017 0.0006329 39

9 14642 0.0006328 38

10 16164 0.0006091 54

11 40517 0.0005963 49

12 59077 0.0005910 53

13 30160 0.0005821 49

14 13648 0.0005785 50

15 61742 0.0005771 50

Initializing infected [1441, 14017, 14642, 16164, 19615, 23420, 30744, 40517, 44262, 63113, 63697]

Initializing infected [1441, 13648, 16164, 19615, 23420, 30744, 44262, 48973, 59077, 61742, 63113]

Initializing infected [1049, 4885, 8660, 14481, 19177, 50213, 51152, 56264, 64829, 66996, 67269]

Test Page Rank, the infection ratio is: 0.582262

Test Largest Degree, the infection ratio is: 0.581756

Test Randomly Select, the infection ratio is: 0.575580

换了几种参数，除了感染率有明显变化外，发现三者的感染率不相上下，无法区分。得出的结论是：此模型的模拟思路无法体现初始感染者选择的差异

**九、实验结论：**

在本次针对复杂网络中重要节点组挖掘的实验研究中，我们采用了度排序策略、基于投票策略的PageRank算法以及随机选择策略来选取种子节点，并通过SIR模型模拟信息传播过程，我得出了如下结论：

* 1. PageRank算法作为一种网页排序算法，在挖掘重要节点组方面也有良好的应用。比起单一的按度选择重要节点，这种方法基于投票进行排序，能够更好地刻画节点的影响力
  2. 在简易的SIR模型中，感染率与恢复率对感染结果的影响过大，导致了初始选择节点对结果的重要性影响变小，需要更多参考现实中地情况：比如加入医院，潜伏期，口罩等一系列抽象

**十、总结及心得体会：**

PageRank算法通过链接关系评估网页重要性，其创新性和简洁性对搜索引擎产生了深远影响，尽管存在局限性，但其迭代计算方式和广泛的应用领域证明了其强大的适用性和历史地位。

**十一、对本实验过程及方法、手段的改进建议：**

* + 1. 可以使用 NDlib提供的 SIR模拟，使用NetworkX网络格式导入关系图，利用MConfig设置模型初始参数，以提供可视化模拟
    2. 也可以尝试将挖掘出来的重要节点剔除，然后随机选择初始感染者，再进行SIR模型地模拟
    3. 亦可以使用已知关键节点的网络，进行基准测试

**报告评分：**

**指导教师签字：**