華中科技大学

课 程 实 验 报 告

课程名称： 大数据分析

专业班级： 计科 2003 班

学　　号： U202015374

姓　　名： 张隽翊

指导教师： 崔金华

报告日期： 2022 年 12 月 28 日

计算机科学与技术学院

目录

实验二 PageRank 算法及其实现

2.1 实验目的

1、学习 pagerank 算法并熟悉其推导过程；
2、实现 pagerank 算法[1]；（可选进阶版）理解阻尼系数的作用；
3、将 pagerank 算法运用于实际，并对结果进行分析。

2.2 实验内容

提供的数据集包含邮件内容（emails.csv），人名与 id 映射（persons.csv），别名信息（aliases.csv），emails 文件中只考虑 MetadataTo 和 MetadataFrom 两列，分别表示收件人和寄件人姓名，但这些姓名包含许多别名，思考如何对邮件中人名进行统一并映射到唯一 id？（提供预处理代码 preprocess.py 以供参考）。

完成这些后，即可由寄件人和收件人为节点构造有向图，不考虑重复边，编写 pagerank 算法的代码，根据每个节点的入度计算其 pagerank 值，迭代直到误差小于 10^{-8}。

实验进阶版考虑加入 teleport β，用以对概率转移矩阵进行修正，解决 dead ends 和 spider trap 的问题。

输出人名 id 及其对应的 pagerank 值。

2.3 实验过程

2.3.1 编程思路

（1）构建邻接矩阵 M

预处理代码输出的 sent_receive.csv 文件给出了每一封邮件的发件人和收件人，从中可以得到通信对<sent_id, receive_id>，每个通信对代表在网络图中存在一条由用户 sent_id 指向 receive_id 的有向边。

初始创建一个 N×N 的全零矩阵 M，当存在用户 i 到用户 j 的有向边时，令

[1] 基本 pagerank 公式 r=Mr

[2] 进阶版 pagerank 公式：r= $\beta Mr + (1-\beta)\left[\frac{1}{N}\right]_{N\times N}$，其中$\beta$为阻尼系数，常见值为 0.85

M[i, j] = 1。然后归一化矩阵 M，先对 M 的每一列进行求和，再让这一列的元素除以这个和，使得矩阵的每一列元素之和为 1（全 0 列除外）。

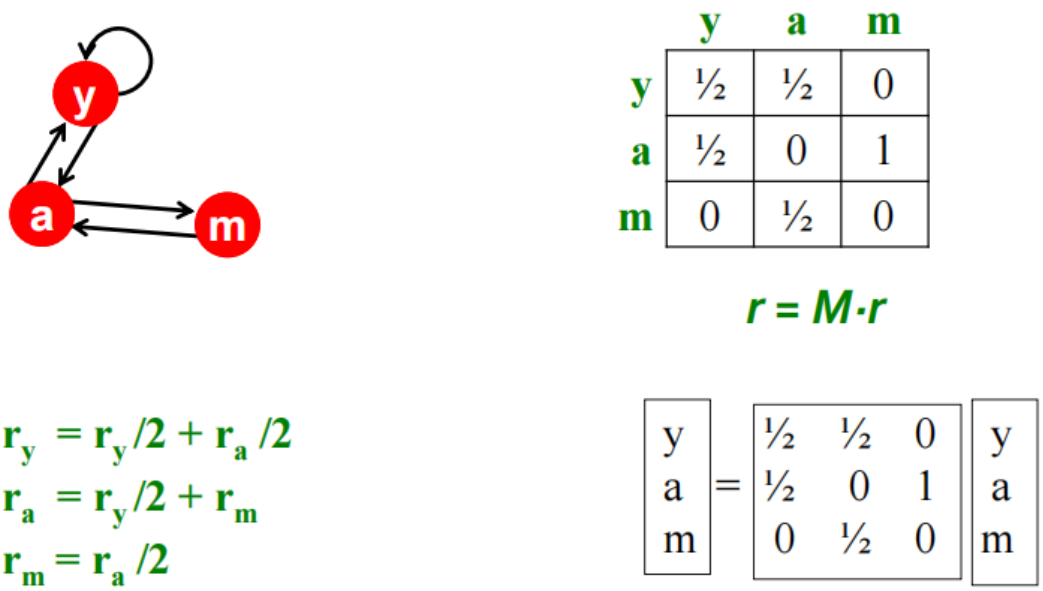

图 1.1 邻接矩阵构造过程

（2）迭代计算结果

对于基础版，将矩阵 M 带入基本公式计算 r_new；对于进阶版，利用矩阵 M 构建矩阵 A，再带入计算。迭代至相邻两次的秩向量之差小于 10^{-8} 时停止。每次迭代更新后将 r_new 归一化，计算误差后用 r_new 更新 r 再进入下一步迭代。

2.3.2 遇到的问题及解决方式

（1）构造归一化前的邻接矩阵 M 较慢

直接由<sent_id, receive_id>通信对构造邻接矩阵 M 比较缓慢，可以先利用 Pandas 读取，然后利用其 groupby 方法快速构建。

```
for (sent_id, tmp_df) in list(df.groupby('sent_id')):
    receivers = []
    for (receiver_id, _) in list(tmp_df.groupby('receive_id')):
        receivers.append(int(receiver_id))
    M[receivers, int(sent_id)] = 1
```

图 1.2 构造邻接矩阵的代码

（2）对更新后的 r_new 也要进行归一化操作

这是由计算误差的方式决定的。误差在 r_new 与 r 之差的基础上计算得到，需要 r_new 与 r 的计算规则一致，而 r 初始时进行了归一化，因此 r_new 也要进

行归一化操作。

2.3.3 实验测试与结果分析

（1）基础 pagerank 测试

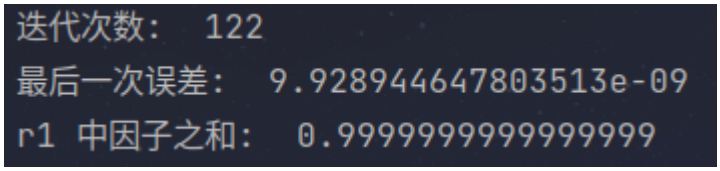

图 1.3 基础 pagerank 测试结果

由测试结果可知，最终得到的 r 向量因子之和在误差允许的范围内等于 1，迭代次数较多，退出时的误差小于 10^{-8}，满足要求。

（2）进阶 pagerank 测试

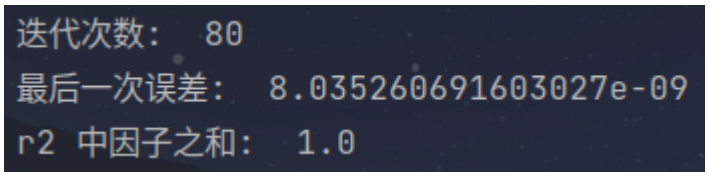

图 1.4 进阶 pagerank 测试结果

由测试结果可知，最终得到的 r 向量因子之和在误差允许的范围内等于 1，迭代次数较少，退出时的误差小于 10^{-8}，满足要求。

2.4 实验总结

本次实验实现了 pagerank 算法的基础版本的进阶版本。

在基础版本的 pagerank 算法中，仅使用归一化的邻接矩阵 M 更新秩向量 r。在这种情况下，当 M 非满秩时会导致 r 在若干次迭代后变成 0。导致 M 非满秩的可能情况有 Spider traps（蜘蛛陷阱问题）和 Dead ends（死角问题）。在 Spider traps 情况下，pagerank 算法的随机选择局限在某些结点组成的集团内部，因为集团内部的这些结点只有指向集团内部结点的边，而没有指向集团外部结点的边，在若干次迭代后集团内部的结点会聚集大多数权重；在 Dead ends 情况下，pagerank 算法的随机选择无路可走，直接导致 r 变为 0。为解决这一问题，Google 提出了改进的 pagerank 算法。

在进阶版本的 pagerank 算法中，引入了随机跳转系数 beta，每次随机选择有 1-beta 的概率通过 teleport 进行 rank 传递。这一机制的引入实质上是在邻接矩阵 M 中引入了一个元素值较小的常量矩阵，对于存在 Spider traps 和 Dead ends 情况的结点进行修正，从而使得修正后的矩阵有极大概率满秩。

在实践中，要注意及时对秩向量 r 进行归一化，否则会得出错误的结果。算法迭代过程中误差的变化趋势如图 1.5 所示。

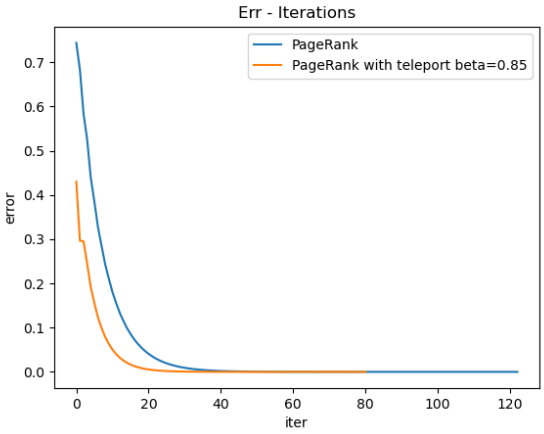

图 1.5 误差随迭代次数变化趋势图