

**课 程 实 验 报 告**

**课程名称： 大数据分析**

**专业班级： CS1806**

**学 号： U201814670**

**姓 名： 李田田**

**指导教师： 崔金华**

**报告日期： 2020.12.10**

**计算机科学与技术学院**

**目录**

[实验二 PageRank算法及其实现 1](#_Toc58252332)

**[2.1实验目的](#_Toc58252333)** [1](#_Toc58252333)

**[2.2 实验内容](#_Toc58252334)** [1](#_Toc58252334)

**[2.3 实验过程](#_Toc58252335)** [1](#_Toc58252335)

[2.3.1 编程思路 1](#_Toc58252336)

[2.3.2 遇到的问题及解决方式 2](#_Toc58252337)

[2.3.3 实验测试与结果分析 2](#_Toc58252338)

**[2.4 实验总结](#_Toc58252339)** [3](#_Toc58252339)

# 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]](#footnote-0)；（可选进阶版）理解阻尼系数[[2]](#endnote-0)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

从已有的sent\_receive.csv中将边读取出来，从这些边中提取所有的点到列表中；

得到点的总个数为N，声明一个N\*N的转移矩阵M矩阵，其所有值都为0，初始化M矩阵，使得各个边对应的矩阵值为1；

构造M矩阵，算出每一列的和为sum\_of\_col，用每一列上不为0的矩阵值除以sum\_of\_col，使得M矩阵每一列上不为0的矩阵值都为1/sum\_of\_col；

建立矩阵r为pagerank初始贡献值矩阵，每个值都为1/N；

进入循环，套用公式next\_r ，e = next\_r - r，从e的所有矩阵值中得到最大的一个值赋给e，出循环的判断条件为e是否大于10-8 。

注意点：每次迭代完要对r矩阵进行归一化处理，得到最后的迭代结果

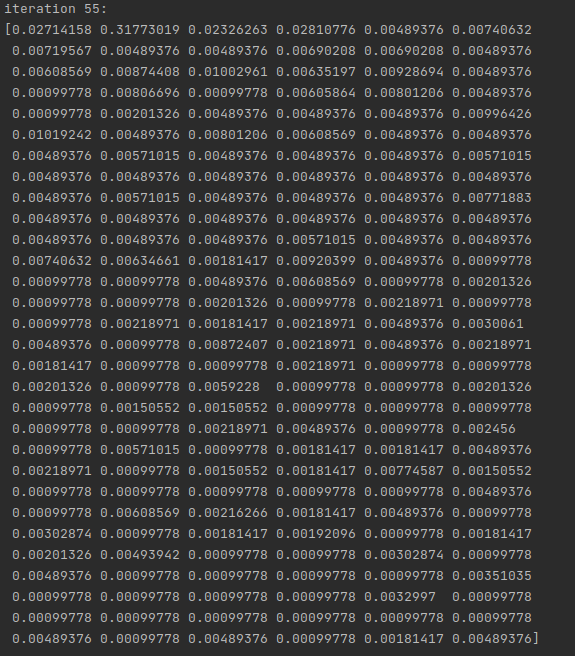
### 2.3.2 遇到的问题及解决方式

1. 在写pagerank时，使用不含阻尼系数的公式遇到最后的网页结果为0的情况，一开始以为是算法出了问题，后来询问助教后发现是会有0的结果出现。

2. 在每次迭代完之后没有对矩阵进行归一化处理，在助教检查的时候才知道每次按照公式迭代完成后需要对矩阵进行归一，即令矩阵每一列的和为1。

### 2.3.3 实验测试与结果分析

测试结果：



结果分析：输出结果为一行，该行的每一个数对应一个id的pageran值。其与上次的误差小于10-8 。

## **2.4 实验总结**

在本次实验过程中，对pagerank算法有了了解，知道其过程，通过在矩阵中的迭代过程，实现贡献度的计算。一开始不怎么了解具体过程，通过搜索查找资料了解pagerank算法的由来，突然觉得大数据都是从生活中开始，结束于规律。

1. 基本pagerank公式r=Mr [↑](#footnote-ref-0)
2. 进阶版pagerank公式：r，其中为阻尼系数，常见值为0.85 [↑](#endnote-ref-0)