**信息系统检索**

**作业一：布尔检索**

**姓名：蒋浩南 学号：2012948**

**一、实验目的：**

根据给定文档集合，利用BSBI算法实现倒排索引的构建，并使⽤可变⻓编码压缩保存到磁盘，然后实现联合查询。并额外实现gamma编码。

**二、实验思路：**

1. **BSBI算法实现倒排索引表构建**
2. IdMap辅助类，实现对BSBIndexI类所储存的term和doc的str和id转换。
3. BSBIIndex类下的parse\_block函数，将每个输入路径作为子块，构建termID和docID对。实现为两层遍历，第一层遍历根据doc划分docid，添加，第二层遍历该doc，提取词项，添加term，同时添加termID和docID对。
4. InvertedIndexWriter(InvertedIndex)中的append，根据要求的格式添加，词项和对应的倒排列表。
5. BSBIIndex(BSBIIndex)下的 invert\_write函数，将解析得到的td\_pairs转换成倒排表。遍历生成的td\_pairs,需去重。如果不是上一个termid且不是初始termid，添加上一termid的倒排列表。是上一个termid，直接将docid添加到倒排列表。
6. InvertedIndexIterator(InvertedIndex)，根据格式实现简单的迭代。
7. 合并：BSBIIndex(BSBIIndex)中的merge，类似于td\_pairs的合并。

**2.布尔联合检索**

（1）实现InvertedIndex的子类InvertedIndexMapper，它能够找到对应terms在索引文件中位置并取出它的倒排记录表。

（2）sorted\_intersect。经典的线性取交集，i和j分别为两个列表的遍历，只有当两个列表都没遍历完的时候继续循环。分三种情况，其中两种为一大一小时，小的自增；而相等时在结果中添加，都自增。

（3）利用sorted\_intersect 和 InvertedIndexMapper来实现retrieve函数。第一个token，将该token的对应id的 index\_mapper加入。非第一个，新加入要与之前加入的取交集。

**3.索引压缩**

可变长编码：

对输入的数字，用一个字节存七位有效位。不断循环右移七位，知道输入的数为零。

**4.额外编码方式**

实现gamma-encoding，同理。对int的编码，当为0和1时我们直接赋值为0，返回。对于gamma-encoding，分为两部分，首先去掉首位1,计算去掉首位1后的位数，即int(log(gap, 2))，int(log(gap, 2))个1加0为前半部分。去掉首位1后的剩余部分为第二部分，即bin(gap)[3:]，即去掉其中'ob1'

**三、说明及注意事项~~**

1.对于验证。稍微更改一下assert，改为set(my\_results) - set(reference\_results)为空，且set(reference\_results) - set(my\_results)为空；与原来的断言等价。因为有个别元素的前后顺序不一样（二分人工查出来的55555）

2. 注意建立InvertedIndexMapper时，一定要记得参数postings\_encoding=self.postings\_encoding。更新编解码方式。（找错误找的心痛）

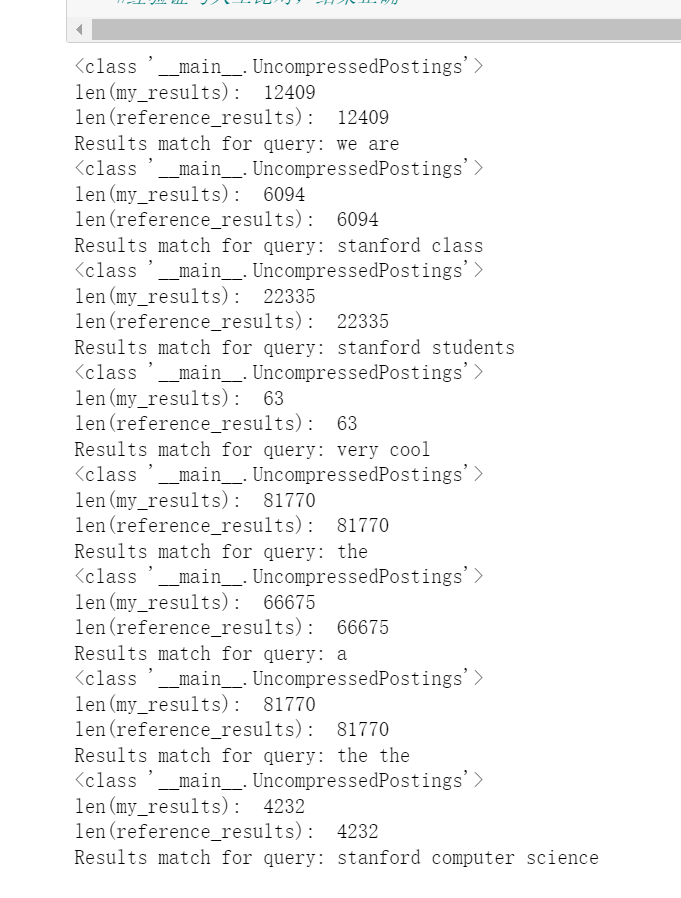
3.注意对于自己构建的路径在验证比较前要将pa1-data\\处理掉。

4.可以直接看源文件中的注释。

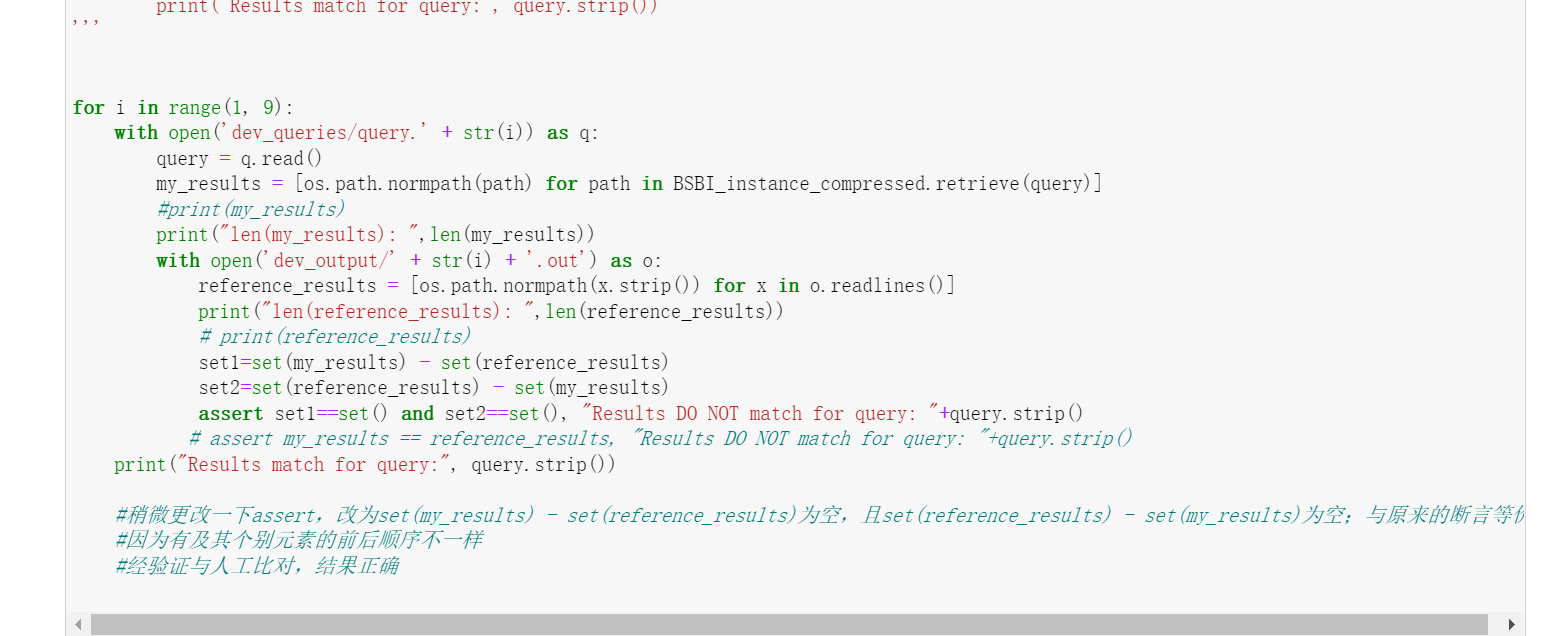
**四、实验结果**

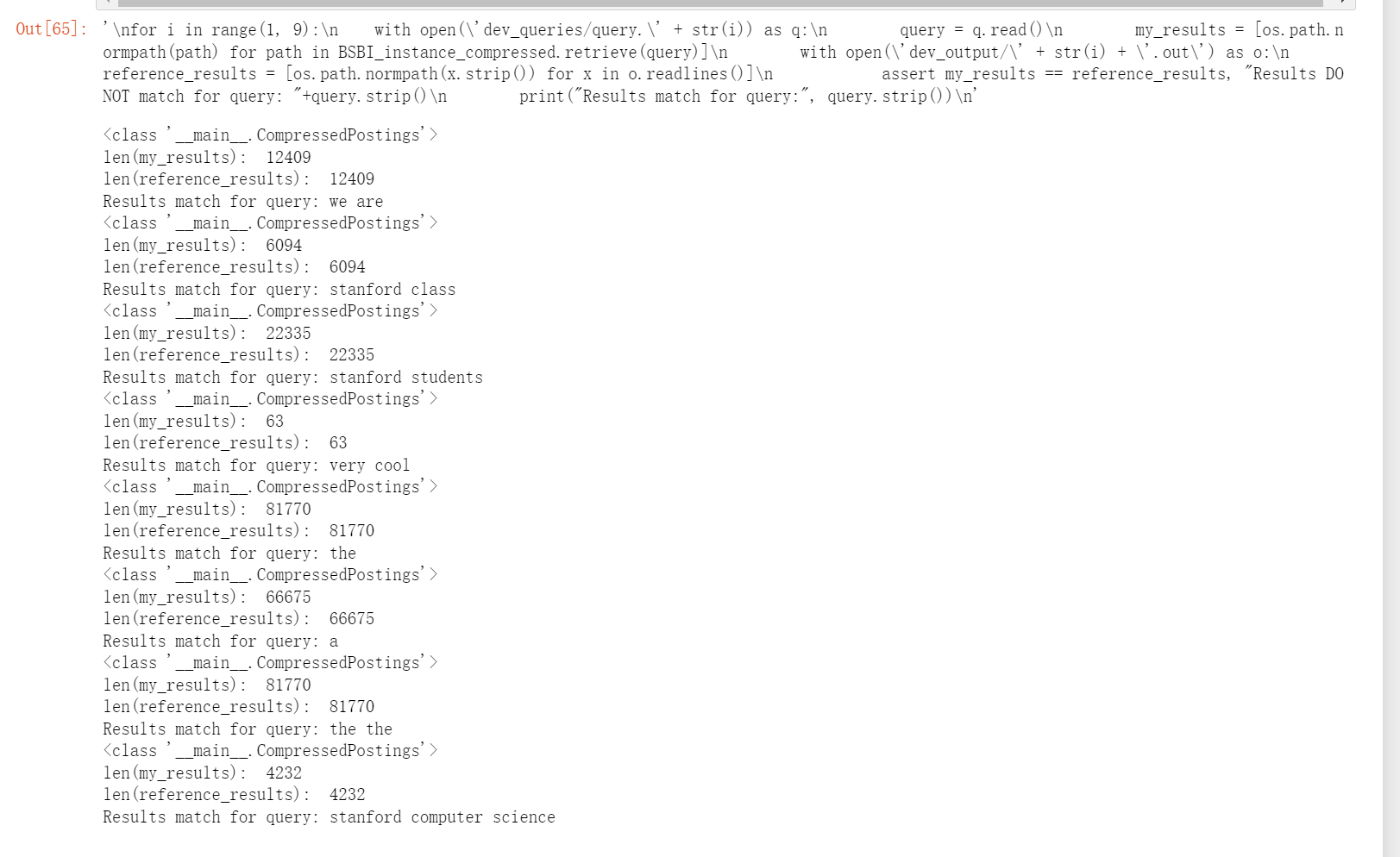
1.对未压缩的索引的验证





1. 对压缩索引的验证





验证通过