HW3.0

张劲暾 PB16111485

Problem 1:

Index	Doc1	Coc2	Doc3	Doc4
prediction	1	0	0	1
of	1	0	0	0
whole	1	0	0	0
country	1	1	0	1
sales	1	1	1	1
rise	0	1	0	1
in	0	1	1	0
July	0	1	0	1
decrease	0	0	1	0
home	0	0	1	0
June	0	0	1	0

term	DF	postings list
prediction	2	1[freq=1],4[freq=1]
of	1	1[freq=1]
whole	1	1[freq=1]
country	3	1[freq=1],2[freq=1],4[freq=1]
sales	4	1[freq=1],2[freq=1],3[freq=1],4[freq=1]
rise	2	2[freq=1],4[freq=1]
in	2	2[freq=1],3[freq=1]
July	2	2[freq=1],4[freq=1]
decrease	1	3[freq=1]
home	1	3[freq=1]
June	1	3[freq=1]

Problem 2:

基本版：

文件说明：

- `InvertedIndex.cpp`：C++ 源文件

使用方法：

```
$ g++ InvertedIndex.cpp -o InvertedIndex
$ ./InvertedIndex (文档数目) (文档所在的路径)：例如：
$ ./InvertedIndex 4 ./text/
```

- 输出：`InvertedList.txt`：倒排索引文件

测试结果：

基本满足要求，截取`InvertedList.txt`的前几行：

```
a 1 2 3 4
abandoned 3
access 3
accompanied 2
according 2
accuracy 2
actions 2
additional 3
adept 2
adjacent 4
admiral 2
advances 2
after 3
```

存在的问题：

因为没有对文本做过多的清洗，健壮性欠佳，所以这个程序在面对夹杂特殊字符的英文文档时可能会出现一些错误。

升级版：

文件说明：

- `InvertedIndexOuter.cpp`：C++源文件

使用方法：

```
$ g++ InvertedIndexOuter.cpp -o InvertedIndexOuter
$ ./InvertedIndexOuter (文档数目) (文档所在的路径)：例如：
$ ./InvertedIndexOuter 4 ./text/
```

- 输出：

`dict.txt`：产生的字典文件

`list.txt`：产生的posting list文件

测试结果：

基本满足要求，

截取`dict.txt`前几行：

```
a 0
abandoned 10
access 14
accompanied 18
according 22
accuracy 26
actions 30
additional 34
adept 38
adjacent 42
```

截取`list.txt`前几行：

```
4 1 2 3 4
1 3
1 3
1 2
1 2
1 2
1 2
1 3
```

存在的问题：

由于C++文件输入输出流对于写入字节宽度的一些特殊设置，这个程序在面对较多的（>=10个）需要处理的文件时可能会出现错误，需要根据情况对其中外部posting list文件的偏移量做一些修改

Problem 3:

假定初始查询Q为“extremly cheap DVDs cheap CDs”。文档d1包含词项“cheap CDs cheap software cheap DVDs”，文档d2包含“cheap thrills DVDs”。用户标记d1为相关文档，d2为不相关文档。假定我们直接使用词项频率作为文档向量中词项的权重，并采用Rocchio 1971算法进行相关性反馈，其中α＝1，β＝0.75，γ＝0.25，请给出修改后的查询向量

	extremly	cheap	DVDS	CDs	software	thrills
ordinary	1	2	1	1	0	0
d1	0	3	1	1	1	0
d2	0	1	1	0	0	1
revised	1	4	1.5	1.75	0.75	0

Problem 4:

查询扩展一般有几种实现方法？请比较一下它们之间的优点和缺点，并说明每一种方法分别适合于什么类型的信息检索应用。

1. **人工构建近义词词典**

- 优点：准确度高，可以包涵深入复杂的专业知识
- 缺点：构建代价大，更新慢，扩展性差
- 适用的信息检索类型：特定专业领域信息检索，行业内信息检索

2. **自动构建近义词词典**

- 优点：易于抓取词的文本特征寻找近义词
- 缺点：需要处理大量文本而且不能挖掘词汇的深刻含义，对于不同领域的近义术语难以发现，没有比较好的可定向性
- 适用的信息检索类型：普遍通用的信息检索，针对文本的信息检索

3. **基于查询日志挖掘出的查询等价类**

- 优点：基于统计学习挖掘查询词语的内涵信息，可以获得更多更深入的信息，具有较快的更新速度和扩展性
- 缺点：更新频繁会导致代价较大，数据量大的垃圾信息可能淹没数据量小的有用信息
- 适用的信息检索类型：新闻检索，商品检索