**深 圳 大 学 实 验 报 告**

**课程名称： 信息检索**

**实验项目名称：文档评分和概率检索模型的实验**

**学院： 计算机与软件学院**

**专业： 计算机科学与技术**

**指导教师：潘微科**

**报告人： 沈晨玙 学号：2019092121 班级：19计科04**

**实验时间：2022年4月29日（周五）-2022年5月11日（周三）**

**实验报告提交时间： 2022年5月8日星期日**

**教务部制**

|  |
| --- |
| **实验目的与要求：**  **实验目的：**掌握向量空间模型（Vector Space Model）、BM25、语言模型（Language Model）等文档评分和基于概率的检索技术。  **实验要求：**  (1). 用Java语言或其他常用语言计算附件“HW4\_1.txt”中的80个英文文档（每行表示一个document，文档编号1~80）两两之间的相似度值，并据此为每个文档返回相似度最大的3个文档（实验报告中只需包含文档1~10的最相似的文档）。要求使用cosine similarity和TF-IDF计算文档之间的相似度（保留小数点后两位），英文单词一律转为小写，不做过滤或其他转换。  请在报告中附上代码截图（不要复制源代码，请用截图的方式）、运行结果截图和详细的文字说明。文档中要包含与cosine similarity和TF-IDF的公式，并明确说明相应的代码所在的位置。程序要有详细的注释。（50分）  (2). 用Java语言或其他常用语言计算附件“HW4\_2.txt”中的600个中文文档（每行表示一个document，文档编号1~600，使用开源的中文分词工具）两两之间的相似度值，并据此为每个文档返回相似度最大的3个文档（实验报告中只需包含文档1~10的最相似的文档）。要求使用cosine similarity和TF-IDF计算文档之间的相似度（保留小数点后两位）。  请在报告中附上代码截图（不要复制源代码，请用截图的方式）、运行结果截图和详细的文字说明。程序要有详细的注释。（30分）  报告写作。要求：主要思路有明确的说明，重点代码有详细的注释，行文逻辑清晰、可读性强，报告整体写作较为专业。（20分）  **说明：**  （1）本次实验课作业满分为100分。  （2）本次实验课作业截至时间2022年5月11日（周三）22:00。  （3）报告正文：请在指定位置填写，本次实验需要单独提交源程序文件（源程序单独打包在Blackboard中上传，不要包含外部导入的包）。  （4）个人信息：WORD文件名中的“姓名”、“学号”，请改为你的姓名和学号；实验报告的首页，请准确填写“学院”、“专业”、“报告人”、“学号”、“班级”、“实验报告提交时间”等信息。  （5）提交方式：截至时间前，请在Blackboard平台中提交。  （6）发现抄袭（包括复制&粘贴整句话、整张图），**抄袭者和被抄袭者的成绩记零分。**  （7）延迟提交，不得分；如有特殊情况，请于截至日期之后的48小时内发邮件到panweike@szu.edu.cn，并在邮件中注明课程名称、作业名称、姓名、学号等信息，以及特殊情况的说明，我收到后会及时回复。  （8）期末考试阶段补交无效。 |

|  |
| --- |
| (1). 用Java语言或其他常用语言计算附件“HW4\_1.txt”中的80个英文文档（每行表示一个document，文档编号1~80）两两之间的相似度值，并据此为每个文档返回相似度最大的3个文档（实验报告中只需包含文档1~10的最相似的文档）。要求使用cosine similarity和TF-IDF计算文档之间的相似度（保留小数点后两位），英文单词一律转为小写，不做过滤或其他转换。  请在报告中附上代码截图（不要复制源代码，请用截图的方式）、运行结果截图和详细的文字说明。文档中要包含与cosine similarity和TF-IDF的公式，并明确说明相应的代码所在的位置。程序要有详细的注释。（50分）   1. 首先读取文件，转换为小写后根据正则表达式对文本进行切割   分割原则是非 字母与’-’  结果：   1. 统计文档集的字典   将所有term都存入set集合之中进行去重   1. 计算tf-idf    1. 计算idf   idf的计算公式如下图所示：  需要首先统计df，即每个词项在文档集中的出现次数（每个文档中只记一次）  这里使用了python的编程技巧，word in line for line in docs 会返回一个list，每个位置上为False或True，只需要统计True的数量就可以直到出现次数了。  代码如下所示：  最终可以得到每个词项的idf值。  idf: term\_set:     * 1. 计算tf   tf的计算公式如下图所示：  需要做的是统计每个词项在每条文档中的出现次数。  如果出现过，根据上述公式进行处理；否则置为0  代码如下所示：  最终可以得到每个文档的词项频率   * 1. 计算tf-idf   tf-idf的计算公式如下图所示：  代码如下所示：  最终可以得到每个文档的tf-idf向量  （其中包含大量0，是因为需要词项并没有在该文档中出现。如果为了节约空间，可以只记录index与对应的tf-idf值。但是为了后续计算余弦相似度方便，这里并没有进行处理。）   1. 计算余弦相似度   余弦相似度计算公式如下图所示：  取出需要比较的两个文档的tf-idf向量，根据上述公式计算出相似度  这里使用了numpy的线性代数库，可以简化代码的编写  最终可以得到所有文档间两两的相似度。  具体实现代码可以见代码附件（只展示了核心代码）  根据余弦相似度从大到小进行排序，展示1-10号文档相似度最大的3个文档  文档1：  文档2：  文档3：  文档4：  文档5：  文档6：  文档7：  文档8：  文档9：  文档10：  (2). 用Java语言或其他常用语言计算附件“HW4\_2.txt”中的600个中文文档（每行表示一个document，文档编号1~600，使用开源的中文分词工具）两两之间的相似度值，并据此为每个文档返回相似度最大的3个文档（实验报告中只需包含文档1~10的最相似的文档）。要求使用cosine similarity和TF-IDF计算文档之间的相似度（保留小数点后两位）。  请在报告中附上代码截图（不要复制源代码，请用截图的方式）、运行结果截图和详细的文字说明。程序要有详细的注释。（30分）  核心算法与第一问中的做法基本类似，不再重新阐释  这里展示，对于中文的分词操作：  手动添加一些计算机专业词汇进入jieba。然后使用jieba的精准模式进行切割  结果：  后续文档相似度的计算方法与第一问相同。展示1-10号文档相似度最大的3个文档  文档1：  文档2：  文档3：  文档4：  文档5：  文档6：  文档7：  文档8：  文档9：  文档10：  ++++++++++++++++++++++++++++++++++++++++++++++++++++++  **其他（例如感想、建议等等）。**  仔细对于相似度较高的几个文档后，我发现尤其是对于第二题的文档集，由于文档比较短而且领域相对集中，该算法对于一些经常出现的词项的权重打的比较低，比如“神经网络”，“协同”等。并且，由于每个文档的长度都比较短，词项基本都只出现1-2次，取对数后结果都比较小。因此我认为此算法更适合于长文本的，可能会有更好的计算结果。 |

深圳大学学生实验报告用纸

|  |
| --- |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  2022年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。