第二次实验报告

软 73 沈冠霖 2017013569

一． 实验目标

本次实验的目标是在之前解析分词的基础上，用平衡树和链表建立倒排文档，实现关键词的查询和新闻推荐，以及把结果显示在图形界面上。

二． 实验环境

查询与推荐：VS2017, Win10。图形界面：QT5.9.0，Win10。

三． 抽象数据定义

定义了一个基于二叉树 NBinaryTree, 以及二叉排序树 NBinarySortTree 的平衡树 NSplayTree, 实现了查找节点（如果找到就返回本身，没找到就返回最后搜索的节点）m_FindNode（函数在 NBinarySortTree 里）, 插入节点 m_InsertNewNode, 以及调整平衡一个二叉树（NSplayTree 的构造函数之一，由另一个二叉树构建一个平衡树）。

定义了一个倒排文档链表 NFileLink, 组合了一个链表。实现了修改文件 m_EditFile,删除文件 m_DeleteFile,搜索文件 m_SearchFile 这三个基于链表函数的功能，以及插入。因为倒排文档的建立是读取挨个文件的分词结果，而一个文件里可能含有许多个相同的单词，因此我的插入方法是：每读取到一个新词，就和对应词的倒排文档末尾比较，如果和末尾不同，则在末尾插入新节点，否则只是末尾节点出现次数+1。每读取完一个文件，再把这个文件的所有词对应的倒排文档的末尾进行调整，这样就省去了链表排序 O（m*n^2）的高复杂度，只需要 O（m*n）。插入相关函数是 m_InsertNewFile 和 m_AdjustTail。

最终的操作还是要看整体倒排文档。我定义了基于平衡树的倒排文档 NReversedWordList 来存储所有词的倒排文档，并且处理 query1 和 QT 中的关键词查询。

四． 算法说明

首先是关键词查找。这个就是先读取关键词，然后分词，然后搜索每个词的倒排文档，把文档链表合并，然后再排序，输出。这个在 NReversedWordList 的 m_HandleWordSearch 和 m_HandleWordSearchByLine 里。我的排序优先级是：先比较关键词总出现次数，再比较出现的关键词个数（比如输入“美元 经济 贸易战”，出现 100 次“经济 贸易战”的网页优先级高于“美元 经济 贸易战”各出现一次的网页）。

之后是新闻推荐。

首先，我推荐的方法是内积求关联度方法【1】。用 Jeccard 关联度和 Cosine 关联度的平均值来代表关联度高低。为什么呢？因为解析和分词算法不够精致，结果中可能有很多高频的非关键词（比如一些套话），用 jeccard 关联度，这些词只出现一次，权重小。而且结果中可能有很多低频的和内容无关的词，用 cosine 方法，这些词权重大大减小。

我推荐的具体方法是先把各个文章按照词建立正排文档，按照字典序排序，然后用类似归并排序的方法求任意两篇文章之间的关联度（jeccard+cosine）/2。

其次，因为我的解析算法并不完美，而且输入新闻标题可能也有异常，我的做法是把所有新闻的标题分词建立正排文档，然后把输入的新闻标题也进行正排文档，通过正排文档做内积的操作（求内积方法同上文求关联度方法，不过需要内积大于一个值，才能认为标题一致）来匹配标题，然后直接读取和这个标题最相关的五个新闻即可。

五． 流程概述

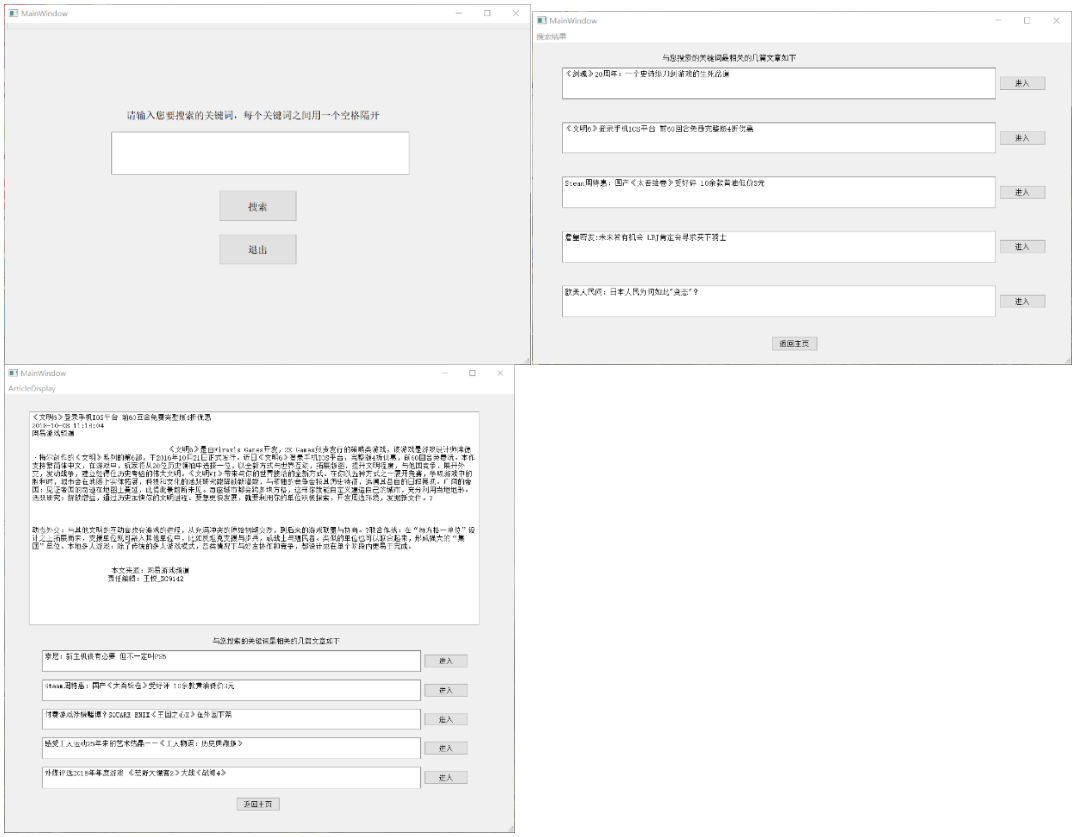
整体流程：先读取词典，解析分词输出。然后读取分词和解析文件，建立正排，倒排文档。之后比较文章之间两两的关联度，并存储每篇文章五个最接近的文章。之后读取 query 文件，进行关键词查找和推荐。

六． 图形界面

图形界面大体内容和原先代码一致，只是增加了三个窗体：mainwindow, articleselection, articledisplay，能实现输入一行关键词，返回出现这些关键词最多的五个新闻，打开新闻，显示新闻内容和相关推荐。

而 QT 显示最大的问题是中文编码问题：QT 显示中文用的是 UTF8 和 Unicode，而我的编码是 GB2312，而且 QT 使用的 QString 和我使用的自定义字符串也不兼容，因此需要复杂的转码【2】。

效果如下。



七． 文件结构说明

src 里是 VS 工程文件源代码（包括 NewsManager.sln 和 NewsManager 文件夹），和 QT 源代码（在文件夹 QT 里）

doc 里是实验报告

exe 里有 query.exe 和 gui.exe，以及其全部依赖项

详细运行方法和要求参见各个目录的 readme.txt

八． 实验结果

实验结果大体符合预期，解析分词查询推荐加一起 3 分钟左右，其中查询和推荐只要几十秒（几千组数据）。效果也大概良好，推荐的网页内容大致都比较相关。

九． 我的编程规范

我的注释主要写在.hpp和.h的函数之前，详细说明了函数的参数，输出，功能，注意事项。每个文件前也有简要的注释。在核心函数的部分代码上，我有简单的注释。

我的命名规范如下：局部变量用小写字母命名，用下划线隔开。类属性和函数用匈牙利命名法，前缀加上 m_,友元函数加上 f_，常量前加 c_,全局函数加 g_。

十． 实验总结

第二期，我吸取了前一期的教训，先以严格的单元测试测试了平衡树的基本操作，这才能让我第二期开发没在平衡树上遇到过问题，大大提高了效率。

但是，这一期也有问题：因为代码过于庞大，功能过于庞杂，维护，备份等较为复杂。以后会吸取教训，在开始写程序前先做好完整的架构。

十一． 引用的代码和感谢

【1】 http://blog.sina.com.cn/s/blog_4b59de07010166z9.html 文章相关性推荐，Jeccard 和 Cosine 方法

【2】 https://blog.csdn.net/qq_31932151/article/details/76888326 正确读取 QT 文本框输入

【3】 https://blog.csdn.net/sinat_36053757/article/details/70142078 正确显示 QT 中文

感谢张家成同学提供的大规模测试数据