《数据结构与算法》课程实验

基于文本内容的音乐检索与推荐

(第二部分)

教师：张力老师

助教：陈凯

2015 年 11 月 27 日

1、 实验目的

本次实验通过实现一个基于文本的音乐检索与推荐系统，对常用的数据结构与算法进行训练，锻炼同学们的实际编程能力。

实验要求实现以下功能：

- 根据 HTML 语法使用栈结构分析网页结构
- 提取网页中的关键信息
- 中文分词
- **倒排文档及查询系统的构建**
- **推荐系统的简单实现**

实验中涉及到的数据结构有：字符串、栈、链表、树、哈希表等。

总体来讲，通过课程实验，希望达到以下三个目标：

1）对课堂上的基础数据结构类型进行训练；

2）将数据结构知识应用到实际的软件开发中，体会数据结构的重要性和广泛用途；

3）通过实验培养学生全方面思考的能力，面对困难解决实际问题的能力。

2、 实验环境

开发环境（建议）

- 操作系统：Windows7/8

- IDE：Visual Studio 2012（建议）/ Visual Studio 2010
- 编程语言：C++

测试环境（检查标准）

- Windows 8 企业版 64 位
- CPU：Intel® Core(TM) i7-2600 CPU @ 3.40GHz 3.70GHz
- 内存：8.00GB
- IDE：Visual Studio 2012

（注：在实验 1 的批改过程中发现，部分同学使用 Visual Studio 2013 或以上版本开发的可执行程序无法直接运行。为确保作业批阅过程的顺利进行，请同学们完成作业后，务必在非 VS2013 环境下测试程序是否可以正常运行。）

3、评分方案

如果在提交的实验结果中发现相互抄袭现象，被抄袭和抄袭者的本次实验分数均为 0 分。如果发现使用第三方代码的情况，若未直接注明出处，则视为抄袭，抄袭者的本次实验分数为 0 分，若注明了，则根据使用情况酌情考虑扣分。

实验评分将依照两部分进行：系统运行、系统实现。

系统运行是指助教根据运行可执行文件的结果进行评分，包括系统的是否可执行，输出结果是否正确，系统效率等；

系统实现是指代码是否实现了要求的数据结构与算法，助教将会检查实验报告及代码实现进行给分。

具体的实验评分项将在实验内容中说明。

实验中鼓励创新，在完成基础任务的情况下，任何与实验相关的、有意义的创新都将有机会获得额外加分。加分项上不封顶，但与基础得分的总分不超过 110 分（基础满分 100 分）。

4、实验提交

最终实验要求提交 3 部分内容，请按照文件夹进行组织。在实验材料中，将包含一个提交样例目录，根据样例目录的格式，在子目录下放置对应内容。

1. 源代码：放置 VS 项目工程，删除.sdf 等大文件、编译产生结果文件。
2. 可执行文件：放置可以直接运行的可执行文件，该目录下应该同时包含 readme 说明文件及配置文件，说明如何使用可执行文件。
3. 实验报告：pdf 格式，不超过 4 页，正文使用宋体小四号字，单倍行距；

实验报告中要求提供包括但不仅限于以下信息：实验目标、实验环境、抽象数据结构说明、算法说明、实验流程、操作说明、实验结果、功能亮点、实验体会；言简意骇阐述清楚即可，不要复制代码或截图代码。鼓励图文并茂辅助说明，但注意引用图片的版权。

注：未按照要求格式提交的作业，会酌情扣分。

5、实验内容

本次实验将有实验 1 已经完成的部分，和实验 2 相关部分组成。

实验 2 在实验 1 的基础上进行，利用实验 1 的接口，以 300 个（暂定）网页作为数据库，实现根据输入内容检索音乐功能，并能够针对特定音乐根据不同的规则进行推荐。

5.1 实验 1——网页信息的提取与分词（参见实验 1 说明文档）

5.2 实验 2——音乐检索与推荐

有时候我们会参加这样的娱乐比赛，主持人报出一个字或一个词，要求每个人唱出一句包含该词的歌曲;有时候我们脑海里浮现着一句歌词，希望知道是来自哪首歌的，我们就会使用搜索引擎进行查找。这里将使用到一个技术，就是基于文本内容的音乐检索。

此外，当我们被某首歌优美的歌词迷倒时，我们往往希望听到更多类似的歌曲。根据歌曲的作词内容，可以找到类似的歌曲进行推荐，这就是基于文本内容的音乐推荐。

在本次实验中，我们希望能够实现这两项功能。

A. 实验目标：

本次实验是整个课程实验的第二部分，目标是构建网页音乐的数据库，利用倒排文档对音乐库进行组织，并对倒排文档结构使用索引（使用 B-树），从而实现对给定输入关键词的音乐检索，并在此基础上进行音乐的推荐。

具体来讲，实验可以分为 2 步。第一步，根据给定的 300 个（暂定）网页文件，使用实验 1 中的接口完成音乐信息提取和分词操作，并使用音乐信息和分词结果构建倒排文档，使用 B-树完成倒排文档的组织和索引；第二步，指定特定的网页文件后，可以根据该音乐文件的信息，返回长度为 10 的推荐音乐列表。

最终程序能够使完成查询关键字返回网页名称、根据特定网页推荐网页的功能。鼓励开发用户界面，实现用户友好的操作方式。

B. 要求实现的功能：

1. 倒排文档的建立：构建以词典和文档链表为基础的倒排文档
2. 词典索引机制实现：要求使用 B 树（必做）或哈希表对词典进行索引。
3. 文档链表的排序：文档链表需要根据单词出现次数进行排序。
4. 关键词查询：输入多个关键词，能够根据倒排文档快速返回搜索结果。
5. 音乐推荐：针对特定音乐文件，自定义规则返回推荐音乐列表。
6. *图形化界面，用户友好的操作模式。

C. 数据结构及算法要求：

实验中涉及到的数据结构与算法有：

- 数据结构：B 树、链表、哈希表
- 算法：倒排文档的构建、推荐算法

注：除特殊声明的相关实验步骤外，以上数据结构和算法需要自行实现。

本次实验中，事实上需要实现两种数据结构：词典和文档链表。其中词典和文档列表需要包含的信息将在下面给出，词典需要使用 B-树进行组织，以便快速定位。**如果有时间，可以尝试使用哈希表替换 B-树对词典进行组织，并比较两种索引机制在建立索引、查询效率等方面的差异，可适当加分。**

词典和文档链表需要实现的基本操作有：

词典：创建、添加、查找、修改、删除（选作，适当加分）等；

文档链表：创建、添加文档、查找文档、修改等。

词典的存储信息如下：

Term(String)	TermID(int)	DF(int)	Occur(int)
单词	单词 ID	单词出现在多少首音乐中	单词总的出现次数

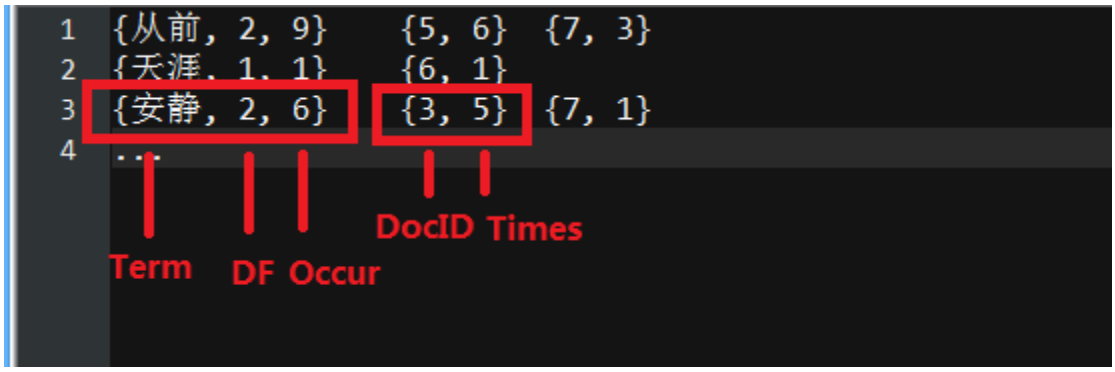
文档链表中，至少需要的存储信息如下：

TermID	DocID(int)	Times(int)	DocID(int)	Times(int)
单词 ID	出现该单词的文档 ID	单词的出现次数	出现该单词的文档 ID	单词的出现次数

其中，词典中每个单词都有一个文档链表，文档链表的节点中包含出现该单

词的文档 ID 和词语在相应文档中的出现次数。

对于部分音乐网页构建倒排文档后，倒排文档的内容可以用如下示意图描述：



文档链表的结构需要自己定义。

D. 测试方案：

输入数据：

- 300 个（暂定）网页文件（*.html）；
- 若干词库文件；
- 一个查询检索文件（query.txt）；

输出结果：

- 批量搜索：根据查询文件 1，获得查询结果 1。（搜索引擎功能）
- 批量推荐：根据查询文件 2，获得查询结果 2。（推荐音乐功能）
- 交互使用：在可执行程序中，可以自由地输入关键词获得查询结果，也可以选定某一首音乐，获得推荐列表。

测试方法：

这次作业检查助教不再使用命令行进行测试，直接运行可执行程序。

请大家在 exe 目录下放置三个 exe 文件，分别命名为 qeury1.exe，query2.exe 和 gui.exe，分别可实现批量搜索、批量推荐和交互使用功能。测试时，助教将【pages_300】文件夹、query1.txt 文件、query2.txt 文件放置到 exe 目录下，分别运行三个可执行文件，测试结果。

总之，助教只负责拷贝三份数据，和双击运行三个可执行文件，

注：

1. 本次实验鼓励但不强求大家使用配置文件。

2. 请同学将词典、停用词表以及程序所需要用到的一切相关信息（包括可能需要的 dll 文件，静态库文件等），放置在对应目录下，保证 exe 文件可以直接运行。

3. 可执行文件必须在裸机环境下测试，即不能要求用户拥有 vs 开发环境的依赖。

4. 建议大家只建立一个项目工程，因此 C++源代码只需要一份，在 main 函数中，完成数据的初始化预处理后，建立三个入口，分别执行上述三种不同的功能。

比如：

```
…（数据预处理、信息初始化）
int taskId = 1;
switch (taskId){
    case 1:
       searchBatch(……);        // 批量搜索
       break;
    case 2:
       recommendBatch(……);     //  批量推荐
       break;
    case 3:
       runGUI(……);             // 交互界面
       break;
    default:
       break;
}
……（其它处理）
```

上述代码仅供参考，只是介绍一种比较简单易读的代码流程风格。如果同学们有其它更好的方法也可以尝试。

5. 由于 C++控制台应用程序本身的限制，可能不适合作为交互界面的展示。这里鼓励大家将相关函数封装成 dll 文件，然后在构建的 GUI 程序中调用相关接口。最朴素的方法是构建一个 C#窗口应用程序，拖一些控件，在按钮的点击事件中调用相关接口获取结果。同时鼓励尝试其它交互界面的构建方式（web 前端、QT 等），但请先与助教沟通。

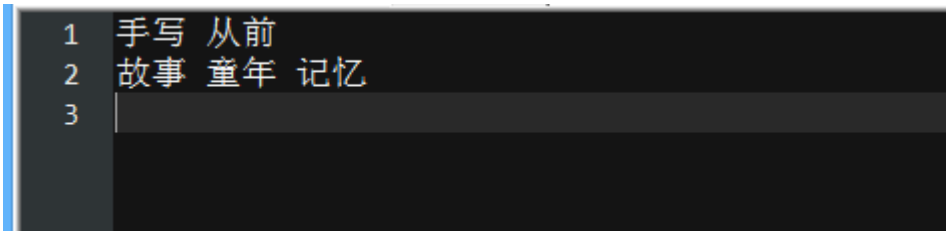
功能测试：

希望大家完成一个用户友好的，真正可用的音乐搜索引擎。另外，务必在相关目录下将程序的操作方法说明清楚。

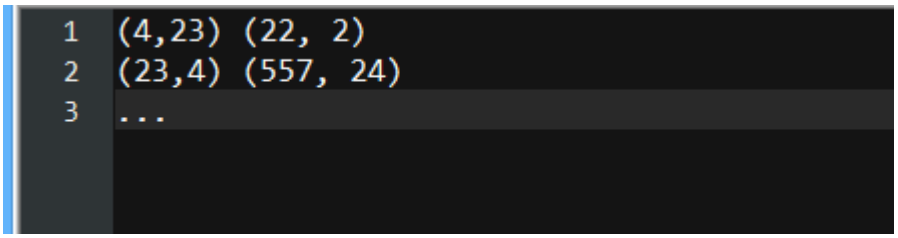
助教将根据程序使用说明，逐个测试以下功能：

1. 批量搜索：能够根据输入查询文件，得到结果文件。

其中查询文件的格式为：每一行为一个查询，关键词之间使用空格分开。比如查询文件如下：



查询结果文件保存格式为：每一行为对应的查询结果，使用(docID, occurTimes)，并用空格隔开多个查询结果；其中 docID 表示对应的文件名，occurTimes 表示多个关键字出现的总次数。例如（由于页面文件编号未定，最终结果不一定与下图相同）：



注意：

1）查询文件与 exe 目录同级，命名为“query1.txt”；

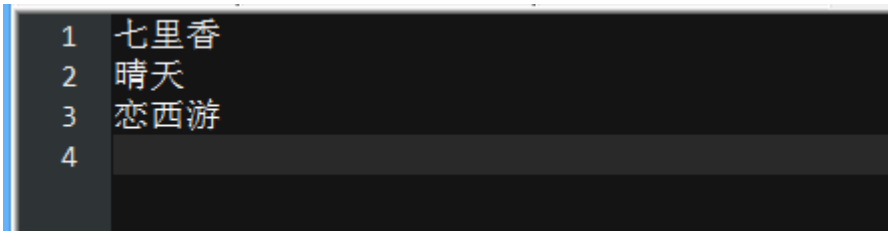
2）结果文件保存在与 exe 同级目录下，命名为“result1.txt”；

3）在执行查询之前，首先需要对输入的关键词进行分词操作；

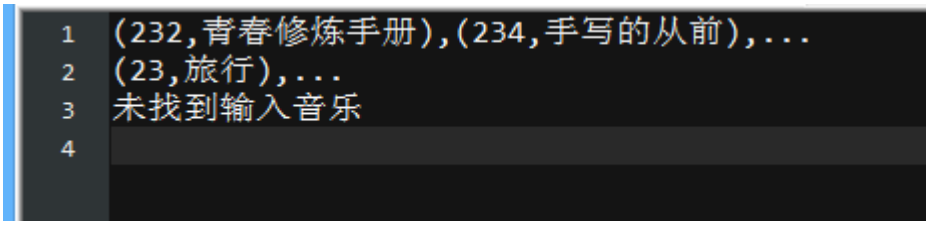
4）查询的逻辑上，需要返回出现任意关键词的文档。即多个查询词之间的关键是逻辑“或”的关系，但同时出现多个关键词文档的排序应该靠前。（此处将根据检索效果评分）

2. 批量推荐：能够根据输入查询文件，得到结果文件。

其中查询文件的格式为：每一行为一个查询，为歌曲的真实名称。



查询结果文件保存格式为：每一行为对应的查询结果，使用（docID，musicName）表示，并用逗号隔开多个查询结果；其中 docID 表示对应的文件名，musicName 表示音乐名称。如果输入歌曲名称不在曲库中，输出相关信息。例如（由于页面文件编号未定，最终结果不一定与下图相同）：



注意：

1）查询文件与 exe 目录同级，命名为“query2.txt”；

2）结果文件保存在与 exe 同级目录下，命名为“result2.txt”；

3）在执行查询前时，可能首先需要对数据库里的音乐标题进行过滤（去除一些括号后的某某音乐片尾曲信息等）；

3. gui 交互界面

输入或选定音乐的方式由程序自定义，请务必说明操作方法。

注：如果没有说明程序的使用方法，导致助教不知如何使用程序进行功能测试；或者助教在使用程序的过程中始终发生崩溃无法继续，将酌情扣分。

注：强烈建议至少完成基本的界面操作，方便选择不同功能；同时，搜索的返回结果能够通过点击等方式直接获取音乐信息，并在显示音乐信息的同时给出搜索关键词所在的位置，以及针对该音乐的推荐结果。

普通的程序界面是使用控制台：

友好的界面是能够带来良好的用户体验，并直观地显示搜索结果。

可以参考搜狗音乐的网页界面：

音乐的详情界面：

具体的界面设计请同学们可以自行完成，简洁有效，方便操作，能够显示结果就好。

E. 评分细则：

模块	内容	分数
数据结构	词典（B-树）	20%
	倒排文档及建立	15%
功能	批量检索功能	15%
	批量推荐功能	5%
	用户交互搜索与推荐	15%
	效率	10%
	用户友好性	5%
文档与代码风格	相关文档	10%
	代码风格与注释	5%
*亮点与加分项	相关特色功能点	10%

注：与实验 1 相同，**如果程序无法执行，将酌情扣分。**另外，需要提醒一下，由于数据量较大，包括数据库文本数量、词表内的词的数量等。所以同学们需要仔细研究，找出不需要的存储和计算开销，进行相应的优化。

F. 实验说明

相比于实验 1，实验 2 的难度减少了许多。主要就是当大家熟悉倒排文档的索引机制，训练 B 树等结构的操作等。

关于倒排文档的相关知识，如果有疑问可以咨询助教。

关于 B 树的构建和查找，基本上按照课程中 B 树的知识撰写即可，毕竟天下的 B 树都一样。这里请注意认真控制指针及内存分配和回收，否则容易出现内存泄露和野指针的奇怪问题。

构建 B 树完成后，基本上已经可以实现根据关键字的查询功能了。

关于音乐推荐，这是一项新颖的内容，也是一种没有标准答案的功能。由于今年第一次尝试在作业中布置这项内容，所以给大家的限制不多。只需要针对特定音乐，根据自定义的规则，返回 Top10 推荐列表即可。

对于推荐规则，只要言之有理（不是随机返回）即可。实验 1 中提取了音乐的各项关键信息，比如专辑、歌手、发行时间等，这些都可以作为推荐规则的信息来源。其中的算法同学们可以自行了解。

当然，也可以使用最简单的方法，选取当前音乐歌词中出现次数最多的 k 个词语，将它们一起/分别作为输入关键词，再执行一次查询，将查询的返回结果作为推荐列表等。只要言之有理即可。**当然，写得越有道理（不是越复杂），在评分时可以得到较高的分数。**

6、其它事项

实验报告：

除了代码工程之外，实验报告是体现你工作量的重要工具，请同学们合理分配写代码和实验报告的时间，实验报告以简洁清晰为主。

代码注释：

在实际工程开发中，代码注释非常重要。在此不给同学们规定哪里一定要写注释，但希望同学们在关键的变量、方法、算法步骤处使用注释进行简单说明，帮助他人（很可能是几年以后的你自己）理解代码的功能。

作业迟交：

作业若未能按时在网络学堂上提交，**可直接在网络学堂迟交作业窗口提交**。迟交的时间点按照助教确认为准。**若出现迟交作业且未向助教说明原因**，需要在作业评分的基础上扣除相应分数，按照迟交的天数，扣分依次为 5%、15%、30%、50%、70%、100%。迟交天数按照向上取整计算；如有特殊情况，请与助教联系协商迟交作业的解决方案。

其它未尽事宜，将在网络学堂上补充通知，谢谢。