文章编号: 1000-5013(2009)02-0155-03

一种全文索引自动答疑系统的设计与实现

林晓丹[1], 陆松年[2]

(1. 华侨大学 信息科学与工程学院, 福建 泉州 362021;
2. 上海交通大学 电子信息与电气工程学院, 上海 200240)

摘要: 提出一种基于自然语言提问的自动答疑系统的设计思想,引入Lucene的全文检索模型. 采用浏览器/服务器模式,设计并实现基于Lucene的自动答疑系统,通过对提问信息进一步的加工和提取,可实现浏览、搜索、提问等功能. 该系统能够胜任一定领域内的,基于自然语言的自动答疑需求,并充分考虑了系统的开放性和可维护性的要求. 然而,知识库还不具备自动学习功能,且对用户问题只进行简单的关键词拆分,没有进行语义分析,这使得系统还不具有完全的智能化特性.

关键词: Lucene; 可扩展Markup语言; 中文分词; 自动答疑
中图分类号: TP 391.3 文献标识码: A

基于自然语言的答疑系统就是使计算机懂得自然语言的含义,并对人给计算机提出的问题,通过对话的方式,用自然语言进行回答. 因此,设计一个使用自然语言的自动答疑系统,使学生与计算机可以自由地交谈,代替教师进行信息传递与认知活动,满足远程教育课程答疑的实时性要求,并避免重复回答某一问题. 目前,这种基于自然语言的自动答疑的研究还比较薄弱. 本文采用了自然语言处理的思想并结合了全文检索技术,提出了一种自动答疑系统的设计方法.

1 Lucene全文检索与数据库索引

全文检索是通过文本的片断信息描述得到文本资源的定位,是由小见大的信息检索模式. 计算机索引程序通过扫描文章中的每一个片断(字或词),对每一个片断建立一个索引,指明该片断在文章中出现的次数和位置. 当用户查询时,检索程序根据事先建立的索引进行查找,并将查找的结果反馈给用户. 这个过程类似于通过字典中的检索字表查字的过程,因为已经事先排好了序,所以查找效率高.

Lucene不是一个完整的全文索引应用,其最初是一个用Java写的全文索引引擎工具包. 它可以方便地嵌入到各种应用中,实现针对应用的全文索引/检索功能. 最新的Lucene版本已经加入了对中文等亚洲语言的支持,目前还有.net, Delphi等其他版本[1-2].

数据库的索引目的是提高查询速度,减少数据库操作以节省硬件资源,并非为全文检索设计. 因此,如果使用"like '%keyword%'"模糊查询,那么数据库索引就根本不起作用. 在需要大量模糊查询的全文检索系统中,使用数据库索引显然是不可行的,它对于系统的损耗相当的大,无法支持更多的用户,除非提供更好的硬件. 因此,本系统不使用数据库系统的索引机制. Lucene最核心的特征是通过特殊的索引结构,实现传统数据库不擅长的全文索引机制,并提供扩展接口以方便不同应用的定制. 此外,Lucene在维护索引文件方面也有其独到之处. 大部分的搜索(数据库)引擎都是用B树结构来维护索引,索引的更新会导致大量的IO操作. Lucene在实现中对此做了些改进,不是维护一个索引文件,而是在扩展索引的时候不断创建新的索引文件. 然后,定期把这些新的小索引文件合并到原先的大索引文件中(针对

收稿日期: 2008-09-18

通信作者: 林晓丹(1983-),女,助教,主要从事领域为中文搜索引擎和流媒体信息安全的研究. E-mail: echo.linxd@gmail.com.

基金项目: 上海市信息安全公共服务平台一期项目(pdpt2005-04)

不同的更新策略,批次的大小可以调整). 这样在不影响检索效率的前提下,提高了索引的效率. 再次,Lucene 提供了数据库索引所不具有的匹配度算法,即能够将匹配程度较高的结果排在前面. 这一特征也正符合自动答疑系统的要求,能够给予用户最满意的答案. 最后,Lucene 可以方便地定制出符合应用需要的索引规则,例如可以实现词根查询,and 和 or 等逻辑操作的组合查询.

2 系统框架及关键技术

2.1 系统框架

系统采用浏览器/服务器(B/S)架构实现. 用户使用浏览器登陆系统,服务器端需要维护一个知识库、一个问题库和词典. 用户通过 JSP 页面提交问题后,自动答疑程序取出问题,并根据词典对用户的问题进行中文分词. 系统采用 Lucene 对知识库预先做好全文索引,再根据中文分词的结果在索引中查找,返回相似度最高的回答. 若无法在知识库中找到答案,则把问题加入问题库,等待其他用户回答. 用户提交答案也是以 JSP 页面的形式将答案返回服务器端,服务器把新的答案加入到知识库中.

2.2 关键技术

(1) 中文分词. 目前的分词算法可分为两大类,即机械性分词和理解性分词法. 理解性分词法在解决歧义方面有其优势,但由于语法分析、语义分析,乃至篇章理解等技术还没有很好的解决方法,故现有的分词实用系统主要采用机械分词法. 机械分词法有正向最大匹配、逆向最大匹配、双向最大匹配、最佳匹配法、最少分词法、词网格算法等[3-4]. 考虑到速度方面的原因,系统采用了正向最大匹配的方法.

这种方法按照一定策略,将待分析的汉字串与一个词典中的词条进行匹配,若在词典中找到某个字符串,则匹配成功,识别出一个词. 基本思想是:设 D 为词典,MAX 表示 D 中的最长词,str 为待切分的字符串. 正向最大匹配法每次从 str 中取长度为 MAX 的子串与 D 中的词进行匹配. 若成功,则该子串为词,指针后移 MAX 个汉字后继续匹配;否则,子串逐次减 1 后再次进行匹配,即匹配的方向是从右向左. 在这种分词方法中,词典对分词的准确度起了决定性的作用. 本系统在分词时,按照专业词典、数字词典、普通词典的顺序分词,既先在专业词典中查找匹配. 例如,“计算机病毒”作为一个词存放于专业词典中,分词时直接在专业词典中匹配到这个词. 即使普通词典中也包含“计算机”和“病毒”这两个词,但这时已经不需要再做分词了. 这个词典可以由教师手动维护. 为了提高分词效率,首先去掉一些没有实际意义的词,然后再对剩余的词进行分词. 经过处理后,需要切分的词长度就减小了.

(2) 问题库和知识库的组织. 问题库和知识库都以 XML 文档的方式存储. 知识库中的每个节点包含了编号、提供答案的用户名、答案 3 个元素,问题库中的每个节点则包含了问题编号、问题内容和提问者的用户名等信息. 首先,XML 页面能包含更多的内容,表现更复杂的形式[5],XML 的页面信息具有结构化的特性且更具访问性,保证了检索结果具有针对性和更加准确. 其次,避免了经常性的数据库连接、释放等大量消耗系统资源和时间的操作. 最后,由于采用的 JSP 技术,XML 数据源用 Java 解析后是 UNICODE,因此无论是日文、繁体中文,还是德文的内容,都可以在一个索引库中同时进行搜索. 针对其他语言的支持只是设计各种语言界面的问题了,使得系统具有良好的可移植性和灵活性.

(3) XML 文档解析. 将知识库的节点和索引中的记录相对应,每个元素下面的子节点也同记录中的域对应. 这里编写 1 个名为 ReplyInfo 的类,包括 3 个数据成员,即 number,username 和 answer,分别对应 XML 文档中存储的编号、提供答案的用户名和答案. 相应的,在索引中建立 1 个域,有 number,username 和 answer.

(4) 全文索引的实现. Lucene 的索引(Index)由段(Segment)组成,段由记录(Document)组成,记录由域(Field)组成,域由字符串(Term)组成. 这种结构类似数据库表记录的结构,索引中的每个段对应着一系列的文件,或者称为记录. 在不同的索引中,段对应文件的数量是不同的,这取决于索引中包含域的数量. 段相当于一个子索引(SubIndex),只是段之间并不像索引之间是相互独立的. 索引一旦建立,索引里段的数量就已经固定下来. 在索引增量扩展的过程中,每当有新的记录添加进索引的时候,段的数量会发生变化. Lucene 会自动控制段的数量,并根据系统设置在适当的时候将段合并.

Lucene 定义的查询结果集(Hits)由匹配的 Document 组成. 创建索引及分析查询有如下 5 个主要步骤. (1) 使用 IndexWriter,在指定的目录里建立索引的文件. 即知识库文件 reply.xml. IndexWriter

writer = new IndexWriter(indexpath, new ChineseAnalyzer(), true), 参数 indexpath 为索引文件存放的路径, 第 2 个参数是按照本文前面的分词算法对 Lucene 的语言分析器改造后的中文分析器. 该分析器与 Lucene 的二元分析器相比, 分词效果优越了许多, 更符合查询要求. 第 3 个参数 true 表示可以重建索引文件, 以便定期更新索引. (2) 将需要检索的数据 answer, username 和 number 转换为 Document 的 Filed 对象, 然后将 Document 用 IndexWriter 添加到索引的文件中. number 和 username 字段只存储不索引, 而对 answer 字段既索引又存储. 方法为

```
Document doc = new Document();
doc.Add(Field.UnIndexed("number", replyInfo.GetNum()));
doc.Add(Field.UnIndexed("username", replyInfo.GetUsername()));
doc.Add(Field.Text("answer", replyInfo.GetAnswer()));
writer.addDocument(doc)
```

(3) 处理索引信息, 关闭 IndexWriter 流. (4) 创建查询分析器 Query. 按照 answer 域对分词后的关键字进行匹配. Query query= QueryParser.parse(queryString, "answer", new SimpleAnalyzer());// queryString 为切分用户提问得到的关键词. (5) 给 IndexSearcher 指定索引文件的路径后, 用 Query 检索后返回 Hits 对象作为结果. Hits hits= searcher.search(query);

3 结束语

提出的一种基于 Lucene 的全文检索自动答疑系统, 对于问题的查询和匹配, 无论是速度还是准确度, 与传统的自动答疑系统相比都有一定优势, 对用户的提问所返回的答案能较好满足需要. 但是, 其知识库还不具备自动学习功能, 且对用户的问题只进行了简单的关键词拆分, 没有进行语义分析, 这使得该系统还不具有完全的智能化特性.

参考文献:

[1] CUTTING D, GOSPODNETIC O, GOETZ B, et al. Lucene open source material[EB/OL]. [2006-05-27]. http://akarta.apache.org/Lucene.

[2] 车　东. 在应用中加入全文检索功能——基于 Java 的全文索引引擎 Lucene 简介[EB/OL]. [2002-08-06]. http://www.chedong.com/tech/lucene.html.

[3] 马玉春, 宋瀚涛. Web 中文文本分词技术研究[J]. 计算机应用, 2004, 24(4): 134-135.

[4] 王　坚, 赵恒永. 专业搜索引擎的实现与研究——中文分词算法[J]. 电子科学技术评论, 2005, 8(3): 77-79.

[5] 李　爽, 陈　丽. 国内外网上智能答疑系统比较研究[J]. 中国电化教育, 2003, 12(1): 80-83.

Design and Implementation of an Automatic Question-Answer System Based on Lucene Full-Text Retrieval Model

LIN Xiao-dan[1], LU Song-nian[2]

(1. College of Information Science and Engineering, Huaqiao University, Quanzhou 362021, China;
2. School of Electronic Information and Engineering, Shanghai Jiaotong University, Shanghai 200240, China)

Abstract: This paper proposes a design ideal of an automatic question-answer system by adopting Jakarta Lucene full-text retrieval model. Built on a browser/server mode, this system is designed for information retrieval by analyzing keywords contained in the questions and avails people to raise new questions. It can meet the requirements of auto-answering in some certain fields and also takes extensibility and maintainability into account. However, the system isn't intelligent enough since only keyword splitting is employed without any semantic parsing and self-learning is not available yet.

Keywords: Lucene; extensible Markup language; Chinese word segmentation; auto-answer

(责任编辑: 陈志贤　　英文审校: 吴逢铁)