CLUCENE在语料库建设中的应用

贺 胜[1], 曲维光[2], 卢亚军[3]

(1. 南京师范大学 文学院, 江苏 南京 210097; 2. 南京师范大学 数学与计算机科学学院, 江苏 南京 210097;
3. 西北民族大学 藏语言文化学院, 甘肃 兰州 730030)

[摘要] 深入分析了现有语料库的构建模式和语料库应具备的功能模块, 提出基于文件系统和 Clucene全文检索引擎工具包的语料库建设方案. 实验证明, Clucene具有丰富的接口设计和良好的扩展性, 为语料库建设提供了一种较好的技术实现方式.

[关键词] Clucene; 语料库; 语料库建设
[中图分类号] TP 301 [文献标识码] A [文章编号] 1672-1292(2008)04-0118-05

Applying CLUCENE in Corpus Building

He Sheng[1], Qu Weiguang[2], Lu Yajun[3]

(1. School of Chinese Language and Literature, Nanjing Normal University, Nanjing 210097, China;
2. School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China;
3. School of Tibetan Language and Culture, Northwest University for Nationalities, Lanzhou 730030, China)

Abstract This paper examines deeply the constructed models of the current corpus building design and the functions corpus should have. A new corpus design based on file system and Clucene full text searching engine package is proposed. Experiments show that Clucene provides various types of interfaces and can be easily extended for large quantity data. These characteristics make the package a promising platform for corpus building.

Key words Clucene; corpus; corpus building

语料库顾名思义是指存放语言材料的仓库, 一般指收集并存储在计算机中并可以用计算机处理的电子文本库, 它已成为语言研究、自然语言处理、人工智能研究等方面不可缺少的基础资源. 语料库的建设涉及语料的采集、管理和语料加工、检索、统计等功能模块的软件开发等方面. 随着 Internet的飞速发展和电子图书的日益普及, 语料的获取已不再困难, 但如何高效地组织、管理和加工大规模语料库, 并根据语言研究、自然语言处理的具体任务快速、准确、全面地从中检索、统计到用户所需要的信息是当前语料库建设中亟待解决的重要问题.

1 语料库的构建模式

语料库的的构建模式是指语料库的组织、存储、管理方式, 它决定着语料库的性能, 包括语料规模 (存储量), 检索效率 (访问速度), 系统的可操作性、开放性和可扩展性等. 综合现有的语料库系统, 主要有 4种构建模式: 即文件系统、XML、关系数据库和文档数据库.

1.1 文件系统构建模式

操作系统中负责管理和存储文件信息的软件机构称为文件管理系统, 简称文件系统. 基于文件系统构建语料库时一般都是以文本文件的形式存储语料, 语料库是相同结构的文本文件的集合. 文件系统构建语料库的优点是: 便于系统从中提取自然语言的各种统计数据; 便于为各种语言处理模型提供训练和测试材料; 便于按字或按词建立索引并进行全文检索; 便于对语料进行自动分词、词性标注等方面的加工; 存储容

收稿日期: 2008-06-18
基金项目: 江苏省社会科学基金 (07YYB003、06JSBYY001)、国家自然科学基金 (60773173)、国家社会科学基金 (07BYY050)、国家社会科学基金 2005重点项目 (05AYY001)和国家"973"计划 (2004CB318102)资助项目.
通讯联系人: 贺 胜, 讲师, 博士生, 研究方向: 中文信息处理. E-mail: hesheng99@sina.com

量不受存储结构的束缚, 开放性好等. 其不足之处在于: 语料对程序依赖性强; 当语料文本包括元数据 (元数据是关于语料特性的描述, 如语料来源、文体、主题、作者等)、正文以及其它格式的内容时, 程序控制就显得比较复杂, 数据的一致性较难控制; 当语料库的数据结构改变时, 程序要作较大的修改[1].

1.2　XML构建模式

XML(Extensible Markup Language)称为可扩展标记语言, 能够很好地表现复杂的数据关系, 可用来描述任何类型的文档, 可定义适合自己所需要的附加标记集合. 使用 XML语言组织语料库时, 一个语料库的文件是一个或多个 XML格式文件的集合. 用 XML组织语料库的优点是: 数据跨平台, 易于交换, 通用性强; 可以依据 DTD(数据类型定义)通过常用的网页浏览软件 (如 IE5.0)来检查语料文件的结构是否规范, 因此解读语料文件的程序就不用像传统的文件系统那样, 过多地在程序中解决数据存储结构的问题, 从而提高了语料数据和程序的独立性、共享性. 其不足之处是: 基于文件的管理机制, 难以管理大批量文档; 基于节点的检索方式, 不适合大规模语料的检索; 另外, 解析手段仍存在缺陷, 修改效率低; 安全性和并发操作机制还不够强大等.

1.3　关系数据库构建模式

用关系数据库来构建语料库时, 一条语料记录就是关系数据库的一条记录, 语料文本用一个能支持大文本的数据字段来存放, 语料库的每一个子库对应关系数据库的一个关系. 关系数据库构建语料库的优点是: 可以充分利用数据库管理系统提供的功能, 使语料的插入、删除、更新、备份、查询、统计都比较容易, 特别是能方便地对语料的元数据进行查询、更新、统计; 结构化的数据存储, 使得数据冗余度低, 程序与数据独立性较高; 系统易于扩充、易于编制应用程序, 开发效率高. 其不足之处在于: 如何对关系数据库中的大文本按字或者词建立索引, 并实现索引和关系数据库的无缝连接, 还是一个有待研究的课题. 虽然一些专用的全文检索软件较好地解决了这个问题, 但是, 当语料库规模大时, 检索效率较低.

1.4　文档数据库构建模式

文档数据库区别于传统的其它数据库, 它的基本要素是文档, 能够存储和管理类似文档这样的非结构化数据. 在传统的数据库中, 文档被分割成离散的数据段进行处理. 而在文档数据库中, 文档作为信息处理的基本单位保持其完整性. 文档可以很长、很复杂, 文档的格式可以灵活多样, 可以无结构, 可以包含各种信息等[2]. 文档数据库构建语料库的优点是: 可以轻松应付成千上万篇文档的维护和管理, 发布、查询、浏览文档方便快捷; 允许创建许多不同类型的非结构化的或任意格式的字段. 不足之处是: 不提供如关系数据库中的参数完整性支持, 不提供分布事务的支持, 并行处理能力较弱. 目前使用文档数据库来存储语料还比较少, 但这种善于存储非结构化的数据的数据库产品可能会成为未来语料库的发展趋势.

2　语料库的功能模块

要使语料库在语言研究和自然语言处理各个领域的研究和应用中发挥作用, 就需要强有力的工具进行数据管理、语料加工、用户服务等[3]. 这些工具就是建设与应用语料库的功能模块, 其中数据管理部分主要涉及文本格式转换、字符编码转换、文本分割、符号置换、文件管理等技术. 为了保证语料格式的统一, 通过各种方式采集的语料需转换成统一的格式后才能加入到语料库中. 语料加工的主要内容是自动分词、词性标注、语法特征标注、语义标注等各种语言学属性的自动标注. 为从语料库获取语言知识, 必须在各个层次上对语料库进行加工. 对语料库的加工深度决定该语料库能为语言研究和自然语言处理提供什么样的信息. 例如, 通常把没有语言学属性标注的语料叫做生语料, 对汉语的生语料只能以字为单位进行检索和统计; 经过分词加工处理的语料则能以词为单位进行检索、统计和定量分析; 如果还作了词性标记, 那么可以获得的语言学信息就更多了. 用户服务主要是指面向用户的语料检索、统计和分析技术. 语料库建立之后, 将提供给各种需求的用户使用, 使其能够访问语料库的内容, 对自然语言进行相关的分析、研究. 因此, 语料的检索是语料库最基本的功能之一. 语料库检索属于全文检索, 但仅用普通的全文检索技术尚不能满足基于语料库的检索需要. 这是因为全文检索一般关心的是检索的内容, 不是检索目标的语言表述形式. 而面向语言研究的语料库检索则特别注重语言的表述形式、语言现象及其上下文, 它既需要按照字、词、字串检索, 也需要把词语或字串的语言学属性作为检索的目标和约束条件, 还要求把检索的结果或结果的出处按照研究的需要排序、输出. 除此之外, 还要有语料元数据、字频、词频和特定语言形式出现频率

的查询、统计功能等[4].

3　Clucene简介

Lucene是国外 Apache软件基金会 Jakarta项目组的一个子项目,是一个开放源代码的全文检索引擎工具包,是用纯 Java语言写的全文检索引擎架构,它可以方便地嵌入到各种应用中实现全文索引和检索功能[5]. Clucene是 C++版的全文检索引擎工具包,完全移植于 Lucene,采用 STL编写.相对于 JAVA版的 Lucene, Clucene运行效率更高,适用于基于文件系统的文档集的全文检索和海量数据的模糊检索,特别是对大数据的字符类型进行检索更显示出它的高效性[6]. 在 Clucene的基础上开发中文全文检索是一种高性能的选择,因为其全文数据库采用倒排索引技术,对于大规模的全文检索系统来说,倒排索引是目前最高效的数据结构,查询速度要优于现有的数据库系统. Clucene中最基本的概念是索引、文档、字段和项.文档代表一种逻辑文件,当与一个物理文件对应时,可以提取出多种数据源,如文件名、作者、文件内容、文件创建时间等,每种数据源对应一个字段,因此字段适合于表示语料的各种元数据与正文.

Clucene有两个主要的服务,即索引和检索. Clucene并未规定索引数据源的格式,而只是提供了一个通用的文档对象来接受索引的输入,因此输入的数据源可以是任何类型的文本数据,如 TXT文本、Word、PDF、HTML文档等,只要设计相应的文档分析器将数据源构造成文档对象中的字段即可进行索引.检索是根据查询条件返回结果, Clucene具有丰富的查询模式,包括布尔查询、短语查询、范围查询、组合查询、跨度查询等.通过 Clucene提供的检索接口,用户只需简单地将查询对象传入,就可以得到返回结果.

Clucene的主要功能都采用了抽象类,开发人员可以对它们进行扩展来满足自己的需求.例如, Clucene是针对西文的文法开发的,在倒排中文文档的时候,无法正确地将中文切分开,也不支持中文查询.我们可通过对语言词法分析接口进行扩展加入中文单字切分、词切分等扩展模块,从而使其支持对中文的处理.在应用接口方面, Clucene提供了丰富的应用接口函数,可以与存储在索引库中的信息交互.通过接口函数的调用,可以轻松地完成元数据与正文的检索和一些信息的统计工作.如:可以列出索引库中文档总数及文档列表;可以查询具体某个项在索引库中的文档频度、总词次,以及在某个文档中出现的频度和在某文档中出现的位置等信息.

4　Clucene在语料库中的应用

由于文件系统在构建语料库时,可以方便、快捷地将不同来源、不同类型的语料分门别类地存放在不同的文件夹下,具有不受存储结构的束缚,开放性强,且易于对语料文本进行加工、检索、统计等优点.结合 Clucene主要适用于文档集的全文检索,系统运行效率高,功能易扩展,接口函数丰富等特点,我们设计了一个基于文件系统和 Clucene全文检索引擎工具包的现代文学作品语料库.系统结构如图 1所示.

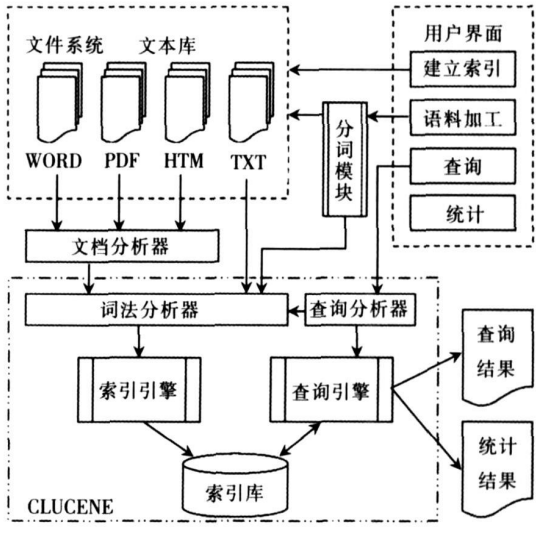

图 1　语料库应用系统结构图

Fig.1　The frame of corpus application

4.1　Clucene在语料库建设中的应用实例

在这里我们以构建汉语现代文学作品语料库为例,对 Clucene在语料库建设中的应用及其技术实现概述如下.

(1) 构建现代文学作品文本库:一个语料库系统首先要做的事情,必须收集和构建一个文本数据库.我们将所收集到的现代文学作品,基于文件系统模式,采用层次的目录结构,将不同类型的文本分门别类存放在不同的文件夹及子文件夹下,使之构成一个文本库.

(2) 语料加工:语料加工主要是文档预处理和自动分词及词性标注.预处理包括文本类型转换、字符编码转换、繁简转换、标点符号置换等;自动分词及词性标注是系统调用自动分词模块,实现对指定文件夹

下所有纯文本文件(TXT)的自动分词、词性标注.

(3) 建立全文索引库: 有了文本库之后, 就可以建立全文索引库. 从语料库多用途的应用角度出发, 系统除了提供字索引、词索引及相关检索方式外, 还提供了词及词性混合索引及相关检索方式. 对应这 3种索引模式, 我们对系统的词法分析器进行了功能扩展, 使之可实现中文字串的字、词、词及词性的切分, 对应建立基于字、词、词及词性的 3种索引库. 利用 Clucene创建索引库只需指定要索引的文本库位置、索引库的存放位置和相应的词法分析器即可. 下面以字索引库的建立为例, 介绍字索引库的建立.

假定语料的元数据为: 文件路径、篇名、作者. 关键代码如下:

```
//构造一个写索引器, 词法分析器为中文字分析器(StandardAnalyzer())
IndexWriter writer= new IndexWriter( indexDir, new StandardAnalyzer(), ture);
//遍历文本库文件夹下的所有文本文档, 为每个文档创建了一个文档(Document)对象
Document doc= new Document();
doc. add(Field UnIndexed("path", getFiles[ i]));  //将文件路径加入 path字段
doc. add(Field UnIndexed("name", getFilename[ i]));  //将篇名加入 name字段
doc. add(Field UnIndexed("writer", getWriter()));  //将作者加入 Writer字段
Reader txtReader= newFileReader(Files[ i]);
doc. add(Field Text("contents", txtReader);  //将文本内容加入 contents字段, 进行全文索引
writer addDocument(doc);  //将文档写入索引库
```

参数 indexDir是存放索引的目录; 创建索引最重要的类是 Indexwriter, 其构造器有 3个参数, 第 1个参数指定了存储索引文件的路径, 第 2个参数指定了在索引过程中使用什么样的词法分析器, 最后一个参数是个布尔变量, 用于控制是重建索引, 还是在原有索引上添加.

4.2　现代文学作品语料库的应用

(1) 进行检索. 在上面的例子中, 我们已经为现代文学作品语料库建立好了基于字的索引库, 现在我们可以利用这个索引库进行检索, 以找到包含某个关键词或短语的所有文档. 假设根据 contents字段进行全文检索, 将查询结果以 name字段内容输出. 关键代码如下:

```
IndexSearcher searcher= new IndexSearcher( indexDir) //实例化 IndexSeacher对象
//声明一个查询解析器对象
Query query= QueryParser. parse( querystring "content", new StandardAnalyzer());
Hits hits= searcher search( query);  //搜索结果
//通过 hits可以访问到相应字段的数据
for ( int i= 0;  i< hits. length();  i+ + ) { printout ( hits. doc( i). get("name")); };
```

这里搜索过程主要用到 2个对象 Indexsearcher和 Query. Indexsearchersh是查询索引库的类, 参数 indexDir为前面索引存放的位置. Query用来处理搜索请求, 它包含了 3个参数: 查询内容、查询字段、采用的词法分析器, 参数 querystring为要查询的字符串.

(2) 对结果进行处理. 通常, 从语料库检索到用户需要的结果集后, 可以根据 Lucene提供的 API取出索引库中相应的数据, 并以一定的规则进行排序、过滤、关键词对齐等, 然后再返回给用户. 如 sort是 Clucene一个排序接口, 通过它可以方便地以某个字段(Field)的值为参数对检索结果进行排序.

(3) 统计功能. 原始文本经词法分析器处理后, 文本切分成一个个的 Term(或 Token), 同时被索引器纪录下它的所在位置、出现次数等信息. 有了这些数据, 利用查询引擎 IndexReader及相关的接口函数就可以从索引库中获取到一些统计信息. 如: NumDocs() 获取整个索引库中文档的个数; Terms()列举索引库中所有的项(Term); DocFreq()获取某个项(Term)在某文档中出现的频次; TermPositions()列出某个项在某文档中出现的位置信息等.

5　结论

本文通过分析现有语料库系统的构建模式和语料库应具备的功能模块, 展示了基于文件系统和 Clucene全文检索引擎工具包的语料库建设的具体示例. 表明以此示例所呈现的方案, 可以方便地将文档以文件系统的方式构建成文本库, 在文本库的基础上建立基于 Clucene的全文索引库, 从而轻松实现语料库

的加工和检索、统计等应用.实验证明,Clucene具有丰富的接口设计和良好的扩展性,适用于海量文本数据的检索和查询,为语料库的建设提供了一种较好的技术实现方式.

[参考文献](References)

[1] 何婷婷.语料库的数据管理方式研究[C]//第一届学生计算语言学研讨会论文集.北京:清华大学出版社,2002:307-310
He Tingting. Study on data management of corpus[C]// Proceedings 1st Students Workshop on Computational Linguistics. Beijing: Tsinghua University Press, 2002: 307-310. (in Chinese)
[2] 金天荣.文档数据库与关系数据库研究[J].微计算机信息,2008(3):173-174.
Jin Tianrong. Research on the document database and relationship database[J]. Microcomputer Information, 2008(3): 173-174. (in Chinese)
[3] 傅爱平.语料库研究与应用综述[DB/OL].[2007-10-22]. http://ccl.pku.edu.cn/doubtfire/CorpusLinguistics/Introduction/FuAiping_Corpus_introduction.pdf
Bo Aiping. Study and application summarization of corpus[DB/OL]. [2007-10-22]. http://ccl.pku.edu.cn/doubtfire/CorpusLinguistics/Introduction/FuAiping_Corpus_introduction.pdf. (in Chinese)
[4] 贺胜.面向大规模语料库的全文检索系统研究[J].图书与情报,2008(4):93-97
He Sheng. Research of full-text retrieval system for large-scale corpus[J]. Library & Information, 2008(4): 93-97. (in Chinese)
[5] 贺胜.基于Lucene的中文全文检索系统[J].中国高校科技与产业化,2007(6):142-144
He Sheng. Chinese full-text retrieval system based on Lucene[J]. Chinese University Technology Transfer, 2007(6): 142-144. (in Chinese)
[6] Clucene-a C++ Search Engine[EB/OL]. [2007-10-12]. http://sourceforge.net/projects/clucene.

[责任编辑:丁蓉]