◎数据库、信号与信息处理◎

基于语义相关度排序的政务信息资源检索算法

陈 旭,陈德华,乐嘉锦
CHEN Xu,CHEN Dehua,LE Jiajin
东华大学 计算机科学与技术学院,上海 201620
College of Computer Science and Technology,Donghua University,Shanghai 201620,China

CHEN Xu,CHEN Dehua,LE Jiajin.Government information resource retrieval algorithm based on metadata semantic relevance oriented ranking.Computer Engineering and Applications,2011,47(25):121-125.

Abstract: Government information resources retrieval is important function in directory service system.Based on the XML metadata standard stipulated in national standards of *Government information resource directory system*,a keywords search algorithm is proposed,which uses the XML *TF*IDF* ranking strategy of government information resource metadata and the keywords dependence to rank the individual matches by semantic relevance.An improved keywords inverted index is proposed to improve the query efficiency.The experimental results show that this algorithm can greatly improve the rank accuracy of search results as well as the time efficiency,which can effectively improve the data-sharing ability of government information resource.

Key words: government information resource;metadata;keyword search;semantic relevance;Extensible Markup Language(XML)

摘 要:政务信息资源检索是政务信息资源共享系统的重要功能。以《政务信息资源目录体系》国家标准中的XML元数据规范为依据,提出了一种支持关键词搜索的政务信息资源检索算法。该算法使用政务信息资源XML元数据的*TF*IDF*和关键词依赖度对检索结果集进行语义相关度排序,通过改进关键词倒排索引来提高检索效率。实验表明该算法在检索结果排序精确度和时间效率上均有较大的改善,可有效提高政务信息资源利用的数据共享服务能力。

关键词:政务信息资源;元数据;关键词检索;语义相关度;可扩展标记语言(XML)
DOI:10.3778/j.issn.1002-8331.2011.25.031 **文章编号:**1002-8331(2011)25-0121-05 **文献标识码:**A **中图分类号:**TP311

1 引言

近年来,各级政府部门的电子政务信息资源建设均有长足的发展。但由于缺乏统一的政务信息管理标准和平台,信息资源在不同政府部门间很难达到互通共享,因此产生了“信息孤岛”现象[1]。当前,以元数据为基础的政务信息资源目录体系是实现政府部门间异构、分布信息资源共享互通的重要设施之一,旨在解决电子政务领域的“信息孤岛”问题。电子政务信息资源元数据是描述电子政务数据集内容的数据[2],描述政务信息资源的内容、标识方式、管理方式和获取方式等特征,通过元数据与分类表、主题词表的结合,组织信息资源分类目录、主题目录和其他目录,实现对信息资源的导航、检索、定位和交换服务[3]。

国家标准《政务信息资源目录体系》(GB/T21063)给出了核心元数据的定义及其核心特征要素,分别包含6个必选和6个可选的元数据元素。按照国家标准的规范,在抽取信息资源核心元数据的基础上,采用XML对元数据进行描述,形成政务信息资源的XML元数据[3]。图1所示为由气象局提供的有关每日天气的政务信息资源XML元数据“Daily data of weather”的例子。

```
<metadata>
    <resTitle> Daily data of weather </resTitle>
    <abstract>2004 records of weather reports
</abstract>
     <IdPoC>
        <rpOrgName> meteorologic center</rpOrgName>
     </IdPoC>
     <TpCat>
          <cateName> meteorology</cateName>
          <cateCode>11AC</cateCode>
          <cateStd>subject classification</cateStd>
     </TpCat>
     <resID>AB345/A00034VG345</resID>
     <mdId>md_001</mdId>
</metadata>
```

图1 政务信息资源XML元数据示例图

基金项目:中央高校基本科研业务费专项资金资助。

作者简介:陈旭(1987—),女,硕士研究生,主要研究方向:信息资源安全利用;陈德华(1976—),通讯作者,男,博士,副教授;乐嘉锦(1951—),男,教授,博士生导师。E-mail:chenxu_87@hotmail.com

收稿日期:2010-06-29;**修回日期:**2010-09-08

在图1中,<resTitle>实体表示该元数据的资源名称,缩略描述信息资源的标题;<abstract>实体为该元数据的资源摘要,对资源内容进行概要的文字说明;<IdPoC>实体表示资源负责方,对资源完整正确性等负责的政务部门进行说明;资源分类实体<TpCat>描述共享政务信息资源分类方式及其相应的分类信息;资源标识符实体<resID>描述信息资源的唯一不变标识编码,国标中规定了该编码的生成规则;元数据标识符实体<mdId>表示元数据的唯一标识编码,该编码依据国标生成,识别不同元数据信息;国标规定以上6个元数据实体为必选选项。关键词说明实体<DescKeys>为可选选项,描述共享政务信息资源的关键词内容及其依据。

在政务信息资源目录体系框架中,用户通过向目录查询系统提交信息资源检索请求的方式,在信息资源元数据库中查询满足检索请求的信息资源标识符及其URL,并根据URL的导引,在一定的权限范围内访问相关的信息资源。

在现有的XML文档检索方法中,关键词检索不需要用户学习任何复杂的查询语言或了解底层数据存储结构。文献[4-5]研究了XML文档的关键词搜索技术,提出了通过分析检索关键词的语义和关键词与XML文档片段相似度来提高检索的精确度和效率。本文借鉴文献[4-5]的研究成果,结合政务信息资源元数据特征提出一种基于元数据语义相关度排序的政务信息资源关键词检索算法RF-MT。该算法利用政务信息资源XML元数据的加权信息和关键词依赖度为符合条件的搜索结果集进行相关度排序,同时改进传统的关键词倒排索引结构,提高查询效率。该技术已成功应用于政务信息资源共享系统中,实验表明基于元数据语义相关度排序的政务信息资源检索算法,在准确度和时间效率上都极大提高了检索系统的有效搜索能力。

2 政务信息资源的XML元数据模型与检索模型

首先给出政务信息资源XML元数据的有序树模型表示,然后给出基于关键词的政务信息资源检索模型。

2.1 政务信息资源的XML元数据模型

通常,政务信息资源元数据的XML文档可视为有序的多叉树。一个元数据的XML文档可包含多个信息资源的元数据,其中每一条元数据均模型化为以<metadata>为根结点且有特定孩子结点(即XML标签元素)的子树。其孩子结点<mdId>唯一标识了该元数据。如图2所示为气象局的每日天气信息资源XML元数据文档,并使用dewey编码标记其结点。图2中,<metadataDB>标识XML文档名称,该文档包含所有的每日天气信息资源元数据,其中每一棵以<metadata>为根结点的子树表示一条完整的元数据信息,包含<mdId>、<resTitle>等国标规定的实体。例如图2中左边以<metadata>为根结点的子树为唯一标识编码<mdID="md_001">的政务信息元数据,对应图1的"Daily data of weather"元数据。从图2中可以看出,只有所有叶子结点即值结点才包含真正的有用信息。

2.2 政务信息资源的检索模型

在给出检索模型之前,首先给出以下两个定义:

定义1(标签路径)一个XML文档树中结点v的标签路径为从根结点到v所经过的标签列表。

例如,在图1中的类目名称"cateName"其对应的标签路径为:metadateDB/metadata/TpCat/cateName。

定义2(结点类型)有相同标签路径的结点为同一类型的结点,并使用标签名命名该结点类型。

例如,cateName类结点表示,标签路径为<metadataDB/metadata/TpCat/cateName>的所有结点的集合。国标规定,各类实体的标签名唯一,因此在政务信息资源元数据XML数据模型中,结点类型与结点名称一一对应。

下面分别从信息资源的检索请求及其结果给出政务信息资源检索模型。检索请求q为用户所指定的一组关键词$q=\{k_1,k_2,\cdots,k_n\}$,其中$k_1,k_2,\cdots,k_n$表示n个关键词。而检索的结果为一组XML元数据包含q中所有关键词$\{k_1,k_2,\cdots,k_n\}$的政务信息资源列表。如图1所示,用户输入关键词"weather reports",则返回<mdID="md_001">的政务信息资源元数据信息。

为表述方便,Metadata类结点简写为MD,MT表示一条完整的政务信息资源元数据(即MT表示一棵以MD类结点为根结点的子树)例如图1最左边MD类结点子树表示<mdId="md_001">的元数据MT。

3 政务信息资源检索的语义相关度

给定一个基于关键词的信息资源检索请求q,在政务信息资源的XML元数据文档d中,影响q中关键词的权值$W_k(d)$的因素主要有2个:

(1)q与返回结果MT的片段相似性,即XML元数据$TF*IDF$;

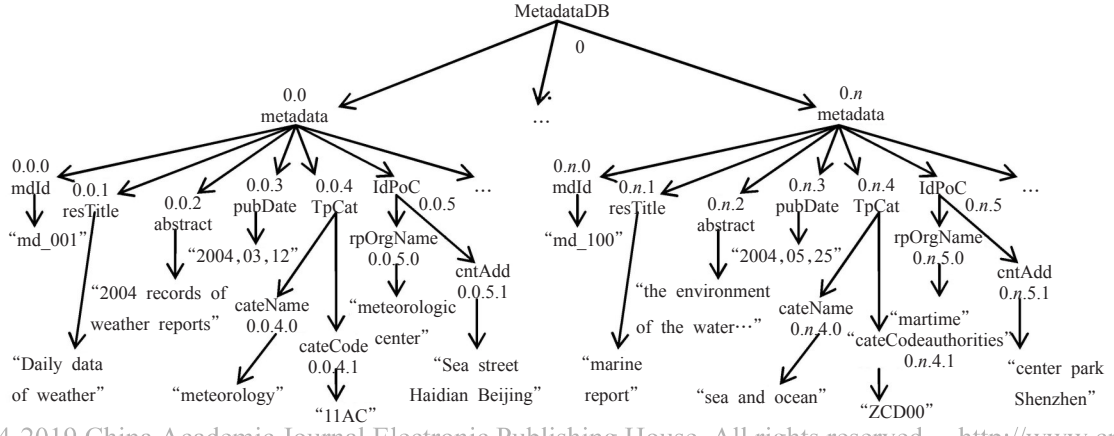

图2 XML数据模型图(使用dewey编码标记结点)

(2)q中的关键词k与标签父结点的语义相关度。

3.1　政务信息资源XML元数据*TF*IDF*

*TF*IDF*加权技术在搜索、文献分类等相关领域应用广泛,传统的基于VSM的扁平文档*TF*IDF*计算方法如式(1)[6]、式(2)[6]所示:

$$tf_{i,j}=\frac{freq_{i,j}}{\max_l freq_{l,j}} \tag{1}$$

$$idf_i=\ln\frac{N}{n_i} \tag{2}$$

*TF*IDF*加权技术首先需要确定文字单元t_i在文档d_j出现的频率$freq_{i,j}$,然后再进一步确定该文字单元在所有文档中出现频率的最大值$\max_l freq_{l,j}$,并记为n_i,即表示包含文字单元t_i的文档数目。文字单元t_i在文档d_j的权重计算如公式:$W_{i,j}=tf_{i,j}\times idf_i$。而文字单元$t_i$在检索请求$q$中的权重信息$W_{i,q}(1\le i\le m)$计算如下:$W_{i,q}=(0.5+0.5tf_{i,q})\times idf_i$。基于向量空间的距离理论,文档$d_j$与检索请求$q$之间的距离可计算如式(3)[6]:

$$sim(d_j,q)=\frac{\boldsymbol{d}_j\cdot\boldsymbol{q}}{|\boldsymbol{d}_j|\times|\boldsymbol{q}|}=\frac{\sum_{i=1}^{m}w_{i,j}\times w_{i,q}}{\sqrt{\sum_{i=1}^{m}w_{i,j}^2}\times\sqrt{\sum_{i=1}^{m}w_{i,q}^2}} \tag{3}$$

在XML元数据中,加权粒度使用结点类型代替公式(1)中的文档粒度。首先对政务信息资源元数据基础数据库定义两个统计量:

$N_{T,k}$:在其子树值结点中包含关键词k的T类结点数量。

$freq_{k,T}$:关键词k在T类结点值结点中出现的次数。

将这两个统计变量引入XML TF-IDF中:

$$tf_{k,T}=\frac{freq_{k,T}}{\max_k freq_{k,N}} \tag{4}$$

$$idf_k=\ln\frac{N_T}{N_{T,k}} \tag{5}$$

其中,$tf_{k,T}$表示关键词k在某类结点T的值结点中的词频,$\max_k freq_{k,N}$表示k在所有类型结点的值结点中出现的最大频率,idf_k表示关键词k在T中反文档频率,N_T表示DB中T类结点的总个数。

将上述XML *TF*IDF*应用到政务信息资源元数据MT,即以MT为粒度进行*TF*IDF*相似度计算,设计公式如下:

$$sim(MD,q)=\frac{\sum_{k\in q\cap MD}w_{k,MD}\times w_{k,q}}{\sqrt{\sum_{k\in MD}w_{k,MD}^2}\times\sqrt{\sum_{k\in q}w_{k,q}^2}} \tag{6}$$

其中,$w_{k,MD}=tf_{k,MD}\times idf_k$表示$k$在MD类结点中的权重,$w_{k,q}=(0.5+0.5tf_{k,q})\times idf_k$为$k$在检索请求$q$中的权重。

公式(6)表明,若政务信息元数据MT与检索请求q的相似度$sim(MD,q)$越大,则说明该元数据MT与检索请求q的相关度越大。

3.2　关键词与政务信息元数据的语义相关度

上述考虑了政务信息资源XML元数据*TF*IDF*,下面考虑检索请求q中关键词与元数据MT的语义相关度。

对于值结点的父结点T,T的值结点中包含关键词k的频率越高,说明T与k的相关度越高;若k在某类结点T的值结点中出现频率很高,可以认为当用户输入关键词k时,意图搜索其值结点中含k的T类型结点的概率更高。如图1所示,若关键词"sea"在"cateName"类结点下出现频率很高,而在"cntAdd"类结点下出现频率很低,则当q={sea}时,可以推断用户意图搜索"cateName"中包含关键词"sea"的元数据的可能性比意图搜索"cntAdd"中包含"sea"的元数据的可能性大。

定义3(依赖度)　T为叶子值的父结点,如果k在T类结点的值结点中出现,则称k依赖于T。包含k的T类结点数量越多,则说明k对T的依赖程度越大。

因此对于一个检索请求q,q对T的依赖度定义为:

$$D(q,T)=\ln(1+\sum_{k\in q}N_{T,k}) \tag{7}$$

提出政务信息资源元数据MT与检索请求q语义相关度计算的两个原则:(1)对于包含所有关键词的MT集合,若q对某个MT1中的结点T依赖度很大,则相对其他MT而言MT1与q相关度高;(2)一个MT子树中含有越多q依赖的结点,则该MT与q相关度越高。

结合上述两个影响语义相关度的因素,可得到政务信息原数据MT与检索请求q的语义相关度公式,如公式(8)所示。

$$rel(MD,q)=\begin{cases}\sum_{a\in l}\dfrac{\sum_{k\in q\cap a}w_{k,a}\times w_{k,q}}{\sqrt{\sum_{k\in l}w_{k,l}^2}\times\sqrt{\sum_{k\in q}w_{k,q}^2}}, & a\text{是值结点} \quad (8a)\\ \sum_{T\in parent(a)}rel(a,q)\times D(q,T), & T\text{是}a\text{的父结点} \quad (8b)\end{cases}$$

公式(8)的设计思路具体为:对于MT中值结点a,由于a没有更深的层次结构,即没有孩子结点,此时可退化为扁平文档的相似度计算方式$rel(a,q)$,如公式(8a)所示;然后自底向上计算该值结点a与其父结点T的依赖度$D(q,T)$,并将$rel(a,q)$$h$和$D(q,T)$作为MT与$q$语义相关度的共同衡量因子。最后,对该MT的所有值结点进行相关度累加,得到该MT与q的相似度$rel(MD,q)$。其中:$w_{k,a}=tf_{k,a}\times idf_k$,表示关键词$k$在值结点$a$中的权重,$w_{k,q}=(0.5+0.5tf_{k,q})\times idf_k$表示关键词$k$在关键词模式$q$中的权重信息。$D(q,T)$表示$q$对$T$的依赖度。

4　基于语义相关度排序的政务信息资源检索算法

在第3章提出的相似度计算公式(8)的基础上,该章给出基于语义相关度排序的政务信息资源搜索算法RF-MT。该算法以政务信息资源元数据MT为粒度,首先检索包含所有关键词的政务信息元数据集合(以唯一标识编码mdID标识元数据),再计算集合中每一个MT与检索请求q的语义相似度$rel(MD,q)$,最后按$rel(MD,q)$对MT进行语义相关度打分,并返回排序后的结果。在详细阐述RF-MT算法之前,先给出一种改进的用于确定包含关键词的文档列表的倒排索引FIL。

4.1　改进的倒排索引FIL

本文采用dewey编码方法对XML数据进行编号,如图1所示。为了提高搜索效率,提出改进的倒排索引FIL。传统的关键词倒排索引只记录出现关键词的dewey编号和位置,为更大程度地提高查询和计算效率,将关键词索引结构扩展为一个五元组FIL<keyword, mdID, DeweyID, NodeType, $freq_{k,T}$>,对应为:<关键词k,包含k的元数据唯一标识符mdID,包含k的结点T的DeweyID,Vk的结点类型,T包含k的频率>。并有相应的获取方法,如获取k的元数据标识mdID的方法FIL[K].get_mdID(),获取标识符为mdID的DeweyID的方法FIL.get_DeweyID(mdID)等。FIL索引视图如表1。

另外建立一个结点类型中包含关键词频率的索引MF<k,

$T,N_{T,k}$>,相应获取方法:getF(k).NodeType获取包含k的父结点T类型,getF(k,T).$N_{T,k}$获取包含k的T类结点个数。MF索引视图如表2所示。

表1 关键词倒排索引FIL视图

关键词	mdID	DeweyID	NodeType	$freq_{k,T}$
k_1	Md_001	0.0.0	resTitle	1
		0.0.2	abstract	2
	…	…	…	…

表2 MF索引视图

关键词	NodeType	$N_{T,k}$
k_1	resTitle	568
	cntAdd	12

4.2 RF-MT算法

RF-MT(keywords[n],FIL[n],MF[n])算法为本文的核心算法,其伪代码如下面RF-MT算法所示。该算法检索包含所有关键词的政务信息元数据,并将这些元数据按与检索请求q的相关度排序并输出。参数keywords[n]为检索请求q的关键词集合,FIL[n]为改进的倒排索引,MF[n]为结点频率索引。RF-MT搜索算法的第1至7行具体实现了如下的功能:找到包含所有关键词的元数据并插入元数据标识符列表mdIDlist中。其中,第2至3行将每个$k\in q$在倒排索引FIL中找到包含k的元数据标识mdID放入列表t_mdID[k]中;第4至7行实现找出在所有关键词k的倒排索引中都出现的元数据唯一标识mdID,即找出包含所有关键词的元数据唯一标识,并插入结果列表mdIDlist中。其中mdID.getShard()方法实现判断该mdID是否出现在所有MD_mdID列表其他成员中。t_mdID[k]为包含关键词k的元数据唯一标识符mdID列表。第9至19行实现为每一个结果元数据MT与检索请求q的相关度进行打分,其中第11至18行表示对每一条结果元数据,对照关键词倒排索引,对该元数据中的所有值结点调用获取相关度函数getRelavancy(),计算所有值结点相关度的总和,即为该元数据MT与q的相关度分数,并按分数从高到低插入分数列表score_MDdewey中。

RF-MT(keywords[n],FIL[n])搜索算法:

```
1  LinkedList mdIDlist,t_mdID[],MD_mdID
2  For each k∈q do
3      t_mdID[k]=FIL[K].get_mdID();
4  MD_mdID=getShortest(t_mdID[k])
5  for each mdID∈MD_mdID
6      If(mdID.getShard())
7          mdIDlist.insert(mdID);
8  Return mdIDlist;
//为结果元数据与q的相关度打分,并按顺序插入打分列表
9  LinkedList scoreList
10 mdID=getNextmdID()
11 While(!end(FIL.get_DeweyID(mdID)))
12     Node T=getMinNode(FIL.get_DeweyID(mdID))
13     If(T !=leafNode) then
14         Rel(T,q,mdID)=0
15     Else if(T is leafNode) then
16         Rel(T,q,mdID)=getRelavancy(T,q,mdID)
17     Score_mdID+=rel(T,q,mdID)
18     scoreList.insert(mdID,Score_mdID)
19 mdID=getNextmdID()
20 Return scoreList;
```

在RF-MT算法中用于获取相似度的函数getRelavancy(Node a,query q,mdID)其伪代码如下面getRelavancy(a,q,mdID)函数所示。该函数使用公式(8)来计算该MT相对于检索请求q的相似度。在算法中,第1至7行使用公式(8a)计算值结点a与q的相似度;第8行使用公式(7)计算该值结点a与其父结点的依赖度;第9至10行计算所有值结点与q的相关度且返回结果。

```
Function getRelavancy(Node a,query q,mdID)
1  For each(k∈q∩a) do
2      tf_{k,a}=freq_{k,a}/max_k freq_{k,N}
3      idf_k=ln(N_a/N_{a,k})
4      w_{k,a}=tf_{k,a}×idf_k
5      w_{k,q}=(0.5+0.5tf_{k,q})×idf_k
6      Sum+=w_{k,a}*w_{k,q}
7      Rel(a,q)=sum/(sqrt(w_{k,a}^2)×sqrt(w_{k,q}^2))
8      D(q,parent(T))=ln(1+getIN(k,parent(T)).N_{T,k}
9      Rel(a,q,mdID)+=Rel(a,q)*D(q,parent(T))
10 Return Rel(a,q,mdID)
```

为了在后续的实验部分对RF-MT算法进行相关度排序效果的检验,根据公式(6)提出一种基于政务信息元数据的纯XML *TF*IDF*相似度打分方法为符合查询条件的元数据进行排序的算法F-MT。该算法与RF-MT类似,只需要删除getRelavancy(Node a,query q,mdID)方法中的8、9两行代码,这样就将给予语义相关的XML *TF*IDF*简化成为纯XML *TF*IDF*相似度,并将此算法命名为F-MT。

5 实验

为了测试本文提出的基于元数据语义相关度排序的政务信息资源检索算法RF-MT的性能,进行了实验分析。实验使用上海浦东数据中心水务、气象等5个部门的220 MB的政务信息,含约260 000条政务信息元数据。实验机器配置是P4 3.0 GHz,1.5 GB内存。

首先,检验本文算法RF-MT的相关度排序性能。测试比较XQuery[7]、F-MT算法和RF-MT算法的Top-20精确度和Top-20查全率,计算方式如下:

Top-20精确度:检索的前20条记录中与查询意图相关的记录数量/20;Top-20查全率:检索的前20条记录中与查询意图相关的记录数量/所有相关记录数量。

随机给出10个关键词查询Q1~Q10,Q平均关键词个数为3个(XQuery使用相应的查询语法),提交系统进行搜索。从图3和图4给出的对比结果可以看出,F-MT算法Top-20精确度和查全率都比XQuery要高,这是因为F-MT算法相比XQuery考虑了XML *TF*IDF*,因此返回结果的匹配度更高;RF-MT算法在F-MT算法基础上进一步考虑关键词对其父结点的依赖度,因此按语义相关度排序后的Top-20结果有更高的精确度和查全率。

由此分析,结构化查询语言按严格语法返回特定结果,使用不便且易造成返回结果不全。元数据XML *TF*IDF*方法只考虑了词在政务信息元数据中的统计特性,这种方法不需要

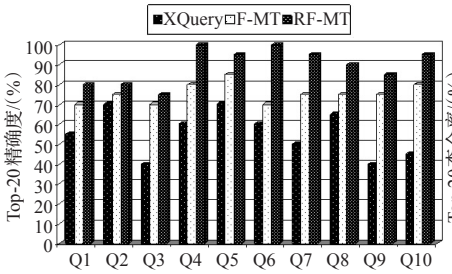

图3　Top-20精确度对比图

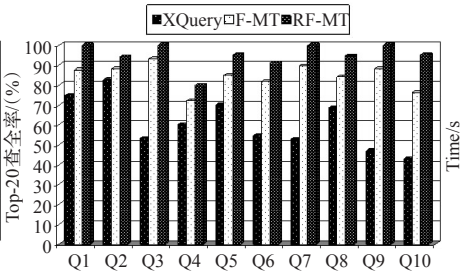

图4　Top-20查全率对比图

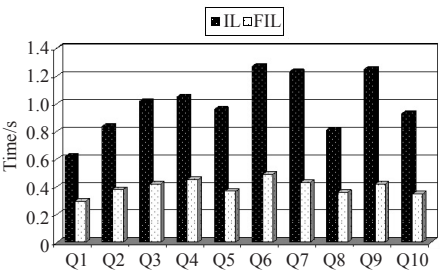

图5　查询总耗时对比图

任何对文本内容的深层理解,但是在含有语义的XML标签中,不考虑词本身信息及元数据语义特征的*TF*IDF*方法往往不能达到预期的效果。而使用XML *TF*IDF*+关键词k对其父结点的依赖度来共同衡量q与结果的语义相关度,在输出结果的Top-20里,语义匹配度明显提高。

其次,比较传统关键词倒排索引IL和改进的关键词倒排索引FIL在搜索中的时间效率。以上述的Q1~Q10为输入,图5给出了IL和FIL两种索引结构在搜索中消耗的总体时间。从图5中可以看出,使用FIL索引大大减少了搜索时间,有效提高了查询效率。

最后,测试比较查询中关键词的个数对搜索时间的影响。分别选定2~8个关键词的查询20次,分别计算IL和FIL的平均查询时间。图6横轴代表关键词个数,从图6中可以看出,在关键词个数2~5之间时呈上升趋势,这是因为索引查找时间消耗上升,而在关键词个数超过5个以后查询时间趋于平缓,这是由于符合条件的政务信息元数据较少,RF-MT排序算法耗时下降。

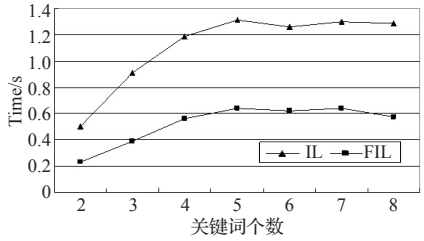

图6　查询耗时随关键词个数变化图

6　结束语

结合政务信息资源元数据特点,通过将政务信息元数据的加权值和关键词依赖度作为进行政务信息元数据与查询q语义相关度打分的共同衡量因素,提出RF-MT算法,并将该算法与政务信息元数据加权值排序算法F-MT进行对比。实验表明,RF-MT该算法在返回结果Top-20精确率和查全率都在85%以上,且使用改进的关键词倒排索引FIL,大大提高了搜索的时间效率,弥补了目录检索返回结果数量大、结果无序的不足,满足用户对海量政务信息资源的快速获取与定位需求。

参考文献:

[1] 华宇清.标准:消除信息孤岛[J].上海标准化,2004(6).
[2] 王仁武,杨洪山,陈家训.电子政务信息资源元数据标准的设计与实现[J].情报资料工作,2007(4).
[3] 中国国家标准化委员会.中华人民共和国国家标准GB/T21063[S].2007.
[4] Liu Z Y,Chen Y.Identifying meaningful return information for XML keyword query[C]//Chan C Y,Ooi B C,Zhou A Y,et al. Proc of the 2007 ACM SIGMOD Int'l Conf on Management of Data(SIGMOD).Beijing:ACM Press,2007:329-340.
[5] Bao Zhifeng,Ling T W,Chen Jiaheng.Effective XML keyword search with relevance oriented ranking[C]//International Conference on Data Engineering(ICDE),Shanghai,2009.
[6] 孔令波,王腾蛟,高军.XML数据的查询技术[J].软件学报,2007,18(6):1400-1418.
[7] Florescu D,Kossmann D,Manolescu I.Integrating keyword search into XML query processing[J].Computer Networks,2000,33(1/6):119-135.

(上接111页)

构成完整的SAFE系统。

参考文献:

[1] 伏晓,蔡圣闻,谢立.网络安全管理技术研究[J].计算机科学,2009(2):15-19.
[2] 郭山清,阳雪林.安全报警事件关联算法研究[J].计算机应用,2005(10):2277-2279.
[3] Siraj A,Vaughn R B,Bridges S M.Intrusion sensor data fusion in an intelligent intrusion detection system arthitecture[C]//Proceedings of the 37th Hawaii International Conference on System Sciences,2004.
[4] 张慧敏,钱亦萍,郑庆华.集成化网络安全监控平台的研究与实现[J].通信学报,2003(7):155-163.
[5] Thomas C,Balakrishnan N.Improvement in intrusion detection with advances in sensor fusion[J].IEEE Transactions on Information Forensics and Security,2009,4.
[6] 张民,罗光春.基于IDMEF的大规模协同IDS架构[J].电子科技大学学报,2009(2):258-261.
[7] 郭帆,叶继华,余敏.基于IDMEF和分类的报警聚合[J].计算机应用,2008(1):250-253.
[8] 霍书全.一个多值逻辑的一阶谓词系统[J].逻辑学研究,2009(1):78-89.