

Web资源的多粒度语义标注及其应用技术研究

朱嘉贤¹ 白伟华¹ 李吉桂²

(肇庆学院计算机学院 肇庆 526061)¹ (华南师范大学计算机学院 广州 510631)²

摘 要 当前的 Web 搜索引擎获得的搜索结果都是基于关键字标注的 Web 文档、页面或链接,不支持对文档内部信息的检索。为支持 Web 资源内部信息的检索,研究多粒度语义标注,即按树根结点、分支结点、叶子结点及资源信息元为粒度单位对 Web 资源进行组织管理,并在此基础上探讨基于本体的搜索技术。初步的分析和实验表明,这样可以提高从形式多样的海量 Web 资源中获取所需信息的效率。

关键词 语义标注, Web 资源, 资源信息元, 本体, 多粒度

中图法分类号 TP393.09 文献标识码 A

Research on the Multi-granularity Semantic Annotation of Web Resource and its Application Technology

ZHU Jia-xian¹ BAI Wei-hua¹ LI Ji-gui²

(School of Computer Science, Zhaoqing University, Zhaoqing 526061, China)¹

(Faculty of Computer Science, South China Normal University, Guangzhou 510631, China)²

Abstract The search results from the present Web search engine are all the Web files, Web pages or the links annotated based on keywords, and it does not support the inner information searching of these files. In order to support the inner searching of a file, it researched on the multi-granularity semantic annotation of Web resource, which named the root node, the branch node, the leaf node and the resource information metadata, to manage the Web resource. Based on it, it also researched the searching technology based on ontology, and by the primary experiment and analyses, it shows that it can enhance the efficiency of searching and getting information data from the multifarious and great deal of Web resources.

Keywords Semantic annotation, Web resource, Resource information metadata, Ontology, Multi-granularity

1 引言

Web 资源形式多样,它表现为网页文字、图形图像、音频、视频等,其数据以各类文件的形式被存储。随着海量存储技术的飞速发展,海量资源的多样性给资源的管理带来了以下几个问题^[1-3]:(1)资源的分类管理;(2)资源的搜索、应用和更新;(3)资源的重新整合;(4)资源的共享。为了能对 Web 资源内部信息进行多粒度的语义标注,支持对文档内部信息的检索,本文研究按树根结点、分支结点、叶子结点及资源信息元为粒度单位,即按多粒度进行 Web 资源的语义标注;同时还探讨了基于本体的搜索技术,即基于粒度匹配计算的资源搜索算法,其根据被检索资源和用户查询之间的相似度实现对资源的检索。

2 信息资源的多粒度语义标注

语义标注的对象可以是一个词组、一段文字、一幅图片、一个数据表格或者是整个文档。利用语义网^[4]的相关技术 XML、RDF、RDF Schema 和本体^[5]的应用将现有的信息资源通过多粒度语义标注形成树形结构的资源库,以方便资源的分类、搜索,并可满足业务流、协同工作的需求。

2.1 语义标注现状

语义标注及其研究^[6-10]受到了广泛的关注,简单而言,语义标注是指利用本体技术将信息资源(文字信息、图、表、视频、程序代码段等)中的概念、属性、关系、型(信息资源类的数据的结构和属性的说明)、域和值标注为计算机可识别的元数据,实现标注与信息资源的关联,它是进行语义推理的基础。

目前,语义标注的研究主要是针对标注方法及过程的研究——主要分类有手工标注和半/全自动标注^[9-14],而有关多粒度标注及其应用的研究还未引起关注。

一方面,大粒度标注——基于关键字的 Web 搜索引擎(获取 Web 资源的主要途径),缺乏对语义纵向标注的支持,Web 资源内部所蕴含的大量信息无法被进一步检索、过滤和提取。使得大量涵盖了底层资源的文档(如 Word 文档、Excel 文档、PPT 文档、PDF 文档、软件源代码文档等)都不能被充分利用。另一方面,虽然资源被半/全自动标注,但仍无法很好地被共享、检索和利用。因此,应按不同的标注粒度来描述资源,使之易于分类和共享,并将基于各粒度单位的相似度计算的搜索算法用于用户查询和检索资源。

2.2 信息资源管理的树形结构组织描述

按照应用需求的方式进行整理,常见资源的语义信息呈

到稿日期:2010-09-21 返修日期:2010-11-30 本文受广东省科技计划项目(2010B060100005),肇庆市科技创新计划项目(2010G201)资助。

朱嘉贤(1977—),女,讲师,主要研究方向为 Web Service、Web 语义、电子商务, E-mail: zhujiaxian@zqu.edu.cn; 白伟华(1976—),男,副教授,主要研究方向为计算机网络、电子商务; 李吉桂(1942—),男,教授,硕士生导师,主要研究方向为计算机网络、电子商务、多媒体技术。

现出树形的管理组织结构,也称为信息资源结构树:例如在电子商务中,双方的协商业务及过程包括:协商的产品信息、协商双方信息(买方/需求方,卖方/供应方)、协商提议及其议题、反提议及其议题、协商业务接口等多个实体、关系、语义信息;在图书/文档知识的组织结构上包括:文档标题、知识大纲、章节标题、内容信息、数据类型、具体文档数据等^[15];教育领域中各类课件的科目分类、数据存储、知识点的定义、知识点之间的从属关系等。将协商信息按树形组织结构进行规划,其呈现出的协商业务语义信息树形结构逻辑图如图1所示。

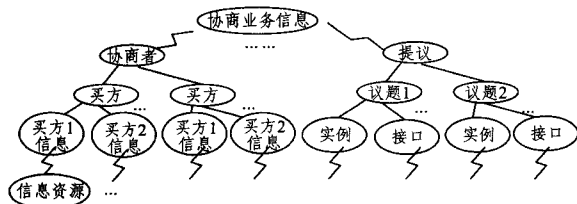


图1 协商业务语义信息树形结构逻辑图

由图1可知,在协商的过程中,对协商业务的管理、分类等过程都可以通过基于信息资源的树形结构来进行组织和管理。

在资源的管理和应用的过程中,基于树形结构的语义信息组织管理方式有以下几个特点:

(1) 树形结构有利于对资源的多粒度语义标注,不同的粒度单位对应于树结构中各类不同的结点;

(2) 树形组织结构实质是一类简单的本体。在逻辑上与XML、RDF、RDF Schema等语义网中的关键技术对资源的描述相一致,有利于对资源的管理、维护和应用的过程也用XML、RDF或RDF Schema来进行描述;

(3) 树形结构容易实现结点的动态性操作,有利于实现资源管理中各类的操作而不影响结点间的逻辑关系;

(4) 树形的管理组织结构有利于表示资源与描述之间的逻辑从属关系,能方便地实现将资源的分类、建立索引和搜索的操作整合在资源的更新和插入的过程中,以提高对资源进行整体管理的性能。

2.3 语义标注中各粒度结点的定义

用户的查询请求和Web资源可通过标注粒度来描述,粗粒度标注是对整个Web页面或一个文件进行标注,细粒度标注是在文档内部利用树结构中的各类结点——树根结点、分支结点和叶子结点为单位对词组、术语、段落、表格、图标等进行标注,可按不同的需求进行匹配计算,实现搜索。

将资源的管理组织映射成树形的逻辑结构,能实现资源物理存储的逻辑独立性——树结构的改变不影响资源的物理存储。为实现资源物理存储的逻辑独立性,按语义标注呈现出的不同语义信息对应树结构的3类结点——根结点、分支结点及叶子结点,其结构逻辑图及文件映射示意图如图2所示。

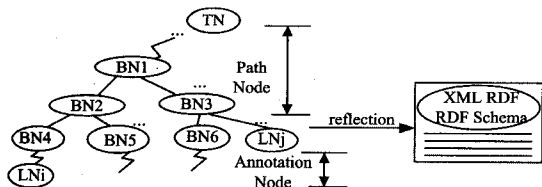


图2 资源管理树及文件映射示意图

(1) 根结点 TN——存储资源树的相关属性(应用领域、

作者信息、资源类型集信息等)及结构信息;

(2) 分支结点 BN——根结点的派生结点(继承了根结点的属性),既可以作为分支结点也可作为路径结点,存储资源的分类属性(分类关键字、索引信息、资源类型集信息等)、结点间的逻辑关系(双亲关系、父子关系)、结点类型;

(3) 叶子结点 LN——分支结点的孩子结点,是分支结点的派生结点,是资源的语义信息映射结点,存储资源的物理存储位置信息、资源的获取接口信息等。

通过上述方式,首先,对资源的存储管理不会影响树的逻辑结构,只需修改/更新或添加叶子结点中资源位置的信息即可;其次,资源管理结构能被方便地映射为XML、RDF或RDF Schema,能方便各用户根据自己对资源的应用构建自己的资源管理结构树,也可以根据应用将相同的资源映射成不同的资源管理结构树;最后,用户只需根据自己的应用维护各自的资源管理树,有利于各用户间共享资源管理树。资源存储的物理独立性也有利于实现资源管理及应用的虚拟化。

2.4 Web资源内部信息的语义标注

结合对多种文档内部涵盖信息的分析,根据Web资源涵盖信息的相互关系,语义标注主要通过文档本体(Document Ontology) $Doc = \{d_{id}, d_t, f_t, d_u, r_u, r_m, d_{des}, d_{info}\}$ 、Web资源本体(Web Resource Ontology) $Wr = \{r_{id}, r_n, r_t, r_d, r_{des}, r_{fid}, r_{info}\}$ 和文档类型本体(Document Type Ontology) $Dt = \{f_{id}, f_{hn}, f_{finfo}\}$ ^[14],在语义网中标识Web资源数据,其关系模型图如图3所示。Web资源表现为各种类型的文档,一个文档则蕴含了一个或多个信息资源,因此利用文档本体模型描述Web资源数据源——文档;用文档类型本体模型描述文档的格式,以方便获取Web资源主体信息;用Web资源本体模型描述主体信息,方便利用数据中介服务^[16]实现资源注册/发布及查询。

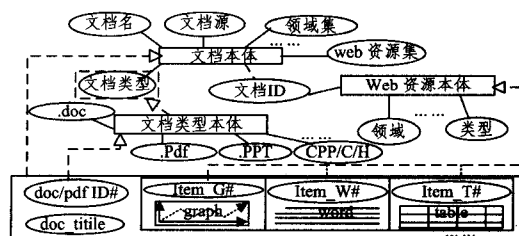


图3 Web资源信息语义标注模型

根据Web资源、文档和数据之间的关系,可以建立资源类的OWL类图(如图4^[16]所示),从而能更方便地利用相似度计算及本体推理功能来实现Web资源的搜索。

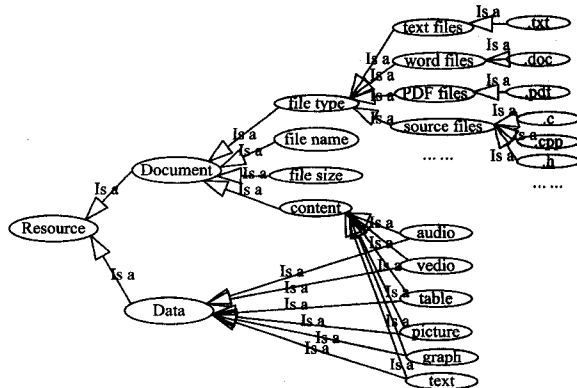


图4 Web资源类的OWL类图

3 多粒度语义标注的应用框架

3.1 应用框架结构

资源信息元是指被标注后的 Web 资源所涵盖的语义信息及数据的物理存储、数据格式所对应的信息,是资源的标注及搜索的基本粒度单位。

根据 Web 资源内部信息的语义标注方式对 Web 资源数据拟定细粒度的信息元本体 IO,配合树形管理结构实现资源的细粒度应用。

利用数据中介服务,基于多粒度语义标注的资源的管理应用框架结构中主要包含的两个核心层分别是:数据中介服务层和资源应用层;3 类数据库分别是:资源标识库、系统 RDF/RDFS 文件镜像库和用户个性化资源库。被标注后资源的管理应用框架结构逻辑图如图 5 所示。

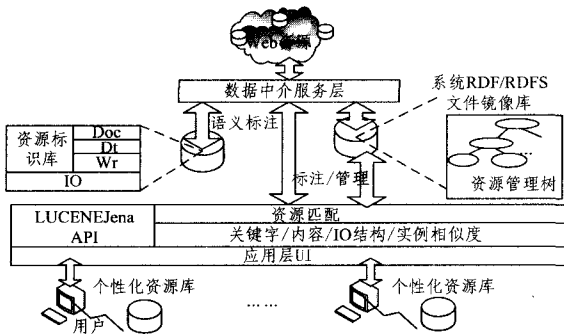


图 5 资源管理的应用框架结构逻辑图

数据中介服务层——主要完成对资源的语义标注,并提供查询/检索和获取资源的服务。在原数据中介数据服务(Web 资源的发布服务 $Ws_upService$ ——更新 Web 资源目录 UDDI、查询/搜索 Web 资源服务 $Ws_queryService$ 、获取 Web 资源服务 $Ws_getService$ ^[16])的基础上,修改了服务 $Ws_upService$,完成构建标注资源的 Doc 本体、Dt 本体和 Wr 本体后,创建资源信息元,使上述 3 类本体作为资源信息元本体的属性集。

资源应用层——主要的作用是通过使用 Lucene、Jena API 建立 RDF/RDFS 文件索引、查询资源以及基于 XML 结点操作等手段完成资源管理树的创建、维护和插入资源信息的操作。另一方面,集成了应用层的用户界面,通过设定的资源查询/检索、获取、资源入库等操作原语完成用户构建个性化资源库,并将用户对应的资源管理树的信息注册在系统 RDF/RDFS 文件镜像库中,实现用户间在资源应用上的共享。

资源标识库及系统 RDF/RDFS 文件镜像数据库作为局部或全局性的共享信息资源,为资源检索和管理过程中涉及的资源匹配提供语义参照,为数据服务中介服务层在获取资源时提供资源 URI 信息;用户的个性化资源库一方面是属于资源应用模型中的资源数据源,另一方面用户可对自己应用中所需的资源(本地原有的资源或从通过数据中介服务层获取得到的资源)构建对应的资源管理树,并可将该树注册到系统的 RDF/RDFS 文件镜像库中。

3.2 资源管理及存储逻辑结构

在基于多粒度语义标注的资源管理应用框架中,对资源的管理及组织方式形成了资源管理树的组织结构,结合拟定的资源管理树以及资源信息元形成由资源库层、本体模型层、

RDF 模型数据以及 RDFS 元模型构成的资源管理及存储结合的逻辑模型。其逻辑结构如图 6 所示。

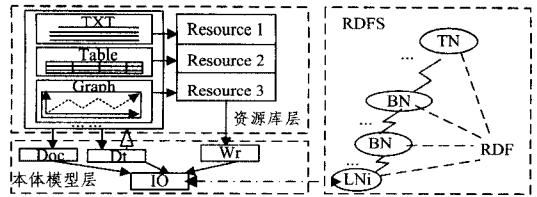


图 6 资源管理及存储逻辑结构模型图

(1)本体模型层是信息提取及完成语义标注的核心层,是具体资源获取及搜索的求解层。按照资源信息元本体的结构完成对具体的数据、资源的语义标注,并根据资源的特征、应用领域、关键字标识和用户的个性化需求将构建出来的资源信息元插入/接入资源管理树的某叶子结点或接入到某分支结点属下新创建的叶子结点上。随着被语义标注的资源增多,形成资源管理树。

(2)资源库层是资源的存储层,资源的分布及存储对使用者/请求者来说可以是透明的,可使得分布在不同用户、不同地理区域的各类数据资源借助于资源信息元本体中定义的接口信息来实现对资源的虚拟化管理。

(3)资源管理树的每个结点通过 RDF 进行描述,RDF 描述的三元组模型,一方面能有效地描述出上下层的逻辑结构;另一方面借助于三元组模型所表示的命题来实现管理过程中的各种推理,提高资源的查询/搜索的效率。

(4)利用 RDFS 描述类和属性(二元关系)、值域和定义域在属性上的约束以及子类和子属性的包孕关系的功能,资源管理树的框架结构通过 RDFS 进行描述,以完成资源管理这一个过程的对象建模。

4 基于粒度匹配计算的资源搜索应用技术

资源搜索应用技术最关键的问题就是在被标注后的数据中查询/搜索所需的信息资源,本文主要通过基于粒度的相似度计算的方式来实现查询请求与被检索资源之间的匹配。

4.1 资源搜索请求过程

资源应用层响应用户的各类操作要求——查询/检索资源信息、获取资源、发布/共享资源、调整资源管理树或建立资源管理树等,在完成对资源的语义标注后,利用数据库技术及 Lucene API 实现对 RDF/RDFS 文件的管理;利用基于各粒度的本体相似度的计算及知识推理技术实现对资源管理树各结点的查询以及资源的搜索匹配。

资源搜索的基本过程如图 7 所示。

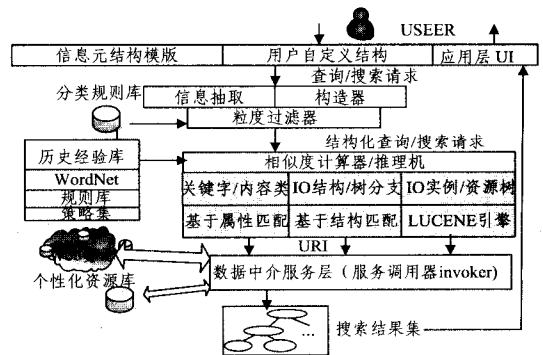


图 7 资源搜索基本过程示意图

为实现搜索,将用户的所有查询/搜索请求看成—个待匹配的资源信息元,其基本过程:①用户使用应用层的 UI,使用资源信息元结构模版或按自己定义的结构提出查询/搜索请求;②经用户层 UI 输入的查询/搜索请求都是结构化信息,故能方便地对其进行信息抽取,然后构造器将请求按资源信息元的模版完成结构化的转换;③根据被结构化的请求,用粒度过滤器的分类规则对搜索请求按相似度计算的匹配策略分类;④按分类结果启动相似度计算器/推理机调用对应的匹配策略进行资源搜索匹配;⑤获得资源的 URI 后,使用数据中介服务层的服务调用器 Invoker 从 Web 资源或个性化资源库中获取相关的信息资源,然后将返回的搜索结果集反馈到应用层 UI。

4.2 基于粒度匹配搜索的主要算法

在将资源信息元挂入资源管理树的叶子节点的过程中,在搜索资源的过程中以及在创建、获取资源管理树的过程中都必须先完成基于粒度的匹配计算,该匹配计算主要通过相似度计算来实现。由资源管理及存储逻辑结构模型图(见图 6)可知,多粒度的语义标注对应多种结构的结点和多种类型的本体模型,决定了在基于粒度的相似度计算中应采取不同的匹配策略,使用不同的相似度计算函数。

设资源请求所拟定的本体对象或待挂入资源管理树的本体对象为: IO_i , 已存在的资源本体对象为: IO_e , 在计算公式中本体中所蕴含的属性等通过 IO_i , IO_e 进行标识。

(1) 基于属性的匹配策略

针对属性的匹配主要是叶子节点中关联的资源与待关联的资源之间的相似匹配。基于属性的匹配策略主要分两步完成:首先,通过计算确定两属性的描述一致性;其次,计算属性取值的相似度。

关联的两资源匹配算法描述如下:

①读资源管理树中取待检查的叶子节点 LN 关联的资源信息元 IO_s 放入 IO_s_SET 中,获取该集合中每元素的属性集 PIO_s ;

②读取待审资源本体对象 IO_i 属性集 PIO_i 中的文档物理信息 G_{io} 、文档类型信息 F_t 及对应的属性名 P_{gn} 集;

③属性名 w_i 通过基于 Wordnet 词相似度的计算公式(式(1)) $\text{Sim}(w_i, w_e)$ 得到:

$$\text{Sim}(w_i, w_e) = \frac{(d(A) * (S(w_i) + S(w_e)))}{(d(A) * (S(w_i) + S(w_e)) - d(Aw_i) * S(w_e) + (Aw_e) * S(w_i))}^{\delta}$$

$$\delta = \log(10 \times \max(d(Aw_i), d(Aw_e))) \quad (1)$$

式中, $d(A)$ 为 WordNet 构成的各词关系网状结构图中的词 w_i, w_e 最近公共祖先 A 到最早祖先 Entity 的距离; $d(Aw_i)$ 和 $d(Aw_e)$ 分别为两词最近公共祖先 A 到词 w_i, w_e 的距离; $S(w_i)$ 和 $S(w_e)$ 分别为词 w_i, w_e 具有 Sense 的个数即 $|Sense(w)|$; $\delta = \log(10 \times \max(d(Aw_i), d(Aw_e)))$ 。

④利用规则 Rule1, if $\text{Sim}(w_i, w_e) > \text{可信度阈值 } V_Sim$, 则获取对应属性的取值 v_i , 值域 d_i , 数据类型 t_i , 转步骤⑤, 否则, 转步骤③;

⑤计算 VP_i 与 VP_e 的相似度(式(2)) $\text{Sim}(VP_i, VP_e)$:

$$\text{Sim}(VP_i, VP_e) = \begin{cases} 1 & \text{若 } VP_i = VP_e \text{ OR } IO_i_G_{io} = IO_e_G_{io} \text{ 表示为相同的两资源} \\ C(t_i, t_e) + C(d_i, d_e) + C(v_i, v_e) * 1/3 & \text{其他情况} \end{cases} \quad (2)$$

式中, 当属性 P_i, P_e 全匹配相容时即有 $VP_i = VP_e$ OR $IO_i_G_{io} = IO_e_G_{io}$, 表示待审资源本体对象(即资源请求所拟定的本体对象或待挂入资源管理树的本体对象)与已存在的资源本体为同一资源。例如: URI 完全一致的资源或文件属性(名字、类型、大小等)完全相同的两资源, 实质是同一资源。否则, 通过数据类型 t_i, t_e 和取值 v_i, v_e 的相容匹配函数 $C(t_i, t_e), C(d_i, d_e)$ 和 $C(v_i, v_e)$ 计算属性取值的可信度。相容匹配函数 $C(i, j)$ 为判断参数 i, j 是否出现在同一类型的集合上的一个集合代数运算。

⑥根据计算得到的 $\text{Sim}(VP_i, VP_e)$ 结果值, 调整相关资源管理树中已存在的资源所关联的叶子结点中聚类可信度值, 使得相似度大于系统设定的可信度阈值聚集在同一个或相邻叶子结点中, 实现资源按可信度的聚类。返回计算值及叶子结点 ID, 算法结束。

(2) 基于结构的匹配策略

基于结构的匹配主要是针对资源信息元的结构匹配以及针对资源管理树的树结构以及分支结构在计算属性相关性的基础上的匹配策略。

资源信息元的结构匹配算法描述如下:

①根据分类关键字 Key , 计算 Key 对应的树根结点 $TNid$, 若 $Key \notin Key_Set$ 集合, 则使用(式(1)) $\text{Sim}(Key_i, Key_e)$, 若 $\text{Sim}(Key_i, Key_e) > V_Sim$, 则将 $TNid$ 放入候选查询集 $CaSet$ 中对满足条件的资源管理树使用递归算法 $Find_DBT(Key_e)$ 查询各分支, 按分支上的路径定位到资源管理树的叶子结点上;

②判断操作类别, 若为资源信息元聚类操作或有新的信息元插入资源管理树时, 则将待调整的资源信息元本体放入 IO_s_SET ; 若为资源检索操作, 则将资源检索请求对照的资源信息元本体放入 IO_s_SET ;

③根据叶子结点中用户自定义领域标准资源信息元本体结构及 IO_s_SET 中的 IO 本体结构分别构建属性集 PS_e 和 PS_i ;

④按 PS_e 和 PS_i 构建 $PS_e \times PS_i$ 的计算矩阵 $PsimA = [PS_e | \times | PS_i]$, 其中, 元素 $PsimA_{ij} \in [0, 1]$, 取值计算为(式(3)):

$$PsimA_{ij} = \begin{cases} 1, & \text{若 } PS_e RPS_{ij} R \in IO(R) \\ \text{Sim}(PS_e, RPS_{ij}), & \text{其他} \end{cases} \quad (3)$$

⑤属性匹配推理: 按行选取每行中最大元素值 $\text{Max}\{PsimA_{ij}\}$, 应用规则 Rule1: $F(P_i') = \text{Max}\{PsimA_{ij}\}$, if $F(P_i') > VP_i \rightarrow (P_i' \in P)$; 对应 PS_i 中属性参与整体匹配计算;

⑥资源信息元匹配推理: 对所有参与整体匹配计算的属性利用式(4)

$$\sum f(VP_i, WP_i) = \sum_{i=1}^{|PS_i|} \text{Max}_{j=1}^{PS_{ij}} (PsimA_{ij}) * WP_i \quad (4)$$

应用规则 Rule2: if $\sum F(P_i') WP_i = \sum f(VP_i, WP_i) \geq IV_{Pi} \rightarrow (IO \equiv IO) \wedge IV_{Pi} \geq \sum WP_i VP_i$; 将满足条件的资源信息元本体放入结果集 $ResultSet$ 中;

⑦向资源应用层返回结果集 $ResultSet$ 并调整 $ResultSet$ 中所关联的资源管理树的叶子结点的聚类结果, 调整资源信息元关联的叶子结点的位置。算法结束。

资源管理树的树结构以及分支结构匹配算法描述如下:

- ①计算两资源管理树的树根结点中包含的 *Key* 相似度,利用 Wordnet 中词语的关系结构标识对应结点的位置关系;
- ②根据 *Key* 相似度,调整结点中 *Key* 的集合,使得两结点的 *Key* 集合映射到同一个 Wordnet 词集中;
- ③由于资源管理树被存储为 RDF/RDFS 格式,利用 Lucene 对文本的搜索能力,利用 Jena 的推理,使用 Jena API 进行判断;
- ④判断对应两树结点的关系 *R*,若 $R \in \{\text{Attribute-of, Part-of, Instance-of, Kind-of}\}$,则建立两树结构点的关系映射说明,形成树结构结点关系映射集 RelSet;
- ⑤根据关系映射集 RelSet,通过树的深度遍历判断两资源管理树的相似度;
- ⑥返回相似度大于可信度阈值的分支结构及关系映射集 RelSet。算法结束。

4.3 RDF/RDFS 文件的管理

资源管理应用中,用户通过管理 RDF/RDFS 文件实现了资源应用和共享,也实现了资源存储及使用的虚拟化。在实现资源语义标注的过程中,使用 Protégé 作为建模工具,得到 Doc 本体、Wr 本体和 Dt 本体的 OWL 文件,完成了对资源信息元核心属性的建立,然后配合文本标注和 XML Schema 使用 Jena RDF API 完成了利用 RDF/RDFS 对资源管理树的描述并形成相应的文件,并以 RDF 三元组的形式存储在数据库中。

通过 Lucene 读取文件的接口,为描述资源管理树的 RDF/RDFS 建立关键字标识号的排列索引,根据该排列索引中的关键字,调整 RDF/RDFS 对应的资源管理树中从根结点到各叶子结点路径所包含的各结点的关键字集,以完成资源管理树的初始化工作。

先按需求完成资源本体的相似度计算,通过 Jena RDF API 以及基于 XML 操作的控件的功能,实现调整、插入、合并结点的过程,即完成对资源管理树(RDF/RDFS 文件)的管理。

结束语 如上所述,本文主要研究了 Web 资源的多粒度语义标注,并给出了基于粒度匹配计算的搜索应用技术的相关算法。从以上分析可知:这种处理方法的效率是肯定的。然而,至于如何考虑标注的开销,和在“查准”与“查全”两方面做出权衡,限于文章篇幅留待下一步探讨。

参 考 文 献

- [1] Pirro G, Mastroianni C, Talia D. A Framework for distributed knowledge management: Design and implementation[J]. Future Generation Computer Systems, 2010, 26: 38-49
- [2] Han K H, Park J W. Process-centered knowledge model and enterprise ontology for the development of knowledge management system[J]. Expert Systems with Applications, 2009, 36: 7441-7447
- [3] Jung J J. Ontology mapping composition for query transformation on distributed environments[J]. Expert Systems with Applications, 2010, 37: 8401-8405
- [4] Kiryakov A, Popov B, Terziev I, et al. Semantic annotation, indexing, and retrieval[J]. Journal of Web Semantics, 2004, 2(1): 49-79
- [5] Dean M, Schreiber G, Bechhofer S, et al. OWL web ontology language reference [EB/OL]. W3C Recommendation. <http://www.w3.org/TR/2004/REC-owl-ref-20040210/>, 2004
- [6] Alani H, Kim S, Millard D, et al. Automatic ontology-based knowledge extraction from Web documents[J]. Intelligent Systems, 2003, 18(1): 14-21
- [7] Jane M, Christopher M, Fernando G, et al. Semantic Annotation of Aerospace Problem Reports to Support Text Mining[J]. IEEE Intelligent Systems, 2010, 25(5): 20-26
- [8] Alcaraz Calero J M, Marín Pérez J M, Bernabé J B, et al. Detection of semantic conflicts in ontology and rule-based information systems[J]. Data and Knowledge Engineering, 2010, 69: 1117-1137
- [9] Jung J J. Reusing ontology mappings for query routing in semantic peer-to-peer environment [J]. Information Sciences, 2010, 180: 3248-3257
- [10] Ahn J W, Brusilovsky P, Grady J, et al. Semantic annotation based exploratory search for information analysts[J]. Information Processing and Management, 2010, 46(4): 383-402
- [11] 陈叶旺, 李文, 彭鑫, 等. 基于本体的文档语义标注改进方法[J]. 东南大学学报: 自然科学版, 2009, 39(6): 1109-1113
- [12] 杨艳萍, 谭庆平. 一种有效的服务资源自动语义标注方法[J]. 计算机研究与发展, 2007, 44(1): 37-43
- [13] Popov B, Kiryakov A, Ognianoff D, et al. KIM: a Semantic Platform for Information Extraction and Retrieval[J]. Natural Language Engineering, 2004, 10(3/4): 375-392
- [14] Kopecky J, Vitvar T, Bournez C, et al. SAWSDL: Semantic Annotation for WSDL and XML Schema[J]. IEEE Internet Computing, 2007, 11(6): 60-67
- [15] 白伟华, 李吉桂. 基于 SOA 的多边多议题协商模型的研究[J]. 计算机科学, 2008, 35(1): 151-153, 175
- [16] Zhu Jia-xian, Bai Wei-hua, Li Ji-gui. The mediation data services application model based on web resource ontology[C]//1st International Conference on E-Business and E-Government, Guangzhou China, 2010: 1468-1472
- [6] Arkko J, Haverinen H. Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)[M]. RFC 4187, January 2006
- [7] Lin Yi-bing, Chang Ming-feng, Hsu M, et al. One-pass GPRS and IMS authentication procedure for UMTS [J]. IEEE Journal on Selected Areas in Communications, 2005, 23(6): 1233-1239
- [8] Huang Chung-ming, Li Jian-wei. One-Pass Authentication and Key Agreement Procedure in IP Multimedia Subsystem for UMTS[C]//IEEE 21st International Conference on Advanced Networking and Applications (AINA'07). Niagara Falls, ON, Canada, May 2007: 482-489
- [9] 徐鹏, 廖建新, 朱晓民, 等. GPRS 与 IMS 认证与密钥分发的一种优化集成方法[J]. 北京邮电大学学报, 2006(05)
- [10] 张奇支, 黄兴平, 范冰冰. IMS 中一种快速的用户注册过程[J]. 计算机科学, 2010, 37(9): 117-120

(上接第 64 页)