基于领域本体的软构件检索

樊晓光 褚文奎 万 明
(空军工程大学工程学院 西安 710038)

摘 要 为了提高刻面分类检索软构件的查准率,结合领域本体,提出了支持自然语言检索的软构件检索过程模型。该模型抽象了领域知识,形成领域本体库,用于匹配用户检索使用的自然语言,提供领域内一致认可的检索术语。该术语然后与软构件描述库中的软构件描述术语进行匹配,进而从软构件库中检索软构件。软构件描述库采用了刻面分类方法。ATS软构件检索实验结果表明,较之于传统的刻面分类方法,该检索策略既提高了检索精度,又增强了检索的灵活性。

关键词 软构件,领域本体,刻面分类,构件检索,基于构件的软件开发
中图法分类号 TP311 **文献标识码** A

Software Component Retrieval Based on Domain Ontology

FAN Xiao-guang CHU Wen-kui WAN Ming
(Inst. of Engineering, Air Force Engineering Univ., Xi'an 710038, China)

Abstract To enhance the precision of component retrieval based on facet classification, a software component retrieval process model was presented with domain ontology to support natural language retrieval. An ontology repository abstracted from domain knowledge was used to match what users input in natural language and outputs coherent retrieval terms in the domain. These terms were then used to match what component described in the component description repository with facet classification. Related software components were retrieved from the component repository. The results of experiments in the ATS field show that the new retrieval method can enhance the retrieval precision compared with the traditional facet classification, and that it is flexible enough because of the use of natural retrieval language.

Keywords Software component, Domain ontology, Facet classification, Component retrieval, Component-based software development

软构件检索是指从软构件库中检索出满足用户需求或近似需求的软构件。软构件检索是基于软构件的软件开发(component-based software development, CBSD)的一个研究热点,是软构件得以复用的前提。当前,典型的软构件检索方法包括关键字搜索、刻面分类[1]、基调匹配[2]和行为匹配[2]等。相较于关键字搜索方法查全率、查准率不高,只适宜于小规模软构件库,以及基调匹配和行为匹配还处于研究状态而言,刻面分类方法最为成熟、实用。

刻面分类法通过反映软构件本质特性的视角(即刻面)对软构件进行分类刻画。每个刻面由一组术语构成,不同刻面的术语之间正交。查询时,提取不同刻面的若干术语形成一个描述子从软构件库中检索软构件。刻面分类方法的优点在于:(1)它将术语(即关键字)置于特定的语境中,避免了术语的杂乱无章;(2)它不针对特定领域,具有一定的普适性;(3)分类策略易于修改,富有弹性。当前,采用刻面分类的典型软构件模型有REBOOT[3]和北大青鸟[4]。

不过,刻面分类方法也存在下述问题,这些会影响到软构件的检索效果:(1)刻面分类具有一定的主观成分;(2)缺乏对于软构件的服务和功能等刻面的形式化描述方式;(3)重视软构件的静态特征描述,缺乏动态行为的描述机制。

为了克服传统刻面分类表示方法的不足,柯文[5]结合分层刻面法和形式化规约说明两种方法,提出了分层综合刻面表示法,将软构件看成是一个六元组,即:

软构件=〈应用领域,层次,功能,关键刻面,使用环境,形式规约说明〉

该方法既描述了软构件的静态特征,比如应用领域、开发层次、功能集合、关键刻面(算法、语言、类型等)、使用环境等,又刻画了软构件的动态行为特征,能够使软构件不同的刻面分类对应于不同的子领域,提高了形式化规约说明方法的准确率,提升了软构件的查准率。

近年来,本体论[6,7]在信息领域、航空航天领域等得到了广泛发展[8,9]。本体论结合具体领域产生的领域本体[10]是对该领域的高度抽象。它结合领域专家知识概括了该领域内的关键概念以及概念之间的关系。

到稿日期:2008-07-28 返修日期:2009-03-03 本文受总装预研项目(9140A17020307JB3201),中国博士后科学基金(20060400999),空军工程大学工程学院优秀博士学位论文创新基金(BC07003)资助。

樊晓光(1965—),男,教授,主要研究方向为软构件复用技术、ATS、航空电子系统;褚文奎(1980—),男,博士生,主要研究方向为综合航电软件技术等,E-mail:chuwenkui@126.com;万 明(1979—),男,讲师,博士,主要研究方向为综合航电等。

与柯文着眼解决传统刻面分类方法面临的3个问题不同，本文致力于解决第一个问题。本文引入自然语言解析模块和领域本体库，一方面期望支持自然语言检索，提高检索的灵活性，另一方面期望借助领域专家知识形成“标准、专业”的查询术语，提升查询的精确度。

1 软构件检索系统

软构件检索系统的本质问题是如何建立软构件的智力模型，精确表达软构件的功能、应用领域、工作环境、动态行为等，在软构件创建者和使用者之间架起“理解”的桥梁，使软构件能够得以复用。可以说，软构件检索系统是大范围内、系统化实施软件复用的必备基础。

1.1 体系结构

一般而言，软构件库中存储的软构件包括软构件实体和相关的描述信息。二者既可以合并存储，也可以分离存储。为了降低软构件检索系统的运行负荷，提高系统的开放性，便于升级维护，本文采用分离存储方案组织软构件。根据B/S三层(用户界面层、应用逻辑层和数据库层)模式，将软构件检索系统的体系结构组织成如图1所示。

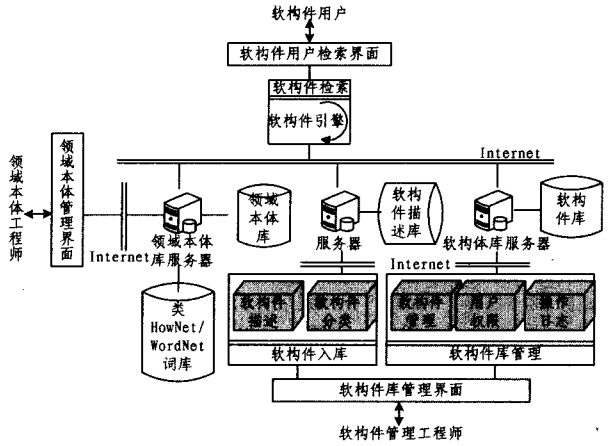

图1 软构件检索系统体系结构

用户界面层采用Web形式，分别为软构件检索用户提供检索界面，为软构件管理工程师提供软构件入库和管理的界面，为领域本体工程师提供管理领域本体库的界面。

应用逻辑层负责软构件的描述、分类、使用管理、反馈信息管理和软构件库的用户权限管理、操作日志管理，可通过基于Web开发的软构件管理界面予以实现。

在数据库层，该结构提供4个数据库：类HowNet[11]/WordNet[12]词库、领域本体库、软构件描述库和软构件库。类HowNet/WordNet词库结合自然语言解析模块处理以自然语言输入的查询信息，分离出初始查询术语。领域本体库存储特定领域的本体，提供精确的查询术语，消除同名异义、同义异名现象。软构件描述库依据软构件描述模型提供对软构件的接口、功能、层次、应用领域、开发语言、使用环境、版本等的描述，便于软构件检索。软构件库中存储软构件实体，提供下载服务。

此外，软构件检索系统提供两类最基本的工具：一类是软构件检索工具，另一类是软构件库管理员使用的工具，其功能包含软构件的描述、分类、管理、用户权限的设置和使用日志等。

1.2 软构件描述模型

软构件描述模型是对软构件本质的刻画，提供了有关软构件属性的描述信息，这些信息存储在软构件描述库中。结合刻面分类的思想，本文选择下述基本属性、分类属性和附加属性来描述软构件，如表1所列。基本属性描述了软构件的基本信息，比如名称、摘要、编号等。分类属性从功能类型、应用领域、适用层级、目标对象等刻面描述了软构件。附加属性则包含了软构件的活动类型、开发方法、开发环境、关联软构件和提交日期等信息。

表1 构件描述模型

软构件实体属性	软构件实体序号	软构件实体名称
基本属性	1	软构件编号
	2	软构件名称
	3	软构件摘要
	4	软构件存储路径
分类属性(关键属性)	5	功能类型
	6	应用领域
	7	层级
	8	目标对象
	9	源对象
	10	中间对象
	11	接口
	12	核心算法
	13	实现语言
	14	工作环境
附加属性	15	系统类型
	16	活动类型
	17	开发方法
	18	开发环境
	19	关联软构件
	20	参考软构件
	21	版本号
	22	提交日期

在这3类属性中，分类属性提供软构件检索时的主要信息，需要尽可能地采用专业术语或者领域内公认的术语进行刻画。此举在于促进领域本体表达的软构件信息与软构件描述模型提供的信息相匹配，提高软构件的检索效果。

2 基于领域本体的软构件检索过程

2.1 检索过程模型

软构件检索基于图1所示的软构件检索系统结构进行。软构件检索用户通过Web检索界面输入简单的自然语言，这些语言经过自然语言解析模块与类HowNet/WordNet词库进行交互，解析用户的需求查询信息，分离出初始查询术语。

初始查询术语然后提交给初级查询模块，通过与领域本体库进行匹配交互，选择最恰当的描述术语集(如果没有与查询术语严格匹配的术语，则利用启发式算法在领域本体库中选择同义词)并反馈给用户，由用户根据实际需求进一步筛选、细化，形成精确需求查询框架(即一个描述子)。

用户的精确需求经由精确查询处理模块映射到基于刻面分类的软构件描述库上，按照既定的检索算法检索合适的软构件并返回查询结果。用户从中筛选合适的软构件后，再从软构件库中下载使用。整个检索过程如图2所示。

在此检索模型中，系统按B/S模式进行组织。自然语言解析模块、初级查询模块由领域本体服务器提供，精确查询处理模块则由软构件描述库服务器提供。

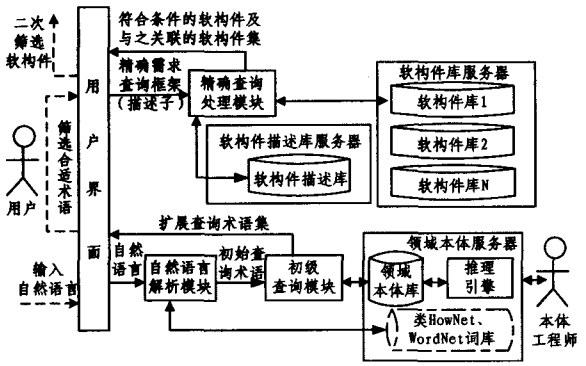

图 2　基于领域本体的软构件检索过程模型

2.2　精确查询匹配算法

假设输入到精确查询处理模块的某一个查询框架可以表示成：$Q_p=\{qp_1,qp_2,\cdots,qp_{10}\}$，其中 qp_1 至 qp_{10} 分别表示软构件描述模型中的 10 个分类属性(如表 1 所列)，对应软构件描述库中软构件 C 的分类属性集是 $C_p=\{cp_1,cp_2,\cdots,cp_{10}\}$，那么查询框架与软构件的匹配程度为：

$$M(Q_p,C_p)=\frac{\sum_{i=1}^{10}\omega_i\times m(qp_i,cp_i)}{\sum_{i=1}^{10}\omega_i} \tag{1}$$

当 $M(Q_p,C_p)\geqslant\theta$ 时，定义软构件 C 的分类属性与查询框架匹配，软构件 C 是所期望的查询结果。θ 是匹配可接受的阈值，比如 0.80。式(1)中，$\omega_i\in\{1,2,\cdots,10\}$ 是根据上述属性重要度分配的权重；$m(qp_i,cp_i)$ 表示查询框架与软构件 C 某一属性的匹配度，其取值范围为：

$$m(qp_i,cp_i)=\begin{cases}1 & (m'(qp_i,cp_i)\geqslant\varepsilon)\\ 0 & (m'(qp_i,cp_i)<\varepsilon)\end{cases}$$

$m'(qp_i,cp_i)$ 是指针对某一个具体属性，比如接口规约，根据 pre-match，post-match，plug-in match，plug-in post match，weak-post match 等匹配方法[13,14]进行匹配的结果，ε 是匹配可接受的最低门限值。

3　实验评测

3.1　ATS 领域本体构建

自动测试领域的软构件按其作用可分为用户界面、数学分析、格式化及文件操作、VXI 通用驱动、GPIB 通用驱动、TCP/IP 通讯、英特网操作、动态数据交换、INI 文件及注册表操作、定时器、通用硬件驱动、专用硬件驱动和一般软构件。基于上述高层分类，图 3 刻画了部分 ATS 领域本体。

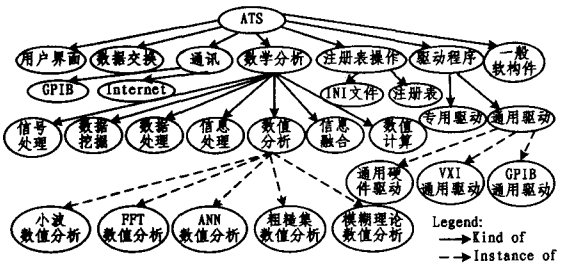

图 3　ATS 领域本体

图 3 中，实线表示子类关系(kind of)，比如通用驱动是驱动程序的一个子类，而虚线表示实例关系(instance of)，比如 ANN 数值分析是数值分析的一个实例。

3.2　ATS 软构件检索过程

本文以查询“数据分析”软构件为例说明软构件的检索过程。

用户首先在 ATS 软构件检索系统客户端的查询栏中输入“请检索数据分析软构件”，在约束栏中输入“当前自动测试领域本体库”，然后提交给系统。经过自然语言解析模块处理后，提炼出“数据分析”术语。初级查询模块然后将之与领域本体库进行匹配。由于领域本体库没有完全与之匹配的术语，因而借助启发式搜索算法，检查本体库中的同义词“数据处理”、“数值分析”、“信息处理”、“信号处理”、“数值计算”、“数据挖掘”、“信息融合”等，并提交给用户，请求选择一个与之匹配。比如选择“数值分析”，那么在“详细信息”栏将会进一步显示相应的数值分析方法，比如小波变换、ANN、FFT、粗糙集等。如果选择“小波数值分析”，通过与软构件描述库进行交互，将获取该软构件的属性、下载地址以及与之相关的软构件列表。

3.3　结果与评价

采用传统刻面法和领域本体法从某一 ATS 软构件库中检索相应软构件的结果如表 2 所列。由此可知，引入领域本体、加入查询精化阶段提高了基于领域本体的软构件检索方法的查准率。不过，由于需要不断与用户进行交互，该方法增加了时间开销，对用户的检索耐心是一大挑战。

表 2　软构件检索结果

检索方法	检索项	检索结果				
		库中相关数	实际检出	检出相关数	查准率	查全率
传统刻面法	通用驱动	101	123	80	65.0%	79.2%
	数值分析	22	27	17	62.9%	77.3%
领域本体法	通用驱动	101	99	97	98.0%	96.0%
	数值分析	22	23	21	91.3%	95.5%

结束语　本文将领域本体引入到基于传统刻面分类方法的软构件检索过程中，在一定程度上解决了查准率不高的问题。此外该模型支持自然语言检索，有利于对分类刻面不清楚的软构件用户。不过，自然语言解析是一个复杂问题[15]，本文未就其内容做深入研究。本文所设计的软构件检索系统目前只支持简单的陈述句、祈使句，还不具备解析复杂语言的能力。这也是下一步的研究重点。

参 考 文 献

[1]　王渊峰，张涌，任洪敏，等. 基于刻面描述的构件检索[J]. 软件学报，2002，13(8)：1546-1551

[2]　徐正权，王家兵，王能超. 软件构件表示与检索形式化的研究与进展[J]. 计算机科学，2003，30(7)：99-102

[3]　Stockwell T，Conradi R，Karlsson EA . The REBOOT Approach to Software Reuse[J]. Journal of System Software，1995，30：201-212

[4]　常继传，李克勤，郭立峰，等. 青鸟系统中可复用软件构件的表示与查询[J]. 电子学报，2000，28(8)：20-23

[5]　柯文. CAPP 领域构件复用技术研究[D]. 南京：南京航空航天大学，2003

[6]　邓志鸿，唐世渭，等. Ontology 研究综述[J]. 北京大学学报：自然科学版，2002，38(5)：730-738

[7]　李善平，尹奇韬，胡玉杰，等. 本体论研究综述[J]. 计算机研究与发展，2004，41(7)：1041-1052

(下转第 238 页)

证明：设 c 是概念，在 F 中没有它的表示，且 $x \notin F$ 是一个未知表示。由定义 4 知，$ext(x) \neq \emptyset$，因为概念是具有非空外延，按照定理 5，概念 c 的表示 $x' \sqsubseteq x$ 可在有限次数的运算中产生。

定理 5 的证明揭示了 3 个迭代。第一是经由折半位置的迭代，为了找到可产生公式 x 的折半位置；第二迭代是折半运算的递归应用，直到找到 x 的实例 x' 为止；第三个迭代是进一步泛化 x' 直到到达 x 的概念；这意味从新生成的公式构建新的折半位置。

定理 5 及其推论证明在有限时间内可以区分概念，即使逻辑查找空间是无限的，即使 lggs 是不确定的或无限的，在不约束应用域逻辑的情况下这个结论通过自顶向下和自底向上查找不可能获得。

5 通用算法及其复杂性

下面给出算法的伪码，设背景 $K=(\mathcal{O},(L,\sqsubseteq),\delta)$，初始公式集 F，从 F 中得到折半位置的初始集 P，事实上，折半位置仅有 2 个部分 (y,o) 存储在 P 中，第三个部分 $Z=lubs_F(y,\delta(o))$ 取决于当前特征集 F，这些位置用于通过折半运算产生新的公式集，进而又产生新位置。当没有新的公式集产生时，位置删除，当没有位置余下时，整个过程结束。

```
1  F←δ(∅);
2  P←{(y,o)∈F×∅|o∉ext_K(y)};
3  while P≠∅ do
4     (y,o)←选取合适的 P;
5     Z←lubs_F(y,δ(o));
6     X←binary(y,δ(o),Z);
7     if X=∅ then
8        P←P\{(y,o)};
9     else
10       F←F∪X;
11       P←P∪{(x,o)∈X×∅|o∉ext_K(x)};
12 done
```

上述算法的输出是表示背景 K 的所有可能概念的公式集。在我们的实现中，生成的公式集 F 内部表示为包含序的 Hasse 图，该表示与表(list)相比具有不少优点：(1)代价高的包含测试仅用于 F 中新元素的插入，在算法其它地方不用；式(2)的外延可以通过该图简单的向下遍历来计算，事实上，在公式插入时完成并储存；(3) 2 元素的 lubs 也能通过图的遍历来计算，节省了许多包含测试。

算法伪码的第 4 行，可通过启发式算法产生最想要的位置，这要求位置是有序的。P 内部用二叉树表示，这种表示能够选择最佳位置，增加新位置用 $O(\log|P|)$ 时间，而采用表结构的时间为 $O(|P|)$。

现在讨论算法的复杂性，首先，定义背景中的对象数量为 $n=|\mathcal{O}|$，$N=|F|$ 为生成的公式的数量，其次，位置的数量 $|P|$ 不超过 Nn，再者，每次折半查找运算只产生一个公式。算法中重要几行的复杂性如下：(1)第 4 行为 $O(\log(Nn))$：最佳位置的检索时间；(2) 第 5 行为 $O(N)$：*lubs* 的计算时间；(3) 第 6 行为 $O(|dicho|)$：折半运算的时间；(4) 第 10 行为 $O(N|\sqsubseteq|)$：在 Hasse 图 $(F,\sqsubseteq)$ 中新公式的插入；(5)第 11 行为 $O(n(|h|+\log(Nn)))$：将计算新位置以及插入它们的时间所产生 N 个公式的局部复杂性综合起来，整个算法的最坏复杂性为：$O(N^2|\sqsubseteq|+N|dicho|+Nn(|h|+\log(N)))$。第一项是非常重要的，在包含图中反映新公式的插入，第二项对应折半运算的应用，第三项表示评价和排序位置的代价。各种实验已经证明两个主要的开销是在图中新公式的插入和折半运算上，这也与理论上的复杂性一致，因为当生成的公式越来越多，第一项所花的运行时间也在增加。

结束语 本文提出一种新的基于概念的折半查找算法。它的优点：(1)该算法是通用算法，所用的逻辑是抽象的，可用简单的基本逻辑，或具体领域使用的复杂逻辑来实例化它。(2) 因为折半位置的顺序是任意，所以它提供了更为灵活的查找，并在查找的同时保持完备性和非冗余性。同时，折半运算也存在一些灵活性，因为它的定义并不像细化运算和泛化运算那样受约束。

参 考 文 献

[1] Califf M E, Mooney R J. Bottom-up relational learning of pattern matching rules for information extraction[J]. Journal of Machine Learning Research, 2003, 4: 177-210

[2] Baader F, Usters R K, Molitor R. Computing least common subsumers in description logics with existential restrictions[C]// Proc. of the 16th Int Joint Conf. on Artificial Intelligence (IJCAI-99). 1999: 96-101

[3] 郑宗汉. 算法设计与分析[M]. 北京：清华大学出版社，2005

[4] Badea L. A refinement Operator for Theories [C] // Inductive Logic Programming, LNCS 2157. 2001

[5] Lita L V. Instance - Based Question Answering [D]. Carnegie Mellon University, 2006

[6] Ganter B, Wille R. Formal Concept Analysis — Mathematical Foundations[M]. Springer, 1999

(上接第 158 页)

[8] 高颖，曹存根，眭跃飞. 音乐领域本体的建立和分析[J]. 计算机科学，2004，31(1)：103-107

[9] 胡艳丽，白亮，张维明，等. 知识网格中基于领域本体的智能检索[J]. 计算机科学，2007，34(8)：202-207

[10] 顾慧翔，俞勇. 基于领域本体和知识推理的语义互联网应用[J]. 上海交通大学学报，2004，38(4)：583-585

[11] Dong Z Q, Dong Q. HowNet[EB/OL]. http://www.keenage.com/, 2008

[12] 赵天忠，苗壮，张亚非，等. 基于 WordNet 重用的领域本体构件方法[J]. 系统仿真学报，2007，19(19)：4583-4586

[13] 李晓博，缪淮扣，刘静. 基于形式规格说明的构件匹配[J]. 计算机应用与软件，2006，23(10)：10-12

[14] Wang H, Feng Y, David C. Verifying the Reusability of Software Component Specification Framwork and Algorithms [J]. Information Science, 1998, 112(12): 169-197

[15] 袁柳，李战怀，陈世亮. 基于本体的 Deep Web 数据标注[J]. 软件学报，2008，19(2)：237-245