

一个面向软件过程的知识挖掘与管理系统^{*}

张晓刚 李雄锋 陈晋川 李明树

(中国科学院软件研究所 北京100080)

摘 要 软件过程的知识密集性特征要求组织人员的密切协作、交流,并能够有效地重用组织的实践经验。数据挖掘技术能够从大规模数据中发现有意义的知识,能够为知识管理提供有益的支持,从而得到更好的知识共享和重用效果。在对软件过程进行分析的基础上,提出了协同知识管理模型。介绍了一个集成的面向软件过程的知识挖掘与管理系统 iKnow。对 iKnow 的知识表示、知识挖掘、知识服务和个性化知识检索进行了论述。系统使软件组织的实践经验成为可视、可控和可重用的知识资源。

关键词 软件过程,数据挖掘,协同知识管理,组织实践,知识管理系统

A Software Process Oriented Knowledge Mining and Management System

ZHANG Xiao-Gang LI Xiong-Feng CHEN Jin-Chuan LI Ming-Shu

(Institute of Software, Chinese Academy of Sciences, Beijing 100080)

Abstract The knowledge intensive feature of software process requires both cooperation & communication among people and reuse of experiences. Data mining (DM) focuses on the extraction of interesting patterns from data in large databases. It can improve the effectiveness of knowledge management (KM) in knowledge sharing and reuse. Based on the requirement analysis about KM in software process, a Cooperative Knowledge Management model is proposed. A software process oriented knowledge mining and management system named iKnow is introduced. Technologies about knowledge representation, DM, knowledge services and personalized knowledge retrieval of iKnow are discussed in detail. This system helps software organization's experiences become a kind of visible, controllable, and reusable knowledge resources.

Keywords Software process, Data mining (DM), Cooperative knowledge management (CKM), Organizational practices, Knowledge management system (KMS)

1 引言

知识管理、学习型软件组织^[1]是当前软件行业研究与实践的热点。通过系统地实施知识管理,能够有效地管理组织内部激增的知识资产、保持和增强组织的竞争能力、减少人员流动带来的损失、为业务方向和组织结构的调整提供决策支持,并为软件组织的规范化管理和过程改进提供有力的支持。

软件组织的知识管理可以分为工程数据管理和组织实践管理。工程数据管理主要针对软件过程和项目度量数据,采用粗糙集、逻辑回归等技术对质量、性能等进行分析,用于预测和指导软件过程改进^[2]。通常的软件项目管理软件中具有一定的工程数据管理功能。组织实践管理实际上是当前知识管理研究中的主要内容,以促进经验、最佳实践的获取、传播和重用为目标。组织实践管理通常采用人工智能、知识工程等技术来实现,当前基于案例的推理(Case-based reasoning, CBR)技术、本体(Ontology)技术得到了广泛的研究与应用。在 CBR 技术中,强调从过去的经验中学习,新问题的解决基于对已知的类似问题的解决方案的修改来完成^[3];案例的表示与相似性计算^[4, 5]是 CBR 技术的关键。BORE (<http://cse-ferg41.unl.edu/bore.html>)是一个采用 CBR 技术的软件开发经验管理环境,通过案例和领域规则为软件过程提供知识服务^[6]。本体由于能够促进智能搜索、系统互操作的实现而成为当前知识管理领域的研究热点。Tautz 等研究了一种基于

本体的软件工程经验共享方法 DISER^[7]。OntoWeb (<http://ontoweb.aifb.uni-karlsruhe.de>)是一个以本体为基础进行知识管理的通用平台,定义了项目、组织、产品、人员、出版物等本体,实现基于本体的知识浏览、访问和提供,并能够与其他系统实现信息的交换^[8]。国内对知识管理技术的研究当前主要体现在一些核心支撑技术方面,比如本体、信息检索、个性化服务等。

实践表明,独立于组织业务过程的知识管理往往被人员看作额外的工作负担,不易取得好的效果。而为了适应软件过程与人员知识需求的多样性和动态性,仍需要知识管理技术在方法和工具上的深入研究。另一方面,利用数据挖掘技术来对组织的知识资产库进行分析,辅助知识管理的实施,将有利于取得更好的知识管理效果。因而,面向软件过程、结合数据挖掘技术来实现组织实践管理是我们的研究目标。

知识管理系统是实施知识管理的重要基础设施,本文对我们研究开发的基于 Internet 的软件组织知识管理系统 iKnow 进行介绍。它通过一个协作环境实现软件过程与知识管理过程的集成,并以数据挖掘技术支持系统的知识管理功能。文章第2部分分析了软件过程的知识管理需求,介绍了提出的协同知识管理模型;第3部分对 iKnow 的系统结构进行介绍;在第4部分对 iKnow 中的主要数据挖掘和知识管理进行论述;最后对全文进行总结。文中数据挖掘与知识挖掘表示同样的概念。

^{*} 基金项目:国家高技术研究发展计划课题“数据挖掘与软件企业知识管理技术”(No. 2001AA113180)。张晓刚 博士研究生,研究方向:智能软件工程、知识管理。李雄锋 硕士,研究方向:智能软件工程、知识管理。陈晋川 硕士研究生,研究方向:智能软件工程、知识管理。李明树 博士,研究员,博士生导师,主要研究方向:智能软件工程、实时系统。

2 软件过程与协同知识管理

2.1 软件过程中的知识管理需求

软件过程是软件组织的主要业务过程,指开发和维护软件及相关产品时所采用的一系列活动、方法、实践和转换^[9]。在软件过程中进行知识管理需要解决以下几个主要问题:

(1)改善和增强人员之间的交流与协作。软件过程是知识密集型业务过程,促进人员之间的交流和协作是软件开发效率和软件产品质量的保证。随着异地、分布式软件开发情况的增多,项目人员之间的协作与工作成果的集成方式变得更加重要;

(2)方便的组织实践获取和传播方式。促进软件过程资产的积累、知识在相关人员间的传播、共享和重用;尤其要适应人员知识需求的多样性和动态性,为人员在任务中提供智能的辅助支持^[10];

(3)合理的组织实践数据模型。以保持组织实践内容之间的关联,便于从多维度对其进行组织和分析;数据模型应该能够很好地支持组织实践的重用,而不能停留在信息的获取与储存上;

(4)有效的组织实践浏览与检索。便于人员对组织实践库的浏览,获得与需求高度相关的检索结果,从而促进项目技术方案的确定、项目中具体问题的及时解决;

(5)组织实践中的知识发现。在软件组织的大量实践信息中,蕴含了实际项目中获得的研发能力信息、知识间的关联信息等有价值的模式,对这些知识的挖掘将对知识重用和软件过程改进提供可靠的决策基础。

此外,在组织层次,软件组织需要宏观上清晰、直观地了解组织内部的知识状况,尤其是技术上的优势、不足、知识分布状况与积累状况,以此作为项目策划、组织发展规划的依据,并针对技术不足有计划地采取相应措施进行弥补和改善。解决以上问题是面向软件过程的知识管理研究的关键。

2.2 协同知识管理

针对软件过程的协作性、动态性和软件人员知识共享与重用的需要,我们提出了协同知识管理的思想。协同知识管理(Cooperative Knowledge Management, CKM),指以软件组织中的软件过程为核心,通过对组织范围内的软件过程和其它相关支持过程的分析,协调知识处理和组织学习,使之与软件过程紧密结合,促进不同软件过程间、组织人员间的交流与沟通 and 知识成果的共享与重用,从而保证组织内知识管理的有序性,提高软件过程的可控性。图1为 Internet 环境下的软件组织协同知识管理模型。

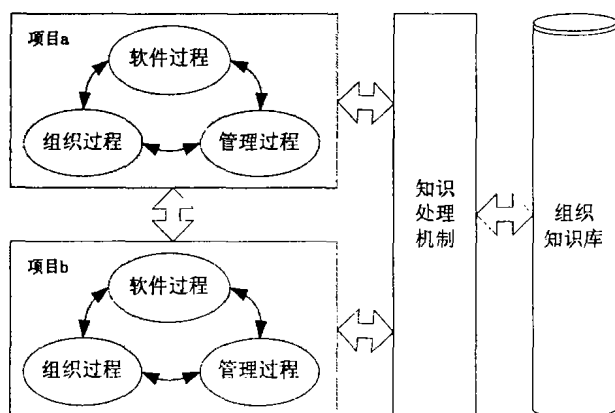


图1 协同知识管理模型

为软件过程提供知识服务和执行保障是知识管理的核心

目标。协同知识管理,强调以全组织范围内的项目过程关联分析为基础,开展知识管理活动。协同知识管理的特点是过程关联、知识共享、人员协作。过程关联指业务过程之间的关联,它包括两方面,任一项目中软件过程、组织过程、管理过程等的关联,组织范围内相关项目的过程间的关联。过程关联提供了知识共享的基础。人员协作既包括项目内相关人员之间的知识交流,也包括组织范围内相关人员之间的知识共享与学习;这种多形式的人员协作是软件产品演进的基础,也是对隐性知识进行管理、促进隐性知识向显性知识转化的有效途径。协同知识管理方法将知识管理与组织学习这两个互补的方面融入组织过程中,体现了面向过程的知识管理和面向制品的知识管理的结合。

协同知识管理模型中,知识管理的服务功能通过知识处理机制结合到组织的业务过程中,包括任务制定与执行过程中与相关组织实践的关联(识别相关实践和适用规范)、历史经验的重用、相关常见问题与领域专家的识别、任务信息与过程产品(及信息)的获取等,实现知识管理过程与业务过程的集成。过程集成促进了组织的任务、人员和实践经验等信息的互相关联,为实现推理、知识检索和主动服务提供了必要前提。

3 知识挖掘与管理系统 iKnow

基于协同知识管理的思想,我们设计开发了一个基于 Internet 的协同知识管理系统 iKnow (Internet-based cooperative KNOWledge management system)(图2),作为辅助软件组织进行知识管理、建立学习型组织的工具。

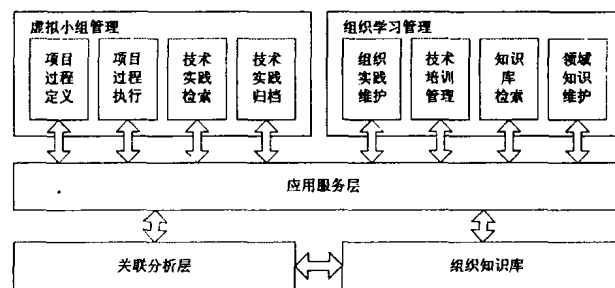


图2 知识管理系统 iKnow 的基本结构

iKnow 是一个支持协作和知识共享的集成环境,在应用层它包括两个主要功能组件,虚拟小组管理组件和组织学习管理组件。系统的应用服务层为顶层的虚拟小组和组织学习处理逻辑提供分类、查询和知识发布方面的核心支持功能。关联分析层为应用服务层提供封装的数据分析服务,即在领域知识的支持下,根据处理任务对组织实践、任务制品和用户特征等进行综合分析挖掘,提供用于分类、查询和推荐的指导信息。

我们将软件组织中的任何项目组都称为虚拟小组,因而它不局限于软件开发项目组,可以是任何具有明确工作目标并分配了资源的任务组。虚拟小组为成员提供了一个基于浏览器的协作环境,因而消除或降低了对成员在地域和时间上的限制;通过项目过程定义、项目过程执行、技术实践检索和技术实践归档等主要操作,小组成员可以在项目执行过程中方便地进行知识的传递、重用组织已有实践经验。在组织学习管理组件中,通过组织实践维护、技术培训管理、知识库检索、领域知识维护,组织的最佳实践和经验教训可以得到有效的传播和学习,促进了个人实践经验向组织实践经验的转化。

在所管理的知识方面,系统具有下面两个特点:

1、以组织实践经验的管理为主,并实现了组织实践经验和软件过程领域知识的结合。通过组织实践维护模块,可以定义组织级和项目级的组织实践框架,并指定组织的最佳实践供新任务参照;领域知识维护模块支持渐进式的领域知识建模与重用,从而便于系统尽快地在软件组织中得到使用,领域知识可以随着组织过程能力的提高和规范化建设的发展而得到扩展。

2、对显性知识(组织实践、技术文献等)和隐性知识进行了有序的管理。在隐性知识方面,通过建立相关人员之间的关联而促进知识的传播。针对知识管理中的核心知识处理需求,对显性知识中隐含的模式进行挖掘,为知识管理提供必要的底层支持。

作为软件过程中的辅助工具,系统还从以下两方面来实现友好的服务功能:分析和利用用户业务过程中的信息,减少(或辅助完成)用户的信息输入;基于用户建模和协作过滤技术来实现个性化用户界面与知识的主动服务。

4 iKnow 的知识挖掘与管理

4.1 知识表示

知识管理中的知识指能够指导行动的、具有可操作性的信息。在内涵上这个定义与数据挖掘所获取的模式(或知识)具有一定的重合,后者指隐含于数据中的“有趣”的信息。知识与信息之间没有绝对的区别,储存起来的“静态”的知识成为信息,使用中的“动态”的信息成为知识。我们把知识从层次上分为领域知识和组织实践两类。领域知识指研究领域的概念(及其属性)、概念间关系等,具有应用无关性。组织实践是组织的实际工作经验与制品(文献、报告等)的统称;组织规范是一种特殊的组织实践,它被强制要求在适用过程与任务中贯彻。iKnow 实现了对领域知识和组织实践的挖掘与管理。领域知识在系统中处于基础位置,辅助组织实践的获取(为其提供概念描述框架)和分类、搜索与传播;同时,系统基于领域知识对组织实践进行数据挖掘,并进而促进领域知识的完善。

1. 领域知识的表示 我们以本体来对领域知识进行表示,描述软件过程相关的概念、概念的属性、概念之间的关系(包括父子关系、组成关系、实例关系等)。这些本体概念和概念间关系定义在项目、任务、人员、制品、参考文献和技术领域等10类本体中,用于描述系统所需的软件过程中的知识要素,各类本体之间的关系如图3所示。

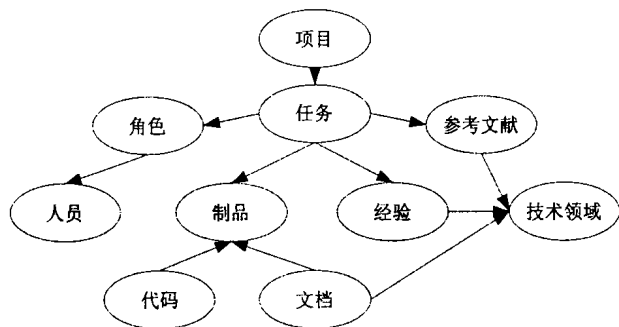


图3 iKnow 的领域知识本体

本体中概念的内涵与对象相一致,因而,我们采用框架对本体的内涵进行表示。一个本体概念的框架由框架名称、属性槽、刻面和框架约束组成^[11];领域本体中的概念集合依据概念间的关系构成了框架网络,框架间通过非终端槽建立语义关联。框架和框架网络分别提供了实例获取和推理的基础。为便于领域本体的扩展,系统内部定义了各类本体的元本体,供

组织人员以此为基础定义新的本体概念。

2. 组织实践的表示 在系统中,组织实践表示为组织实践框架,简称实践框架;组织实践框架的实例称为案例。组织实践框架以领域知识中的本体概念框架为基本元素,表示为嵌套结构的框架。一个典型的组织实践框架可以表示为(TF, RF, AF),其中 TF、RF、AF 分别为任务框架、角色框架、任务制品框架;参考文献和经验(指任务执行中的“问题—解答”信息)也可以根据需要定义到组织实践框架中。我们把组织实践框架分为组织级和项目级两种。组织级实践框架由软件过程过程的成员定义,描述组织规范;典型的组织级实践框架描述了组织的标准软件过程,框架的约束对应于适用项目任务类别、标准软件过程的裁剪准则。项目级实践框架由项目负责人员定义,以满足特定项目的知识表示需要。

通过这样的框架形式,我们实现了对包括数据、文本、文档等多种数据、信息源的包装;同时,任务、执行人员、制品(比如文档)、技术领域等信息之间互相关联,为搜索和挖掘提供了较为全面的上下文信息。

4.2 组织实践库挖掘

基于组织实践库和领域知识库,系统主要实现了下面四种知识挖掘功能:

1. 组织实践模式挖掘 在特定的组织内部,由于项目和人员技术方面的特点,任务、角色、应用技术之间存在相对稳定的模式。这些模式主要表现为信息之间的关联规则,隐含于组织实践库中。我们将组织实践关系数据变换为适合挖掘的形式(任务类型 ID,角色 ID,技术领域 ID 项集),对 Apriori 算法^[12]进行修改,挖掘这些关联规则;挖掘结果作为特殊的组织经验保存,为实现知识推荐、知识相关性分析和决策支持(如任务分派、培训策划)提供基础。同时,以上模式只覆盖领域知识的一部分或一小部分,是对于特定软件组织较为重要的领域知识,因而可以指导领域知识的细化和完善。随着组织实践信息的增多,这些规则逐步递增,使系统在知识管理中具有学习能力。

我们通过组织实践模式挖掘主要得到下述三种关联规则:①任务特征规则: $Task(x, T) \Rightarrow \{Knowledge(x, K_i)\}$, 表示 T 类任务与技术领域之间的关联;②任务—角色特征规则: $Task(x, T) \wedge Role(x, R) \Rightarrow \{Knowledge(x, K_i)\}$, 表示任务、角色与技术领域之间的关联;③领域实践知识规则: $\{Knowledge(x, K_i)\} \Rightarrow \{Knowledge(x, K_j)\}$, 表示不同技术领域知识类别之间的关联。

2. 资料分类 资料主要指项目文档、参考文献。由于资料的数量要远大于组织项目和任务的数量,并且资料的检索与重用是最主要的使用需求之一,因而系统对资料进行细致的分类处理。

我们利用 Naïve Bayes 方法^[12],由训练数据集得到对应于不同资料类别的分类器;分类器用于新资料的自动分类和资料查询。在资料分类时,一个资料以不同的概率属于多个类别,这一点也与实际情况相符。

资料的检索词(特征词)提取是分类的基础。在实际工作中,用户提交的资料具有多种文件格式。为简化实现,我们只从标题、关键词、摘要中提取检索词。利用向量空间模型中的 TF-IDF 公式对检索词的权重进行计算,最后保留限定个权重最高的词汇作为对应资料的检索词。

3. 人员特征和虚拟小组特征挖掘 人员的特征信息(Profile)是其在信息系统中的操作行为所蕴含的模式,表达了用户的兴趣点,是信息检索、过滤与推荐的主要依据。在知识管理中,人员特征还可以用于识别领域专家、识别兴趣相近

人员从而促进人员之间的沟通。虚拟小组是 iKnow 中的人员组织形式,其特征由小组成员所承担的任务、应用的技术所决定。刻画小组特征有助于知识在小组成员以及组织成员之间的传播,同时也反映了小组的知识积累状况,进而支持对其能力的评价。

已有很多对人员 Profile 表示和计算方面的研究^[13],用特征向量 $P=(v_1, v_2, \dots, v_n)^T$ 对人员兴趣建模, v_i 表示人员对领域 i 的兴趣度或权重。实际上,用户的一个兴趣点往往表现为多个关联的概念,而不是一些独立的概念,因而以独立的概念进行信息过滤往往导致搜索范围无益地扩大,影响结果的精度和系统性能;为此,我们采用 Apriori 算法,对特定人员(小组)提交、浏览和阅读的资料进行挖掘,用得到的置信度较高的关联规则 $\{Knowledge(x, K_i)\} \Rightarrow \{Knowledge(x, K_j)\}$ ($Confidence > 60\%$) 来表示兴趣点。这种关联规则形式的特征表示方法主要用于对搜索范围进行可靠的扩展和知识推荐。

另一方面,考虑到即使对于同一兴趣领域,人员或小组在不同类任务背景下兴趣度也会不同,因而我们将其 Profile 表示为 $P=(p_{t1}, p_{t2}, \dots, p_{tn})$, p_{ti} 表示在任务类 t_i 中所具有的特征。这种对特征信息的分类挖掘与管理进一步加强了知识(尤其是组织实践)过滤与推荐时的精确性;同时,兴趣变化仅影响特征信息中的一个子集,避免了通常的单一特征向量在处理兴趣“漂移”(Drift)方面的弱点。

为实现一般性的主动知识推荐,我们利用 Dempster 合成公式^[14]对以上特征关联规则和相应的置信度进行计算,得到描述人员或小组对技术领域的兴趣度的特征向量。Dempster 合成公式是 Dempster-Shafer 理论中的重要内容,提供了融合多个证据源计算对命题的支持度的方法。这种基于概率的证据融合方法所获得的领域兴趣度数值在合理性上要优于通常所采取的加权求和方法。

4. 组织实践库维护 组织实践库内容(主要指组织实践实例,即案例)的质量直接影响到系统的功用。通常,案例的质量可以通过系统在数据获取时进行基本控制。但是,当系统从组织遗留数据导入信息时,或不同系统间进行数据交换时,数据中存在的问题则无法避免。随着组织实践库规模的增大,检测其数据质量的难度也将增大。通过对隐含于组织实践库中的模式进行挖掘,可以有益地促进自动或半自动维护的实现,从而提高组织实践库的质量。系统对组织实践库和案例访问记录进行挖掘,通过下面四种技术实现组织实践库的内容维护^[15]:

检测不一致案例:不一致案例主要指与其它案例有太大区别、或某个属性值非法的案例。系统利用基于距离的孤立点挖掘算法来发现不一致的案例。

完善不完整案例:不完整案例指属性值出现非法空缺的案例。我们采用 K-最近邻算法,通过对该属性值的类别预测来得到空缺属性值。

检测冗余案例:冗余案例指案例的内容包含于其它案例的情况。系统通过 K-中心点聚类算法对案例进行聚类,然后由人员对同类的案例进行合并或删除操作。

检测垃圾案例:垃圾案例是组织不再需要的、一定时间内很少有人访问的案例。系统采用 n 阶加权移动平均的算法挖掘案例访问记录,将访问量趋于 0 的序列所对应的案例标记为垃圾案例。

4.3 自适应集成知识服务

人员的知识需求通常产生于特定的工作任务。任务相关的、及时的信息才是知识,所以对知识管理的最主要的性能要

求是知识服务的相关性与及时性。相关性包括两方面的要求:内容相关和用户相关。由于组织人员的分布性、知识需求的动态性及其与可获得的知识之间的不同步,知识在组织内的有效传播不易实现^[16]。对此,我们通过对人员任务上下文、组织实践实例进行跟踪与分析,动态地获取人员的知识需求和识别新的相关知识源,实现图4所示的自适应集成知识服务过程。

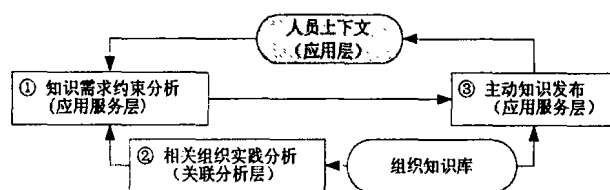


图4 自适应的集成知识服务处理过程

①知识需求约束分析 知识需求约束分析模块位于应用服务层。为组织人员在任务的执行中提供及时、在线的知识服务是实现知识管理的关键。我们把由具体任务与执行人员特征所确定的知识服务需求定义为知识服务约束,表示为 $kc=f(t, a)$, 其中 t 和 a 分别表示任务(Task)与其执行者(Actor)。执行者因素的引入是考虑即使对同类任务,不同人员的知识兴趣或需求仍然会有所区别。 kc 通过三元组 (Q_t, Q_p, Q_a) 来表示, Q_t 、 Q_p 和 Q_a 分别代表与该任务相关的实践、制品和组织人员(领域专家)查询条件集。在确定知识约束的前提下,系统能够主动把新的相关知识推荐给适当人员。

②相关组织实践分析 相关组织实践分析为知识需求约束分析提供支持,分析对象包括进行中的软件过程和组织已有的实践经验,从中搜索满足知识需求约束的相关组织实践。对于进行中的软件过程任务,在任务间建立关联,进而自动传递新的相关信息。这种关联关系可以由软件过程定义来分析;通过过程关联分析,实现业务实践经验的传递与重用。关联分析更主要的是根据任务与组织实践案例间在目标、应用技术、最终产品方面的相似性来进行;相似性由度量函数在指定的框架属性集中进行比较来确定。

③主动知识发布 系统以前两个步骤的分析为基础,为人员在任务执行中提供主动的知识服务。在人员接受新任务时、组织实践库获取新数据时,系统均会在人员间自动完成相应的知识传递。

上述知识服务的自适应性体现在对人员任务和新知识源的跟踪与及时处理上,集成性表现在知识内容上的集成和服务与业务过程的集成两方面。这种自适应的集成知识服务促进了知识资产的流通和人员之间的交流,使组织内部人员与知识资产之间形成了一个动态的网络。

4.4 个性化知识检索

为提高检索的准确率、帮助用户快速定位所需的信息,个性化检索是目前多个领域的研究热点。系统提供对组织实践中各类信息(比如案例、人员、文档等,统称为知识项)的检索功能,实现了表单式检索、基于样例的检索和关键词检索三种方式。

表单式检索 即根据相应组织实践框架的属性生成检索表单,人员输入查询条件后,系统利用相似性度量函数在组织实践库中搜索。我们定义了对数值型、枚举型、字符串型属性值的相似性度量公式^[15,17],通过相关的属性相似性的加权平均值来得到查询与实例数据间的相似性。

基于样例的检索 是系统以人员选择的一个特定的知识项为检索需求,通过相似性计算搜索与其最相似的其他知识

项。这种方法降低了对用户特征信息的需求,同时便于用户提供概念属性相关的检索标准。检索结果包括最相关的有限数据、基于关联规则所得出的潜在相关的主题类别(供用户进一步检索)。

关键词检索 系统根据查询关键词进行搜索,将分类的

检索结果提交给用户。在进行关键词检索时,由于检索词汇的随机性,系统需要根据用户特征、领域知识和组织实践模式对检索范围进行适当的扩展,以保证查准率和查全率。

系统实现的个性化知识检索过程如图5所示,主要处理步骤包括分析、检索和调整。

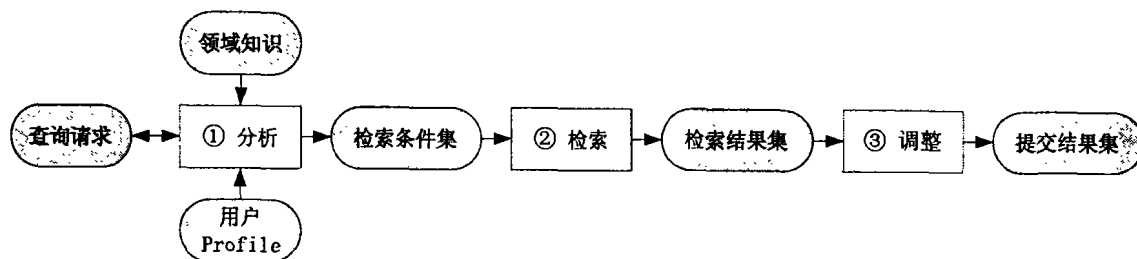


图5 个性化的知识检索处理过程

分析:根据人员的查询请求 R 确定扩展后的检索条件集 C_R 。扩展主要包括两方面:根据人员的查询请求 R 和人员当前执行的任务类型 t ,系统从其 Profile 中搜索适用的关联规则 $R \Rightarrow R_1$;根据领域知识搜索同义词和近义词集 R_2 。最终的检索条件集为 $C_R = \{(R:\eta), (R_1:\eta_1), (R_2:\eta_2)\}$;其中, η, η_1, η_2 为检索结果相似性修正系数, $\eta = \eta_2 = 1$, η_1 为相应的关联规则的置信度。

检索:对于案例类检索目标,系统采用相似性函数进行处理。对于资料类检索目标,系统采用向量模型进行检索;对每一个检索条件主要包括三个处理步骤:1、以前述 Naive Bayes 分类器为基础,计算检索条件属于所有每一个资料类别的概率;2、选取概率最高的限定个类别,通过夹角余弦公式计算检索条件与该类别中所有资料的相似性;3、选取相似性最高的限定个结果,利用检索条件对应的相似性修正系数计算最终的相似性,并对检索结果进行排序。

调整:上面得到的检索结果集是经过条件扩展的具有类别标记的检索结果集。由于同一个资料可能属于多个类别,而且不同的检索条件得出的检索结果会有所重合,我们只在具有最大分类概率值的类别中保留重复的资料。最终按类别提交排序的检索结果。这种分类显示方式便于用户快速识别出有用的资料。

结束语 软件过程是软件组织中最主要的知识产生和应用活动,在这个过程中有大量的数据、信息和文档产生、传递和利用。针对软件过程知识资产应用数据挖掘技术、实施知识管理具有很大的现实意义。本文从组织实践管理的角度研究了软件过程中的知识管理,并通过对组织实践的挖掘来支持知识管理的实现。文章提出了协同知识管理模型,以满足软件过程中的知识管理需求。我们开发的基于 Internet 的协同知识管理系统 iKnow 实现了数据挖掘与知识管理的集成、知识管理和软件过程的集成。我们设计了可扩展的柔性的组织实践表示方法,以支持对组织实践的有效挖掘与管理。设计了一种新颖的用户(人员和虚拟小组)特征表示方法,以适应知识推荐与检索的需求。以领域知识和数据挖掘为基础,系统实现了自适应的集成知识服务、个性化的知识检索,对软件组织的显性知识和隐性知识进行了有序的管理。这些特点使系统区别于国外机构所研究的同类工具,为软件组织知识管理的实施提供了有效的支持。同时,系统的设计思想与实现技术也适用于开发其它应用领域的知识管理系统。

下一步的研究内容包括:对人员知识结构模型的学习,从而建立随人员需求而动态调整的自适应知识资产空间,便于

对知识源的快速重组,实现更具主动性和自适应性的知识服务;引入适当的具有增量学习能力的数据挖掘算法,以提高系统的性能;在系统中增加工程数据管理、分析和挖掘功能,以实现软件过程中问题的诊断和预测。

参考文献

- 1 Ruhe G, Bomarius F. Learning software organizations: methodology and applications. Lecture Notes in Computer Science, 2000, 1756: 4~19
- 2 Morasa S, Ruhe G. Knowledge discovery from software engineering measurement data: a comparative study of two analysis techniques. In: Proc of 9th Int. Conf. on Software Engineering and Knowledge Engineering, 1997. 450~458
- 3 Althoff K-D. Case-based Reasoning. Handbook of Software Engineering and Knowledge Engineering, Vol. 1, World Scientific Publishing Co, 2002. 549~587
- 4 Bergmann R, Stahl A. Similarity measures for object-oriented case representations. In: B. Smyth, P. Cunningham, eds. Lecture Notes in Artificial Intelligence, 1998, 1488: 25~36
- 5 Tautz C. Towards engineering similarity functions for software engineering experience. IESE-Rep No. 064. 99/E, 1999
- 6 Henninger S. Tool support for experience-based software development methodologies. Advances in Computers, 2003, 59: 29~82
- 7 Tautz C, Althoff K-D. A case study on engineering ontologies and related processes for sharing software engineering experience. In: Proc. of the 12th intl. conf. on software engineering and knowledge engineering, 2000
- 8 Hartmann J, Sure Y, et al. Extended OntoWeb.org Portal. 2003. [Online] <http://ontoweb.aifb.uni-karlsruhe.de/About/Deliverables/Deliverable6.4>
- 9 Paulk M C, Curtis B, et al. Capability maturity model for software, Version 1.1: [Technical Report: CMU/SEI-93-TR-024]. 1993
- 10 Abecker A, et al. Toward a technology for organizational memories. IEEE Intelligent Systems, 1998, 13(3): 40~48
- 11 Noy N F, Ferguson R W, Musen M A. The knowledge model of Protégé-2000: combining interoperability and flexibility. In: Proc of 2th Int. Conf. on Software Engineering and Knowledge Management, 2000. 17~32
- 12 Han Jianwei, Kamber M. Data mining: concepts and techniques. Morgan Kaufmann Publishers, Inc. 2001
- 13 曾春, 邢春晓, 周立柱. 基于内容过滤的个性化搜索算法. 软件学报, 2002, 14(5): 999~4004
- 14 Shafer G. A Mathematical Theory of Evidence. Princeton University Press, 1976
- 15 李雄峰. 基于数据挖掘的案例库维护技术: [硕士学位论文]. 中国科学院软件研究所, 2003
- 16 Wolverson M. Exploiting enterprise models for the automatic distribution of corporate information. In: Proc of 6th ACM Intl. Conf. on information and knowledge management (CIKM-97), ACM Press, 1997. 341~347
- 17 von Wangenheim C G, Althoff K-D, Barcia R M. Intelligent retrieval of software engineering experienceware. In: Proc of the 11th Int. Conf. on Software Engineering and Knowledge Engineering, 1999. 128~135