

一种基于语义的业务活动推荐方法

张龙 应时 贾向阳 龚致远 李琳

(武汉大学软件工程国家重点实验室 武汉 430072) (武汉大学计算机学院 武汉 430072)

摘要 以企业复杂业务逻辑为背景,针对动态构造业务流程中的用户遗漏或错选工作等问题,提出了一种基于语义的业务活动推荐方法。首先定义面向业务活动的基础本体模型并在此基础上构建了基本推理规则,接着详细描述了方法所采用的推荐策略并在此基础上设计了推荐算法。方法的工作机制是接收并分析业务活动事件并调用推荐算法,推荐算法访问知识库,生成最终的推荐结果呈现给用户。最后通过国家综合减灾应用系统案例来验证方法的有效性。

关键词 事件驱动,语义,推理规则,业务活动,推荐

中图分类号 TP311 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.09.035

Semantic-based Approach for Business Activities Recommendation

ZHANG Long YING Shi JIA Xiang-yang GONG Zhi-yuan LI Lin

(State Key Lab of Software Engineering, Wuhan University, Wuhan 430072, China)

(Computer School, Wuhan University, Wuhan 430072, China)

Abstract In a dynamic constructed business workflow, users always leave out the activities they should do or select the wrong activities they shouldn't do at the background of complex enterprise business logic. To solve this problem, a semantic-based approach for business activities recommendation was proposed. First we defined the foundation ontology model for business activities, and on the basis of the model built some basic reasoning rules. Then we described the recommendation strategy in detail, and designed the recommendation algorithm based on the strategy. The mechanism of the approach is receiving and analyzing business activity events to invoke the recommendation algorithm, then the algorithm accesses to knowledge base to generate the final recommended results to the end user. Finally, a case of state comprehensive disaster reduction system was studied to validate the proposed approach.

Keywords Event-driven, Semantic, Reasoning rules, Business activities, Recommendation

1 引言

随着社会信息化的不断发展,企业应用中的业务逻辑和数据依赖关系变得越来越复杂。传统的基于固定业务流程的方式不再适用,构造一种随需而变的动态业务流程成为一种行之有效的解决方案。但是构造随需而变的动态业务流程一直是企业应用软件开发的挑战性难题。目前的学术界和工业界解决该类问题常常采用基于工作流的方法。然而,在一些涉及复杂业务活动的企业中,由于存在的业务操作数量众多,使用这些业务操作构造的业务流程复杂多变,难以使用固定的业务流程来驱动业务执行。在工作中,用户常常需要自己根据当前的情况,人工选择应该做的工作。在这种情况下,如果应用系统能够根据当前的业务和用户的具体情况给出下一

步工作的合理推荐,就可以有效提高工作效率,避免遗漏重要工作或者启动错误的 workflows。本文将针对这种情况,提出一种基于语义的业务活动推荐方法。该方法可以根据用户的业务上下文、数据依赖关系动态推荐用户需要做的下一步工作,为用户执行业务任务提供决策支持。

本文中的推荐方法将基于语义网技术来完成。语义网是现有网络的一种扩展,其中的信息有定义完好的含意,利于人机之间的协同工作,计算机将根据关键名称定义的超链接和逻辑推理规则发现语义数据的含义^[1]。网络本体语言(Web Ontology Language, OWL)是语义网的一个组成部分,被设计用来处理资讯的内容而不是仅仅向人类呈现信息^[2]。

OWL语言在表达能力上具有一定的局限性,而规则却能够提供较强的逻辑表达能力,据此研究者提出了语义网规则

到稿日期:2013-11-18 返修日期:2014-03-20 本文受国家自然科学基金面上项目:基于框架的面向服务软件异常处理方法研究(61070012),国家自然科学基金重点项目:面向服务软件体系结构设计阶段异常处理的建模和验证(61272113),国家自然科学基金重点项目:软件异常的双路传播模型及其构建与分析方法(61272108),国家自然科学基金重点项目:可信软件构造理论与方法研究(91118003),国家自然科学基金面上项目:基于编译的嵌入式系统优化研究(61170022),海外及港澳学者合作研究基金:网络电子图书的自动编辑与维护(61028003),教育部高等学校学科创新引智计划:软件工程及导航信息网络科学与技术创新引智基地(B07037)资助。

张龙(1989-),男,硕士生,主要研究领域为语义网、服务计算和软件工程, E-mail: taihejin@whu.edu.cn; 应时(1965-),男,博士,教授, CCF 高级会员,主要研究领域为面向服务的软件工程、语义智能软件、高可信软件、云计算与物联网; 贾向阳(1972-),男,讲师, CCF 会员,主要研究领域为服务计算、中间件技术、高可信软件; 龚致远(1984-),男,博士生,主要研究领域为云计算、互联网上的软件工程和智能计算。

语言 SWRL^[3]。SWRL 具有较丰富的关系表达能力,并且采取了一种将本体和规则相分离的策略,用 SWRL 语言描述的规则可以很方便地转化为其他现存规则系统中的规则^[4]。

基于上述问题和事实,本文提出了一种基于语义的业务活动推荐方法。方法构建了一种领域无关的业务活动基础本体模型。在基础本体模型之上定义了相应的推理规则,使基础本体能够根据规则通过推理生成新的语义知识。之后详细讨论了推荐策略并据此设计了推荐算法,当用户完成某一操作时推荐算法会根据用户当前的操作和所处的上下文环境以及相应的数据依赖关系得出最终的用户推荐结果。

本文第 2 节分析国内外相关的研究方法;第 3 节给出基础本体模型;第 4 节说明建立在基础本体模型的推理规则;第 5 节详细讨论了推荐策略和推荐算法,并分析了算法时间复杂度;第 6 节通过减灾防灾系统案例来验证本文所提方法的有效性;最后做出总结并说明未来的研究工作。

2 相关工作

作为推荐系统的核心技术,推荐方法得到了国内外广泛的研究和关注。推荐方法通常分为 3 种:协同过滤(Collaborative Filtering)、基于内容的过滤(Content-based Filtering)和混合式过滤(Hybrid Filtering)^[5]。协同过滤方法是一种集体智慧的思想,利用当前用户或者其他用户对部分项目的已知偏好数据来预测当前用户对其他项目的潜在偏好^[6]。基于内容的过滤方法首先计算已知用户偏好和待预测项目的描述文档之间的匹配程度,然后按照偏好排序结果向用户推荐其可能感兴趣的项目^[7]。混合式过滤方法是将不同的推荐算法或推荐类型进行组合并生成推荐。

从广义上讲,基于语义的推荐方法属于基于内容的过滤方法的一种。目前,在基于语义的推荐方面已经有研究人员做了一些相关工作。文献[8]针对普通计算中的移动应用提出了一种基于规则的本体模型(RBO)以支持基于内容的逻辑推理,模型能够根据上下文环境做出相应的推荐。文献[9]提出了一种新的融合了协同过滤和语义推理的混合式推荐方法并成功将其运用到了电影推荐领域。该方法把各项功能模块化,使得可以在多视角提供准则以得到不同的推荐结果。文献[10]综述了有关上下文感知系统的研究进展,对其系统框架、关键技术、主要模型、效用评价以及应用实践等进行了前沿概况、比较和分析。文献[11]提出了一种基于内容和语义推理的推荐方法,并对用户的偏好进行建模,但是其仅能适合数字电视推荐而不能用于其他领域。

3 基础本体模型

本体的构建是一项基本任务。出于对特定问题和具体工程需求的考虑,构建本体的过程各不相同。但是,研究者在本体构建的实践过程中,归纳出一些本体构建准则。其中具有影响的是 Gruber 在 1995 年提出的 5 条准则^[12]:(1)清晰性;(2)一致性;(3)可扩展性;(4)编码偏好程度最小;(5)最小本体承诺。本文将按照这 5 条准则对业务活动领域内的本体进行构建。

定义 1 将业务活动基础本体 O_A 定义为一个五元组 $(C_A, V_A, H^C, R^C, RULE)$,其中,(1) C_A 为业务活动概念的集合;(2) V_A 为 C_A 中业务活动概念值的集合;(3) H^C 为 C_A 中

各概念之间层次关系的集合, $H^C \subseteq C_A \times C_A$ 是一种有向关系, $H^C(C_1, C_2)$ 表示 C_1 是 C_2 的子概念,其中 $C_1, C_2 \in C_A$ 是两个概念;(4) R^C 为 C_A 中两个概念之间的关系的集合;(5) $RULE$ 表示业务活动本体中的规则。

定义 2 概念集 $C_A = \{Work, Activity, Data, Event, Phase, Role, Workgroup, User, Department\}$,枚举了基础本体概念集中的所有元素。

定义 3 关系集 $R^C = \{hasRole, hasDepart, belongTo, hasData, hasPrivilege, hasInitdata, inPhase, inAct, hasWork, hasNext\}$,其中核心关系的具体含义如下:

(1)hasRole 关系。hasRole(User, Role)表示用户和角色之间的所属关系。记 $user' = \{x | x \text{ 是概念 User 的个体}\}$, $role' = \{x | x \text{ 是概念 Role 的个体}\}$,则 hasRole(Tom, Manager)表示用户 Tom 拥有 Manager 角色,其中 $Tom \in user'$, $Manager \in role'$ 。

(2)hasInitdata 关系。hasInitdata(Activity, Data)表示活动与数据类型之间的初始依赖关系。记 $activity' = \{x | x \text{ 是概念 Activity 的个体}\}$, $data' = \{x | x \text{ 是概念 Data 的个体}\}$,则 hasInitdata(ExtractFlood, RemotesensingMap)表示活动 ExtractFlood 在开始之前必须具有数据类型为 Remotesensing-Map 的数据,其中 $ExtractFlood \in activity'$, $Remotesensing-Map \in data'$ 。

(3)inPhase 关系。inPhase(Activity, Phase)表示活动与阶段之间的所属关系。记 $phase' = \{x | x \text{ 是概念 Phase 的个体}\}$,则 inPhase(ExtractFlood, Processing)表示活动 ExtractFlood 处于 Processing 阶段,其中 $Processing \in phase'$;同样地, inPhase(Data, Phase)表示数据与阶段之间的关系。本文做出约定,记 $event' = \{x | x \text{ 是 Event 的个体}\}$,对 $\forall x \in event'$,则 x 是在某一活动完成之后产生的,换句话说就是任何事件都是在活动完成之后产生。并且对 $\forall x \in event'$, x 包含有零个或多个数据,其中任何一个数据都有所处的阶段 Phase。在本体模型中 Event 个体包含的数据所处于的阶段与产生该事件的活动所处于的阶段相同。

(4)hasPrivilege 关系。hasPrivilege(Role, Activity)表示角色和活动之间的可执行关系。如, hasPrivilege(Manager, ExtractFlood)表示角色 Manager 具有执行活动 ExtractFlood 的权限。

业务活动基础本体模型创建了一种面向业务活动的环境,在这种环境中不同业务领域共享着公共的概念集合。所有具体的领域本体也必须根据本领域内业务逻辑的相关需求扩展基础本体而形成。业务活动基础总体模型如图 1 所示。

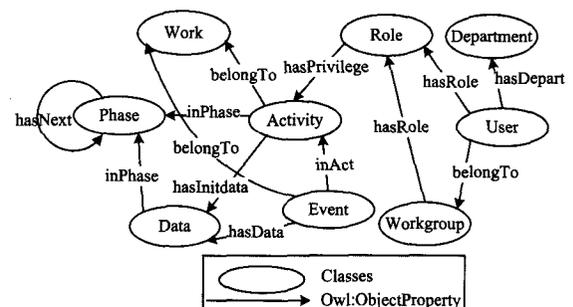


图 1 基础本体模型

4 推理规则

虽然本体能够面向具体领域建立丰富的语义信息,但其本身存在着不支持对属性的组合、内置的使用、本体中介和有限制的假设等问题。本节将利用 SWRL 规则语言来定义业务活动基础本体模型中的基本推理规则。

定义 4

$$RULE_1 = User(?user) \wedge Workgroup(?workgroup) \wedge Role(?role) \wedge hasRole(?workgroup, ?role) \wedge belongTo(?user, ?workgroup) \rightarrow hasRole(?user, ?role) \quad (1)$$

$RULE_1$ 所表达的含义是如果用户?user 属于工作组?workgroup 并且工作组?workgroup 拥有角色?role, 那么用户?user 也拥有角色?role。

定义 5

$$RULE_2 = Event(?evt) \wedge Activity(?act) \wedge Work(?work) \wedge inAct(?evt, ?act) \wedge belongTo(?act, ?work) \rightarrow belongTo(?evt, ?act) \quad (2)$$

$RULE_2$ 规定了若事件?evt 是在活动?act 完成之后产生的并且?act 是工作?work 中的一项活动, 那么事件?evt 是工作?work 中的事件。

定义 6

$$RULE_3 = Event(?evt) \wedge Activity(?act) \wedge Data(?data) \wedge Phase(?phase) \wedge inPhase(?act, ?phase) \wedge inAct(?evt, ?act) \wedge hasData(?evt, ?data) \rightarrow inPhase(?data, ?phase) \quad (3)$$

$RULE_3$ 说明事件所含数据的阶段与产生事件的活动所处的阶段相同。

上述 3 项规则是面向基础本体的基础规则, 任何扩展了基础本体的具体应用都受基本规则的约束。在建立了推理规则之后, 推理机根据规则通过语义推理产生新的语义知识。这些新的语义知识会被添加到原有知识库中从而达到丰富补充的作用。

5 推荐策略和推荐算法

5.1 推荐策略

推荐策略是指在不考虑用户偏好的情况下对项目列表排列顺序的产生所采用的策略。目前对推荐策略还没有形成统一的标准, 不同的推荐方法或推荐模型一般都是根据自身特点制定出符合自身逻辑的推荐策略, 本文提出以下推荐策略:

定义 7 $STRATEGY_1$ 在同一阶段内, 与事件数据相匹配的活动最优先推荐。

定义 8 $STRATEGY_2$ 在同一阶段内, 与事件集数据相匹配的活动次之。

定义 9 $STRATEGY_3$ 在同一阶段内, 若 $STRATEGY_2$ 中所推荐的活动已经推荐并且用户已经接受, 则应置于最后推荐。

定义 10 $STRATEGY_4$ 在下一阶段, 按照 $STRATEGY_1$ 、 $STRATEGY_2$ 、 $STRATEGY_3$ 生成阶段内推荐并将推荐结果置于上一阶段的推荐结果之后。

为了解释推荐策略的含义, 我们结合具体的场景来说明。图 2 中给出了 3 个阶段, 分别为 collection、preprocessing 和 processing。collection 阶段展示了事件及其包含的数据; preprocessing 和 processing 阶段展示了相关活动及其依赖的数据, 具体场景如下:

场景 1 在某时刻 T_1 处于 collection 阶段的某一活动完成并产生了 Evt_1 事件, Evt_1 包含 A 类型的数据。现在根据推荐策略 $STRATEGY_1$ 在 preprocessing 阶段中找符合数据依赖关系的活动, 图中只有 Act_1 符合要求, 下一阶段 processing 则没有符合要求的活动。最后推荐 Act_1 活动并假设用户接受了推荐。假设初始事件集 $EventSet = \emptyset$, 则现在 $EventSet = \{Evt_1\}$ 。

场景 2 在另一时刻 $T_2 (T_1 < T_2)$ 处于 collection 阶段的另一活动也完成并产生了 Evt_2 事件, Evt_2 包含 B 类型的数据。根据 $STRATEGY_1$ 在 preprocessing 阶段中找符合数据依赖关系的活动, 图中只有 Act_2 符合要求。现在事件集 $EventSet = \{Evt_1, Evt_2\}$, 据 $STRATEGY_2$ 用事件集 $EventSet$ 去匹配活动, 符合依赖关系的有 Act_1 和 Act_3。又根据 $STRATEGY_3$ 推荐策略, 由于 Act_1 在 Evt_1 事件发生时已经推荐并且用户已经接受, 因此 Act_1 应该置于 Act_3 之后。以同样的方式在 processing 阶段匹配活动发现 Act_4 符合要求(这个例子中 Act_4 所依赖的数据类型为 A 和 B, 并假设 A 和 B 处于 collection 阶段; 若不做出此假设, 在场景 2 中 Act_4 是不会匹配成功的), 根据 $STRATEGY_4$ 推荐策略, 应该把 Act_4 置于 preprocessing 的结果之后。综上所述, 最终的推荐结果为 Act_2、Act_3、Act_1、Act_4。

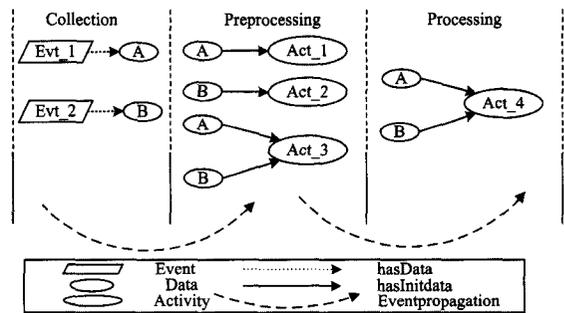


图 2 推荐策略示意图

上面介绍了方法所采用的 4 条推荐策略, 并结合例子说明了每条策略的具体含义。推荐策略是产生推荐的原则和依据, 推荐算法也是在此基础上设计的。

5.2 基本操作及概念

通过形式化定义基础操作及概念, 使其能够清晰明了地表达自身的含义。由于推荐算法会用到这些操作及概念, 因此本节内容是理解推荐算法的基础。

定义 11 (存取用户角色集 $GetRoles(User)$) 记 $user' = \{x | x \text{ 是 } User \text{ 的个体}\}$, $role' = \{x | x \text{ 是 } Role \text{ 的个体}\}$, 则 $GetRoles(u) = \{r | \exists hasRole(u, r), u \in user', r \in role'\}$ (4)

式(4)定义了所有和特定用户存在 hasRole 关系的角色集, 其 SPARQL 实现语句如下:

```
SELECT ?role WHERE {?u; hasRole ?role.}
```

定义 12 (存取当前阶段的下一个阶段 $Next(Phase)$) 记 $event' = \{x | x \text{ 是 } Event \text{ 的个体}\}$, $phase' = \{x | x \text{ 是 } Phase \text{ 的个体}\}$, 约定:

$$phase(e), e \in event' \quad (5)$$

式(5)表示事件 e 所处的阶段;

$$Next(p) = \{p' | \exists hasNext(p, p'), p, p' \in phase'\} \quad (6)$$

式(6)定义了和特定阶段存在 hasNext 关系的阶段集, 其

SPARQL 实现语句如下:

SELECT ?p' WHERE { ?p; hasNext ?p'. }

定义 13(存取某一阶段中特定角色拥有可执行权限的活动集 $GetActivities(Role, Phase)$) 记 $activity' = \{x | x \text{ 是 Activity 的个体}\}$, 则

$GetActivities(r, p) = \{a | \exists (hasPrivilege(r, a) \wedge inPhase(a, p)), r \in role', p \in phase', a \in activity'\}$ (7)

式(7)定义了特定阶段 p 中所有角色 r 具有可执行权限的活动的集合, 其 SPARQL 实现语句如下:

SELECT ?a WHERE { ?r; hasPrivilege ?a; ?a; inPhase ?p }

定义 14(存取活动所需的初始数据集 $GetInits(Activi-ty)$) 记 $data' = \{x | x \text{ 是 Data 的个体}\}$, 则

$GetInits(a) = \{d | \exists hasInitdata(a, d), a \in activity', d \in data'\}$ (8)

式(8)定义了特定活动 a 所需的所有初始数据集, 其 SPARQL 实现语句如下:

SELECT ?d WHERE { ?a; hasInitdata ?d. }

定义 15(数据集包含关系 \subseteq) 如果有 $\forall x_1 \exists x_2 (type(x_1) = type(x_2) \wedge phase(x_1) = phase(x_2))$, 那么有 $S_1 \subseteq S_2$, 其中 $S_1, S_2 \in data', x_1 \in S_1, x_2 \in S_2$. 约定:

$dataSet(e), e \in event'$ (9)

表示事件 e 所包含的数据集。

基本操作和基本概念的形式化定义是理解推荐算法的前提, 下一小节介绍推荐算法。

5.3 推荐算法

算法上下文假设:

假设 1 事件所产生的历史数据集 $hiDataOfWork$ 已经初始化。

假设 2 用户已接受的历史推荐活动集 $hiActOfWork$ 已经初始化。

算法输入: 当前用户 $user$, 事件 $event$

算法输出: 所推荐的活动列表 $(act_1, act_2, \dots, act_n)$

Recommend($user, event$)

1. $recActs \leftarrow \emptyset$
2. $roles \leftarrow GetRoles(user)$
3. $curPhase \leftarrow Next(phase(event))$
4. WHILE $curPhase \neq NULL$ DO
5. $peActs \leftarrow \emptyset$
6. $pesActs \leftarrow \emptyset$
7. $hisActs \leftarrow \emptyset$
8. $candoActs \leftarrow \emptyset$
9. FOR All elements in $roles$ DO
10. $candoAct \leftarrow GetActivities(role, curPhase)$
11. $candoActs \leftarrow candoActs \cup candoAct$
12. FOR ALL elements in $candoActs$ DO
13. $actInits \leftarrow GetInits(act)$
14. IF $actInits \subseteq dataSet(event)$
15. $peActs \leftarrow peActs \cup act$
16. ELSE IF $actInits \subseteq hiDataOfWork$ AND $act \notin hiActOfWorks$
17. $pesActs \leftarrow pesActs \cup act$
18. ELSE IF $actInits \subseteq hiDataOfWork$ AND $act \in hiActOfWorks$
19. $hisActs \leftarrow hisActs \cup act$
20. $recActs \leftarrow peActs \cup pesActs \cup hisActs$
21. $curPhase \leftarrow Next(curPhase)$
22. RETURN $recActs$.

算法的复杂度与知识库规模、历史数据集和历史活动集的规模有关。设本体知识库的规模为 m , 历史数据集的规模为 x , 历史活动集的规模为 y , 并设 $n = \min\{x, y\}$, 则此算法的时间复杂度近似为 $O(mn)$ 。从具体实现上看, 算法严格遵循推荐策略。

6 案例研究

本案例来自国家民政部综合减灾空间信息服务应用系统的运行管理部分, 项目目标是当灾害将要到来或已经发生时, 协调各机关单位的协同工作, 准确地收集掌握灾害的实时信息以便做出正确的决策, 及时发布权威信息并最大程度地减少灾害地区的损失。项目中的业务逻辑、数据依赖以及用户权限控制多而繁杂, 如何解决这些问题成为一项重要而棘手的任务。本文所提出的业务活动推荐方法成功应用在了这一项目中并在一定程度上解决了上述问题。为了使案例简明扼要, 这里只介绍相关重点部分而淡化其他不相关的内容。

6.1 领域本体

领域本体是根据领域内业务逻辑的具体需要在基础本体之上扩展而得到的, 因而领域本体具有一定的独特性。本案例扩展的领域本体如下:

阶段 Phase 其子类有 Collection, Preprocessing, Processing, Evaluation, Publish, 代表的含义分别为数据采集、数据预处理、数据处理、综合评估和信息发布, 并且各阶段之间具有 hasNext 关系。

数据 Data 其子类有 LiveData, SatelliteData, AirData, 代表的含义分别为现场数据、卫星数据、航空数据。数据 Data 和阶段 Phase 之间具有 inPhase 关系。

活动 Activity 其子类有 Collecting, ImageSubset, FormatConvert, ExtRoad, ExtRiver, ExtHouse, ExtFlood, Evaluating, Publishing, 代表的含义分别为数据采集、图像裁剪、格式转换、道路提出、河流提取、房屋提取、洪水提取、综合评估、信息发布。活动 Activity 和阶段 Phase 之间具有 inPhase 关系。

领域本体之间的关系如图 3 所示。数据采集阶段的下一步是数据预处理阶段, 接着是数据处理阶段, 然后是综合评估阶段, 最后是信息发布阶段。阶段对活动起到约束作用, 一般一个阶段内的活动完成之后才能进行下一个阶段的活动, 也有跨阶段的情况, 但是总的先后顺序是不变的。图中的 Collecting 属于数据采集阶段, ImageSubset 和 FormatConvert 属于数据预处理阶段, ExtRoad, ExtRiver, ExtHouse 和 ExtFlood 属于数据处理阶段, Evaluating 属于综合评估阶段, Publishing 属于信息发布阶段。

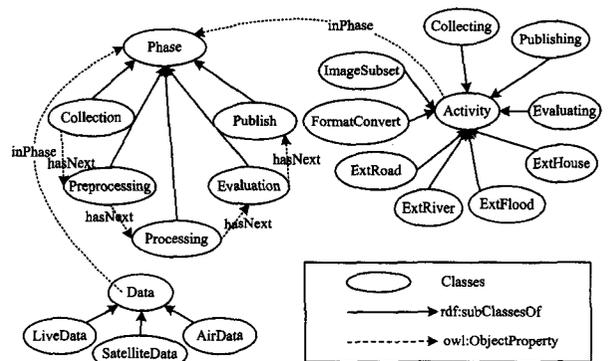


图 3 减灾系统的部分领域本体

6.2 领域个体

在语义网中个体代表了类的一个实例,主要是根据具体业务逻辑来建立个体。所有的个体及其关系构成本案例的知识库,其中部分个体的 OWL 描述如下(Turtle 格式):

```
sklse:collection rdf:type sklse:Collection,
    sklse:hasNext sklse:preprocessing.
sklse:convertA rdf:type sklse:FormatConvert,
    sklse:hasInitdata sklse:air_collect;
    sklse:inPhase sklse:preprocessing.
sklse:convertB rdf:type sklse:FormatConvert,
    sklse:hasInitdata sklse:live_collect;
```

```
sklse:inPhase sklse:preprocessing.
sklse:convertC rdf:type sklse:FormatConvert,
    sklse:inPhase sklse:preprocessing;
    sklse:hasInitdata sklse:sate_collect.
.....
```

6.3 实验验证

实验目的是在现有知识库的基础上设计相应的事件,通过分析推荐结果来验证方法的有效性。为此我们设计了一些包含不同数据类型并处于不同阶段的事件,对于本案例的知识库来说这些事件足以测试方法的有效性。实验结果如表 1 所列,其中“ \sqrt{N} ”表示第 N 个被推荐到的活动。

表 1 实验结果

阶段	事件及其数据	活动集							
		A1	A2	A3	A4	A5	A6	A7	A8
Collection	evt1	satelliteData \wedge P1	$\sqrt{1}$	$\sqrt{2}$					
	evt2	liveData \wedge P1	$\sqrt{1}$	$\sqrt{2}$			$\sqrt{3}$		
	evt3	airData \wedge P1	$\sqrt{1}$	$\sqrt{2}$					
Preprocessing	evt4	(AirData, LiveData) \wedge P2			$\sqrt{1}$	$\sqrt{2}$	$\sqrt{3}$		
	evt5	(SatelliteData, LiveData) \wedge P2			$\sqrt{1}$	$\sqrt{2}$	$\sqrt{3}$	$\sqrt{4}$	
	evt6	(SatelliteData, AirData) \wedge P2			$\sqrt{2}$	$\sqrt{3}$	$\sqrt{4}$	$\sqrt{1}$	
	evt7	(LiveData, AirData, SatelliteData) \wedge P2			$\sqrt{1}$	$\sqrt{2}$	$\sqrt{3}$	$\sqrt{4}$	
Processing	evt8	(LiveData, AirData, SatelliteData) \wedge P3						$\sqrt{1}$	
Evaluation	evt9	(LiveData, AirData, SatelliteData) \wedge P4							$\sqrt{1}$

注:表中 \wedge 表示 inPhase 关系;P1/P2/P3/P4 分别表示阶段 Collection/Preprocessing/Processing/Evaluation;A1/A2/A3/A4/A5/A6/A7/A8 分别表示活动 ImageSubSet/FormatConvert/ExtRoad/ExtRiver/ExtHouse/ExtFlood/Evaluating/Publish.

从表 1 实验结果可以看出,算法可以有效推荐出所有下一步可能进行的活动并且推荐顺序也符合推荐策略的规定。

上面实验仅仅验证了方法的有效性,并没有研究方法的性能。从推荐的及时性方面来说,推荐方法必须具有较高的性能。上面实验使用的知识库规模相对较小,方法在复杂大规模知识库上的性能还需要进一步实验分析。下面用控制变量法进行实验分析。

保持知识库规模一定,连续多次调用推荐算法得到冷启动时间和算法执行时间曲线(冷启动时间指将外存文件或数据库中的知识库加载到内存所需的时间;算法执行时间是指推荐算法访问内存知识库到返回推荐结果所需的时间)。保持其他条件不变,在扩大知识库规模的基础上按照上述方法进行多次实验,得到的实验结果如图 4、图 5 所示。

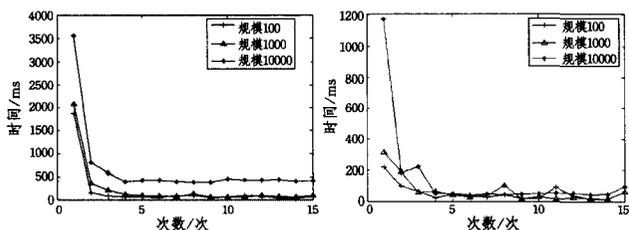


图 4 冷启动时间

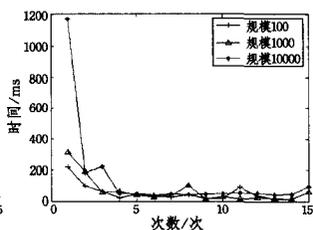


图 5 算法执行时间

从实验结果中可以看出初次冷启动和调用推荐算法都需要相对较长的时间,随着次数的增加所需时间会锐减并趋于稳定。在不同规模的知识库上也表现出相同的规律,但是在大规模知识库上所需的时间比小规模的时间有所增加。

将多次实验的结果取平均值绘制成如图 6、图 7 所示的曲线。分析结果可以得出平均冷启动时间和算法平均执行时间都近似呈线性增长,但后者在线性范围内有所波动。在推荐算法部分已经介绍了算法的时间复杂度为 $O(mn)$,即算法

的执行时间不仅仅受知识库规模的影响,历史事件集或历史活动集的大小都可能对算法的执行时间造成影响。这就导致了算法的平均执行时间在线性增长的范围内有波动的事实。

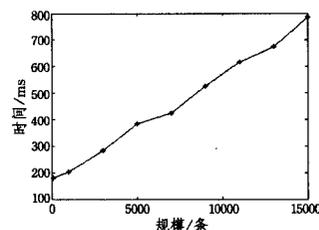


图 6 平均冷启动时间

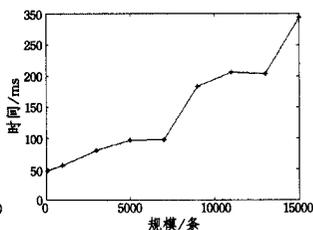


图 7 算法平均执行时间

综合上述实验可以得出结论:1)本文所提的业务活动推荐方法是有效的;2)方法具有较高的性能,可以满足及时性推荐的要求。

结束语 本文提出的基于语义的业务活动推荐方法,定义了面向业务活动的基础本体模型并构建了基本推理规则,详细讨论了推荐策略并给出了推荐算法。实验结果表明,该方法对业务活动的推荐是有效的。在动态构造的业务流程中,方法可以辅助用户决策并做出下一步的选择,在一定程度上解决了用户遗漏或错选工作的问题。该方法具有一定的普适性和较高的可扩展性,可以方便地将方法应用到不同的领域。另外方法在面向服务的计算环境中的应用未做详细的讨论,此问题将作为后续研究工作。

参考文献

- [1] Li Ding, Kolar P, Ding Zhong-li, et al. Using Ontologies in the Semantic Web; A Survey[J]. Ontologies Integrated Series in Information Systems, 2007, 14: 79-113

(下转第 195 页)

结束语 本文针对道路网环境,考虑用户隐私保护和定位设备的精度问题,研究了基于范围的道路网 skyline 查询,提出了两种有效的查询算法,并通过实验对其查询性能进行比较分析。B-RNS 算法通过边界点替换法将 RNS 查询转化为基本的 PNS 查询,降低了复杂性,但仍有大量复杂的道路距离计算。针对这一问题,提出了一种道路网 skyline 模型,并基于模型设计了能有效地支持 RNS 查询的道路网 skyline 索引 SSR-tree,提出了基于索引的查询算法 I-RNS,经过实验验证,其提高了查询效率和查询结果的精度。

参 考 文 献

[1] Borzoyani S, Kossmann D, Stocker K. The Skyline Operator [C] // Proceedings of the 17th International Conference on Data Engineering (ICDE), 2001:421-430

[2] 周傲英,杨彬,金澈清,等. 基于位置的服务:架构与进展 [J]. 计算机学报,2011,34(7):1155-1171

[3] Tan K-L, Eng P-K, Ooi B-C. Efficient progressive skyline computation [C] // Proceedings of the 27th International Conference on Very Large Data Bases (VLDB), 2001:301-310

[4] Kossmann D, Ramsak F, Rost S. Shooting Stars in the Sky: An Online Algorithm for Skyline Queries [C] // Proceedings of the 28th International Conference on Very Large Data Bases (VLDB), 2002:275-296

[5] Papadias D, Tao Y-F, Fu G, et al. An Optimal and Progressive Algorithm for Skyline Queries [C] // Proceedings of SIGMOD, 2003:467-478

[6] Papadias D, Tao Y-F, Fu G, et al. Progressive Skyline Computation in Database System [J]. ACM Transactions on Database System (TODS), 2005, 30(1):41-82

[7] 付世昌,董一鸿,陈华辉,等. 基于道路网不确定移动对象的连续概率 skyline 查询 [J]. 计算机科学,2011,38(7):152-156

[8] Sharifzadeh M, Shahabi C. The Spatial Skyline Queries [C] // Proceedings of the 32nd International Conference on Very Large Data Bases (VLDB), 2006:751-762

[9] Lin X, Xu J-L, Hu H-B. Range-Based Skyline Queries in Mobile Environments [J]. IEEE Transactions on Knowledge and Data Engineering (TKDE), 2013, 25(4):835-849

[10] Deng K, Zhou X-F, Shen H-T. Multi-source Skyline Query Processing in Road Networks [C] // Proceedings of the 23rd International Conference on Data Engineering (ICDE), 2007:796-805

[11] Huang Y-K, Chang C-H, et al. Continuous Distance-based Skyline Queries in Road Networks [J]. Information Systems (IS), 2012, 37(7):611-633

[12] Hu L, Jing Y, Ku W-S, et al. Enforcing K-Nearest Neighbor Query Integrity on Road Networks [C] // Proceedings of the 20th ACM SIGSPATIAL/GIS, 2012:422-425

[13] Ku W-S, Zimmermann R, Peng W-C, et al. Privacy Protected Query Processing on Spatial Networks [C] // Proceedings of ICDE Workshop Privacy Data Management, 2007:215-220

[14] Xu J, Tang X, Hu H, et al. Privacy-Conscious Location-Based Queries in Mobile Environments [J]. IEEE Transactions on Parallel and Distributed Systems, 2010, 21(3):313-326

[15] <http://www.bbbike.de/cgi-bin/bbbike.cgi>

(上接第 189 页)

[2] Wang T D, Parsia B, Hendler J, et al. A Survey of the Web Ontology Landscape [J]. Lecture Notes in Computer Science, 2006, 4273:682-694

[3] Ma Li, Yu He-long, Chen Gui-fen, et al. Research on Construction and SWRL Reasoning of Ontology of Maize Diseases [J]. Computer and Computing Technologies in Agriculture VI IFIP Advances in Information and Communication Technology, 2013, 393:386-23

[4] Guerrero A, Villagra V A, de Vergara J E L, et al. Ontology-Based Policy Refinement Using SWRL Rules for Management Information Definitions in OWL [J]. Large Scale Management of Distributed Systems Lecture Notes in Computer Science, 2006, 4269:227-232

[5] Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions [J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17:734-749

[6] Hurrell E, Smeaton A F. A Conversational Collaborative Filtering Approach to Recommendation [J]. Advances in Visual Informatics Lecture Notes in Computer Science, 2013, 8237:13-24

[7] Martin G H, Schockaert S, Cornelis C, et al. An Exploratory Study on Content-Based Filtering of Call for Papers [J]. Multidisciplinary Information Retrieval Lecture Notes in Computer Science, 2013, 8201:58-69

[8] Liu Xiao-qin, Mo Xiu-liang, Wang Chun-dong, et al. A Rule-Based Ontology for Context-Aware Computing [C] // International Conference on Biomedical Engineering and Informatics, 2009. Tianjin: IEEE Computer Society, 2009:1-4

[9] Lemdani R, Bennacer N, Polaillon G, et al. A collaborative and semantic-based approach for recommender systems [C] // International Conference on Intelligent Systems Design and Applications (ISDA), 2010. Cairo: IEEE Computer Society, 2010:469-476

[10] 王立才,孟祥武,张玉洁. 上下文感知推荐系统 [J]. 软件学报, 2012, 23(1):1-20

[11] Blanco-Fernandez Y, Pazos-arias J, Gil-Solla A, et al. Providing entertainment by content-based filtering and semantic reasoning in intelligent recommender systems [J]. IEEE Transactions on Consumer Electronics, 2008, 54:727-735

[12] 支丽平,王恒山,张楠. 专利领域本体的构建方法研究 [J]. 图书情报工作, 2010, 8:116-119