

计算本体映射纯语义查准率和查全率的框架^{*}

徐德智 Dwight G. R. 陈建二

(中南大学信息科学与工程学院 长沙 410083)

摘要 要评价和比较不同的本体映射算法,就要用它们分别计算一组本体对之间的映射关系(其结果也叫做映射),并与一组参考映射进行比较,因此需要一组标准来评价映射结果与参考映射的相近程度。已有的本体映射评价标准都有各自的局限性,而且它们都不是纯语义的。举例指出了过去评价标准的缺陷,之后在分布式描述逻辑的基础上,提出了一个计算本体映射纯语义查准率和查全率的框架,并证明了在该框架内实现的查准率和查全率都必然是纯语义的。

关键词 本体映射,评价框架,纯语义,查准率,查全率

Framework for Computing Purely Semantic Precision and Recall of Ontology Mapping

XU De-zhi Dwight G. R. CHEN Jian-er

(School of Information Science and Engineering, Central South University, Changsha 410083, China)

Abstract To evaluate and compare various algorithms of ontology mapping, it is indispensable to apply them to a group of test ontology pairs, and compare the results (named mappings/alignments) with some references, so a group of criteria is needed to evaluate the closeness of results. Former measures all have their own limitations, and do not rely purely on the semantics of ontologies and alignments. The disadvantages of former measures were demonstrated through some examples, then based on distributed description logics, a framework for computing purely semantic precision and recall of ontology mapping was proposed, which was proved to be able to guarantee that measures in it are purely semantic.

Keywords Ontology mapping, Evaluation framework, Purely semantic, Precision, Recall

1 引言

语义 Web^[1]是对当前 Web 的扩充,它赋予 Web 上的资源以机器可理解的语义信息,使得机器可以自动对 Web 上的资源进行处理,提高人们使用 Web 的效率。在这一领域中,本体映射是一个重要的问题,它有很多本体映射算法^[2,3]。给定一对本体,每个本体映射算法返回一个映射。映射是两本体间实体(概念、属性或实例)对应关系的集合。

目前应用最广泛的评价标准是源自信息检索领域的查准率和查全率^[4],分别为算法返回的正确对应关系数量 $|R \cap A|$ 和算法返回对应关系数量 $|A|$ 的比值,以及 $|R \cap A|$ 与参考映射中对应关系(正确对应关系)数量 $|R|$ 的比值,即分别为 $|R \cap A| / |A|$ 与 $|R \cap A| / |R|$,其中 A 是算法返回的映射结果, R 是参考映射。它们可以用于评价任何情况下的本体映射,但它们的缺陷是只考虑了结果中完全正确的对应关系,而忽略了近似正确的那些。

文献[2]提出了一个框架,以计算松弛的查准率和查全率。该框架只能评价 1:1 映射,以保证计算得到的查准率和查全率不会大于 1。在这个框架中,松弛的查准率和查全率分别为 $\omega(A, R) / |A|$ 和 $\omega(A, R) / |R|$,其中 $\omega(A, R)$ 取代了标准查准率和查全率中的 $|R \cap A|$ 。之后文献[2]提出了 4 组松弛的查准率和查全率,其中第一组就是标准查准率和查全率(它们与该框架相容)。另外 3 组松弛的查准率和查全率

都考虑了近似正确的对应关系(但仍然只考虑了非常相近的,例如某概念被映射到其正确对应概念的直接子概念或直接父概念),而且它们都不是纯语义的(第 3 节的例子中有说明)。

文献[2]提出的框架对本体和映射结果中语义信息的利用仍然非常有限。文献[5]用 $M(o)$ 来表示本体 o 模型的集合,用 Γ 表示所有均一化函数的集合(其中均一化函数的作用是将每个本体论域中的元素映射到一个全局论域上)。之后该文献定义了两个本体 o, o' 和映射 A 的全局模型——三元组 $[m, m', \gamma] \in M(o) \times M(o') \times \Gamma$,并定义了 $Cn(A)$ 为 A 的所有 α -consequence 的集合(α -consequence 为 o, o' 和 A 的所有全局模型都蕴涵的对应关系),然后分别定义语义查准率和查全率为 $|A \cap Cn(R)| / |A|$ 和 $|R \cap Cn(A)| / |R|$ 。它们利用了本体和映射中的语义信息,然而其适用性受到了一定限制,即使两个本体本身都满足一致性, A 中的某一个或一些错误的对应关系也可能不允许一个可满足的全局模型存在(这在实际中可能经常发生,例如第 3 节里, A_3 中的对应关系 $4WD \geq Front\ wheel\ drive$),此时 $Cn(A)$ 为两个本体间任意可能对应关系的集合,依照公式,语义查全率为 1,这显然违背直觉。此外语义查准率和查全率也并不是纯语义的(第 3 节的例子中有说明)。

本文举例指出了过去评价标准的缺陷,之后基于分布式描述逻辑提出了一个计算本体映射纯语义查准率和查全率的框架,并证明了该框架内实现的查准率和查全率都是纯语义

^{*} 本课题获国家自然科学基金重点项目(60433020)资助。徐德智 教授,研究方向为 Web 计算、语义 Web; Dwight G. R. 硕士生,研究方向为本体映射;陈建二 教授,博导,研究方向为计算优化、语义 Web。

的。

2 背景知识介绍

2.1 映射的定义

文献[4]对映射的定义如下：

定义 1 给定两个本体 O_1 和 O_2 ，那么 O_1 和 O_2 之间的一个映射是一组对应关系(四元组)的集合。四元组为 $\langle e_1, e_2, r, n \rangle$ ， e_1 和 e_2 分别为 O_1 和 O_2 中被映射的两个实体， r 是它们之间的关系， n 取值在 $[0, 1]$ 之间，表示该对应关系的可信度。

如同文献[5]，本文只考虑四元组 $\langle e_1, e_2, r, n \rangle$ 的前 3 个元素，也就是将它当作三元组 $\langle e_1, e_2, r \rangle$ 来对待，其中 e_1 和 e_2 的关系 $r \in \{=, \geq, \leq\}$ 。概念和概念之间以及属性和属性之间， $=, \geq$ 和 \leq 分别表示等价、包含和被包含的关系。实例和实例之间，它们分别表示等价、整体-部分和部分-整体关系。要对包含可信度的映射结果进行评价，最好的方法就是绘制查准率/查全率图[5]。

2.2 分布式描述逻辑中映射的语义

目前本体映射主要作用是为本体查询和本体集成提供支持。语义 Web 的逻辑基础是描述逻辑[6]，分布式描述逻辑(Distributed Description Logics, DDL)[7] 是它在分布式环境下的扩展，为映射提供了语义。在 DDL 的架构中，第 i 个本体和第 j 个本体之间，概念对应关系和属性对应关系都被看作桥规则，分为两类：into 桥规则，形式为 $i: C \xrightarrow{\subseteq} j: D$ ，对应着 D 包含 C 的对应关系；onto 桥规则，形式为 $i: C \xrightarrow{\supseteq} j: D$ ，对应着 C 包含 D 的对应关系(C 和 D 等价的对应关系是相应的一个 into 桥规则和一个 onto 桥规则的组合)。

源本体 O_s ，目标本体 O_t 和它们之间的映射 A 共同构成一个知识库，可表示为二元组 $\langle \mathfrak{A}, \mathfrak{A} \rangle$ ，其中 \mathfrak{A} 是一个分布式 T-Box，而 \mathfrak{A} 是一个分布式 A-Box。DDL 中，一个 onto 桥规则 $i: C \xrightarrow{\supseteq} j: D$ 表示 O_s 中的 C 和 O_t 中的 D 之间有 C 包含 D 的关系。因此，如果所有满足 \mathfrak{A} 的分布式解释 \mathfrak{S} 都 d -满足 $i: C \xrightarrow{\supseteq} j: D$ ，则对分布式 T-Box \mathfrak{A} 有 $\mathfrak{A} \models i: C \xrightarrow{\supseteq} j: D$ ，表示在映射 A 中， O_s 中的 C 和 O_t 中的 D 之间有 C 包含 D 的关系[7]。这样当一个针对 O_t 要求获得所有 C 的实例(或者实例对，如果 C 是属性)的查询要求被提交后，每个 O_s 中所有有这样的 D 的实例(实例对)也会与 O_t 中对 C 的查询结果一并被返回。这使得用户可以提交一个查询而得到所有本体中的对应信息。

从 DDL 的角度来看，一个映射 A 将 O_s 中每个概念/属性 C 映射到 O_t 中一组概念/属性的并(其结果很可能是一个复合概念/映射)。由于实例之间不像概念之间或属性之间那样存在继承关系， O_s 每个实例 e_s 被映射到一个由 O_t 中实例构成的实例集。

文献[5]使用均一化函数将每个本体论域中的每个元素映射到一个全局论域中的一个元素，因此需要有至少一个可满足的全局模型，才能正常进行下一步计算。当全局模型存在时，其语义与 DDL 提供的语义相同，而 DDL 并不要求可满足的全局模型，适用性更广泛。本文提出的框架以 DDL 作为逻辑基础，因此可用于评价任意两个自身一致的本体之间的任意映射。

3 映射实例和已有评价标准值的计算

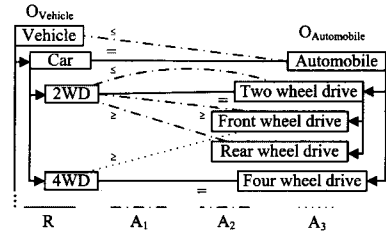


图 1 两个本体之间的 4 个映射

3.1 映射实例

图 1 中是两个本体之间的一组映射。为了简单起见，图中两个本体都只是概念层次树。图 1 中包括两个本体 $O_{Vehicle}$ 和 $O_{Automobile}$ ，以及它们之间的 4 个映射， R, A_1, A_2 和 A_3 。 R 是参考映射。图 1 左下角 3 个点表示概念“Vehicle”只有一个直接子概念“Car”，但“Vehicle” \supseteq “Car”。除了概念“Vehicle”，每个概念都恰好等于其所有子概念的并(如果它有子概念的话)。

这些映射可表示为如下对应关系集合：

$$R = \begin{cases} Car = Automobile & 2WD = Two\ wheel\ drive \\ 4WD = Four\ wheel\ drive \end{cases}$$

$$A_1 = \begin{cases} Vehicle \leq Automobile \\ 2WD = Two\ wheel\ drive \end{cases}$$

$$A_2 = \begin{cases} Vehicle \leq Automobile & 2WD \leq Two\ wheel\ drive \\ 2WD \geq Front\ wheel\ drive & 2WD \geq Rear\ wheel\ drive \end{cases}$$

$$A_3 = \begin{cases} Car = Automobile & 2WD = Two\ wheel\ drive \\ 4WD \geq Front\ wheel\ drive \end{cases}$$

3.2 已有评价标准值的计算

表 1 已有评价标准的值

P&R	standard		symmetric		semantic	
	P	R	P_w	R_w	P_{sem}	R_{sem}
R	1.00	1.00	1.00	1.00	1.00	1.00
A ₁	0.50	0.33	0.63	0.42	0.50	0.33
A ₂	0.00	0.00	0.19	0.25	0.75	0.33
A ₃	0.67	0.67	0.67	0.67	0.67	1.00

表 1 给出了图 1 中 4 个映射在已有评价标准下的得分。文献[2]中 3 组松弛的查准率和查全率之间彼此极为相似，所以本文只用其中“symmetric”一组作为代表。

文献[5]指出了语义等价的映射语法表述上可能并不完全一致，这种情况下，它们评价得分应该是一样的，或者它们得分的差别至少比和它们和其余语义不与之等价的映射之间的差别要小。直觉上来说，如果说 m 是一个纯语义的评价标准，等价于说，如果 A 和 A' ， R 和 R' 分别等价，则 A 被 A' 代替，和/或 R 被 R' 代替后， m 值不会变，即 $m(A, R) = m(A', R)$ ， $R = m(A, R') = m(A', R')$ 。

如同本文之前指出的一样，已有评价标准都未能给语义等价的 A_1 和 A_2 以同样的分数，而且 A_3 的语义查全率是 1，明显违反直觉，因为 O_1, O_2 和 A_3 的可满足全局模型不存在，仅仅因为一个错误的对应关系 $4WD \geq Front\ wheel\ drive$ 。

4 用于计算本体映射中纯语义查准率和查全率的框架

为解决已有评价标准都不是纯语义的问题，本文提出了

一个用于计算本体映射中纯语义查准率和查全率的框架。它计算本体中每个实体的纯语义查准率和查全率,再求它们的加权平均,以计算整体评价结果。

4.1 预处理

在应用本文的框架之前,先要对本体和映射进行一些预处理,以保证本体同义异构的情况不会影响最终评价结果。

预处理的第一个步骤是保证待映射本体都是一致的。因为框架内的评价应该是纯语义的,需要用到推理技术,那么本体应该都是一致的,否则推理失去意义。实际应用中,一般也都要求本体是一致的,因此这对本文所提出框架的适用性影响不大。可以先消除本体中的不一致^[8],然后再对这些处理后的本体进行映射。这部分的预处理操作在应用映射算法进行计算之前进行。

预处理的第二个步骤是去除冗余实体,同一本体内,语义相同名字不同的实体只留下一个。因为它们同义不同名,去掉哪个名字对映射的语义并没有影响,所以这个操作能完整保留本体和映射的语义。这部分的预处理操作在应用映射算法之后,计算评价结果前进行。

在本文余下的部分里,所有的本体和映射默认都是经过了如上两个预处理步骤的。

4.2 分块函数

每个本体由3部分组成:概念、属性和实例,本体之间的映射也可以对应划分为3个部分。已有评价标准都是将3个部分混在一起进行计算,这样就无法比较各个算法在本体不同部分的表现。在本文的框架中,这3个方面是分开评价的。对映射的总体评价可以通过计算3个部分评价值的加权平均而得到。本文定义分块函数如下:

定义2 分块函数接受本体 O 或映射 A , 及类型参数 T ($T \in \{“c”, “p”, “i”\}$, 分别代表概念、属性、实例), 返回本体或映射中对应类型参数 T 的部分。给定 T , $Part(O, T)$ 返回 O 中相应类型实体的集合, 而 $Part(A, T)$ 返回 A 中相应类型的对应关系的集合。

4.3 计算纯语义查准率和查全率的框架

根据 DDL 的语义, 本文给出对应实体函数的定义如下:

定义3 对应实体函数接受源本体中一个实体作为参数, 返回根据映射中 onto 桥规则, 该实体在目标本体中的对应实体。假设源本体为 O_s , 目标本体为 O_t , 分布式 T-Box 为 \mathfrak{A} , 分布式 A-Box 为 \mathfrak{A} , 映射为 A , 对应实体函数的定义如下:

$$M_A(e_s) = \begin{cases} \bigcup_{e_t \in O_t, \mathfrak{A}_A | =_{\mathfrak{A}} O_s : e_s \stackrel{?}{=} O_t : e_t} e_t & (\text{当 } e_s \text{ 为概念/属性时}) \\ \{e_t | e_t \in O_t, \mathfrak{A}_A | =_{\mathfrak{A}} O_s : e_s \geq O_t : e_t\} & (\text{当 } e_s \text{ 为实例时}) \end{cases}$$

该函数为 O_s 中概念/属性返回一个 O_t 中的概念/属性(很可能为复合概念/属性), 为 O_s 中实例返回一个 O_t 中的实例集合。

在实际应用中, 可能 O_1 是源本体, O_2 是目标本体, 或者刚好相反, 所以本文给出语义等价的定义如下:

定义4 对两个本体 O_1 和 O_2 , 以及映射 A 和 A' 来说, 如果对 O_1 中每个概念或属性 e_1 , 都有 $O_2 | = M_A(e_1) = M_{A'}(e_1)$, 而对 O_2 中每个概念或属性 e_2 , 都有 $O_1 | = M_A(e_2) = M_{A'}(e_2)$, 而对所有 O_1 和 O_2 中的实例 i , 有 $M_A(i) = M_{A'}(i)$, 那么 A 和 A' 是语义等价的。

当 O_1 和 O_2 互换后, into 桥规则和 onto 桥规则也随之互

换。由于只有 onto 桥规则对源本体中实体的查询结果有影响, 那么能影响查询结果的对应关系也是不同的。而在本体集成中, 两本体地位相当, 因此有3种情况需要考虑: 多本体查询中两种情况(O_1 和 O_2 分别为源本体), 以及本体集成这种情况。

假设参与映射的两个本体分别为 O_1 和 O_2 , 待评价映射为 A , 参考映射为 R 。要计算 $Part(A, T)$ 的纯语义查准率和查全率, 可以将映射 A 的纯语义查准率和查全率定义为 O_s 中每个实体的纯语义查准率 $SP(e_s)$ 和查全率 $SR(e_s)$ 的加权平均结果。为确保它们的确是纯语义的, 有以下几个约束条件:

1) 映射的纯语义查准率和查全率取值应该在 $[0, 1]$ 之间, 所以规定 O_s 中实体的纯语义查准率和查全率取值必须在 $[0, 1]$ 之间, 而且实体相应的权重函数 $w_P(e_s)$ 和 $w_R(e_s)$ 必须非负。

2) 对任何 $e_s \in O_s$, 如果 $M_A(e_s)$ 和 $M_R(e_s)$ 都等价于空概念/属性, 那么 e_s 用于计算纯语义查准率和查全率的权重都必须为 0, 因为它对于映射的评价来说并不重要。

3) 实体的纯语义查准率和查全率应该是纯语义的, 所以如果映射 A 和 A' , R 和 R' 分别等价, 则当 A 被 A' 替换和/或 R 被 R' 替换后, 那么对每个 $e_s \in O_s$, 其纯语义查准率和查全率以及相应的权重函数都应该保持不变。

当实体的纯语义查准率和查全率函数以及权重函数给定后, 就可以计算映射的纯语义查准率和查全率了。

定义5 给定源本体 O_s (可能是 O_1 也可能是 O_2), 目标本体(另一个), 它们之间的映射 A 以及参考映射 R , 以及要评价的类型参数 T , 映射的纯语义查准率和查全率给定如下:

$$SP(O_s, O_t, A, R, T) = \frac{\sum_{e_s \in Part(O_s, T)} (SP(e_s) * w_P(e_s))}{\sum_{e_s \in Part(O_s, T)} w_P(e_s)}$$

$$SR(O_s, O_t, A, R, T) = \frac{\sum_{e_s \in Part(O_s, T)} (SR(e_s) * w_R(e_s))}{\sum_{e_s \in Part(O_s, T)} w_R(e_s)}$$

它们考虑的都是多本体查询背景下, O_s 为源本体的情况。要在本体集成背景下评价映射, 可以计算 O_1 分别为源本体和目标本体两种情况下, 两组纯语义查准率和查全率的加权平均值。

在计算中有可能出现分母为 0 的情况。本文中分母总是对应一个集合的基, 因此如果分母为 0, 那么表示其对应集合为空, 根据查准率和查全率的概念, 本文统一定义所有分母为 0 的分数值为 1。由于所有实体的纯语义查准率和查全率都在 $[0, 1]$ 内, 而且权重函数非负, 那么映射的纯语义查准率和查全率也必然在 $[0, 1]$ 内。

定理1 在该框架内实现的查准率和查全率都必然是纯语义的。

证明: 根据第3条约束, 如果映射 A 和 A' , R 和 R' 分别等价, 则当 A 被 A' 替换和/或 R 被 R' 替换后, 所有实体的纯语义查准率和查全率和权重函数都保持不变, 根据映射的纯语义查准率和查全率计算公式, 它们的值也必然相同, 所以必然满足 $m(A, R) = m(A', R) = m(A, R') = m(A', R')$ 。因此在该框架内实现的评价标准都必然是纯语义的。证毕。

结束语 本文基于分布式描述逻辑, 提出了一个用于计算本体映射中纯语义查准率和查全率的框架, 并证明了它能保证框架内的查准率和查全率都是纯语义的, 克服了已有本

(下转第 254 页)

(2) 映射参数

当用户接口的参数不能移植到远程节点时,即在远程节点没有对应的等价或转换类型,那么这些参数应该本地访问。这将通过类型匹配的方式把远程对象映射到本地参数来实现。在上面的例子中,操作 refreshDisplay 和 redisplay_all 就属于此种情况,为此需要在节点 R_node 上建立如下的类型匹配关系:

```
Hybrid_node :: DISPLAY → display {……}
```

这样,refreshDisplay 和 redisplay_all 之间捆绑可表达如下:

```
refreshDisplay:redisplay_all ( 1 : display ← DISPLAY ) ~
int_to_bool(RET)
```

其中,方法 redisplay_all 中第一个参数为一个对象,该对象将按照节点 R_node 上指定的类型匹配映射到 refreshDisplay 中的第一个参数。另外,方法 redisplay_all 的返回类型为整数,需经适配函数 int_to_bool 转换为 Boolean 类型。

(3) 多类型映射参数

在有些情况下,一种参数的类型可能由远程节点上的多种参数类型的组合来表达。在上面的例子中,R_node 节点的 IO-DEVICES 就属于此种情况,它对应节点 L_node 上的类型 Mouse,Keyboard 和 touchScreen 的组合。类型关系定义如下:

```
Hybrid_node :: IO_DEVICES → {keyboard@, mouse@,
touchScreen@}{…}
```

在上面的例子中,中间代理对象 IO_DEVICES 将被连接到节点 L_node 上的 Keyboard,Mouse 和 touchScreen 对象上。

4.2.2 接口适配

在定义类型之间的操作捆绑时,有一种情况是本地局部请求的功能由多个远程操作功能的“组合”来实现。为了实现请求接口与服务接口之间的匹配,所要做的工作包括从简单的操作组合到复杂的编程。TMSL 允许在操作捆绑的说明中定义简单的操作组合。比如节点 L_node 上的操作 windowSelected 可以通过节点 R_node 上的操作 get_Position 和 select_window 的嵌套组合来实现。操作捆绑定义如下:

```
windowSelected: select_Window
(WINDOW_CONTROL.get_Position (<2, 1, 3> : <key-
board, mouse, touchScreen>
← IO_DEVICES, 4) ) RET
```

在上面的定义中,首先在远程节点上调用方法 get-Position,其结果不返回调用节点,而是作为方法 select_Window 的第一个参数。

(上接第 130 页)

体映射评价标准的缺陷,同时保证了广泛的适用性。另外,根据分布式描述逻辑,我们借鉴了概念相似度的计算方法,提出了一组针对实体的纯语义查全率和查准率函数以及权重函数,实现了这个框架。由于篇幅所限,研究成果将另文发表。

参 考 文 献

- [1] Berners-Lee T, Hendler J, Lassila O. The Semantic Web. Scientific American, 2001, 284(5): 34-43
- [2] Ehrig M, Euzenat J. Relaxed Precision and Recall for Ontology Matching // Ashpole B, et al, eds. Proc. K-Cap Workshop on Integrating Ontology. New York: ACM, 2005; 25-32
- [3] 袁洋,李善平. 基于语义 Web 的本体映射方法综述. 计算机科

5 对象映射

类型匹配解决了接口层互操作的类型失配问题。在通信层互操作中我们采用一种对象映射机制,对象映射提供了对象互操作实施过程中的动态支持,实现对远程对象服务的访问。对象映射分静态和动态两部分:对象映射的静态部分依据类型匹配说明建立代理对象和代理类,代理对象是代理类的一个实例,代理对象负责客户对象和服务对象之间连接的建立、请求和结果传送等;对象映射的动态部分负责互操作过程中所用对象的建立(实例化)和管理。对象映射的实现模型如图 6 所示。

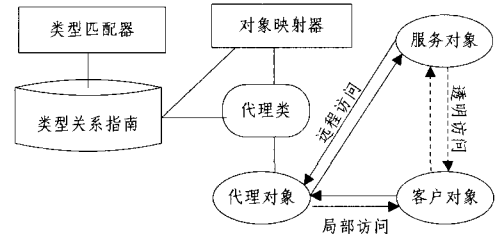


图 6 对象映射的实现模型

结束语 对象互操作是实现开放分布处理的一种有效途径。本文主要针对对象互操作的分层模型进行了研究,提出了三层结构模型,并对每层互操作提出了具体的解决方法。针对该模型的实现机制与方法方面,本文课题组基于原型系统正做进一步的研究,并在其他文献里做进一步介绍。

参 考 文 献

- [1] Vallecillo A, Hernández J, Troya J M, et al. Object Interoperability // Proceedings of the ECOOP'99 Workshop on Object Interoperability (WOI'99). Universidad de Málaga, 1999
- [2] Hernández J, Vallecillo A, Troya J M, et al. New Issues in Object Interoperability // Proceedings of the ECOOP'00 Workshop on Object Interoperability (WOI'00). Universidad de Extremadura, 2000
- [3] Vallecillo A, Vasconcelos V, Ravara A. Typing the behavior of objects and components using session types // Proc. of FOCLASA 2002. ENTCS, 2002, 68 (3)
- [4] Durán F, Vallecillo A. Formalizing ODP Enterprise Specifications in Maude. Computer Standard & Interfaces, 2003, 25(2): 83-102
- [5] Durán F, Roldán M, Vallecillo A. Using Maude to Write and Execute ODP Information Viewpoint Specifications. Computer Standard & Interfaces, 2005, 27(6): 597-620

学, 2004, 31(5): 5-8

- [4] van Rijsbergen C. Information Retrieval. London: Butterworths, 1979
- [5] Euzenat J. Semantic Precision and Recall for Ontology Alignment Evaluation // Proc. IJCAI'07. Berlin: Springer, 2007: 348-353
- [6] 石莲,孙吉贵. 描述逻辑综述. 计算机科学, 2006, 33(1): 194-197
- [7] Borgida A, Serafini L. Distributed Description Logics: Assimilating Information from Peer Sources // LNCS 2800. Journal of Data Semantics, Berlin: Springer, 2003: 153-184
- [8] Haase P, et al. A Framework for Handling Inconsistency in Changing Ontologies // Gil Y, et al, eds. Proceedings of the Fourth International Semantic Web Conference (ISWC 2005). LNCS 3729. Berlin: Springer, 2005: 353-367