扩展结构包含推理算法的本体匹配

廖建新[1,2]，刘秀磊[1,2]，朱晓民[1,2]，孙海峰[1,2]，王敬宇[1,2]
(1.北京邮电大学 网络与交换技术国家重点实验室，北京 100876； 2. 东信北邮信息技术有限公司，北京 100191)

摘 要：为了匹配本体时直接分析构造器和公理中所蕴含的语义信息，提出一种扩展结构包含推理算法的本体匹配方法以解决该问题。首先将本体中实体重定向为范式，使得被蕴含的语义信息明显地表示，然后比较范式之间的句法结构以推理来自不同本体的实体间的匹配。针对一组工业本体的测试结果表明该方法具有较好的性能。

关键词：语义分析；本体匹配；结构包含推理算法
中图分类号：TP182 **文献标识码**：A **文章编号**：1000-436X(2012)08-0190-10

Extending structural subsumption reasoning algorithms for ontology matching

LIAO Jian-xin[1, 2], LIU Xiu-lei[1, 2], ZHU Xiao-min[1, 2], SUN Hai-feng[1, 2], WANG Jing-yu[1, 2]
(1. State Key Lab of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China;
2. EBUPT Information Technology Ltd., Beijing 100191, China)

Abstract: For exploiting semantics implied in constructors and axioms in an ontology, a structural subsumption reasoning algorithm was extended to match ontologies. The implied semantics was first enabled to be read-off easily by rephrasing entities in an ontology into normal forms; then the syntactical structure of the normal forms were compared to infer correspondences between entities, one from each ontology. The prototype was evaluated over four in use ontologies and compared against twelve methods. The experiments show that its performance is higher than other solutions on average.

Key words: semantic analysis; ontology matching; structural subsumption reasoning algorithms

1 引言

随着本体的广泛应用，表示相似领域共享概念模型的大部分本体往往是由不同背景知识的工程师使用各种术语构造和维护。这些表示相似领域的不同本体之间的异构性阻碍了应用系统对知识的共享、重用和互操作。本体匹配则是解决本体异构问题的方法之一。本体匹配在本体工程、生物医学、P2P 信息共享、Web 服务组合以及语义物联网等领域有着广泛的应用。

OWL（ontology Web language）是目前较为流行的本体描述语言。使用 OWL 表示的本体由各种构造器和公理构成。文献[1~3]中一些方法通过使用词法（lexical）分析和语义（semantic）分析的方法匹配本体。在语义分析阶段，它们通常将词法分析阶段的结论输入推理机（reasoners）直接产生实体

收稿日期：2011-12-05；**修回日期**：2012-04-06
基金项目：国家自然科学基金资助项目（61121001，61072057，60902051，61101119）；长江学者和创新团队发展计划基金资助项目（IRT1049）；国家科技重大专项基金资助项目——移动互联网总体架构研究（2011ZX03002-001-01）
Foundation Items: The National Natural Science Foundation of China (61121001, 61072057, 60902051, 61101119); PCSIRT (IRT1049); The National Key Science & Technology Specific Project of China——Research About Architecture of Mobile Interne (2011ZX03002-001-01)

间的匹配[4,5]，或基于推理机的推论计算实体间的相似性[6~8]。因此，目前基于语义分析的本体匹配方法多是通过推理机利用本体中的语义信息。然而，它们并没有充分探索本体中的构造器和公理（比如⊇、∩、∪、∀等）所蕴含的语义信息。例如有如下 2 组公理：

1) $Book \subseteq Reference$, $Reference \subseteq Publication$;

2) $Book \subseteq Publication$, $Publication \subseteq Reference$。

如果直接应用推理机，这 2 组公理可得到相同的推论，即 $Book \subseteq Publication$，然而这些公理并不表示相同的语义信息。因此，基于推理机的语义分析方法并不能够完全反映本体中的语义，它仅反映了这些语义的推论。

基于以上考虑，本文扩展了结构包含推理算法以分析本体的语义信息。该方法首先剖析组成本体的各种构造器和公理（比如⊇、∩、∪、∀等），基于描述逻辑和代数集合中的定理构建实体的范式（normal forms），这使得本体里蕴含的语义信息和词法信息能够容易读出；然后通过调节实体间被允许的差异程度比较 2 个实体范式之间的句法结构以产生实体间匹配。

2　相关工作

随着语义 Web 和大量基于本体应用的发展，本体匹配已经成为目前的研究热点之一。文献[1~3]调查了各种本体匹配方法，从不同方面对它们进行归类，并提出在本体匹配中可利用的信息。它们包括：词法信息、结构信息、语义信息、外部数据信息和个体信息。通常不同的本体匹配方法通过各种信息技术使用上述中的一种或多种信息进行匹配。

这些信息技术包括系数的计算（coefficient computation）[9]、机器学习（machine learning）[10,11]、合成理论（hybrid methods）[12]、图匹配（graph matching）[13]、马尔科夫网（Markov network）[14]、向量空间模型（vector space models）[15]、优化技术（optimization techniques）[16,17]、贝叶斯决策理论（Bayesian decision theory）[18]以及各种推理机制（reasoning mechanisms）[6,7,9,19]等。

本体中公理和构造器不仅含有本体的结构（structural）信息，也蕴含了本体的语义信息。本文仅关注于本体匹配系统对语义信息的使用方法。根据归约（deductions）规则，本体匹配中的语义分析方法主要分为 2 类：可满足性问题（propositional satisfiability problem）和描述逻辑推理（description logics reasoning）。

可满足性问题的决策器（deciders）通常输入交范式（conjunctive normal forms），然后判定输入范式之间的语义关系，但交范式并不能很好地处理一些 OWL 本体中的构造器和公理，比如并（disjunction）、完全否（full negation）和完全存在限制（full existential restriction）等，这导致了基于交范式的本体匹配方法（比如 S-Match[4]）可被用在简单的本体语言表示的本体匹配中，但并不被用于 OWL 表示的本体匹配中。

通常，描述逻辑推理方法采用能够解释并、完全否和完全存在限制等的表演算法（tableau algorithms），所以使用推理机推论的方法（比如 ILIADS[6]（integrated learning in alignment of data and schema）和 ALOWS[7]）能处理在 OWL 本体中的各种句法元素。

ILIADS 通过启发式算法抽取了有限的本体公理，并基于这些公理使用推理机进行推理，因此它并没有使用来自本体的所有公理进行推理，这有可能导致某些语义的丢失，甚至导致不正确的推论。ALOWS 通过在所有的公理上直接使用推理机来解决该问题。CtxMatch[19]将表达式提交到基于表演算法的推理机，并直接将推理机的推论作为实体间的关系。S-Match 不仅使用推理机制，而且提供了 2 种不同的概念表示：标签概念和实体概念。标签概念仅关于实体标签里上下文无关的单词词义，它简单地表示了标签的词法信息。实体概念与上下文相关，它表达了一定的逻辑关系。S-Match 通过使用从根实体到需要计算实体之间所有实体标签的交计算实体概念。

ASMOV[9]（automated semantic matching ontologies with verification）首先获得本体之间的匹配，然后检验这些匹配以确保它们不包含任何不一致的语义。ASMOV 中的语义匹配是基于结构信息的翻译，实际上它依然采用相似性方法计算来自不同本体的实体间的匹配，仅使用语义的方法检验获得的本体匹配，以提高系统的精度。

综上所述，目前大部分使用语义信息的本体匹配方法并没有分析公理和构造器中的语义，而仅是使用推理机的推论。本文提出的扩展结构包含推理算法的方法（DLOM）分析了公理和构造器中的各

种元素使得本体的语义显示地表现出来，比较了表现语义的实体范式以推理来自不同本体的实体间的匹配。

3 背景知识和系统架构

3.1 背景知识

随着语义 Web 的发展，OWL 成为目前较为流行的本体描述语言，依据表达能力，它被划分为 3 个表达层次：OWL-Lite、OWL-DL 和 OWL-Full。OWL-Lite 对应于描述逻辑$\Sigma HI\Phi(\Delta)$，具有概念分层和简单限制，虽然推理服务相对有效，但表达能力较弱。OWL-DL 对应于描述逻辑$\Sigma HOIN(\Delta)$，其表达能力强，推理服务相对有效。OWL-Full 的表达能力最强，但它的语义具有逻辑不可判定性。因此，相对于 OWL-Lite 和 OWL-Full，OWL-DL 应用更为广泛。本文只关注于以 OWL-DL 表示的本体之间的匹配。

OWL-DL 的语义可翻译（interpretation）成知识库（knowledge base），因此当匹配本体时为了使用蕴含的语义信息，仅需分析其对应的知识库。知识库通常包含 3 部分：TBox、PBox 和 ABox。其中 TBox 是概念公理的集合；PBox 是属性公理的集合；ABox 是个体公理的集合[20]。本文不关注个体信息，因此仅分析知识库中的 TBox 和 PBox，同时也假设在本体里没有对实体进行圈定义（cyclical definition），比如 $Human \equiv People \cap \forall hasParent.Human$。

在本体匹配过程中，有必要匹配来自不同本体的数据属性（datatype property）和对象属性（object property）。因此本文不区分本体里的数据属性和对象属性，统称它们为属性（或角色）。基于以上考虑，需将 OWL-DL 表示的本体中的数据属性转化为对象属性，转化过程如下。

1) 转换数据类型属性的范围（range），即数据类型（datatype），为概念（concepts）。

2) 转换数据类型的值（value of datatypes）为相应概念的个体（individuals）。

3) 转换数据类型属性（datatype properties）为对象类型（object properties）。

由于 OWL-DL 所对应的$\Sigma HOIN(\Delta)$使用数据值、数据类型和数据属性扩展$\Sigma HOIN$而得到。因此，转换后的本体句法与$\Sigma HOIN$句法相同。本文以下内容均指转换后的本体。

3.2 系统架构

本节简述了原型系统 DLOM（如图 1 所示）。DLOM 系统主要分为 2 个阶段：候选计算阶段（图 1 中竖虚线左侧所示）和匹配推理阶段（图 1 中竖虚线右侧所示）。它输入 2 个 OWL-DL 表示的本体，输出一组来自不同本体的实体间的匹配。本文使用 2 个样例本体解释各种示例，它们分别来自于 OAEI (ontology alignment evaluation initiative) 2009 标准测试集的 101 文件夹和 302 文件夹。为了表示样例本体中的实体，采用<101 (或 302)：实体标签>的方式。

候选计算阶段与文献[7]中的词法分析阶段相同。词法分析器的主要目的是分析本体中实体的标签和评论，自动地获得它们中每个单词在 WordNet 本体中的合适词义[21]，并扩展这些词义。实体标记表示器基于单词的词义及其扩展定义实体标记以表示本体中实体的词法信息，简记为 $C(E)$，其中 E 表示本体中的实体。匹配候选生成器基于实体标记生成一组匹配候选供匹配推理阶

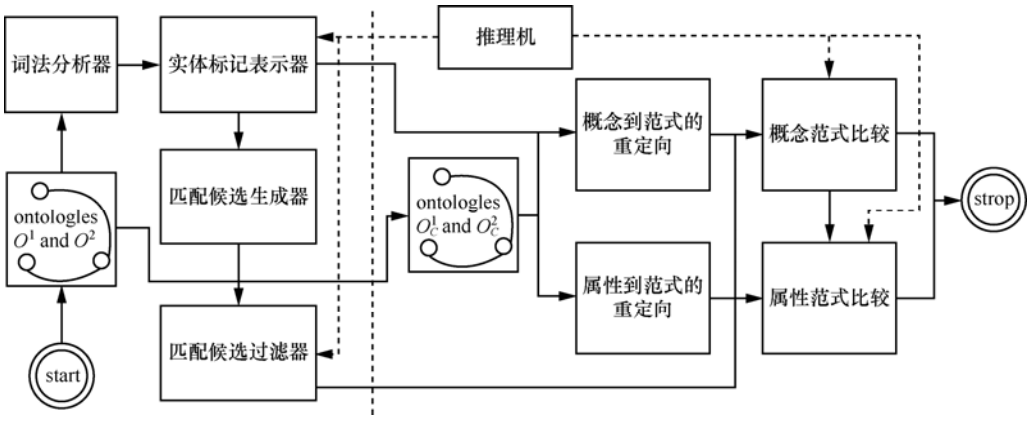


图 1 系统架构

段使用。匹配候选过滤器将删除冗余的候选，过滤后的匹配候选集合，简记为 MCF。基于过滤的匹配候选集合，匹配推理阶段扩展结构包含推理算法以推理来自不同本体的实体间的匹配（第 4 节描述了该阶段）。

4　扩展结构包含推理算法的本体匹配

匹配推理阶段扩展结构包含推理算法以推理出实体间的匹配。该阶段首先基于候选计算阶段定义的 $C(E)$通过重定向实体到范式的方式分析了本体中的构造器和公理，然后基于候选计算阶段的 MCF 通过比较来自不同本体的范式之间的句法结构产生匹配。

4.1　定义范式

将本体里实体（包括概念和属性）重定向为范式（包括概念范式和属性范式）的目的是使被蕴含的语义信息能够明显地表示出来。本节形式化地定义了范式。该定义不仅明显地表示了本体的语义信息，也表示了本体的词法信息。

概念范式的形式化定义如下。

定义 1　C 表示本体中的概念。称 C 是范式，当且仅当 $Normal_C = C_1 \cup \cdots \cup C_m$ 时

$$C_i := \prod_{A \in prim(C_i)} A \cap \prod_{R \in N_R} \{ \prod_{C' \in exist_R(C_i)} (\exists R.C') \cap$$

$$\cap \forall R.forall_R(C_i) \cap < min_R(C_i), max_R(C_i) > R\}$$

1) 集合 $prim(C_i)$表示在 C_i的顶层中所有的原子概念和它们的补概念（negated）以及相应实体标记的表示。

2) N_R 是在 C_i 顶层中可用角色（或属性）的集合。

3) $exist_R(C_i)$是一个概念描述集合。这个集合里的任何元素 C 在 C_i 顶层中存在 $\exists R.C$。

4) $forall_R(C_i)$是通过在 C_i 顶层中合并角色 R 的所有值限制（$C_1 \cap \cdots \cap C_n$）形成若干个概念描述的交（$\forall R.C_1 \cap \cdots \cap \forall R.C_n$）。

5）$min_R(C_i)$表示在 C_i 顶层中角色 R 的至少限制（at-least restrictions）的最大势（cardinality），$max_R(C_i)$ 表示该角色 R 的至大限制（at-most restrictions）的最小势（cardinality）。如果存在角色 R 的相等限制（at-equivalence restrictions），则 $min_R(C_i)$和 $max_R(C_i)$与相等限制中的势相同。如果仅 $min_R(C_i)$存在，$max_R(C_i)$是+∞。如果仅 $max_R(C_i)$存在，$min_R(C_i)$是 0。如果相应的 $\exists R$ 存在，$min_R(C_i)$大于 0。

6）对于任意的 i，$forall_R(C_i)$和 C'都在范式形式。如果 $C_i \equiv \perp$（通过推理机进行判断），则在公式中将它删除。

从范式的定义中可以看到，集合补（negation）总是直接出现在原子概念前面，$\exists R$ 在 C_i 顶层中出现多次。属性范式的形式化定义与上述概念范式的形式定义类似，此处不再赘述。

4.2　重定向实体到范式

在概念到范式的重定向组件中，主要有以下几个步骤。

首先，基于下面的规则将本体知识库 TBox 中的包含公理和不相交公理转换为定义公理，转换后的 TBox 记作 *TBox*。转换公理的规则和顺序如下：

1）如果 A 是命名的概念，$A \perp B \Rightarrow A \subseteq \neg B$；

2）$A \subseteq B$；$A \subseteq C$；$A \subseteq D \Rightarrow A \equiv B \cap C \cap D$；

3）如果 A 尚未定义，$A \subseteq B \Rightarrow A \equiv \tilde{A} \cap B$；

4）如果 A 已经被定义，$A \equiv C$；$A \equiv \tilde{A} \cap B \Rightarrow A \equiv \tilde{A} \cap B \cap C$。

其中，A 是命名概念，$\tilde{A}$ 是相应的实体标记，即 $C(A)$。

正如规则 4)所述，通过引入实体标记，将包含公理转换为定义公理。这里实体标记代表了概念 A 与它的父概念的不同，并没有改变原有包含公理的语义。这些规则保证了 *TBox* 是定义的（关于定义的 *TBox*，参考文献[22]）。规则 4)也保证了在 *TBox* 里概念仅被定义一次，也就是说，至多有一个左面是 A 的公理存在于 *TBox* 中。规则 3)和规则 4)引入的实体标记保证了实体间潜在的关系。

转换概念到范式组件的第 2 步利用一个迭代的过程将 *TBox* 中的定义公理扩展。它使用相应概念的定义替换每个定义公理中右面的概念。因为在输入本体里没有圈定义，所以该过程最后将终结。

转换概念到范式组件的第 3 步通过描述逻辑和代数集合的定律（比如结合律、吸收律、摩根律、分配律、同一律等[17,18]）将上一步得到的定义公理的扩展化简到 4.1 节所述的范式形式。属性到范式的重定向方法与概念到范式的重定向方法类似，此处不再赘述。

假设有简单本体 O，由如下公理构成：

①$Woman \equiv Person \cap Female$

②$Man \equiv Person \cap \neg Woman$

③$Mother \equiv Woman \cap \exists hasChild.Person$

④$Father \equiv Man \cap \exists hasChild.Person$

⑤$Parent \equiv Father \cup Mother$

⑥$Grandmother \subseteq Mother \cap \exists hasChild.Parent$

通过上述步骤以及 4.1 节范式的定义，可将本体 O 转化到范式形式，如下所示：

①$Woman \equiv C(Woman) \cap Person \cap Female$

②$Man \equiv C(Man) \cap Person \cap \neg Person \cap Female$

③ $Mother \equiv C(Mother) \cap Person \cap Female \cap \exists hasChild.Person$

④$Father \equiv C(Father) \cap (Person \cap \neg (Female \cap Person)) \cap \exists hasChild.Person$

⑤$Parent \equiv C(Parent) \cap ((Person \cap \neg (Person \cap Female)) \cap \exists hasChild.Person) \cup ((Person \cap Female) \cap \exists hasChild.Person)$

⑥$Grandmother \equiv C(Grandmother) \cap ((Person \cap Female) \cap \exists hasChild.Person) \cap \exists hasChild.(((Person \cap \exists \neg (Person \cap Female)) \cap \exists hasChild.\ Person) \cup ((Person \cap Female) \cap \exists hasChild.Person))$

从条款⑥可以看到概念“*Grandmother*”在本体 O 中的形式化语义由原子实体 *Person*，*Female*，*hasChild* 所构成；$C(A)$表示了概念 A 与其父概念的不同，也表示了在 WordNet 中的词义（即人们对概念 A 的词法信息在现实世界中的认识），这种方法没有改变原有公理的语义。重定向实体到范式方法将实体在本体中语义信息和词法信息都显示地表示出来，为推理匹配奠定了基础。

4.3 推理实体间匹配

当分别定义不同本体里相似的概念时，知识工程师们有时会忽略一些元素或者使用不同的范围限制它们。即便是工程师们为不同的本体构建 2 个相同的概念时，这些概念之间通常也存在着差异。因此本节引入阈值α、β和γ以便调节来自不同本体的实体间被允许的差异程度。

在概念范式比较组件里假如有来自不同本体的 2 个概念 C 和 D，它们不是⊥也不是 T。它们的概念范式如下（分别简记为 $Normal_C$ 和 $Normal_D$）。

$$Normal_C = C_1 \cup \cdots \cup C_m$$

$$C_i \coloneqq \prod_{A \in prim(C_i)} A \cap \prod_{R \in N_R} \{ \prod_{C' \in exist_R(C_i)} (\exists R.C') \cap \cap \forall R.forall_R(C_i) \cap < \min_R(C_i), \max_R(C_i) > R \}$$

$$Normal_D = D_1 \cup \cdots \cup D_m \text{ with}$$

$$D_j \coloneqq \prod_{B \in prim(D_j)} B \cap \prod_{P \in N_P} \{ \prod_{D' \in exist_R(D_j)} (\exists P.D') \cap \cap \forall P.forall_P(D_j) \cap < min_P(D_j), max_P(D_j) > P \}$$

如果下列条件成立则称之为 $C \subseteq D$：

对于所有的 i，$1 \leqslant i \leqslant n$，存在 j，$1 \leqslant j \leqslant m$ 使得 $C_i \subseteq D_j$。

如果下列条件成立（ $0 \leqslant \alpha \leqslant 1$， $0 \leqslant \beta \leqslant 1$，$\gamma \leqslant 1$），则称之为 $C_i \subseteq D_j$：

1) $\dfrac{Counter}{prim(D_j)size} \geqslant \alpha$，其中，对于在 $prim(D_j)$ 的任何元素 B，如果在 $prim(C_i)$中存在元素 A，使得 $A \subseteq B$，则 Counter++；

2) $\dfrac{Counter}{prim(N_P)size} \geqslant \beta$，其中，对于在 $prim(N_P)$ 中的任何元素 P，如果在 $prim(N_R)$中存在元素 R，使得下列条件成立，则 Counter++：

① $R \subseteq P$；

② $\gamma \leqslant min_R(C_i) - min_P(D_j)$，

$\gamma \leqslant max_P(D_j) - max_R(C_i)$ ；

③ $forall_P(D_j) \supseteq forall_R(C_i)$；

④ $\forall D' \in exsit_P(D_j) \rightarrow \exists C' \in exist_R(C_i)$ such that $C' \subseteq D'$ ；

在条件 1)中，当判定 $prim(C_i)$中的元素是否是 $prim(D_j)$中元素的子概念时（即 $A \subseteq B$），使用相应的实体标记替代这些原子概念，然后基于 MCF 推理它们之间的关系。例如，¬ $C(A)$和 $C(B)$之间的关系替代原子概念¬ A 和 B 之间的关系。条件①中属性关系的判定，由属性范式比较组件完成。

引入阈值α、β和γ是为了调节不同本体中实体间被允许的差异程度。阈值α在[0，1]区间变化时，它反映了在 $prim(C_i)$和 $prim(D_j)$中忽略元素的个数。α 越大，在 $prim(C_i)$和 $prim(D_j)$中忽略的元素越多。如果α=0，说明系统并不考虑来自 $prim(C_i)$和 $prim(D_j)$ 元素之间的任何关系。如果α=1，说明系统考虑来自 $prim(C_i)$和 $prim(D_j)$任何元素之间的关系。阈值β 与α类似，它反映的是属性之间的关系。阈值γ反映的是 at-least 限制和 at-most 限制中范围被允许的差异。它越小，范围之间被允许的差异越大。当阈值 γ=0，意味着 at-least 和 at-most 限制中不被允许任何差异。

根据上述方法，可以推理出实体间的包含关系，基于此，使用下列定律也可以推理实体间的相等（equivalence）和不相交（disjoint）关系[18]。

$C \equiv D \Leftrightarrow C \subseteq D$ 和 $C \supseteq D$

$C \perp D \Leftrightarrow C \subseteq \neg D$

在直觉上该方法能够保证推理的有力性（sound），但不能保证其完整性（complete），这与结构包含算法的特殊性有关。正如条件③和条件④所示，该方法包含一个递归的过程。属性范式比较方法与概念范式比较方法类似，此处不再赘述。通过比较范式的句法结构，将产生一组来自不同本体概念间的匹配，每个匹配包含一个源实体、一个目标实体和它们之间的逻辑关系。在匹配样例本体的过程中，此阶段产生的匹配情况如图 2 所示。图 2 中匹配的左侧方框和区配的右侧方框分别表示实体来自样例本体 *Ontology*101 和 *Ontology*302；黑线表示 DLOM 系统发现的且存在于标准答案中的匹配；黑粗线表示 DLOM 系统未发现的但存在于标准答案中的匹配；点线表示 DLOM 发现的但较少出现在其他解决方案中的匹配；虚线表示 DLOM 发现的但不存在于标准答案中的匹配。

描述逻辑领域的非标准推理技术结构包含推理算法的基本思想是比较范式间原子概念以得到概念间的关系。4.2 节将实体转换到范式显示地展现了实体的语义信息和词法信息，本节比较了实体范式间的句法结构，实际上是比较展现的语义信息和词法信息。明显地，DLOM 继承了结构包含推理算法的基本思想，为适应本体匹配它也进行了扩展。在比较范式句法结构时，它基于实体标记利用 WordNet[21]作为知识库推理出原子概念之间的关系，也引入了α、γ和β表示本体匹配时所容许的差异，这些是原有的结构包含推理算法所没有的。

5　算法复杂度分析

本文包括 2 个主要步骤：重定向实体到范式和推理实体间匹配，本节分别阐述它们的算法复杂性。

5.1　重定向实体到范式算法分析

由 4.2 节可看到重定向实体到范式是一个迭代过程，使用相应概念的定义替换定义公理右侧的概念直到公理右侧无概念可替换。

假设 *TBox* 中有 *n* 个公理且 *TBox* 中无圈定义，

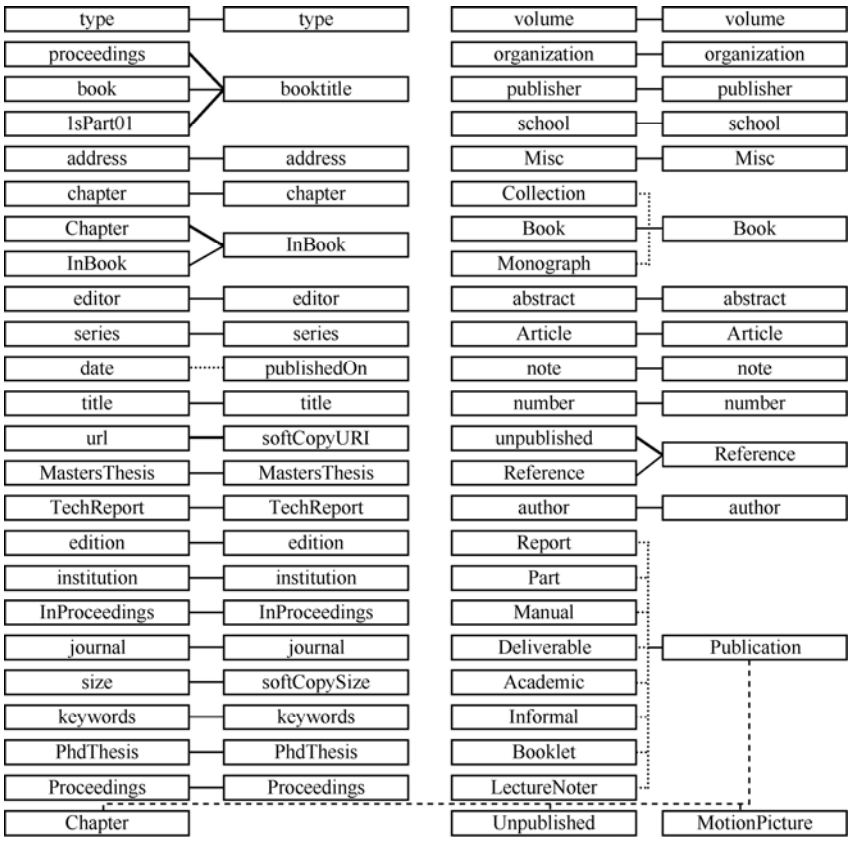


图 2　匹配样例本体时的匹配情况

A_n 表示最后的原子概念，下面的 n 个公理描述了最复杂的情况：

$A_1=A_2\cap A_3\cap\cdots\cap A_{n-1}\cap A_n$

$A_2=A_3\cap A_4\cap\cdots\cap A_{n-1}\cap A_n$

$A_3=A_4\cap A_5\cap\cdots\cap A_{n-1}\cap A_n$

$A_i=A_{i+1}\cap A_{i+2}\cap\cdots\cap A_{n-1}\cap A_n$

…

$A_{n-4}=A_{n-3}\cap A_{n-2}\cap A_{n-1}\cap A_n$

$A_{n-3}=A_{n-2}\cap A_{n-1}\cap A_n$

$A_{n-2}=A_{n-1}\cap A_n$

$A_{n-1}=A_n$

A_n

本文将采用由底向上的迭代过程，从时间来看，其基本操作是公理右侧概念的替代。先分析重写 A_{n-2} 到范式的情况，需将定义 A_{n-1} 公理的右侧代入定义 A_{n-2} 公理右侧的 A_{n-1}。当重写实体 A_i 到范式时，需要将其后面的已写成范式的 $n-i$ 个概念代入其右侧的相应概念。则在整个重写实体到范式的过程中，所需进行替代的次数为 $1+2+3\cdots+(n-2)=(n-1)\times(n-2)/2$。上述最简单的情况是每个定义公理右侧都是原子概念，所以替代次数为 0。若 *TBox* 中定义公理的复杂程度是随机的，即定义公理右侧出现各种情况的概率相同，则可取上述最大值和最小值的平均值，作为概念替代的次数，约为 $(n-1)\times(n-2)/4$，因此时间复杂度为 $O(n^2)$。

根据由底向上的迭代过程，从空间来看，其基本空间单位是原子概念，计算上述过程的空间复杂度，即是计算增加的使用原子概念的次数。现分析重写 A_{n-3} 到范式的情况，需分别将 A_{n-2} 和 A_{n-1} 右侧的原子概念代入到 A_{n-3} 右侧的相应概念，它增加了 1 次原子概念的使用。若重写 A_i 到范式，需要将其后面的已写成范式的 $n-i$ 个概念代入其右侧的相应位置，它增加了 $f(A_{i+1})+f(A_{i+2})+\cdots f(A_{n-1})+f(A_n)$ 次原子概念的使用，其中，$f(A_j)$ 表示概念 A_j 中增加的使用原子概念的次数，其中，A_{n-3} 为 1。假设 $n=8$，则有：

$f(A_1)=f(A_2)+f(A_3)+\cdots f(A_7)+f(A_8)=2^4$

$f(A_2)=f(A_3)+f(A_4)+\cdots f(A_7)+f(A_8)=2^3$

$f(A_3)=f(A_4)+f(A_5)+\cdots f(A_7)+f(A_8)=2^2$

$f(A_4)=f(A_5)+f(A_6)+\cdots f(A_7)+f(A_8)=2^1$

$f(A_5)=f(A_6)+f(A_7)+f(A_8)=2^0$

$f(A_7)=f(A_8)=0$

所以，在整个重写实体到范式的过程中，增加地使用原子概念的次数为 $2^0+2^1+\cdots+2^{n-2}=2^{n-1}-1$。上述最简单的情况是每个定义公理右侧都是原子概念，所以增加地使用原子概念的次数为 0。若 *TBox* 中定义公理的复杂程度是随机的，则可取上述最大值和最小值的平均值，作为增加地使用原子概念的次数，约为 $(2^{n-1}-1)/2$，因此时间复杂度为 $O(2^n)$。所以 DLOM 更加适合于小本体的匹配。大规模本体的匹配将是下一步重要的研究工作。

5.2 推理实体间匹配算法分析

由 4.3 节条件③和条件④可看出，DLOM 推理实体间匹配时包含一个递归的过程。假设有分别来自 $TBox^1$ 的概念 A_1 的范式和 $TBox^2$ 的概念 B_1 的范式（为计算方便省去 $<min, max>R$ 和 $\forall R.C$），如下所示：

$A_1=C_1\cap C_2\cap\cdots\cap C_x\cap\exists\forall R.A_2\cap\cdots\cap\exists\forall R.A_n$

$B_1=D_1\cap D_2\cap\cdots\cap D_y\cap\exists\forall R.B_2\cap\cdots\cap\exists\forall R.B_m$

其中，n 表示 $TBox^1$ 中命名概念的个数，m 表示 $TBox^2$ 中命名概念的个数，C_i 和 D_j 分别表示原子概念。概念 A_i 和 B_j 分别表示 $TBox^1$ 和 $TBox^2$ 中概念最为复杂的情况（见 5.1 节）。由 5.1 节可以看到，A_i 中原子概念的个数约为 2^i，B_j 中原子概念个数约为 2^j。

根据 4.3 节算法的过程来看，不需要任何辅助存储空间，因此无需计算该算法的空间复杂度。从时间来看，基本操作是来自不同本体的原子概念间关系的比较。如果要判断 $A_1\subseteq B_1$，首先需要将 C_1 到 C_x 中任何元素与 $D_1\cdots D_y$ 依次比较，计算次数为 $x\times y$；然后需要依次比较 A_2 与 $B_1\cdots B_m$ 中的元素，共比较次数为 $2^0\times((2^{m-1}-1)/2)$；最后将 A_2 到 A_n 中任何元素与 $B_1\cdots B_m$ 依次比较，计算总次数为 $x\times y+(2^{n-1}-1)\times(2^{m-1}-1)/4$。上述最简单的情况是 $A_1=C_1\cap C_2\cap\cdots\cap C_x$ 和 $B_1=D_1\cap D_2\cap\cdots\cap D_y$，则比较次数为 $x\times y$。若 *TBox* 中定义公理的复杂程度是随机的，则可取上述最大值和最小值的平均值，作为原子概念间关系计算的次数，约为 $x\times y+(2^{n-1}-1)\times(2^{m-1}-1)/4$，因此时间复杂度为 $O(2^{n+m})$。所以，再一次说明 DLOM 更加适合于小本体的匹配。

6 实验评估

本节讨论了第 2 节中原型系统（DLOM）的评估。DLOM 使用多种开源分组来操作本体、执行推

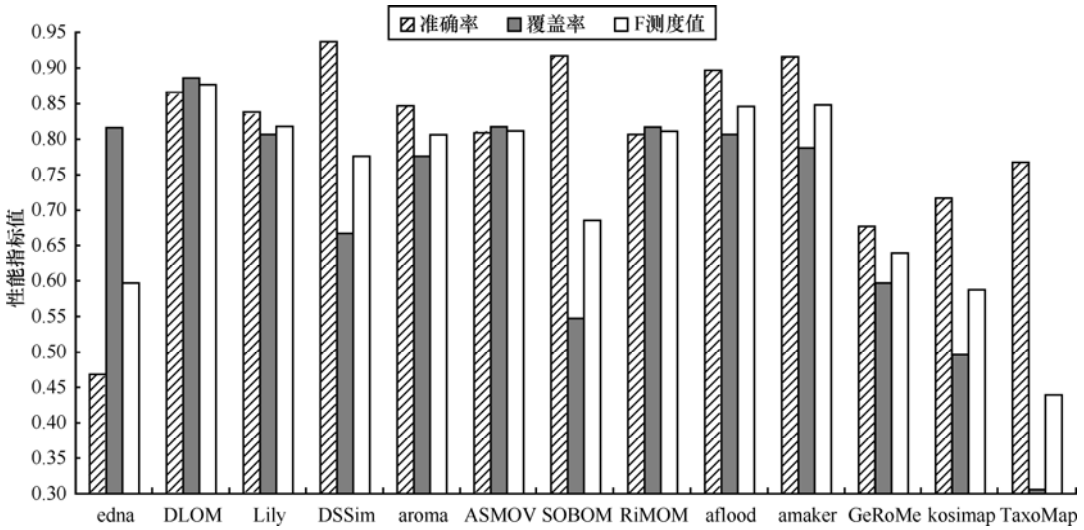


图 3　系统评估结果：准确率，覆盖率和 F 测度值

理、比较匹配和与 WordNet 交互，包括 JENA、Java WordNet Library API、Pellet API、OWL2.0 API、Protégé API、SKOS API 和 JOWL 等。

本节主要使用 3 种性能指标（准确率（precision）、覆盖率（recall）和 F 测度值（F-measure））评估 DLOM 系统的性能。其中 F 测度值反映了准确率和覆盖率的综合值（关于它们的计算，请参阅文献[9,18]）。

6.1　实验数据

本节选择了本体匹配评估倡议组织（OAEI , ontology alignment evaluation initiative）标准中 4 个工业本体作为测试集（可通过文献[23]下载）。它们都是书目领域的本体，分别来自 BibTeX、UMBC、Karlsruhe、INRIA，依次标记它们为 *Ontology*301、*Ontology*302、*Ontology*303、*Ontology* 304。同时，OAEI 也提供了这些本体匹配任务的标准匹配，以便计算方案的性能。通常，这些本体包括大约 37 个概念、72 个属性和 108 个公理。

6.2　参数优化

在 DLOM 系统中，有 3 个参数表示匹配过程中被允许的实体间的差异程度：α、β和γ。本节通过匹配书目领域 4 个本体的实验设置它们。其中α和β在[0，1]之间变化，变化间隔为 0.01；由于在实验中$\gamma \leqslant -10$ 没有实际意义，所以实验中γ在[−10，0]之间变化，变化间隔为 0.01。实验时，固定 2 个参数不变，仅变动一个参数，并标识当时的 F 测度值。表 1 显示了实验的部分结果，当α=0.5、β=0.44 和γ=−2 时 F 测度值最大。

表 1　　DLOM 参数优化实验结果

序号	F 测度值	α	β	γ
1	0.88	0.12	0.51	0.50
2	0.57	0.16	0.57	0.74
3	0.83	0.23	0.77	0.65
4	0.57	0.73	0.87	0.25
5	0.79	0.80	0.30	0.90
6	0.55	0.50	0.91	0.23

6.3　与其他匹配方法的比较

本节通过测试书目领域的本体比较 DLOM 和其他 12 个系统，包括 SOBOM、DSSim、amaker、aroma、Lily、ASMOV、RiMOM、GeRoMe、aflood、kosimap、TaxoMap 以及一种基本的匹配系统（即 edna）。如图 3 所示，DLOM 显示出较好的性能。

图 3 显示了 13 个系统的准确率、覆盖率和 F 测度值。从图 3 可以看到，DLOM 在覆盖率方面有较好的质量，最高提高达到 58%（相对于 TaxoMap），最低也有 7%（相对于 RiMOM）。DLOM 扩展了结构包含推理算法使得被蕴含的语义能够明显地表示出来。因此 DLOM 能够深入地探索本体中的语义信息，并通过比较范式句法结构的方式对比相似实体间的语义信息。该特点使得 DLOM 发现了较多匹配，同时也产生了一些并不经常出现在其他方法中的匹配（但出现在标准匹配中）。因此，相对于其他方法，DLOM 显示了较高的覆盖率。下面列出了匹配样例本体时，并不经常出现在其他方法中但出现在标准答案中的匹配（如图 2 所示）。

①<101:*Manual*, 302:*Publication*, ⊆>;
② <101:*Academic*, 302:*Publication*, ⊆>;
③<101:*Deliverable*, 302:*Publication*, ⊆>;
④ <101:*Informal*, 302:*Publication*, ⊆>;
⑤ <101:*Report*, 302:*Publication*, ⊆>;
⑥<101:*LectureNotes*, 302:*Publication*, ⊆>;
⑦ <101:*Collection*, 302:*Book*, ⊆>;
⑧<101:*Monograph*, 302:*Book*, ⊆>

图 3 也显示了 edna 有较高的覆盖率，但准确率很低。因为 edna 发现了太多的匹配，虽然覆盖了大部分标准匹配，但也产生了许多不精确的匹配。

从图 3 可以看到 DLOM 的准确率较低（相对于 DSSim 和 SOBOM 等）。SOBOM 和 DSSim 等方法的基本思想是希望得到的每个匹配都是正确的，如果产生不确定的匹配，则舍去。因此在测试时它们仅产生约 50 个匹配，而其他方法通常产生约 70 个匹配。这保证了它们较高的准确率，然而这也导致了它们较低的覆盖率。同时，DLOM 使用 MCF 作为推理实体间匹配时的知识库（如 4.3 节所示）。因此 MCF 中的异常匹配候选会导致最后错误匹配的产生。例如在匹配样例本体时，MCF 中的 C(101:*notes*)$\supseteq C$ (302:*school*)导致了匹配<101: *notes*, 302: *school*, ⊇>的产生。因此继续提高匹配候选过滤器的性能是下一步的重要工作。文献[24]已指出 OAEI 建议的标准匹配并不包含所有合理的匹配；标准匹配中的匹配也不都合乎逻辑。在 DLOM 中产了这样符合本体语义的匹配，然而它们并没有出现在标准匹配中。例如当匹配 *Ontology*101 和 *Ontology*303 时，DLOM 产生了<101:*Deliverable*, 303:*Report*, ⊆>和<101:*Report*, 303:*ProjectReport*, ⊇>等，这些匹配是合理的且与本体的语义一致，但并没有出现在标准匹配中。这也是 DLOM 方法准确率较低的原因之一。

尽管 DLOM 的准确率不是最高的，但从图 3 可以看到，相比其他 12 个方法，DLOM 在 F 测度值方面（反映了准确率和覆盖率的综合性能）有较好的质量，最高提高达到 44%（相对于 TaxoMap），最低也有 3%（相对于 aflood）。

7 结束语

扩展结构包含推理算法的方法首先将本体中实体重定向为范式，使得被蕴含的语义信息形式化显示出来，然后比较范式之间的句法结构，推理出不同实体间的匹配。实验表明，该方法具有较好的性能。

参考文献：

[1] SHVAIKO P, EUZENAT J. Ten challenges for ontology matching[A]. Proceedings of the OTM 2008 Confederated International Conferences[C]. 2008. 300-313.

[2] ZHAO H. Semantic matching across heterogeneous data sources[J]. Communication ACM, 2007, 50(1): 45-50.

[3] KALFOGLOU Y, SCHORLEMMER M. Ontology mapping: the state of the art[J]. Knowl Eng Rev, 2003, 18(1): 1-31.

[4] GIUNCHIGLIA F, SHVAIKO P. Semantic matching[J]. Knowl Eng Rev, 2003, 18: 265-280.

[5] REUL Q, PAN J Z. Kosimap: use of description logic reasoning to align heterogeneous ontologies[A]. Proceedings of the 23rd International Workshop on Description Logics, Ser CEUR Workshop Proceedings[C]. Waterloo, Canada, 2010.

[6] UDREA O, GETOOR L, MILLER R J. Leveraging data and structure in ontology integration[A]. SIGMOD 07: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data[C]. New York, NY, USA, 2007. 449-460.

[7] LIU X, BARNAGHI P, MOESSNER K. Lexical and Semantic Analysis for Ontology Matching[R]. CCSR, Surrey University, Guildford, Surrey, Tech Rep CCSR-TR-101211, 2011.

[8] KOLLI M, BOUFAIDA Z. A description logics formalization for the ontology matching[J]. Procedia Computer Science, 2011, 3(1): 29-35.

[9] JEAN Y R, SHIRONOSHITA E P, KABUKA M R. Ontology matching with semantic verification[J]. Web Semantics: Science, Services and Agents on the World Wide Web, 2009, 7(3): 235-251.

[10] SPILIOPOULOS V, VOUROS G A, KARKALETSIS V. On the discovery of subsumption relations for the alignment of ontologies[J]. Web Semantics:Science, Services and Agents on the World Wide Web, 2010, 8(1): 69-88.

[11] ALGERGAWY A, MASSMANN S, RAHM E. A clustering-based approach for large-scale ontology matching[A]. Databases and Information Systems, Ser Lecture Notes in Computer Science[C]. 2011. 415-428.

[12] BUCCELLA A, CECHICH A, GENDARMI D, *et al.* Building a global normalized ontology for integrating geographic data sources[J]. Computers & Geosciences, 2011, 37(7): 893-916.

[13] JAMES N, TODOROV K, HUDELOT C. Combining visual and textual modalities for multimedia ontology matching[A]. Semantic Multimedia, Ser Lecture Notes in Computer Science[C]. Springer, 2011. 95-110.

[14] ALBAGLI S, BEN-ELIYAHU-ZOHARY R, SHIMONY S E. Markov

network based ontology matching[A]. Proceedings IJCAI'09 Proceedings of the 21st International Jont Conference on Artifical Intelligence[C]. San Francisco, CA, USA, 2009.

[15] MOUSELLY-SERGIEH H, UNLAND R. Irom: information retrieval-based ontology matching[A]. Semantic Multimedia, Ser Lecture Notes in Computer Science[C]. 2011.127-142.

[16] BOCK J, HETTENHAUSEN J. Discrete particle swarm optimization for ontology alignment[J]. Information Sciences, 2010, 192: 152-173.

[17] WANG X, XU Q. An improved ant colony optimization for ontology matching[A]. Computer Research and Development (ICCRD), 3rd International Conference on[C]. 2011. 234-238.

[18] TANG J, LI J, LIANG B, HUANG X, *et al*. Using Bayesian decision for ontology mapping[J]. Web Semantics: Science, Services and Agents on the World Wide Web, 2006, 4(4): 243-262.

[19] BONIFACIO M, DONA A, MOLANI A, *et al*. Context matching for electronic marketplaces - a case study[A]. Proceedings of the Workshop on Ontologies and Distributed Systems, 18th Int, Joint Conf on Artificial Intelligence[C]. 2003.

[20] GUARINO N, GIARETTA P. Ontologies and knowledge bases: towards a terminological clarification[A]. Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing[C]. 1995.

[21] MILLER G A, BECKWITH R, FELLBAUM C, *et al*. Introduction to wordnet-an online lexical database[J]. International Journal of Lexicography, 1990, 4(3): 235-244.

[22] BAADER F, CALVANESE D, MCGUINESS D, *et al*.. The Description Logic Handbook: Theory, Implementation and Applications[M]. Cambridge University Press, 2003.

[23] OAEI[EB/OL].http://oaei.ontologymatching.org/2009/benchmarks/, 2009.

[24] MEILICKE C. The relevance of reasoning and alignment incoherence in ontology matching[A]. ESWC, Ser Lecture Notes in Computer Science[C]. Springer, 2009. 934-938.

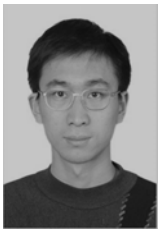
作者简介：

廖建新（1965-），男，四川宜宾人，博士后，北京邮电大学教授，北京邮电大学网络与交换技术国家重点实验室网络智能研究中心主任，主要研究方向为业务网络智能化。

刘秀磊（1981-），男，河南濮阳人，北京邮电大学博士生，主要研究方向为语义 Web、本体匹配等。

朱晓民（1974-），男，浙江义乌人，博士，北京邮电大学副教授、硕士生导师，主要研究方向为智能网、下一代业务网络、3G 核心网、协议工程等。

孙海峰（1989-），男，天津人，北京邮电大学博士生，主要研究方向为语义 Web、语义标注等。



王敬宇（1976-），男，吉林省吉林市人，博士，北京邮电大学讲师，主要研究方向为 P2P 网络、网络虚拟化等。