

一种支持自定义模糊数据类型表示的模糊 OWL 扩展

赵法信¹ 马宗民² 王海龙²

(湛江师范学院信息科学与技术学院 湛江 524048)¹ (东北大学信息科学与工程学院 沈阳 110004)²

摘要 语义网需要能够以一种智能的方式来表示和处理模糊知识与数据信息,但最近的研究证明,OWL在数据类型表示方面存在严重的局限性:首先,它不能处理在语义网和本体中广泛存在的不精确以及不确定信息;其次,它不支持自定义数据类型及自定义数据类型谓词的表示。为了解决上述局限性,提出了一种新的模糊描述逻辑 F-SHOIN(G),给出了 F-SHOIN(G)的语法、语义。然后,以 F-SHOIN(G)为逻辑基础,并结合 RDF/XML 的语法规则,重新编码 OWL 中的描述算子,用来表示模糊本体知识,形成 f-OWL。f-OWL 不仅能够表示本体中的模糊知识,而且能够表示本体中含有自定义模糊数据类型及自定义模糊数据类型谓词的模糊数据信息,为模糊本体的研究提供了新的表示方法。最后,还给出了由 OWL 文件到 f-OWL 文件的转换规则,从而实现了 f-OWL 对 OWL 的兼容。

关键词 语义网,模糊描述逻辑, F-SHOIN(G),模糊 OWL,自定义数据类型

中图分类号 TP301 文献标识码 A

Fuzzy Extension of OWL Supporting Representation of Customized Fuzzy Data Type

ZHAO Fa-xin¹ MA Zong-min² WANG Hai-long²

(School of Information Science and Technology, Zhanjiang Normal University, Zhanjiang 524048, China)¹

(School of Information Science and Engineering, Northeastern University, Shenyang 110004, China)²

Abstract The Semantic Web is expected to process fuzzy concept knowledge and data type information in an intelligent and automatic way. Recent research has shown that OWL has some serious limitations; i. e., it can't process imprecise and uncertain information which widely exists in the Semantic Web and ontology; furthermore, it does not support customized data types and customized data type predicates. In this paper, we made the following two contributions to solve the above limitations; (i) presented a new kind of fuzzy description logic F-SHOIN(G), and gave its syntax and semantics; (ii) extended OWL to f-OWL by recoding the description norms of OWL, based on F-SHOIN(G) and the syntax specification of RDF/XML. f-OWL can not only support the representation of fuzzy concept knowledge, but also fuzzy data information with customized fuzzy data types and customized fuzzy data type predicates. It provides a new representation method for the investigation of ontology.

Keywords Semantic Web, Fuzzy description logic, F-SHOIN(G), Fuzzy OWL, Customized data type

1 引言

目前,面向语义 Web^[1]的本体语言主要有 OIL, DAML+OIL 和 OWL 等。其中,OWL 是 W3C 关于本体描述语言的最新推荐标准,它按照表达能力由强到弱有 3 种子语言: OWL Full, OWL DL 和 OWL Lite。上述的本体语言只能处理精确和完备的知识及其推理任务,不能对现实世界和自然语言中大量存在的模糊和不精确知识进行表示和推理。为了使语义 Web 具备处理模糊信息的能力,文献[2]在模糊描述逻辑的基础上扩展本体语言 OWL 以处理模糊知识;在模糊描述逻辑的基础上构建了 RDF/XML 式的模糊算子和公理,用来表达本体中的模糊概念及声明,提出了 OWL 到模糊算子的转换规则,统一了模糊本体中确定知识的表示方法。文献[3]对本体语言 OWL DL 进行了基于模糊逻辑的扩展,给

出了形式化的语法和语义,并通过实例说明了其扩展在表达能力上的灵活性。Stoilos 提出了模糊描述逻辑 f-SHOIN,然后基于 f-SHOIN 给出了模糊 OWL 的抽象语法和语义,并给出了从模糊 OWL 本体到 f-SHOIN 知识库的转换方法^[4]。但是,上述 3 种模糊 OWL 只能处理抽象模糊知识,并不能处理现实应用中大量存在的具体的模糊数据信息。

在实际应用中,一个本体不仅包含许多重要的知识,而且包括大量的数据信息,因此,语义 Web 需要能够以一种智能的方式来表示和处理这些信息。实际上,对数据类型表示的支持作为 OWL 自身设计的一个重要特征,已经引起了 SWB-PD (W3C Semantic Web Best Practices and Deployment Working Group) 的广泛讨论与关注。众所周知,XML 的扩展数据类型机制^[5]为其增加了丰富的表达能力,它允许用户在基本数据类型基础上扩展自己的数据类型,但 OWL DL 语

到稿日期:2010-07-28 返修日期:2010-12-13 本文受国家自然科学基金项目(60873010),湛江师范学院专项科研项目(ZL0902)资助。

赵法信(1974-),男,博士,高级工程师,主要研究方向为智能数据处理、数据挖掘,E-mail: zfx0405@163.com; 马宗民(1965-),男,博士,教授,博士生导师,主要研究方向为智能数据、知识工程; 王海龙(1983-),男,博士,主要研究方向为描述逻辑、语义 Web。

法规并不支持相应的机制,即 OWL DL 不支持自定义数据类型和自定义数据类型谓词,使得 OWL DL 难以满足用户需求。为此, Pan 引入了数据类型域 (data type group, 缩写为 G), 提出了描述逻辑 SHOQ(G) 和 SHIQ(G) 来支持精确情形下自定义数据类型和自定义数据类型谓词的表示, 并在此基础上对 OWL 进行了扩展, 提出了 OWL-Eu^[6] 以及 OWL-E^[7]。但 OWL-Eu 以及 OWL-E 只是针对精确的情形, 并不能对模糊和不精确知识与数据信息进行表示和推理。作为处理模糊 OWL 数据信息的一个尝试, Calegari 在模糊描述逻辑 f-SHOIN(D) 的基础上对 OWL 进行了模糊扩展^[8], 但扩展后的模糊 OWL 只能表示简单模糊数据类型信息, 并不能表示复杂的自定义模糊数据类型及自定义模糊数据类型谓词, 而后者在实际应用中有时非常重要。实际上, 很多潜在的用户指出, 如果不克服 OWL 的上述局限性, 他们将不会在应用中采用此语言。SWBPD 已经制定了一项课题, 致力于解决 OWL 表达能力中存在的这些问题^[9]。

为此, 本文基于模糊数据类型域的概念和模糊数据类型表示的一般形式^[10], 将模糊数据类型域 G 引入到模糊描述逻辑 F-SHOIN^[11,12] 中, 提出了一种新的模糊描述逻辑 F-SHOIN(G), 并定义了 F-SHOIN(G) 的语法、语义及相应的知识库。然后, 基于描述逻辑 F-SHOIN(G), 并结合 RDF/XML 的语法规则, 对文献[2-4]中的模糊 OWL 进行模糊数据类型支持的模糊扩展, 从而提出 f-OWL。f-OWL 不仅支持抽象模糊知识的表示, 而且支持含有自定义模糊数据类型及自定义模糊数据类型谓词的模糊数据信息的表示。最后, 还给出了由 OWL 文件到 f-OWL 文件的转换规则, 从而实现了 f-OWL 对 OWL 的兼容。

2 模糊描述逻辑 F-SHOIN(G)

将一致性模糊数据类型域 G 引入到模糊描述逻辑 F-SHOIN^[11,12] 中, 可得到一种新的模糊描述逻辑 F-SHOIN(G)。F-SHOIN(G) 不仅包含原有描述逻辑 F-SHOIN 中的概念构子, 而且还增加了与模糊数据类型表达式相关的概念构子, 具备了更丰富的表达能力, 从而能够作为扩展模糊 OWL DL 语言的逻辑基础。

定义 1 假设 C, R_A, R_D, I 分别是 F-SHOIN(G) 的不相交的概念名、抽象角色名、数据类型角色名以及个体名的集合。 $R = R_A \cup R_D; R \in R_A$ 叫做抽象角色, 其中 $R_A := R_A \cup \{R^- \mid R \in R_A\}$, 即 R_A 由其本身或其逆角色组成; $T \in R_D$ 叫做数据类型角色。假设 A 是 C 中的原子概念, $R \in R_A, T_1, \dots, T_n \in R_D, o \in I, G$ 是一致性模糊数据类型域, $E \in \text{Dexp}(G)$ 。F-SHOIN(G) 概念遵循以下抽象语法:

$$C := T \mid \perp \mid A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R. C \mid \forall R. C \mid \{o_1, \dots, o_n\} \mid \geq nR \mid \leq nR \mid \exists T_1, \dots, T_n. E \mid \forall T_1, \dots, T_n. E \mid \geq mT_1, \dots, T_n. E \mid \leq mT_1, \dots, T_n. E$$

式中, $\exists T_1, \dots, T_n. E, \forall T_1, \dots, T_n. E, \geq mT_1, \dots, T_n. E, \leq mT_1, \dots, T_n. E$, 称之为模糊数据类型概念, 其余的称之为模糊抽象概念。

描述逻辑 F-SHOIN(G) 的语义由解释 $I = (\Delta^I, \cdot^I)$ 给出。I 包括非空域 Δ^I 和解释函数 \cdot^I , 并且 Δ^I 与 Δ_D 互不相交。 \cdot^I 将每个抽象个体 $a \in I$ 映射为一个元素 $a^I \in \Delta^I$, 将任

意概念 $A \in C$ 映射为一个函数 $A^I: \Delta^I \rightarrow [0, 1]$, 将抽象角色 $R \in R_A$ 映射为一个函数 $R^I: \Delta^I \times \Delta^I \rightarrow [0, 1]$, 将数据类型角色 $T \in R_D$ 映射为函数 $T^I: \Delta^I \times \Delta_D \rightarrow [0, 1]$ 。上述模糊解释可以被扩展到解释任意的 F-SHOIN(G) 概念和角色。在形式上, 对模糊抽象概念的语义解释类似于文献[11], 对模糊数据类型概念的语义解释类似于文献[10]。

定义 2 一个 F-SHOIN(G) 知识库 \mathcal{K} 由模糊 $TBox \mathcal{T}$, 模糊 $RBox \mathcal{R}$ 和模糊 $ABox \mathcal{A}$ 组成, 即 $\mathcal{K} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$, 其中:

(1) 模糊 $TBox$ 是有限个形如 $C \sqsubseteq D$ 的术语包含公理的集合, 其中 C 和 D 是 F-SHOIN(G) 概念。形如 $C \equiv D$ 的模糊概念公理可以拆分为 $C \sqsubseteq D$ 且 $D \sqsubseteq C$ 。与 F-SHOIN 知识库的模糊 $TBox$ 不同的是, 这里的概念 C 和 D 可以是模糊数据类型概念 $\exists T_1, \dots, T_n. E, \forall T_1, \dots, T_n. E, \geq mT_1, \dots, T_n. E, \leq mT_1, \dots, T_n. E$ 等。

(2) 在形式上, 模糊 $RBox$ 的定义与 F-SHOIN^[11] 对模糊 $RBox$ 的定义相同。

(3) 模糊 $ABox$ 是有限个形如 $\langle \alpha, \triangleright \langle k \rangle \rangle$ 的模糊断言的集合, 其中, $k \in [0, 1]$, 并且模糊断言 α 或者形如 $x: C$, 或者形如 $\langle (x, y): R \rangle$, 或者形如 $\langle (x, v): T \rangle$ ($x, y \in \Delta^I, v$ 是数据类型变量)。

一个 F-SHOIN(G) 概念 C 是可满足的当且仅当存在一个解释 I 对某个 $x \in \Delta^I$, 满足 $C^I(x) = k, k \in (0, 1]$ 。此时, 解释 I 叫做概念 C 的一个模型。一个解释 I 满足模糊 $TBox \mathcal{T}$ 中的一条公理 $C \sqsubseteq D$, 当且仅当 $\forall x \in \Delta^I, C^I(x) \leq D^I(x)$ 。I 是 $TBox \mathcal{T}$ 的一个模型当且仅当 I 满足 \mathcal{T} 中的所有公理。一个解释 I 满足 $\langle x: C, \triangleright \langle k \rangle \rangle$ 当且仅当 $C^I(x) \triangleright \langle k \rangle$, 满足 $\langle (x, y): R, \triangleright \langle k \rangle \rangle$ 当且仅当 $R^I(x, y) \triangleright \langle k \rangle$, 满足 $\langle (x, v): T, \triangleright \langle k \rangle \rangle$ 当且仅当 $T^I(x, v) \triangleright \langle k \rangle$ 。I 叫做 $ABox \mathcal{A}$ 的一个模型当且仅当 I 满足 \mathcal{A} 中的所有断言。一个知识库 \mathcal{K} 是可满足的当且仅当存在一个解释 I 同时满足 $TBox \mathcal{T}, RBox \mathcal{R}$ 和 $ABox \mathcal{A}$ 。

3 模糊 OWL

基于 F-SHOIN(G), 通过向 OWL 事实 (OWL facts) 中引入隶属度, 并参考 OWL 的 RDF/XML 格式的语法形式, 重新编码 OWL 中的描述算子, 对其进行模糊扩展, 从而能够表示模糊本体知识与数据, 并将模糊扩展后的 OWL 称为 f-OWL。f-OWL 作为 OWL DL 的必要而且有效的模糊扩展, 应该满足以下要求:

(1) f-OWL 要克服现有 OWL 在模糊数据类型表示方面存在的不足, 提供对逆谓词以及自定义模糊数据类型和自定义模糊数据类型谓词的支持。

因为模糊数据类型域与 OWL 现有的数据类型表示机制兼容, 同时具备表示自定义模糊数据类型和自定义模糊数据类型谓词的能力, 所以, f-OWL 的扩展应该基于模糊数据类型域。

(2) 扩展后的 f-OWL 的推理问题应该是可判定的, 即 f-OWL 的推理问题可以转化为相应的模糊描述逻辑的判定问题, 而后者本身是可判定的。

(3) f-OWL 应该与现有的 OWL 语法规则兼容, 即现有的用 OWL 语法规则描述的本体文档, 可以通过相应的转换规则转换为 f-OWL 语法描述的文档。f-OWL 对于 OWL 的

扩展,应该是最小、必要而且有效的。

根据模糊数据类型域 G 的一致性的定义^[10]可知,模糊描述逻辑 F-SHOIN(G)的推理问题是可判定的,则以 F-SHOIN(G)作为逻辑基础的 f-OWL 的推理问题也是可判定的,从而满足上述要求(2)。下面根据扩展 f-OWL 的要求(1)、(3),基于模糊描述逻辑 F-SHOIN(G)对 OWL 进行扩展。

从模糊集合的角度看,普通概念是模糊概念的特例。所以 OWL 中的各种类描述可以很自然地推广到模糊概念情况。假定 fowl(fuzzy OWL)是模糊知识的命名空间,则各种类描述(交,并,补等)和术语公理(概念特化,概念定义等)采用与 OWL 相似的表示方式,只是加上 fowl 的命名空间。表 1 中给出了 f-OWL 类描述的语法形式及对应的描述逻辑 F-SHOIN(G)的语法、语义形式。其中, $C_1 \dots C_n$ 是模糊概念, R 是模糊抽象关系,此外,与 rdf 名称空间相关的表示依然遵循 RDF 规范的要求。

表 1 f-OWL 类描述语法及相应的 F-SHOIN(G)语法、语义

f-OWL 语法	F-SHOIN(G)语法	F-SHOIN(G)语义
Class	$A(A \text{ is a URIref of a class})$	$A^I: \Delta^I \rightarrow [0,1]$
Thing	T	$T^I(x) = 1$
Nothing	\perp	$\perp^I(x) = 0$
intersectionOf($C_1 \dots C_n$)	$C_1 \sqcap \dots \sqcap C_n$	$C_1^I(x) \wedge \dots \wedge C_n^I(x)$
unionOf($C_1 \dots C_n$)	$C_1 \sqcup \dots \sqcup C_n$	$C_1^I(x) \vee \dots \vee C_n^I(x)$
complementOf(C)	$\neg C$	$1 - C^I(x)$
oneOf($o_1 \dots o_n$)	$\{o_1, \dots, o_n\}$	$\bigvee_{i=1}^n (o_i^I = x)$
restriction(R someValuesFrom(C))	$\exists R.C$	$\sup_{y \in \Delta^I} (R^I(x, y) \wedge C^I(y))$
restriction(R allValuesFrom(C))	$\forall R.C$	$\inf_{y \in \Delta^I} (R^I(x, y) \rightarrow C^I(y))$
restriction(R value(o))	$\exists R.o$	$R^I(x, o^I)$
restriction(R minCardinality(n))	$\geq nR$	$\sup_{y_1, \dots, y_n \in \Delta^I} (\bigwedge_{i=1}^n (R^I(x, y_i) \wedge C^I(y_i)))$
restriction(R maxCardinality(n))	$\leq nR$	$1 - (\geq (n+1) R.C)^I(x)$

根据模糊集合理论,对 OWL 事实的模糊语义解释可以进一步扩展到解释 OWL 公理^[13],包括 OWL 类公理、OWL 属性公理以及 OWL 个体公理,如表 2 所列。需要说明的是,如果没有显式地注明某一条个体公理的隶属度值,那么说明这条个体公理的隶属度值为 1。从语义解释的角度来看,表 2 中第 1 列的 OWL 公理是可满足的,当且仅当表 2 中对应的第 3 列中的式子是可满足的。一个模糊本体 O 由若干模糊公理组成,一个模糊解释 I 叫做 O 的一个模型,当且仅当 I 满足 O 中的所有公理。

根据对 f-OWL 的扩展要求(1),要使 f-OWL 支持逆谓词以及自定义模糊数据类型及自定义模糊数据类型谓词,那么 f-OWL 要包含一致性模糊数据类型域,而且首先要支持模糊数据类型表达式的表示。表 3 中给出了表示模糊数据类型表达式的 f-OWL 语法及相应的模糊描述逻辑 F-SHOIN(G)语法。其中, p 是一个模糊数据类型谓词的统一资源定位符,“ s_i ”^^“ d_i ”是有型文字, v_i 是一元模糊数据类型谓词或其逆谓词。进而, f-OWL 应该基于定义 1 提供一些新的关于模糊数据类型概念表示的类描述,如表 4 所列。这些类描述与表 1 中的支持抽象模糊概念表示的类描述共同组成 f-OWL 语言的类描述。在表 4 中, T 是模糊数据类型属性, R 是模糊抽象属性, E 是模糊数据类型表达式。

表 2 f-OWL 公理语法及相应的 F-SHOIN(G)语法、语义

f-OWL 语法	F-SHOIN(G)语法	F-SHOIN(G)语义
(Class A partial $C_1 \dots C_n$)	$A \sqsubseteq C_1 \sqcap \dots \sqcap C_n$	$A^I(x) \leq C_1^I(x) \wedge \dots \wedge C_n^I(x)$
(Class A complete $C_1 \dots C_n$)	$A = C_1 \sqcap \dots \sqcap C_n$	$A^I(x) = C_1^I(x) \wedge \dots \wedge C_n^I(x)$
(EnumeratedClass A $o_1 \dots o_n$)	$A = \{o_1, \dots, o_n\}$	$A^I(x) = 1$ 如果 $x \in \{o_1^I \dots o_n^I\}$, 否则 $A^I(x) = 0$
(SubClassOf $C_1 C_2$)	$C_1 \sqsubseteq C_2$	$C_1^I(x) \leq C_2^I(x)$
(EquivalentClasses $C_1 \dots C_n$)	$C_1 = \dots = C_n$	$C_1^I(x) = \dots = C_n^I(x)$
(DisjointClasses $C_1 \dots C_n$)	$C_1 \sqcup C_j = \perp, 1 \leq i < j \leq n$	$C_i^I(x) \wedge C_j^I(x) = 0, 1 \leq i < j \leq n$
DatatypeProperty(T super(T_1) super(T_n))	$T \sqsubseteq T_i$	$T^I(x, v) \leq T_i^I(x, v)$
domain(C_1) \dots domain(C_k)	$\forall \geq 1 T \sqsubseteq C_i$	$T^I(x, v) \leq C_i^I(x)$
range(E_1) \dots range(E_h)	$T \sqsubseteq \forall T. E_i$	$T^I(x, v) \leq E_i^I(v)$ $\forall x \in \Delta^I \inf_{v_1, v_2 \in \Delta_D^I}$
[Functional]	$T \sqsubseteq \leq 1 T$	$T^I(x, v_1) \vee T^I(x, v_2) \geq 1$
ObjectProperty(R super(R_1) super(R_n))	$R \sqsubseteq R_i$	$R^I(x, y) \leq R_i^I(x, y)$
domain(C_1) \dots domain(C_k)	$\geq 1 R \sqsubseteq C_i$	$R^I(x, y) \leq C_i^I(x)$
range(C_1) \dots range(C_h)	$T \sqsubseteq \forall T. C_i$	$R^I(x, y) \leq C_i^I(y)$
[inverseOf(R_0)]	$R = R_0^-$	$R^I(x, y) = (R_0^-)^I(x, y)$
[Symmetric]	$R = R^-$	$R^I(x, y) = (R^-)^I(x, y)$
[Functional]	$T \sqsubseteq \leq 1 R$	$\forall x \in \Delta^I \inf_{y_1, y_2 \in \Delta^I} R^I(x, y_1) \vee R^I(x, y_2) \geq 1$
[InverseFunctional]	$T \sqsubseteq \leq 1 R^-$	$\forall x \in \Delta^I \inf_{y_1, y_2 \in \Delta^I} R^{-I}(x, y_1) \vee R^{-I}(x, y_2) \geq 1$
[Transitive]	Trans(R)	$\sup_{y \in \Delta^I} R^I(x, y) \wedge R^I(y, z) \leq R^I(x, z)$
SubPropretyOf ($T_1 T_2$)	$T_1 \sqsubseteq T_2$	$T_1^I(x, v) \leq T_2^I(x, v)$
EquivalentPropertes ($T_1 \dots T_n$)	$T_1 = \dots = T_n$	$T_1^I(x, v) = \dots = T_n^I(x, v)$
SubPropretyOf R_1, R_2	$R_1 \sqsubseteq R_2$	$R_1^I(x, y) \leq R_2^I(x, y)$
EquivalentPropertes ($R_1 \dots R_n$)	$R_1 = \dots = R_n$	$R_1^I(x, y) = \dots = R_n^I(x, y)$
Individual(o type(C_1) [\bowtie degree(m_1)]) \dots	$o_i: C_i \bowtie m_i$	$C_i^I(o^I) \bowtie m_i, m_i \in [0, 1], 1 \leq i \leq n$
value(R_1, o_1) [\bowtie degree(k_1)] \dots	$(o, o_i): R_i \bowtie k_i$	$R_i^I(o^I, o_i^I) \bowtie k_i, k_i \in [0, 1], 1 \leq i \leq n$
value(T_1, v_1) [\bowtie degree(l_1)] \dots	$(o, v_i): T_i \bowtie l_i$	$T_i^I(o^I, v_i^I) \bowtie l_i, l_i \in [0, 1], 1 \leq i \leq n$
SameIndividual($o_1 \dots o_n$)	$o_1 = \dots = o_n$	$o_1^I = \dots = o_n^I$
DifferentIndividuals($o_1 \dots o_n$)	$o_i \neq o_j, 1 \leq i < j \leq n$	$o_i^I \neq o_j^I, 1 \leq i < j \leq n$

表 3 f-OWL 的模糊数据类型表达式表示语法及相应的 F-SHOIN(G)语法、语义

f-OWL 语法	F-SHOIN(G)语法	F-SHOIN(G)语义
rdfs:Literal	T_D	$T_D^D = 1$
DatatypeBottom	\perp_D	$\perp_D^D = 0$
DatatypePredicate	$p(v_1, \dots, v_n)$	$p^D(v_1, \dots, v_n)$
not(p)	$\bar{p}(v_1, \dots, v_n)$	$1 - p^D(v_1, \dots, v_n)$
domain(u_1, \dots, u_s)	$([u_1, \dots, u_s]) (v)$	$((u_1)^D(v), \dots, (u_s)^D(v))$
and (E_1, \dots, E_s)	$(E_1 \wedge \dots \wedge E_s)$	$(E_1)^D(v_1, \dots, v_n) \wedge \dots \wedge (E_s)^D(v_1, \dots, v_n)$
or (E_1, \dots, E_s)	$(E_1 \vee \dots \vee E_s)$	$(E_1)^D(v_1, \dots, v_n) \vee \dots \vee (E_s)^D(v_1, \dots, v_n)$
datatypeoneOf(l_1, \dots, l_s)	$\{l_1, \dots, l_s\}$	$\bigvee_{i=1}^s (l_i^D = v)$

表4 f-OWL的模糊数据类型描述语法及相应的F-SHOIN(G)语法、语义

f-OWL 语法	F-SHOIN(G)语法	F-SHOIN(G)语义
restriction({T} someTuplesSatisfy(E))	$\exists T_1, \dots, T_n. E$	$\sup_{v_1, \dots, v_n \in \Delta_D} ((\bigwedge_{i=1}^n T_i^1(x, v_i)) \wedge EP(v_1, \dots, v_n))$
restriction({T} allTuplesSatisfy(E))	$\forall T_1, \dots, T_n. E$	$\inf_{v_1, \dots, v_n \in \Delta_D} ((\bigwedge_{i=1}^n T_i^1(x, v_i)) \rightarrow EP(v_1, \dots, v_n))$
restriction({T} minCardinality(m) someTuplesSatisfy(E))	$\geq m T_1, \dots, T_n. E$	$\sup_{v_{11}, \dots, v_{1n}, \dots, v_{m1}, \dots, v_{mn} \in \Delta_D} \bigwedge_{j=1}^m ((\bigwedge_{i=1}^n T_{ij}^1(x, v_{ij})) \wedge EP(v_{i1}, \dots, v_{in}))$
restriction({T} maxCardinality(m) someTuplesSatisfy(E))	$\leq m T_1, \dots, T_n. E$	$1 - (\geq (m+1) T_1, \dots, T_n. E)^1(x)$

模糊约束^[2]是模糊知识库中另一种重要的知识表示形式,OWL语言中没有对应成分,需要重新定义。从本质上说,模糊约束可以看成是域中的个体属于相应模糊概念的范围,所以可以用该模糊概念的隶属函数范围来表示。为了表示模糊本体中的单一个体和复数个体,分别定义了对应的词汇“fowl:individual”和“fowl:individuals”,它们有属性词“fowl:name”,其值用来限制个体名,如果表示复数个体,该属性词出现的次数和复数个数相同。需要说明的是,模糊约束涉及的对象有常量也有变量,如果是常量则“fowl:name”的值用引号引起来;如果是变量则不引。另外定义词汇“fowl:membershipOf”,表示对象属于某一概念(原子概念或类描述)的程度,与该词相匹配的4个范围词分别是“fowl:moreorEquivalent”,“fowl:lessorEquivalent”,“fowl:moreThan”,“fowl:lessThan”,依次表示 $\geq, <=, >, <$ 这样的约束限制。如果对象所属的是原子概念,则该原子概念可作为属性词“rdf:resource”的值出现,否则表示概念的类描述要作为内嵌元素。具体使用如下所示。

例1 如果要表示个体a属于类A的隶属度大于等于n,即 $A(a) \geq n$,可用基于RDF/XML格式的f-OWL语法表示为:

```
<fowl:individual fowl:name="a">
  <fowl:membershipOf rdf:resource="#A"/>
  <fowl:moreorEquivalent fowl:value=n/>
</fowl:individual>
```

例2 如果要表示个体a,b属于抽象属性R的隶属度大于等于n,即 $R(a,b) \geq n$,可用基于RDF/XML格式的f-OWL语法表示为:

```
<fowl:individuals fowl:name="a" fowl:name="b">
  <fowl:membershipOf rdf:resource="#R"/>
  <fowl:moreOrEquivalent fowl:value=n/>
</fowl:individuals>
```

扩展的f-OWL语言在自定义模糊数据类型及自定义模糊数据类型谓词表示方面具备比OWL更强的表达能力。下面通过举例说明。

例3 假如要在某地区做年龄小于40岁的年轻人的消费水平的社会调查。显然,一个人的消费水平与他的收入以及支出有关。假设现在需要表示如下的查询需求:收入比支出多且月收入高于500美元的年龄小于40岁的年轻人。因为在现实情况下,“年轻”是一个模糊概念;人的收入及支出也不是固定不变的值,因此“收入比支出多”就内含了模糊和不确定的因素。这两种情形都超出了经典OWL的表达能力,

而且在上述的查询需求中存在自定义模糊数据类型谓词“多且高于”及“小于40且年轻”,这也超出了文献[2-4]中的扩展模糊OWL的表达能力。但是,上述查询需求可以用f-OWL表示和解决。

首先给出上述查询需求的模糊描述逻辑形式的描述,即:
 $person \sqcap \exists age. (young \wedge <_{40}) \sqcap \exists \langle income, expense \rangle.$

$$\langle > \wedge >_{income > \$500} \rangle \quad (1)$$

式中,“age”是简单的模糊数据类型属性,“young”是一元模糊数据类型谓词,“ $<_{40}$ ”是一元谓词,也是一元模糊数据类型谓词的特例,要表示语义“小于40并且年轻”,就必须用到模糊数据类型表达式中的合取表达式。“ $>$ ”本身是具有精确意义的二元谓词,但因为“收入”以及“支出”本身的不确定性,使得“ $>$ ”具备了模糊和不确定的含义;“ $>_{income > \$500}$ ”是一个特殊的二元谓词,它只对模糊数据类型属性“收入”进行限制,而对“支出”不做限制,要表示语义“多并且高于500美元”,也必须利用模糊数据类型表达式中的合取表达式。

根据表1—表4中给出的f-OWL语法规则,可对式(1)所示的查询要求如下描述:

```
<fowl:intersectionOf>
  <fowl:Class ref:ID="person"/>
  <fowl:restriction>
    <fowl:DatatypeProperty rdf:ID="age">
      <fowl:someTuplesSatisfy>
        <fowl:and rdf:parseType="collection">
          <fowl:DatatypePredicate rdf:ID="young"/>
          <fowl:DatatypePredicate rdf:ID="<40"/>
        </fowl:and>
      </fowl:someTuplesSatisfy>
    </fowl:restriction>
  <fowl:restriction>
    <fowl:DatatypeProperty rdf:ID="income">
      <fowl:DatatypeProperty rdf:ID="expense">
        <fowl:someTuplesSatisfy>
          <fowl:and rdf:parseType="collection">
            <fowl:DatatypePredicate rdf:ID=">">
              <fowl:DatatypePredicate rdf:ID="
                <income> $500"/>
          </fowl:and>
        </fowl:someTuplesSatisfy>
      </fowl:restriction>
    </fowl:intersectionOf>
```

4 从OWL转换到f-OWL

模糊集合理论认为,模糊集合中的每个元素都具有隶属度,而经典集合中的元素隶属度仅有两种情况,“0”或者“1”。经典集合是模糊集合的一种特殊情况,因此,可以将经典的OWL文档转换成f-OWL文档,实现两者表示上的统一。基于文献[2]中的转换规则,对其进行扩展研究,使其同样适用于含有自定义数据类型及谓词的OWL-E^[7]文档的转换,进而推广至一般OWL文档到f-OWL文档的转换。详细转换规则描述如下:

(1) 每个简单类、复杂类(交(Intersection)、并(Union)、补(Complement)、等价(Equivalence))和匿名类(用“Restriction”定义)映射成对应的模糊简单类、模糊复杂类和模糊匿

名类;

(2) 类的层次结构 (“subClassOf”) 直接映射成对应的模糊包含;

(3) 每个类的实例映射成模糊类声明的形式, 其中的隶属度值 ≥ 1 ;

(4) 每个抽象属性或数据类型属性, 映射成对应的模糊抽象属性或模糊数据类型属性, 每个属性的实例映射成模糊属性声明, 其中的隶属度值 ≥ 1 ;

(5) OWL 中的简单数据类型谓词约束 (包括基本数据类型和枚举数据类型谓词), 以及 OWL-E 中的自定义数据类型谓词约束, 映射为表 3 中相应的模糊数据类型表达式约束, 并且对相应模糊数据类型属性进行约束的隶属度值 ≥ 1 。

下面举例说明具有代表性的转换规则(1)、(3)的使用, 其它转换规则的使用方式类似。

例 4 根据规则(1), 简单类、复杂类及匿名类转换时只需将对应的名称空间改变。OWL 源文档片段如下:

```
<owl:Class rdf:ID= "RedCar">
  <owl:intersectionOf rdf:parseType= "Collection">
    <owl:Class rdf:about= "# Car"/>
    <owl:Restriction>
      <owl:onProperty rdf:resource= "# hasColor"/>
      <owl:hasValue rdf:resource= "# Red"/>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

转换后的等价的 f-OWL 文档片段如下:

```
<fowl:Class rdf:ID= "RedCar" >
  <fowl:intersectionOf rdf:parseType= "Collection">
    <fowl:Class rdf:about= "# Car"/>
    <fowl:Restriction>
      <fowl:onProperty rdf:resource= "# hasColor"/>
      <fowl:hasValue rdf:resource= "# Red"/>
    </fowl:Restriction>
  </fowl:intersectionOf>
</fowl:Class>
```

在上述转换过程中, 只是将命名空间改变, 但已经使其具备了模糊的语义解释, 因为具备 fowl 命名空间的 f-OWL 文档将分别采用表 1—表 4 中的模糊语义解释, 而不是文献 [14] 中的经典语义解释。

例 5 根据规则(3), 对如下 OWL 源文档片段进行转换:

```
<owl:Thing rdf:about= "# Xiaoming">
  <rdf:type rdf:resource= "# Student"/>
</owl:Thing>
```

转换后的等价的 f-OWL 文档片段如下:

```
<fowl:individual fowl:name= "# Xiaoming">
  <fowl:membershipOf rdf:resource= "# Student">
  <fowl:moreOrEquivalent fowl:value= 1/>
</fowl:individual>
```

总之, 通过上述 5 条转换规则, 可以将经典的 OWL 文档转换成 f-OWL 文档, 进而实现表示方式的统一。所以, 这里扩展的 f-OWL 语言具备 OWL DL 语言的表达能力, 能够兼容 OWL DL 语言, 从而扩展的 f-OWL 符合扩展要求(3), 是有效的扩展。

结束语 为了克服 OWL 在模糊信息表示和自定义模糊数据类型及谓词表示两个方面的局限性, 以模糊描述逻辑 F-SHOIN(G) 为逻辑基础, 借鉴 OWL 的 RDF/XML 语法, 对表示类描述、公理和事实的 OWL DL 描述算子进行了重新编码, 用来表示模糊本体中的知识与数据, 形成 f-OWL。f-OWL 不仅能够表示本体中的模糊知识, 而且能够表示本体中含有自定义模糊数据类型及自定义模糊数据类型谓词的模糊数据信息。此外, 通过文中列出的转换规则, 可以将 OWL 文档转换为 f-OWL 文档, 所以 f-OWL 能够兼容 OWL, 是有效的扩展。

在以后的工作中, 将重点研究模糊描述逻辑 f-SHOIN(G) 的推理算法等, 以便提供对 f-OWL 的推理支持; 此外, 还将开发健壮的基于 f-OWL 的推理机, 以促进 f-OWL 的实际应用。

参 考 文 献

- [1] Berners-Lee T, Hendler J, Lassila O. The semantic Web [J]. Scientific American, 2001, 284(5): 34-43
- [2] Gao Mingxia, Liu Chunnian. Extending OWL by Fuzzy Description Logic [C]//Proc of the 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05). Washington: IEEE Computer Society, 2005: 562-567
- [3] 赵德新, 冯志勇. 基于本体语言 OWL 的模糊扩展[J]. 计算机科学, 2008, 35(8): 170-171, 175
- [4] Stoilos G, Stamou G, Pan J, et al. Reasoning with very expressive fuzzy description logics [J]. Journal of Artificial Intelligence Research, 2007, 30(8): 273-320
- [5] Bray T, Paoli J, Sperberg-McQueen C M, et al. Extensible Markup Language (XML) 1. 0 (Fifth Edition) [S/OL]. <http://www.w3.org/TR/REC-xml/>, 2008
- [6] Pan J Z, Horrocks I. OWL-Eu: Adding customized datatypes into OWL [J]. Journal of Web Semantics, 2006, 4(1): 29-39
- [7] Pan J Z. A flexible ontology reasoning architecture for the Semantic Web [J]. IEEE Transaction on Knowledge and Data Engineering, 2007, 19(2): 246-260
- [8] Calegari S, Ciucci D. Fuzzy ontology, fuzzy description logics and fuzzy OWL [C]//Proc of the 7th international workshop on Fuzzy Logic and Applications: Applications of Fuzzy Sets Theory. Berlin: Springer, 2007: 118-126
- [9] Mailing List, starts from 2004 [G/OL]. <http://lists.w3.org/archives/public/public-swbp-wg/>, 2004
- [10] 王海龙, 马宗民, 严丽, 等. 支持模糊数据类型表示的模糊描述逻辑 F-SHOIQ(G)[J]. 计算机学报, 2009, 32(8): 1511-1524
- [11] Stoilos G, Stamou G, Tzouvaras V, et al. Fuzzy OWL: Uncertainty and the Semantic Web [A]//Proceedings of the 2005 International Workshop on OWL: Experience and Directions [C]. Maryland, USA, 2005: 80-89
- [12] 蒋运承, 史忠植, 汤庸, 等. 面向语义 Web 语义表示的模糊描述逻辑[J]. 软件学报, 2007, 18(6): 1257-1269
- [13] Carroll J J, Roo J D. OWL Web Ontology Language Test Cases [S/OL]. W3C Recommendation. <http://www.w3.org/TR/owl-test/>, 2004
- [14] Deborah L, Harmelen F. OWL Web Ontology Language Overview [S/OL]. <http://www.w3.org/TR/owl-features/>, 2004