

面向 Deep Web 不确定知识表示的动态模糊描述逻辑

方 巍^{1,2} 崔志明²

(南京信息工程大学计算机与软件学院 南京 210044)¹

(江苏省现代企业信息化应用支撑软件工程技术研发中心 苏州 215104)²

摘 要 Deep Web 信息量大,主题专一,信息质量好。然而 Deep Web 信息存在着不确定问题,因此难以对其进行知识表示和推理。基于动态模糊逻辑理论,提出了一种新的描述逻辑,即动态模糊描述逻辑(DFDLs)。给出了 DFDLs 形式化定义以及 DFDLs 的语法和语义,设计了 DFDLs 的 tableau 的推理算法和策略。采用动态模糊描述逻辑对面向 Deep Web 的不确定知识进行表示并实现合理的推理和利用,能更好地表达 Deep Web 的动态和模糊信息。

关键词 Deep Web, 不确定知识, 动态模糊描述逻辑, tableau 推理

中图法分类号 TP182 **文献标识码** A

Dynamic Fuzzy Description Logic for Deep Web Uncertain Knowledge Representation

FANG Wei^{1,2} CUI Zhi-ming²

(School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China)¹

(Jiangsu Province Support Software Engineering R&D Center for Information Technology Application in Enterprise, Suzhou 215104, China)²

Abstract A large number of information of Deep Web is valued and topic-oriented, it is difficult for us to use knowledge representation formalisms to encode and reason within the uncertain knowledge in Deep Web. The paper presented a method to represent uncertain knowledge in Deep Web using dynamic fuzzy description logic(DFDLs). The syntax and semantics of DFDLs were given. Then, we provided a tableau-based decision algorithm for this logic. The DFDLs provides more reasonable logic foundation for the Deep Web uncertain knowledge, and it can present more fuzzy and dynamic information for the Deep Web.

Keywords Deep Web, Uncertain knowledge, DFDLs, Tableau inference

1 引言

目前主流搜索引擎还只能搜索 Internet 表面可索引的信息,但在 Internet 深处还隐含着大量的通过主流搜索引擎无法涉及的大量信息,这些信息被称之为 Deep Web(又称为 Invisible Web 或 Hidden Web)^[1]。根据 Bright Planet 研究表明^[2],Deep Web 信息量非常庞大,是可索引 Web 信息的 500 倍,并且这些 Deep Web 内容的 95% 都是可以通过 Internet 无需付费注册就可以公开访问的。Deep Web 的信息一般存储在服务端 Web 数据库中,与静态页面相比通常信息量更大,主题更专一,信息质量和结构更好^[3,4]。面对这些 Deep Web 海量信息,人们不再满足于基本的信息检索,提出了从海量数据中获取深层信息为决策服务的 Deep Web 搜索和处理问题。这就要求将搜索技术与知识处理技术相结合,不仅能从海量信息中获取知识,提供更深入的搜索结果,还能充分地利用这些知识为生产和决策服务^[5],造成了 Deep Web 集成服务和模式匹配过程存在不确定性知识。而且,集成接口模式与各 Web 数据库查询接口模式之间的匹配往往是不准确的,使得通过 Deep Web 搜索建立的知识库存在着动态性、

不一致性和不确定性等问题,而且保持 Deep Web 信息处理后形成的知识库的绝对一致和精确也是非常困难的,要求推理所依据的知识绝对一致、精确也是不切实际的,这势必造成通过 Deep Web 搜索到的信息不能被充分地利用,甚至产生有害的结果,造成网络资源的巨大浪费。因此,对面向 Deep Web 的不确定知识进行表示和实现合理的推理和利用,具有重要的理论价值和现实意义,并已成为 Deep Web 信息集成和知识工程领域的研究热点之一,引起了国内外学者的高度重视。

目前国内外学者^[6-8,10-13]对这方面的研究工作采用的理论依据主要是模态逻辑、非单调逻辑和时序逻辑等。目前国内利用逻辑方法表示和处理知识的研究也越来越受到重视,北京大学林作铨教授领导的课题组在不协调知识的缺省推理方面取得了可喜的成果^[12],中科院计算所史忠植研究员利用动态描述逻辑 DDL 作为逻辑基础处理语义 Web 知识取得了重要的进展^[10,11]。王驹等在动态描述逻辑 DDL 的基础上,对 DDL 进行模糊化扩充,提出了一种模糊动态描述逻辑 FDDL^[13],该方法主要是基于 DDL,对动作的变化状况进行模糊化扩充。而本文提出的 DFDLs 则是基于 DFL 理论,对概念

到稿日期:2009-10-30 返修日期:2010-01-14 本文受国家自然科学基金(60970015),南京信息工程大学科研启动项目资助。

方 巍(1975—),男,博士,高级工程师,CCF 会员,主要研究方向为智能信息处理、数据挖掘和知识工程,E-mail:hsfangwei@sina.com;崔志明(1961—),教授,博士生导师,CCF 高级会员,主要研究方向为智能信息处理。

变化的程度和发展趋势进行模糊度扩充,实现的理论基础和方法是不同的,所表达的对象和能力也有所区别。鉴于现有方法仅能解决一般的经典问题,要真正从本质上解决协调工作机制中它们之间存在的动态模糊性问题,这些理论还是不足的,倘若用模糊逻辑(Fuzzy Logic)来解决,也存在同样的问题,因为 Fuzzy Logic 仅能解决静态问题。因此,本文借用的理论工具是动态模糊集(Dynamic Fuzzy Sets,DFS)、动态模糊逻辑(Dynamic Fuzzy Logic,DFL)和基于此扩展的动态模糊描述逻辑(Dynamic Fuzzy Description Logic,DFDLs)。本文给出了 DFDLs 的定义和语法规则以及基于 tableau 的推理规则和算法。

2 动态模糊描述逻辑

2.1 动态模糊集(DFS)简介

定义 1 设在论域 U 上定义映射 $\tilde{A}: \tilde{U} \rightarrow [0, 1]$, $\tilde{u} \mapsto \tilde{A}(\tilde{u})$ 或 $\tilde{A}: \tilde{U} \rightarrow [0, 1]$, $\tilde{u} \mapsto \tilde{A}(\tilde{u})$, 记 $(\tilde{A}, \tilde{A}) = \tilde{A}, \tilde{A}$, 则称 $(\tilde{A}, \tilde{A})$ 为 U 上的动态模糊集,称 $(\tilde{A}(\tilde{u}), \tilde{A}(\tilde{u}))$ 为隶属函数对 $(\tilde{A}, \tilde{A})$ 的隶属度。

注:对于任何一个 $a \in [0, 1]$, 均可把它动态模糊化为 $a \xrightarrow{DF} (\tilde{a}, \tilde{a}) = \tilde{a} \text{ 或 } \tilde{a}$, 且 $\max(\tilde{a}, \tilde{a}) \triangleq \tilde{a}, \min(\tilde{a}, \tilde{a}) \triangleq \tilde{a}$ 。

当两个动态模糊(Dynamic Fuzzy, DF)子集间进行运算时,完全可以理解为是对隶属函数做相应的运算。

动态模糊逻辑是建立在动态模糊集的基础上。有关动态模糊集的理论见文献[6]。

2.2 动态模糊逻辑(DFL)简介

定义 2 一个具有动态模糊性的陈述句称为 DF 命题,用大写 $A, B, C \dots$ 表示。

例 1 “天气越来越热了。”这是一个 DF 命题。

定义 3 用来度量一个 DF 命题真假用 DF 数 $(\tilde{a}, \tilde{a}) \in [0, 1]$ 来表示,称为该命题的真假度,常用小写字母 $(\tilde{a}, \tilde{a}), (\tilde{b}, \tilde{b}), (\tilde{c}, \tilde{c}), \dots$ 表示。其中 $(\tilde{a}, \tilde{a}) = \tilde{a}$ 或 $\tilde{a}$, $\min(\tilde{a}, \tilde{a}) \triangleq \tilde{a}, \max(\tilde{a}, \tilde{a}) \triangleq \tilde{a}$ 。

定义 4 一个 DF 命题可以看成变量在区间 $[0, 1] \times [\leftarrow, \rightarrow]$ 上的取值。该变量称为动态模糊谓词变量,通常用小写字母表示。

对于 DF 变量 $(\tilde{x}, \tilde{x}), (\tilde{y}, \tilde{y}) \in [0, 1]$ 规定如下运算:

①否定“ $\neg$ ”

$(\tilde{x}, \tilde{x})$ 的否定记为 $\overline{(\tilde{x}, \tilde{x})}$, 且 $\overline{(\tilde{x}, \tilde{x})} \triangleq ((1-\tilde{x}, 1-\tilde{x}))$

②析取“ $\vee$ ”

$(\tilde{x}, \tilde{x}) \vee (\tilde{y}, \tilde{y}) \triangleq \max((\tilde{x}, \tilde{x}), (\tilde{y}, \tilde{y}))$

③合取“ $\wedge$ ”

$(\tilde{x}, \tilde{x}) \wedge (\tilde{y}, \tilde{y}) \triangleq \min((\tilde{x}, \tilde{x}), (\tilde{y}, \tilde{y}))$

④条件“ $\rightarrow$ ”

$(\tilde{x}, \tilde{x}) \rightarrow (\tilde{y}, \tilde{y}) \triangleq \overline{(\tilde{x}, \tilde{x})} \vee (\tilde{y}, \tilde{y}) \triangleq \max(\overline{(\tilde{x}, \tilde{x})}, (\tilde{y}, \tilde{y}))$

⑤双条件“ $\leftrightarrow$ ”

$(\tilde{x}, \tilde{x}) \leftrightarrow (\tilde{y}, \tilde{y}) \triangleq \min(\max(\overline{(\tilde{x}, \tilde{x})}, (\tilde{y}, \tilde{y})), \max(\overline{(\tilde{y}, \tilde{y})}, (\tilde{x}, \tilde{x})))$

定义 5 DF 命题演算公式可以定义为:

(1) 单个 DF 命题变元本身是一个合式公式;

(2) 如果 $(\tilde{x}, \tilde{x})P$ 是一个合式公式,那么 $\overline{(\tilde{x}, \tilde{x})P}$ 也是合式

公式;

(3) 如果 $(\tilde{x}, \tilde{x})P$ 和 $(\tilde{y}, \tilde{y})Q$ 是合式公式,那么 $(\tilde{x}, \tilde{x})P \vee (\tilde{y}, \tilde{y})Q, (\tilde{x}, \tilde{x})P \wedge (\tilde{y}, \tilde{y})Q, (\tilde{x}, \tilde{x})P \rightarrow (\tilde{y}, \tilde{y})Q, (\tilde{x}, \tilde{x})P \leftrightarrow (\tilde{y}, \tilde{y})Q$ 都是合式公式。

(4) 当且仅当有限次地应用(1)、(2)、(3)所得到的命题变元联结词和括号的符号串是合式公式。

其它相关公式参考文献[9]。

2.3 动态模糊描述逻辑(DFDLs)

基于逻辑的知识表示和推理是语义 Web 实现的关键技术。描述逻辑是语义 Web 的逻辑基础,支持基于概念和关系描述的知识表示和知识推理[5]。经典的描述逻辑仅能处理精确信息,而对 Deep Web 中海量高价值的动态模糊信息等不确定知识的表示,并未给出完整的逻辑体系和形式化推理规则算法。为此,我们在动态模糊逻辑(DFL)的基础上扩展经典描述逻辑(ALC)而得到动态模糊描述逻辑 DFDLs (Dynamic Fuzzy Description Logics)。例如,在 DFDLs 中,一个概念 C 表示为一个动态模糊集形如断言“ a is C ”具有真值 $[0, 1] \times [\leftarrow, \rightarrow]$ 刻画的陈述。

2.3.1 DFDLs 语法结构

DFDLs 主要由概念 N_C 和角色 N_R 组成。DFDLs 语法是用 DFL 对 FALC 的扩展。

DFDLs 的概念 C, D 和关系 R 可以按如下规则定义:

(1) $C, D \rightarrow \top | \perp | \neg C | C \cap D | C \cup D | \exists R. C | \forall R. C | \langle \alpha \rangle C | \langle \alpha \rangle D$

(2) $C, C' \rightarrow \langle \alpha \rangle C | \langle \alpha \rangle C'$

(3) $R \rightarrow r | \langle \alpha \rangle R$

其中, $C_i \in N_C, R \in N_R, \alpha$ 表示动作刻画 $[0, 1] \times [\leftarrow, \rightarrow], r$ 表示一个原子关系。

定义 6 DFDLs 谓词公式的递归定义:

(1) 原子(一阶谓词符号)是公式。

(2) 断言公式和生成公式都是公式。

(3) 若 G 和 H 是公式, T 是一个 DF 真值指派值, $(\tilde{x}, \tilde{x})$ 是 DFDLs 中的自由变量,则 $G \vee H, G \wedge H, G \rightarrow H, G \leftrightarrow H, (\tilde{x}, \tilde{x})G, \forall (\tilde{x}, \tilde{x})G, \exists (\tilde{x}, \tilde{x})G$ 均是公式。

(4) DFDLs 中所有公式有限次使用(1)、(2)和(3)后产生的公式。

DFDLs 断言公式定义如下:

定义 7 动态模糊断言是由形如 $\langle a: C \triangleright \triangleleft n \rangle$ 或 $\langle (a, b): R \triangleright \triangleleft n \rangle, a=b$ 和 $a \neq b$ 的概念断言公理、关系断言公理、个体相等公理和个体不等公理所组成的集合。其中, C 是概念, R 是关系, a, b 是个体, $\triangleright, \triangleleft$ 表示 $\geq, >, \leq$ 和 $<$, $n \in [0, 1] \times [\leftarrow, \rightarrow]$ 。

动态模糊 RBox DFRB。动态模糊 RBox DFRB 由有限传递公理和动态模糊角色形如 $\langle r \geq n \rangle, \langle r \leq n \rangle, \langle r > n \rangle, \langle a \rangle C(r), \langle r < n \rangle$ 组成, r 包含公理的角色。

动态模糊 TBox DFTB。动态模糊 TBox DFTB 由有限包含形如 $\langle c \geq n \rangle, \langle c \leq n \rangle, \langle c > n \rangle, \langle c \geq n \rangle, \alpha \sqsubseteq \beta$ 和 $\alpha \equiv \beta$ 公理的动态模糊概念组成,其中 c 是包含公理的概念, α 和 β 是动作。

动态模糊 ABox DFAB。动态模糊 ABox DFAB 由有限动态模糊概念和模糊角色断言组成,模糊角色断言公理形式为 $\langle a \geq n \rangle, \langle a \leq n \rangle, R(a, b), \langle a \rangle C(a)$ 和 $\langle a \rangle C(a, b)$, 其中 a, b 是一个概念和角色断言。

动态模糊知识库 DFKB。DFKB = (DFRB, DFTB, DFAB)由动态模糊 TBox DFTB、动态模糊 RBox DFRB 和动态模糊 ABox DFAB 组成。

2.3.2 DFDLs 语义

DFDLs 的语义是根据 Zadeh 模糊集、传统描述逻辑和动态模糊逻辑的语义解释方法给出的。对概念的动作、并、交、非等连接操作都进行动态模糊扩展, s 表示 DFDLs 状态。动态模糊解释 I 是一个领域 D 二元对 $I = (\Delta^{I(s)}, \cdot^{I(s)})$, 其中 $\Delta^{I(s)}$ 代表论域的非空集合(称为状态 s 时的领域)和论域 Δ_D 并, $\cdot^{I(s)}$ 表示映射函数将概念映射为论域 $\Delta^{I(s)}$ 的动态模糊子集, 而关系映射成 $\Delta^{I(s)} \times \Delta^{I(s)}$ 的模糊二元关系子集。每个 $R \in N_R$ 映射为 $\Delta^{I(s)} \times \Delta^{I(s)} \rightarrow [0, 1] \times [\leftarrow, \rightarrow]$ 的子集, 分别称为原子概念 A 和原子关系 R 的解释, 定义概念的动态模糊子集为 $C^{I(s)}(R^{I(s)})$ 。隶属函数 $C^{I(s)}$ 通过 $C^{I(s)}: \Delta^{I(s)} \rightarrow [0, 1] \times [\leftarrow, \rightarrow]$ 来刻画。在 DFDLs 中概念和角色映射为一个模糊集, 因而公理满足度(true)或不满足度(false)映射成真值为 $[0, 1] \times [\leftarrow, \rightarrow]$ 上的刻画。

动态模糊算子(DF) 定义成 $[\leftarrow, \rightarrow]$ 。它表示当前状态在某个时间有两种改变均势: “ $\rightarrow$ ”表示向好的或更强的方向发展, 而“ $\leftarrow$ ”表示向坏或弱的方向发展。

动态模糊语义(DFS) 描述了具有动态和模糊特征陈述的意义。DFS 由一组动态模糊规则(或公理)组成, 其基本元素表示为 $\langle s, \delta, [\leftarrow, \rightarrow] \rangle$, 这里 $[\leftarrow, \rightarrow]$ 表示动态模糊算子。 $\langle s, \delta, [\leftarrow, \rightarrow] \rangle$ 表示在当前状态 δ 下动作状态准备转向状态 s 。这里 s 表示为状态, $\leftarrow$ 是状态集符号, $s(\vec{x}, \vec{x})$ 表示变量 $(\vec{x}, \vec{x})$ 在当前状态 s 时的值或内容, $\langle \vec{x}, \vec{x}, s \rangle$ 表示变量 $(\vec{x}, \vec{x})$ 等待 s 状态下某个值出现, $s(\vec{n}, \vec{n})(\vec{d}, \vec{d})/\delta(\vec{x}, \vec{x})$ 表示出现状态 s 时用 $(\vec{n}, \vec{n})(\vec{d}, \vec{d})$ 替换状态 δ 时的变量 $(\vec{x}, \vec{x})$ 。

一般根据构造子和描述逻辑表示复杂概念和概念之间的关系。DFDLs 至少包含下列构造子: Conjunction(交) $\cap$, Disjunction(并) $\cup$, Not(取反) $\neg$, Existential quantification(存在量词) $\exists$ 和 Value restriction(值限制) $\forall$ 等。DFDLs 中还包含基本状态算子 $\alpha \times [\leftarrow, \rightarrow]$ 。

定义 8 若 DFDLs 的知识库 DFKB = (DFTB, DFRB, DFAB) 满足公式: $W \models \text{DFTB}$, $W \models \text{DFRB}$ 和 $W \models \text{DFAB}$, 那么 DFKB 是可满足的, 表示为 $W \models \text{DFKB}$, 其中 $W \models$ 表示相应的断言公式是可满足的。

2.3.3 DFDLs 推理规则

现有的模糊描述逻辑 DLs(FALC), 一般假定一个元素 s 在模型 I 有 $(\exists R. C)^I(s) \geq n$, 这里 t 满足 $\min(R^I(s, t), C^I(t)) \geq n$ 。

例 2 让 $A = \{ \langle a: \exists R. C \geq 1 \rangle, \langle a: \exists R. [C \geq 1] \leq 0 \rangle \}$ 。对于模型 I 中每个元素 s 用 $(\exists R. [C \geq 1])(s) \leq 0$, 没有 t 在 $\min(R^I(s, t), C^I(t)) \geq 1$ 。然而, A 是一致的: 有限元素 $t_1, t_2, \dots$, 例如 $\sup_{i=1,2,\dots} \{ \min(R^I(s, t_i), C^I(t_i)) \} = 1$, 和任何 $i = 1, 2, \dots, \min(R^I(s, t_i), C^I(t_i)) < 1^{[12]}$ 。

DFDLs 中所有推理问题都能归结为解释的可满足性问题。动态模糊规则是将传统规则 DF 化, 可以从以下几个方面进行:

(1) 前提 DF 化: 在规则前提条件中引入 DF 谓词和 DF 状态量词来表达 DF 关系和 DF 状态, 并定义一种 DF 匹配规则。

(2) 动作或结论 DF 化: 使规则的动作或结论具有一种

DF 度 $(\vec{a}, \vec{a})$ 或结论本身就是一个 DF 谓词或一种 DF 状态或动作本身就是一种带 DF 性的动作以操作 DF 数据库中 DF 数据。

(3) 设置规则激活阈值 $(\vec{\tau}, \vec{\tau})$ 。当前提条件的匹配度大于或等于 $(\vec{\tau}, \vec{\tau})$ 时, 规则才被激活。

(4) 设置规则的可信度 $(\vec{CF}, \vec{CF})$ 。以确定的可信度来反映 DF 规则的可信程度, 它将在推理中以某种方式影响结论或动作的可信度。

定义 9 DFDLs 公式 G 的一个解释(Explanation) I 由非空域和如下规则组成:

(1) G 中每个变量符号指定 U 中一 DF 元素。

(2) 对 G 中每个 n 元函数符号指定映射 $U \xrightarrow{T} D$ 。

(3) 对 G 中每个 n 元谓词符号指定映射 $D \xrightarrow{T} B$, 这里 B 是 DFDLs 布尔量。

定义 10 DF 产生式规则一般形式为

$P \leftarrow Q, CDF, I$

其右部一般表示一组前提或条件, 左部表示若干结论动作, 前提 Q 和结论 P 都可以是 DF 的。CDF 是 $(\vec{0}, \vec{0}) \leq CDF \leq (\vec{1}, \vec{1})$, 称为规则的置信度(Degree of Confidence)。上述规则的含义是: 如果前提 Q 在某种程度上被满足, 则能够以一定的真度推出结论 P (或执行动作 P), 规则的置信度为 CDF。

当 $Q = a_1(u_1)a_2(u_2)\dots a_n(u_n)$ 时, 若在已知 $Q^1 = a_1^1(v_1)a_2^1(v_2)\dots a_n^1(v_n) \cdot t^1$ (其中 $t^1: 0 < t^1 \leq 1$ 为知识 Q^1 的真度), 并且 $a^1 = a$ 或 $*$, $i = 1, 2, \dots, n, 0 \leq u_i \leq v_i \leq 1 (0 \leq v_i \leq u_i \leq 1), i = 1, 2, \dots, n$, 则称 Q 可匹配(Matching)。这时 Q 与 Q^1 的距离定义为:

$$d = \max\{|v_1 - u_1|, |v_2 - u_2|, \dots, |v_n - u_n|\}$$

于是前提 Q 的匹配程度定义为

$$m = \min\{1 - d, t^1\} \text{ (或 } m = t^1 * (1 - d) \text{, 等)}$$

因此, DFDLs 的概念和关系的语义可以扩展为

$$T^I(s) = \Delta^{I(s)}$$

$$\perp^{I(s)} = 0$$

$$(\rightarrow C)^{I(s)} = \Delta^{I(s)} \setminus C^{I(s)}$$

$$(C_1 \cap C_2)^{I(s)} = C_1^{I(s)} \cap C_2^{I(s)}$$

$$(C_1 \cup C_2)^{I(s)} = C_1^{I(s)} \cup C_2^{I(s)}$$

$$(\forall R. C)^{I(s)} = \{x \in \Delta^{I(s)} \mid \forall y \cdot (x, y) \in R^{I(s)} \Rightarrow y \in C^{I(s)}\}$$

$$(\exists R. C)^{I(s)} = \{x \in \Delta^{I(s)} \mid \exists y \cdot (x, y) \in R^{I(s)} \wedge y \in C^{I(s)}\}$$

$$(\geq nR)^{I(s)} = \{x \mid \#(\{y \cdot (x, y) \in R^{I(s)}\}) \geq n\}$$

$$(\leq nR)^{I(s)} = \{x \mid \#(\{y \cdot (x, y) \in R^{I(s)}\}) \leq n\}$$

$$\{o_1, \dots, o_n\}^{I(s)} = \{o_1^{I(s)}, \dots, o_n^{I(s)}\}$$

一个动态模糊概念 C 是可满足的 iff 有动态模糊解释 I 存在 $a \in \Delta^{I(s)}$, 例如 $C^{I(s)}(a) = n$, 和 $n \in [0, 1] \times [\leftarrow, \rightarrow]$, 动态模糊解释 I 满足 TBOX T iff $\forall a \in \Delta^{I(s)} R^{I(s)} \leq D^{I(s)}$, 有每个 $R \subseteq C$ 和 $\forall a \in \Delta^{I(s)} R^{I(s)} = D^{I(s)}, R \equiv C$ 。

DFDLs 中断言为 $\langle a: Con \rangle, I \models \langle a: C \geq n \rangle$ iff $C^{I(s)}(a^{I(s)}) \geq n$, 而术语公理表示为: $A = C$ 或 $A \subseteq C \mid I \models A \subseteq C$ iff $\forall x \in \Delta^{I(s)} \cdot A^{I(s)}(x) \leq C^{I(s)}$ 。

3 Tableau 算法

tableau 算法广泛用于各种描述逻辑中, 以判定概念的可满足性或概念之间的包含关系。由于 tableau 方法是基于语义的一种推理方法, 是用推理树方法来表示二元公式的逻辑

关系,因此建模方法具有适应多种逻辑的优点。tableau 方法不但能够证明逻辑公式的不可满足性,而且当公式可满足时,能够给出公式的模型。即它是一种通用的逻辑公式推理方法和模型生成方法,可用于命题逻辑和一阶逻辑的模型生成。进一步,对于动态模糊逻辑方法仍然适用。因此拟选用 tableau 方法作为动态模糊描述逻辑的可满足性推理工具。

ABox A 的一致性可以通过 tableaux 算法检查来证明其断言的可满足性。这个能通过提供分解规则来在概念表达 A 中完成。模型用所谓的完全树(completion-trees)表示,完全树中的结点对应于模型中的实例,每个结点用一个三元组 $\langle D, \triangleright \Delta, n \rangle$ 形式表示,分别表示类型、概念和结点属于某个概念 D 的隶属度。对于模糊断言,同样表示为单个结点的三元组形式,共轭概念、正向概念和反向概念三元组都可以由上述方法表示,因为扩展的规则已经分解了复杂的概念,三元组中出现的概念是初始概念的子概念。概念 D 的子概念用 $\text{sub}(D)$ 表示。因此,所有子概念出现在 ABox 中,均用 $\text{sub}(A)$ 表示。以此来表示语言语义的算子满足 De'Morgan 定律。这样,概念的反向规范形式可以通过 negation normal form (NNF) 得到。因此我们假定所有的概念在 NNF 范型中。本文将 tableau 定义中的 ABox A 扩展成动态模糊 dynamic fuzzy tableau。

一般,根据 DFDLs 构造子构建复制概念和关系。基本的 FALC 加上动态模糊算子 $(\overleftarrow{x}, \overrightarrow{x})$ 后就形成 DFDLs,从而更好地表示 Deep Web 中动态模糊的概念和关系。应用 tableau 方法可进行知识库的完备性检查。在推理模型的基础上,通过包删除、启发式增枝等技术,对多世界融合的不确定知识库进行修正和完善,使得人们可以从不确定知识库中查询到完备的知识,提高不确定知识库中知识的可信度和可依赖性。

表 1 给出了基于动态模糊 tableau 的可满足性推理算法。

表 1 tableau 算法规则

规则	描述
$(\neg)$	如果 $(1) \langle \neg C_1^{(s)}, \triangleright \Delta, n \rangle \in L(x)$ (2) 而且 $\langle C_1^{(s)}, \triangleright \Delta, 1-n \rangle \notin L(x), n \in [0, 1] \times [\leftarrow, \rightarrow]$ 那么 $L(x) \rightarrow L(x) \cup \{ \langle C_1^{(s)}, \triangleright \Delta, 1-n \rangle \}$
$(\cap \triangleright)$	如果 $(1) \langle C_1^{(s)} \cap C_2^{(s)}, \triangleright, n \rangle \in L(x)$, 而且 x 没有被直接阻塞 (2) $\{ \langle C_1^{(s)}, \triangleright, n \rangle, \langle C_2^{(s)}, \triangleright, n \rangle \} \not\subseteq L(x), n \in [0, 1] \times [\leftarrow, \rightarrow]$ 那么 $L(x) \rightarrow L(x) \cup \{ \langle C_1^{(s)}, \triangleright, n \rangle, \langle C_2^{(s)}, \triangleright, n \rangle \}$
$(\cup \triangleleft)$	如果 $(1) \langle C_1^{(s)} \cup C_2^{(s)}, \triangleleft, n \rangle \in L(x)$, 而且 x 没有被直接阻塞 (2) $\{ \langle C_1^{(s)}, \triangleleft, n \rangle, \langle C_2^{(s)}, \triangleleft, n \rangle \} \not\subseteq L(x), n \in [0, 1] \times [\leftarrow, \rightarrow]$ 那么 $L(x) \rightarrow L(x) \cup \{ \langle C_1^{(s)}, \triangleleft, n \rangle, \langle C_2^{(s)}, \triangleleft, n \rangle \}$
$(\cup \triangleright)$	如果 $(1) \langle C_1^{(s)} \cup C_2^{(s)}, \triangleright, n \rangle \in L(x)$, 而且 x 没有被直接阻塞 (2) $\{ \langle C_1^{(s)}, \triangleright, n \rangle, \langle C_2^{(s)}, \triangleright, n \rangle \} \cap L(x) = \emptyset, n \in [0, 1] \times [\leftarrow, \rightarrow]$ 那么 $L(x) \rightarrow L(x) \cup \{ C \}$, 对于某个 $C \in \{ \langle C_1^{(s)}, \triangleright, n \rangle, \langle C_2^{(s)}, \triangleright, n \rangle \}$
$(\cap \triangleleft)$	如果 $(1) \langle C_1^{(s)} \cap C_2^{(s)}, \triangleleft, n \rangle \in L(x)$, 而且 x 没有被直接阻塞 (2) $\{ \langle C_1^{(s)}, \triangleleft, n \rangle, \langle C_2^{(s)}, \triangleleft, n \rangle \} \cap L(x) = \emptyset, n \in [0, 1] \times [\leftarrow, \rightarrow]$ 那么 $L(x) \rightarrow L(x) \cup \{ C_1^{(s)} \}$, 对于某个 $C \in \{ \langle C_1^{(s)}, \triangleleft, n \rangle, \langle C_2^{(s)}, \triangleleft, n \rangle \}$
$(\exists \triangleright)$	如果 $(1) \langle \exists R. C^{(s)}, \triangleright, n \rangle \in L(x)$, 而且 x 没有被直接阻塞, $n \in [0, 1] \times [\leftarrow, \rightarrow]$ (2) x 没有一个 R-后续 y 连接三元组 $\langle R^*, \triangleright, n \rangle$ 和 $\langle C^{(s)}, \triangleright, n \rangle \in L(y)$ 那么新建一个结点 y, 赋值 $L(x, y) = \{ \langle R, \triangleright, n \rangle \}$, $L(y) = \{ \langle C^{(s)}, \triangleright, n \rangle \}$

$(\forall \triangleleft)$	如果 $(1) \langle \forall R. C^{(s)}, \triangleleft, n \rangle \in L(x)$, 而且 x 没有被直接阻塞, $n \in [0, 1] \times [\leftarrow, \rightarrow]$ (2) x 没有一个 R-后续 y 连接三元组 $\langle R^*, \triangleleft, n \rangle$ 和 $\langle C^{(s)}, \triangleleft, n \rangle \in L(y)$ 那么新增一个结点 y, 赋值 $L(x, y) = \{ \langle R, \triangleright, 1-n \rangle \}$, $L(y) = \{ \langle C^{(s)}, \triangleleft, n \rangle \}$
$(\forall \triangleright)$	如果 $(1) \langle \forall R. C^{(s)}, \triangleright, n \rangle \in L(x)$, 而且 x 没有被直接阻塞 (2) x 没有一个 R-后续 y 连接三元组 $\langle C^{(s)}, \triangleright, n \rangle \in L(y)$ 且 (3) $\langle R^*, \triangleright, 1-n \rangle$ 用 x, y 连接一个正向三元组, $n \in [0, 1] \times [\leftarrow, \rightarrow]$ 那么 $L(y) \rightarrow L(y) \cup \{ \langle C^{(s)}, \triangleright, n \rangle \}$
$(\exists \triangleleft)$	如果 $(1) \langle \exists R. C^{(s)}, \triangleleft, n \rangle \in L(x)$, 而且 x 没有被直接阻塞 (2) x 没有一个 R-后续 y 用 $\langle C^{(s)}, \triangleleft, n \rangle \in L(y)$ 而且, $n \in [0, 1] \times [\leftarrow, \rightarrow]$ (3) $\langle R^*, \triangleleft, n \rangle$ 用 x, y 连接一个正向三元组 那么 $L(y) \rightarrow L(y) \cup \{ \langle C^{(s)}, \triangleleft, n \rangle \}$
$(\forall +)$	如果 $(1) \langle \forall R. C^{(s)}, \triangleright, n \rangle \in L(x)$, Trans(R), 而且 x 没有被直接阻塞 (2) x 没有一个 R-后续 y 用 $\langle \forall R. C^{(s)}, \triangleright, n \rangle \in L(y)$ 而且 (3) $\langle R^*, \triangleright, 1-n \rangle$ 用 x, y 连接一个正向三元组, $n \in [0, 1] \times [\leftarrow, \rightarrow]$ 那么 $L(y) \rightarrow L(y) \cup \{ \langle \forall R. C^{(s)}, \triangleright, n \rangle \}$
$(\exists +)$	如果 $(1) \langle \exists R. C^{(s)}, \triangleleft, n \rangle \in L(x)$, 而且 x 没有被直接阻塞, $n \in [0, 1] \times [\leftarrow, \rightarrow]$ (2) x 没有一个 R-后续 y 用 $\langle \exists R. C^{(s)}, \triangleleft, n \rangle \in L(y)$ 而且 (3) $\langle R^*, \triangleleft, n \rangle$ 用 x, y 连接一个正向三元组 那么 $L(y) \rightarrow L(y) \cup \{ \langle \exists R. C^{(s)}, \triangleleft, n \rangle \}$

下面给出描述 DFDLs 的 ABox DFAB 的可满足性推理算法的正确性的相关定理。由于篇幅所限,其中定理的证明过程略。

定理 1 给定 DFDLs 的 ABox DFAB 和 RBox DFRB, 相对于 DFRB, DFAB 是可满足的, 当且仅当 DFAB 存在一个相对于 DFRB 的动态模糊 tableau DFT。

定理 2 DFDLs 的 ABox DFAB 的可满足性推理算法能在有限步终止。

定理 3 若 DFAB 是 DFDLs 的 ABox, 那么

1) 该算法是可终止的;

2) 给定 DFDLs 的 ABox DFAB 和 RBox DFRB。相对于 DFRB, DFAB 是可满足的, 当且仅当将 DFAB 的可满足性推理算法的扩充规则应用于 DFAB 和 DFRB 时得到一个无冲突的、完备的完全森林。

例 3 下面以查询接口属性中两类水果实例来说明其 tableau 可满足过程。

- (1) $\langle i: \exists \text{ about}, (\text{Orange} \cap \text{Apple}) \geq 0.7 \rangle$
- (2) $\langle i: \exists \text{ about}, \text{Orange} \geq 0.7 \rangle$
- (3) $\langle (i, x): \text{about} \geq 0.7 \rangle \quad \exists \geq; (1)$
 $\langle x: (\text{Orange} \cap \text{Apple}) \geq 0.7 \rangle$
- (4) $\langle x: \text{Orange} \geq 0.7 \rangle \quad \cap \geq; (3)$
 $\langle x: \text{Apple} \geq 0.7 \rangle$
- (5) $\langle x: \text{Orange} < 0.7 \rangle \quad \exists < (2), (3), (4), (5)$
- (6) Clash

采用语义 tableau 构造模型, 该方法将证明理论与语义联系起来, 使之更接近逻辑系统的表示。将 tableau 模型生成方法推广到动态模糊逻辑中, 这样生成的模型利用相关反馈技术, 通过对搜索出的不确定知识进行评价, 得到合理的约束条件, 然后对反馈的信息利用模糊定性、布尔剪枝等技术修正逻辑模型, 修改关联规则、分类与预测、聚类分析等算法的初始条件, 使得搜索出的知识逐步趋近完备和合理。

进一步可以利用 DFDLs 把 OWL 扩展为 DFOWL, 表示

Deep Web 信息集成和模式匹配过程中存在的不确定知识和推理问题。例如,可以采用 DFDLs 扩展的 OWL 来描述股票市场某只股票的变化趋势情况。如下所示:

```
<owl:Class rdf:ID="Stock">
  <owl:unionOf rdf:parse Type="Collection">
    <owl:Class rdf:about="#000001"
Type="→" fuzzy:degree="0.8"/>
    <owl:Class rdf:about="#000002"
Type="←" fuzzy:degree="0.75"/>
  </owl:unionOf>
</owl:Class>
```

结束语 本文在 Deep Web 中的不确定知识的动态性、模糊性等方面的基础上,研究了将传统描述逻辑和动态模糊理论相融合,建立一套基于动态模糊逻辑的面向 Deep Web 的不确定知识的表示、模型生成的理论与方法,即动态模糊描述逻辑 (DFDLs),并提出了一种基于动态模糊 tableau 的 DFDLs 的 ABox 约束下的可满足性推理算法。DFDLs 可以用于面向 Deep Web 的不确定知识的合理推理和利用,它弥补了描述逻辑在模糊性和动态性方面的不足,既为语义 Web 提供了进一步的逻辑支撑,也为表示和利用 Deep Web 中动态和模糊等不确定知识提供了强有力的理论基础。提出有效的推理算法并分析推理算法的可判断性和时间复杂性,是今后需要进一步研究的问题。

参考文献

- [1] Barbosa L, Freire J. Siphoning Hidden-Web Data through Keyword-based Interfaces[C]//Brazilian Symposium on Databases (SBB), 2004
- [2] Ghanem T M, Aref W G. Databases Deepen the Web[J]. IEEE

Computer, 2004, 73(1): 116-117

- [3] He B K, Chang C-C, Han J. Mining Complex Matchings Across Web Query Interfaces[C]//Proceedings of the 9th ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery (SIGMOD-DMKD'04). Paris, France, June 2004
- [4] Baader F, Nutt W. Basic Description Logic [M]. Handbook of Description Logic. Cambridge University Press, January 2003
- [5] Horrocks I. Using an expressive description logic: FaCT or fiction? [C]//Proc. of the 6th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'98). 2001, 199-204
- [6] Horrocks I, Patel-Schneider P F. Reducing OWL entailment to description logic satisfiability[C]//the 2003 International Semantic Web Conference (ISWC 2003). 2003, 17-29
- [7] Hendler J, McGuinness D L. The DARPA Agent Markup Language[J]. IEEE Intelligent Systems, 2000, 15(6): 67-73
- [8] Chang Liang, Shi Zhongzhi, Qiu Lirong, et al. Dynamic Description Logic: Embracing Actions into Description Logic[C]//Proceedings of the 2007 International Workshop on Description Logics. 2007
- [9] 李凡长, 刘贵全, 余玉梅. 动态模糊逻辑引论[M]. 昆明: 云南科技出版社, 2005
- [10] 史忠植, 董明楷, 蒋运承, 等. 语义 Web 的逻辑基础[J]. 中国科学 (E 辑), 2004, 34(10): 1123-1138
- [11] 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
- [12] 梅婧, 林作铨. 从 ALC 到 SHOQ(D): 描述逻辑及其 Tableau 算法[J]. 计算机科学, 2005, 32(3): 1-11
- [13] 王驹, 蒋运承, 唐素勤. 一种模糊动态描述逻辑[J]. 计算机科学与探索, 2007, 1(2): 216-227

(上接第 160 页)

的设计方案连同其实现代码一起进行重用是支持实现阶段而非设计阶段的重用, 本文基于反射式软件体系结构的软件重用是针对软件设计阶段。

我们提出的基于反射机制的软件体系结构重用方法——ArchBean 方法是一种更通用、更便捷的体系结构制品本身的重用方法, 它具有统一的体系结构建模方法的信息, 可以完成设计阶段软件体系结构制品的重用; 为了保证重用软件体系结构的正确性和一致性, 本文给出反射式软件体系结构的元级和基本级的一致性性质的定义, 以及经过重用操作后, 元级和基本级的一致性的证明方法和过程。所以本文以及后续的研究内容在一定程度上可以较好地解决软件体系结构重用存在的主要问题。

结束语 本文的研究工作与已有研究的不同之处在于: (1) 提供了一种在设计阶段支持软件体系结构重用的反射式软件体系结构。(2) 基于 Object-Z 形式化描述支持软件重用的部分操作; 给出反射式软件体系结构的元级和基本级的一致性性质的定义, 以及经过重用操作后元级和基本级的一致性的证明方法和过程。

作为今后的工作, 我们将完善 PMB 协议的定义, 开发一个基于 Eclipse 插件技术的支撑工具 ArchBean Studio 来实现

反射式软件体系结构的基于 Object-Z 形式化描述, 以及自动实现基级和元级的一致性验证。

参考文献

- [1] Binns P, Engelhart, Vestal M. Domain-Specific Software Architectures for Guidance, Navigation, and Control [J]. Software Eng. and Knowledge Eng., 1996, 6(2): 1011-1017
- [2] Shaw M. Some Patterns for Software Architecture, Pattern Languages of Program Design [M]//Vlissides, Coplien and Kerth, eds. Addison-Wesley, 1996, 255-270
- [3] Schmerl B, Garlan D. AcmeStudio: Supporting Style-Centered Architecture Development [C]//Proceedings of International Conference on Software Engineering. Edinburgh, Scotland, May 2004
- [4] Froehlich G, Hoover H J, Liu Ling, et al. Sorenson. Designing object-oriented frameworks, In CRC Handbook of Object Technology [M]. CRC Press, 1998
- [5] 叶鹏, 应时, 罗巨波. 一种支持软件体系结构重用的元信息模型[J]. 计算机科学, 2009, 36(5): 145-150
- [6] 罗巨波, 应时. 一种支持软件体系结构重用的反射机制及其形式化[J]. 计算机科学, 2009, 36(8): 145-148