arXiv:2303.07192v1 [cs.AI] 1 Mar 2023

PN-OWL：从OWL本体中学习模糊概念包含的两阶段算法

Franco Alberto Cardillo Franca Debole UmbertoStraccia
CNR-ILC CNR-ISTI CNR-ISTI 意大利比萨 意大利比萨 意大利比萨

2023年3月14日

摘要

OWL本体是一种颇受欢迎的方式，可用于描述由类、类之间的关系以及类实例构成的结构化知识。

本文针对OWL本体中的一个目标类 T，重点关注具有实数值和布尔值数据属性的本体，探讨如何学习分级模糊概念包含公理，以描述个体被分类为类 T实例所需的充分条件。

为此，我们提出了 PN-OWL，这是一种由 P 阶段和 N 阶段组成的两阶段学习算法。大致而言，在 P 阶段，算法会尽可能多地覆盖正例（提高 $recall$），同时避免过度损害 $precision$；而在 N 阶段，算法会尽可能排除 P 阶段所覆盖的假阳性。 PN-OWL随后通过聚合函数组合 P 阶段和 N 阶段学习到的模糊包含公理，并对其进行聚合，从而最终判定某个个体是否为 T 的实例。

我们还通过实验展示了该算法的有效性。其一个有趣的特点是，模糊数据类型能够自动构建，学习到的模糊概念包含可以直接表示为模糊 OWL 2，因此可以使用任何模糊 OWL 2 推理机自动判定或分类某个个体是否属于目标类 T，并确定其所属程度。

1 引言

OWL 2 本体 [67] 是表示结构化 知识的常用方式，其形式语义基于描述逻辑 （DLs） [6]。描述逻辑的基本要素包括概念描述、概念之间的继承关系以及概念的实例。

在这项工作中，我们聚焦于这样一个问题：如何根据 OWL 2 本体中的术语和实例，自动学习模糊 $\mathcal{EL}$（**D**）概念包含公理。尽管已有大量关于描述逻辑的研究，但与归纳逻辑程序设计（ILP）场景相比，将机器学习技术应用于 OWL 2 本体的研究相对较少（有关 ILP 的更多信息，参见例如 [69, 70] ）。有关综述，请参阅 [54, 71] 以及第5节。

本文所要解决的问题如下：给定 OWL 本体的一个目标类 T，学习规则形式的分级模糊 $\mathcal{EL}$（**D**） [13, 16, 86] 概念包含公理，以描述被归类为该类实例的个体所需的充分条件 T。

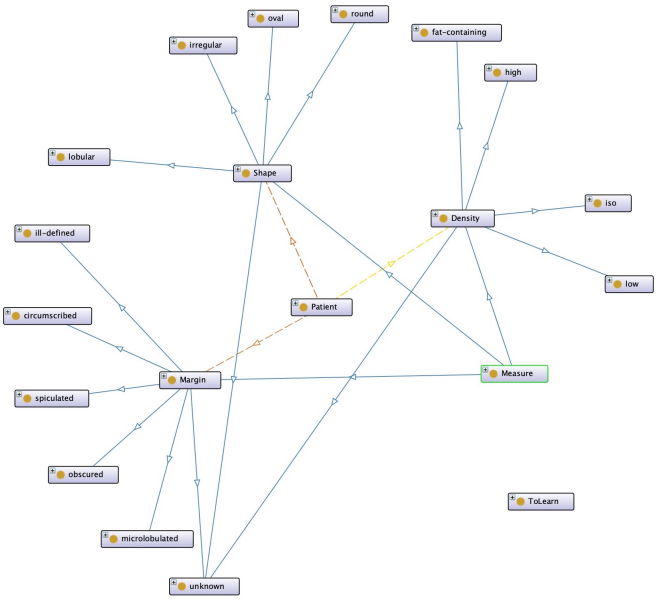

图 1：乳腺摄影本体的摘录。

以下示例说明了我们要解决的问题。[1]

示例 1.1 考虑一个描述乳腺摄影图像分析中有意义实体的本体 *[18, 20]* 。图 *1* 展示了该本体的一个摘录。现在，假设我们有一组患有癌症的患者（正例），以及另一组未患癌症的患者（负例）。那么，我们可能会问，患癌症的患者（我们的目标类 T）有哪些特征。接着，我们可以从本体中学习到以下模糊 $\mathcal{EL}(\mathbf{D})$ 概念包含关系（以所谓的模糊 *OWL 2* 语法 *[13]*表示）[2]

(蕴含 (且 (存在 hasDensity 含脂肪的) **(存在** hasMargin 毛刺状的) **(存在** hasShape 不规则的) **(** 存在 hasAge hasAgehigh)) 癌症 0。86) ,**

其中模糊集 `hasAge high` 定义为 s

(define-fuzzy-concept `hasAge_high right- (0,150,60,80))`

换言之，

> “如果密度为含脂肪的，边缘呈毛刺状，形状不规则且年龄较高，那么就患有癌症，其置信度为 0.86”。

在这项工作中，目标与例如 [20, 53, 87] 相同，只不过现在我们提出将 PN-rule [2, 3, 40, 41]算法改编后用于（模糊）OWL 场景。此外，与 [51, 87], 类似，我们继续支持所谓的模糊概念描述 和模糊具体域 [59, 85, 86], , 例如表达式（**某个** hasAge hasAge 高）（也就是说，一位老年人）

[1]另请参阅 *e.g.* [20, 51, 53, 87] ，其中有一个类似的示例。

[2]`http://www.umbertostraccia.it/cs/software/fuzzyDL/fuzzyDL.html`

在上文的示例 1.1 中，它是一个模糊概念，也就是说，一个个体是否属于该类不一定是二元是非问题，而是在 $[0,1]$ 中取决于（真）值的程度。

例如，在我们的示例中，程度取决于一个人的年龄：年龄越大，这个人就越年长，例如，可通过所谓的右肩函数建模（见图 2(d)）。在这里，属性 hasAge 的取值范围变成了所谓的模糊具体域 [85, 86]。

我们回顾一下，PN-rule 的基本原理包括一个*P* 阶段，在此阶段学习正规则（称为*P* 规则），以尽可能多地覆盖目标类的实例，同时将非正例率维持在合理水平；还包括一个*N* 阶段，在此阶段学习负规则（称为*N* 规则），以移除 P 阶段所覆盖的大多数非正例。随后，使用这两组规则构建一种决策方法，以判定一个对象是否为目标类的实例 [2, 3, 40, 41]。值得注意的是，使这种方法有别于其他所有方法的，是它的第二阶段。该阶段学习 N 规则，以整体移除由所有 P 规则的并集覆盖的非正例（即所谓的假阳性）。

以下是我们称为 PN-OWL 的两阶段算法的主要特点：

- 在 P 阶段，它会生成一组模糊 $\mathcal{EL}$(D)包含公理，即 P 规则，这些公理尽可能多地覆盖目标类 T 的实例，同时不过多增加非正例的数量（即，尝试提高所谓的召回率）；
- 在 N 阶段，它会生成一组模糊 $\mathcal{EL}$(D)包含公理，即 N 规则，这些公理尽可能多地覆盖类 T 的非正例实例（即，尝试提高所谓的精确率）；
- 随后将这些模糊包含公理组合（聚合）成一个新的模糊包含公理，用于描述个体被归类为目标类 T 的实例所需的充分条件（即，该组合旨在提高整体效果，例如提高所谓的 F1 度量）；
- 所有模糊包含公理都可能包含模糊概念和模糊具体论域，其中每条公理都带有一个加权系数（具体称为置信度 或精确率）；
- 所有生成的模糊概念包含公理都可以直接编码为模糊 *OWL 2*公理 [12, 13]。因此，可以使用模糊 OWL 2 推理机（例如*fuzzyDL* [11, 15], ）自动判定某个个体是否属于目标类，以及其属于目标类的程度 T。

我们将通过一项实验来展示 PN-OWL 的有效性。实验结果表明，与仅基于 P 阶段的基线相比，组合方法的有效性有所提升。

下文将按如下结构展开：第2节为完整起见，回顾本文将用到的重要概念。接着，第3节将介绍我们的算法PN-OWL，并在第4节对其进行评估。第5节将我们的工作与迄今为止密切相关的工作进行比较。第6节总结全文，并指出一些后续研究方向。

2 背景

我们介绍本文将使用的与（数学）模糊逻辑和模糊描述逻辑相关的主要概念（另见 [16, 86]）。

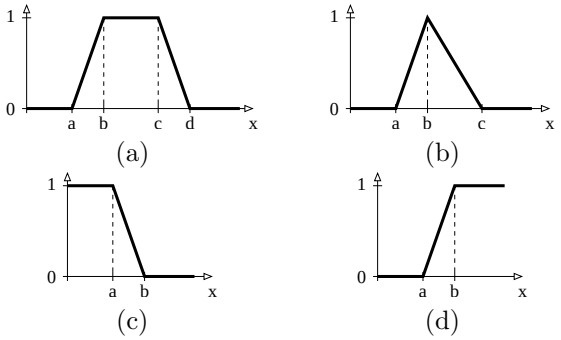

图 2：（a）梯形函数 $trz(a,\ b,\ c,\ d)$，（b）三角函数 $tri(a,\ b,\ c)$，（c）左肩函数 $ls(a,\ b)$，以及（d）右肩函数 $rs(a,\ b)$。

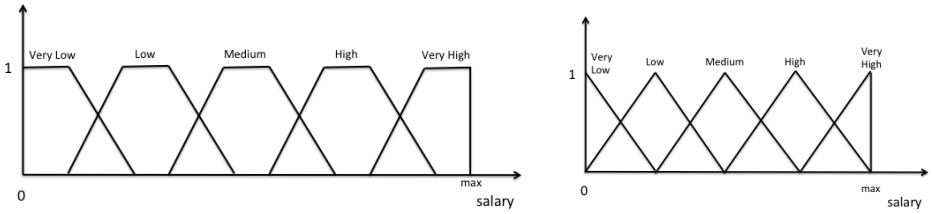

图 3：薪资的均匀模糊集：梯形（左）或三角形（右）。

数学模糊逻辑。模糊逻辑是模糊集的逻辑 [93]。定义在可数清晰集 X 上的模糊集 A 是一个函数 $A\colon X \to [0,1]$，称为 A 的模糊隶属函数。清晰集 A 则由隶属函数 $A\colon X \to \{0,1\}$ 来刻画。通常，模糊集运算符合 $(A \cap B)(x) = \min(A(x)\ B(x))$、$(A \cup B)(x) = \max(A(x)\ B(x))$ 以及 $_\ _A(x) = 1 - A(x)$（A 是 A 的集合补集）；模糊集的基数定义为 $|A| = \sum_{x \in X} A(x)$，而 A 与 B 之间的包含度定义为 $deg(A, B) = \frac{|A \cap B|}{|A|}$。

梯形函数、三角函数、左肩函数和右肩函数常用于定义模糊集的隶属函数（见图 2）。

一种简单且通常令人满意的方法是，将个人年龄值（由最小值和最大值限定）的范围均匀划分为 3、5 或 7 个模糊集，并使用三角函数（或梯形函数）定义隶属函数（见图 3）。另一种常用方法是采用所谓的*C* 均值 模糊聚类算法（参见例如[8]），将数据划分为 3、5 或 7 个簇，其中模糊隶属函数是围绕各簇质心构建的三角函数（另见 [38]）。

在数学模糊逻辑 [37], 中，关于公式 ϕ 相对于解释 $\mathcal{I}$要么为真、要么为假的约定有所改变，其真假程度由有序尺度衡量，该尺度不再是 $\{0,1\}$，而通常是 $[0,1]$。这一程度称为公式 ϕ 在解释 $\mathcal{I}$中的真度。模糊公式的形式为 $\langle \phi, \alpha \rangle$，其中 $\alpha \in (0,1]$，而 ϕ是一个一阶逻辑（FOL）公式，用于表示 ϕ 的真度大于或等于 α。从语义的角度来看，模糊解释 $\mathcal{I}$ 将每个原子公式

表 1：模糊逻辑的真值组合函数。

	卢卡西维茨	G¨odel	乘积
$\alpha_1 \otimes \alpha_2$	$\max(\alpha_1 + \alpha_2 - 1, 0)$	$\min(\alpha_1, \alpha_2)$	$\alpha_1 \cdot \alpha_2$
$\alpha_1 \oplus \alpha_2$	$\min(\alpha_1 + \alpha_2, 1)$	$\max(\alpha_1, \alpha_2)$	$\alpha_1 + \alpha_2 - \alpha_1 \cdot \alpha_2$
$\alpha_1 \Rightarrow \alpha_2$	$\min(1 - \alpha_1 + \alpha_2, 1)$	$\begin{cases} 1 & \text{若 } \alpha_1 \leq \alpha_2 \\ \alpha_2 & \text{否则} \end{cases}$	$\begin{cases} 1 & \text{若 } \alpha_1 \leq \alpha_2 \\ \alpha_2/\alpha_1 & \text{否则} \end{cases}$
$\ominus \alpha$	$1 - \alpha$	$\begin{cases} 1 \text{ 如果} \\ \quad \alpha = 0 \\ 0 \qquad \text{否则} \end{cases}$	$\begin{cases} 1 \text{ 如果} \\ \quad \alpha = 0 \\ 0 \qquad \text{否则} \end{cases}$

映射到 $[0, 1]$，然后按归纳方式扩展到所有一阶逻辑公式，如下所示：

$$\begin{aligned} \mathcal{I}(\phi \wedge \psi) &= \mathcal{I}(\phi) \otimes \mathcal{I}(\psi) \ , \ \mathcal{I}(\phi \vee \psi) = \mathcal{I}(\phi) \oplus \mathcal{I}(\psi) \\ \mathcal{I}(\phi \rightarrow \psi) &= \mathcal{I}(\phi) \Rightarrow \mathcal{I}(\psi) \ , \ \mathcal{I}(\neg \phi) = \ominus \mathcal{I}(\phi) \\ \mathcal{I}(\exists x.\phi(x)) &= \sup_{y \in \Delta^{\mathcal{I}}} \mathcal{I}(\phi(y)) \ , \ \mathcal{I}(\forall x.\phi(x)) = \inf_{y \in \Delta^{\mathcal{I}}} \mathcal{I}(\phi(y)) \ , \end{aligned}$$

其中，$\Delta^{\mathcal{I}}$ 是 $\mathcal{I}$的（非空）论域，而 $\otimes$、$\oplus$、$\Rightarrow$和 $\ominus$ 分别是所谓的*t*-范数、*t*-余范数、蕴涵函数和否定函数，它们分别将布尔合取、析取、蕴涵和否定推广到模糊情形。

通常区分三种不同的逻辑，即 卢卡西维茨逻辑、G¨odel逻辑和乘积逻辑 [37],[3]，它们的真值组合函数列于表1。

r-蕴涵 是一种由连续 t-范数$\otimes$的剩余 得到的蕴涵函数，即 $\alpha_1 \Rightarrow \alpha_2 = \sup\{\alpha_3 \mid \alpha_1 \otimes \alpha_3 \leq \alpha_2\}$。另请注意，给定一个 r-蕴涵 $\Rightarrow_r$，我们还可以通过 $\alpha \Rightarrow_r 0$ 为每个 $\alpha \in [0, 1]$定义其相关否定 $\ominus_r \alpha$ 。

可满足性和逻辑后承的概念按标准方式定义：若模糊解释 $\mathcal{I}$ 满足 模糊公式 $\langle \phi, \alpha \rangle$，或者 $\mathcal{I}$ 是 $\langle \phi, \alpha \rangle$的模型，记作$\mathcal{I} \models \langle \phi, \alpha \rangle$，当且仅当 $\mathcal{I}(\phi) \geq \alpha$。值得注意的是，由 $\langle \phi, \alpha_1 \rangle$ 和 $\langle \phi \rightarrow \psi, \alpha_2 \rangle$ 可得出结论（若将 $\rightarrow$ 解释为 r-蕴涵）：$\langle \psi, \alpha_1 \otimes \alpha_2 \rangle$（这一推理称为模糊假言推理）。

模糊描述逻辑基础。这里我们回顾模糊描述逻辑 $\mathcal{ALC}_{@}$**(D)**。它扩展了广为人知的模糊描述逻辑 $\mathcal{ALC}$**(D)** [85]，并加入了聚合概念 构造 [14]（用符号 @ 表示）。$\mathcal{ALC}_{@}$**(D)**具有足够的表达能力，能够涵盖我们将在此讨论的模糊描述逻辑的主要要素。

我们从模糊具体域的概念开始，它是一个元组$\mathbf{D} = \langle \Delta^{\mathbf{D}}, \cdot^{\mathbf{D}} \rangle$，其中包含数据类型域 $\Delta^{\mathbf{D}}$ 以及一个映射 $\cdot^{\mathbf{D}}$。该映射为每个数据值分配 $\Delta^{\mathbf{D}}$中的一个元素，并为每个一元数据类型谓词$\mathbf{d}$ 分配 $\Delta^{\mathbf{D}}$上的一元模糊关系。因此，$\cdot^{\mathbf{D}}$ 确实将每个数据类型谓词映射为一个从 $\Delta^{\mathbf{D}}$ 到 $[0, 1]$的函数。在数值论域中，典型的数据类型谓词$\mathbf{d}$ 由众所周知的隶属函数刻画（另见图2）

$$\begin{aligned} \mathbf{d} \ \rightarrow \ & ls(a, b) \mid rs(a, b) \mid tri(a, b, c) \mid trz(a, b, c, d) \\ & \mid \ \geq_v \ \mid \ \leq_v \ \mid \ =_v \ , \end{aligned}$$

[3]值得注意的是，有一个定理指出，任何其他连续 t-范数都可以由它们组合得到。

其中， $\geq_v$ （分别是 $\leq_v$ 和 $=_v$）对应于不小于（分别是不大于且等于）值 v的数据值所构成的清晰集。

聚合算子（AOs）是用于组合不同信息片段的数学函数。存在大量不同的 AOs，它们对数据（数据类型）的假设以及对我们可以纳入模型中的信息类型的假设各不相同 [90]。AO 没有标准定义。通常，给定一个论域 $\mathbb{D}$（例如实数集），维度为 n 的 AO 是一个映射 @： $\mathbb{D}^n \to \mathbb{D}$。对于我们而言， $\mathbb{D} = [0,1]$。因此，AO 会聚合n 个关于 n 种不同标准的值。在我们的场景中，这些标准将通过模糊本体中的模糊概念来表示，并且我们假定语言中有一个有限的 AO 族 $@_1$、……、$@_l$。

现在，考虑两两不相交的字母表 **I**、**A** 和 **R**，其中 **I** 是个体集合，**A** 是概念名称集合（也称为原子概念或类名），而 **R**是角色名称集合。每个角色要么是对象属性，要么是数据类型属性。概念集合由概念名称 A 通过联结词以及基于对象属性 R和数据类型属性 S的量化构造构建，如下列语法规则（$n_i \geq 1$）所述:

$$\begin{aligned} C \quad \to \quad & \top \mid \bot \mid A \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \neg C \mid C_1 \to C_2 \mid \\ & \exists R.C \mid \forall R.C \mid \exists S.\mathbf{d} \mid \forall S.\mathbf{d} \mid \\ & @_i(C_1, \ldots, C_{n_i}) \ . \end{aligned}$$

ABox $\mathcal{A}$ 由有限个断言公理组成。断言公理的形式为 $\langle a{:}C, \alpha\rangle$ （称为概念断言， a 是概念 C 的一个实例，其真值度大于或等于 α），或者为 $\langle (a_1, a_2){:}R, \alpha\rangle$（称为角色断言，$(a_1, a_2)$ 是对象属性 R 的一个实例，其真值度大于或等于 α）；其中， a, a_1, a_2 是个体名称，C 是一个概念， R 是一个对象属性， $\alpha \in (0,1]$ 是一个真值。术语盒（TBox） $\mathcal{T}$是由有限个一般概念包含（GCI）公理组成的集合，其中模糊 GCI 的形式为 $\langle C_1 \sqsubseteq C_2, \alpha\rangle$ （C_1 是 C_2 的子概念，其真值度大于或等于 α），其中C_i 是一个概念， $\alpha \in (0,1]$。我们可以省略公理的真值度 α；在这种情况下，假定 $\alpha = 1$，并称该公理为清晰公理。我们还将 $C_1 = C_2$ 作为两个 GCI$C_1 \sqsubseteq C_2$ 和 $C_2 \sqsubseteq C_1$的简写。我们也可以将形式为 $\langle C \sqsubseteq A, \alpha\rangle$的模糊 GCI（其中 A 是概念名称）称为规则，并将 C 称为其规则体。知识库（KB）是一个二元组 $\mathcal{K} = \langle \mathcal{T}, \mathcal{A}\rangle$，其中 $\mathcal{T}$是 TBox， $\mathcal{A}$ 是 ABox。我们用 $\mathrm{I}_{\mathcal{K}}$ 表示 $\mathcal{K}$中出现的个体集合。

关于语义，我们先固定一种模糊逻辑、一个模糊具体域 $\mathrm{D} = \langle \Delta^{\mathbf{D}}, \cdot^{\mathrm{D}}\rangle$ 以及聚合算子 $@_i$: $[0,1]^{n_i} \to [0,1]$。现在，与经典描述逻辑不同，在经典描述逻辑中，解释$\mathcal{I}$ 将例如概念 C 映射为个体集合 $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$，即 $\mathcal{I}$ 将 C 映射为函数$C^{\mathcal{I}}$: $\Delta^{\mathcal{I}} \to \{0,1\}$ （个体要么属于 C 的外延，要么不属于它）；而在模糊描述逻辑中， $\mathcal{I}$ 将 C 映射为函数 $C^{\mathcal{I}}$: $\Delta^{\mathcal{I}} \to [0,1]$ ，因此，个体以某种程度属于 C 的外延，该程度取值于 $[0,1]$, ，也就是说， $C^{\mathcal{I}}$ 是一个模糊集。具体而言，模糊解释是一个二元组 $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$，由一个非空的（清晰的）集合 $\Delta^{\mathcal{I}}$ （论域）以及一个模糊解释函数 $\cdot^{\mathcal{I}}$构成，该函数赋予：（i）每个原子概念 A一个函数 $A^{\mathcal{I}}$: $\Delta^{\mathcal{I}} \to [0,1]$;（ii）每个对象属性 R一个函数 $R^{\mathcal{I}}$: $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0,1]$；（iii）每个数据类型属性$S$ 一个函数 $S^{\mathcal{I}}$: $\Delta^{\mathcal{I}} \times \Delta^{\mathbf{D}} \to [0,1]$；（iv）每个个体 a一个元素 $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$，使得$a^{\mathcal{I}} \neq b^{\mathcal{I}}$ 若 $a \neq b$ （即所谓的唯一名称假设）；以及（v）每个数据值 v 一个元素 $v^{\mathcal{I}} \in \Delta^{\mathbf{D}}$。现在，模糊解释函数按如下方式扩展到概念

如下（其中 $x \in \Delta^{\mathcal{I}}$）：

$$\top^{\mathcal{I}}(x) = 1\ ,\ \bot^{\mathcal{I}}(x)\ =\ 0\ ,\ (C \sqcap D)^{\mathcal{I}}(x) = C^{\mathcal{I}}(x) \otimes D^{\mathcal{I}}(x)$$
$$(C \sqcup D)^{\mathcal{I}}(x) = C^{\mathcal{I}}(x) \oplus D^{\mathcal{I}}(x)\ ,\ (\neg C)^{\mathcal{I}}(x) = \ominus C^{\mathcal{I}}(x)$$
$$(C \to D)^{\mathcal{I}}(x) = C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x)\ ,\ (\forall R.C)^{\mathcal{I}}(x) = \inf_{y \in \Delta^{\mathcal{I}}} \{R^{\mathcal{I}}(x,y) \Rightarrow C^{\mathcal{I}}(y)\}$$
$$(\exists R.C)^{\mathcal{I}}(x) = \sup_{y \in \Delta^{\mathcal{I}}} \{R^{\mathcal{I}}(x,y) \otimes C^{\mathcal{I}}(y)\}\ ,\ (\forall S.\mathbf{d})^{\mathcal{I}}(x) = \inf_{y \in \Delta^{\mathbf{D}}} \{S^{\mathcal{I}}(x,y) \Rightarrow \mathbf{d}^{\mathbf{D}}(y)\}$$
$$(\exists S.\mathbf{d})^{\mathcal{I}}(x) = \sup_{y \in \Delta^{\mathbf{D}}} \{S^{\mathcal{I}}(x,y) \otimes \mathbf{d}^{\mathbf{D}}(y)\}\ ,$$
$$(@_i(C_1, \ldots, C_{n_i}))^{\mathcal{I}}(x) = @_i({C_1}^{\mathcal{I}}(x), \ldots, {C_{n_i}}^{\mathcal{I}}(x))\ .$$

公理的可满足性由以下条件定义：(i) $\mathcal{I}$ 满足公理$\langle a{:}C, \alpha\rangle$，当且仅当 $C^{\mathcal{I}}(a^{\mathcal{I}}) \geq \alpha$；(ii) $\mathcal{I}$满足公理 $\langle(a, b){:}R, \alpha\rangle$，当且仅当 $R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \geq \alpha$；(iii) $\mathcal{I}$满足公理 $\langle C \sqsubseteq D, \alpha\rangle$，当且仅当$(C \sqsubseteq D)^{\mathcal{I}} \geq \alpha$，且[4] $(C \sqsubseteq D)^{\mathcal{I}} = \inf_{x \in \Delta^{\mathcal{I}}} \{C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x)\}$。$\mathcal{I}$是 $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}\rangle$ 的一个模型，当且仅当 $\mathcal{I}$ 满足 $\mathcal{K}$中的每条公理。如果 $\mathcal{K}$ 有一个模型，我们就称 $\mathcal{K}$ 是可满足的（或一致的）。我们称 $\mathcal{K}$ 蕴含公理 τ，记作 $\mathcal{K} \models \tau$，当 $\mathcal{K}$ 的任意模型都满足 τ时。对于形式为 $C \sqsubseteq D$, $a{:}C$或$(a, b){:}R$的 τ，其最佳蕴含度记作 $bed(\mathcal{K}, \tau)$，定义如下：

$$bed(\mathcal{K}, \tau) = \sup\{\alpha \mid \mathcal{K} \models \langle \tau, \alpha\rangle\}\ .$$

注释 *1* 请注意，$bed(\mathcal{K}, a{:}C) = 1$（即 $\mathcal{K} \models a{:}C$）蕴含 $bed(\mathcal{K}, a{:}\neg C) = 0$成立，类似地，$bed(\mathcal{K}, a{:}\neg C) = 1$（即 $\mathcal{K} \models a{:}\neg C$）蕴含 $bed(\mathcal{K}, a{:}C) = 0$ 成立。然而，这两种情况的逆命题均不成立。此外，$bed(\mathcal{K}, a{:}C) = \alpha_1 > 0$ 和 $bed(\mathcal{K}, a{:}\neg C) = \alpha_2 > 0$ 也可能同时成立。

现在，考虑概念 C、规则 $C \sqsubseteq A$、知识库 $\mathcal{K}$ 以及个体集合 I。然后，C 关于 $\mathcal{K}$ 和 I 的基数（记作 $|C|^{\mathsf{I}}_{\mathcal{K}}$）定义为

$$|C|^{\mathsf{I}}_{\mathcal{K}} = \sum_{a \in \mathsf{I}} bed(\mathcal{K}, a{:}C)\ . \tag{1}$$

清晰基数（记作 $\lceil C\rceil^{\mathsf{I}}_{\mathcal{K}}$）的定义类似，只需将公式 1 中的项 $bed(\mathcal{K}, a{:}C)$ 替换为 $\lceil bed(\mathcal{K}, a{:}C)\rceil$。

最后，我们称规则 $C \sqsubseteq A$ 相对于 $\mathcal{K}$ 应用于个体 a 的结果为 $bed(\mathcal{K},\ C{:}a)$；若 $bed(\mathcal{K},\ C{:}a) > 0$，则称规则 $C \sqsubseteq A$ 适用于个体 a（相对于 $\mathcal{K}$）。

3 PN-OWL

首先，我们介绍要研究的学习问题。

3.1 学习问题

我们将要研究的学习问题概括如下。考虑 r

1. 一个可满足的清晰知识库 $\mathcal{K}$ 及其个体 $\mathrm{I}_{\mathcal{K}}$；2.
一个目标概念名称 T；

[4]然而请注意，在标准逻辑下，$\sqsubseteq$ 被解释为 $\Rightarrow_z$，而不是 $\Rightarrow_{kd}$。

3. 一个相关的分类函数 $f_T\colon \mathsf{I}_{\mathcal{K}} \to \{-1, 0, 1\}$，其中对于每个 $a \in \mathsf{I}_{\mathcal{K}}$，这些值（标签）对应于

$$f_T(a) = \begin{cases} 1 & a \text{ is a } \textit{positive} \text{ example w.r.t. } T \\ -1 & a \text{ is a } \textit{negative} \text{ example w.r.t. } T \\ 0 & a \text{ is an } \textit{unlabelled} \text{ example w.r.t. } T \end{cases}$$

4. 将示例划分为

$$\begin{aligned} \mathcal{E}^+ &= \{(a, 1) \mid a \in \mathsf{I}_{\mathcal{K}}, f_T(a) = 1\} \triangleright \text{the positive examples} \\ \mathcal{E}^- &= \{(a, -1) \mid a \in \mathsf{I}_{\mathcal{K}}, f_T(a) = -1\} \triangleright \text{the negative examples} \\ \mathcal{E}^u &= \{(a, 0) \mid a \in \mathsf{I}_{\mathcal{K}}, f_T(a) = 0\} \triangleright \text{the unlabelled examples} \end{aligned}$$

其中假定 $\mathcal{E}^+ \neq \emptyset$。我们将 $\mathcal{E} = \mathcal{E}^+ \cup \mathcal{E}^- \cup \mathcal{E}^u$ 定义为所有示例的集合，并用 $\overline{\mathcal{E}^+} = \mathcal{E} \setminus \mathcal{E}^+$ 表示 非正例示例的集合。

5. 个体集合 $\mathsf{I}_{\mathcal{S}} = \{a \mid (a,\ l) \in \mathcal{S}\}$，其中 $\mathcal{S} \subseteq \mathcal{E}$ 是一个示例集合。此外，我们定义

$$\begin{aligned} \mathsf{I}_{\mathcal{E}^+} &= \{a \mid (a, 1) \in \mathcal{E}^+\} \triangleright \text{the positive individuals} \\ \mathsf{I}_{\mathcal{E}^-} &= \{a \mid (a, -1) \mid a \in \mathcal{E}^-\} \triangleright \text{the negative individuals} \\ \mathsf{I}_{\mathcal{E}^u} &= \{a \mid (a, 0) \mid a \in \mathcal{E}^u\} \triangleright \text{the unlabelled individuals} \\ \mathsf{I}_{\overline{\mathcal{E}^+}} &= \mathsf{I}_{\mathcal{K}} \setminus \mathsf{I}_{\mathcal{E}^+} \triangleright \text{the non-positive individuals} \end{aligned}$$

6. 分类器 $\mathcal{H} = \{h$ 的 假设空间：$\mathsf{I}_{\mathcal{K}} \to [0, 1]\}$；

7. 个体-标签对的 训练集 $\mathcal{E}_{train} \subset \mathcal{E}$，其中 $\mathcal{E}_{train} \cap \mathcal{E}^+ \neq \emptyset$；

8. 测试集 $\mathcal{E}_{test} = \mathcal{E} \setminus \mathcal{E}_{train}$。

我们假设，在 $\mathcal{K}$ 中涉及 T 的公理只有以下形式： $a{:}T$ 或 $a{:}\neg T$。如果 a 是正例（即 $a \in \mathsf{I}_{\mathcal{E}^+}$），我们记作 $\mathcal{E}(a) = 1$；如果 a 是负例（即$a \in \mathsf{I}\varepsilon^-$），记作 $\mathcal{E}(a) = -1$；其他情况则记作 $\mathcal{E}(a) = 0$。

总体目标是学习分类函数 $h \in \mathcal{H}$，它是对 $\mathcal{E}_{train}$ 进行经验风险最小化（ERM）的结果，即

$$\bar{h} \quad = \quad \arg\min_{h \in \mathcal{H}} R(h, \mathcal{E}_{train}) \quad = \quad \frac{1}{|\mathcal{E}_{train}|} \sum_{a \in \mathsf{I}_{\mathcal{E}_{train}}} L(h(a), \mathcal{E}_{train}(a)) \ ,$$

其中，L 是一个损失函数，使得 $L(l, l)$ 度量假设的预测 l 与真实标签 l 之间的差异，而 $R(h, \mathcal{E}_{train})$ 是假设 h 在 $\mathcal{E}_{train}$ 上的风险，定义为损失函数在训练集 $\mathcal{E}_{train}$ 上的期望。

学习得到的分类器的有效性 $\bar{h}$ 随后通过确定 $R(\bar{h},\ \mathcal{E}_{test})$ 来评估 the test 集合 $\mathcal{E}_{test}$。

在我们的学习设定中，假设 $h \in \mathcal{H}$ 是一组形式如下的一般概念包含公理

$$\langle C_1 \sqsubseteq P_1, \alpha_1 \rangle \ , \dots \ , \langle C_h \sqsubseteq P_h, \alpha_h \rangle \tag{2}$$

$$@^+(P_1, \dots, P_h) \sqsubseteq P \tag{3}$$

$$\langle D_1 \sqsubseteq N_1, \beta_1 \rangle \ , \dots \ , \langle D_k \sqsubseteq N_k, \beta_k \rangle \tag{4}$$

$$@^-(N_1, \dots, N_k) \sqsubseteq N \tag{5}$$

$$@(P, N) \sqsubseteq T \tag{6}$$

其中，每个 P_i、P、N_j、N都是在 $\mathcal{K}$ 中未出现过的新原子概念名称， α_i、β_j 是相应一般概念包含公理的置信度，$@^+$、$@^-$、$@$ 是聚合算子，而每个 C_i、D_j 都是一个模糊 $\mathcal{EL}$**(D)** 概念表达式，其定义如下（v 是一个布尔值）

$$\begin{array}{lcl} C & \longrightarrow & \top \mid A \mid \exists r.C \mid \exists s.\mathbf{d} \mid C_1 \sqcap C_2 \\ \mathbf{d} & \rightarrow & ls(a,b) \mid rs(a,b) \mid tri(a,b,c) \mid trz(a,b,c,d) \mid \ =_v \ . \end{array}$$

非正式地说，*(i)* 每条 P_i “规则”都会说明个体为何应为正例；*(ii)* 然后，我们通过聚合算子 $@^+$ 聚合各种正例程度；*(iii)* 另一方面，每条 N_i “规则”都会说明个体为何不应为 正例；*(iv)* 然后，我们通过聚合算子 $@^-$ 聚合各种非正例程度。通常，$@^+$ 和 $@^-$ 都是最大值算子；最后，*(v)* 我们使用最后一条“规则”，通过 $@$ 运算符组合正例程度与非正例程度，以确定个体是否为 T 的实例（即 是否为正例或非正例）。$@$ 的一种简单选择如下，也是我们将采用的选择:

> ($\star$) if the degree p of being positive is greater than the degree of being non-positive n then p, else 0.

N现在，对于 $a \in \mathbf{I}_{\mathcal{K}}$， a 、 T 和 $\mathcal{K}$ 的分类预测值 $h(a)$定义为

$$h(a) = bed(\mathcal{K} \cup h, a{:}T) \ . \tag{7}$$

备注 2 请注意，如上所述，假设本质上是某个个体在一定程度上属于目标概念实例的充分条件。如果 $h(a) = 0$ ，那么我们称 a 不是 T 的正实例；而如果 $h(a) > 0$ ，那么 a 是 T 的正实例，其程度为 $h(a)$。因此，我们只区分 T 的正例和非正例。也就是说，负例与未标记样例无法区分。

请注意，即使 $\mathcal{K}$ 是清晰知识库，假设中形如 $\exists S.\mathbf{d}$ 的表达式可能出现模糊具体域，这也意味着 $h(a) \in \{0, 1\}$ 未必成立。聚合算子也可能产生类似的影响。

注释 3 显然，由于概念表达式中的合取以及存在限制的嵌套，这种语法所定义的假设集合可能是无限的。通过对生成过程施加进一步限制，例如规定合取项的最大数量以及允许的存在限制嵌套最大深度，可以使该集合变为有限集。

我们最后定义：假设 h 覆盖 （分别地， θ-覆盖） $\theta \in (0, 1]$ 个体$a \in \mathbf{I}_{\mathcal{K}}$ ，当且仅当 $h(a) > 0$ （分别地， $h(a) \geq \theta$），并用 $Cov(h)$（分别地， $Cov_\theta(h)$）表示被覆盖

（分别地， θ-被覆盖的）个体。此外，对于一个 GCI $C \sqsubseteq T$， $C \sqsubseteq T$相对于 $\mathcal{K}$和一组正个体 P的置信度 （也称为包含度），记作 $cf(C \sqsubseteq T, \mathcal{K}, P)$，定义为

$$cf(C \sqsubseteq T, \mathcal{K}, P) = \frac{|C|^P_{\mathcal{K}}}{|C|^{\mathsf{I}_{\mathcal{K}}}_{\mathcal{K}}}, \tag{8}$$

即 C 所覆盖的正个体占C所覆盖个体的比例。显然， $cf(C \sqsubseteq T, \mathcal{K}, P) \in [0, 1]$，且置信度越接近 1，$C \sqsubseteq T$就越“精确”，也就是说，它覆盖的非正例越少。此外， $C \sqsubseteq T$相对于 $\mathcal{K}$和个体集合 I 的支持度 记为 $supp(C \sqsubseteq T, \mathcal{K}, \mathsf{I})$，其定义为

$$supp(C \sqsubseteq T, \mathcal{K}, \mathsf{I}) = \frac{|C|^{\mathsf{I}}_{\mathcal{K}}}{|\mathsf{I}|} \tag{9}$$

3.2 学习方法的概念示意。

Bef在介绍学习算法之前，我们先通过概念示意来说明其原理，
rely参见图4。

首先，考虑所有个体、正个体、负个体和未标记个体的集合，分别记为集合 $\mathsf{I}_{\mathcal{K}}$，$\mathsf{I}_{\mathcal{E}^+}$，$\mathsf{I}_{\mathcal{E}^-}$ 和 $\mathsf{I}_{\mathcal{E}^u}$，如图4 (a) 所示。

在第一阶段，即 P 阶段，我们考虑整个训练集 $\mathcal{E}$，并尝试最大限度地覆盖正个体，同时尽量减少对负个体的覆盖。具体而言，假设我们已经学习到一个假设 h_P （规则集），其覆盖范围为$Cov_{\theta_P}(h_P)$，如图4 (b) 所示。这里，值 θ_P充当假设 h_P中所学习规则的置信度阈值。请注意， $Cov_{\theta_P}(h_P)$ 必须包含正个体，即 $Cov_{\theta_P}(h_P) \cap \mathsf{I}_{\mathcal{E}^+} \neq \emptyset$，但也可能包含负个体和未标记个体。我们将$TP = Cov_{\theta_P}(h_P)$ 中的个体称为 $\cap \mathsf{I}_{\mathcal{E}^+}$ 真正例，而将$FP = Cov_{\theta_P}(h_P)$ 中的个体称为 $\setminus \mathsf{I}_{\mathcal{E}^+}$ 假阳性。即 假阳性是指被 h_P 错误地分类为目标类 T实例的个体，但实际上并非如此（它可能是未标记个体或负例）。此阶段以一组形式为 (2) –(3) 的规则结束。

现在，在下一阶段，即 N 阶段，为了提高分类器的效果，我们希望尽可能多地从 FP 中移除假阳性，同时尽可能避免移除 TP 中的任何真正例。为此，我们设置一个新的学习问题，其中新的目标类为 FP，负个体是 TP 中的个体，正例则是 FP 中的个体。当然，只有在 $FP \neq \emptyset$ 时才会应用 N 阶段。N 阶段的设置如图4(c)所示。具体而言，假设我们现在已经学习得到一个假设 h_N，其覆盖范围为 $Cov_{\theta_N}(h_N)$，如图4(d)所示。请注意，我们还可以设置另一个参数θ_N，作为假设 h_N 中已学习规则的置信度阈值。该阶段最终得到一个由形式为 (4) –(5) 的规则组成的规则集。

因此，通常在两个阶段结束时，情况可能如图4(e)所示。然而，实际应用中，人们可能希望规定，初始正个体中没有任何个体被 h_N 覆盖，因此 TP 中也不会有任何真正例被 h_N 移除。

最后，我们通过（$\star$）汇总 P 规则和 N 规则。后一步以形如（6）的规则结束。在这一两阶段过程中，我们的目标是捕获目标类的大多数正个体，同时尽量少包含负个体和未标记个体（假阳性）。

3.3 PN-OWL 学习算法

现在，我们介绍两阶段学习算法 PN-OWL，它在上一节中已有概念性说明。本质上，在 P 阶段（分别为 N 阶段），我们的算法会调用

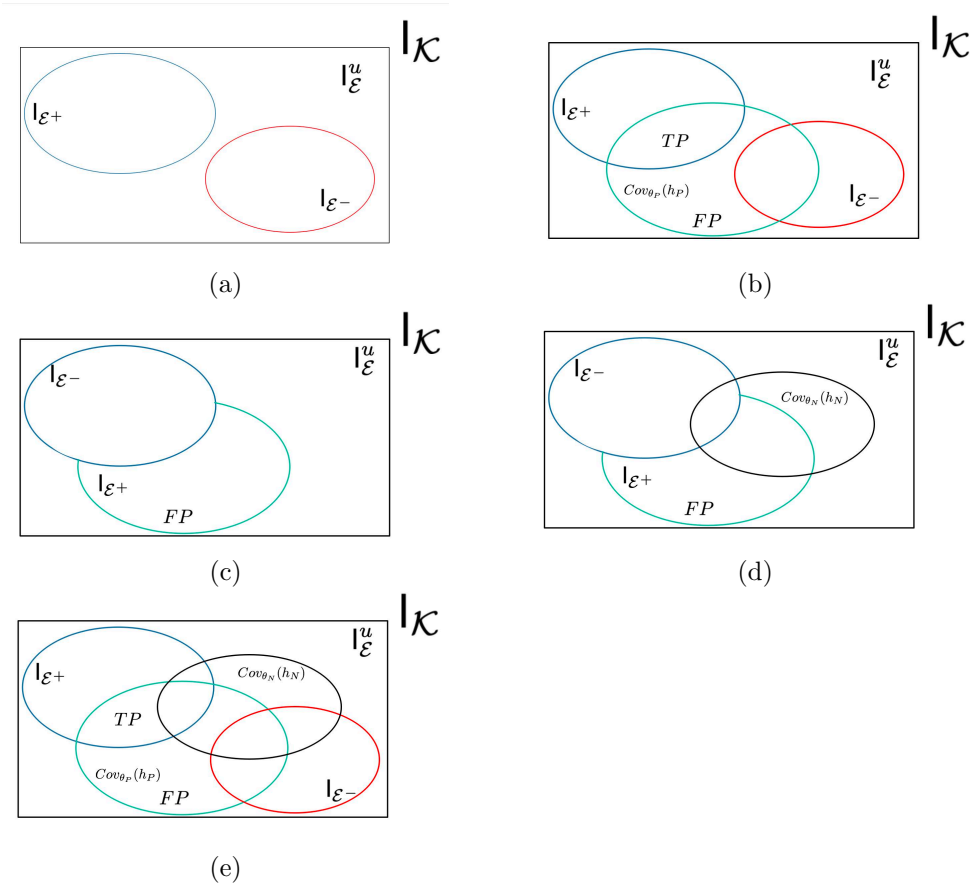

图4： PN-OWL 的工作原理。(a) 原始训练集；(b) P 阶段后学习得到的假设 h_P 的覆盖度 $Cov_{\theta_P}(h_P)$；(c) N 阶段的起始数据集：新的目标类是 P 阶段的假阳性 FP，而负个体是初始正例；(d) N 阶段后学习得到的假设 h_N 的覆盖度 $Cov_{\theta_N}(h_N)$；(e) 最终情形。

一个称为阶段学习器的学习器，它会生成一组 h_P（分别为 h_N）模糊 $\mathcal{EL}$（**D**）候选一般概念包含公理，其形式分别为

$$h_P = \{\langle C_1 \sqsubseteq T, \alpha_1\rangle, \ldots, \langle C_h \sqsubseteq T, \alpha_h\rangle\} \quad (10)$$

$$h_N = \{\langle D_1 \sqsubseteq FP, \beta_1\rangle, \ldots, \langle D_k \sqsubseteq FP, \beta_k\rangle\} \quad (11)$$

称为阶段假设。下文中，我们用 p_i 表示模糊 GCI $\langle C_i \sqsubseteq T, \alpha_i\rangle$，用 n_j 表示模糊 GCI $\langle D_j \sqsubseteq FP, \beta_j\rangle$。随后，将使用最大聚合算子聚合 h_P（分别为 h_N）中的规则。

随后，使用聚合算子（$\star$）将各阶段假设合并为目标类 T 的最终假设。

作为阶段学习器，我们将使用修改版的模糊 FOIL-$\mathcal{DL}$ [50, 51, 53] 学习器，具体内容将在第3.4节中介绍。

然后， PN-OWL 算法如算法 1 所示。请注意，P 阶段是步骤 s

1-5，而 N 阶段是步骤 15-19；在步骤 19 中，我们调用阶段学习器，尝试尽可能多地覆盖假阳性。其余步骤则根据公式 (2)-(6) 构建最终分类器集成。

最终，对于个体 $a \in \mathrm{I}_{\mathcal{K}}$，个体 a 的分类预测值 PN-OWL 为 $h(a)$，其中 h 是 PN-OWL返回的假设。此外，如果 $h(a) > 0$，则称PN-OWL 将 a 分类为目标类 T 的实例。

算法 1 PN-OWL

输入： 知识库 $\mathcal{K}$，训练集 $\mathcal{E}$，目标概念名称 T，置信度阈值 θ_P，$\theta_N \in [0,1]$，非正覆盖率百分比 η_P，$\eta_N \in [0,1]$

输出： 假设 h 由公式 (2)-(6) 得出。

1: // P 阶段
2: $Pos \leftarrow$ 我$_{\mathcal{E}^+}$;
3: $Neg \leftarrow$ Iε^-;
4: $U \leftarrow \mathrm{I}_{\mathcal{K}} \setminus (Pos \cup Neg)$;
5: $h_P \leftarrow$ FuzzyStageLearner($\mathcal{K}$, T, Pos, Neg, U, θ_P, η_P); ▷ P 阶段假设 h_P，即公理集合 $\langle C_i \sqsubseteq T, \alpha_i \rangle$
6: **如果** $h_P = \emptyset$ **，则返回** $\emptyset$; ▷ 未学到任何内容，退出
7: $Cov \leftarrow Cov_{\theta_P}(h_P)$; ▷ P 阶段覆盖率
8: $TP \leftarrow Cov_{\theta_P}(h_P) \cap \mathrm{I}_{\mathcal{E}^+}$; ▷ 真正例
9: $FP \leftarrow Cov_{\theta_P}(h_P) \setminus \mathrm{I}_{\mathcal{E}^+}$; ▷ 假阳性
10: // 开始构建分类器 h
11: $h \leftarrow \{\langle C_i \sqsubseteq P_i, \alpha_i \rangle, \mid \langle C_i \sqsubseteq T, \alpha_i \rangle \in h_P, P_i$新建 $\}$; ▷ 根据公式 2
12: **如果** $FP = \emptyset$ **，那么** ▷ 没有 N 阶段，退出并返回汇总后的 h_P
13: $h \leftarrow h \cup \{@^+(P_1, \ldots, P_h) \sqsubseteq T\}$; ▷ 公式 3 中无需新的 P
14: **返回** h;
15: // N 阶段
16: $Pos \leftarrow FP$;
17: $Neg \leftarrow$ 我$_{\mathcal{E}^+}$;
18: $U \leftarrow \mathrm{I}_{\mathcal{K}} \setminus (Pos \cup Neg)$;
19: $h_N \leftarrow$ FuzzyStageLearner($\mathcal{K}$, FP, Pos, Neg, U, θ_N, η_N); ▷N 阶段假设 h_N，即公理集合 $\langle D_j \sqsubseteq FP, \beta_j \rangle$
20: // 构建最终分类器集成 h
21: **如果** $h_N = \emptyset$ **则** ▷ N 阶段无需学习，返回聚合结果 h_P
22: $h \leftarrow h \cup \{@^+(P_1, \ldots, P_h) \sqsubseteq T\}$; ▷ 根据公式 3，无需新的 P
23: **返回** h;
24: $h \leftarrow h \cup \{@^+(P_1, \ldots, P_h) \sqsubseteq P \mid P$new $\}$; ▷ 根据公式 3
25: $h \leftarrow h \cup \{\langle D_j \sqsubseteq N_j, \beta_j \rangle, \mid \langle D_j \sqsubseteq FP, \beta_j \rangle \in h_N, N_j$新建 $\}$; ▷ 根据公式 4
26: $h \leftarrow h \cup \{@^-(N_1, \ldots, N_k) \sqsubseteq N \mid N$new $\}$; ▷ 根据公式 5
27: $h \leftarrow h \cup \{@(P, N) \sqsubseteq T\}$; ▷ 根据公式 6
28: **返回** h;

3.4 阶段学习器 pnFoil-$\mathcal{DL}$

作为阶段学习器，我们将使用模糊 Foil-$\mathcal{DL}$ [20, 50, 51, 53]，不过会对其进行修改，以适应我们的特定设定（见算法 2）；我们将其称为 pnFoil-$\mathcal{DL}$。也就是说，PN-OWL 算法第 5 行和第 19 行中的过程调用 FuzzyStageLearner 实际上都是对 pnFoil-$\mathcal{DL}$ 的调用。

本质上，pnFoil-$\mathcal{DL}$ 会持续归纳 GCI，直到覆盖尽可能多的正例，或者无法再学习到任何新内容。归纳出一条公理时（见算法 2 的步骤 4），尚待覆盖的正例会随之更新（步骤 10 和 11）。

为了归纳一条公理（步骤 4），会调用 Learn-One-Axiom （见算法 3），其一般流程如下:

1. 从概念 $\top$ 开始；2. 应用细化算子，寻找更具体的模糊 $\mathcal{EL}$（**D**）概念描述候选项；3. 利用评分函数选出最佳候选项；4. 重新应用细化算子，直到找到合适的候选项；5. 重复整个过程，直到正例达到满意的覆盖度。

算法 2 pnFoil-$\mathcal{DL}$输入：知识库 $\mathcal{K}$、目标概念名称 T、正例集合 P（以及负例和未标记示例集合 N 和 U）、置信度阈值 $\theta \in [0,1]$、非正例覆盖率 $\eta \in [0,1]$输出：一个假设，即一组模糊 $\mathcal{EL}$(D) GCI 1： $h \leftarrow \emptyset$， $Pos \leftarrow P$， $\phi \leftarrow \top \sqsubseteq T$； 2：// 循环直到没有改进 3：当（ $Pos \neq \emptyset$ ）且（$\phi \neq$ 非空）时执行 4：$\phi \leftarrow$ Learn-One-Axiom（$\mathcal{K}$， T， Pos， P， N， U， θ， η）； ▷ 学习一条形如 $C \sqsubseteq T$ 的模糊 $\mathcal{EL}$(D) GCI 5：如果 $\phi \in h$ ，那么▷ 公理已学习 6：$\phi \leftarrow$ 为空； 7：如果 $\phi \neq$不为空，那么 8：$\delta \leftarrow cf$（ϕ， $\mathcal{K}$， P）；▷计算 ϕ 的置信度 9：$h \leftarrow h \cup \{\langle \phi, \delta \rangle\}$； ▷ 更新假设 10：$Pos_\phi \leftarrow Pos \cap Cov$（$\langle \phi, \delta \rangle$）；▷ 被 $\langle \phi, \delta \rangle$ 覆盖的正例 11：$Pos \leftarrow Pos \setminus Pos_\phi$； ▷ 更新仍待覆盖的正例 12：返回 h；

现在我们详细说明 Learn-One-Axiom (算法 3) 的步骤。

计算模糊数据类型。 对于数值数据类型 s，我们将值 $V_s = \{v \mid \mathcal{K} \models a{:}\exists s.=_v\}$ 划分为有限个模糊集（3、5 个或 7 个），采用等宽三角划分，这与 [50, 53, 87] 相同（参见例如，图 3）。此外，我们也考虑在 V_s上使用模糊 C 均值聚类算法，其中模糊隶属函数是围绕簇 [20, 50, 53, 87] 质心构建的三角函数。

细化算子。 我们采用的细化算子与 [20, 50, 51, 57, 87] 中的基本相同。具体而言，它以概念 C 为输入，并生成更具体的新概念描述候选项 D （即， $\mathcal{K} \models D \sqsubseteq C$）。为求完整起见，我们在此回顾该细化算子。设 $\mathcal{K}$ 为一个知识库，$\mathbf{A}_\mathcal{K}$ 为 $\mathcal{K}$ 中所有原子概念的集合，$\mathbf{R}_\mathcal{K}$ 为 $\mathcal{K}$ 中所有对象属性的集合，$\mathbf{S}_\mathcal{K}$ 为 $\mathcal{K}$ 中所有数值数据类型属性的集合，$\mathbf{B}_\mathcal{K}$ 为 $\mathcal{K}$ 中所有布尔数据类型属性的集合， $\mathcal{D}$ 为一组（模糊）数据类型。细化算子 ρ 如表2所示。

表2：向下细化算子。

$$\rho(C) = \begin{cases} \mathbf{A}_{\mathcal{K}} \cup \{\exists r.\top \mid r \in \mathbf{R}_{\mathcal{K}}\} \cup \{\exists s.d \mid s \in \mathbf{S}_{\mathcal{K}}, d \in \mathcal{D}\} \cup & \\ \qquad \{\exists s. =_b, \mid s \in \mathbf{B}_{\mathcal{K}}, b \in \{\mathbf{true}, \mathbf{false}\}\} & \text{if} \quad C = \top \\ \{A' \mid A' \in \mathbf{A}_{\mathcal{K}}, \mathcal{K} \models A' \sqsubseteq A\} \cup & \\ \qquad \{A \sqcap A'' \mid A'' \in \rho(\top)\} & \text{if} \quad C = A \\ \{\exists r.D' \mid D' \in \rho(D)\} \cup \{(\exists r.D) \sqcap D'' \mid D'' \in \rho(\top)\} & \text{if} \quad C = \exists r.D, r \in \mathbf{R}_{\mathcal{K}} \\ \{(\exists s.d) \sqcap D \mid D \in \rho(\top)\} & \text{if} \quad C = \exists s.d, s \in \mathbf{S}_{\mathcal{K}}, d \in \mathcal{D} \\ \{(\exists s. =_b) \sqcap D \mid D \in \rho(\top)\} & \text{if} \quad C = \exists s. =_b, s \in \mathbf{B}_{\mathcal{K}}, \\ & \qquad b \in \{\mathbf{true}, \mathbf{false}\} \\ \{C_1 \sqcap ... \sqcap C_i' \sqcap ... \sqcap C_n \mid i = 1, ..., n, C_i' \in \rho(C_i)\} & \text{if} \quad C = C_1 \sqcap ... \sqcap C_n \end{cases}$$

评分函数。 我们使用的评分函数用于为每个候选假设赋分，其本质上是一个增益 函数，与 [20, 50, 51, 57, 87]，中采用的函数类似，并实现了一种信息论准则，用于在每个细化步骤中选择最佳候选项。具体而言，给定前一步选定的、形式为 $C \sqsubseteq T$ 的模糊 $\mathcal{EL}$（$\mathbf{D}$）GCI ϕ，一个知识库 $\mathcal{K}$、一组尚未覆盖的正例 Pos，以及一个形式为 $C' \sqsubseteq T$ 的候选模糊 $\mathcal{EL}$（$\mathbf{D}$）GCI ϕ'，那么

$$gain(\phi', \phi, \mathcal{K}, Pos) = p * (log_2(cf(\phi', \mathcal{K}, Pos)) - log_2(cf(\phi, \mathcal{K}, Pos))) \ , \tag{12}$$

wh其中， $p = |C' \sqcap C|_{\mathcal{K}}^{Pos}$是 Pos 中被 ϕ 覆盖的正例的模糊基数，这些正例 e仍然 被 ϕ'覆盖。

请注意，在公式 12 中，置信度是相对于尚未覆盖的正例（Pos）计算的。这样，Learn-One-Axiom 会在一定程度上被引导去寻找 pnFoil-$\mathcal{DL}$迄今尚未覆盖的正例。还要注意，如果置信度提高，增益就是正值。

停止准则。 Learn-One-Axiom 会在置信度高于给定阈值 $\theta \in [0, 1]$ 且非正例覆盖率低于 $\eta \in [0, 1]$时停止；如果无法再学习到任何 GCI，也会停止。

Learn-One-Axiom 算法。 该 Learn-One-Axiom 算法与算法 3 中定义的一样：步骤 1 - 3 是简单的初始化步骤。请注意，根据备注 2， NP 是非正例；备注 2 指出，我们只区分正例和非正例（另见 步骤 18，其中使用了非正例覆盖率）。步骤 5-21 是主循环，满足停止准则时即可退出。在步骤 8 中，我们确定所有新的细化，然后在步骤 10-15 中对其评分，以确定最大增益。算法结束时，退出主循环后，返回找到的最佳 GCI（步骤 22）。

备注 4 *pn*Foil-$\mathcal{DL}$也允许使用回溯机制（步骤 *19*）。为便于说明，我们省略了这一机制。该机制与 *[87,* 算法 *3]* 中描述的 *p*Foil-$\mathcal{DL}$*-learnOneAxiom* 相同。具体而言，系统维护一个按置信度从高到低排列的顶层细化栈；如果没有取得改进，便从中弹出下一个最佳细化（前提是栈非空）。C_{best} 将成为弹出的细化。

算法 3 LEARN-ONE-AXIOM

输入： 知识库 $\mathcal{K}$、目标概念名称 T、尚待覆盖的正例集合 Pos、训练集
集合 P，N，U分别为正例、负例和未标记样例，置信度阈值
$\theta \in [0,1]$，非正例覆盖率 $\eta \in [0,1]$
输出： 形式为 $C \sqsubseteq T$ 的模糊 $\mathcal{EL}$（**D**）GCI

$NP \leftarrow N \cup U$; ▷ 注：NP 为非正例
$C \leftarrow \top$; ▷ 从 $\top$ 开始
$\phi \leftarrow C \sqsubseteq T$;
//循环直到不再有改进
当 $C \neq$ **为 null 时执行**
    $C_{best} \leftarrow C$;
    $maxgain \leftarrow 0$;
    $\mathcal{C} \leftarrow \rho(C)$; ▷ 计算 C 的所有细化
    // 计算各细化的得分并选出最佳项
    对所有 $C' \in \mathcal{C}$ **执行**
        $\phi' \leftarrow C' \sqsubseteq T$;
        $gain \leftarrow gain(\phi', \phi, \mathcal{K}, Pos)$;
        如果$(gain > maxgain)$ **则**
            $maxgain \leftarrow gain$;
            $C_{best} \leftarrow C'$;
    如果 $C_{best} = C$ **则** ▷ 没有改进
        //如果置信度高于阈值，或非正例覆盖度低于阈值，则停止
        如果 $(cf(C_{best} \sqsubseteq T, \mathcal{K}, P) \geq \theta)$ **且** $supp(C_{best} \sqsubseteq T, \mathcal{K}, NP) \leq \eta)$ **则中断**;
        // 在此处处理回溯（如果有预期）
    $C \leftarrow C_{best}$;
    $\phi \leftarrow C \sqsubseteq T$;
返回 ϕ;

4 评估

我们已在 *FuzzyDL-Learner*[5] 系统中实现该算法，并在一组（清晰的）OWL本体上对其进行了评估。

数据集。我们选取了来自不同领域的若干个 OWL 本体，如表 3 所示。表中列出了本体所涉及的描述逻辑、本体中的概念/类名称、对象属性、数据类型属性和个体的数量。对于每个本体 $\mathcal{K}$，我们还列出了 $|\mathcal{E}^+|$ 正例的数量。其余示例均为非正例，我们设定$\mathcal{E}^- = \mathcal{E}^+ = \mathrm{I}_{\mathcal{K}} \backslash \mathrm{I}_{\mathcal{E}^+}$。Iris、葡萄酒、¯葡萄酒质量和酵母本体均基于知名的 加州大学欧文分校机器学习知识库 （UCIMLR） [27] 构建，并按照 [20]中描述的流程，将该知识库提供的 CSV 格式数据转换为 OWL 本体。在葡萄酒质量本体中，我们移除了质量属性，因为正例（优质红葡萄酒）是“质量”大于或等于 7 的葡萄酒。

除恶意软件本体外，所有其他本体均属于著名的 SML-Bench 数据集 [91]。[6]
Th恶意软件本体已在 [88, 89]中描述。

为求完整，附录 A 简要介绍了这些本体的主题。

[5]数据与实现 `http://www.umbertostraccia.it/cs/software/FuzzyDL-Learner/`。[6]另请参阅，`https://github.com/SmartDataAnalytics/SML-Bench`

表 3：评估中各本体的事实。

本体	DL	类	对象属性	数据属性	ind.	目标 T	pos
NTN	$\mathcal{SHOIN}(\mathcal{D})$	51	29	9	723	ToLearn_Woman	46
Lymphography	$\mathcal{ALC}$	50	0	0	148	ToLearn	81
Mammographic	$\mathcal{ALC}(\mathcal{D})$	20	3	2	975	ToLearn	445
Malware	$\mathcal{ALH}(\mathcal{D})$	192	6	10	5669	malware	500
Iris	$\mathcal{ALEHF}(\mathcal{D})$	4	0	5	150	Iris-versicolor Iris-virginica	50 50
Wine	$\mathcal{ALEHF}(\mathcal{D})$	3	0	13	178	1 2 3	59 71 48
Wine Quality	$\mathcal{ALEHF}(\mathcal{D})$	7	0	11	6497	GoodRedWine	217
Yeast	$\mathcal{ALEHF}(\mathcal{D})$	11	0	8	1462	CYT	444

备注 5 虽然在数值型数据类型属性上评估基于本体的学习算法并不常见，[7] 但我们认为这样做很有意义，因为我们的算法的一个重要组成部分是利用模糊具体数据类型属性来提升分类决策过程的可理解性。

备注 6 我们将把研究从训练数据中学习阈值 $0 \leq \tau_p \leq 1$ 的方法留待未来工作，例如使 h 预测个体 a 为正例，当且仅当 $h(a) > \tau_p$。不过，在本文中，我们始终会有 $\tau_p = 0$。

更一般地说，与我们目前的做法不同，如果我们希望区分负例和未标记样本，可以学习一个用于负例的分类器 h^-，然后定义一种决策方法，根据预测值 $h(a)$（相应地，$h^-(a)$）判断个体 a 是正例（相应地，是负例），其中 a 是正（相应地，是负）例。也就是说，根据这对 $\langle h(a)\ h^-(a)\rangle$，可以定义决策准则来判断 a 是正例还是负例；如果没有足够证据判定其属于其中一类，也可以将预测结果保留为未知。

度量指标。我们考虑了以下有效性度量指标（另见 [87, 20]），为求完整起见，在此重述。具体而言，设有一个已学习的分类器 h，并假设我们已将其添加到知识库 $\mathcal{K}$ 中。在我们的设定中，始终满足以下条件：如果个体 a 的分类器预测值 $h(a)$ 非零，那么学习器会将 a 分类为 T 的一个实例，即 h 预测 a 为正例，当且仅当 $h(a) > 0$。

根据上述内容，由于所有个体要么是正例，要么是非正例，我们将考虑以下度量指标，它们都基于清晰基数（另见公式 1）。

真正例： 记作 TP，定义为 T 的实例中属于正例的数量

$$TP = \lceil T \rceil_{\mathcal{K}}^{\mathsf{I}\,\mathcal{E}^+} \tag{13}$$

Fal假正例： 记作 FP，定义为 T 的实例中不属于正例的数量

$$FP = \lceil T \rceil_{\mathcal{K}}^{\mathsf{I}\,\overline{\mathcal{E}^+}} \tag{14}$$

精确率/置信度： 记作 P，定义为 h 的覆盖样本中真正例所占的比例

$$P = \frac{TP}{\lceil T \rceil_{\mathcal{K}}^{\mathsf{I}\,\mathcal{E}}} \tag{15}$$

[7] 据我们所知，目前尚无针对这些数据的基于本体的方法评估 a 集合。

召回率：记为 R，定义为真正例占所有正例的比例

$$R = \frac{TP}{|\mathrm{I}_{\mathcal{E}^+}|} \ , \tag{16}$$

$F1$**-分数：**记为 $F1$，定义为

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \ . \tag{17}$$

对于每种参数配置，我们采用分层 k折交叉验证方案[8]（具体而言， $k = 5$），以确定上述性能度量的宏平均值。在所有测试中，$\mathrm{I}_{\mathcal{E}}$ =$\mathrm{I}_{\mathcal{K}}$，并且当然每一折中至少有一个正例。在每一折的训练阶段，我们从本体中移除所有涉及测试样例的断言，因此将训练阶段限制为仅使用训练样例。

最佳运行所用的所有配置参数均可从可下载的数据中获取，此处不再列出。表 4 列出了我们算法中使用的一些重要参数。

表 4： PN-OWL 算法的一些重要参数。

θ_P	P 阶段正规则的置信度阈值
θ_N	N 阶段负规则的置信度阈值
η_P	P 阶段正规则的非正例覆盖率阈值
η_N	N 阶段负规则的非正例覆盖率阈值
c_P	P 阶段正规则的合取项最大数量
c_N	P 阶段负规则的合取项最大数量
d_P	P 阶段正规则的最大角色深度
d_N	P 阶段负规则的最大角色深度

典型的参数设置如下，但可能会因本体而异，也可能需要搜索以找到最优设置。

P 阶段。 $c_P = 5,\ d_P = 1,\ \theta_P = 0.1,\ \eta_P = 1.0$

N-stage. $c_N = 10, d_N = 1, \theta_N = 0.3, \eta_N = 0.2$

下面简要说明这些参数。在 P 阶段，我们希望提高召回率，即覆盖的正例占所有正例的比例。为此，我们选择较低的正规则置信度阈值 θ_P 和较高的非正例覆盖率阈值 η_P。而在 N 阶段，我们希望更精确地去除假阳性，以避免移除 P 阶段的真正例。因此，我们提高置信度阈值 θ_N，降低非正例覆盖率阈值 η_N，并增加最大合取项数c_N。最大角色深度则根据对本体的检查事先手动确定。

对于 $@^+$，$@^-$（分别为 @），我们使用 max（分别为 $(\star)$）；对于概念合取 $\sqcap$（分别为 GCI 运算符 $\sqsubseteq$），我们使用三角范数 min（分别为 -卢卡西维茨蕴涵）。这些也完全可以作为另一组待优化的参数。然而，参数空间已经相当大，

[8]此处的分层是指每一折中目标类的正例实例和非正例实例大致保持相同比例。

因此我们按照指定固定了这些逻辑运算符。[9] 关于其他参数设置，我们还调整了模糊集的数量（3，5 或 7）。对于 C 均值，我们将超参数固定为默认值 $m = 2$，将阈值设为 $\epsilon = 0.05$，并将最大迭代次数设为 100。

作为基线，我们考虑 Fuzzy FOIL-$\mathcal{DL}$ [50, 51, 53, 20]，其最佳参数设置如 [20] 所述。基本上，Fuzzy FOIL-$\mathcal{DL}$ 与 PN-OWL 相同，只是在 P 阶段后停止，因此相当于 PN-OWL，其中负规则集 h_N 按定义为空（参见 PN-OWL 算法的第 21–23 行）。这使我们能够评估 N 阶段所带来的额外价值（如果有）。

结果见表5。对于 UCIMLR 数据集，如果存在多个目标，
the 则考虑各项度量指标的平均值。

示例 4.1 下面给出将 PN-OWL 应用于乳腺X线摄影 本体后学习到的规则示例（采用模糊 *OWL* 语法）。第一条规则是在 *P* 阶段学习到的规则之一，第二条则是在 *N* 阶段学习到的规则之一。在后一种情况下，FALSEPToLearn 表示由 *P* 阶段学习到的规则所覆盖的假阳性类。规则旁的数字表示其置信度/精确率。我们还报告了通过模糊 *C* 均值获得的一些模糊集的定义。

```
(implies (and (some hasDensity low)
              (some hasShape irregular)
              (some hasAge hasAge_veryHigh)
              (some hasBiRads hasBiRads_high))
   ToLearn 0.965068)

(implies (and (some hasDensity low)
              (hasMargin some microlobulated)
              (hasShape some oval)
              (hasBiRads some hasBiRads_medium))
   FALSEP_ToLearn 0.75)

(define-fuzzy-concept hasBiRads_medium    triangular(1,6,2.780,3.997,5.022))
(define-fuzzy-concept hasBiRads_high      right-shoulder(1,6,3.997,5.022))
(define-fuzzy-concept hasAge_veryHigh     right-shoulder(1,6,62.793,71.882))
```

[9] 使用固定参数 *e.g.* 在 malware本体上运行，可能就需要最多 4 天的计算时间
time.

表5：结果表。度量指标为测试集上5折的宏平均值。

数据集	算法	精确率	召回率	F1	改进百分比
NTN	模糊 DL-FOIL	0.661	0.513	0.548	**80.47%**
	PN-OWL	**1.000**	**0.980**	**0.989**	
淋巴造影	模糊 DL-FOIL	**0.861**	**0.851**	**0.855**	**-2.57%**
	PN-OWL	0.836	0.841	0.833	
乳腺X线摄影	模糊 DL-FOIL	0.737	0.692	0.710	**11.27%**
	PN-OWL	**0.746**	**0.831**	**0.790**	
恶意软件	模糊 DL-FOIL	0.623	**0.830**	0.704	**5.06%**
	PN-OWL	**0.701**	0.818	**0.740**	
Iris	模糊 DL-FOIL	0.886	0.910	0.890	**4.16%**
	PN-OWL	**0.949**	0.910	**0.927**	
Wine	模糊 DL-FOIL	0.884	**0.971**	0.895	**0.98%**
	PN-OWL	**0.933**	0.904	**0.914**	
葡萄酒质量	模糊 DL-FOIL	0.227	**0.917**	0.363	**27.93%**
	PN-OWL	**0.365**	0.659	**0.464**	
酵母	模糊 DL-FOIL	0.427	0.746	0.540	**4.37%**
	PN-OWL	**0.432**	**0.815**	**0.564**	

讨论。 表5的最后一列报告了 PN-OWL 相对于度量 $F1$ （见公式17）的改进情况，比较基准为我们的模糊 FOIL-$\mathcal{DL}$。总体而言， PN-OWL 优于模糊 FOIL-$\mathcal{DL}$ （淋巴造影除外），且在某些情况下提升尤为显著，例如 NTN、乳腺X线摄影和葡萄酒质量。

总体而言，对于 PN-OWL，与 FOIL-$\mathcal{DL}$ 相比，我们在精确率和召回率之间找到了更好的平衡。特别是，我们提高了精确率，这证实了我们的猜想：N 阶段确实能够去除假阳性。

关于淋巴造影，我们未能复现 Fuzzy FOIL-$\mathcal{DL}$ in [20],的结果：现在得到的 F1 指标为 0.805，而非 0.855。差异在于少数被错误分类的示例。我们还注意到，在这种情况下，PN-OWL 在训练阶段达到了 $F1 = 1.0$ ，这可能意味着存在过拟合问题。

最后但同样重要的是，我们要指出， PN-OWL （Fuzzy FOIL-$\mathcal{DL}$也一样）在葡萄酒质量和酵母数据集上的表现确实仍不理想，这将是后续进一步研究的主题。

从 PN-OWL 中获得的总体经验是，N 阶段确实有助于提升分类过程的效果，前提是能够在精确率和召回率之间找到适当的平衡。不幸的是，在 PN-OWL 的参数空间中寻找最优值相当耗时，暴力搜索方法很可能并不可行（至少以我们现有的计算资源而言不可行）。事实上，我们每次只运行一次，并通过分析结果来尝试判断是否以及如何调整表 4 中的一些参数，以提高召回率和/或精确率。另一方面，优化 FOIL-$\mathcal{DL}$ 要容易得多，因为它的参数数量只有 PN-OWL的一半。

5 相关工作

描述逻辑中的概念包含公理学习本质上受统计关系学习的启发；在统计关系学习中，分类规则是（可能带权重的）霍恩子句理论（参见例如 [69, 70]），而迄今为止，描述逻辑领域也已提出了各种方法（参见例如 [54, 24, 71]）。其总体思路是探索潜在概念描述的搜索空间，利用所谓的细化算子（参见例如 [7, 22, 45, 46, 47,48, 49]）覆盖现有的训练样例。随后，目标是学习一种底层描述逻辑语言中的概念描述，使其覆盖（可能）所有给定的正例，并且（可能）不覆盖任何给定的负例。模糊情形（参见 [50, 53, 87, 20]）则是自然的扩展，它转而依托模糊描述逻辑 [10, 86]和模糊归纳逻辑编程（参见例如 [82]）。

如前所述，我们的两阶段算法在概念上受到 PN-rule [2, 3, 40,41] 的启发。该方法由 P 阶段和 N 阶段组成：在 P 阶段中，学习正规则（称为 P 规则），以尽可能多地覆盖目标类的实例；在 N 阶段中，学习负规则（称为 N 规则），以剔除 P 阶段覆盖的大多数非正例。随后利用这两组规则构建决策方法，以判定某个对象是否为目标类的实例 [2, 3, 40, 41]。值得注意的是，该方法与其他所有方法的区别在于其第二阶段。 PN-OWL与 PN-rule 的主要区别如下：*(i)* PN-rule 仅处理表格数据，即数据由属性值对组成（A, v），而我们研究的是 OWL本体。[10]；PN-rules 的形式为 $cond \to T$，其中条件 $cond$ 的形式为 $(A \in [l, h])$ 或 $(A \notin [l, h])$（适用于连续属性 A）。[11], 而我们的规则体包含条件的合取，每个条件都可以是模糊的，也可以是类名或属性限制（属性还可以嵌套）；以及*(iii)* PN-rule 使用的规则评分和组合策略与我们在 PN-OWL中采用的完全不同。后者可以用 Fuzzy OWL 2 [12, 13], 表示，而我们推测前者无法表示，因此没有采用这种方案，因为模糊描述逻辑推理机无法对这类规则进行推理。

其他密切相关的研究包括 [30, 28, 36, 35, 50, 53, 87]。事实上， [30, 28, 36, 78]可视为针对描述逻辑情形对著名的 FOIL 算法所作的改编；而本质上源自 [51, 52, 55, 56, 57, 58]，的 [50, 53] 则提出了类似 模糊 FOIL 的算法，并受到诸如 [26, 82, 84] 等模糊归纳逻辑编程变体的启发。[12] 值得注意的是， [50, 56] 采用了较弱的假设表示语言 DL-Lite [5], ，而本文则依赖于对模糊 $\mathcal{EL}$(D) 包含公理的聚合。 $\mathcal{EL}$也曾在 [87], 中得到研究；不过，该研究与 [50, 53]不同，因为它考虑了对模糊概念描述候选项进行（模糊）概率集成评估。[13] 最后重申，我们认为，模糊$\mathcal{EL}$(D) 概念表达式很有吸引力，因为它们可以直接转换为自然语言，从而有助于提升归纳分类器的可解释性。

文献 [35] 探讨了离散提升，并展示了如何利用某种随机向下细化算子，从传统学习器中导出一个弱学习器（称为 wDLF）；该算子至少覆盖一个正例，并产生一个由阈值确定的最小分数。与这项工作相关的是 [20] ，它处理假设语言中的模糊性，并采用实值型 AdaBoost 变体；不同于前者，它使用类似下降的梯度算法来搜索最佳备选方案。值得注意的是，这种方法也不同于“模糊”规则学习的 AdaBoost 变体，例如 [25, 66, 68, 81, 92] ： 在选择当前最佳备选方案之前，弱学习器必须预先生成整个规则搜索空间。这种

[10]表格数据可以轻松映射到 OWL本体中，如 [20]所示。[11]如果 A 是类别型，那么显然 $cond$ 的形式要么是 $A = v$ ，要么是 $A \neq v$。[12]关于模糊规则学习方法的概述，请参见 *e.g.* [23]。[13]此外，据我们所知，评估中并未涉及具体数据类型。

方法在 OWL 场景下基本上不可行，因为搜索空间过于庞大。

在 [39] 中，介绍了一种方法，该方法通过与 *fuzzyDL* 推理机 [15]交互，从 FuzzyOWL 2 本体中学习模糊 OWL DL 概念等价公理。候选概念表达式由底层 DL-Learner [44, 18, 19] 系统提供。不过，迄今为止，该方法仅在一个玩具本体上进行了测试。此外，我们还提及 [42]，它基于将模糊 -卢卡西维茨 $\mathcal{ALC}$ DL 构造转换为模糊 逻辑程序设计 （模糊 LP）的临时方案，并使用传统的归纳逻辑程序设计方法来学习规则。遗憾的是，该方法并不可靠，因为已有研究表明，从模糊描述逻辑到 LP 的映射并不完备 [64]，而 -卢卡西维茨 $\mathcal{ALC}$ 中的蕴涵问题是不可判定的 [21]。更准确地说，在无限值 -卢卡西维茨语义 [17]下，$\mathcal{EL}$ 就已经是不可判定的。[14]

虽然我们的目标并非全面综述本体相关的学习文献，但我们仍在此回顾一下：除了本文介绍的方法之外，还有其他方法，不过它们与本文的关联仅在于同样在描述逻辑的背景下进行概念描述归纳。例如，例如，一系列研究 [32, 33, 75, 74, 76, 72, 80, 77, 79] 受到决策树/随机森林的启发； [9, 29, 31, 34]采用核方法 来归纳概念描述，而 [60, 62, 61, 63, 94] 基本上采用朴素贝叶斯 方法。最后但同样重要的是， [43]受到遗传编程 的启发来归纳概念表达式，而 [65] 则基于强化学习 框架。最后， [73] 提出使用决策树来学习所谓的不相交公理，即 形如 $C \sqcap D \sqsubseteq \bot$ 的表达式，声明类 C 与 D 不相交。

6 结论与未来工作

在本研究中，我们探讨了如何从 OWL 2 本体中自动学习模糊概念包含公理，以描述个体被分类为目标类 T 的实例的充分条件。也就是说，给定一个 OWL 本体的目标类 T，我们探讨了如何推导模糊概念包含公理，以描述成为 T 的个体实例的充分条件。具体而言，我们提出了一种名为 PN-OWL 的两阶段算法，该算法受 PN-rule [2, 3, 40, 41] 启发，并针对 OWL 环境进行了调整。该算法的主要特点是：*(i)* 在 P 阶段，它生成一组模糊包含公理，即 P 规则，尽可能多地覆盖目标类 T 的实例，同时避免非正例数量过多；*(ii)* 在 N 阶段，它生成一组模糊包含公理，即 N 规则，尽可能多地覆盖 P 阶段所识别的类 T 的非正例实例；*(iii)* 随后，将这些模糊包含公理组合（聚合）为一个新的模糊包含公理，用以描述个体被分类为目标类 T 的实例的充分条件。此外，所有模糊包含公理都可能包含模糊概念和模糊具体域，每条公理都有一个加权系数（具体称为置信度或精确率），并且所有生成的模糊概念包含公理都可以直接编码为 模糊 *OWL 2* 公理。

我们还进行了广泛的评估，并将其与模糊 Foil-$\mathcal{DL}$进行了比较。评估结果表明，在有效性方面， PN-OWL 通常优于模糊 Foil-$\mathcal{DL}$ ；不过，由于 PN-OWL 的参数数量是模糊 Foil-$\mathcal{DL}$的两倍，寻找最优参数配置所花费的时间比为 Foil-$\mathcal{DL}$ 寻找最优参数配置要长得多。

关于未来工作，除了研究其他学习方法，以及论文各处提及的未来工作外，我们还设想了若干值得深入探讨的方面：*(i)* 模糊集的构造可能如何影响有效性，目前仍不明确。迄今为止，在用于构建模糊数据类型的均匀聚类算法和 C 均值聚类算法之间，我们尚未发现明显更优者——

[14]我们回顾一下， $\mathcal{EL}$ 是 $\mathcal{ALC}$的严格子逻辑。

用于构建模糊数据类型。这有些出乎意料。我们希望通过考虑各种替代方案，并 [1]和/或考虑基于数据属性聚合的聚类方法（即 多维聚类与单维聚类），对此进行更深入的研究；*(ii)* 此外，我们希望涵盖迄今为止尚未涉及的更多 OWL 数据类型（数值型和布尔型），例如字符串、日期等，也可能将其与一些经典机器学习方法结合使用（参见例如 [83]）；*(iii)* 我们希望研究计算方面的问题：迄今为止，对于某些本体，一次学习运行甚至可能需要一周时间（就我们可用的资源而言）。在这方面，我们希望研究并行化方法，同时考察高效、逻辑上可靠但不一定完备的推理算法对有效性的影响；*(iv)* 原则上，我们的两阶段算法 PN-OWL 在 P 阶段和 N 阶段使用的学习器方面都是参数化的（参见 PN-OWL 算法的第 5 行和第 19 行）：我们希望研究如何接入另一种替代方案，例如 Fuzzy OWL-Boost [20]，并验证其有效性；*(v)* 我们还希望评估在我们的设置中，信息增益之外的其他评分函数所带来的影响（参见 式 12），包括三角范数和 r-蕴涵的各种替代选择；此外（*vi*），我们正在探索将模糊描述逻辑学习与亚符号学习方法相结合，例如神经网络，这项工作已经在进行中。

此外，我们确实希望考虑扩展假设语言 $\mathcal{EL}$(**D**)，加入所谓的阈值概念 [11]，其形式为 $C[\geq d]$ （或 $C[\leq d]$），其中 $d \in [0,1]$ 和 C 要么是类名，要么是存在限制，其预期含义是“$C[\geq d]$ （或 $C[\leq d]$）是由那些作为 C 实例的程度大于（或小于）或等于 d的个体构成的模糊集。”这样便能提供一种更精细的假设语言，使规则的每个合取项都可以分别定义阈值，而不是像现在这样通过规则置信度阈值来定义。参考葡萄酒质量本体和目标葡萄酒 1，下面是一个此类规则的模糊 OWL 2 示例

```
(implies (and (some alcohol alcohol_VH)[<= 0.786]
              (some sulphates sulphates_H)[>= 0.289]
              (some pH pH_L)[<= 0.106])
              1)
```

其预期含义是：“如果对于某个个体（葡萄酒） a，酒精含量为很高的程度小于或等于 0.786，硫酸盐含量为高的程度大于或等于 0.289，并且 pH 值为低的程度小于或等于 0.106，那么就将 a在某种程度上（例如，作为一个合取项实例的程度中的最小值）分类为目标类 1 的实例。

致谢

本研究部分由 TAILOR 项目支持，该项目由欧盟地平线2020研究与创新计划资助（拨款协议编号：952215）。本研究也部分获得了 H2020 DeepHealth 项目（拨款协议编号：825111）的支持。本文还获得了 FAIR（未来人工智能研究）项目的支持，该项目由下一代欧盟计划在 PNRR-PE-AI 计划下资助（M4C2，投资 1.3，人工智能方向）。此外，本研究也部分获得了 H2020 STARWARS 项目（拨款协议编号：101086252）的支持，该项目属于地平线 TMA MSCA 人员交流项目行动类型。

我们谨向国家研究委员会 ICT-SAC 部门的 CNR 服务中心致以谢意，感谢他们提供宝贵的计算服务与资源。我们特别感谢 ICT-SAC 的 Giorgio Bartoccioni 工程师给予的技术支持。

参考文献

[1] *Special issue on Fuzzy Clustering, Fuzzy Sets and Systems*，第389卷。爱思唯尔，2020年。[2] Ramesh C. Agarwal 和 Mahesh V. Joshi。Pnrule：一种用于数据挖掘中学习分类器模型的新框架（网络入侵检测案例研究）。技术报告 rc 21719，IBM Research 报告，2000 年。[3] Ramesh C. Agarwal 和 Mahesh V. Joshi。Pnrule：一种用于数据挖掘中学习分类器模型的新框架（网络入侵检测案例研究）。载于 *Proceedings of the 2001 SIAM International Conference on Data Mining (SDM-01)* 1 17, 2001 ，页码 – 。

[4] H. Anderson 和 Phil Roth。Ember：用于训练静态 PE 恶意软件机器学习模型的开放数据集。*ArXiv*，abs/1804.04637，2018年。[5] Alessandro Artale、Diego Calvanese、Roman Kontchakov 和 Michael Zakharyaschev。DL-Lite家族及其关系。 *Journal of Artificial Intelligence Research*，36:1–69，2009年。[6] Franz Baader、Rafael Pe˜naloza 和 Boontawee Suntisrivaraporn。描述逻辑 $\mathcal{EL}^+$中的定位。载于 *Proceedings of the 30th Annual German Conference on Advances in Artificial Intelligence (KI-07)*，《计算机科学讲义》第4667卷，页码52–67，柏林，海德堡，2007年。施普林格出版社。[7] Liviu Badea 和 Shan-Hwei Nienhuys-Cheng。描述逻辑的细化算子。载于 *Inductive LogicProgramming, 10th International Conference, ILP-00*，《 *Lecture Notes in Computer Science*》第1866卷，页码40–59。施普林格，2000年。[8] James C. Bezdek。 *Pattern Recognition with Fuzzy Objective Function Algorithms*。施普林格出版社，1981年。[9] Stephan Bloehdorn 和 York Sure。用于本体实例数据挖掘的核方法。载于 *The Semantic Web 6th International Semantic Web Conference 2nd Asian Semantic Web Conference ISWC 2007 +* , , , , , , , , *ASWC 2007 Busan Korea November 11-15 2007.* 4825 *Lecture Notes in Computer Science* 第卷，页码58–71。施普林格出版社，2007年。[10] Fernando Bobillo、Marco Cerami、Francesc Esteva、 ̀Angel Garc´ıa-Cerda˜na、Rafael Pe˜naloza 和 Umberto Straccia。数学模糊逻辑框架下的模糊描述逻辑。载于 Carles Noguera Petr Cin-tula、Christian Ferm ¨ *Handbook of Mathematical Fuzzy Logic Volume 3* 58 *Studies in* uller 编， , , *Logic Mathematical Logic and Foundations* 16, 1105 1181 2015 第卷， ，第章页码 – 。College Publications， 。[11] Fernando Bobillo 和 Umberto Straccia。fuzzyDL：一种表达能力强的模糊描述逻辑推理器。载于 *2008 International Conference on Fuzzy Systems (FUZZ-08)* 923 930 2008 ，页码 – 。

IEEE计算机学会， 。[12] Fernando Bobillo 和 Umberto Straccia。在 OWL 2 *Proceedings of the 19th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2010)* 中表示模糊本体。载于 ，页码 – 。IEEE出版社，2010年7月。[13] Fernando Bobillo 和 Umberto Straccia。使用 OWL 2 *International Journal of Approximate Reasoning* 52 1073 1094, 2011 表示模糊本体。 ， : – 。[14] Fernando Bobillo 和 Umberto Straccia。模糊本体的聚合算子。 *Applied Soft Computing*，13 9 3816( ): –3830，2013年。[15] Fernando Bobillo 和 Umberto Straccia。模糊本体推理器 *fuzzyDL*。 *Knowledge-Based Systems*，95 12 : – 34，2016年。[16] Fernando Bobillo 和 Umberto Straccia。重新审视模糊 OWL 2 EL 中的推理。 *Fuzzy Sets and Systems*，351 1: –40，2018年。[17] Stefan Borgwardt、Marco Cerami 和 Rafael Pe˜naloza。Łukasiewicz t-范数下模糊 $\mathcal{EL}$ 的复杂性。 *International Journal of Approximate Reasoning*，91:179–201，2017年。[18] Lorenz B¨uhmann、Jens Lehmann 和 Patrick Westphal。DL-Learner——语义网上归纳学习的框架。 *Journal of Web Semantics*，39:15–24，2016年。[19] Lorenz B¨uhmann、Jens Lehmann、Patrick Westphal 和 Simon Bin。DL-Learner：面向语义网数据的结构化机器学习。载于 *Companion of the The Web Conference 2018 on The Web Conference 2018*, *WWW 2018 France 2018* 467 471 2018 , *Lyon* , *April 23-27*，页码 – 。ACM， 。[20] Franco Alberto Cardillo 和 Umberto Straccia。Fuzzy OWL-Boost：通过实值提升学习模糊概念包含。 *Fuzzy Sets and Systems*，438:164–186，2022年。[21] Marco Cerami 和 Umberto Straccia。Łukasiewicz t-范数下模糊描述逻辑的（不可）判定性。 *Information Sciences*，227:1–21，2013年。[22] Mahsa Chitsaz、Kewen Wang、Michael Blumenstein 和 Guilin Qi。通过细化与强化进行 $\mathcal{EL}^{++}$的概念学习。载于 *Proceedings of the 12th Pacific Rim international conference on Trends in Artificial Intelligence* 12, 15 26, 2012 ，PRICAI’ 页码 – 柏林，海德堡， 。施普林格出版社。

[23] Marcos E. Cintra、Maria Carolina Monard 和 Heloisa de Arruda Camargo。关于规则学习方法：经典方法与模糊方法的比较分析。载于 *Soft Computing: State of the Art Theory and Novel Applications* 291, 89 104 2013 , 卷页码 – 。

施普林格出版社，。[24] Claudia d'Amato。面向语义网的机器学习：经验教训与未来研究方向。*Semantic Web*，11(1):195–203，2020年。[25] Mar´ıa Jos´é del Jes´ús、Frank Hoffmann、Luis Junco Navascu´és 和 Luciano S´ánchez。利用进化提升算法归纳模糊规则分类器。

IEEE Transactions on Fuzzy Systems 12 3 296 308, , ( ): – 。 2004

[26] Mario Drobics、Ulrich Bodenhofer 和 Erich-Peter Klement。Fs-foil：一种用于提取可解释模糊描述的归纳学习方法。*International Journal of Approximate Reasoning*，32(2-3):131–152，2003年。[27] Dheeru Dua 和 Casey Graff。UCI 机器学习库，2017年。[28] Nicola Fanizzi。利用信息论启发式方法进行描述逻辑中的概念归纳。

Int. J. Se- , ( ): – 。 *mantic Web Inf. Syst.* 7 2 23 44, 2011

[29] Nicola Fanizzi 和 Claudia d'Amato。用于 *ALC* 概念描述的声明式核。载于 *Foundationsof Intelligent Systems, 16th International Symposium, ISMIS 2006, Bari, Italy, September 27-29*，2006年，*Proceedings* 4203 *Lecture Notes in Computer Science* 322 331 2006 , 卷，页码 – 。施普林格出版社，。[30] Nicola Fanizzi 、 Claudia d'Amato 和 Floriana Esposito。描述逻辑中的 DL-FOIL 概念学习。载于 Filip Zelezn ´ *Inductive Logic Programming* 5194 *Lecture Notes in* y 和 *Computer Science* 107 121 2008 Nada Lavra˘c（编），，卷，页码 – 。施普林格，。[31] Nicola Fanizzi、Claudia d'Amato 和 Floriana Esposito。利用非参数方法对本体进行近似分类和检索的分类器归纳。 *Int. J. Semantic Computing*，2(3):403–423，2008年。[32] Nicola Fanizzi、Claudia d'Amato 和 Floriana Esposito。通过术语决策树在Web本体中归纳概念。载于 *Machine Learning and Knowledge Discovery in Databases, European Con-ference, ECML PKDD 2010, Barcelona, Spain, September 20-24*，2010年，*Proceedings, Part I*，卷6321，*Lecture Notes in Computer Science*，页码442–457。施普林格出版社，2010年。[33] Nicola Fanizzi、Claudia d'A mato 和 Floriana Esposito。迈向术语决策树的归纳。载于 *Proceedings of the 2010 ACM Symposium on Applied Computing*，SAC '10，页码1423–1427，美国纽约州纽约市，2010年。ACM。[34] Nicola Fanizzi、Claudia d'Amato 和 Floriana Esposito。通过核机器为Web本体归纳鲁棒分类器。*J. Web Sem.*，11:1–13，2012年。[35] Nicola Fanizzi、Giuseppe Rizzo 和 Claudia d'Amato。提升描述逻辑概念学习器。载于 *The Semantic Web- 16th International Conference, ESWC-19*，卷11503，*Lecture Notes in Computer Science*，页码68–83，2019年。[36] Nicola Fanizzi、Giuseppe Rizzo、Claudia d'Amato 和 Floriana Esposito。Dlfoil：重新审视类表达式学习。载于 *Knowledge Engineering and Knowledge Management - 21st International Conference EKAW-* , , 卷，页 *18* 11313 *Lecture Notes in Computer Science* 98 113, 2018 码 – 。[37] Petr H´ájek。 *Metamathematics of Fuzzy Logic*。Kluwer，1998年。[38] Ignacio Huitzil、Umberto Straccia、Natalia D´íaz-Rodr´íguez 和 Fernando Bobillo。Datil：学习模糊本体数据类型。载于 *Proceedings of the 17th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2018) Part II* 854 *Communi-* , , 卷，页 *cations in Computer and Information Science* 100 112 2018 码 – 。施普林格，六月。[39] Josu´é Iglesias 和 Jens Lehmann。探索将模糊逻辑能力整合到基于本体的归纳逻辑程序设计框架中。载于 *Proceedings of the 11th International Conference on Intelligent Systems Design and Applications (ISDA 2011)* 1323 1328, 2011 , 页码 – 。

[40] Mahesh V. Joshi、Ramesh C. Agarwal 和 Vipin Kumar。大海捞针：通过两阶段规则归纳对稀有类别进行分类。载于 Sharad Mehrotra 和 Timos K. Sellis（编）， *Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data Santa Barbara CA USA May , , , , , 21-24 2001* 91 102 2001 页码 – 。ACM，。[41] Mahesh V. Joshi、Ramesh C. Agarwal 和 Vipin Kumar。预测稀有类别：提升能否让任何弱学习器变强? 载于 *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining July 23-26 2002 Edmonton Alberta Canada* 297 306 2002 , , , , , 页码 – 。ACM，。

[42] S. Konstantopoulos和 A. Charalambidis。将描述逻辑学习表述为归纳逻辑程序设计任务。载于 *Proceedings of the 19th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2010)* 1 7 2010 ，页码 – 。IEEE出版社，。[43] Jens Lehmann。本体类的混合学习。载于 *Machine Learning and Data Mining in PatternRecognition, 5th International Conference, MLDM 2007, Leipzig, Germany, July 18-20*, 2007年, *Proceedings*, 第4571卷, *Lecture Notes in Computer Science*, 页码883–898。施普林格出版社，2007年。[44] Jens Lehmann。DL-Learner：描述逻辑中的概念学习。 *Journal of Machine Learning Research*, 10 2639: –2642, 2009年。[45] Jens Lehmann 和 Christoph Haase。 *EL\ mathcal{EL}* 描述逻辑中的理想向下细化。载于 *Inductive Logic Programming, 19th International Conference, ILP 2009, Leuven, Belgium, July 02-04, 2009. Revised Papers* 5989 *Lecture Notes in Computer Science* 73 87 2009 ，卷，页码 – 。施普林格，。[46] Jens Lehmann 和 Pascal Hitzler。基于细化算子的 *ALC* 描述逻辑学习算法。载于 *Inductive Logic Programming 17th International Conference ILP 2007 Corvallis OR USA June 19-21 2007 Revised Selected Papers* 4894 *Lecture Notes in Computer Science* 147 160 ，，，，，，，，，卷，页码 – 。施普林格出版社，2007年。[47] Jens Lehmann 和 Pascal Hitzler。《描述逻辑细化算子的基础》。载于 H. Blockeel、J. Ramon、J. W. Shavlik 和 P. Tadepalli（编）， *Inductive Logic Programming*, 第4894卷, *Lecture Notes in Artificial Intelligence*, 页码161–174。施普林格，2008年。[48] Jens Lehmann 和 Pascal Hitzler。利用细化算子进行描述逻辑中的概念学习。 *Machine Learning* 78 1 2 203 250, 2010 ，(-): – 。[49] Francesca A. Lisi 和 Donato Malerba。al-log 中描述的理想细化。载于 *Inductive Logic Programming: 13th International Conference, ILP-03*, 第2835卷, *Lecture Notes in Computer Science*, 页码215–232，2003年。[50] Francesca A. Lisi 和 Umberto Straccia。用于自动归纳模糊本体公理的基于逻辑的计算方法。 *Fundamenta Informaticae*, 124(4):503–519，2013年。[51] Francesca A. Lisi 和 Umberto Straccia。用于学习模糊描述逻辑中 GCI 公理的系统。载于 *Proceedings of the 26th International Workshop on Description Logics (DL-13)* 1014 *CEUR Workshop Proceedings* 760 778 2013 ，卷，页码 – 。CEUR-WS.org，。[52] Francesca A. Lisi 和 Umberto Straccia。归纳逻辑程序设计能否处理不完整且模糊的结构化知识? 载于 Stephen H. Muggleton 和 Hiroaki Watanabe（编）， *Latest Advances in Inductive Logic Programming*, 第21章，页码199–206。世界科学出版社，2014年。[53] Francesca A. Lisi 和 Umberto Straccia。具有模糊具体域的描述逻辑学习。 *Fundamenta Informaticae* 140 3 4 373 391, 2015 ，(-): – 。[54] Francesca Alessandra Lisi。机器学习与数据挖掘中的逻辑：成果与未决问题。载于 *Proceedings of the 34th Italian Conference on Computational Logic, Trieste, Italy, June 19-21*, 2019年, 第2396卷, *CEUR Workshop Proceedings*, 页码82–88。CEUR-WS.org，2019年。[55] Francesca Alessandra Lisi 和 Umberto Straccia。利用归纳逻辑程序设计学习模糊描述逻辑中的包含公理。载于 *26th Italian Conference on Computational Logic (CILC-11)*, 第810卷，页码57–71。CEUR电子研讨会论文集，2011年。[56] Francesca Alessandra Lisi 和 Umberto Straccia。迈向模糊描述逻辑包含公理的学习。载于 *9th International Workshop on Fuzzy Logic and Applications (WILF-11)* 6857 *Lecture Notes in Computer Science* 58 2011 ，卷，页码–66，柏林，。施普林格出版社。[57] Francesca Alessandra Lisi 和 Umberto Straccia。归纳逻辑程序设计中的不完备性与模糊性处理。载于 *28th Italian Conference on Computational Logic (CILC-13)*, 第1068卷，页码179–193。CEUR电子研讨会论文集，2013年。[58] Francesca Alessandra Lisi 和 Umberto Straccia。一种面向不完备性与模糊性情形的类FOIL学习方法。载于 *23rd International Conference on Inductive Logic Programming* 8812 *Lecture Notes in Artificial Intelligence* 123 139, 2014 ，卷，页码 – 柏林，。施普林格出版社。修订版精选论文。[59] Thomas Lukasiewicz 和 Umberto Straccia。语义网描述逻辑中的不确定性与模糊性管理。 *Journal of Web Semantics*，6:291–308，2008年。[60] Pasquale Minervini、Claudia d'Amato 和 Nicola Fanizzi。在缺失知识的不同假设下学习术语朴素贝叶斯分类器。载于 *Proceedings of the 7th International Workshop onUncertainty Reasoning for the Semantic Web (URSW-11*, 第778卷, *CEUR Workshop Proceedings*，页码63–74。CEUR-WS.org，2011年。

[61] Pasquale Minervini、Claudia d'Amato 和 Nicola Fanizzi。《学习概率描述逻辑概念：关于缺失知识的不同假设》。载于 *Proceedings of the 27th Annual ACM Symposium o n Applied Computing*，SAC '12，第 378–383 页，纽约，纽约州，美国，2012年。ACM。[62] Pasquale Minervini、Claudia d'Amato 和 Nicola Fanizzi。《学习术语贝叶斯分类器——处理未知概念隶属关系的不同方法比较》。载于 *Proceedings o f the 9th Italian Convention on Computational Logic Rome Italy June 6-7 2012* 857 *CEUR* , , , , 卷号 , 页码 – 。[63] Pasquale Minervini 、 Claudia d'Amato 、 Nicola Fanizzi 和 Floriana Esposito。《在不完备性的不同假设下学习概率描述逻辑概念》。载于 *Uncertainty Reasoning for th e Semantic Web III - ISWC International Workshops, URSW 2011-2013, Revised Selected Papers*，《 *Lecture Notes in Computer Science*》第8816卷，页码184–201，2014年。[64] Boris Motik和Riccardo Rosati。描述逻辑与逻辑程序设计的忠实集成。载于 *Proceedings of the 20th international joint conference on Artifical intelligence*，页码477–482，旧金山，加利福尼亚州，美国，2007年。摩根·考夫曼出版社公司。[65] Matthias Nickles和Achim Rettinger。概念语义的交互式关系强化学习。*Machine Learning*，94(2):169–204，2014年。[66] Jos´e Otero和Luciano S´anchez。利用LogitBoost算法归纳描述性模糊分类器。*Soft Computing*，10(9):825–835，2006年。[67] OWL 2 Web本体语言文档概览。 `https://www.w3.org/TR/owl2-overview/`。万维网联盟，2009年。[68] Ana M. Palacios、Luciano S´anchez和In´es Couso。使用AdaBoost算法从低质量数据中提取模糊规则：一些初步结果。载于 *FUZZ-IEEE 2011, IEEE International Conference o n Fuzzy Systems, Taipei, Taiwan, 27-30 June, 2011*年， *Proceedings*，第1263–1270页。IEEE，2011年。[69] Luc De Raedt。 *Logical and relational learning*。认知技术。施普林格，2008年。[70] Luc De Raedt 和 Kristian Kersting。统计关系学习。见 *Encyclopedia of Machine Learnin g and Data Mining*，第1177–1187页。施普林格，2017年。[71] Achim Rettinger、Uta L¨osch、Volker Tresp、Claudia d'Amato 和 Nicola Fanizzi。挖掘语义网——面向下一代知识库的统计学习。 *Data Minining and Knowledge Discovery*，24 3 613( ): –662，2012年。[72] Giuseppe Rizzo、Claudia d' Amato 和 Nicola Fanizzi。论基于证据的术语决策树的有效性。见 *Foundations of Intelligent Systems - 22nd International Symposium, ISMIS-15*，第9384卷，收录于 *Lecture Notes in Computer Science*，第139–149页，2015年。[73] Giuseppe Rizzo、Claudia d'Amato 和 Nicola Fanizzi。一种基于术语聚类树的不相交性学习无监督方法。 *Semantic Web*，12(3):423–447，2021年。[74] Giuseppe Rizzo、Claudia d'Amato、Nicola Fanizzi 和 Floriana Esposito。解决语义网知识库中的类别不平衡学习问题。见 *Knowledge Engineering and Knowledge Managemen t - 19th International Conference, EKAW-14*，第8876卷，收录于 *Lecture Notes in Computer Science*，第453–468页，2014年。[75] Giuseppe Rizzo、Claudia d'Amato、Nicola Fanizzi 和 Floriana Esposito。《迈向基于证据的术语决策树》。载于 *Information Processing and Management of Uncertainty in Knowledge-Base d Systems - 15th International Conference, IPMU 2014, Montpellier, France, July 15-19, 2014*年， *Proceedings, Part I* 442 36 45 2014 ，收录于 *Communications in Computer and Information Science*，第–页。施普林格，。[76] Giuseppe Rizzo、Claudia d'Amato、Nicola Fanizzi 和 Floriana Esposito。《通过基于证据的模型及其集成进行归纳分类》。载于 *The Semantic Web. Latest Advances and New Domains -12th European Semantic Web Conference, ESWC-15*，第9088卷，收录于 *Lecture Notes in Computer Science*，第418–433页，2015年。[77] Giuseppe Rizzo、Claudia d'Amato、Nicola Fanizzi 和 Floriana Esposito。《面向数据网归纳分类的基于树的模型》。 *Journal of Web Semantics*，45:1–22，2017年。[78] Giuseppe Rizzo、Nicola Fanizzi 和 Claudia d'Amato。《类表达式归纳作为概念空间探索：从 DL-FOIL 到 DL-FOCL》。 *Future Generation Computing Systems*，108:256–272，2020年。[79] Giuseppe Rizzo、Nicola Fanizzi、Claudia d'Amato 和 Floriana Esposito。《通过证据术语树与森林利用 Web本体进行近似分类》。 *International Journal of Approximat e Reasoning*, 92:340–362, 2018年。

[80] Giuseppe Rizzo、Nicola Fanizzi、Jens Lehmann 和 Lorenz B¨uhmann。在术语决策树学习中集成新的细化算子。载于 *Knowledge Engineering and Knowledge Management - 20th International Conference EKAW-16* 10024 *Lecture Notes in Computer Science* 511 , ,第 卷，页码 –526，2016年。[81] Luciano S´anchez 和 Jos´e Otero。在单胜者推理下提升分类问题中的模糊规则。*International Journal of Intelligent Systems*，22(9):1021–1034，2007年。[82] Mathieu Serrurier 和 Henri Prade。通过学习不同类型的模糊规则，提升归纳逻辑程序设计的表达能力。 *Soft Computing*，11(5):459–466，2007年。[83] Shai Shalev-Shwartz 和 Shai Ben-David。*Understanding Machine Learning: From Theory to Algorithms*。剑桥大学出版社，2014年。[84] D. Shibata、N. Inuzuka、S. Kato、T. Matsui 和 H. Itoh。一种基于模糊逻辑程序设计的归纳算法。载于 N. Zhong 和 L. Zhou 编，*Methodologies for Knowledge Discovery and Data Mining,Third Pacific-Asia Conference, PAKDD-99, Beijing, China, April 26-28*，1999年， *Proceedings*， *Lecture Notes in Computer Science*第1574卷，页码268–273。施普林格，1999年。[85] Umberto Straccia。具有模糊具体域的描述逻辑。载于 Fahiem Bachus 和 Tommi Jaakkola 编，*21st Conference on Uncertainty in Artificial Intelligence (UAI-05)*，页码559–567，爱丁堡，苏格兰，2005年。AUAI出版社。[86] Umberto Straccia。 *Foundations of Fuzzy Logic and Semantic Web Languages*。CRC信息学研究丛书。Chapman & Hall，2013年。[87] Umberto Straccia 和 Matteo Mucci。pFOIL-DL：使用概率集成估计从清晰OWL数据中学习（模糊） $\mathcal{EL}$ 概念描述。载于 *Proceedings of the 30th Annual ACM Symposium on Applied Computing (SAC-15)* 345 352, 2015 ，页码 –，萨拉曼卡，西班牙，。ACM。[88] Peter Svec、Stefan Balogh 和 Martin Homola。静态恶意软件检测描述逻辑概念学习算法的实验评估。载于 Paolo Mori、Gabriele Lenzini 和 Steven Furnell 编，*Proceedings of the 7th International Conference on Information Systems Security and Privacy ICISSP* , , *2021* *2021* 792 799 2021 *Online Streaming, February 11-13*, ,页码 –。SCITEPRESS，。[89] Peter Svec、Stefan Balogh、Martin Homola 和 J´an Kluka。用于训练PE恶意软件检测模型的知识库数据集。 *CoRR*，abs/2301.00153，2023年。[90] Vicen.c Torra 和 Yasuo Narukawa。 *Information Fusion and Aggregation Operators*。认知技术。施普林格出版社，2007年。[91] Patrick Westphal、Lorenz B¨uhmann、Simon Bin、Hajira Jabeen 和 Jens Lehmann。SML-bench——结构化机器学习基准测试框架。 *Semantic Web*，10(2):231–245，2019年。[92] Hong yang Zhu、Yi Ding、Hong Gao 和 Wei Liu。AdaBoost算法分类中的模糊预测。载于 *International Conference on Oriental Thinking and Fuzzy Logic* 443 *Advances in Intelligent Systems and Computing* 129 136 2016 , 第卷，页码 –。施普林格，。[93] L. A. Zadeh。模糊集。 *Information and Control*，8(3):338–353，1965 年。[94] Man Zhu、Zhiqiang Gao、Jeff Z. Pan、Yuting Zhao、Ying Xu 和 Zhibin Quan。通过在belnet+中进行推理，从不完整数据中学习Tbox。 *Knoweledge-Based Systems*，75:30–40，2015年。

数据集简要介绍

下文简要介绍了我们实验中使用的表 3 中的 OWL本体 s.

SemanticBible (NTN)。 新约人物名称 (NTN) 是一个本体，用于描述新约中的每个具名事物，总计约 600 个名称。每个具名事物（一个实体）都根据其类别进行分类，包括上帝、耶稣、男性和女性个体、人群以及地点。这些实体通过属性彼此关联，构成一个信息网络。[15] 目标是学习成为女性的充分条件。

淋巴造影。该本体描述淋巴造影患者数据，目标是根据淋巴造影患者数据预测诊断类别 [91]。

[15] `http://semanticbible.com/ntn/ntn-overview.html`

乳腺X线摄影。该本体描述乳腺X线筛查数据，目标是根据筛查数据预测乳腺癌严重程度 [91]。

恶意软件。该本体描述了 PE 恶意软件本体，为可移植可执行文件（PE，Windows二进制格式）恶意软件文件提供可重复使用的语义模式 [88,89]。该本体的设计借鉴了 EMBER数据集 中数据的结构，[16] 该数据集旨在用于静态恶意软件分析 [4]。

以下数据集取自知名的加州大学欧文分校机器学习资料库 [27]。

鸢尾花。该数据集包含 3 个类，每类有 50 个实例；每个类对应一种鸢尾属植物。属性包括：萼片长度（厘米）、萼片宽度（厘米）、花瓣长度（厘米）和花瓣宽度（厘米）。目标类为：刚毛鸢尾、变色鸢尾和维吉尼亚鸢尾。

葡萄酒。这些数据是对意大利同一地区种植、但来自三种不同栽培品种的葡萄酒进行化学分析所得的结果。分析测定了三种葡萄酒中各自含有的 13 种成分的含量。属性包括酒精、苹果酸、灰分、灰分碱度、镁、总酚、类黄酮、非黄酮类酚、原花青素、颜色强度、色调、稀释葡萄酒的 OD280/OD315 和脯氨酸。目标类为葡萄酒 1，2 和 3。

葡萄酒质量。 该数据集与葡萄牙“Vinho Verde”葡萄酒的红葡萄酒和白葡萄酒品种有关。其目标是根据理化测试对葡萄酒质量进行建模。由于隐私和物流方面的问题，仅提供理化（输入）变量和感官（输出）变量（例如，没有关于葡萄品种、葡萄酒品牌、葡萄酒售价等的数据）。属性包括：固定酸度、挥发性酸度、柠檬酸、残糖、氯化物、游离二氧化硫、总二氧化硫、密度、pH值、硫酸盐、酒精和质量（评分范围为0至10）。目标是描述优质红葡萄酒，其定义为质量评分大于或等于7的红葡萄酒。在训练和测试期间，质量属性已从本体中移除。

酵母。 该数据集用于预测蛋白质的细胞定位位点（10个目标类）。属性集合包括：序列名称（SWISS-PROT数据库中的登录号）、mcg（用于信号序列识别的McGeoch方法）；gvh（用于信号序列识别的von Heijne方法）；alm（ALOM跨膜区预测程序的评分）；mit（对线粒体和非线粒体蛋白质N端区域（长20个残基）的氨基酸含量进行判别分析所得的评分）；erl（是否存在子串“HDEL”，该子串被认为可作为信号，使蛋白质滞留在内质网腔中；二元属性）；pox（C端的过氧化物酶体靶向信号）；vac（对液泡蛋白质和细胞外蛋白质的氨基酸含量进行判别分析所得的评分）；以及nuc（对核蛋白质和非核蛋白质的核定位信号进行判别分析所得的评分）。

[16] https://GitHub.com/Elastic/Ember