LYRA: Belief-Driven Scalable Class Expression Learning in Description Logics

Amgad Abdulmaqsod[1,2][0009−0002−4151−4588], Yasir Mahmood[1][0000−0002−5651−5391], Axel-Cyrille Ngonga Ngomo[1][0000−0001−7112−3516], and Mohamed Sherif[1][0000−0002−9927−2203]

[1] Data Science Group (DICE), Heinz Nixdorf Institute, Paderborn University
https://dice-research.org/
[2] UMR SayFood, Université Paris-Saclay, INRAE, AgroParisTech, 91120 Palaiseau, France
{amgad.mahmoud,yasir.mahmood}@uni-paderborn.de,
{axel.ngonga,mohamed.sherif}@upb.de

Abstract. Explainable artificial intelligence (XAI) is essential for critical domains such as healthcare and autonomous systems to build trust and confidence in real-world deployment. In this context, description logic knowledge bases (KBs) provide structured and semantically rich representations that support reasoning and informed decision-making. A core task in applying KBs to XAI is class expression learning (CEL), which generates explainable logical descriptions for classifying instances within KBs. Unlike black-box models with opaque internal mechanisms, CEL provides global explainability and ease of integration with domain knowledge. However, current approaches to CEL face significant limitations such as poor scalability, failure to capture rare patterns, and limited exploration of the vast class expression search space. To overcome these limitations, we introduce LYRA, a novel *multi-agent deep reinforcement learning* framework that formulates CEL as a collaborative planning task under uncertainty. The integration of the *Dempster–Shafer theory* enables agents to effectively reason under ambiguity and manage conflicting or inconsistent information. Our experiments show that LYRA outperforms state-of-the-art methods on seven out of eight datasets, demonstrating robust and scalable CEL. Additionally, LYRA offers interpretable decisions and employs advanced search strategies, enabling the discovery of more precise and expressive class expressions than existing approaches.

Keywords: Explainable Artificial Intelligence · Description Logics · Deep Reinforcement Learning.

1 Introduction

One significant challenge associated with artificial intelligence (AI) is its limited explainability. This poses a critical obstacle in high-stakes domains such as healthcare and autonomous driving, where transparent decision-making is essential. For instance, *graph neural networks* (GNNs) are employed to classify nodes for tasks like predicting potential drug candidates [7] or forecasting material properties in materials design [33]. However, the reasoning behind their predictions remains largely opaque to humans. Moreover, debugging such models for error correction is non-trivial, particularly at

scale. Description logics (DLs) [1,2] offer a formal foundation for structured knowledge representation and reasoning. In this context, class expression learning (CEL) is the task of automatically identifying a DL concept that effectively distinguishes between positive and negative individuals within a given DL knowledge base (KB). CEL has been studied extensively, with numerous methods proposed to address it. One major line of work formulates CEL as a search problem over an infinite, quasi-ordered space of class expressions [4]. This search is guided by a so-called *refinement operator* [17,15] and steered by a heuristic function to converge on a suitable solution. Alternatively, many algorithms rely on pre-trained models [12], which can facilitate faster convergence but constrain the search space and risk overlooking class expressions during training. Other approaches are based on knowledge graph (KG) traversal [3]. However, the effectiveness of such approaches can vary with the structure of the underlying KG. For example, some methods struggle with scalability on large KGs [8], while others underperform on smaller ones [11].

The objective of this work is to develop a solution to CEL with an interpretable reasoning process, enabling domain experts to validate its predictions. To this end, we enhance *multi-agent deep reinforcement learning* (MADRL) by integrating *belief function theory* (BFT) [32], which augments both the expressiveness and the predictive control of agents' decision-making. We propose LYRA, a novel approach that formulates CEL as a planning task and sequentially generates class expressions by combining BFT with MADRL. LYRA is distinguished from prior work in the following aspects: (1) In contrast to neural-based approaches, LYRA learns from its own experience, omitting the need of large training KGs. (2) Unlike traversal-based methods that degrade on large KGs and neural models that underperform on small ones, LYRA is empirically robust to KG size and topology. (3) The integration of BFT enables predictive control in decision-making through its combination rules and improves the interpretability of agent actions. (4) Leveraging the properties of BFT combination rules, LYRA preserves complete evidential information during updates, ensuring stability in sparse or noisy ontological environments and enhancing the sample efficiency of LYRA.

A summary of our contributions is as follows:

- We propose a solution to CEL in terms of planning. To the best of our knowledge, this approach to CEL has not been considered before. LYRA achieves this by applying MADRL, allowing adaptive and experience-driven concept discovery.
- LYRA is the first approach to integrate MADRL with BFT to solve CEL for the DL $\mathcal{ALCQ}(\mathcal{D})$, opening further research directions within AI planning and CEL.
- We present an extensive experimental evaluation of LYRA against state-of-the-art models over eight datasets of varying sizes.

Code and supplementary material are available at: https://github.com/lyrdl/LYRA. The remainder of this paper is organized as follows. We begin by reviewing related work on various CEL methodologies based on different classes of approaches in Section 2. Next, in Section 3, we introduce the necessary background concepts and terminology to support a clear understanding of the proposed method. We then present our approach in Section 4. The experimental setup and evaluation results are discussed in Section 5. Finally, we conclude the paper and present some of our future work ideas in Section 6.

2 Related Works

Search-based approaches represent the earliest attempts at CEL. DL-LEARNER [17] utilizes refinement operators to traverse the subsumption hierarchy $\sqsubseteq$ in the description logic ($\mathcal{ALCQ}$). However, the operator's inherent (infinite) size necessitates iterative horizontal expansion and redundancy checks, limiting its scalability. DL-FOC [23] performs the exploration of the conceptual search space under the open-world assumption. The authors incorporate strategies like lookahead, repeated hill-climbing, and tabu search meta-heuristics to overcome the limitations of traditional hill-climbing methods. However, this exhaustive search is prone to overfitting and high runtimes. In most cases, search-based approaches are infeasible in terms of both time complexity and scalability [12] due to the sheer size of the search space.

Evolutionary Algorithms (EA)-based approaches [29] are another class of techniques for the CEL problem, first proposed in [14] for DLs. This study combined genetic programming with refinement operators for learning class expressions. These operators are transformed into stochastic genetic refinement operators that choose between specialization and generalization based on the explored expressions. Nevertheless, this approach faces severe issues regarding the runtime complexity; for example, it needed 950 seconds for a population of 700 expressions. Another promising approach, EVOLEARNER [8], is widely regarded as state-of-the-art. It initializes the population by generating candidates through a biased random walk starting from positive examples. Yet, it lacks robustness and generality across complex or noisy KBs.

Recently, more advanced *neural-based approaches* have been introduced. For example, *NCES2* [11] begins by generating a training dataset to fit the model and employs the Set Transformer [13] to encode positive and negative examples. *NCES2* maps high-dimensional latent representations directly to symbolic expressions, which complicates optimization and limits the model's capacity to generate novel expressions that substantially diverge from those seen during training.

Deep reinforcement learning (DRL) is another line of work within the CEL setting. DRILL [6] integrates *deep q-networks* (DQN) [20] with symbolic refinement-based search in DLs. It overcomes the myopic behavior of traditional heuristic functions by learning a non-myopic heuristic that estimates cumulative discounted rewards for state transitions, enabling efficient navigation through the quasi-ordered space of class expressions. However, DQN suffers from sample inefficiency and slow convergence [9].

In parallel, *sampling-based approaches* have also been explored. Baci and Heindorf [3] addressed the problem by first generating a reduced KG using both classical sampling methods (e.g., random node/edge selection, walks, and forest-fire) and task-specific, problem-centered variants. Yet, their sampling strategy relies mainly on graph topology, overlooking logical axioms and limiting semantic representation. Moreover, the stochastic nature of sampling methods often results in output inconsistency.

3 Preliminaries

We next present essential background concepts for a smoother reading of the remainder of the paper. Additional background material is included within the Approach section, where it naturally supports the context for more clarity.

Table 1: Syntax & semantics for $\mathcal{ALCQ}$ concepts. $\mathcal{I}$ denotes an interpretation with domain $\Delta^{\mathcal{I}}$. $\mathcal{D}$ is the set of data values (strings, numbers, Booleans, etc).

Construct	**Syntax**	**Semantics**
Atomic concept	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
Atomic role	r	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
		$\mathcal{ALC}$
Top concept	$\top$	$\Delta^{\mathcal{I}}$
Bottom concept	$\bot$	$\emptyset$
Negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
Conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
Disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
Existential restriction	$\exists r.C$	$\{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid \exists b^{\mathcal{I}} \in C^{\mathcal{I}}, (a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}\}$
Universal restriction	$\forall r.C$	$\{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid \forall b^{\mathcal{I}}, (a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}} \Rightarrow b^{\mathcal{I}} \in C^{\mathcal{I}}\}$
		$\mathcal{Q}$
At least restriction	$\geq n\, r.C$	$\{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid |\{b^{\mathcal{I}} \in C^{\mathcal{I}} \mid (a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}\}| \geq n\}$
At most restriction	$\leq n\, r.C$	$\{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid |\{b^{\mathcal{I}} \in C^{\mathcal{I}} \mid (a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}\}| \leq n\}$
		$(\mathcal{D})$
Concrete role	d	$d^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \mathcal{D}$
Exact value restriction	$d = v$	$\{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid (a^{\mathcal{I}}, v) \in d^{\mathcal{I}}\}$
Max restriction	$d \leq v$	$\{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid \exists w \in \mathcal{D}, (a^{\mathcal{I}}, w) \in d^{\mathcal{I}} \wedge w \leq v\}$
Min restriction	$d \geq v$	$\{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid \exists w \in \mathcal{D}, (a^{\mathcal{I}}, w) \in d^{\mathcal{I}} \wedge w \geq v\}$

3.1 Description Logic Knowledge Bases

We denote by N_C the set of *atomic concepts*. Similarly, N_I and N_R denote the set of *individuals* and *roles*, respectively. A *TBox* $\mathcal{T}$ contains General Concept Inclusions (GCIs) of the form $C \sqsubseteq D$, where C, D are concepts (also known as *class expressions*). An *ABox* $\mathcal{A}$ includes the assertions having the form $C(a)$ (concept assertion) or $r(a, b)$ (role assertion), for individuals a, b, concept C, and role r. A DL knowledge base is a pair $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, for a TBox $\mathcal{T}$ and ABox $\mathcal{A}$. The *vocabulary* of a KB $\mathcal{K}$ is the set of all the concept, role and individual names that appear in $\mathcal{K}$.

In this paper, we focus on the DL $\mathcal{ALCQ}(\mathcal{D})$ [24]. $\mathcal{ALC}$ allows basic Boolean operations on concepts along with existential and universal restrictions, $\mathcal{Q}$ extends $\mathcal{ALC}$ by allowing cardinality restrictions, and $\mathcal{D}$ allows expressing the data properties. The syntax and semantics for concepts in $\mathcal{ALCQ}$ are given in Table 1.

Let $C \sqsubseteq D$ be a GCI, and $\mathcal{I}$ be an interpretation. Then, $\mathcal{I}$ satisfies $C \sqsubseteq D$, denoted as $\mathcal{I} \models C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. Similarly, $\mathcal{I}$ satisfies an assertion $C(a)$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$. The assertion $r(a, b)$ is satisfied by $\mathcal{I}$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$.

We next introduce the *Class Expression Learning* (CEL) problem over a DL KB.

Definition 1. *Let $\mathcal{K}$ be a DL KB and $E^+, E^- \subset N_I$ be sets of positive and negative individuals, respectively, such that $E^+ \cap E^- = \emptyset$. The CEL problem is to find a DL class expression H such that:*

$$\forall p \in E^+ : \mathcal{K} \models H(p) \quad \textit{and} \quad \forall n \in E^- : \mathcal{K} \not\models H(n). \tag{1}$$

3.2 Decentralized Partially Observable Markov Decision Process

To present CEL as a planning task, we require the following terminology.

Definition 2. *A **decentralized partially observable Markov decision process** (short, Dec-POMDP) is a tuple $M = \langle N, S, \mathbb{A}, \Omega, P, O, R, T, s_0 \rangle$, where $N = \{1, \ldots, k\}$ is the finite set of **agents**, S is the **state space** with initial state $s_0 \in S$. For an agent $i \in N$, we let A_i and Ω_i denote i's action space and observation space, respectively. Then, $\mathbb{A} = \prod_{i \in N} A_i$ and $\Omega = \prod_{i \in N} \Omega_i$ represent the **joint action** and **joint observation** space of M, respectively. In other words, a joint action in $\mathbb{A}$ takes the form $\mathbf{a} = (a_1, \ldots, a_k)$ and a joint observation in Ω as $\mathbf{o} = (o_1, \ldots, o_k)$. The **transition function** $P(s' \mid s, \mathbf{a})$ specifies the transition probability from state s to s' under joint action $\mathbf{a} \in \mathbb{A}$, and the **observation function** $O(\mathbf{o} \mid s', \mathbf{a})$ gives the probability of joint observation $\mathbf{o} \in \Omega$ conditioned on successor state s' and joint action $\mathbf{a}$. The **reward function** $R : S \times \mathbb{A} \to \mathbb{R}$ assigns a shared immediate reward, and $T \in \mathbb{N}$ denotes the planning **time horizon**.*

We next introduce some terminology from the belief function theory. A *frame of discernment* Θ is a finite set of mutually exclusive and exhaustive hypotheses. A *mass function* over Θ is a mapping $m : 2^\Theta \to [0, 1]$ satisfying: $m(\varnothing) = 0$ and $\sum_{A \subseteq \Theta} m(A) = 1$. Notice that a mass function is also known as a *basic probability assignment* (BPA). For a mass function m on a frame of discernment Θ, the associated *belief function* is the mapping $Bel : 2^\Theta \to [0, 1]$ defined as $Bel(A) = \sum_{B \subseteq A} m(B)$ for each $A \subseteq \Theta$, and satisfies $Bel(\varnothing) = 0$ as well as $Bel(\Theta) = 1$. Moreover, the associated *plausibility function* is the mapping $Pl : 2^\Theta \to [0, 1]$ defined by: $Pl(A) = \sum_{B \cap A \neq \varnothing} m(B)$ for each $A \subseteq \Theta$. Observe that $Bel(A) \leq Pl(A)$ and $Pl(A) = 1 - Bel(\Theta \setminus A)$ holds for every $A \subseteq \Theta$.

We recall Dempster's rule of combination, which merges two independent mass functions into a joint mass function, and the pignistic probability transform, which converts a mass function into a probability distribution.

Definition 3 (Dempster's Rule of Combination). *Let m_1 and m_2 be two independent mass functions on a frame of discernment Θ. Their Dempster combination $m = m_1 \oplus m_2$ is defined by*

$$m(C) = \frac{1}{1 - K} \sum_{\substack{A, B \subseteq \Theta \\ A \cap B = C}} m_1(A) m_2(B), \tag{2}$$

where $K < 1$ is the so-called conflict coefficient, and specified as follows:

$$K = \sum_{\substack{A, B \subseteq \Theta \\ A \cap B = \varnothing}} m_1(A) m_2(B). \tag{3}$$

Definition 4 (Pignistic Probability Transform). *Let m be a mass function on Θ. The* pignistic probability transform *associated with m is the mapping* $\mathrm{PPT} : \Theta \to [0, 1]$ *defined, for each $x \in \Theta$, by*

$$\mathrm{PPT}(x) = \sum_{\substack{A \subseteq \Theta \\ x \in A}} \frac{m(A)}{|A|\big(1 - m(\varnothing)\big)}. \tag{4}$$

4 Our Class Expression Learning Approach: LYRA

We formulate the CEL problem as a sequential planning task, where DRL agents iteratively construct a class expression by selecting concepts from the vocabulary of the knowledge base $\mathcal{K}$. To ensure *ontology awareness*, we model this process as a Dec-POMDP (cf. Definition 2). We next outline the environment design and our novel BFT policy network that predicts a distribution of actions used to update the class expression being generated. Throughout this section, we provide running examples to highlight the functioning of LYRA (e.g., see Figure 1 and Figure 3).

4.1 Class Expression Learning as a Dec-POMDP

Let E^+, E^- be a learning problem over a KB $\mathcal{K}$. In the generation or optimization of class expressions, each step involves one or more of the following *action categories*, depending on the agents' decisions: Adding (1) a concept, (2) an object property, (3) a data property, (4) cardinality constraints, (5) a set operator, or (6) a quantifier operator. The structure of the resulting expression is determined solely by the KB and the specific modifications applied. We next outline individual components of CEL as a Dec-POMDP. Throughout this discussion, we assume $\mathcal{K}$ as our KB in the CEL instance.

Agents (N): LYRA consists of two agents that share a common feature extractor neural network part for stable learning. Additionally, each agent $i \in \{1, 2\}$ has independent decision-making neural network heads acting as a policy network $\pi_{i,\psi}$. The necessity of multi-agents arises due to our reliance on the DS rule in Equation (2) which requires evidence sources to be independent. The DS rule is applied to combine the decisions (beliefs) of individual agents. If the combined beliefs are a singleton, agents perform this joint action, and otherwise, we perform an intermediate step of transforming the masses using Equation 4. This transformation is necessary to obtain standard probability distribution over singleton hypotheses for training neural networks. LYRA uses two agents for the sake of simplicity and ease of notation. However, since the DS rule is associative and commutative, the proposed framework can be readily extended to n agents through iterative pairwise combination, i.e., $m = m_1 \oplus m_2 \oplus \cdots \oplus m_n$. Although additional agents can be incorporated, doing so would unnecessarily increase the computational complexity.

State space (S): We represent S as a set of pairs (ζ, t), where ζ is a valid class expression over the vocabulary of $\mathcal{K}$ and t denotes the number of steps taken to reach ζ. The termination (goal) state is specified as either when a predefined parameter max_t is reached or the best predicted action is $stop$. We will later specify both max_t and $stop$.

Initial state (s_0): The initial expression is generated by a one-hop random walk. Beginning with a positive example $p \in E^+$, we randomly include ABox assertions involving p as a subject (i.e., concept assertions as well as direct object and data properties) and then terminate the walk.

Action space ($\mathbb{A}$): Actions by both agents are taken according to their policy $\pi_{i,\psi}$ for $i \in \{1, 2\}$. The generation or optimization of an expression ζ is seen as a sequential decision process, each step involving one or more of the following actions:

1. *Concept addition:* Agents define a concept from those available over the given KB $\mathcal{K}$. The agents also select a set operator simultaneously (i.e., $\sqcap$ or $\sqcup$) to combine the new concept with the existing one.
2. *Object property addition:* Agents can also associate an object property with the current expression ζ using a quantifier operator ($\exists, \forall, \leq, \geq$), which is selected by the agents in parallel. Then, a learned cardinality is added simultaneously if agents choose a cardinality restriction.
3. *Data property addition:* Agents can include a data property and join it with ζ using a selected set operator in parallel as follows. (1) *Numeric data* are included following [8] by deciding on the split value such that it maximizes the information gain. (2) *Other types of data* (e.g., *Boolean*, *String*, or *Geospatial*), the value found in the sampled individual from the positive example $p \in E^+$ is added.
4. *Stopping generation process:* Agents can choose to terminate the episode upon determining that the target expression length has been achieved. This is determined by $stop$ action flag. This action helps to predict the expression length that has proven to be beneficial in CEL [10].

Formally, the **action space** $\mathbb{A}$ for the agent $i \in \{1, 2\}$ is defined as a pair $\mathbb{A}_i = (\mathbb{A}_i^{MC}, \mathbb{A}_i^{BPA})$, where $\mathbb{A}_i^{MC}$ is a finite set of all possible, mutually exclusive hypotheses, and $\mathbb{A}_i^{BPA}$ denotes the set of belief degrees over all subsets of $\mathbb{A}_i^{MC}$. In the CEL context, hypotheses in $\mathbb{A}_i^{MC}$ correspond to the elements in the vocabulary of $\mathcal{K}$ (e.g., sets of concepts and operators), as demonstrated in the following example.

Example 1 (Construction of Mass Functions). We illustrate the multi-categorical action distribution ($\mathbb{A}_1^{MC}$) for agent 1. Assume that the knowledge base contains only two atomic classes, namely $\{\text{Molecule}, \text{Bond}\}$, such that the frame of discernment is given by $\Theta = \{\text{Molecule}, \text{Bond}\}$. The associated power set is therefore $2^\Theta = \{\emptyset, \{\text{Molecule}\}, \{\text{Bond}\}, \{\text{Molecule}, \text{Bond}\}\}$. Subsequently, the action $\mathbb{A}_1^{BPA}$ for agent 1 assigns BPA to each subset of 2^Θ as summarized in Figure 1. Observe that $m(\varnothing) = 0$ and $\sum_{A \subseteq \Theta} m(A) = 1$ as discussed in the background section. A small mass assigned to the full frame, i.e., $m(\Theta) = 0.15$, indicates a low degree of epistemic ignorance of the agent, who commits most of its belief to *more specific* hypotheses rather than to the entire frame of discernment.

As Example 1 illustrates, the power set for a frame of discernment can be large and thus problematic for KBs with large vocabularies. In $\mathcal{ALCQ}(\mathcal{D})$, we can divide the candidate elements in a frame of discernment into five distinct classes: concepts, roles, set operators, quantifier restrictions, and data type properties. Naturally, our DL admits only three set operators ($\sqcap, \sqcup, \neg$) and four restrictions ($\exists, \forall, \leq, \geq$). Thus, the size of the frame of discernment is mainly determined by the number of concepts, roles and data properties in the vocabulary. To mitigate the aforementioned issue, each agent selects only a subset of each *vocabulary class* at a given time step containing only k elements of these three problematic classes. Subsequently, by training the neural network parameters, the agent learns the *best* part of the vocabulary to be selected at each time step conditioned on the current class expression.

To illustrate the validity of the sampling mechanism, we define *focal elements* (FE) with respect to an agent as the set of vocabulary elements that are assigned non-zero

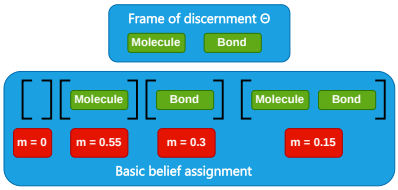

Fig. 1: Illustration of an agent's mass function (see Example 1). Green boxes are hypotheses obtained from $\mathbb{A}_1^{MC}$ and red boxes are BPAs obtained from $\mathbb{A}_1^{BPA}$.

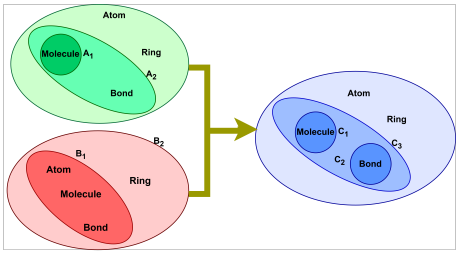

Fig. 2: Combination of masses after sampling four atomic concepts from the vocabulary (see Example 2).

value under the mass function m by that agent. Then, the core $C_f(m)$ is defined as the union of all the sets in FE(m) for the mass function m of the agent:

$$C_f(m) = \bigcup_{A \in \text{FE}(m)} A \tag{5}$$

The sampling mechanism is valid iff the cores of the agent's mass functions have a non-empty intersection. In our setting with two agents, this yields $C_f(m_1) \cap C_f(m_2) \neq \emptyset$, where m_i is the mass function for agent $i \in \{1, 2\}$.

Example 2. Consider $\Theta = \{\text{Molecule}, \text{Atom}, \text{Bond}, \text{Ring}\}$ as our frame. Let agent 1's BPA be m_1 where $m_1(\{\text{Molecule}\}) = 0.7$, and $m_1(\{\text{Molecule}, \text{Bond}\}) = 0.3$. The corresponding FEs for agent 1 thus contains two sets: $A_1 = \{\text{Molecule}\}$ and $A_2 = \{\text{Molecule}, \text{Bond}\}$. In particular, agent 1's belief in the remaining hypotheses in 2^Θ is zero. Now, consider the agent 2's BPA (m_2) such that $m_2(\{\text{Atom}, \text{Molecule}, \text{Bond}\}) = 0.6$ and $m_2(\Theta) = 0.4$, with corresponding FEs: $B_1 = \{\text{Atom}, \text{Molecule}, \text{Bond}\}$ and $B_2 = \Theta$. Since their cores overlap, the corresponding masses can be combined using the DS rule of Equation 2 (with the value of K calculated to be 0 using Equation 3). This yields $m(\{\text{Molecule}\}) = 0.7 \times 0.6 + 0.7 \times 0.4 = 0.70$, $m(\{\text{Bond}\}) = 0$, and $m(\{\text{Molecule}, \text{Bond}\}) = 0.3 \times 0.6 + 0.3 \times 0.4 = 0.30$. Consequently, the best action at this step is to choose and add the concept "Molecule". For illustration, see Figure 2.

Joint observations (Ω): For agent $i \in \{1, 2\}$, we define its partial observation as: $\Omega_i = (ST_i(\zeta), t)$ where $ST_i : \zeta \to \mathbb{R}^d$ is the class expression encoder. We used a pretrained *Sentence-BERT* [22] for semantic-aware encoding of the class expression.

State transition function (P): At each time step, the joint action $\mathbf{a}^t = (a_1, a_2)$ of both agents guides a transition from the current state s^t to s^{t+1}. This transition function for the agent $i \in \{1, 2\}$ is governed by their policy network $\pi_{i,\psi}$.

Immediate rewards (R): The reward function is designed to encourage the agents' collaboration. Also, CEL can be considered as a multi-objective optimization problem. For example, minimizing expression length while maximizing $F1$ score and ensuring

expression validity. For each time step t, R_t is defined as:

$$R_t(s^t, \mathbf{a^t}) = F1 + \frac{1}{L_c}, \tag{6}$$

where $F1$ is the harmonic mean of precision and recall and L_c is the length of the expression ζ. The $F1$ metric is selected due to the critical need for balanced precision and recall in high-stakes application domains.

4.2 Integration of Belief Function Theory

We argue that the integration of BFT significantly enhances the expressiveness of agent actions and supports ambiguity-averse strategies, which are particularly advantageous in partially observable environments. Incorporating BFT into the MADRL yields the following key improvements:

Agents decision-making under epistemic uncertainty. BFT generalizes Bayesian probability by allowing mass on sets of hypotheses rather than forcing a point-probability on each singleton. For instance, suppose an agent's mass function over the binary frame $\{A, \neg A\}$ is: $m(A) = 0.8, m(\neg A) = 0.1$. In BFT, the belief in a hypothesis and its negation do not necessarily sum to 1, as there might be mass assigned to the frame of discernment Θ which represents ignorance. In this case $m(\Theta) = 0.1$, it means that agent has at least an 80% justified confidence that hypothesis A is true, based on the evidence it committed. Nevertheless, confidence could rise up to 90%.

Agents conflicting policies. Conflicting policies arise when agents' individual optimal strategies lead to suboptimal or even negative outcomes for the overall system or other agents. This can happen because agents might pursue conflicting goals, lack coordination, or misinterpret each other's actions. BFT offers a rich family of fusion rules to handle conflict, each inducing different agent-level behaviors. For example, Dempster's [26] rule normalizes away all conflicting mass, which biases agents to abandon conflicting proposals. Yager's rule [31] reallocates conflict mass onto the whole frame of discernment, effectively increasing collective ignorance and encouraging exploration under high disagreement.

Opaque decision-making in policy networks. Diagnosing failures in black-box policy networks is challenging: softmax logits over categorical actions do not distinguish whether a choice reflects strong supporting evidence, conflicting cues, or genuine uncertainty, hence offering limited interpretability. In contrast, the BFT framework provides several diagnostic signals. For example, the width of the uncertainty interval: $[Bel(A), Pl(A)]$ quantifies epistemic uncertainty about the hypothesis A. A large interval indicates that the agent's evidence neither confirms nor falsifies A. Another example would be measuring the conflict (Equation 3). A higher conflict can indicate an error in the communication among agents.

Example 3 (Belief and Plausibility gap). Consider the hypothesis $\{\text{Molecule}\}$ from Example 1. We calculate its belief (see Sec 3.2) as $Bel(\{\text{Molecule}\}) = m(\{\text{Molecule}\}) =$

0.55, and plausibility as $Pl(\{\text{Molecule}\}) = m(\{\text{Molecule}\}) + m(\Theta) = 0.55 + 0.15 = 0.7$. Hence, the uncertainty interval associated with this hypothesis is: $[Bel(\{\text{Molecule}\}), Pl(\{\text{Molecule}\})] = [0.55, 0.7]$.

As Example 3 highlights, the uncertainty gap provides an additional layer of interpretability regarding the system's confidence in its reasoning process. A smaller gap indicates improved certainty and stronger convergence of the system's understanding, whereas a larger gap reflects greater epistemic uncertainty.

Integrating BFT into MADRL Agents The integration requires modifications to certain foundational definitions. The necessary adaptations are outlined below.

Beliefs combination. This step is crucial for making the final decision. Agents aggregate their information to determine the best action to take for generating ζ. After each agent successfully constructs its own mass function per each dimension, all masses are then unified in one piece of evidence using the combination rule from Equation 2.

Since the problem of generating DL class expression is formulated as a planning task under the MADRL framework, calculating the trajectory of optimal actions is our core research objective. However, the existing MADRL algorithms are not compatible with BFT framework. Hence, we formulate our algorithm that is derived from the *Multi-Agent Proximal Policy Optimization* (MAPPO) [34] algorithm. Our algorithm consists of two distinct neural networks, corresponding to agents $i \in \{1, 2\}$. The policy network $\pi_{i,\psi} \colon \Omega_i^t \rightarrow 2^\Theta \times [0, 1]$ maps agent i's observations at each time-step t to a pair consisting of (i) a multi-categorical distribution, producing $\mathbb{A}_i^{MC}$, and (ii) the mean and standard deviation for $\mathbb{A}_i^{BPA}$. In our policy, $\mathbb{A}_i^{BPA}$ must be conditioned and dependent on $\mathbb{A}_i^{MC}$. To achieve this, we implemented a simple trick of gradient flow during training as follows:

$$\pi_{i,\psi} \coloneqq \big(\mathcal{M}_C(s^t, \mathbb{A}_i^{MC}),\ \ \mathcal{N}_L(\mathbb{A}_i^{BPA}; \mu_\psi(s^t, \mathbb{A}_i^{MC}), \sigma_\psi(s^t, \mathbb{A}_i^{MC}))\big)$$

where $\mathcal{M}_C$ is the multicategorical, and $\mathcal{N}_L$ is the logistic normal distribution. The critic network $V_\phi : \Omega_i \rightarrow \mathbb{R}$ maps agent i's observation state to a scalar estimate of expected cumulative discounted return: $V_\phi(s_0) \approx \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_i^t | s_0]$. We use $\gamma = 0.9$ to discount rewards over time for balancing between immediate feedback and long-term return estimation. The objective of $\pi_{i,\psi}$ is to maximize: $L(\psi) = \mathbb{E}_{(s^t, a_i^t) \sim \pi_{i,\psi}}[\hat{r} \hat{A}^t_{\pi_{i,\psi}}(s^t, a_i^t)]$, where $\hat{r}$ is the ratio of the new policy over the old policy, which is denoted by:

$$\hat{r} = \frac{\pi_{i,\psi}^t(s^t, a_i^t)}{\pi_{i,\psi}^{t-1}(s^t, a_i^t)}. \tag{7}$$

$\hat{A}^t_{\pi_{i,\psi}}$ is the relative value of an action compared to the expected value at time step t, formally calculated using the *General Advantage Estimation* (GAE) [25] as $\hat{A}^t_{\pi_{i,\psi}} = \sum_{t=0}^{\infty} (\gamma\lambda)^l \delta^{V_\phi}_{t+l}$. γ is the discount value, λ is a smoothing parameter for variance reduction, and $\delta_t = R(s^t, \mathbf{a}^t) + \gamma V_\phi(s^{t+1}) - V_\phi(s^t)$. The policy objective is different from the traditional PPO algorithm in many aspects. Policy of PPO is a purely probabilistic function that outputs a single point of probability, and it uses a clipping function directly

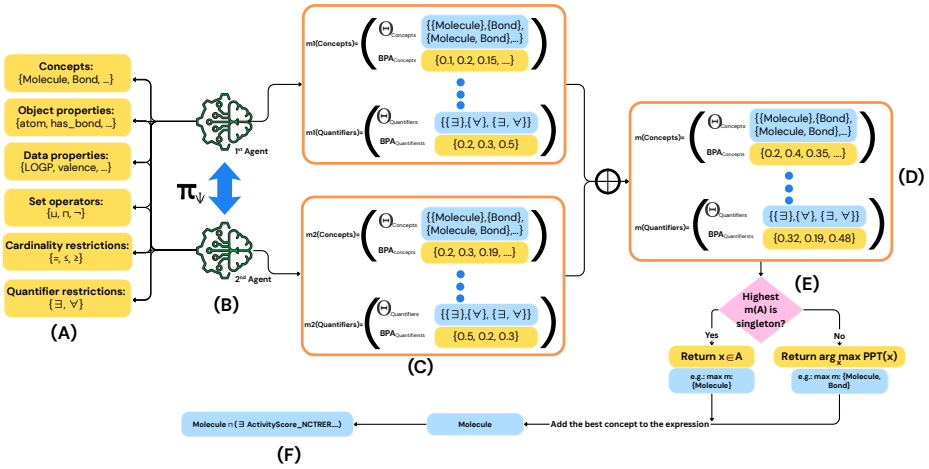

Fig. 3: An architecture overview of LYRA on the running example from the NCTRER dataset. (A) denotes the extraction of vocabulary from the KB. (B) describes agents' sampling from this vocabulary pool to construct a frame of discernment Θ and assign beliefs for each subset of Θ. (C) shows the mass function, defined for each set of vocabularies for each agent. (D) describes the combination of mass functions using DS rule (Equation 2). (E) denotes returning the hypothesis with maximum beliefs if the subset s_i is singleton; otherwise, PPT is applied on singleton subsets (Equation 4). Finally, (F) joins the selected elements (i.e., concepts, data properties, etc.) to the current expression. Steps from (B) to (E) are repeated until a termination criterion is met.

on the gradients. This results in zero gradients when $\hat{r}$ in Equation 7 falls outside the clipping range, which leads to the exclusion of those samples from training, resulting in a loss of information. However, despite its empirical effectiveness, it remains a conceptual limitation in PPO. Nevertheless, clipping gradients in the BFT framework can lead to severe problems. For example, removing the clipping function ensures that every piece of evidence is fully integrated into the mass update. This is crucial for the efficiency of combining beliefs (Equation 2). Finally, we incorporated the entropy in our objective function to prevent premature convergence. We use *Nguyen's entropy* [21], which is determined by $\mathcal{H}_N = -\sum_{A \subseteq \Theta} m(A) log_2(m(A))$, and LYRA's policy objective function is then defined by:

$$L(\psi) = \mathbb{E}_{(s^t, a_i^t) \sim \pi_{i,\psi}} [r A^t_{\pi_{i,\psi}}(s^t, a_i^t) + \mathcal{H}_N]. \tag{8}$$

Incorporating Nguyen's entropy into the policy objective encourages agents for sufficient exploration. By penalizing beliefs that are either overly confident (low entropy) or excessively vague (high entropy), it effectively mitigates premature convergence and fosters robust exploration in the presence of ambiguity. Let $G_t = \sum_{t=0}^{T} \gamma^t r_i^t$ be the total discounted reward. The critic network objective in our approach is defined simply by the mean squared error between critic prediction and the return (i.e., $L(\phi) = \mathbb{E}[(V_\phi(s_t) - G_t)^2]$).

Table 2: Overview of the benchmark datasets.

Dataset	N_I	**Axioms**	N_C	P_O	P_D
DBPedia	4.23M+	850 M+	768+	1K+	1.7K+
Carcinogenesis	22,372	74,566	142	4	15
Animals	20	138	39	2	1
Hepatitis	6,812	79,935	14	5	12
Mammographic	975	6,808	19	3	4
Mutagenesis	14,145	62,066	86	5	6
NCTRER	10,209	103,070	37	9	5
Suramin	2,979	13,506	46	3	1

5 Experimental Evaluation

We conducted experiments on eight datasets with five different random seeds for each dataset (resulting in a total of 40 experiments). We are aiming to answer the following research questions:

Q_1: Can LYRA outperform state-of-the-art approaches in efficiency and generating shorter class expressions while achieving a higher $F1$ score?
Q_2: Does LYRA enhance interpretability in the decision-making process?
Q_3: Does BFT enhance the capabilities of DRL agents in the CEL setting?

5.1 Experiment setup

Implementation details. We used the PettingZoo [27] framework to implement the logic of our Dec-POMDP. Moreover, the action and observation spaces for each agent were represented using Gymnasium [28].

Datasets. For our experimental evaluation, we use datasets from the SML-Bench [30], which provides a diverse collection of real-world datasets primarily focused on biomedical and molecular prediction tasks (see Table 2). For example, the *Carcinogenesis* dataset predicts the carcinogenic potential of chemical compounds. The *Animals* dataset is used to classify different types of animals. The *Hepatitis* dataset is used to classify hepatitis types based on patients' clinical data, while the *Mammographic* dataset assesses the severity of breast cancer from screening results. The *Mutagenesis* dataset predicts the mutagenic properties of molecular structures. The *NCTRER* dataset evaluates estrogen receptor binding activity of molecules, and the *Suramin* dataset identifies predictive descriptors for suramin analogues in cancer treatment. Additionally, we evaluate LYRA on *DBpedia* [18] a large-scale, general-purpose KG extracted from Wikipedia. DBpedia is the largest dataset used in our evaluation.

Hardware and software. All experiments were conducted on a server with AMD EPYC 7543P 32-Core Processor, 503.5GB RAM, running on Debian 12.10, Python 3.12.7, Pytorch 2.7.0, and random seeds for each running trial in experiments for reproducibility.

Baselines. We compare LYRA with the following state-of-the-art approaches: (I.) EVOLEARNER [8] uses a bottom-up approach and integrates random walks with evolutionary algorithms. (II.) DRILL [6] is a DRL-based approach that utilizes DQN and downward refinement operator. (III.) CELOE [16] is a heuristic-driven refinement operator based approach that iteratively specializes candidate concepts starting from the most general expression—using the F1-measure to guide the search. These baselines were chosen to represent distinct approaches, with one best performing candidate selected from each paradigm to ensure a diverse, robust, and fair evaluation. All approaches have the same DL expressiveness provided by their implementation in *Ontolearn* [5].

5.2 Q_1: LYRA's efficiency

For addressing Q_1, we validated the effectiveness of our approach of generating class expressions with higher $F1$ scores and shorter lengths, assisting easier interpretation by domain experts. Our reward function (Equation 6), assisted in this multi-objective optimization. These experiments show significant consistency and scalability in LYRA's generated expressions across KBs of varying sizes. Experiments were conducted 5 times for each model with predefined random seeds to ensure reproducibility across trials. The average $F1$ and concept length are recorded in Table 3. Moreover, each attribute in our approach is learnable. For instance, unlike EVOLEARNER, which randomly selects cardinality within a predefined range, LYRA learns the optimal cardinality directly from experience. In general, approaches based on DRL often outperform those based on Evolutionary Algorithms (EA) by using feedback from both successful and unsuccessful experiences, whereas EA relies solely on the fittest outcomes. LYRA outperforms the state-of-the-art on 7 out of 8 benchmark datasets across all experiments and matches the performance of the best-performing methods on the remaining single dataset, thereby providing a clear answer to Q_1.

Statistical Hypothesis Testing. To evaluate whether the performance differences among LYRA and the baseline approaches were statistically significant, we performed pairwise one-sided Wilcoxon signed-rank tests using the best F1-score obtained in each of the independent runs with different random seeds. The pairing was based on identical random seeds applied across all models. All p-values were adjusted using the Holm correction for multiple comparisons at $\alpha = 0.05$. For each pairwise comparison, the null hypothesis H_0 states that there is no difference in F1-score performance between LYRA and the corresponding baseline approach, while the alternative hypothesis H_1 states that LYRA achieves a higher F1-score than the baseline. Results demonstrated that LYRA significantly outperformed all baseline approaches. In seven out of eight cases, H_0 was rejected at the 5% significance level, confirming that the superior performance of LYRA over all competing baselines is statistically significant.

Complexity. LYRA samples candidate vocabulary using a combination of policy networks at each time step as demonstrated in our running example (Figure 3). This sampling approach makes LYRA independent of the KB size and renders it dependent on the vocabulary size alone. This implies a linear computational complexity in the size of the vocabulary in the KB, thus making LYRA significantly scalable compared to other approaches.

Table 3: Comparison of F1 score and concept length across datasets. ‡ indicates a runtime error encountered when using the official implementation. LYRA's objective is to find the concept with the highest F1 score with the shortest possible length. Performance and concept length results are reported as mean values over five random seeds, with standard deviations indicated after the ± symbol.

	Performance				**Concept Length**			
Dataset	**LYRA**	**EvoLearner**	**Drill**	**CELOE**	**LYRA**	**EvoLearner**	**Drill**	**CELOE**
DBPedia	**0.98 ± 0.0**	‡	0.85±0.0	‡	8.0±4.5	‡	1.0±0.0	‡
NCTRER	0.97±0.01	**1.00**±0	0.45±0.1	0.80±0.0	5.0±1.5	8.5±4.5	1.0±0.0	3.0±0.0
Mutagenesis	**1.00**±0.0	**1.00**±0.0	0.45±0.0	0.45±0.0	5.33 ± 0.47	3.00 ± 0.0	1.0 ± 0.0	1.0 ± 0.0
Suramin	**0.67**±0.02	0.60±0.1	0.56±0	0.55±0.0	4.0 ± 1.63	6 ±1.5	1.0 ± 0.0	1.0 ± 0.0
Hepatitis	**0.66**±0.0	0.53±0.1	0.56±0	0.58±0.0	3.0 ± 0.0	3.67 ± 0.47	1.0 ± 0.0	1.0 ± 0.0
Carcinogenesis	**0.72**±0.2	0.69±0	0.70±0	0.70±0.0	4.0 ± 0.0	5.5 ± 0.47	1.0 ± 0.0	1.0 ± 0.0
Mammographic	**0.78**±0.0	0.72±0.05	0.63±0	0.63±0.0	3.0 ± 0.0	3.0 ± 0.0	1.0 ± 0.0	1.0 ± 0.0
Animals	**1.00**±0.0	‡	0.80±0.0	0.80±0.0	1.0±1.0	‡	1.0±0.0	1.0±0.0

Limitations. LYRA offers the advantage of generating creative class expressions independently of the knowledge base size. However, the sampling process requires an initial exploration period before producing meaningful expressions. Its convergence rate follows stochastic policy gradient methods, with complexity $O(1/\sqrt{T})$, where T denotes the total number of training steps [19]. An early stopping criterion can be introduced once domain-satisfactory class expressions are obtained, for example if the concept reaches a desirable F1 score.

5.3 Q_2: LYRA's additional layer of interpretability

BFT framework provides further tools for understanding the agents decision-making process. For example, monitoring the belief interval of the agents (i.e. $[Bel(A), Pl(A)]$), provides more indications of epistemic uncertainty regarding the hypothesis A. A narrow interval with low values for belief and plausibility gap (i.e., $Bel(A) \approx Pl(A)$) can be interpreted as the agents having strong evidence *for* the hypothesis A, effectively indicating a consensus that the optimal action does not lie within A. In contrast, probabilistic agents lacking such interval semantics would merely reflect low probability for A, which is ambiguous—it conflates active rejection with lack of evidence. We have tracked the belief interval for all experiments (Figure 4). The results demonstrate a significant contraction of the belief interval, indicating reduced epistemic uncertainty. This reflects belief convergence under accumulating evidence and suggests enhanced decision stability in Dec-POMDP. Belief intervals offer a quantitative description of uncertainty to the agents. This is useful in interpreting how the entire system behaves and reveals how much the system was successful in modeling the uncertainty. Thus, Q_2 is answered affirmatively, as LYRA improves interpretability by explicitly modeling epistemic uncertainty.

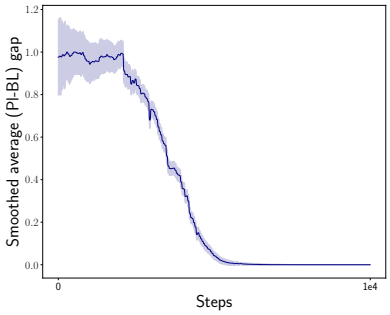

Fig. 4: Aggregated belief intervals across all experiments indicating the overall performance.

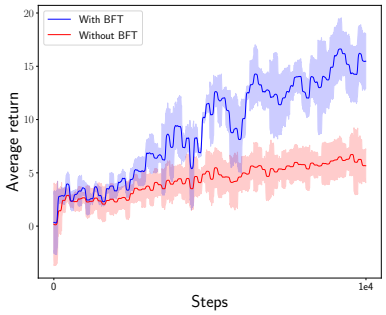

Fig. 5: Overview of averaged cumulative rewards for LYRA with and without BFT across five different random seeds.

5.4 Q_3: Ablation Study

To address Q_3, we assess the contribution of BFT to DRL-based planning under high rates of missing information through a controlled ablation study. In particular, we perform CEL using two variants: (1) a BFT-enhanced DRL agent, and (2) a PPO agent. We kept the network architectures the same for both variants. Noise and incomplete information are relevant problems in CEL; for example, *DBPedia*[18] has 68% of missing information about the *place of birth* property. The BFT-enhanced model consistently outperforms other ablated variant, which empirically confirms that the BFT component contributes positively to overall performance, thus answering Q_3. Results for all KBs and five different random seeds are summarized in Figure 5.

6 Conclusion and Future Work

We develop LYRA by integrating BFT with a modified version of a multi-agent version of PPO. An evaluation of experiments on eight datasets with five different random seeds reveals that our approach achieves higher or comparable performance when compared to the state of the art. Furthermore, we provide additional tools to understand the decision-making process for interpretability and model debugging. Interestingly, our approach generates novel expressions, as it is not limited to local logical components such as searching-based approaches or to training datasets as in neural-based approaches. Future work will investigate the effect of combination rules on the agents' behavior. Moreover, we aim to extend LYRA to support more expressive DLs and to investigate additional models of uncertainty-aware policies for CEL.

Supplemental Material Statement: The source code, datasets, logs from our experiments, as well as instructions to run our approach for reproducing the results are available in our GitHub https://github.com/lyrdl/LYRA.

Declaration on Generative AI: The authors have not employed Generative AI tools in this work.

Acknowledgements: This work was funded by the European Union's Horizon Europe programme (Marie Skłodowska-Curie Grant No. 101120449), the Ministry of Culture and Science of North Rhine-Westphalia (MKW NRW) through the projects SAIL (NW21-059D) and WHALE (LFN 1-04, Lamarr Fellow Network), and the German Federal Ministry of Research, Technology and Space (BMFTR) within the project KI-Akademie OWL under the grant no 16IS24057B, and European Union's Horizon Europe research and innovation programme under grant agreement No 101070305, and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation): TRR 318/3 2026 – 438445824. Authors are grateful to Dr. Pamela Heidi Douglas for language editing the manuscript.

References

1. Baader, F.: The description logic handbook: Theory, implementation and applications. Cambridge university press (2003)
2. Baader, F., Horrocks, I., Lutz, C., Sattler, U.: An Introduction to Description Logic. Cambridge University Press (2017), http://www.cambridge.org/de/academic/subjects/computer-science/knowledge-management-databases-and-data-mining/introduction-description-logic?format=PB#17zVGeWD2TZUeu6s.97
3. Baci, A., Heindorf, S.: Accelerating concept learning via sampling. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. p. 3733–3737. CIKM '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3583780.3615158, https://doi.org/10.1145/3583780.3615158
4. Bühmann, L., Lehmann, J., Westphal, P., Bin, S.: Dl-learner structured machine learning on semantic web data. In: Companion Proceedings of the The Web Conference 2018. p. 467–471. WWW '18, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (2018). https://doi.org/10.1145/3184558.3186235, https://doi.org/10.1145/3184558.3186235
5. Demir, C., Baci, A., Kouagou, N.J., Sieger, L.N., Heindorf, S., Bin, S., Blübaum, L., Bigerl, A., Ngonga Ngomo, A.C.: Ontolearn—A Framework for Large-scale OWL Class Expression Learning in Python. Journal of Machine Learning Research, JMLR (2025), https://papers.dice-research.org/2025/JMLR_Ontolearn/Ontolearn_public.pdf
6. Demir, C., Ngomo, A.C.N.: Neuro-symbolic class expression learning. In: The 32nd International Joint Conference on Artificial Intelligence, IJCAI 2023 (2023), https://papers.dice-research.org/2023/IJCAI_DRILL/public.pdf
7. Gao, Y., Zhang, X., Sun, Z., Chandak, P., Bu, J., Wang, H.: Precision adverse drug reactions prediction with heterogeneous graph neural network. Advanced Science **12**(4), 2404671 (2025)
8. Heindorf, S., Blübaum, L., Düsterhus, N., Werner, T., Golani Nandkumar, V., Demir, C., Ngonga Ngomo, A.C.: Evolearner: Learning description logics with evolutionary algorithms. In: WWW. pp. 818–828. ACM (2022). https://doi.org/10.1145/3485447.3511925, https://arxiv.org/abs/2111.04879

9. Jäger, J., Helfenstein, F., Scharf, F.: Bring Color to Deep Q-Networks: Limitations and Improvements of DQN Leading to Rainbow DQN, pp. 135–149. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-41188-6_12, https://doi.org/10.1007/978-3-030-41188-6_12
10. Kouagou, N.J., Heindorf, S., Demir, C., Ngomo, N.A.C.: Learning concept lengths accelerates concept learning in alc. In: ESWC. Springer (2022), http://arxiv.org/abs/2107.04911
11. Kouagou, N.J., Heindorf, S., Demir, C., Ngonga Ngomo, A.C.: Neural class expression synthesis in alchiq(d). In: Koutra, D., Plant, C., Rodriguez, M.G., Baralis, E., Bonchi, F. (eds.) "Machine Learning and Knowledge Discovery in Databases". pp. –. Springer Nature Switzerland, Cham (2023), https://papers.dice-research.org/2023/ECML_NCES2/NCES2_public.pdf
12. Kouagou, N.J., Heindorf, S., Demir, C., Ngonga Ngomo, A.C.: ROCES: Robust Class Expression Synthesis in Description Logics via Iterative Sampling. In: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24. International Joint Conferences on Artificial Intelligence Organization (2024), https://papers.dice-research.org/2024/IJCAI_ROCES/public.pdf, main Track
13. Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., Teh, Y.W.: Set transformer: A framework for attention-based permutation-invariant neural networks. In: International Conference on Machine Learning (ICML). pp. 3744–3753. PMLR (2019)
14. Lehmann, J.: Hybrid learning of ontology classes. In: Proc. of the 5th Int. Conference on Machine Learning and Data Mining MLDM. Lecture Notes in Computer Science, vol. 4571, pp. 883–898. Springer (2007), http://dx.doi.org/10.1007/978-3-540-73499-4_66
15. Lehmann, J.: Learning OWL class expressions, vol. 22. IOS Press (2010)
16. Lehmann, J., Auer, S., Bühmann, L., Tramp, S.: Class expression learning for ontology engineering. Journal of Web Semantics **9**(1), 71–81 (2011). https://doi.org/https://doi.org/10.1016/j.websem.2011.01.001, https://www.sciencedirect.com/science/article/pii/S1570826811000023
17. Lehmann, J., Hitzler, P.: Concept learning in description logics using refinement operators. Mach. Learn. **78**(1–2), 203–250 (Jan 2010). https://doi.org/10.1007/s10994-009-5146-2, https://doi.org/10.1007/s10994-009-5146-2
18. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., Bizer, C.: Dbpedia - a large-scale, multilingual knowledge base extracted from wikipedia. Semantic Web Journal **6** (01 2014). https://doi.org/10.3233/SW-140134
19. Liu, B., Cai, Q., Yang, Z., Wang, Z.: Neural proximal/trust region policy optimization attains globally optimal policy. Curran Associates Inc., Red Hook, NY, USA (2019)
20. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529–533 (Feb 2015), http://dx.doi.org/10.1038/nature14236
21. Nguyen, H.T.: On entropy of random sets and possibility distributions. The Analysis of Fuzzy Information **1**, 145–156 (1987)
22. Reimers, N., Gurevych, I.: Making monolingual sentence embeddings multilingual using knowledge distillation. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics (11 2020), https://arxiv.org/abs/2004.09813
23. Rizzo, G., Fanizzi, N., d'Amato, C.: Class expression induction as concept space exploration: From dl-foil to dl-focl. Future Generation Computer Systems **108**, 256–272 (2020). https://doi.org/https://doi.org/10.1016/j.future.2020.02.071, https://www.sciencedirect.com/science/article/pii/S0167739X19303991

24. Rudolph, S.: Foundations of Description Logics, pp. 76–136. Springer Berlin Heidelberg, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23032-5_2, https://doi.org/10.1007/978-3-642-23032-5_2"
25. Schulman, J., Moritz, P., Levine, S., Jordan, M.I., Abbeel, P.: High-dimensional continuous control using generalized advantage estimation. In: Bengio, Y., LeCun, Y. (eds.) 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings (2016), http://arxiv.org/abs/1506.02438
26. Shafer, G.: Dempster's rule of combination. International Journal of Approximate Reasoning **79**, 26–40 (2016). https://doi.org/https://doi.org/10.1016/j.ijar.2015.12.009, https://www.sciencedirect.com/science/article/pii/S0888613X15001978, 40 years of Research on Dempster-Shafer Theory
27. Terry, J., Black, B., Grammel, N., Jayakumar, M., Hari, A., Sullivan, R., Santos, L.S., Dieffendahl, C., Horsch, C., Perez-Vicente, R., et al.: Pettingzoo: Gym for multi-agent reinforcement learning. Advances in Neural Information Processing Systems **34**, 15032–15043 (2021)
28. Towers, M., Kwiatkowski, A., Terry, J., Balis, J.U., De Cola, G., Deleu, T., Goulão, M., Kallinteris, A., Krimmel, M., KG, A., et al.: Gymnasium: A standard interface for reinforcement learning environments. arXiv preprint arXiv:2407.17032 (2024)
29. Vikhar, P.A.: Evolutionary algorithms: A critical review and its future prospects. In: 2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC). pp. 261–265 (2016). https://doi.org/10.1109/ICGTSPICC.2016.7955308
30. Westphal, P., Bühmann, L., Bin, S., Jabeen, H., Lehmann, J.: Sml-bench – a benchmarking framework for structured machine learning. Semantic Web Journal (2018), http://jens-lehmann.org/files/2018/swj_sml_bench.pdf
31. Yager, R.R.: On the dempster-shafer framework and new combination rules. Information Sciences **41**(2), 93–137 (1987). https://doi.org/https://doi.org/10.1016/0020-0255(87)90007-7, https://www.sciencedirect.com/science/article/pii/0020025587900077
32. Yager, R.R., Liu, L.: Classic Works of the Dempster-Shafer Theory of Belief Functions. Springer Publishing Company, Incorporated, 1st edn. (2010)
33. You, W.: Machine learning based structural design and performance prediction of advanced materials. In: Proceedings of the 2025 3rd International Conference on Communication Networks and Machine Learning. pp. 269–272 (2025)
34. Yu, C., Velu, A., Vinitsky, E., Gao, J., Wang, Y., Bayen, A., Wu, Y.: The surprising effectiveness of ppo in cooperative multi-agent games. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. NIPS '22, Curran Associates Inc., Red Hook, NY, USA (2022)